Clustering Large Raw DNA Sequencing Datasets by Species of Origin using Signature Features of Genomic Sequence Composition

Tobias Hodges

PhD

University of York

Biology

July 2012

Abstract

The establishment of high-throughput massively-parallel DNA sequencing technology has broadened the scope of metagenomics. The size and complexity of the datasets produced in such studies present considerable challenges.

The aim of this project was to investigate the potential for genomic signature features to be applied to raw high-throughput sequencing reads generated from multi-species samples. Grouping reads according to the genome from which they originate could allow for the study of previously unknown or poorly-understood pathogens, and improve the performance of assembly of genome sequences from these reads.

Genomic signatures were compared to find the best feature or combination for grouping reads by species of origin. A range of datasets were developed to provide an effective basis for such analysis. The performance of a number of clustering methods was also compared. The accuracy of grouping that could be achieved was evaluated, and the effect of such a grouping on the performance of sequence assembly was assessed.

It was found that perfect species-specific grouping of raw sequencing data was outside of the scope of the approaches assessed here, but the enrichment of groups for reads from particular species was achievable. The single greatest obstacle to effective grouping was thought to be the short length of reads produced from current sequencing platforms. The individual assembly of grouped reads was found to produce results similar to those from assembling the dataset as a whole but with a reduction in the time required.

The future of DNA sequencing is bright, with technology advancing at a startling pace, providing improvements in read length, dataset size and experimental run-time. It is hoped that these advancements will prove beneficial to the approaches investigated here, which are likely to remain useful as the size and complexity of datasets increases.

Table of contents

List of Tables and Figures

Chapter 2

Chapter 3

Chapter 4

Chapter 5

Chapter 6

Acknowledgements

I would like to extend my gratitude to the following:

- My supervisors Dr. Peter Ashton and Dr. Neil Boonham, for their invaluable help, guidance and support throughout the project.
- The Food and Environment Research Agency (Fera) and Defra, for funding the project through the Seedcorn Fund, and the Department of Biology for providing space and resources.
- Dr. Naveed Aziz and Dr. Leo Caves for their guidance and interest in both the project and my personal wellbeing.
- Dr. Christian Martin and Prof. Carlos Bastos for providing assistance in implementing the HHSOM and inter-nucleotide distances respectively.
- Prof. Jim Austin and Dr. James Cussens for their interest and advice at an important stage of the project.

Special thanks are further extended to:

- Members of the Novel Methods Team and Glasshouse staff at Fera for their assistance, especially Neil, Ian Adams, Rachel Glover and Jenn Hodgetts for their patience, and to Ummey for carrying out the sequencing. Thanks also to Tom, Sam, Mark and Andy for their help in obtaining the plant pathogens and to Sarah Kendall for her help in obtaining *A. thaliana* seeds.
- Members of the Technology Facility. In particular, Pete for tolerating my incessant visits, Jerry for the tea, Becky for the office space, Naveed and Celina for their help, and anyone who ever brought cake and talked to me about football.
- Julie, Anne and Darren for their help and efficiency throughout.
- My warmest gratitude to all my friends and family, and my parents Marian and Crispin in particular, whose support has been vital.
- Lastly and most importantly to Rachel, for her endless encouragement, support, help and patience. She is a constant source of inspiration, and I am certain that I couldn't have done this without her.

This thesis is dedicated to the memory of Joyce Morris and Berni Strongitharm.

Declaration

- The implementation of the hyperbolic, hierarchically-growing self-organising map (HHSOM) used in Chapter 5 was adapted from MatLab code kindly provided by Christian Martin of Bielefeld University, Germany.
- The *A. thaliana* samples prepared in the work described in Chapter 3 were sequenced on 454 GS FLX by Ummey Hany at the Food and Environment Research Agency (FERA), Sand Hutton, UK.

al
sequences
sequence
sequencing
reads
et
genome
species
assembly
reference
fragments
nucleotide
organisms
comparison
sample
study
platforms
produced
datasets
length
number
process
information
genomic
data
obtained
metagenomic
technologies
alignment-based
strand
sequenced
particular
compared
methods
massively-parallel
analysis
genomes
short
different
dataset
grouping
approach
DNA
high-throughput
nucleotides
single
alignment
present
previously
identified
effective
identification
polymerase
database
environmental
longer
also
based
large
known
whole
allow
research
multiple
order
may
use
used
example
range
many
reaction
individual
time
available
method
homology
features
fluorescent
determine
within
much
case
provide
referred
target
using
required
differences
identify
allows
rRNA
Hoffmann
samples
chip
relative
fragment
genes
regions
associated
composition
coverage
greater
according
Illumina
position
huge
allowing
oligonucleotide
new
read
identity
studies
feature
lengths
years
specific
expression
individuals
prior
applied
set
sampled
Scheibye-Alsing
typically
possible
originate
multi-species
full
size
e.g
made
gene
probe
taken
cost
SOLiD
addition
experiments
pathogen
communities
providing
common
studied
community
throughout
added
metagenomics
group
directly

1

Introduction

Abstract

The size and complexity of high-throughput sequencing datasets, and the short length of the reads produced, has presented the biological research community with a new set of challenges to overcome. One of the greatest difficulties associated with metagenomic sequencing data is the need for effective methods to determine the phylogeny of sequenced samples, and to correctly assemble longer sequences from the reads, requiring an efficient means to distinguish between reads originating from the genomes of the different species in the sampled communities. Alignment-based approaches to sequence comparison scale poorly with large datasets and rely on the prior availability of sequences similar to those under investigation. Alignment-free approaches that utilise features of sequence composition to predict similarity between sequences are much more suited to large datasets and can group reads without the need for any prior information or the presence of sequence from a particular gene.

The aim of this project was to investigate the capability of composition-based methods of sequence comparison for the grouping and separation of reads from massively parallel high-throughput sequencing of multi-species environmental samples, according to the genome from which they originate.

Context

Differences between the genomes of organisms are responsible for the full diversity of life, from the multitude of relatively subtle differences between individuals of a species to the vast and obvious differences between organisms of different species. The compositional differences most commonly studied between the genomes of distinct species are the inconsistencies between the number and sequence of their genes. The accumulation of mutations, either individually within a gene sequence or in the loss/gain or duplication of whole genes, coupled with environmental selective pressures, form the major driving force of evolution. Over time, this process has created the diverse range of species we observe in the world. The study of genome sequences and the differences between them is an important element of modern biological research.

Over the last ten years, genomic research has been altered completely by the introduction of increasingly high-throughput methods of DNA sequencing. The technology currently available enables the elucidation of enough sequence to cover a whole eukaryote genome several times over in a single day. This provides the means to study differences between the genomes of different individuals and species on a scale scarcely imaginable twenty years ago.

These new methods have led to a rapid increase in the number of genome sequences known and the volume of sequencing data readily available, and opened up whole new avenues of research. These include the possibility of quickly and cheaply sequencing the genome of individuals, which may soon render it cost-effective to sequence the genome of every member of a population in order to provide genome-specific healthcare throughout their lifetime (Mardis 2011); the ability to rapidly identify the binding sites of a protein across an entire genome; the capability to study the transcriptome through high-throughput EST sequencing (Nagaraj, Gasser et al. 2007); and the investigation of the combined genome of whole microbial communities sampled directly from their natural environment, a field known as *metagenomics*.

A short summary of DNA sequencing is provided here, including an overview of the advances in sequencing technology that have led to this point, the advances that can be expected in the immediate future, and the avenues of

research that have recently been opened up. Particular attention will be paid to metagenomics, the challenges associated with analysis of the sequencing datasets generated in such studies, and possible ways to overcome these.

DNA sequencing - an overview

Sanger sequencing

Prior to the relatively recent advancements that have transformed DNA sequencing research, this data collection was generally carried out one sequence fragment at a time using the sequencing-by-synthesis method established by Frederick Sanger (Sanger, Nicklen et al. 1977).

In the Sanger method, a polymerase enzyme is used to replicate a cloned fragment of DNA in four separate reactions, initiated by a primer molecule labelled with a radioactive or fluorescent group. In each reaction, all four deoxynucleotides (dNTPs) are present for incorporation into the new strand of DNA, but with a modified version of one of these dNTPs, which has a second deoxygenated group at the 3'-end, also present in a lower concentration. These dideoxynucleotides (ddNTPs) can be incorporated into a novel strand of DNA by the polymerase, but lack the 3'-hydroxyl group that allows another dNTP to be added, thus terminating the strand synthesis. Each of the four reactions contains a different dideoxynucleotide, ddATP, ddCTP, ddGTP or ddTTP, so that the synthesis of strands in each reaction will be terminated at random with the inclusion of a ddNTP of a specific type, producing multiple strands of different lengths in each reaction. Each length produced corresponds to the position of a nucleotide of the modified type present in the reaction.

After synthesis has been allowed to complete, the newly formed double-stranded DNA (dsDNA) is denatured to release single strands. The single-stranded DNA (ssDNA) from each reaction is then analysed by one-dimensional polyacrylamide gel electrophoresis, with the strands from each reaction analysed in a separate lane, and the resultant gel imaged to detect the radioactive or fluorescent groups attached to the 5'-primer on each strand. The sequence of the original fragment is identified by the distance that the different lengths of strands have travelled through the gel in each lane.

Platforms were designed to automate this process using microfluidic capillary systems and a collection of four different fluorescent dyes, corresponding to the four different nucleotides, with synthesised ssDNA flowing past a beam interrogating fluorescence at each length and interpreting the sequence. This automation reduces the time and manpower required for sequencing, and

increases the throughput accordingly, but the output is still limited to ~100 kbp per day per system, rendering the sequencing of a whole genome by such means a costly and lengthy process (Mardis 2011).

'Second-generation' sequencing platforms

The introduction of high-throughput 'massively parallel' sequencing platforms in the last decade led to a dramatic increase in the total length of sequence that could be resolved in a given period of time, and at a stroke removed the limitations associated with the time and resources required for sequence data production. For an illustration of these changes, see 'DNA Sequencing Costs: Data from the NHGRI Large-Scale Genome Sequencing Program', available at: www.genome.gov/sequencingcosts (Wetterstrand 2012). These figures show how the cost of producing 1 Mbp of raw sequence has fallen from ~$3800 in March 2002 to $0.09 in January 2012. About 5 years ago, as the high-throughput 'second generation' sequencing platforms became widely available, the rate of decrease in the cost of sequencing accelerated dramatically. This relatively sudden decrease in the time required to generate large quantities of sequence data has placed a new emphasis on finding fast and effective methods of analysis.

These massively parallel sequencing technologies have been referred to as 'next-generation' sequencing or more recently as 'second-generation' sequencing platforms, to distinguish them from the automated Sanger sequencing platforms that were widely used in the past and the more advanced platforms becoming available now and in the near future. Here, this second generation of sequencing platforms will be referred to as 'high-throughput' or 'massively parallel', with a distinction made where necessary between these and the single-molecule (broadly speaking, third-generation) sequencing platforms.

Massively parallel sequencing technologies are those platforms that produce large volumes of sequence data by performing many thousands of sequencing-by-synthesis reactions at once, with each reaction providing a short read from a fragment of the target sequence. Several different platforms exist, sharing a common central theme in their method of sequencing. These technologies sequence a large target by determining many much smaller fragments of that

sequence, which can then be assembled together to reconstitute the original target sequence.

Three examples of massively parallel sequencing technology, notable for their widespread use in genomic research, are implemented in the Genome Analyser and HiSeq series of systems from *Illumina* (CA, USA), the SOLiD series from *Applied Biosystems*/*Life Technologies* (CA, USA) and the GS FLX series from *454 Life Sciences/Roche* (CT, USA), referred to here as '454 sequencing'. All three of these platform types share many aspects in the methodology used to prepare and sequence samples.

In each case, the DNA of interest is first broken into fragments (e.g. by sonication), which are selected for at an appropriate length (from several hundred basepairs up to around 1 kbp). A series of platform-specific adapter sequences are attached to these sequence fragments, which allow them to be immobilised onto a surface from which they will be interrogated. For the *Illumina* technologies, this is the surface of the sequencing chip, while in the case of SOLiD and 454 sequencing the sequence fragments are annealed to the surface of a bead that will later be placed onto the sequencing chip. Next, many copies of the fixed sequence fragments are produced through a polymerase chain reaction, with beads subjected to this reaction in an emulsion to allow a separate amplification of the sequence on each bead. After this amplification, beads are applied to individual wells on a sequencing chip, or in the case of *Illumina* platforms, the chip holds spots of many copies of the sequence fragments.

The sequence of these fragments is determined by synthesising an opposing strand with the incorporation of nucleotides being signalled in some way to detect and distinguish bases as they are added.

In the case of *Illumina* platforms, each free nucleotide (A, C, G or T) is modified with a fluorescent dye, which prevents another nucleotide from being added to the sequence after incorporation. In each cycle, after free nucleotides have been washed away, the wavelength of fluorescence emitted at each spot on the chip is identified to determine the nucleotides incorporated at each location, before the fluorescent dyes are removed to allow the addition of the next nucleotide to the sequence in the following cycle.

The SOLiD system uses sets of labelled oligonucleotide primers, containing a specific dinucleotide sequence at the 5'-end followed by three non-specific bases, which bind and are ligated to a primer paired with the adaptor sequence on the opposing strand. The binding of an oligonucleotide probe is dependent on the complementarity of the dinucleotide sequence with the opposing strand. After binding, the oligonucleotide probe is ligated to the preceding sequence on the synthesised strand, and fluorescence from the probe detected to determine the added dinucleotide. The fluorescent label is then removed from the oligonucleotide to allow the next probe sequence to be ligated, and the process is repeated for a set number of cycles before the ligated primer and probes are removed. Each probe interrogates a dinucleotide on the opposing strand at 5 bp intervals, with five primers used in cycles that bind the adaptor sequence at single-nucleotide intervals. The use of the multiple primers allows the application of a set of oligonucleotide probes from a different starting point with each primer, such that every nucleotide in the target strand is bound by a probe and interrogated twice.

In 454 sequencing, the addition of nucleotides is identified through the detection of phospho-luminescence instead of fluorescent probes. The beads onto which DNA fragments have been immobilised are applied to wells on the chip. Other beads, onto which are immobilised other materials required for sequencing, are also added to these wells. Nucleotides are introduced to and removed from these wells sequentially (one 'letter' after another) in a set number of cycles, with the release of inorganic pyrophosphate as a nucleotide is added triggering a series of enzymatic reactions that result in the release of light by luciferase, which is detected by a high-resolution charge-coupled device. The repeated sequential application of nucleotides allows the order in which nucleotides are added during replication in each well to be determined, providing a sequence for each fragment (Margulies, Egholm et al. 2005).

More recently, the Ion Torrent series of sequencers, also from *Life Technologies*, has been introduced (Rothberg, Hinz et al. 2011). As with the other massively parallel sequencing platforms described previously, these produce many short reads determined from the individual replication of short fragments of sequence. Nucleotides are again applied sequentially for fragment replication, and the release of H^+ ions (protons) associated with the addition of

nucleotides to a strand is detected by a highly sensitive ion sensor on the base of the sequencing chip. This proton detection identifies the fragments that have been extended at each step.

In the case of Ion Torrent and 454 sequencing, homopolymers (multiple sequential instances of a particular nucleotide) may be incorporated at once if the complementary strand being sequenced contains several of the same base in succession. The addition of a homopolymer is detected as a proportional increase in either brightness (454 sequencing) or ion concentration (Ion Torrent) at the site of the multiple addition. This is precluded in *Illumina* and SOLiD sequencing, where the addition of further nucleotides is blocked until a cleavage step has taken place following the previous addition step.

The similarities between these massively parallel sequencing methods are manifested in similarities between the data produced from them. The product of a sequencing analysis in these cases is a large number of short sequence reads, each corresponding to a single sequencing reaction in the run. Quality information associated with each read is also produced, to provide a measure of the confidence with which nucleotide identity was called at each position.

The length of reads produced depends on the platform used for sequencing. At the shorter end of the spectrum of read lengths are those from SOLiD and *Illumina* platforms, with SOLiD reads ~75 bp in length, older *Illumina* machines producing reads ~30 bp in length, and more recent models ~150 bp. Reads produced in 454 and Ion Torrent sequencing are longer, with Ion Torrent and the early 454 GS FLX technology producing sequences ~250-400 bp in length, and more recently upgraded 454 systems increasing the maximum read length to ~1000 bp (Mardis 2011).

Although the individual reads are very short relative to the size of a genome, considerable coverage can be obtained in a single run by virtue of the huge number of reads that can be produced. A single massively parallel sequencing run produces hundreds of thousands or millions of reads, providing a massive amount of sequencing data for analysis.

Each platform produces reads with a different sequencing error profile. For example, SOLiD systems have high accuracy in terms of nucleotide identity as each position in the target sequence is interrogated twice, although differences

in intensity of emitted fluorescence can make base-calling more difficult in later cycles of a run. As described previously, Ion Torrent and 454 sequencing can introduce multiple copies of the same nucleotide into a sequence in a single cycle, with errors in identifying the true length of these homopolymers more common in these platforms. The signal observed from the incorporation of a homopolymer scales with the number of nucleotides included, but imprecision in this proportionality can result in an erroneous call of the true length of the homopolymer, especially where this length is larger than just a few nucleotides (Le and Durbin 2011). The position of a base in a fragment, and the nucleotides that surround it have both been shown to influence the likelihood of an erroneous nucleotide identity call (Gilles, Meglecz et al. 2011; Nakamura, Oshima et al. 2011).

Although the huge yield of high-throughput platforms has facilitated a dramatic decrease in the cost of sequencing, and a huge increase in the rate at which sequencing can be completed, the size of the datasets and the short length of reads produced has introduced a new set of challenges. Where the pace of genomic research was previously limited by the time required to determine the sequence of interest, the limiting factor is now the time taken to effectively store, extract and analyse the sequencing data produced from these methods (Mardis 2011; Scholz, Lo et al. 2012).

Sequence assembly

The short length of sequencing reads constitutes one of the greatest obstacles to effective analysis of genomic sequences targeted with massively parallel approaches (Miller, Koren et al. 2010; Mardis 2011). Longer sequences can be reconstituted from the short reads through a process known as *assembly*, where sections of identical sequence are used to identify overlaps between reads and subsequently join them together. The shorter the reads produced, the more difficult it is to identify these overlaps, and the more reads are required to produce a given coverage of a target sequence (Scheibye-Alsing, Hoffmann et al. 2009; Miller, Koren et al. 2010).

Massively parallel sequencing of a genome or large section of sequence is usually performed using the ‘whole genome shotgun’ (WGS) approach. In WGS sequencing, the target genome is first sheared into small fragments and

separated based on size (typically by gel electrophoresis) before an appropriate size range is selected for sequencing. The fragments are the applied directly to sequencing, without any prior tagging or cloning to preserve an idea of the order in which they existed in the original target sequence. The sequencing reads produced from these fragments are then assembled into longer, contiguous sequences known as *contigs* (Scheibye-Alsing, Hoffmann et al. 2009; Miller, Koren et al. 2010).

This process is computationally very expensive for large datasets, as it requires each read to be compared with every other in the dataset. The computational time required for sequence assembly scales with the square of the number of sequences (Vinga 2003).

The processes behind sequence assembly from high-throughput sequencing data are discussed in more detail later.

The alternative to WGS for whole genome sequencing is to use a ‘tiled’ approach, whereby a library of cloned longer sections of the genome is prepared. These cloned sections are typically generated by treatment of the genome with a restriction enzyme, and are sequenced and assembled individually. The longer genome sections are cloned into a sequencing vector such as an artificial bacterial chromosome (BAC), which can be identified by a fingerprint sequence to determine the order of the reassembled sequences and reconstitute the genome sequence (Scheibye-Alsing, Hoffmann et al. 2009).

The WGS approach is much more widely used in modern research, owing to the simplicity of sample preparation, and the extensive genome coverage that can be obtained in a minimal number of individual experiments performed on high-throughput sequencing platforms. This speed and simplicity comes at the cost of a greater burden of sequence assembly, created by the production of so many sequencing reads with no prior information regarding their location, orientation, or order (Scheibye-Alsing, Hoffmann et al. 2009; Miller, Koren et al. 2010).

Generally speaking, two approaches can be taken to assembly. One method is to use a reference sequence as a scaffold on which to base the assembly. This reference could be a genome sequence already obtained from an individual of the sampled species, or a homologous sequence from a closely related

species. Reads are aligned to the reference genome to determine their relative locations, allowing gaps in coverage and structural rearrangements in the genome to be identified. This guided assembly is particularly helpful in the sequencing of larger, more complex genomes, which contain a greater degree of repetitive sequence. Where a region of repetition is longer than the length of the sequencing reads produced, the assembly of reads covering this region becomes complicated as tangles are formed between overlapping sequences. The use of a reference scaffold helps to simplify the process of untangling these reads. An assembly guided by such a reference sequence is also less limited by the effects of the length of particularly short reads produced in sequencing with *Illumina* and SOLiD platforms, as less emphasis is placed on finding significant overlap between reads (Cronn, Liston et al. 2008).

The second approach is to assemble the reads without any previously obtained template information, a process known as *de novo* assembly (Paszkiewicz and Studholme 2010). Where no closely related reference genome is available, as is the case for the majority of organisms, this is the only option for sequence assembly. This approach has been successful when applied to reads at the longer end of the spectrum, and *de novo* genome assembly is becoming ever more viable. However, for shorter reads, and those generated from more complex genomes, the more directed method using a reference template makes extensive assembly significantly easier to achieve (Paszkiewicz and Studholme 2010).

The success of assembly can be improved by the generation of paired-end or mate-paired reads in sequencing. These are reads produced simultaneously from opposite ends of the same sequence fragment, which include a tag sequence that allows the pairs to be identified and considered in tandem during assembly (Scheibye-Alsing, Hoffmann et al. 2009).

When sequence fragments are prepared for sequencing they are selected for by size, which provides an estimate for the distance that should exist between pairs of reads when assembling them after sequencing. This prior knowledge of the rough spacing between two reads can simplify the process of assembly considerably. This information can be used to provide a more reliable prediction of where these reads belong in the assembly, and also identify structural rearrangements in a sequenced genome when compared to the reference. For

example, a pair of reads produced from each end of a short fragment but mapped to distant points on the reference genome would indicate the removal/ relocation of a large section of the reference sequence between these reads in the target sequence. Similarly, the reversal of a section of sequence could be identified by the mapping of a pair of reads in the opposite direction to that observed in the reference (Scheibye-Alsing, Hoffmann et al. 2009).

The greatest obstacle to fast and effective sequence assembly is the short length and sheer number of sequencing reads produced by current sequencing experiments (Scheibye-Alsing, Hoffmann et al. 2009). The need for every read to be compared with all the others in a dataset, and for overlapping sections of sequence to be found within the short lengths of these reads, places a considerable computational burden on the process. As massively parallel sequencing technologies improve, the typical length of reads produced is increasing, but so too is the typical size of a single sequencing dataset, providing greater coverage in a single experiment but further increasing the burden on the assembly process.

‘Third-generation’ sequencing platforms

It is predicted that another generation of platforms, often referred to as the ‘single-molecule’ sequencers, will become widely available in the near future. These technologies do not rely on amplification of DNA fragments before sequencing, as in massively parallel sequencing, instead determining the sequence of a single copy of the target. The first of these platforms to become available is the PacBio RS from *Pacific Biosciences* (CA, USA). In this case, sequencing occurs via polymerase enzymes immobilised in tiny holes in a foil film on a glass slide.

These polymerase molecules bind target DNA strands and replicate them using nucleotides labelled with type-specific fluorescent dyes. The extremely small width of the holes in the foil prevents light at the excitation wavelength of the fluorescent labels from penetrating the reaction solution much beyond the immobilised polymerase. As a labelled nucleotide is incorporated into the extending strand by the enzyme, it is held near the surface of the glass where it can be interrogated by the light beam. The nucleotide being added is identified by the wavelength of fluorescence emitted. This approach allows the sequence

of fragments to be determined as quickly as the polymerase can replicate them, returning sequencing reads at a faster rate and at greater lengths than those produced in massively parallel sequencing (Eid, Fehr et al. 2009).

Another example of single molecule sequencing is the 'nanopore' approach described in (Clarke, Wu et al. 2009) and currently under development by *Oxford Nanopore Technologies* (Oxford, UK). Nanopore sequencing is based on the use of protein pores that bind and transport DNA through a membrane. Detection of nucleotides is achieved by measuring tiny changes in electric capacitance across the pore. Theoretically, this methodology should allow the direct sequencing of DNA molecules of any length as a whole, at the high speed at which the molecule passes through the pore. The technology is likely to become widely available over the next few years, and is predicted to instigate another revolution in the speed and convenience of genome sequencing, shifting the focus and the challenges of genomics research further towards the handling and analysis of the data produced.

Genome sequencing

The principal aim of many sequencing experiments is the determination of the genome sequence(s) of an individual, a species, or group of species. This information acts as a gateway to a greater understanding of many aspects of life, and forms a common theme throughout many different fields of research in the biological sciences.

For example, a researcher interested in the genetic traits behind a heritable disorder in humans might aim to establish the differences between the genomes of an affected and a 'normal' individual in order to develop targeted treatments. A plant biologist interested in increasing yields in a particular crop species might wish to obtain the full genome sequence of this plant, to identify genes in this sequence, and use this information in conjunction with results from other experiments (such as microarray analysis) to identify those genes linked to the yield of the plants and the differences between sequences that might be responsible for this phenotype.

Genome sequencing is also an integral step in the characterisation of pathogens, for example in the identification of mutations between bacterial strains to understand differences in pathogenicity (Harrison, Dyer et al. 2005).

The recent improvements in sequencing have made it possible to include genome sequence information as part of a wider research project with relative speed and ease, and at low cost.

EST sequencing

The introduction of high-throughput sequencing technologies has also improved the scope for transcriptomic investigations, studying the profile of expression in coding regions of the genome. Other technologies, in particular microarray analysis platforms, have contributed to this field of study, allowing for the complete comparison of expression profiles in different tissues. This information can be particularly useful in the characterisation of disease, for example in comparing the genome-wide expression profiles of cancerous tissue with an equivalent healthy sample (Volinia, Calin et al. 2006), or in studying the knock-on effects of variation in the levels of expression of a single gene product (e.g. Branney, Faas et al. 2009).

Expressed sequence tags (ESTs) are reads sequenced from cDNA prepared from a sample, providing insight into the expression profile of the sample based on the mRNA present. Rather than determining the full cDNA sequence, ESTs are read from either end of the transcript and can be used with a reference genome to identify their original position (Nagaraj, Gasser et al. 2007; Scheibye-Alsing, Hoffmann et al. 2009). The high-throughput nature of modern sequencing methods allows for a picture of the full expression profile (the transcriptome) of a sample to be built, based on the assumption that the number of ESTs sequenced for a gene is directly proportional to the level of expression of that gene (after copy number variations have been considered).

Amplicon sequencing

High-throughput sequencing has also allowed for the identification and study of polymorphisms between the genomes of individuals of the same, or closely related, species. Primer sets are used in PCR to amplify specific target regions of sequence from each genome into ‘amplicons’, which are then sequenced. By sequencing amplicons on a massively parallel platform, the amplicon sequences are covered at great depth, allowing for even rare polymorphisms to be identified when the sequences are compared (Rosani, Varotto et al. 2011).

SNP analysis

The large-scale identification of single nucleotide polymorphisms (SNPs) throughout genomic sequences has been made possible by high-throughput sequencing. The process requires that sequencing be performed at high depth, so that each potential polymorphism site is covered multiple times. If this criterion is fulfilled, the reads may be mapped to a reference genome sequence and positions at which the nucleotide sequence differs can be identified as candidate SNP sites. Many methods have been developed that determine the confidence with which an SNP can be called at a particular site (e.g. McKenna, Hanna et al. 2010; Le and Durbin 2011), but as a rule the more consistently that a polymorphism is detected (i.e. the more it appears in the sequencing reads covering that site), the greater confidence can be had in the assignment of a SNP at that position (Nielsen, Paul et al. 2011).

SNP calling is an important field in the study of population genomics, as it allows for the genetic differences between individuals of a species to be identified and studied (Nielsen, Paul et al. 2011). It forms a key part of studies such as the 1000 Genomes Project (www.1000genomes.org), which aim to elucidate the key genomic differences between individuals and populations.

Metagenomics and sequencing of multi-species samples

The capacity for massively parallel sequencing to produce a huge number of reads at much higher speed and lower cost has made possible the simultaneous investigation of the genomics of complex environmental communities, a field of study known as metagenomics (Handelsman, Rondon et al. 1998). Whereas experiments aimed at sequencing the genome of a single species rely on the isolation and preparation of a pure sample, metagenomic datasets are typically sequenced directly from the genetic material of many species, extracted from an environmental sample.

A metagenome contains sequences from the range of organisms present in the sample (under-representation of minority organisms notwithstanding), and can provide information about the make-up of communities that are difficult to study by other means. It is thought that less than 1% of all bacterial species can be successfully cultured in the laboratory, so in most cases this approach of directly sequencing species offers the only opportunity to study the genome (Torsvik, Sørheim et al. 1996; Rappe and Giovannoni 2003).

Metagenomic datasets have been generated and studied from a wide range of environments, including microbial communities in oceanic water samples (Venter, Remington et al. 2004; Rusch, Halpern et al. 2007), various soil samples (Rondon, August et al. 2000; Voget, Leggewie et al. 2003; Tringe, von Mering et al. 2005), the interior and exterior of the human body (Gill, Pop et al. 2006; Grice, Kong et al. 2009; Qin, Li et al. 2010), the respiratory system of aquatic mammals (Lima, Rogers et al. 2012), and even the scrapings from a car windscreen (Kosakovsky Pond, Wadhawan et al. 2009).

Metagenomic studies predate second-generation sequencing methods, but the scale on which such samples could be studied was considerably smaller than is possible today (Handelsman, Rondon et al. 1998; Rondon, August et al. 2000). Initially, metagenomic analysis of a microbial community relied on the isolation, cloning, sequencing and comparison of 16S rRNA sequences, to identify and characterise the phylogeny of species present in a sample.

The ribosome is present in every form of cellular life, and the genes coding for the proteins and rRNA molecules from which it is constructed are well conserved between species, with well-established regions of variability. As a

common reference point between genomes, the sequences of these rRNAs - the 16S rRNA for prokaryotic species, and 18S rRNA for eukaryotes - are selectively reverse-transcribed and amplified via PCR, using sets of generic primers, to allow a comparison of organisms (Lane, Pace et al. 1985; Pace 1997; Venter, Remington et al. 2004).

Using this method, a phylogeny of the organisms in the sample can be built up based on variations in rRNA reads, compared to each other, or to a reference database of sequences to provide a wider context to the sample. Sequences within a sample can be grouped together into ‘phylotypes’, based upon a cutoff threshold of sequence identity, and such a grouping provides a measure of the diversity within a sample (Pace 1997). However, it has been found that typically less than 0.1% of sequences in a metagenomic dataset can be assigned to ribosomal RNA genes (McHardy and Rigoutsos 2007), so the likelihood of gaining a truly meaningful and comprehensive insight into the species contained within a metagenome using this method is very small. Biases in the amplification of rRNA sequences can also lead to misrepresentation of the true number and proportions of different species within a sample (Baker, Tyson et al. 2006; Eisen 2007).

Metagenomic analysis has been made considerably more viable by the introduction of high-throughput sequencing platforms. The massively parallel nature of most current sequencing platforms is particularly well-suited, providing a large sample of sequence fragments from across the range of organisms in a community through shotgun sequencing of the sampled material.

By sequencing fragments at random from a sample of a community, rather than targeting a specific region of sequence as in the comparison of rRNA genes, biases in the representation of species can be avoided, and much more information can be gained about the community. This sequence information can be used to investigate the wider range of genes present in a sample, and introduces the possibility of constructing full genome sequences of the organisms present (Tyson, Chapman et al. 2004; Chen and Pachter 2005). The 16S rRNA sequence is still widely used to study the diversity of microbial communities (Tringe and Hugenholtz 2008), but the range of metagenomic studies facilitated by high-throughput sequencing exceeds this. Modern research has come to include metatranscriptomic and metaproteomic studies to

complement metagenomics and provide an even greater insight into communities that remained practically unstudied until very recently (Simon and Daniel 2011).

The unculturability of the vast majority of microbial organisms has led to great bias in the representation of species in biological sequence databases, towards those organisms that have historically proven easiest to study (Huson, Richter et al. 2009). Metagenomics has provided the means to begin redressing this balance, by allowing the study of species and communities that were previously inaccessible for analysis, and the elucidation of genome sequence from these species.

As a consequence of these changes, metagenomics has been a rapidly growing field over recent years, and a range of databases and resources have been established to allow easier storage, access, browsing and analysis of these datasets (Teeling, Waldmann et al. 2004; Seshadri, Kravitz et al. 2007; Richter, Ott et al. 2008; Schloss and Handelsman 2008; Lorenzi, Hoover et al. 2011; Markowitz, Chen et al. 2012). For example, the MG-RAST database (Meyer, Paarmann et al. 2008) now contains >10,000 metagenomes for investigation and annotation. As sequencing technologies improve, and methods of data storage and analysis continue to develop, the number, range and size of metagenomic studies seems likely to continue to grow.

To gain a meaningful understanding of sampled communities from metagenomic shotgun sequencing data, the reads produced must be compared in some way. This allows for the phylogeny of the sample to be investigated, function to be assigned to coding sequences obtained, and potentially for genomes to be reconstructed individually from the mixture of fragments in the metagenome.

Example Metagenomic Projects and Datasets

The Sorcerer II Global Ocean Sampling Project

Inspired by the voyage of the HMS Challenger, the vessel lead by Charles Wyville Thomson in the 1870s to dredge the ocean floors of the globe in order to study ad describe previously unidentified fauna, J. Craig Venter established the Global Ocean Sampling Project to perform a similar survey of the microbial populations of oceanic waters. Following a similar route to that of the Challenger, sampling of surface waters was carried out by Venter and his crew aboard his personal yacht, the Sorcerer II.

Following a pilot study of the Sargasso Sea, the full expedition began collecting samples of surface water at intervals along the route from the east of Canada, through the Panama Canal, to the Eastern Tropical Pacific. These samples were taken from coastal and open waters, as well as some freshwater samples for comparison. The aim of the project was to uncover the microbial diversity of these waters, and to compare these samples in order to better understand how the microbial populations and communities change across a range of locations and conditions.

After samples were filtered to isolate the microbial organisms of interest, environmental shotgun sequencing (Sanger method) produced a dataset of ~6.4 million sequences of combined length ~5.9 Gbp (5,900 Mbp). The sheer scale and complexity of this dataset served as an introduction to the difficulties that researchers would face in the analysis of metagenomes (Rusch, Halpern et al. 2007).

In the years since the data was first published, a number of investigations have been carried out, each with the aim of studying and understanding a particular aspect of the communities sampled during the expedition.

An early study identified the ribosomal 16S rRNA sequences present in the data after assembly into contiguous sequences, and used these fragments to analyse the taxonomic diversity within the whole oceanic metagenome (Rusch, Halpern et al. 2007). The scale of the dataset lead the researchers to develop novel methods for assembling contiguous sequences from complicated and noisy data, comparing the similarity of samples and visualising information about these samples, from sequence coverage and identity to metadata data

such as sampling time and location, and details of the sampled environment.

In tandem with this initial study of organismal diversity, the diversity of protein sequences within the sampled communities was also investigated (Yooseph, Sutton et al. 2007). After open reading frames (ORFs, sections of DNA sequence between a start and stop codon in the translated sequence) were identified in all six potential frames of translation within contigs assembled from the data, these sequences were compared to each other and all publicly available protein sequences in the NCBI, TIGR and Ensembl databases. This protein BLAST searching required over 1M CPU hours to complete (Yooseph, Sutton et al. 2007). The alignment of all of these sequences allowed a clustering to be performed, grouping similar sequences to represent protein families. This novel approach to clustering and filtering of predicted protein sequences on such a large scale allowed for novel protein families to be predicted, and for additional information to be uncovered relating to those families already known at the time.

The novel methods of analysis and visualisation for metagenomic data developed and introduced as means of investigating data on such a huge scale were combined into an online resource and database, CAMERA (Seshadri, Kravitz et al. 2007), that is updated regularly with new tools for metagenome analyses such as protein prediction and functional annotation and acts as a repository for other environmental datasets. The full dataset produced from the filtered water samples was also made publicly available for investigation by the wider scientific community.

A similar study of the global ocean sampling metagenome data focussed on viruses sequenced within the samples (Williamson, Rusch et al. 2008), using a conservative system of sequence classification against the NCBI database of non-redundant protein sequences (*nr*) to describe the diversity of viruses within and between the different samples taken during the expedition. The same group of researchers have recently investigated water samples taken in another round of sampling from the Indian Ocean, using a combination of Sanger and massively parallel sequencing of size-filtered samples to predict the virus diversity within the samples, and the predicted functions and taxonomic origin of viral protein sequences (Williamson, Allen et al. 2012).

Many other studies have been carried out on the data, including efforts to identify trends in environmental conditions associated with the makeup of sample populations at different locations (Yilmaz, Iversen et al. 2012), and, similarly but conversely, the associations between microbial community/gene diversity and prevalence and environmental indicators such as dissolved iron (Desai, Desai et al. 2012; Toulza, Tagliabue et al. 2012).

The early studies of the oceanic metagenome dataset analysed sequences assembled from the original Sanger sequencing reads pooled from all samples taken as part of the project. The microbial diversity between the individual samples taken throughout the Sorcerer II expedition was also studied, with 16S rRNA sequences within each sample being identified by sequence similarity with a reference sequence from *E. coli* (Biers, Sun et al. 2009). These 16S sequences were found to constitute only ~0.24% of the metagenome. The 16S reads were clustered by sequence similarity and combined with information regarding gene counts within the data to compare the diversity between samples and to predict the characteristics of the 'average' prokaryotic genome present within the dataset.

In each of these studies, the methods used to compare sequences and to group them taxonomically were largely reliant on alignment between sequences and reference databases. The demands of such an alignment-based approach required the harnessing of huge computational resources in order to bring the time requirement down to manageable levels (Yooseph, Sutton et al. 2007), especially in early studies where more advanced assembly software was not yet available. However, even using more recent assembly software, as in (Williamson, Allen et al. 2012), to assemble such huge volumes of sequencing data is a demanding task, especially as the number of sequences that can be obtained in a single experiment is ever-increasing, while the length of the individual reads is typically shorter than the Sanger sequences generated in the original Ocean Sampling project.

The Human Microbiome Project

Established in 2007, the Human Microbiome Project (HMP) is an ongoing global initiative funded to the tune of $150M by the National Institutes for Health (NIH) of the USA. Intended to complement the sequencing of the human genome, the

aims of the HMP are described as:

'*(1) to take advantage of new, high-throughput technologies to characterize the human microbiome more fully by studying samples from multiple body sites from each of at least 250 "normal" volunteers; (2) to determine whether there are associations between changes in the microbiome and health/disease by studying several different medical conditions; and (3) to provide both a standardized data resource and new technological approaches to enable such studies to be undertaken broadly in the scientific community. The ethical, legal, and social implications of such research are being systematically studied as well. The ultimate objective of the HMP is to demonstrate that there are opportunities to improve human health through monitoring or manipulation of the human microbiome.*' (Peterson, Garges et al. 2009).

At the time of writing, the project's website lists >200 publications relating to the HMP (http://commonfund.nih.gov/hmp/publications.aspx, 25/11/12), pertaining to the processes and ethics of microbiome sample collection, production of both 16S rRNA and shotgun metagenomic sequencing data, and the subsequent storage, availability and analysis of this data. The output of new findings and techniques of analysis developed to address the huge volume of data associated with the project looks set to continue.

To date, these publications include descriptions of metagenomic analyses providing insight into the differences between tissues and between individuals (e.g. Faust, Sathirapongsasuti et al. 2012), differences between healthy and diseased tissue (e.g. Pushalkar, Mane et al. 2011; e.g. Liu, Faller et al. 2012), and the metabolic pathways of microbial communities within the body (e.g. Cantarel, Lombard et al. 2012). Also included are more general discussions of the analytical and ethical issues faced in such a significant project (McGuire, Achenbaum et al. 2012; Segata, Waldron et al. 2012).

Both the Global Ocean Sampling and Human Microbiome projects have produced a deluge of sequence data, which is the subject of a large proportion of the total metagenomics research currently carried out, and the source of many new bioinformatic tools for the analysis of such data (e.g. Seshadri, Kravitz et al. 2007; Markowitz, Chen et al. 2012; Wang, Ye et al. 2012).

One of the major challenges associated with this kind of data, containing many sequences from many genomes of species that are often closely related, is in the separation of these sequences according to the species such that the individual genes or genomes can be studied and compared within the context of the sampled community.

In both of these major metagenomics projects, the majority of work has revolved around alignment-based sequence comparison, either in the form of 16S ribosomal RNA gene sequence comparison (e.g. Huse, Ye et al. 2012; Li, Bihan et al. 2012), or other marker genes (Segata, Waldron et al. 2012). The methods used to carry out these comparisons have improved greatly over the years, both in terms of speed and accuracy, and in the ever-increasing body of reference sequences that are available. However, as discussed in detail in the next section, this alignment-based approach, comparing either 16S rRNA or other marker gene sequences, has it's disadvantages.

Low-complexity metagenomes

While the large-scale Human Microbiome and Ocean Sampling projects involve the investigation of complex and diverse communities of many different species varying in their prevalence, the metagenomic approach can also be applied to the study of simpler communities. These communities, consisting of a smaller number of different species overall, and often dominated by one species in considerably greater abundance, are particularly relevant to this project as they more closely resemble the plant-host systems that are of most interest here (see Project Overview for more detail).

Several examples of such investigations exist in the literature, with the most extensively studied example being the metagenome of biofilms sampled from acid mine drainage in North America (Tyson, Chapman et al. 2004). Other examples include several studies of symbiosis between animals and bacteria (e.g. Woyke, Teeling et al. 2006; Wu, Daugherty et al. 2006), and the analysis of sludge produced in the removal of inorganic phosphate from waste water (Martin, Ivanova et al. 2006). These low-complexity metagenomes offer the possibility of assembling whole microbial genomes from reads shotgun sequenced from the environmental sample (Tyson, Chapman et al. 2004; Martin, Ivanova et al. 2006).

As with the larger and more complex metagenomes discussed previously, such low-complexity datasets are often studied by alignment-based methods, comparing 16S rRNA and other gene sequences (e.g. Tyson, Chapman et al. 2004; e.g. Martin, Ivanova et al. 2006; e.g. Wu, Daugherty et al. 2006; e.g. Schloss and Handelsman 2008). However, the use of alignment-free comparison has become more widespread recently, with researchers using compositional features of sequences to group metagenomic sequences based on a prediction of shared taxonomic origin.

One example of this type of analysis is the use of tetranucleotide frequency distributions of sequences, combined with neural network clustering to group the acid mine drainage dataset (Abe, Sugawara et al. 2006; Dick, Andersson et al. 2009). Oligonucleotide frequencies have been used alongside alignment-based techniques to separate sequences from mixed datasets, including those from host and symbiote genomes sequenced together (Chatterji, Yamazaki et al. 2008). Such oligonucleotide frequency patterns have also been used alongside a measure of the net read depth of sequences assembled from a shotgun metagenomic dataset, to group assembled sequences belonging to the different bacterial species in symbiosis with a marine worm (Woyke, Teeling et al. 2006), and a combination of read depth and GC content comparison was used to isolate sequences from different species in the original study of the acid mine drainage biofilm metagenome (Tyson, Chapman et al. 2004). A similar approach, identifying groups of assembled contigs based on GC content and read depth, has been used more recently as part of a mechanism for improving the simultaneous assembly of bacterial symbiont and nematode host genome sequences (Kumar and Blaxter 2011).

Several studies have also been published aiming to discover novel pathogens through shotgun sequencing of environmental samples. These experiments have largely focussed on prospecting for viral pathogens.

Two such studies have been carried out in honey bee populations in recent years. Motivated by the phenomenon known as colony collapse disorder, which has been associated with rapid and accelerated loss of bee colonies throughout the world, much research has been undertaken in an attempt to uncover the underlying cause(s). In one case, whole genome shotgun sequencing of the mite *Varroa destructor* uncovered genomic sequence of what the researchers

believe to be a novel bacterial and novel Baculovirus species (Cornman, Schatz et al. 2010). These microbial sequences were discovered by plotting read depth and GC content of assembled contigs, as described previously for other low-complexity metagenomes.

A second study into honey bee pathogens, was based on sequencing of samples collected over a 10-month period from several colonies transported between multiple locations around the USA. Samples collected over the course of the study were analysed by microarray and qPCR for detection of known pathogens, and by shotgun sequencing at great depth for novel pathogens. This sequencing of one sample uncovered sequence predicted to belong to four new viruses amongst the data produced for other species and pathogens known to be present in the sample (Runckel, Flenniken et al. 2011). The sequences were assigned to these novel viruses after screening of assembled contigs against a database of known honey bee pathogens had failed to produce hits of acceptable strength. Any contigs that remained unassigned after this screening were mapped to a more comprehensive database and extended by further assembly with the complete dataset.

Many other similar studies have uncovered previously undescribed viruses, for example in samples from pig and bat (Sachsenroder, Twardziok et al. 2012; Tse, Tsang et al. 2012), with a similar approach taken to sequence analysis.

As with several of the other studies mentioned here, this approach relies upon the availability of suitable reference data with which to compare the sequences of interest within a sample.

Methods of sequence comparison

Alignment-based sequence comparison

As discussed previously, one of the greatest challenges facing the bioinformatics community is the need for effective ways to analyse the huge and complex datasets generated in sequencing experiments using high-throughput, massively parallel platforms. This problem is compounded where the sequenced sample consists of many different species in varying proportions (Scholz, Lo et al. 2012).

Where a sequenced sample is known or predicted to contain a number of different species, it can be beneficial to group the sequences in the dataset according to a prediction of the genome from which they originate, a process often referred to as *binning*. Such a grouping can provide an insight into the phylogeny and complexity of a sample, through the investigation of the number, identity and relatedness of organisms represented by the grouped genomic sequences (Teeling, Meyerdierks et al. 2004).

Grouping sequencing reads that originate from the same genome may also be useful as a technique to reduce the time required for assembly of individual genomes from the mixed data, and decrease the likelihood of erroneous sequence assembly incorporating reads from multiple genomes.

In order to predict a shared origin between reads, the reads must be compared in some way, using only the primary sequence information available in the sequencing dataset. Broadly, methods for the comparison of sequences fall into two categories: those based on alignment of sequences to assess similarity between the specific pattern of nucleotides in the sequence, and those based on the composition of the sequence, which aim to identify more general shared characteristics between sequences (Vinga 2003).

The alignment-based approach is most commonly taken when measuring the similarity between DNA sequences. This type of comparison forms the basis for BLAST (Altschul, Gish et al. 1990), which provides the means for searching a database of reference nucleotide or protein sequences, to determine homology and identify shared regions of sequence and common domains between a query and reference. With a range of tools and a huge database of submitted sequences hosted and publicly available for access through the National Center

for Biotechnology Information (NCBI, USA), the use of BLAST for measuring and studying the similarities between DNA sequences has become a familiar resource for the biological science community.

All alignment methods operate on a common central process of pairwise comparison of the bases in each sequence, finding the alignment of the sequences with the highest score computed from a set of parameters describing the reward for a correct pairwise match and the costs of allowing a mismatch or the introduction of a gap. This alignment-scoring methodology was first established by Waterman and Smith (Waterman and Smith 1981), and is commonly referred to as Smith-Waterman scoring.

As mentioned previously, the process of sequence assembly from short reads obtained from sequencing experiments is based on alignment to find overlaps between reads. DNA sequence alignment also plays an important role in the prediction of gene function, species and gene evolution, and the prediction of relatedness between organisms. Such predictions made through sequence alignment are based on significant levels of homology between sequences.

As DNA sequences evolve, the primary nucleotide sequence changes through point mutations, insertions and deletions (including horizontal gene transfer, gene and chromosome duplications and increases/decreases in repetitive regions), and the splitting and merging of chromosomes. Whether these alterations to a genome are conserved through multiple generations and allowed to spread throughout the population depends on the consequences of these mutations to the sequence.

A mutation in a vital section of a coding sequence is less likely to be maintained through multiple generations as it would be likely to have a deleterious effect on cellular function, and so these important regions of sequence - whole genes or smaller sections most closely related to the function or regulation of a gene - are conserved across generations and between species. This conservation of sequence leads to the homology observed between sequences of different genes (where a particular domain may be shared between many proteins), individuals and organisms, and forms the basis of comparison by sequence alignment.

Alignment is also a very effective method for analysis of longer sequences,

where local alignments can identify regions of homology amongst sequences that are otherwise divergent. It is an excellent approach to take in the study of small datasets of sequences, where primary sequence similarity is of particular interest and a good database of reference sequences for alignment is available.

This prior knowledge of references is vital for alignment-based investigation of sequences: without a good basis for comparison, fewer homologous alignments may be found and the strength of conclusions that may be drawn will be limited. The easy access and availability of huge sequence databases like those maintained by the NCBI is enormously empowering for alignment-based analysis of new sequences, while the ever-increasing amount of sequence data under production ensures that these resources will only become more informative in the coming years.

Despite it's popularity, the alignment-based approach to sequence comparison can have disadvantages, especially in the age of massively parallel DNA sequencing. While sequence alignment is a hugely beneficial tool for the study of individual primary sequences in the context of a reference dataset, these approaches do not scale well and are unsuited to large datasets of short sequences. The reliance on a reference database for predicting the phylogeny and function of sequencing reads prevents such methods from providing insight into sequences originating from poorly understood branches of the Tree of Life, such as are commonly produced in metagenomic studies.

The size and complexity of metagenomic datasets makes these datasets particularly difficult to handle using alignment-based methods, and finding effective and efficient methods of analysis is a great challenge (Eisen 2007). The time required for alignment-based comparison of a set of sequences scales with the square of the number of sequences in the set. For a typical high-throughput sequencing dataset containing 10^5 - 10^7 individual reads, this exponential scaling is prohibitive (Vinga 2003).

The requirement for a database of reference sequences with which to compare either rRNA or shotgun-sequenced genomic reads also presents a set of obstacles to effective analysis of environmental, multi-species samples. A sequence compared to a reference database can only be characterised according to the sequences contained within that database. As such, the scope

for comparison is limited by the availability of references drawn from a wide taxonomic range.

While modern sequence databases, such as the *nt*/*nr* database hosted and maintained by the NCBI, contain vast numbers of sequences obtained from a wide range of genomes, the distribution of available sequence information is far from uniform across phylogenies (Huson, Richter et al. 2009). This varied representation in available reference sequences could introduce bias into the classification of sequences, based on homology found in an alignment-based comparison. It also complicates the issue of when and how a sampled sequence can be classified as sufficiently dissimilar to any sequence already known.

Without the use of a reference database, alignment could be used to group sequences in a metagenomic dataset based on homology between sequencing reads. However, with minority organisms in the sample likely to be represented sparsely and at low genome coverage, this approach is unlikely to result in effective grouping.

When considered alongside the computational challenges associated with the alignment-based analysis of large numbers of sequences, the limitations of such an approach render it far from ideal for the study of environmental, metagenomic datasets.

Composition-based sequence comparison

An alternative approach to the comparison of sequences is the use of features characterising the nucleotide composition of sequences as a basis for the identification of similarities and dissimilarities between them. This approach provides a much more suitable method of analysis for large sequencing datasets obtained from multi-species environmental samples,

The features used in such a comparison represent patterns within the primary nucleotide sequence: trends in the composition of a molecule that are conserved throughout the genome. A simple and well-known example of such a feature is the GC-content of a genome, a property that has been used for the characterisation and comparison of genomic sequences for many years.

The establishment of the relative frequency distribution of oligonucleotide ‘words’ as an identifying characteristic of DNA sequences by Karlin and

Ladunga (1994), led to the coining of the phrase ‘genomic signature’ to describe these features (Karlin and Burge 1995). Oligonucleotide relative frequency distributions are the most well-established of several genomic signature features, discussed in more detail later, which have all been shown to allow differentiation between sequence fragments originating from different genomes.

For a feature of sequence composition to qualify as a genomic signature, the degree of variation observed between feature values of sequences originating from the same genome must be consistently smaller than that observed between the values of sequences from different species. The feature signature should be consistent throughout the whole genome, such that a feature value obtained from a short length of sequence (e.g. 1-10 kbp), is similar to the value obtained from a longer length of sequence (e.g. 1 Mbp), and to the value obtained from a whole chromosome or the entire genome. As such, the grouping of sequences according to their origin is not based on regions of specific nucleotide sequence identity but instead on the identification of a common pattern running through their composition.

Using such features to compare a set of sequences makes it possible to identify similarity between sequences without the need for a reference database and without relying on sequence homology. Each sequence can be considered using a single value or distribution calculated from its composition, allowing a much more straightforward grouping of the data that typically scales in proportion with the number of sequences as opposed to the square of this number as in alignment-based comparison (Vinga 2003). Such a composition-based approach allows for sequencing reads to be grouped together regardless of whether the genome from which they originate has been studied previously or not, removing the need for a reference database for comparison, and in the absence of any overlapping sequence homology.

This alignment-free approach has been successfully applied to the grouping and separation of sequences within multi-species datasets (Teeling, Meyerdierks et al. 2004; Abe, Sugawara et al. 2006; Martin, Diaz et al. 2008; Afreixo, Bastos et al. 2009; Saeed and Halgamuge 2009). A supervised approach can be taken to such grouping, where prior sequence information from a range of genomes can be ‘learnt’ and used to construct predictions of the nature and origin of query sequences (e.g. Martin, Diaz et al. 2008). However,

these supervised methods suffer from many of the same limitations imposed by the requirement for a reference database on alignment-based methods.

The principal limitation to composition-based methods of sequence comparison, when applied to massively parallel sequencing datasets, is the short length of the individual sequences that are compared. Even reads produced from platforms with the longest mean length - the GS FLX Titanium (*Roche*/*454 Life Sciences*) - rarely approach the '1 kbp barrier', that has been described previously as the lower length limit at which fragment assignments can be made with confidence (McHardy and Rigoutsos 2007). At such short lengths, the feature patterns borne out over the whole genome and on which such comparison relies, can be difficult to identify amongst short-range, local variations in nucleotide content. Such local variations include regions of repetitive sequence or overrepresentation of particular codons in a coding region.

Project summary

The aim of this project is to investigate the capability of composition-based methods of sequence comparison for the grouping and separation of reads from massively parallel high-throughput sequencing of multi-species environmental samples, according to the genome from which they originate.

The types of dataset of particular interest in this project are those produced from an environment containing a few different species. Such a dataset could be obtained from a sample of infected tissue, where sequencing reads would be expected to originate from the genomes of the host (e.g. a plant or insect), and the pathogen(s) (e.g. a bacterium, virus, or fungus). These datasets will generally be referred to as ‘multi-species’, rather than metagenomic, because metagenomics is typically associated with an environmental system or community on a much larger scale.

The motivation for this research is to establish the possible benefits of such a phylogenetic grouping of reads. An effective clustering may allow the identification of the species present in a sequenced sample allowing, for example, the identification of a particular pathogen, and the isolation of sequences belonging to the genome of a particular species in the sample. The isolation of pathogen reads from a dataset may facilitate the study of the genome of the pathogen where more conventional laboratory methods have failed. Grouping reads originating from a single genome may also improve the performance of sequence assembly methods applied to the dataset.

By using supervised methods of grouping to provide a reference comparison with known pathogens, the nature or specific identity of the infectious agent(s) in the sample can be predicted. As in metagenomics, this approach of sequencing genetic material sampled directly from the tissue, removes the requirement for the pathogen to be isolated and cultured prior to analysis.

The grouping of reads by genome can also remove contaminants within the dataset, to allow the genome of the host or pathogen species to be studied more easily. The alignment-free approach to grouping of sequences does not rely on the prior availability of a full genome for either species in order to predict the reads that belong to each group, which allows the study of potentially novel pathogens, in host species that are themselves not well-characterised.

Grouping of sequencing datasets using genomic signature features usually takes place after the reads have undergone assembly into longer contigs (Chen and Pachter 2005). The use of longer sequences improves the accuracy of the grouping obtained but, as has been discussed already, the increased burden of larger datasets adversely affects the performance and runtime of the assembly process. Where a dataset contains a large number of individual reads originating from multiple genomes - as is the case for an environmental sample sequenced directly - it may be beneficial in terms of assembly speed and quality to separate these reads according to a prediction of shared origin prior to assembly.

Grouping reads according to the genome from which they originate and assembling the groups of reads separately reduces the time taken to assemble these reads into longer stretches of sequence, relative to considering all reads in the dataset at once. It should also result in fewer erroneous, chimeric sequences being assembled from reads obtained from multiple genomes with regions of homology.

The following work describes the investigation of a range of published sequence features and the capacity for these features to group and separate sequencing reads in a multi-species sequencing dataset. The features and their combinations are compared to determine the set that best differentiates between reads based on their species of origin. This optimal set of features is used as a basis for the comparison of a range of clustering methods in order to find the optimal combination of feature and method to group a dataset. The extent of grouping and separation of reads that was achieved is then discussed in the context of the datasets studied. The effect of such a grouping on the performance of a sequence assembly algorithm is then investigated, to determine the benefit of such an approach.


sequences
species
clustering
dataset
sequence
clusters
feature
features
organisms
vectors
clusters
reads
sequencing
distribution
TNF
simLC
content
mean
frequency
data
GC
Pr
RPH
OFDEG
length
number
combinations
produced
values
separation
oligonucleotide
containing
genome
two
observed
results
obtained
quality
different
Dataset
used
cluster
randomised
described
whole
et
calculated
analysis
classification
CLARA
Rc
also
within
DNA
Cluster
use
expected
datasets
proportion
size
derived
lengths
fragments
statistics
variation
hybrid
Clustering
nucleotides
sample
BR1
grouped
oligonucleotides
compared
CPH
three
equal
almost
previously
using
vector
taken
proportions
much
returned
product
according
grouping
Fig
relative
IND
single
classes
indicated
increasing
closely-related
Proteobacterial
However
sampled
predominant
total
represented
comparison
genomes
found
discussed
value
combination
groups
al
likely
class
level
sets
group
effect
result
similar
shown
process
types
true
work
genomic
recall
nucleotide
contained
type
one
distances
applied
improvement
sub-sequence
relatively
distance
clustered
method
larger
Halgamuge
increased
across
generated
remaining
distributions
possible
error
taxonomic
together
higher
precision
also
Rc
within
variation
hybrid
statistics
random
Clustering
nucleotides
fragments
sample
BR1
Pr
grouped
oligonucleotides
GC
compared
many
equal
CPH
three
length
dataset
proportions
RPH
content
almost
simLC
much
returned
product
according
grouping
TNF
previously
using
number
combinations
vector
taken
distribution
sampled
However
sequencing
sequence
produced
OFDEG
found
discussed
value
combination
groups
Fig
relative
IND
single
classes
indicated
increasing
closely-related
Proteobacterial
reads
predominant
total
represented
comparison
genomes
clusters
mean
al
likely
class
level
sets
group
effect
result
similar
shown
process
types
true
work
genomic
recall
nucleotide
contained
type
one
values
separation
oligonucleotide
containing
genome
two
distances
applied
improvement
sub-sequence
relatively
distance
clustered
method
larger
Halgamuge
increased
across
generated
remaining
distributions
possible
error
taxonomic
observed
analysis
classification
CLARA
used
Dataset
cluster
randomised
described
whole
et
results
obtained
quality
different
feature
calculated
DNA
Cluster
use
expected
datasets
proportion
size
derived
lengths
frequency
data
clustering
species
features
organisms
higher
together
also
Rc
within
precision
variation
hybrid
statistics
random
Clustering
nucleotides
fragments
sample
BR1
Pr
grouped
oligonucleotides
GC
compared
CPH
three
equal
almost
content
simLC
much
returned
product
according
grouping
TNF
previously
using
number
combinations
vector
taken
proportions
RPH
distribution
sampled
However
sequencing
sequence
produced
OFDEG
found
discussed
value
combination
groups
Fig
relative
IND
single
classes
indicated
increasing
closely-related
Proteobacterial
reads
predominant
total
represented
comparison
genomes
clusters
mean
al
likely
class
level
sets
group
effect
result
similar
shown
process
types
true
work
genomic
recall
nucleotide
contained
type
one
values
separation
oligonucleotide
containing
genome
two
distances
applied
improvement
sub-sequence
relatively
distance
clustered
method
larger
Halgamuge
increased
across
generated
remaining
distributions
possible
error
taxonomic
observed
analysis
classification
CLARA
used
Dataset
cluster
randomised
described
whole
et
results
obtained
quality
different
feature

2

A comparison of genomic signature features applied to the clustering of simulated multi-species sequencing data by origin

Abstract

The capacity of four genomic signature sequence features to group short DNA sequencing reads according to their species of origin was compared. Two datasets, designed to simulate to a greater or lesser extent a typical multi-species sequencing dataset, were represented by these four signature features and their combinations, and grouped by a simple clustering method. It was found that clustering of tetranucleotide relative frequency distribution and GC content vectors was most successful. Although it was possible to accurately group together by species sequences <1kb in length in a low-complexity dataset, species-specific grouping of sequences was not feasible for data obtained from a larger number of species that were more closely related. In this case, a broader taxonomic grouping was achievable, enriching clusters for sequences from related species. It was concluded that a more appropriate dataset should be developed to determine the extent of grouping that could be achieved with true sequencing reads from a multi-species sample.

Introduction

The major theme of this work is the use of sequence composition-based feature vectors to group together individual reads produced from high-throughput DNA sequencing of a multi-organismal sample, according to the species from which they originated.

Once an appropriate sequencing dataset has been obtained - a process discussed in more detail later - the process of separating or clustering the reads by species of origin can be divided into two principal aspects: the generation of feature vectors from the sequences contained in the dataset; and the application of a clustering method to these feature vectors to produce groups of reads.

The desired result of the second of these steps is that the groups produced contain those sequences in the dataset that were derived from the same species in the sample. Ideally, the result would be a number of groups or clusters that is equal to the number of species that contributed to the dataset, with each cluster containing all the sequences that originated from a single species with no contamination in the form of sequences from other organisms.

To maximise the success of this clustering, an appropriate type of feature vector should be chosen to represent the sequences in the first step. Feature vectors can be thought of as statistical representations of a sequence, which describe it in some way that is comparable between sequences. If these descriptions identify the differences between sequences from different species, while expressing the similarity of sequences from the same species then the feature vectors can be used to group the data accordingly.

In this chapter, a comparison of four different types of feature vector taken from the literature is described. These four types of feature are applied, individually and in combination, to two synthetic datasets of different composition and complexity. The datasets simulate to a greater or lesser extent the type of data generated in high-throughput DNA sequencing, with the aim of finding the optimal vector type for clustering of these sequences.

GC content

The GC content of DNA, the most well-known feature for characterising and comparing DNA sequencing, has been used to characterise sequences for many years (Musto, Naya et al. 2006; Ussery, Wassenaar et al. 2009), particularly in the classification and comparison of bacterial species, where it has become commonplace to refer to particular species as ‘GC-rich’ and ‘GC-poor’ (e.g. Romero, Zavala et al. 2000; Naya, Romero et al. 2001). The GC content of bacterial species has been shown to vary across a wide range between species, in a bimodal distribution either side of the 50% GC content that might be expected as the most common (Ussery, Wassenaar et al. 2009), and it is this difference in GC content between the genomes of species, and the ease with which the content can be calculated that makes it such a widely used means of comparison.

The GC content of a sequence is simply the proportion of the sequence that is made up from guanine (G) and cytosine (C) nucleotides, calculated as the combined frequency of these two nucleotides, divided by the total number of nucleotides in the sequence. This proportion is equal across both strands of the DNA double helix, as each C or G nucleotide on one strand is coupled with G or C on the opposing strand in Watson-Crick base-pairing (Watson and Crick 1953).

Much investigation has been made into GC content, with a great deal of focus on the predicted increase in thermal stability conferred upon double-stranded DNA with an increased proportion of G/C basepairs (Bernardi and Bernardi 1986; Galtier and Lobry 1997). This stability at higher temperatures is thought to be a product of the three hydrogen bonds formed in the pairing of these two nucleotides on opposing strands, compared to two hydrogen bonds in Watson-Crick pairing of adenine (A) and thymine (T) (Wada and Suyama 1986).

The evolutionary cause of GC content variation between genomes has been debated at length for years, with increased thermal stability one much discussed theory, and alternatives including many other environmental pressures (Basak and Ghosh 2005; Musto, Naya et al. 2006; Wang, Susko et al. 2006) and biases in DNA replication and maintenance machinery (Wu, Zhang et al. 2012).

GC content shows little variation throughout the genome in prokaryotic species (Sueoka 1962) although regional variations do exist, particularly around the centre of replication origin (Ussery, Wassenaar et al. 2009), and in coding regions (Bohlin, Skjerve et al. 2008).

GC content varies much more markedly within the genomes of eukaryotic organisms, in regions known as isochores. Isochores in the human genome have been described as ~1 Mbp in length, and placed into five categories based on the proportion of G and C nucelotides in the sequence (Oliver, Bernaola-Galván et al. 2001; Oliver, Carpena et al. 2002; Bernardi 2007). A correlation has been found between the locations of isochores and the distribution of genes within the genome (Zoubak, Clay et al. 1996).

It was originally thought that isochores were much more prominent in the genomes of warm-blooded organisms (Bernardi, Olofsson et al. 1985), but long-range variations in GC content have been observed to some extent in yeasts and plants (Oliver, Bernaola-Galván et al. 2001; Zhang and Zhang 2004; Oliver, Bernaola-Galvan et al. 2008).

Since full genome sequences have begun to be made available, the existence of isochores has been challenged (IHGC 2001), and uncertainty over their nature and definition remains (Elhaik, Graur et al. 2010).

The genome of *Arabidopsis thaliana* has been described as containing 15 isochores across it's five chromosomes (Zhang and Zhang 2004), although this number and the criteria for assigning regions of sequence as isochores have been challenged (Chen and Gao 2005). The isochores described by (Zhang and Zhang 2004) span huge regions of the genome, up to ~10 Mbp in length. This length and overall number of isochores is in stark contrast to the many ~1 Mbp regions originally described as isochores in the human genome.

The existence of isochores in a genome could interfere with the grouping of sequencing reads obtained from a sample. If the variation in GC content within a genome in the sample were great enough between isochores, the sequencing reads generated from these different regions could be grouped separately. These separate groups of reads may overlap with those reads produced from other genomes in the multi-species dataset, resulting in poorer inter-species separation of reads.

In order to ascertain the likelihood of such interference occurring in the work in this project, involving the genome of *A. thaliana*, a clustering analysis of sequences from this genome, classified by isochore, was performed here. This investigation was carried out to determine what effect, if any, the existence of isochores (as defined by Zhang and Zhang 2004) might have on the grouping only of sequences from *A. thaliana* using the feature types and clustering methods used throughout this project.

As a descriptive property of DNA, GC content has been used previously, sometimes in conjunction with other sequence feature types, to group sequence fragments based on their predicted species of origin (Teeling, Meyerdierks et al. 2004; Saeed and Halgamuge 2009). Given the consistency of GC content within the genome of prokaryotes, the ease with which it can be calculated, and its historical context as an established characteristic of DNA, the feature was used in the investigations described here (sometimes referred to simply as GC). The greater degree of variation of GC content within eukaryotic genomes was predicted to be a limitation when the methods were applied to datasets containing sequences from higher organisms.

Tetra-nucleotide frequency

First described by Karlin and Ladunga (1994), the distribution of the relative frequencies of oligonucleotide ‘words’ in DNA has become a popular basis for the comparison of genomic sequences of various lengths (Pride, Meinersmann et al. 2003; Teeling, Meyerdierks et al. 2004; Paz, Kirzhner et al. 2006), gaining the nickname ‘genomic signature’ in the process (Karlin and Burge 1995). Previously published works have demonstrated that the distribution of oligonucleotide frequencies in a sequence are more similar in sequence fragments from the same species that in those from different species (Gentles and Karlin 2001), and established the capacity for these distributions to be used to group sequences according to the genome from which they originate (e.g. Teeling, Meyerdierks et al. 2004).

The process for the calculation of the full oligonucleotide relative frequency distribution of a sequence is described in Figure 2.1. The distribution can be calculated for any user-defined integer ‘word’ length, ℓ (ℓ = 2 in Fig. 2.1).

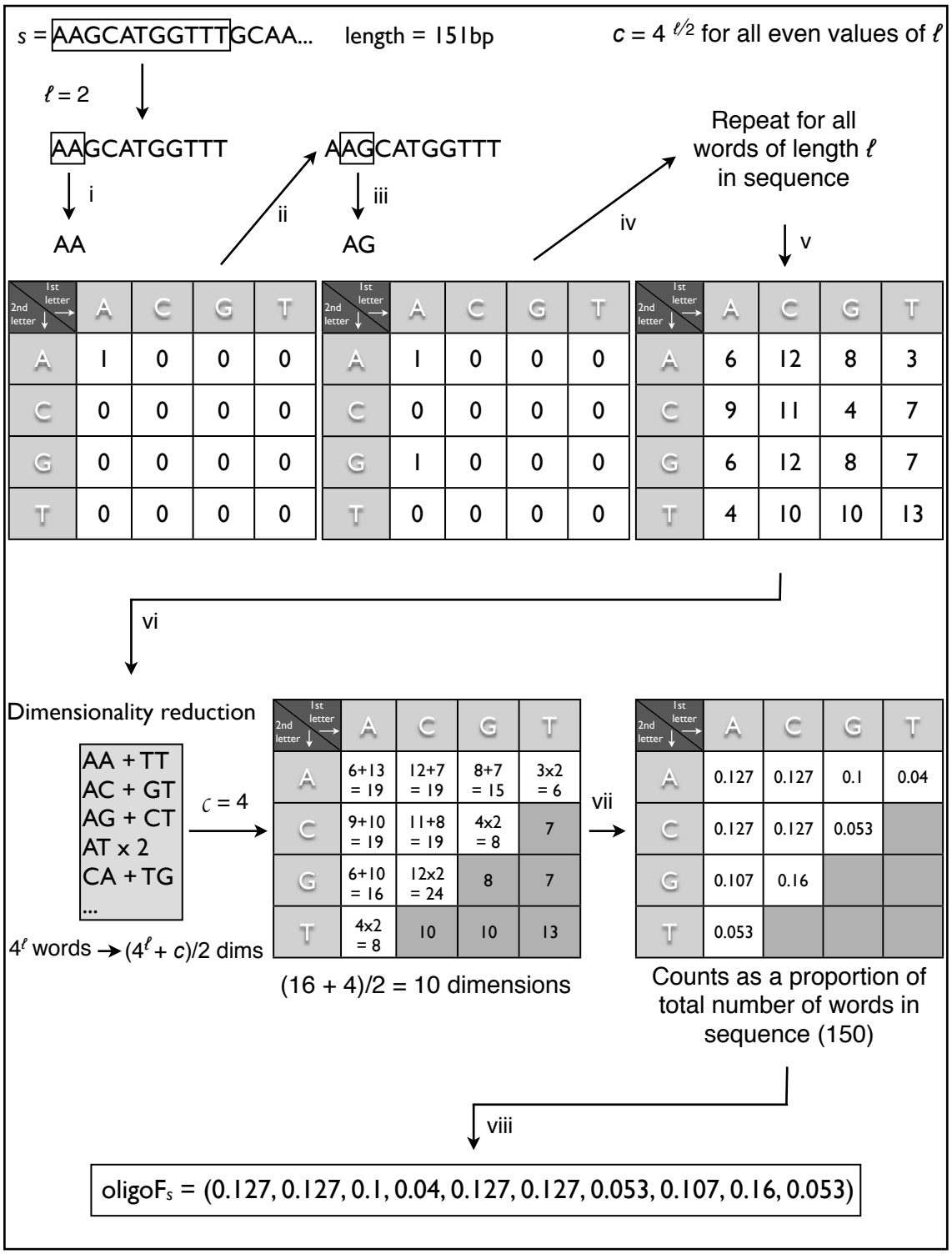


Figure 2.1 *Flow diagram describing the generation of an oligonucleotide frequency feature vector oligoF_s, for sequence* s. *In the example above, the word length ℓ = 2, while a ℓ = 4 was used in the work described in this project. A sliding (overlapping) window of size ℓ letters is used to count the frequency of all $4^ℓ$ possible combinations of letters in the DNA alphabet* A ∋ *[A, C, G, T] (i - v). These $4^ℓ$ counts, which constitute individual dimensions of the feature vector can be reduced to $(4^ℓ + c)/2$ counts, where* c *is the number of palindromic words of length ℓ (*c = $4^{ℓ/2}$ *for even values of ℓ, and 0 for odd values of ℓ), by combining counts for words with those for their reverse complement and doubling the counts for palindromic words. This has the effect of counting words on both strands of the sequence, and reduces the overall dimensionality of the resulting feature vector (vi). These combined counts are normalised to the number of words in the sequence [sequence length - (word length - 1)] (vii), producing a feature vector, oligoF_s, of relative oligonucleotide frequencies (viii).*

For a given sequence, the number of occurrences of each possible combination of nucleotides (word) of the given length is counted, with the words overlapping, such that each sequence produces a total of S - (ℓ - 1) words, where S is defined as the length of the sequence in base pairs. Each of these oligonucleotide (word) frequencies is then normalised to the total number of oligonucleotides in the sequence, and these relative frequency values form the feature vector.

As each DNA sequence is composed from a four-letter nucleotide alphabet - A, C, G and T - the relative nucleotide frequency distribution is composed of 4^{ℓ} values; one for each combination of ℓ nucleotides. For example, the relative di-nucleotide (ℓ = 2) frequency distribution of a sequence contains 16 values, while the tetra-nucleotide (ℓ = 4) distribution contains 256. As depicted in Fig. 2.1, this dimensionality of the relative frequency distribution can be reduced by almost half by combining the frequencies of oligonucleotides with those of their reverse complement, and doubling the observed frequency of palindromic oligonucleotides. This process accounts for the oligonucleotide frequency distribution across both strands of the sequence, and reduces the size of the distributions produced from 4^{ℓ} to $(4^{\ell} + c)/2$, where c is the number of palindromic oligonucleotides ($c = 4^{\ell/2}$ for even values of ℓ, and 0 for odd values of ℓ). As such, the di-nucleotide frequency distribution of a sequence can be expressed in 10 values instead of 16, and the tetra-nucleotide frequency distribution can be expressed in 136 values instead of 256. This reduction could dramatically increase the runtime of any clustering methods applied to the data that are sensitive to high-dimensionality.

A balance must be struck between the increased specificity of the distribution generated from longer oligonucleotides and the increased sparseness that this exponential increase in distribution size introduces. A longer oligonucleotide produces a distribution of more values per sequence, which provides a greater scope for the differences and similarities between sequences to be expressed, but the larger number of possible oligonucleotide frequencies, counted from the same length of sequence, results in a more sparsely-populated distribution. Indeed, it can be shown that >1% of all the approximately one trillion possible 20 nucleotide ‘words’ can occur in the whole human genome of around 3 billion base pairs (Fedorova and Fedorov 2011).

To illustrate this point further, if the di-nucleotide frequency distribution is taken from a sequence fragment 200 bp in length, sampled from a larger sequence, the distribution produced will be the product of 199 di-nucleotides in 16 possible combinations, reduced to 10 dimensions as described previously. However, if the tetra-nucleotide frequency distribution is taken from the same 200 bp sequence, the distribution produced will be the product of 197 tetra-nucleotides in 256 possible combinations (reduced to 136 as above).

It follows from these calculations that the tetra-nucleotide frequency distribution will be more sparsely populated than that of the di-nucleotides in the sequence, having been sampled from a smaller space relative to the size of the distribution - the depth of sampling per oligonucleotide is greater for di-nucleotides, where the mean sampled frequency is equal to 199/16, than for tetra-nucleotides where the mean sampled frequency is 197/256. This distribution, then, is less likely to represent correctly the true distribution of tetra-nucleotides within the whole sequence (genome, chromosome etc.) from which the 200 bp was sampled, and is subsequently less likely to be grouped accurately with another sequence sampled from the same source.

The other complication associated with increasing oligonucleotide length is in the calculation of the distribution. The computational requirement for calculation of these distributions scales exponentially with the increase in oligonucleotide length, proportional to the increase in the number of possible oligonucleotide combinations (4^{ℓ}).

Previously published work has shown that the benefit in discriminatory power associated with each additional nucleotide in the word length used (i.e. use of tri-nucleotides instead of di-nucleotides, use of penta-nucleotides instead of tetra-nucleotides etc.) decreases considerably beyond a word length of four (Bohlin, Skjerve et al. 2008), and the research community has largely settled on the use of tetra-nucleotides (e.g. Pride, Meinersmann et al. 2003; Teeling, Meyerdierks et al. 2004; Willner, Thurber et al. 2009). Tetra-nucleotide frequency distributions (TNF) are the feature vectors used in the work described here.

Several variations on oligonucleotide frequency distributions have been used as genomic signatures, with the frequency counts being represented in different

ways to maximise the information provided by the distribution. A commonly used approach has been to represent observed oligonucleotide frequencies relative to their expected frequencies given the overall nucleotide composition of the sequence. For example, in a genome with a high overall GC content the frequency of oligonucleotides consisting largely of A and T nucleotides could be expected to be uniformly lower than of those of predominantly G and C nucleotides. By comparing the observed frequencies with the expected values, distribution vectors can be normalised to account in part for localised variation in nucleotide frequency. *z*-scores calculated between observed and expected tetranucleotide frequencies (Teeling, Meyerdierks et al. 2004; Teeling, Waldmann et al. 2004) and probabilities of observed tri- and tetranucleotides calculated by Markov chain estimation of expected values (Nasser, Breland et al. 2008) have been used to group sequences by species of origin.

Another variation on frequency counts that has been used as a genomic signature feature is the *tf-ti* (term frequency-term importance) representation described by (Martin, Diaz et al. 2008). These feature distributions are calculated by dividing the observed frequency of a tetranucleotide by the product of the total number of tetranucleotides in that sequence and the total frequency of that tetranucleotide in all sequences in the dataset. This process results in the feature values for those tetranucleotides within the distribution that are rare within the sequence and/or common throughout the dataset being represented by a lower *tf-ti* score than those that are common within a sequence but relatively rare throughout the dataset as a whole. This representation is based on the rationale that it is the oligonucleotides that are not uniformly common or rare throughout the dataset but appear more often within individual sequences that are the most informative to any grouping performed.

Recently, it has been shown that the use of frequency distributions of only the palidromic words in a sequence can be used as a genomic signature feature (Lamprea-Burgunder, Ludin et al. 2011). These palindromic oligonucleotides are underrepresented in the genomes of many different species, and their distributions have been shown to allow grouping of ~10 kbp sequences by genome of origin (Lamprea-Burgunder, Ludin et al. 2011). The use of palindromic oligonucleotides, rather than the full distribution of all possible

words in each sequence, is beneficial to the analysis of large datasets, as the dimensionality and overall size of the feature vectors produced from the sequences is reduced. Reduced dimensionality may be beneficial to the performance of any subsequent clustering analysis sensitive to this factor.

It is noted here that the GC content features described above as a method for describing sequences can be interpreted as a very basic form of oligonucleotide frequency distribution vector, where the frequency of two mono-nucleotides have been combined to give a single value.

Similarly to the case of GC content, the mechanisms leading to these signature patterns in oligonucleotide frequency distributions of genomes are not well understood, but are thought to be the the product of slight variation in the DNA repair and replication machinery introducing slight bias towards/against the presence of certain oligonucleotides (Karlin, Mrázek et al. 1997).

Oligonucleotide frequency-derived error gradient

The oligonucleotide frequency-derived error gradient (OFDEG) was introduced by Isaam Saeed and Saman Halgamuge in 2009. As the name suggests, this sequence feature is derived from oligonucleotide frequency distributions taken from sub-sequences sampled from each sequence in a dataset. An overview of the method by which these feature values are calculated is given in Figure 2.2.

For each sequence, a sub-sequence, L_C, of length equal to that of the shortest sequence in the dataset, C, is taken. First, the frequency distribution of oligonucleotides of a defined length, m, is taken from this sub-sequence as a whole. Next, a sample, s_i, of length $i = m$ is taken from the sub-sequence, and the frequency distribution calculated from this sampled sequence. As this sampled sequence usually forms only a small fraction of the whole sub-sequence, the frequency distribution is likely to be considerably divergent from that of the whole, calculated previously. The error between these two distributions is calculated and stored.

This process of sampling, distribution calculation and error measurement is repeated, with i increased by a step-size, t, at each iteration until the sampled sequence length is equal to that of the sub-sequence originally taken. As the sequences are at this point identical, the error between the two distributions at this final stage will be equal to zero. Saeed and Halgamuge found that the error

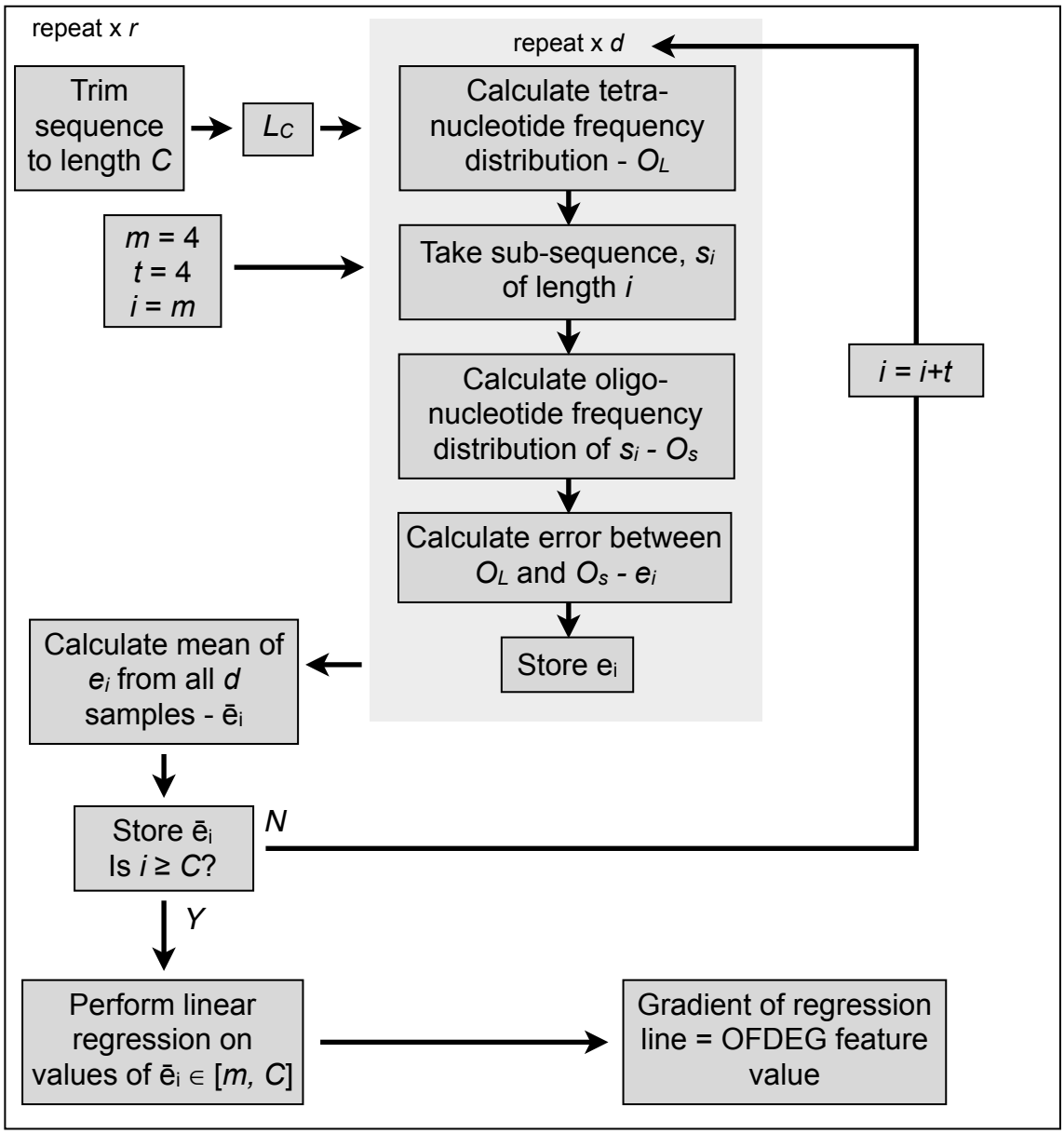


Figure 2.2 *Flow diagram representing the processing of sequences to generate oligonucleotide frequency derived error gradient (OFDEG) feature values, as described in (Saeed & Halgamuge, 2009). The sequence is first trimmed from a random point up to length* C*, the length of the shortest sequence in the dataset, producing* L_C*. The tetranucleotide frequency distribution of* L_C*,* O_L *is calculated. A sub-sequence,* s_i*, of length* i *is taken. Initially,* i = m*, the length of the oligonucleotides used to calculate the frequency distribution (in this case,* m = 4*) is taken. The tetranucleotide frequency distribution of* s_i*,* O_s*, is then calculated, and* e_i*, the error between* O_s *and* O_L*, is calculated and stored. The process of sampling* s_i *from* L_C *can be repeated a number of times, known as the sampling depth,* d*, and the mean error value,* $\bar{e}_i$*, taken. The process of sampling sub-sequences from* L_C *is repeated for values of* i *increasing by a step size,* t*, with error values being calculated and stored at each sub-sequence length, until* i ≥ C*. The OFDEG statistic is calculated as the gradient of a linear regression of the values of* $\bar{e}_i$ *for lengths i between 0 and 0.8**L_C*. For sequences of length greater than* C*,the whole process may be repeated several times, with the mean value of the error regression gradient subsequently used as the OFDEG value for the sequence. The user-defined number of these repeats is known as the re-sampling depth,* r*.*

values, collected as the sample size increases, decrease in a linear fashion for sampled sequences up to ~80% of the total length of the sub-sequence taken. When these error values calculated for sampled sequences between *w* and (0.8*L_C) are plotted in a linear regression, the gradient of this regression forms the value for the OFDEG feature vector.

For all sequences longer than the shortest in the dataset, the process of taking a sub-sequence and calculating its OFDEG value may be repeated several times, with the sub-sequence taken at random on each iteration and the mean of the gradients calculated returned as the OFDEG feature value for the whole sequence. The number of times, *r*, that this sub-sequence selection takes place is referred to as the *re-sampling depth* (Saeed and Halgamuge 2009).

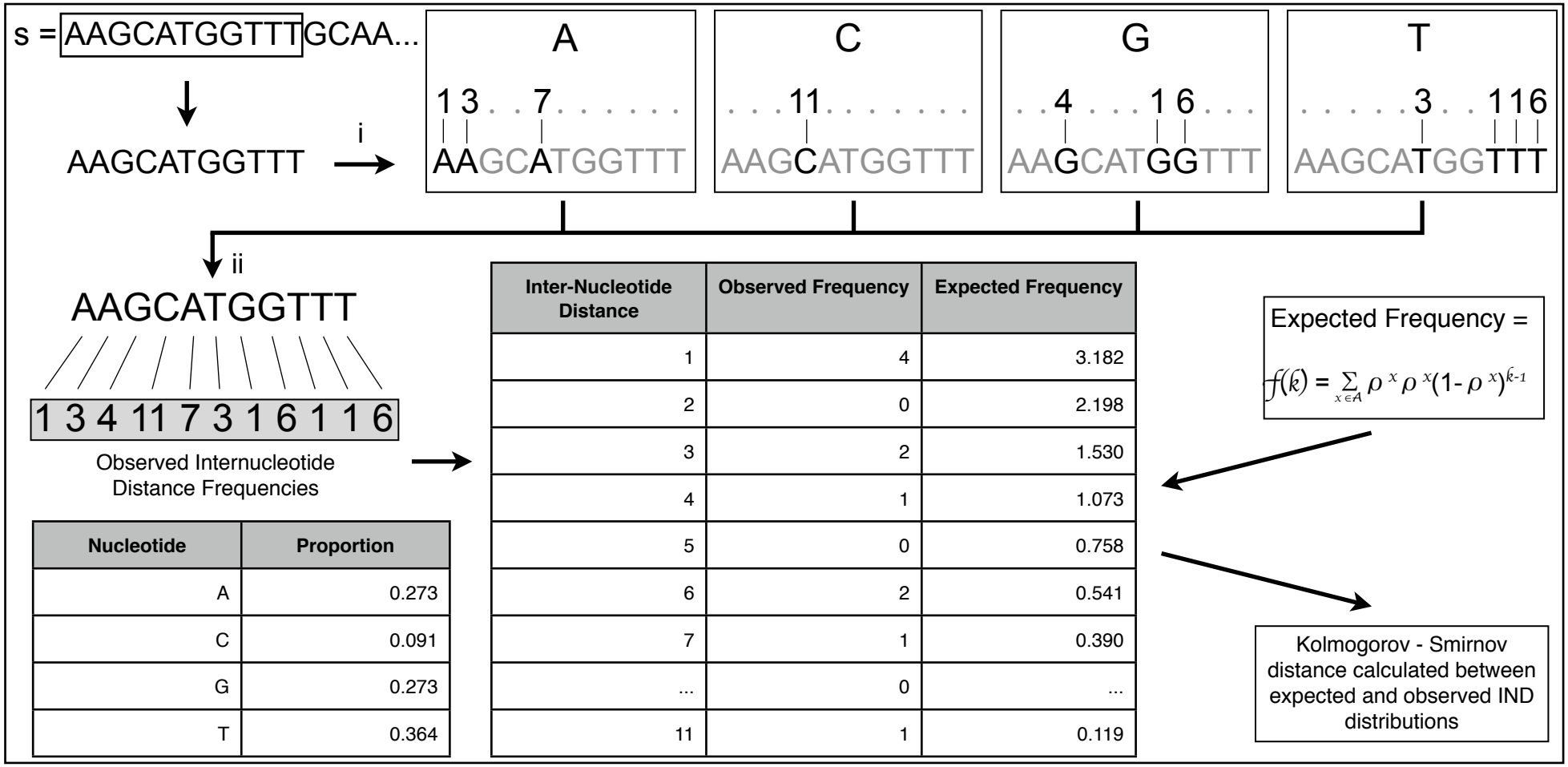
The process of sampling a sequence at each step for calculation of the oligonucleotide distribution and error value may also be repeated, to increase the sample size at each step. The mean error across all sampled sequences is used for calculation of the OFDEG value. The number of times, *d,* that sampling is repeated at each step is referred to as the *sampling depth* (Saeed and Halgamuge 2009).

In accordance with the values used in the original implementation of OFDEG features (Saeed and Halgamuge 2009), the re-sampling depth and sampling depth values used in the generation of these feature values in this work were set at 5 and 20 respectively.

Sequences were removed from the analysis if the OFDEG gradient was obtained from a regression with an R^2 value <0.9. This measure was taken to prevent OFDEG values being used that were not derived from a consistent gradient on a linear regression. A lower R^2 value is likely to be the result of OFDEG measurement from a partially highly-repetitive sequence, where the oligonucleotide distribution of shorter sections of the sequence is more similar to that of the sequence as a whole than would be expected from a ‘normal’ sequence taken from a non-repetitious section of the genome.

Inter-nucleotide distance

The concept of inter-nucleotide distances (INDs) was introduced by Nair and Mahalakshmi (2005) as a progression from binary indicator sequences (Voss 1992; Anastassiou 2000), as a tool applied to the identification of promoter



Nucleotide	Proportion
A	0.273
C	0.091
G	0.273
T	0.364

Inter-Nucleotide Distance	Observed Frequency	Expected Frequency
1	4	3.182
2	0	2.198
3	2	1.530
4	1	1.073
5	0	0.758
6	2	0.541
7	1	0.390
...	0	...
11	1	0.119

Figure 2.3 *Flow diagram describing the generation of internucleotide distance (IND) feature values. For each member of the nucleotide alphabet $A \ni$ [A, C, G, T] distances are recorded between each instance of a nucleotide and the next (i). These distance vectors are combined to form a global distance vector (ii). The frequency distribution of these distances is recorded and compared to the frequency distribution expected if the sequence was generated by an independent random process with fixed parameters, calculated as described by (Afreixo et al, 2009) (iii & iv). The IND feature value is a measure of the difference between these two distributions, calculated as the Kolmogorov-Smirnov distance between them(v).*

regions in DNA sequences. A binary indicator sequence is a binary string of equal length to a string of symbols (in this case, the DNA sequence), such that the positions of particular type of symbol (A, C, G or T) in the sequence is denoted by a ‘1’ in the indicator sequence, and all other symbols by a ‘0’ (Voss 1992). The principle was later applied to the characterisation and comparison of genomic sequences by Afreixo and colleagues (2009), introducing the use of IND features as a genomic signature pattern and using them to perform a phylogenetic analysis on DNA sequences from multiple species.

The process implemented in (Afreixo, Bastos et al. 2009) to generate these IND feature vectors from a DNA sequence is outlined in Figure 2.3.

The global inter-nucleotide distance is defined as the number of nucleotide bases in a DNA sequence between one instance of a particular nucleotide - adenine (A), cytosine (C), guanine (G) or tyrosine (T) - and the next instance of that nucleotide. If these distances are counted and placed in a vector corresponding to the sequence, the frequency distribution of the distances in the vector can be determined. The IND feature value is a measure of how dissimilar this frequency distribution is to the distribution of distances that would be expected if the sequence of nucleotides were the result of an independent random process, that is, where each nucleotide in the sequence was determined independently from the last, according to the relative proportions of the nucleotides observed in the whole sequence.

The dissimilarity is measured as the Kolmogorov-Smirnov (K-S) distance between the observed and expected IND distributions for each sequence. The distribution is calculated across the whole sequence, but only the observed and expected distribution of the first 25 distances - that is, the frequency distribution of inter-nucleotide distances 1, 2, 3, ..., 24, and 25 - are used to calculate the K-S distance that forms the IND feature value for the sequence. In (Afreixo, Bastos et al. 2009), the first 100 distances were used for characterising sequences 500 kbp in length. Here, a distance limit of 25 was chosen for calculation of the distribution to account for the much shorter length of sequences under comparison. The limit was introduced as a mechanism to guard against the possible generation of IND distributions that were excessively sparse, which could make comparison of sequences more difficult.

On a related note, the authors of the original research establishing inter-nucleotide distances as a descriptive feature of DNA sequences have recently published work on inter-dinucleotide distances of the human genome (Bastos, Afreixo et al. 2011). So far, no details of investigation into the capacity for these features to be used as a genomic signature pattern have been published.

Feature type comparison

In order to find the optimal sequence feature vector, composed of any combination of the four feature types described, the quality of separation obtainable using each possible combination was studied and compared.

One of the challenges associated with performing clustering analysis with multi-species sequencing data is the lack of specific knowledge of the true content and proportions of the dataset. This limits the extent to which a quantitative evaluation can be performed on any results obtained, and the conclusions that can be drawn from such an analysis (Mavromatis, Ivanova et al. 2007).

In order to address this problem, synthetic datasets have been introduced that either combine sequencing reads from several sequencing experiments performed on individual species (Mavromatis, Ivanova et al. 2007), or else contain sequencing reads produced from a mixed sample of known species in known proportions (Morgan, Darling et al. 2010).

The use of a simulated or synthetic dataset such as these allows the clustering results produced with each feature type and combination of features to be compared directly through the error observed between the clustering and the ‘true’ distribution of sequences within the dataset. When engineering such a dataset, it remains extremely challenging to accurately model the properties of a true multi-species sequencing dataset, such as the particular sequence error and length profiles, the relative proportions of reads from different species, and the introduction of reads that do not effectively map to any contributing genome, due to high sequencing error, low quality base-calling or sequencing noise (e.g. Morgan, Darling et al. 2010).

In this chapter, the application of GC, IND, OFDEG and TNF features to clustering of two synthetic datasets is described, and the results compared with the aim of identifying the feature(s) that enable the best separation of sequences according to their species of origin.

First, a simplistic dataset was used as a basis for investigation of clustering performance with an idealised dataset. Second, a more complex dataset containing sequences from more species, represented disproportionately, is used to provide a more stringent assessment of the clustering performance of each feature type and combination.

Dataset 1

The first round of feature comparisons was performed with an equally-proportioned dataset of sequence fragments taken from the genome of three species that are not closely related: the plant *Arabidopsis thaliana*, the fungus *Aspergillus fumigatus*, and the bacteria *Escherichia coli*. The simplistic nature of this dataset, referred to as Dataset 1, allowed the upper limit of the clustering quality achievable with each feature and combination to be estimated.

Several variants of the dataset were generated, each containing sequence fragments of a different mean length from 200 bp, at intervals of 200 bp, up to 1000 bp. It has been shown extensively that longer sequences can be grouped more easily, due to the larger sample space from which the genomic signature pattern can be approximated (Abe, Kanaya et al. 2003; Huson, Auch et al. 2007; Martin, Diaz et al. 2008; Saeed and Halgamuge 2009). However, most massively parallel sequencing technologies do not produce reads whose length approaches the upper boundary of the range chosen here. The most recent 454 GS FLX Titanium platforms can produce reads as long as 1000 bp, but typically reads generated with this technology are ~700 bp in length, with those produced from *Illumina,* SOLiD and Ion Torrent platforms <400 bp in length. The evaluation of clustering results with each variant allowed the effect of increasing mean sequence length to be investigated, and provide an estimate of the potential for such an approach to be applied to data obtained from different sequencing platforms.

Dataset 1 contained 60,000 sequence fragments in total, divided into 20,000 each from each of the three organisms. The size, constituents and proportions of this dataset are not typical of the properties that would be expected from a metagenomic or multi-species sequencing dataset, which would be larger (contain a larger number of sequences in total) and contain sequences originating from a larger number of species. In addition, the contributing species

would likely be more closely related and less equally proportioned.

simLC

The second dataset, *simLC*, was a previously published simulated dataset (Mavromatis, Ivanova et al. 2007), composed from Sanger sequencing reads of the genomes of 112 individual strains of 108 microbial species in varying proportions. (The published *simLC* dataset contains sequences from 113 strains from the 108 species, but, due to a database error, sequences from one of these species were omitted from the dataset implemented in the work described here.) A breakdown of *simLC*, showing the proportion of reads derived from each species in the dataset, is given in Figure 2.4.

In total, after reads were filtered according to the R^2 parameter described for OFDEG features previously, the *simLC* dataset used consisted of 97,255 individual reads, distributed in uneven proportions between the 112 constituent strains, with 43,306 reads (44.28%) derived from the three most highly represented - *Rhodopseudomonas palustris* HaA2, *Bradyrhizobium* sp. BTAi1, and *Cytophaga hutchinsonii* ATCC 33406 - while the least-represented species accounted for fewer than 200 sequences each. Additionally, the dataset consisted of four strains of *R. palustris* (BisA53, BisB5 BisB18 and the predominant HaA2), further increasing the dominance of this species within the dataset.

Rhodopseudomonas palustris is a species of gram-negative bacteria studied in large part because of its highly adaptable metabolism *(Larimer, Chain et al. 2004)*. *R. palustris* is capable of switching between four types of metabolism - aerobic, anaerobic, chemo-autotrophic and photo-autotrophic *(Larimer, Chain et al. 2004; Bell, Tan et al. 2010)*.

Bradyrhizobium sp. BTAi1 is a species strain of bacteria that is symbiotic to the roots of plants and important in the process of nitrogen-fixation (van Rhijn and Vanderleyden, 1995). It is the only species of *Bradyrhizobium* represented in *simLC*.

Cytophaga hutchinsonii is a species of gram-negative bacteria common in soil. It is able to rapidly digest cellulose from plant tissue (Zhu et al. 2010). *C. hutchinsonii* ATCC 33406 is the only strain of the bacterium represented in *simLC*.

The mean length of sequencing reads in the dataset was ~933 bp, which is considerably longer than that of reads produced in current high-throughput sequencing. Of the current high-throughput systems, 454 GS FLX Titanium sequencing (*Roche*/*454 Life Sciences*) produces the longest reads, but even the mean length of these reads is ~750 bp.

The larger mean length of sequencing reads notwithstanding, the disproportionality of the dataset, the close relatedness of the species that constitute the dataset, and the large number of different species used all contributed to *simLC* providing a much more stringent test of the clustering capacity of the features investigated.

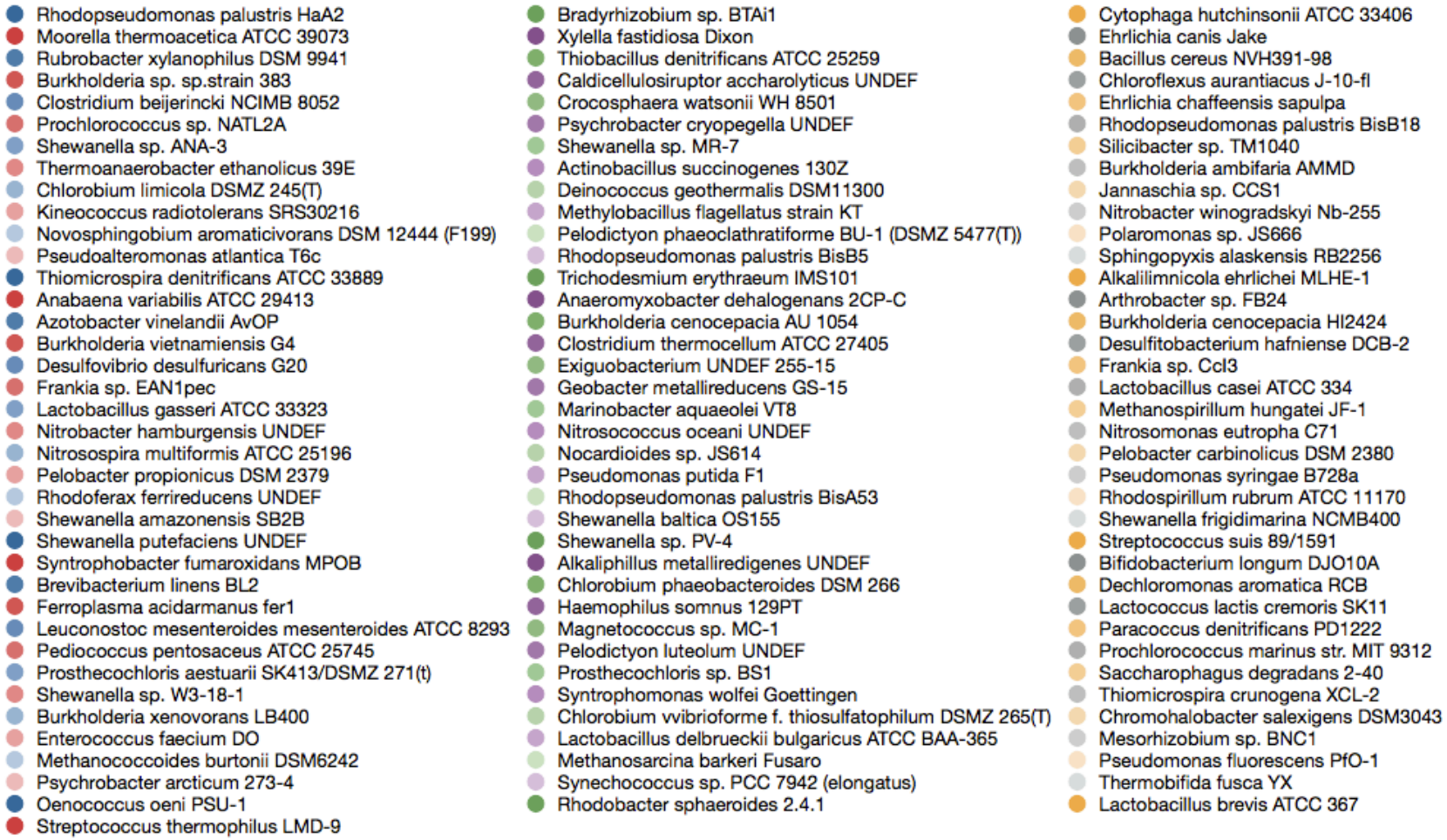
CLARA

To ensure that any variation observed in results from each feature and combination could be accounted for solely by the different feature types used in each clustering experiment, the same clustering method was used for all experiments.

Clustering LARge Applications (CLARA) is a variant of partitioning around mediods (PAM, Kaufman and Rousseeuw 1990), itself closely related to k-medians clustering (Kaufman and Rousseeuw 1990). Designed to apply the PAM method to larger datasets, CLARA takes a sample of the whole dataset and computes a dissimilarity matrix between these sampled datapoints. A user-defined number of mediods, *k*, are selected to form the central points of *k* clusters by finding the points with minimal summed dissimilarity to all the others in the same cluster within the sampled data. A datapoint within the sample is said to belong to the same cluster as a mediod if that datapoint is closer to that mediod than any other. After this mediod selection has been completed for the sampled data, the remaining datapoints are assigned to the cluster containing their nearest mediod.

This clustering process is repeated multiple times with the best result returned, as determined by the mean distance from each datapoint in each cluster to its mediod.

Clustering by minimising dissimilarity between points makes the method robust to noise in the dataset (Kaufman and Rousseeuw 1990).


Rhodopseudomonas palustris HaA2
Moorella thermoacetica ATCC 39073
Rubrobacter xylanophilus DSM 9941
Burkholderia sp. sp.strain 383
Clostridium beijerincki NCIMB 8052
Prochlorococcus sp. NATL2A
Shewanella sp. ANA-3
Thermoanaerobacter ethanolicus 39E
Chlorobium limicola DSMZ 245(T)
Kineococcus radiotolerans SRS30216
Novosphingobium aromaticivorans DSM 12444 (F199)
Pseudoalteromonas atlantica T6c
Thiomicrospira denitrificans ATCC 33889
Anabaena variabilis ATCC 29413
Azotobacter vinelandii AvOP
Burkholderia vietnamiensis G4
Desulfovibrio desulfuricans G20
Frankia sp. EAN1pec
Lactobacillus gasseri ATCC 33323
Nitrobacter hamburgensis UNDEF
Nitrosospira multiformis ATCC 25196
Pelobacter propionicus DSM 2379
Rhodoferax ferrireducens UNDEF
Shewanella amazonensis SB2B
Shewanella putefaciens UNDEF
Syntrophobacter fumaroxidans MPOB
Brevibacterium linens BL2
Ferroplasma acidarmanus fer1
Leuconostoc mesenteroides mesenteroides ATCC 8293
Pediococcus pentosaceus ATCC 25745
Prosthecochloris aestuarii SK413/DSMZ 271(t)
Shewanella sp. W3-18-1
Burkholderia xenovorans LB400
Enterococcus faecium DO
Methanococcoides burtonii DSM6242
Psychrobacter arcticum 273-4
Oenococcus oeni PSU-1
Streptococcus thermophilus LMD-9
Bradyrhizobium sp. BTAi1
Xylella fastidiosa Dixon
Thiobacillus denitrificans ATCC 25259
Caldicellulosiruptor accharolyticus UNDEF
Crocosphaera watsonii WH 8501
Psychrobacter cryopegella UNDEF
Shewanella sp. MR-7
Actinobacillus succinogenes 130Z
Deinococcus geothermalis DSM11300
Methylobacillus flagellatus strain KT
Pelodictyon phaeoclathratiforme BU-1 (DSMZ 5477(T))
Rhodopseudomonas palustris BisB5
Trichodesmium erythraeum IMS101
Anaeromyxobacter dehalogenans 2CP-C
Burkholderia cenocepacia AU 1054
Clostridium thermocellum ATCC 27405
Exiguobacterium UNDEF 255-15
Geobacter metallireducens GS-15
Marinobacter aquaeolei VT8
Nitrosococcus oceani UNDEF
Nocardioides sp. JS614
Pseudomonas putida F1
Rhodopseudomonas palustris BisA53
Shewanella baltica OS155
Shewanella sp. PV-4
Alkaliphillus metalliredigenes UNDEF
Chlorobium phaeobacteroides DSM 266
Haemophilus somnus 129PT
Magnetococcus sp. MC-1
Pelodictyon luteolum UNDEF
Prosthecochloris sp. BS1
Syntrophomonas wolfei Goettingen
Chlorobium vvibrioforme f. thiosulfatophilum DSMZ 265(T)
Lactobacillus delbrueckii bulgaricus ATCC BAA-365
Methanosarcina barkeri Fusaro
Synechococcus sp. PCC 7942 (elongatus)
Rhodobacter sphaeroides 2.4.1
Cytophaga hutchinsonii ATCC 33406
Ehrlichia canis Jake
Bacillus cereus NVH391-98
Chloroflexus aurantiacus J-10-fl
Ehrlichia chaffeensis sapulpa
Rhodopseudomonas palustris BisB18
Silicibacter sp. TM1040
Burkholderia ambifaria AMMD
Jannaschia sp. CCS1
Nitrobacter winogradskyi Nb-255
Polaromonas sp. JS666
Sphingopyxis alaskensis RB2256
Alkalilimnicola ehrlichei MLHE-1
Arthrobacter sp. FB24
Burkholderia cenocepacia HI2424
Desulfitobacterium hafniense DCB-2
Frankia sp. CcI3
Lactobacillus casei ATCC 334
Methanospirillum hungatei JF-1
Nitrosomonas eutropha C71
Pelobacter carbinolicus DSM 2380
Pseudomonas syringae B728a
Rhodospirillum rubrum ATCC 11170
Shewanella frigidimarina NCMB400
Streptococcus suis 89/1591
Bifidobacterium longum DJO10A
Dechloromonas aromatica RCB
Lactococcus lactis cremoris SK11
Paracoccus denitrificans PD1222
Prochlorococcus marinus str. MIT 9312
Saccharophagus degradans 2-40
Thiomicrospira crunogena XCL-2
Chromohalobacter salexigens DSM3043
Mesorhizobium sp. BNC1
Pseudomonas fluorescens PfO-1
Thermobifida fusca YX
Lactobacillus brevis ATCC 367

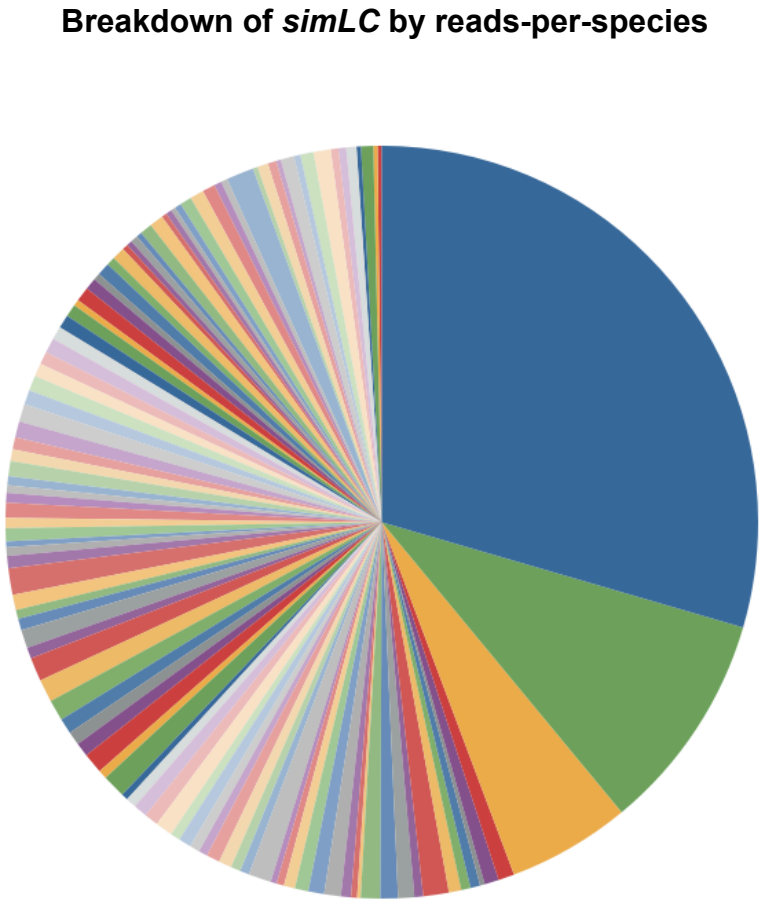


Figure 2.4 *The proportion of reads in the* simLC *dataset derived from each of the 112 species and strains present. A key is provided on the opposite page.*

Materials and Methods

Preparation of *A. thaliana* fragments for isochore clustering analysis

The complete sequence of chromosome 1 of *Arabidopsis thaliana* (NC_003070.9, 30.4 Mbp, genomic GC content ~36%), obtained from the NCBI Genome database, was divided into fragments of mean length 1 kb using the *perl* script 'shortSeqCutter.pl', reproduced in the Appendix. These sequence fragments were labelled according to the isochore/non-isochore region from which they originated along the chromosome. A breakdown of these regions is as follows (from Zhang and Zhang 2004): GC isochore, 0-9.74 Mbp; AT isochore, 9.74-13.48 Mbp; Centromeric isochore (Cen), 14.15-14.9 Mbp; non-isochoric, all other sequence.

Dataset preparation - "Dataset 1"

The complete genome of *Esherichia coli* HS (NCBI accession number NC_009800.1, 4.6 Mbp, GC content ~51%) and *Aspergillus fumigatus* (NC_007197, 29.4 Mbp, GC content ~50%) and the complete sequence of chromosome 1 of *Arabidopsis thaliana* (NC_003070.9) were obtained from the NCBI Genome database and used to generate 60,000 short sequence fragments, with 20,000 fragments derived from the sequence of each species. Where insufficient sequence was available in one copy of the genome of *E. coli*, the sequence was treated as a circular, infinite repeat.

All non-ACGT characters had been removed from the complete sequences, to prevent these characters interfering with feature generation. Using 'shortSeqCutter.pl' the genome and chromosome sequences were 'cut', starting from the 5' terminus, into shorter fragments $n \pm (0.1*n)$ bp in length, where n = [200, 400, 600, 800, 1000]. Therefore, Dataset 1 could be more appropriately described as a collection of datasets, consisting of sequence fragments of different mean lengths (n bp), derived from the same three genomic and chromosomal sequences.

Randomised sequences were also prepared to correspond to these datasets of sequence fragments. These randomised sequences were produced with the same length distribution as the sequence fragments and the same nucleotide frequencies observed in these datasets. That is, the same relative frequency of A, C, G and T was observed across all sequence fragments in the

corresponding ‘true’ dataset. These datasets of randomised sequences were generated using the *perl* script ‘randomSeqWriter.pl’ reproduced in Appendix A.

All *perl* scripts used in this work were written and annotated by the author, unless otherwise stated.

Dataset preparation - simLC

The sequence dataset ‘simLC’ (Mavromatis, Ivanova et al. 2007) was used for feature evaluation. The dataset used contains sequencing reads from 112 different species. Due to difficulties encountered when obtaining the data: the sequencing reads from *Xylella fastidiosa* Ann-1 were absent from the data used in this work.

A full breakdown of species and corresponding sequence counts, and other additional information on the breakdown of the dataset is available at [http://fames.jgi-psf.org./cgi-bin/dataset_desc.pl?dataset=sludge] (the row of details corresponding to *X. fastidiosa* Ann-1 should be disregarded in relation to this work), and is also reproduced in Appendix B.

Any non-ACGT characters were removed from the sequences prior to use.

Generation of feature vectors

Files containing GC, IND, OFDEG and TNF features and their combinations were generated from sequences in FASTA format using the *perl* scripts ‘featureWriter.pl’ and ‘featureComboWriter.pl’, reproduced in Appendix A. All sequences were filtered to remove any non-ACGT characters.

Clustering - CLARA

Clustering of feature vectors was performed with CLARA (Clustering LARge Applications) (Kaufman and Rousseeuw 1990), using an implementation available in the *R* package *cluster*. Default settings were used unless otherwise stated. CLARA analysis of feature vector files was implemented using the *perl* script ‘claraAnalysisMulti.pl’, reproduced in Appendix A.

Clustering - evaluation by precision and recall

Effectivity of clustering was measured by two statistics, precision and recall (Kelley and Salzberg 2010).

Precision (Pr) and recall (Rc) of clustering is calculated for each cluster as follows:

- The predominant class of data within the cluster is determined as the class represented by the largest number of datapoints in the cluster.
- The precision value of the cluster is calculated as the proportion of the total datapoints contained within the cluster that belong to this predominant class.
- The recall value of the cluster is calculated as the proportion of the total datapoints belonging to this predominant class that are contained within the cluster.

This is perhaps best illustrated with an example similar to the experiments carried out in this work. Let our example dataset for clustering contain 100 sequences from three different species, species A, species B and species C, in a ratio of 2:1:1 such that 50 sequences are derived from A and 25 each from B and C. After clustering into three groups, we might observe the following results:

Table 2.1 *Clustering of a 100-sequence dataset, derived from three species: A, B and C.*

Species	Cluster 1	Cluster 2	Cluster 3
A	**35**	5	10
B	0	**25**	0
C	10	0	**15**

For each cluster, we identify the most common class (species) of datapoint (sequence) present. These sequence counts are highlighted in bold above, showing that the predominant species for cluster 1 is species A, B for cluster 2 and C for cluster 3.

The precision of clustering for each cluster is the proportion of all sequences in the cluster that belong to the predominant species:

35/(35+0+10) = 35/45 = 77.78% for cluster 1,

25/(5+25+0) = 25/30 = 83.33% for cluster 2, and

15/(10+0+15) = 15/25 = 60.00% for cluster 3.

The recall of clustering for each cluster is the proportion of all sequences in the dataset belonging to the predominant species that are contained the cluster. So,

if the dataset contains 50 sequences from species A, and 25 from B and C, the recall of the three clusters is calculated as follows:

35/50 = 70% for cluster 1,

25/25 = 100% for cluster 2, and

15/25 = 60% for cluster 3.

Analysis of cluster files to produce precision and recall statistics was implemented using the *perl* scripts ‘claraResultsSummariser.pl’ and ‘avePRwriter.pl’, reproduced in Appendix A.

Where clustering was evaluated at a higher level of taxonomy (genus, family etc.), precision and recall statistics were calculated using classifications of each sequence at this level.

Results

Clustering of *Arabidopsis thaliana* isochores

Figure 2.5 shows the results of clustering of 1 kbp fragments of *A. thaliana* chromosome 1 into four groups, where sequences have been represented by the four individual feature types under investigation. The sequence fragments are labelled according to the isochore from which they originate within the chromosome.

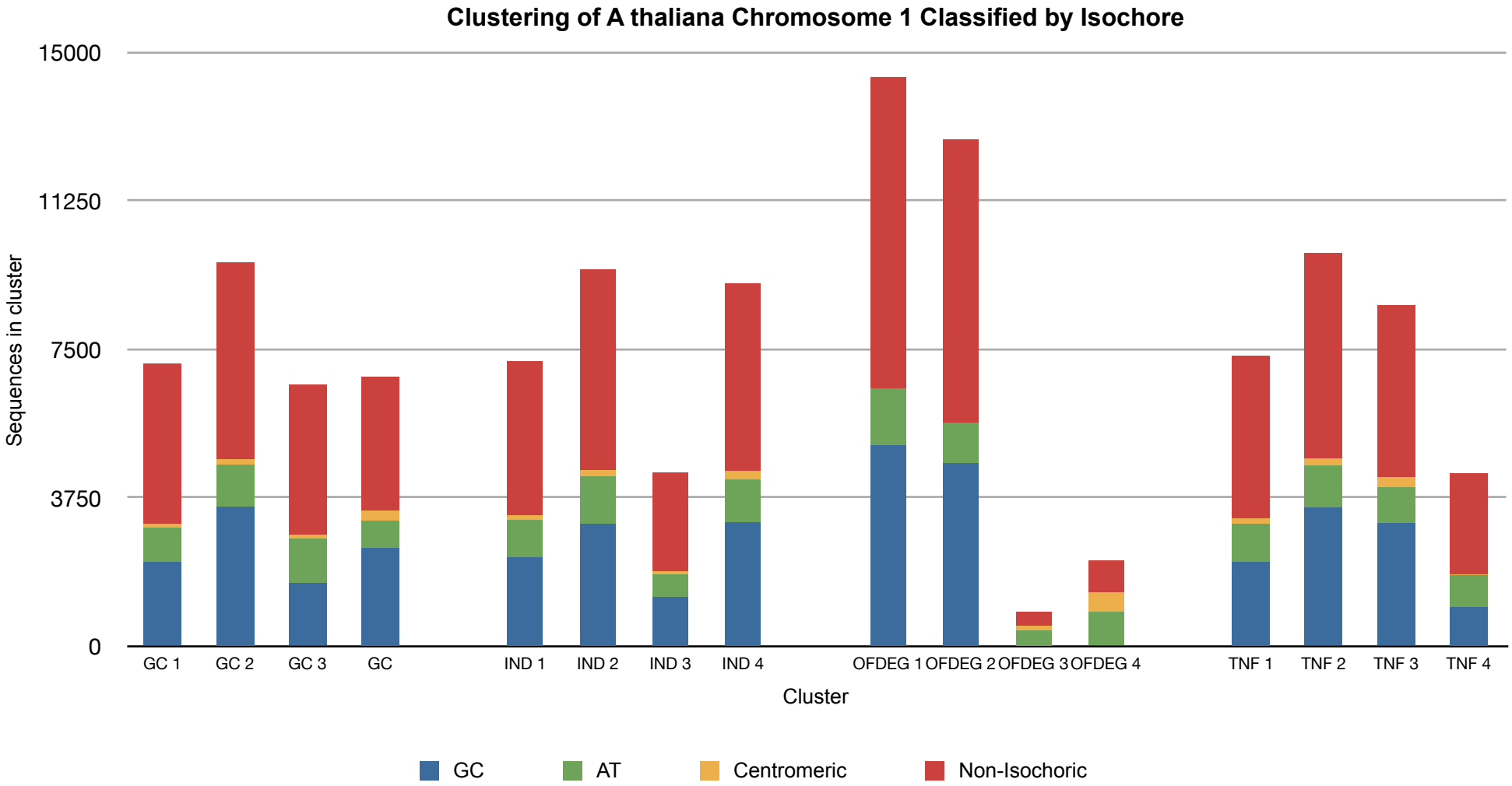
The results in Fig. 2.5 indicated that sequences represented by GC, IND and TNF features were not grouped by isochore, with the clusters produced in CLARA analysis of these feature vectors containing sequences from each isochore in proportions close to those present in the data overall.

The groups produced with OFDEG feature vectors suggested that centromeric (Cen) sequences could be grouped separately from the majority of the remaining fragments under these conditions. However, it was hypothesised that this separation was more likely to be due to the more repetitive nature of centromeric sequence, which would be likely to result in an altered OFDEG profile in these sequences.

The greater degree of repetition in the sequence would cause the error values of the sampled sub-sequences taken during OFDEG value calculation to decrease more rapidly. This difference in gradient, manifested in the OFDEG feature vector, would set these sequences apart from those from the less-repetitive portions of the chromosome, resulting in their isolation in the clustering described here.

This analysis suggested that the presence of isochores in the genome of *A. thaliana* had little effect on the grouping of sequences obtained from each of the four feature types being compared. The read length of 1 kbp and uniform sampling strategy used here were not a good approximation of the sequencing reads that would be obtained through shotgun sequencing of the genome, but were considered acceptable for this preliminary analysis. It was assumed that the long range effect of increased or decreased GC content in isochores would be less likely to be discernible in shorter sequencing reads.

Figure 2.5 *Clustering of sequences from chromosome 1 of* A. thaliana *using CLARA with four types of feature, GC, IND, OFDEG, and TNF. The figure shows the proportion of sequences from the three isochores of the chromosome that was grouped into each of four clusters by CLARA. Sequences were taken from the chromosome at ~1 kb intervals, and classified according to the isochores of the chromosome described by (Zhang and Zhang 2004). Sequences from the relatively GC-rich (GC), relatively GC-poor (AT) and centromeric isochores of the chromosome are labelled, as are sequences from non-isochoric regions.*


Clustering of A thaliana Chromosome 1 Classified by Isochore
15000
11250
7500
3750
0
Sequences in cluster
GC 1
GC 2
GC 3
GC
IND 1
IND 2
IND 3
IND 4
OFDEG 1
OFDEG 2
OFDEG 3
OFDEG 4
TNF 1
TNF 2
TNF 3
TNF 4
Cluster
GC
AT
Centromeric
Non-Isochoric

Clustering of Dataset 1

Tables 2.2 and 2.3 detail the precision and recall values obtained from CLARA clustering analysis of Dataset 1, for each feature type and their combinations generated across a range of mean sequence lengths. The dataset was clustered into three groups.

The values provided in Tables 2.2 and 2.3 are the mean Pr and Rc statistics obtained across all three clusters, for each feature and mean sequence length. Also included in these tables are mean Pr and Rc values obtained from clustering of randomised sequences of Dataset 1 at two different mean sequence lengths.

Table 2.2 *Mean precision values of clusters produced by CLARA analysis of Dataset 1, characterised by four sequence composition features and their combinations. The table shows values obtained from clustering with each feature or combination of features, across a range of increasing mean sequence lengths of Dataset 1, and includes two sets of precision values obtained from clustering of randomised sequences of Dataset 1. The data was separated into three clusters. Mean precision values between 0.6 and 0.7999 are highlighted in yellow, while values greater than or equal to 0.8 (clustering at mean precision of ≥ 80%) are emphasised and highlighted in amber.*

Feature(s)	200 bp	200 bp Random	400 bp	400 bp Random	600 bp	800 bp	1000 bp
GC	0.6408	0.3340	0.6594	0.3379	0.6848	0.7029	0.7155
IND	0.3557	0.3347	0.3874	0.3372	0.3997	0.3968	0.4062
OFDEG	0.5143	0.3365	0.5584	0.3348	0.5721	0.5568	0.4753
TNF	0.7156	0.3360	0.7795	0.3391	**0.8783**	**0.8302**	**0.8879**
GC/IND	0.6348	0.3345	0.6762	0.3381	0.6950	0.6966	0.7152
GC/OFDEG	0.6359	0.3352	0.6756	0.3374	0.6935	0.6997	0.7140
GC/TNF	0.6534	0.3356	0.6699	0.3375	0.6709	0.6802	0.7470
IND/OFDEG	0.3548	0.3355	0.3934	0.3379	0.4124	0.3926	0.3923
IND/TNF	0.5736	0.3344	**0.8441**	0.3367	**0.8791**	0.7628	**0.8921**
OFDEG/TNF	0.7535	0.3377	0.7776	0.3350	0.7280	**0.8757**	**0.8102**
GC/IND/OFDEG	0.6274	0.3347	0.6732	0.3383	0.6776	0.7018	0.6965
GC/IND/TNF	0.6199	0.3342	0.6651	0.3391	0.7456	0.6914	0.7548
GC/OFDEG/TNF	0.6598	0.3358	0.6811	0.3374	0.6539	0.7571	0.6810
IND/OFDEG/TNF	0.5675	0.3344	0.7085	0.3370	**0.8464**	0.7694	0.6764
GC/IND/OFDEG/TNF	0.6542	0.3344	0.6462	0.3389	0.7266	0.7266	0.6728

Table 2.3 *Mean recall values of clusters produced by CLARA analysis of Dataset 1, characterised by four sequence composition features and their combinations. The table shows values obtained from clustering with each feature or combination of features, across a range of increasing mean sequence lengths of Dataset 1, and includes two sets of recall values obtained from clustering of randomised sequences of Dataset 1 (columns 3 and 5). The data was separated into three clusters. Mean recall values between 0.6 and 0.7999 are highlighted in yellow, while values greater than or equal to 0.8 (clustering at mean recall of ≥ 80%) are emphasised and highlighted in amber.*

Feature(s)	200bp	200bp Random	400bp	400bp Random	600bp	800bp	1000bp
GC	0.6132	0.3340	0.6436	0.3380	0.6753	0.6995	0.7124
IND	0.3548	0.3347	0.3788	0.3372	0.3800	0.3870	0.3885
OFDEG	0.4810	0.3364	0.5405	0.3346	0.5536	0.5325	0.4581
TNF	0.6503	0.3350	0.6989	0.3390	**0.8665**	**0.8000**	**0.8789**
GC/IND	0.6280	0.3344	0.6696	0.3378	0.6862	0.6869	0.7110
GC/OFDEG	0.6246	0.3353	0.6718	0.3374	0.6899	0.6951	0.7085
GC/TNF	0.6035	0.3357	0.6522	0.3374	0.6470	0.6541	0.6392
IND/OFDEG	0.3530	0.3357	0.3790	0.3377	0.3793	0.3882	0.3879
IND/TNF	0.5912	0.3344	**0.8286**	0.3350	**0.8653**	0.7350	**0.8906**
OFDEG/TNF	0.7552	0.3376	0.7673	0.3350	0.6518	**0.8715**	0.7549
GC/IND/OFDEG	0.5825	0.3347	0.6535	0.3380	0.6685	0.6984	0.6804
GC/IND/TNF	0.6115	0.3342	0.6449	0.3389	0.7449	0.6755	0.7505
GC/OFDEG/TNF	0.6007	0.3357	0.6080	0.3374	0.6082	0.7562	0.6593
IND/OFDEG/TNF	0.5593	0.3343	0.7122	0.3352	**0.8465**	0.7589	0.6331
GC/IND/OFDEG/TNF	0.6167	0.3343	0.6082	0.3388	0.7127	0.7175	0.6284

Effect of increasing sequence size on clustering quality

A general trend could be identified in the results of CLARA clustering of Dataset 1, of increasing quality of clustering with increasing mean sequence length in the dataset. For the majority of feature types and combinations, the precision and recall values obtained from clustering increased with the stepwise increases in mean sequence length. This effect is particularly apparent in the Pr and Rc values obtained for clustering with GC content only, which displayed a roughly linear increase with increasing mean sequence length.

An exception to this trend was observed between the values obtained from clustering at a mean length of 600 bp and 800 bp, where the quality of clustering with most feature types and combinations was observed to fall slightly. It was concluded that this was unlikely to be indicative of any upper sequence-length limit on clustering quality and may have been the result of chance inclusion of less easily distinguished regions of the genomes sampled as the total sampled area of each genome increased.

Clustering of randomised sequences - Dataset 1

The Pr and Rc values from clustering of randomised sequences of mean length 200 bp and 400 bp (and of randomised sequences generated for the other mean sequence lengths but not shown here), remained at ~33% for all features and combinations and all mean sequence lengths. Given that the dataset consisted of sequences taken from three species in equal proportions, these results were what would be expected if the clustering occurred at random into three clusters of roughly equal size. This proved that any successful clustering of the true biological sequence fragments was the product of the specific order of nucleotides in the sequences, characterised by the feature vectors.

Feature evaluation - Dataset 1

In contrast to that of randomised sequences, the clustering of biological sequence fragments was shown to be more successful, albeit by varying degrees. The results showed that clustering with IND and IND+OFDEG features was only marginally better than those from the same features characterising randomised sequences, and showed little improvement with increasing sequence lengths - the mean Pr and Rc values for these two feature vector types failed to climb far beyond 40%. The clustering quality observed with the

use of OFDEG features was only marginally better, with Pr and Rc values ~55% for all mean sequence lengths except 1000 bp, where a substantial drop in quality to 45% and 47% was observed in mean Pr and Rc of clustering respectively.

Most of the remaining features and combinations returned mean Pr and Rc values of ~66-56% for the shortest set of sequences, with the quality of clustering increasing with sequence length, with some exceptions between 600 bp and 800 bp as mentioned previously. The best quality clustering was observed with TNF, IND+TNF, OFDEG+TNF and IND+OFDEG+TNF feature vectors. These feature combinations returned clusters with mean Pr and Rc values of ~80-90% for longer sequence lengths (600 bp, 800 bp and 1000 bp).

The single best set of clusters were obtained from the use of IND+TNF feature vectors with sequences of mean length 1000 bp (Pr = 89.21%, Rc = 89.06%). The same feature combination also returned markedly higher quality clusters than other feature vectors for sequences of mean length 400 bp (Pr = 84.41%, Rc = 82.86%).

Clustering of *simLC*

Clustering of random and nonrandom sequences

Initially, the *simLC* dataset was separated into 108 clusters in accordance with the total number of different species that were represented by the sequence reads present. Table 2.4 provides a summary of this clustering of the data with CLARA, represented as mean precision and recall values across all clusters and for all feature combinations, and includes the equivalent statistics for clustering of a corresponding dataset of randomised sequences.

Table 2.4 *Mean precision and recall values of clusters produced by CLARA analysis of simLC, characterised by four sequence composition features and their combinations. The table shows values obtained from clustering the sequences with each feature or combination of features and includes precision and recall values obtained from clustering of randomised sequences. The data was separated into 108 clusters, in accordance with the total number of different species represented in the dataset.*

	simLC		simLC Randomised	
Feature(s)	**Precision**	**Recall**	**Precision**	**Recall**
GC	0.2909	0.0340	0.2972	0.0093
IND	0.3055	0.0098	0.2966	0.0093
OFDEG	0.3078	0.0134	0.2918	0.0093
TNF	0.4476	0.1148	0.3369	0.0097
GC/IND	0.3118	0.0386	0.2971	0.0093
GC/OFDEG	0.2931	0.0336	0.2972	0.0093
GC/TNF	0.4166	0.0864	0.3186	0.0094
IND/OFDEG	0.2997	0.0141	0.2959	0.0093
IND/TNF	0.3927	0.0866	0.3335	0.0094
OFDEG/TNF	0.4116	0.0969	0.3333	0.0096
GC/IND/OFDEG	0.2932	0.0401	0.2969	0.0093
GC/IND/TNF	0.3560	0.0782	0.3140	0.0095
GC/OFDEG/TNF	0.3941	0.0771	0.3223	0.0094
IND/OFDEG/TNF	0.4116	0.0768	0.3111	0.0094
GC/IND/OFDEG/TNF	0.4109	0.0734	0.3002	0.0093

Clustering of randomised sequences - *simLC*

These results provided a comparison between the quality of clustering achieved with the *simLC* dataset and randomised sequences. When applied to randomised sequences, all fifteen feature types and combinations returned clusters with mean Pr values of ~29-33% and mean Rc values of ~0.93-0.97%. The predominant species in the dataset - that is, the species that is most represented in the sequences comprising the dataset - *Rhodopseudomonas palustris* (RP) contributed 30,739 sequences, or 31.6% of the total sequences clustered.

As expected, the Pr and Rc statistics obtained here were consistent with those that would be obtained from uniform clustering of the data at random. The Pr statistics were calculated based on the best-represented species in each cluster, which after random clustering would be RP accounting for ~31.6% of sequences in the cluster. The Rc statistics of ~0.92-0.93% for all features indicated that each cluster contained ~1/108 of the total sequences from the best-represented species in the dataset.

The Pr and Rc values obtained from clustering of the non-random, 'true' *simLC* sequences differ from those obtained from the randomised data, indicating that this clustering was not due to randomised grouping of the data.

Quality of clustering at species-level resolution

Variation was observed in the quality of clustering with different features and combinations. The highest-quality clustering at species level was observed with TNF feature vectors (mean Pr = 44.76%; mean Rc = 11.48%), while GC and GC+OFDEG feature vectors returned the lowest-quality clustering, with mean Pr values below those obtained with randomised sequences (29.09% and 29.31% respectively) and mean recall values only marginally greater than those from clustering of randomised sequences with the same features (3.40% and 3.36%).

The statistics detailed in Table 2.4 indicated that clustering at the species level was poor for all features and combinations. On average, fewer than half of the sequences - in many cases fewer than one third - in any given cluster generated in this analysis originated from a single genome, and these sequences accounted for only a very small proportion (~1-10%) of the total

sequences from that genome in the dataset.

With a relatively large number of clusters, and such closely related sequences, it was concluded that species-specific separation was outside the capabilities of this approach.

Nevertheless, a consistency was observed between the quality of clustering returned with the use of certain feature vectors in this analysis and in the equivalent analysis with Dataset 1. The TNF and OFDEG+TNF feature vectors in particular returned relatively successful clustering in both cases.

Quality of clustering at higher levels of taxonomy

In view of the relatively poor separation observed in clustering *simLC* at species-level resolution, further investigation was performed comparing the quality of clustering obtained at lower resolution, where the data was clustered into a number of groups determined by the number of different classes at each taxonomic level from Genus, through Family and Order, to Class. A full taxonomic breakdown of the dataset is provided in Appendix B.

At each level, the data was clustered as before with all feature types and combinations for comparison and the results were evaluated using labels at the same level. The results of this analysis are detailed in Tables 2.5 and 2.6.

Table 2.5 *Mean precision values of clusters produced by CLARA analysis of simLC at a range of taxonomic levels, characterised by four sequence composition features and their combinations. The table shows values obtained from clustering the sequences with each feature or combination of features, calculated based on the taxonomic groups of organisms present in the simLC dataset. The number of clusters that the data was separated into is given in the header to each column, with the corresponding phylogenetic level. Cluster numbers were determined in accordance with the total number of different groups represented in the dataset at each level.*

Feature(s)	108 (Species)	79 (Genus)	57 (Family)	39 (Order)	18 (Class)
GC	0.2909	0.3256	0.3998	0.4028	0.5037
IND	0.3055	0.3376	0.4328	0.4366	0.4797
OFDEG	0.3078	0.3346	0.4336	0.4342	0.5078
TNF	0.4476	0.4361	0.4878	0.5077	0.5436
GC/IND	0.3118	0.3656	0.4340	0.4674	0.5149
GC/OFDEG	0.2931	0.3240	0.3823	0.3950	0.4792
GC/TNF	0.4166	0.3977	0.4455	0.4835	0.4567
IND/OFDEG	0.2997	0.3530	0.4327	0.4423	0.4659
IND/TNF	0.3927	0.3856	0.4255	0.4931	0.5107
OFDEG/TNF	0.4116	0.4164	0.4712	0.4842	0.546
GC/IND/OFDEG	0.2932	0.3554	0.4190	0.4713	0.5305
GC/IND/TNF	0.3560	0.3650	0.4195	0.4760	0.5113
GC/OFDEG/TNF	0.3941	0.3894	0.4120	0.4767	0.4838
IND/OFDEG/TNF	0.4116	0.4132	0.4285	0.4487	0.4917
GC/IND/OFDEG/TNF	0.4109	0.3694	0.4162	0.4466	0.5038

Table 2.6 *Mean recall values of clusters produced by CLARA analysis of simLC at a range of taxonomic levels, characterised by four sequence composition features and their combinations. The table shows values obtained from clustering the sequences with each feature or combination of features, calculated based on the taxonomic groups of organisms present in the simLC dataset. The number of clusters that the data was separated into is given in the header to each column, with the corresponding phylogenetic level. Cluster numbers were determined in accordance with the total number of different taxonomic groups represented in the dataset at each level.*

Feature(s)	108 (Species)	79 (Genus)	57 (Family)	39 (Order)	18 (Class)
GC	0.0340	0.0385	0.0526	0.0739	0.1337
IND	0.0098	0.0143	0.0175	0.0256	0.0556
OFDEG	0.0134	0.0216	0.0274	0.0408	0.0838
TNF	0.1148	0.1106	0.1167	0.1160	0.1549
GC/IND	0.0386	0.0398	0.0540	0.0673	0.1297
GC/OFDEG	0.0336	0.0401	0.0528	0.0675	0.1291
GC/TNF	0.0864	0.0756	0.0949	0.0938	0.1573
IND/OFDEG	0.0141	0.0176	0.0175	0.0273	0.0555
IND/TNF	0.0866	0.0690	0.0732	0.0905	0.1265
OFDEG/TNF	0.0969	0.1060	0.1198	0.1428	0.1717
GC/IND/OFDEG	0.0401	0.0417	0.0494	0.0634	0.1223
GC/IND/TNF	0.0782	0.0558	0.0558	0.0733	0.149
GC/OFDEG/TNF	0.0771	0.0902	0.1033	0.1070	0.1606
IND/OFDEG/TNF	0.0768	0.0676	0.0767	0.0878	0.1138
GC/IND/OFDEG/TNF	0.0734	0.0677	0.0765	0.0890	0.1257

This clustering analysis applied to randomised sequences (results not shown here) was found to produce results similar to those obtained at the previously, with clusters produced at random from the dataset and no specific grouping observed for any feature type or sequence classification.

In analysis of the true *simLC* dataset, the quality of clustering was found to improve with decreasing specificity of sequence classification (i.e. at higher levels of taxonomy). Some of this improvement in clustering quality was accounted for by the decrease in the number of clusters generated at each resolution, as was observed in the results from the grouping of randomised sequences. With the exception of clustering with IND, OFDEG and IND +OFDEG feature vectors, quality of clustering *simLC* exceeded considerably that of randomised sequences.

As observed in the results discussed previously the feature vector types that consistently produced the best clustering here were TNF and OFDEG+TNF. Clustering with GC+OFDEG+TNF features produced clusters with relatively good mean Rc, but conversely poor mean Pr values at each taxonomic level. In each case, these clustering statistics were inferior to those returned from clustering with OFDEG+TNF, the same feature vectors discounting GC content.

On average, clusters produced and evaluated at the Class level - the highest taxonomic level investigated here - with OFDEG+TNF feature vectors contained 17.17% of the total sequences in the dataset that originated from the predominant Class in the cluster (Rc), which constituted 54.6% of the sequences in the cluster (Pr). Similar results were obtained from TNF feature vectors (mean Pr = 54.36%; mean Rc = 15.49%).

Quality of clustering using a hybrid labeling system

The results discussed previously for Dataset 1 indicated that a selection of the sequence features implemented here could be used to cluster a simple, equally proportioned dataset into groups that largely consisted of sequences from one species. When feature vectors such as TNF and OFDEG+TNF were used, each cluster produced corresponded in the most part to a single class (species) of the sequences in the dataset.

The same accuracy of clustering was not observed with *simLC*, where more sequences were grouped from more, more closely related species in vastly

different proportions were grouped. The results given in Tables 2.4-2.6, and discussed above, indicated that the accurate grouping of the sequences from a single class (species, genus, family etc.) in the dataset, into a single cluster was beyond the scope of the feature vectors compared here.

simLC was more complex than the type of multi-species sequencing datasets of particular interest here, obtained from samples containing a smaller number of species that could be expected to be more evolutionarily distinct. In this context, if one or several of the clusters produced from *simLC* was found to be considerably enriched with sequences from one class within the dataset, this could indicate that similar results could be expected when the features are applied to clustering of true sequencing read data.

After consideration had been made of these results in the context of the composition of the *simLC* dataset, clustering was repeated with the data being separated into five clusters, based on and evaluated with a hybrid classification of the sequencing reads.

This hybrid classification placed each sequence read in one of five groups according to whether the read originated from one of the three most well-represented species in the dataset - RP, *Bradyrhizobium* sp. BTAi1 (BR1), and *Cytophaga hutchinsonii* ATCC 33406 (CPH), sequences from which constituted almost half of those present in the dataset - or from any of the other 105 species. These reads were split into two further classes based on their phylum: placing reads from those 39 species belonging to the proteobacteria, including RP and BR1, in one group and those from the remaining 70 species in another. The relative proportions of the dataset according to this hybrid taxonomic classification are presented in Figure 2.6 and Table 2.7.

This hybrid classification was chosen to provide an indication of the separation and grouping that could be achieved across the range of abundances in simLC, allowing the clustering to be interpreted at the species level for the three most prevalent organisms, and at phylum level for the large number of other species also present in much smaller proportions. Using this system of classification, the wider patterns of grouping in this relatively complex dataset can be more easily identified: the specific separation of the reads taken from species represented in great numbers can be viewed alongside the more general patterns of

taxonomic grouping throughout the dataset.

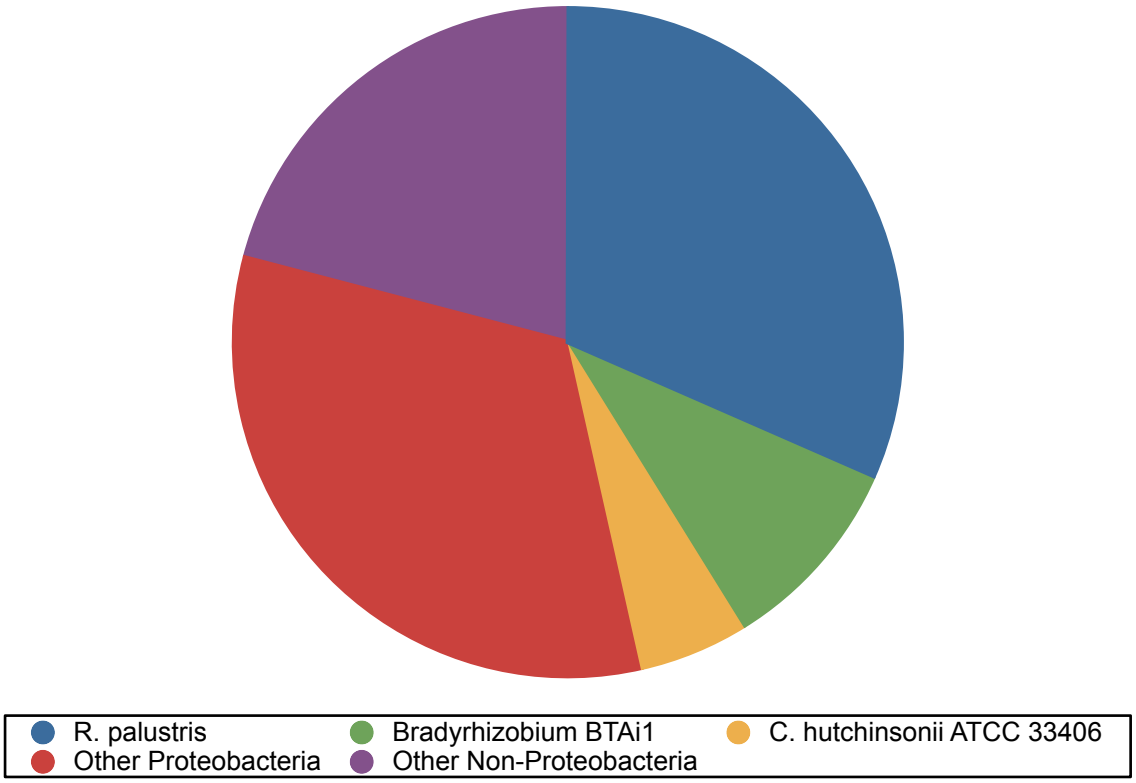


Figure 2.6 *A breakdown of sequences in* simLC*, showing the proportions derived from each of the three most well-represented species (*R. palustris*,* Bradyrhizobium *sp. BTAi1 and* C. hutchinsonii *ATCC 33406), and from the remaining 105 species represented, divided between those that belong to the phylum proteobacteria (the group containing* R. palustris *and* Bradyrhizobium *sp. BTAi1), and those that do not.*

Table 2.7 *The total number of sequences derived from each of the five groups in the hybrid classification of* simLC.

Organism/Taxonomic Group	Number of sequencing reads in dataset
R. palustris	30739
***Bradyrhizobium* sp. BTAi1**	9272
***C. hutchinsonii* ATCC 33406**	5154
Other Proteobacteria	31788
Other Non-Proteobacteria	20302
Cluster Totals	97255

This hybrid classification was used to allow a much more straightforward interpretation of the clustering achieved with each feature and combination. As the dataset was dominated by sequences from a small fraction of the organisms represented overall - the three best-represented species account for almost one half of sequences in the dataset - a separation of the sequences from these organisms in the clustering would indicate that a similar separation might be possible with sequencing data from a sample containing fewer species overall.

The results of grouping *simLC* into five clusters with CLARA, in accordance with the number of different classes of sequence in the data, are depicted by the sets of comparative pie charts in Fig. 2.7. There are fifteen sets of charts in total, with each set corresponding to a different feature vector type and containing one pie chart for each cluster produced from the dataset. The total area of each pie chart is directly proportional to the total number of sequences contained within the cluster it represents.

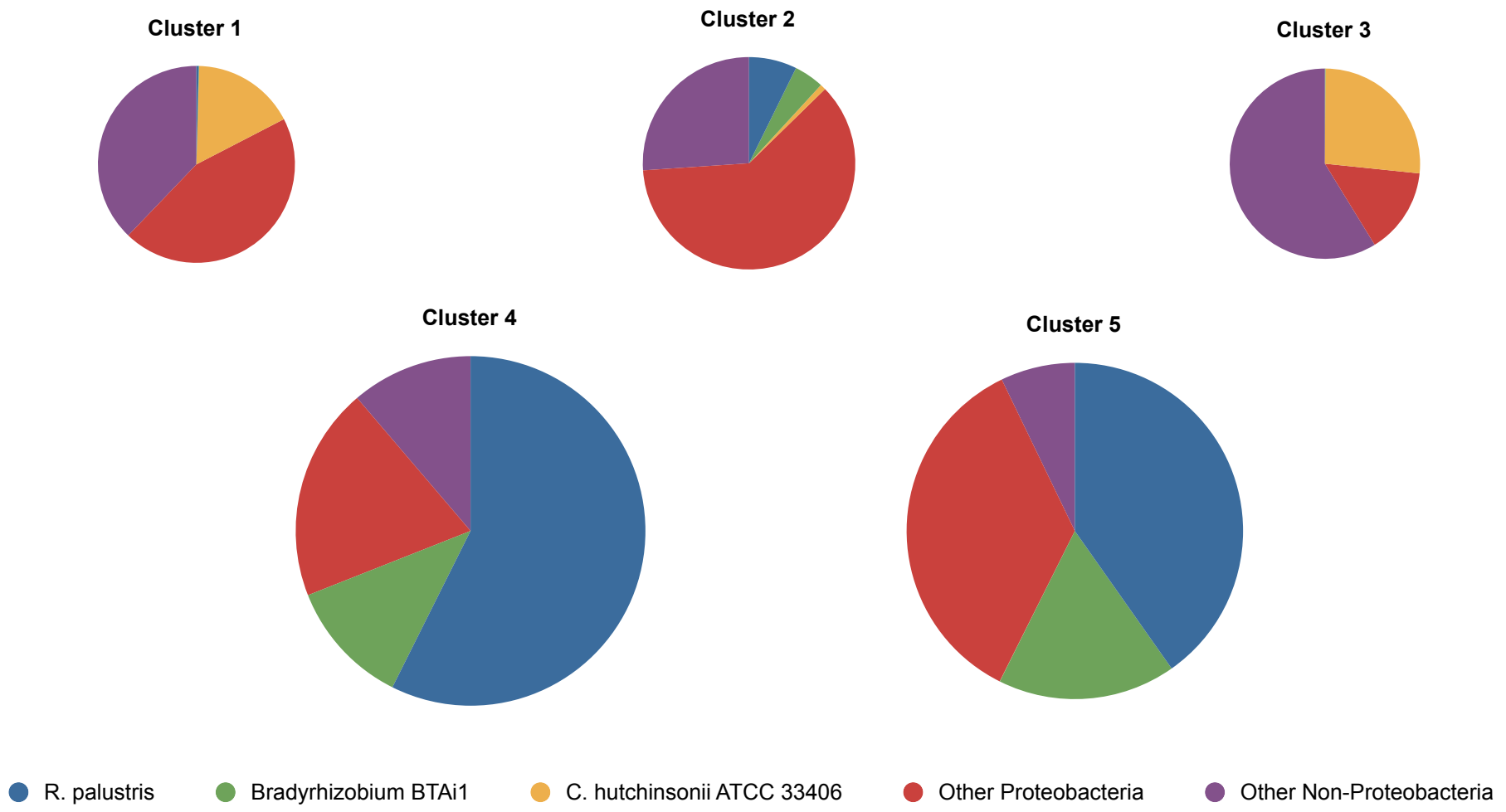
It would not be reasonable to expect clustering to produce a perfect grouping of the reads into the five classes assigned in the data. Any grouping and separation achieved for the dataset could not be expected to take place selectively at the species and phylum level, depending on the origin of each read. As such, a perfect clustering into each of the five classes is unattainable for the method, and the results reported with this classification do not provide great insight into the overall quality of grouping throughout the dataset.

Instead, this hybrid classification of the dataset is intended to provide a clearer visualisation of the general trends in clustering throughout the dataset. The clustering can be expected to remain virtually identical after each run if starting conditions and parameters are conserved, and the classification of sequences within the dataset determines the lines along which the effects of this clustering are interpreted. The results presented previously, where cluster analysis was performed at different taxonomic levels, proved difficult to interpret for the wide range of representation and relatedness for the different contributing organisms. The hybrid classification used here was intended to provide an indication of the grouping of species from the three most predominant species in the dataset, while also allowing the general trends in grouping of sequences from the many remaining, and much less well-represented, species.

Figure 2.7(i) - 2.7(xv) *Comparative pie charts describing the distribution of sequence reads in simLC between five clusters generated by CLARA analysis with each sequence feature and their combinations. Each set of pie charts corresponds to a feature set. The differently coloured sections of each chart correspond to the proportion of sequence reads in the cluster that are derived from one of the three most well-represented species in the dataset, or from any of the remaining 105 species divided by phylum into groups of Proteobacteria and non-Proteobacteria. The pie charts are comparable by size - the area of each chart if directly proportional to the number of sequence reads contained in the cluster that it represents.*

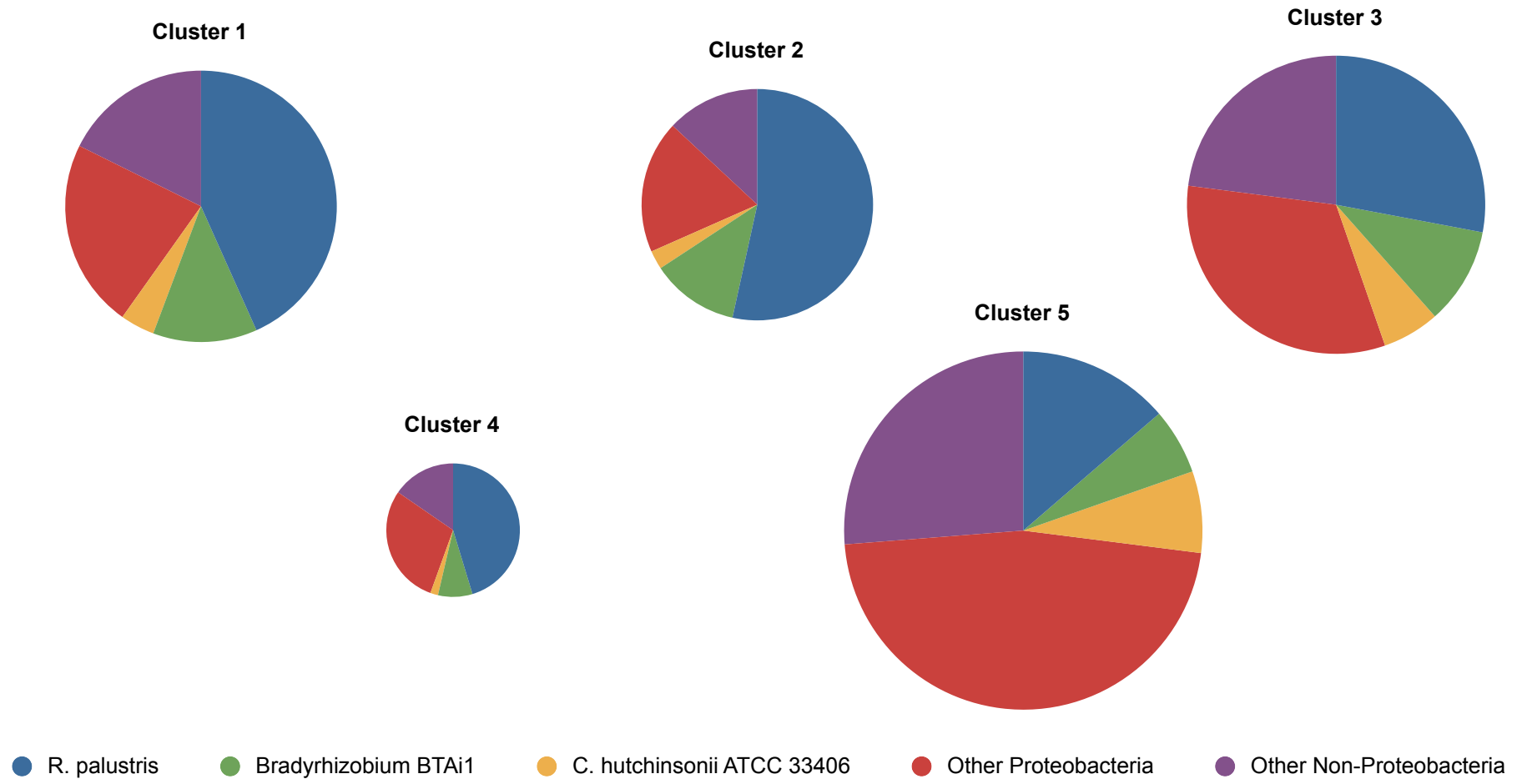
GC

(i)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

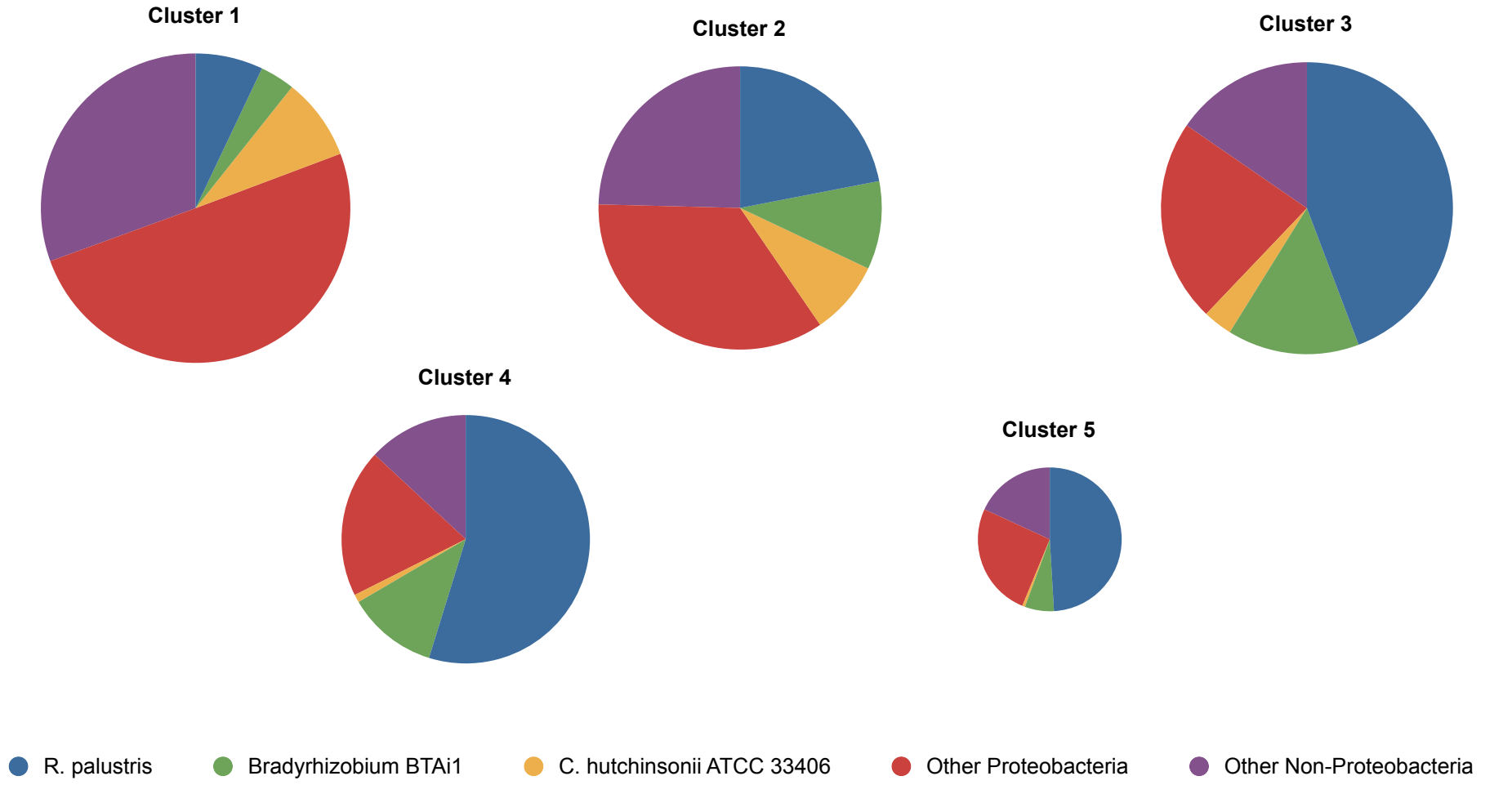
IND

(ii)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

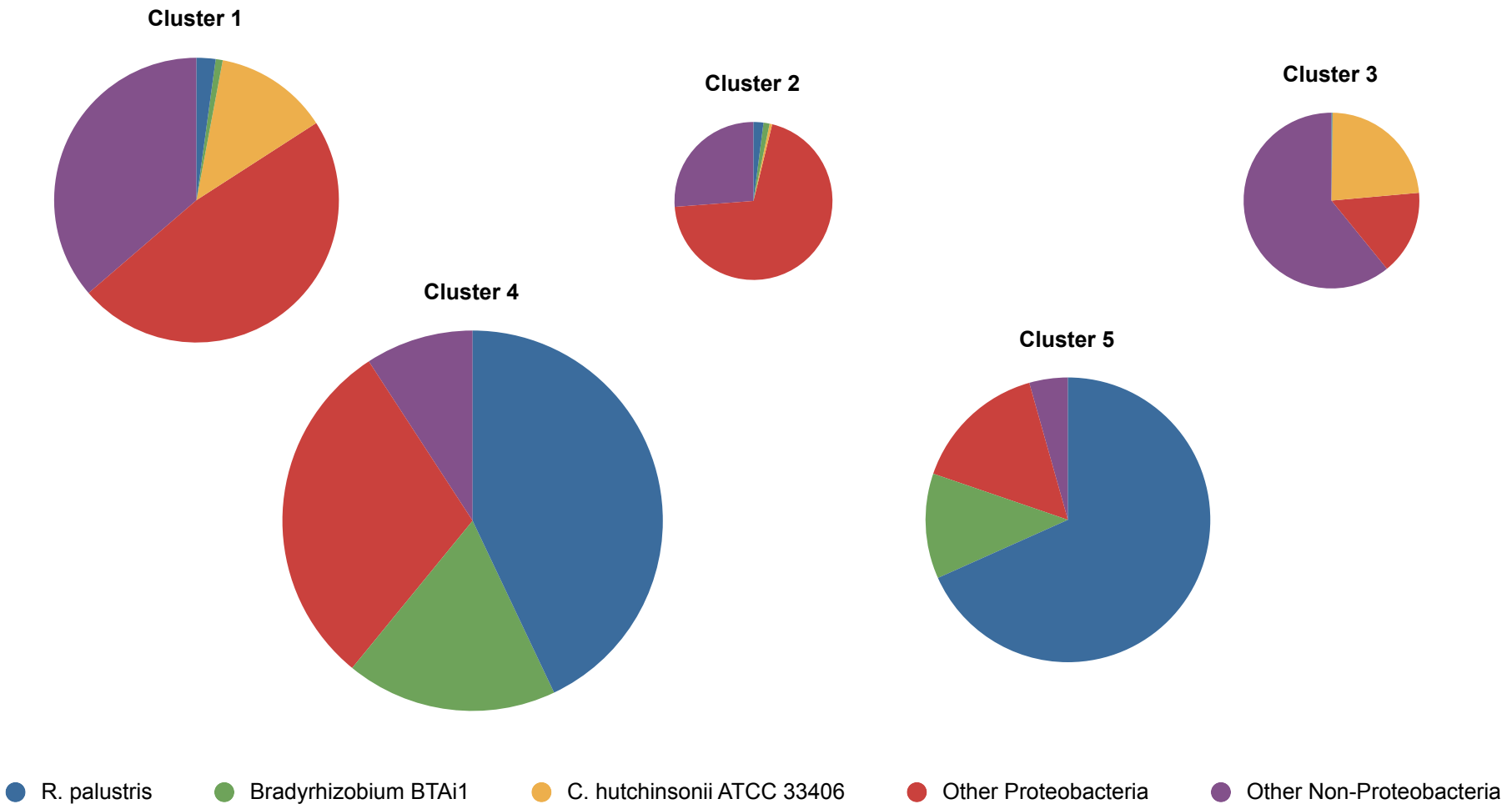
OFDEG

(iii)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

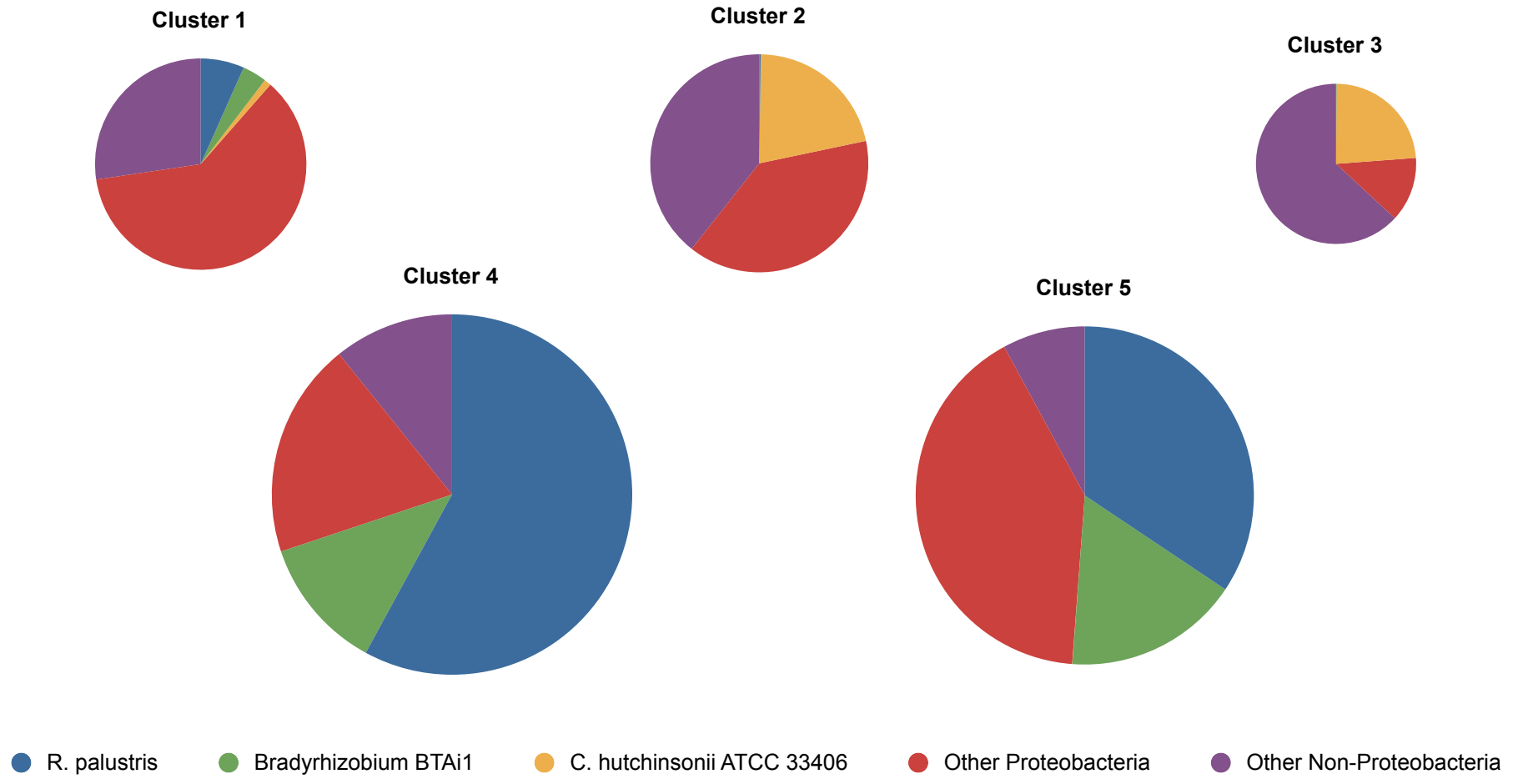
TNF

(iv)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

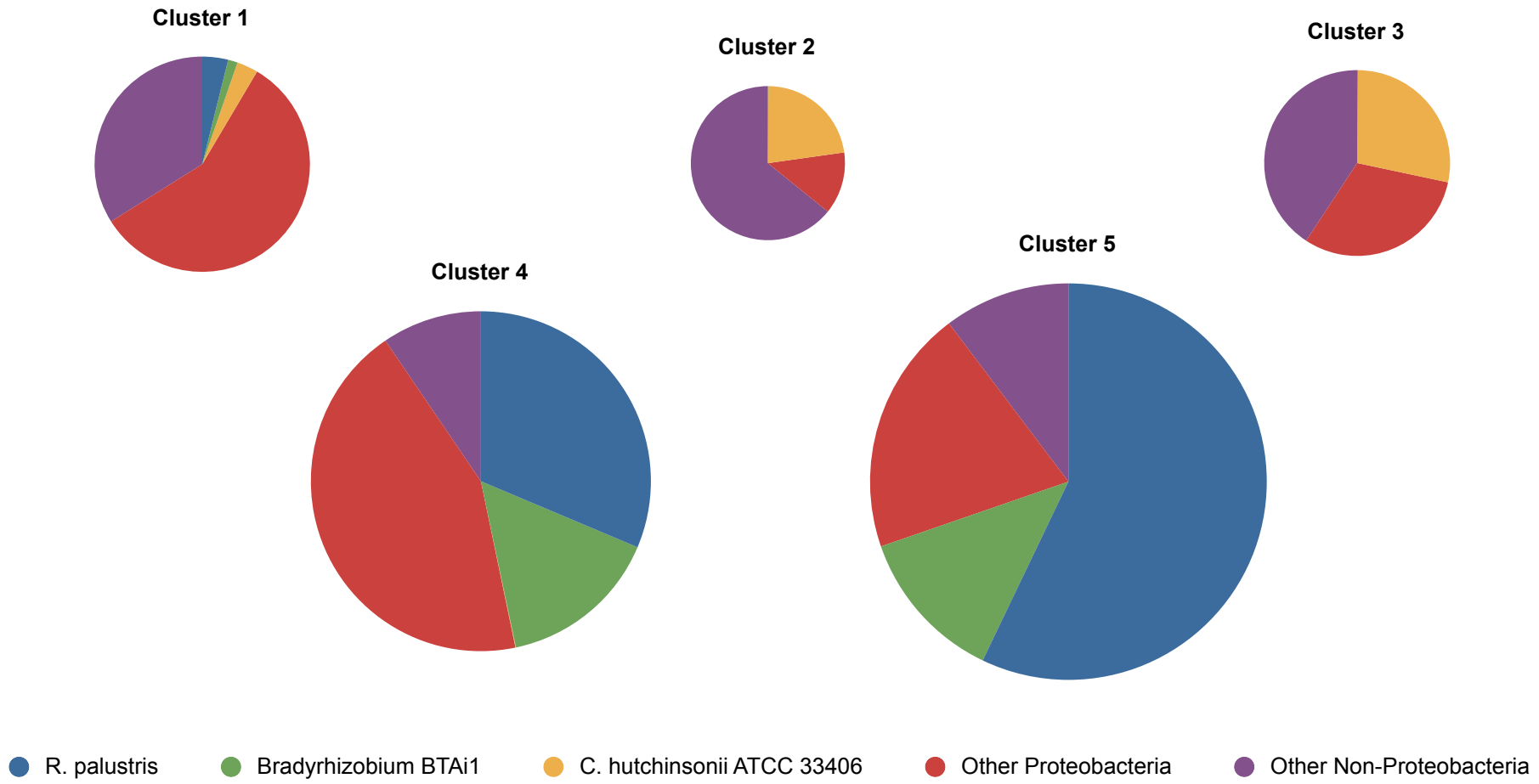
GC + IND

(v)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

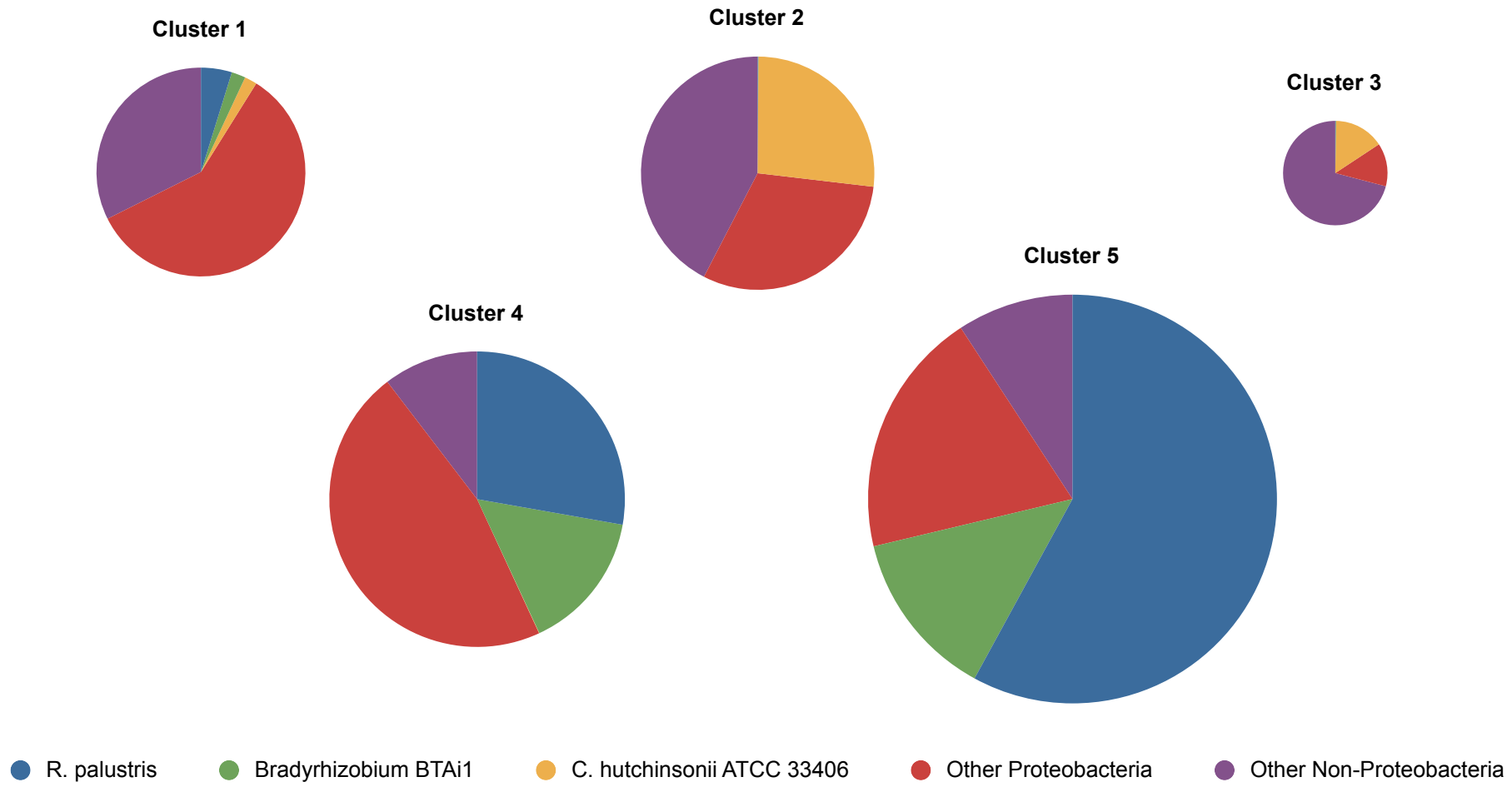
GC + OFDEG

(vi)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

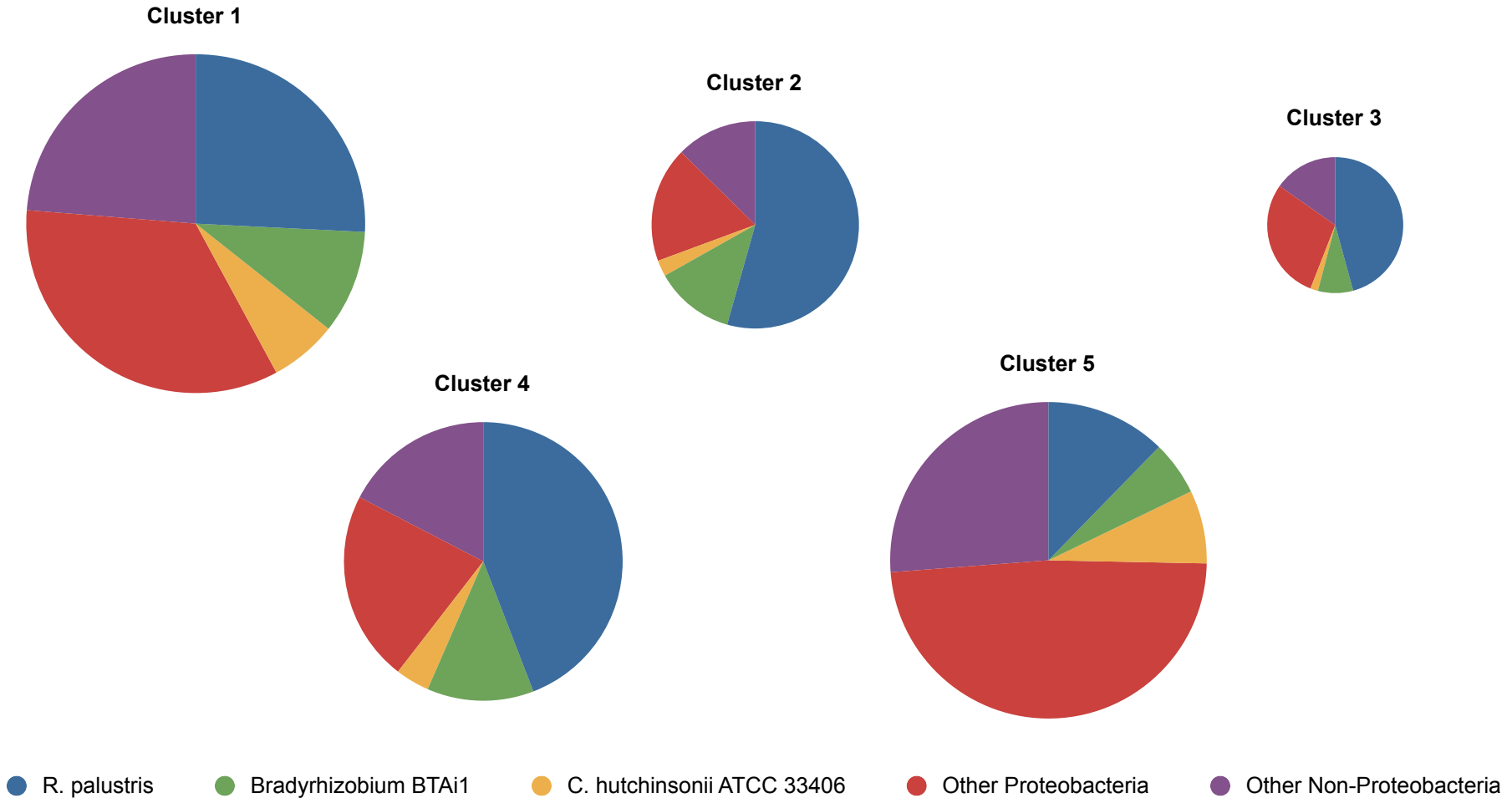
GC + TNF

(vii)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

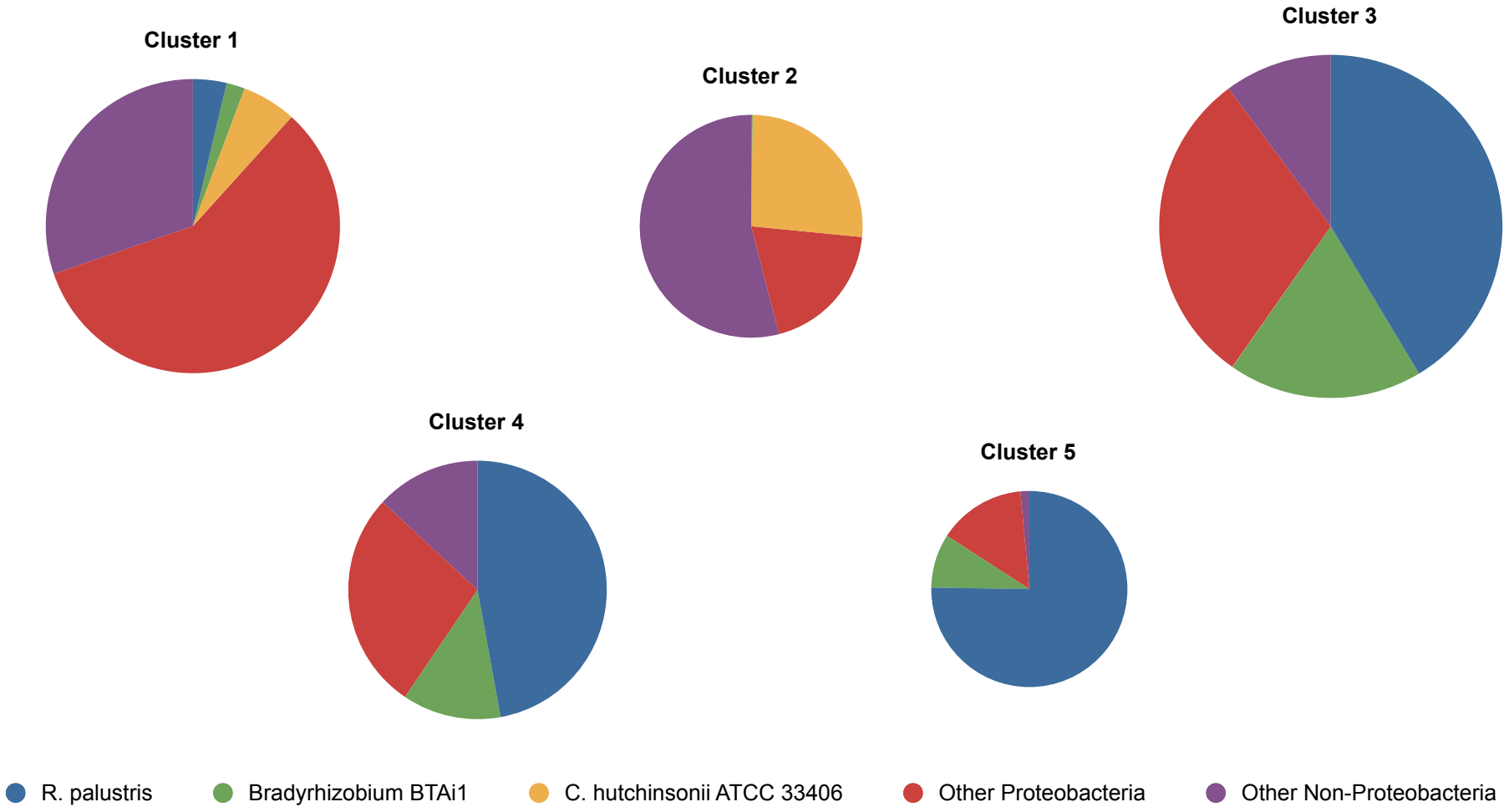
IND + OFDEG

(viii)



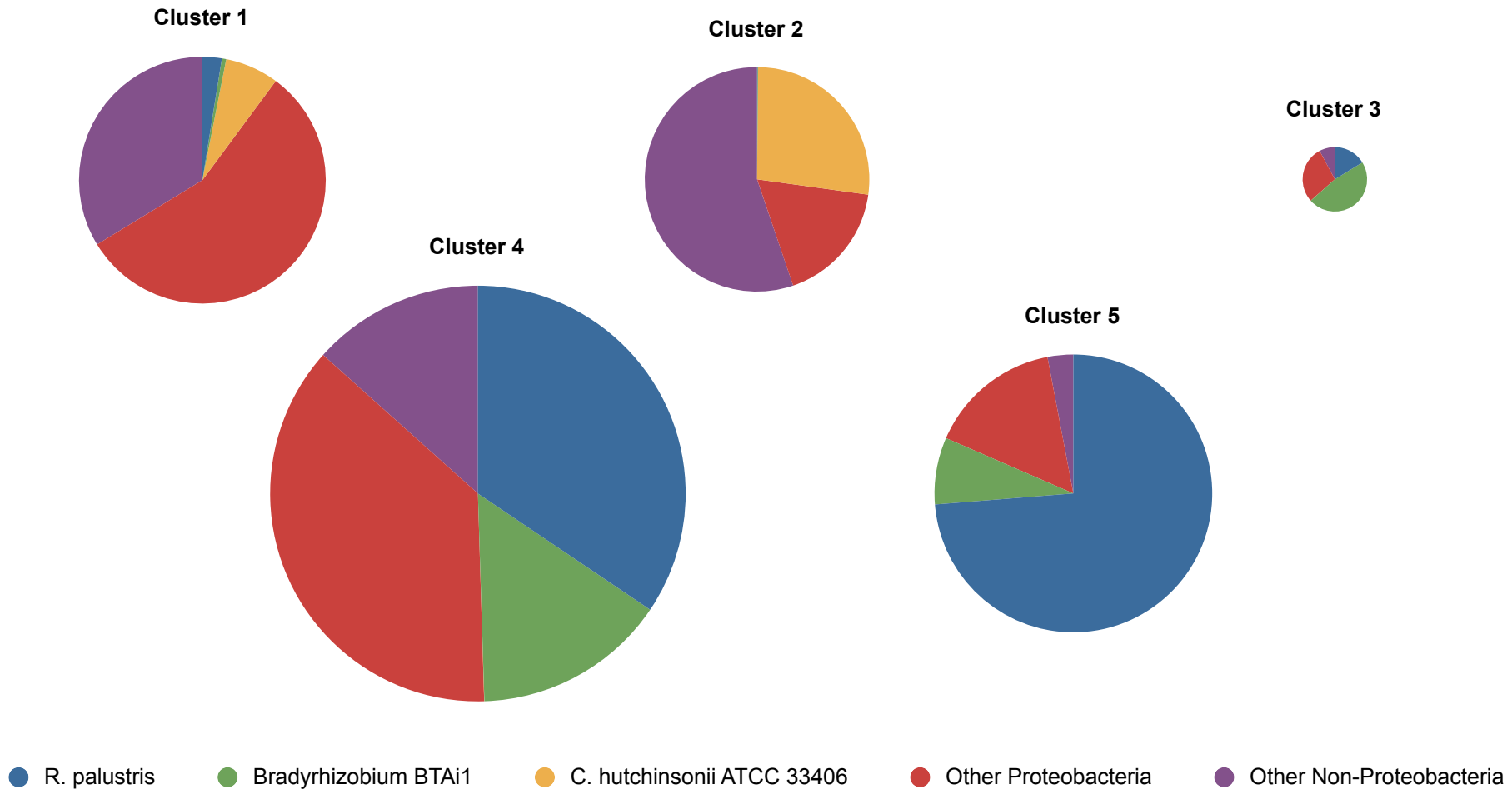
IND + TNF

(ix)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

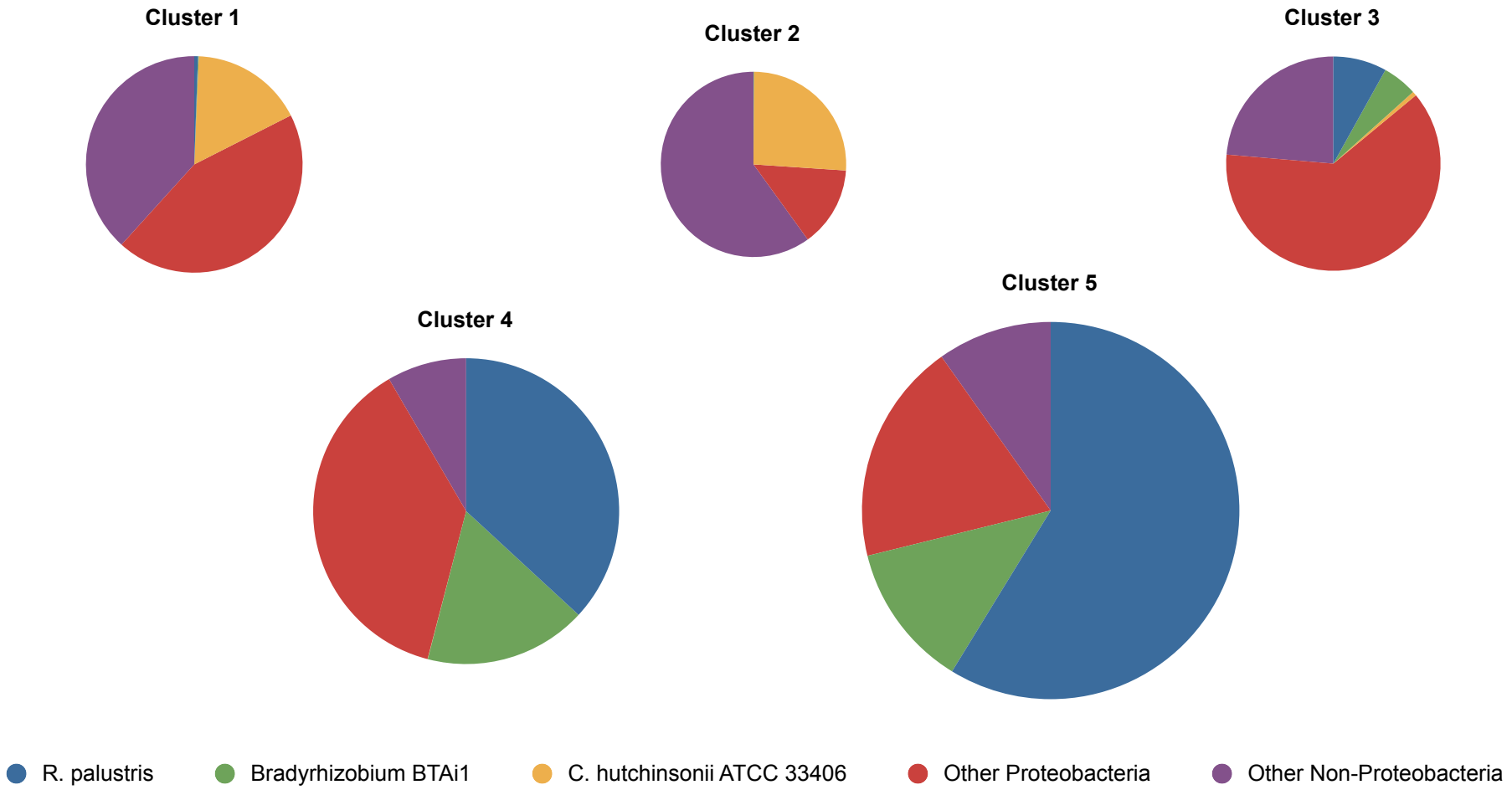
OFDEG + TNF

(x)


Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
R. palustris
Bradyrhizobium BTAi1
C. hutchinsonii ATCC 33406
Other Proteobacteria
Other Non-Proteobacteria

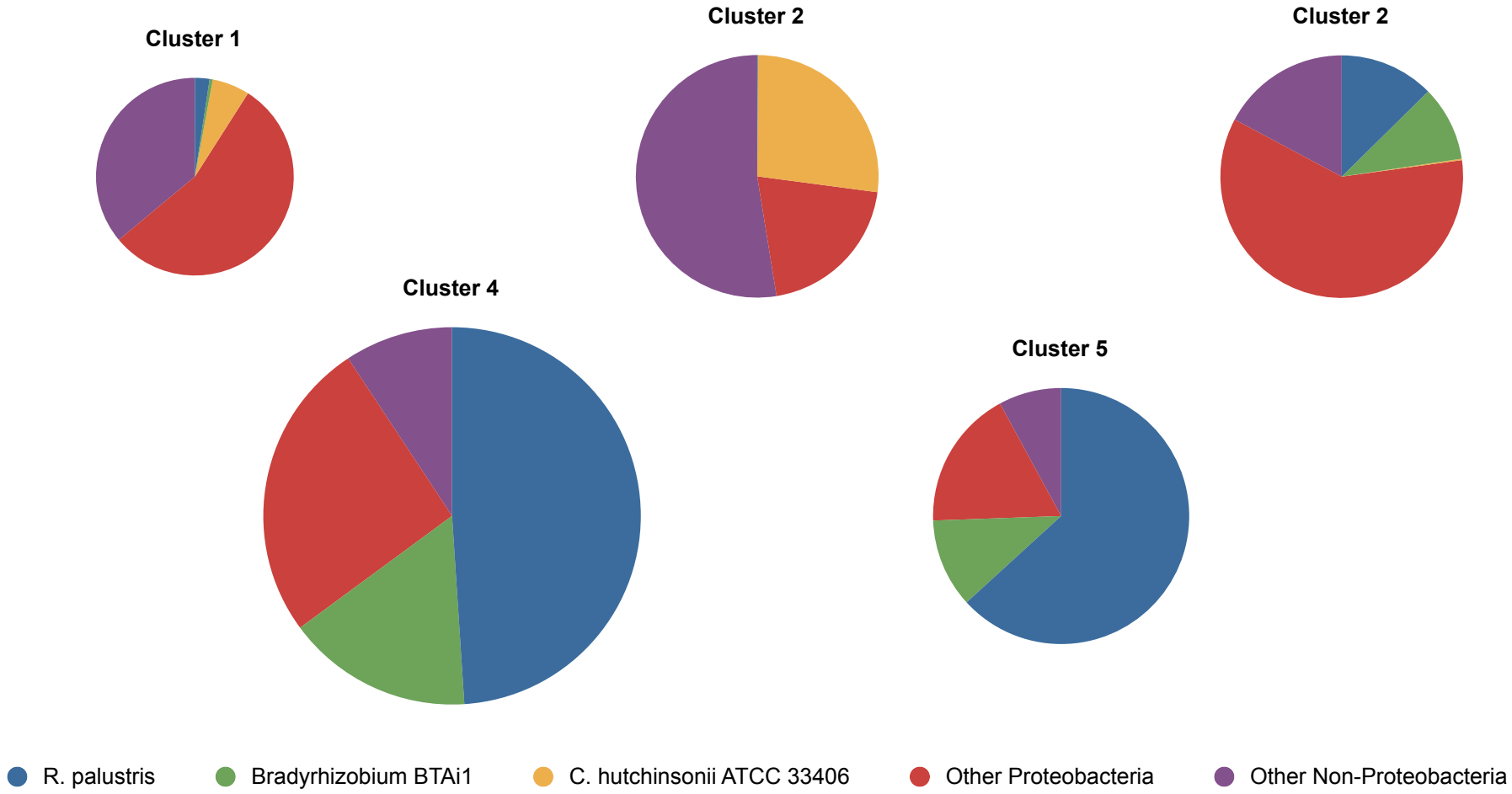
GC + IND + OFDEG

(xi)



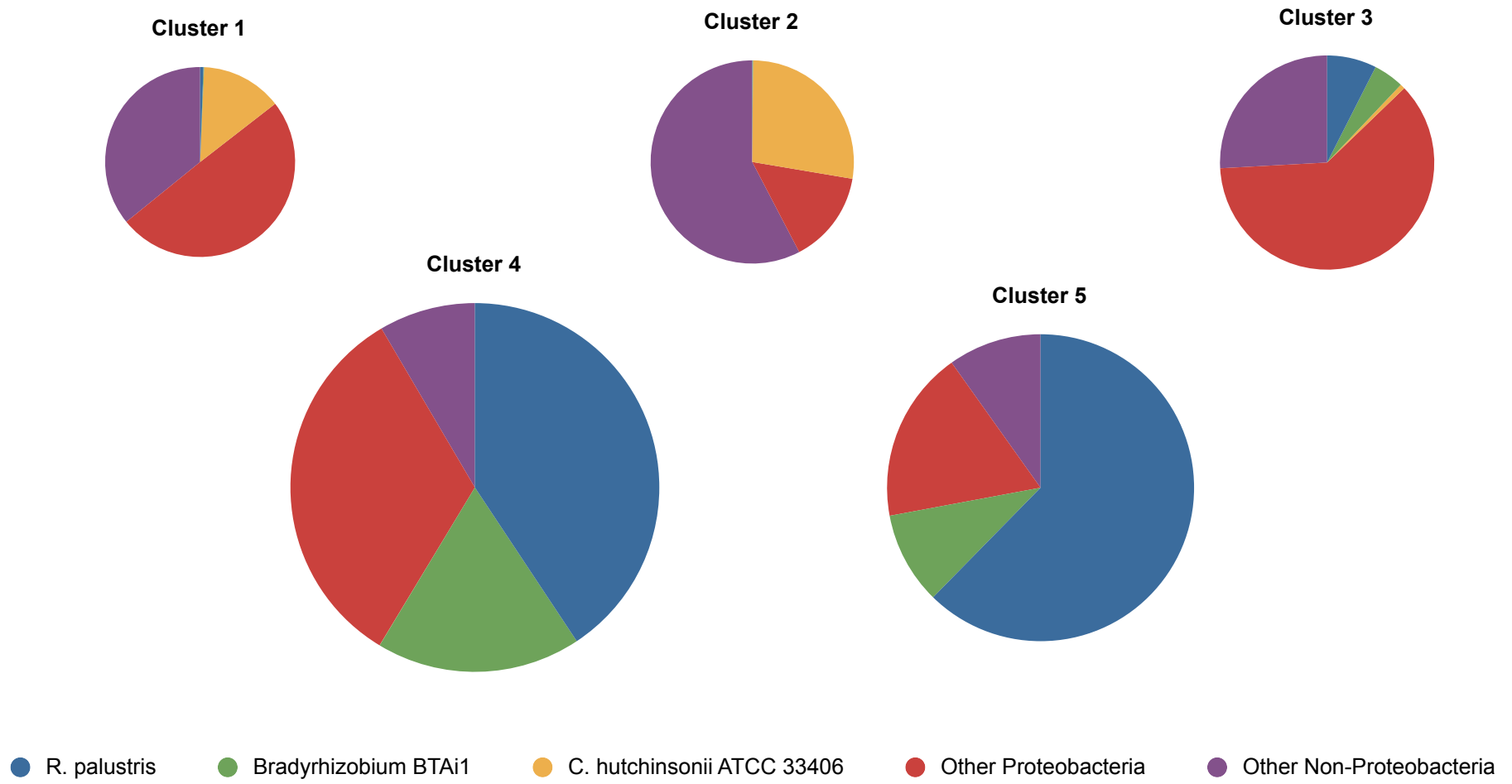
GC + IND + TNF

(xii)



GC + OFDEG + TNF

(xiii)

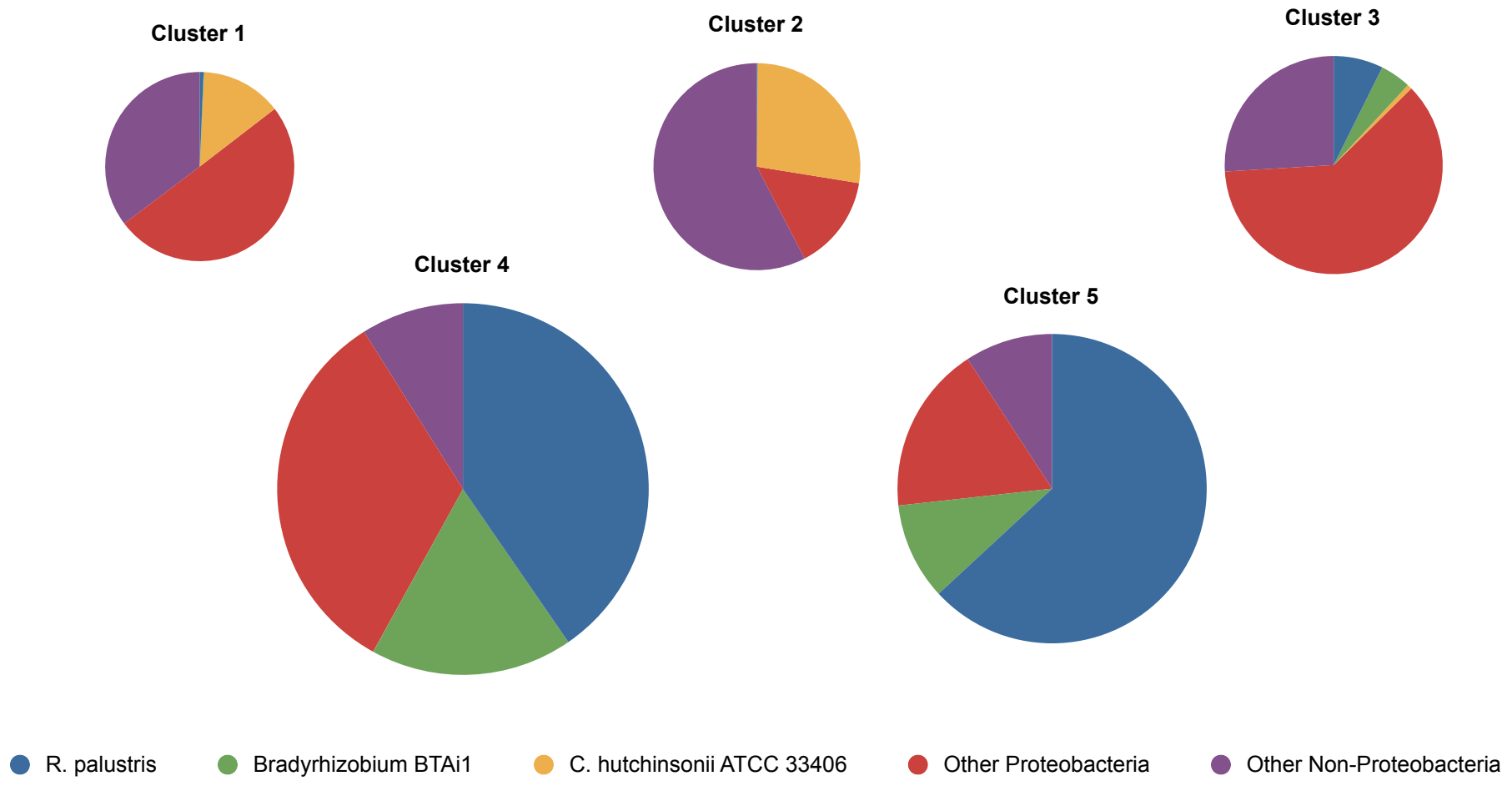


IND + OFDEG + TNF

(xiv)

GC + IND + OFDEG + TNF

(xv)



When compared to the distribution of sequences in *simLC* as a whole (Fig. 2.6), it is clear that some features and combinations - specifically IND, OFDEG, IND +OFDEG vectors - produce clusters whose distributions are not markedly dissimilar (see Fig. 2.7(ii), 2.7(iii) and 2.7(viii)). This similarity in distribution within clusters indicates that the separation of the data in these instances was close to random, with no notable enrichment in the clusters for any of the classes in the dataset. Unless stated otherwise, the clustering results produced with these feature vectors will not be included in the subsequent further discussion.

Some of the clusters produced with the remaining feature variants were enriched for one or a number of the classes in the dataset. Several patterns were observed.

One prominent trend in the results was the tendency for sequences from CPH to be clustered separately from sequences from the other two predominant species in *simLC*, RP and BR1. This effect could be identified in most of the sets of clusters produced, and was particularly apparent in clustering with GC, GC + IND + OFDEG, GC + OFDEG + TNF, and GC + IND + OFDEG + TNF feature vectors (see Fig. 2.7(i), 2.7(xi), 2.7(xiii), and 2.7(xv) respectively).

In the results from these feature sets, all or the vast majority of the sequences from CPH were grouped into one or two clusters, with the majority of the other sequences in the cluster(s) originating from species in the Proteobacteria and non-Proteobacteria classes, with very few or no sequences from the other two predominant species in the dataset also present.

The same trend could be seen in the results from the remaining feature combinations (not including those identified previously as having returned poor clustering of the data). However, the extent of ‘contamination’ with RP and BR1 in those clusters containing the bulk of the CPH sequences was increased in these cases, indicating poorer separation of the sequences between species.

Another trend observed throughout the sets of clusters depicted in Fig. 2.7 was the tendency for sequences from RP and BR1 to be clustered together. The sequences from these two species were observed to be consistently grouped together and separately from those derived from CPH. Most commonly, the vast majority of sequences from these two species were grouped into two of the

clusters produced (or three, in the case of IND + TNF vectors), with the majority of remaining sequences in these two clusters originating from other proteobacteria.

This pattern was particularly well-established in the clusters produced with the use of TNF, GC + TNF, IND + TNF, and IND + OFDEG + TNF vectors (see Fig. 2.7(iv), 2.7(vii), 2.7(ix) and 2.7(xiv) respectively).

Feature evaluation - *simLC*

While the clusters generated with the different feature types and sets were similar (excepting IND, OFDEG and IND+OFDEG, as discussed previously), some important differences in the quality of separation were identified.

Of the clusters produced with the use of a single feature type only, those obtained with TNF features (Fig. 2.7(iv)) were the most successful. This set consisted of three larger and two smaller clusters. Almost all CPH sequences were grouped into two clusters, which contained a negligible amount of sequences from RP and BR1.

CPH reads were grouped into TNF Cluster 1 with Rc 54.62%, at Pr 12.94%, and into TNF Cluster 3 with Rc 44.51%, at Pr 23.36%. RP and BR1 reads accounted for only 2.16% and 0.80% of TNF Cluster 1, and 0.12% and 0.10% of TNF Cluster 3, respectively. The remainder of these two clusters was accounted for by sequences from other proteobacterial (TNF Cluster 1: 47.79%; TNF Cluster 3: 15.55%) and non-proteobacterial species (TNF Cluster 1: 36.31%; TNF Cluster 3: 60.87%).

Almost all sequences from RP and BR1 were split between two of the remaining clusters. RP reads were grouped with Rc 49.78% and Pr 42.95% in TNF Cluster 4, and Rc 48.10% and Pr 68.31% in TNF Cluster 5, while BR1 reads were grouped into the same clusters at Rc 69.00% and Pr 17.96%, and Rc 27.89% and Pr 11.95%. (These Rc and Pr statistics are calculated for the species stated for ease of description, and are not necessarily the best-represented in the relevant clusters.) Sequences from other proteobacterial species constituted 29.89% and 15.37% of the clusters, with the remaining reads originating from other non-proteobacterial species. No CPH reads were grouped into either of these clusters.

The quality of this grouping with TNF features was marginally improved relative

to that achieved with GC feature vectors (Fig. 2.7(i)), where the clusters produced displayed a similar pattern, with a slightly less marked separation achieved between proteobacterial and non-proteobacterial species.

Using the clusters produced with TNF vectors as a baseline, little improvement was observed in the clustering obtained using a combination of two feature types. With the exception of IND + OFDEG, discussed previously, similar patterns were observed in the clusters produced with these combinations.

A slight improvement in the quality of clustering was obtained with OFDEG + TNF vectors (Fig. 2.7(x)), where 99.83% of CPH sequences were grouped into two clusters, and 98.56% and 98.80% of sequences from RP and BR1 respectively spread between the remaining three clusters. As such, the overlap between reads from these two predominant proteobacterial species and CPH was slightly reduced in the clustering, relative to the clusters produced with TNF features alone. However, this improvement was only very marginal, and could be accounted for in the most part by the consideration in this case of a third cluster containing sequences from RP and BR1. In the grouping produced with TNF features alone, these reads were concentrated almost entirely in only two clusters.

Another improvement was observed, with OFDEG + TNF vectors, in the enrichment of one particular cluster for RP sequences. In the results from both TNF and OFDEG + TNF feature vectors, one cluster was found to contain approximately half of the RP reads in the dataset (TNF Cluster 5 Rc: 48.10%; OFDEG + TNF Cluster 5 Rc: 50.08%). In results from OFDEG + TNF feature vectors, these reads constituted 73.73% of the sequences in the cluster (this figure is equivalent to the precision value used previously), compared to 68.31% with the use of TNF features alone.

GC + TNF feature vectors were also found to provide a slight improvement in clustering relative to the use of TNF features alone. As opposed to the results from TNF and OFDEG + TNF vectors described previously, a large proportion of RP reads were clustered together into a single cluster. GC + TNF Cluster 5 contained 77.09% of all the sequences belonging to this species (Rc), with these reads constituting 57.95% of the whole cluster (Pr).

This increased proportion of RP sequences grouped into a single cluster was a

product of the large size of this cluster relative to those produced with the other vector variants. As such, these RP sequences did not constitute a greater portion of the cluster itself than was observed from other feature sets such as GC + IND and GC + OFDEG.

Clusters produced from the use of combinations of three, and all four, feature types displayed separation of the dataset that was generally comparable to that achieved with TNF or GC feature vectors alone, with little improvement observed from any set of features beyond the marginal increase in clustering accuracy described for GC + TNF and OFDEG + TNF vectors.

The clusters produced using IND + OFDEG + TNF feature vectors (Fig. 2.7(xiv)) displayed good separation of proteobacterial and non-proteobacterial species into four principal groups, with one of the clusters produced containing markedly fewer sequences. These clusters showed particularly clearly the effect of grouping and separation along broad taxonomic lines that was observed throughout the results with most of the vector variants used.

Feature generation times

A comparison of the time taken to generate each of the four feature types was performed. GC, IND, OFDEG and TNF feature vectors were each produced from FASTA files of 1000, 5000 and 10,000 sequences, randomly generated from the four-letter DNA alphabet to a mean length of 300 bp. The time taken for feature generation to be completed from these files was recorded in each case. The results are summarised in Table 4.2.

Feature vector files were produced on a single 2.2 GHz AMD Opteron 6174 CPU with 512 KB memory. This exercise was carried out to compare the computational burden associated with each feature type, and highlights an important consideration.

From the results in the table, it can be seen that GC features took the least time to produce, with feature generation taking <1s for all data files.

IND and TNF feature vectors were found to take approximately the same amount of time to produce, at a rate of ~550 sequences/second and generating 10,000 feature vectors in <20s.

Table 2.8 *Time taken (in seconds) to produce GC, IND, OFDEG and TNF feature vectors from a dataset of 1000, 5000 and 10,000 randomly generated sequences with a mean length of 300 bp.*

	Time taken (seconds)		
Feature	**1000 sequences**	**5000 sequences**	**10000 sequences**
GC	0	0	0
IND	1	9	18
OFDEG	54	7245	18022
TNF	1	9	19

OFDEG features were found to take considerably longer to produce, with feature generation for 1000 sequences taking nearly one minute, while characterising 10,000 sequences took more than five hours.

It is likely that this huge difference in time required for feature generation per sequence was the result of the nested multiple sampling steps required in the production of these features. In the methodology used here to produce these features, the sampling and re-sampling depth parameters were set to 20 and 5 respectively. As each re-sampling loop included the set number of sampling loops at each sub-sequence size from which an error value is calculated, the number of individual calculations required for feature generation was very large. For example, if the process of generating the OFDEG value for a single sequence involved 50 stepwise increases in sampled sub-sequence length, this would include the calculation of 50 x 20 x 5, or 5000, tetra-nucleotide frequency distributions alone! The calculation of error values and the linear regression of these values was also repeated multiple times for each sequence during feature generation.

Discussion

Dataset 1

At this early stage of analysis, the results of clustering Dataset 1 indicated that TNF features were a particularly effective representation of sequences for this type of analysis. This feature was common to all the vectors that provided the best clustering of this dataset - TNF alone, and IND+TNF, OFDEG+TNF and IND+OFDEG+TNF combined vectors.

The use of some of these combinations provided a slight improvement on the clustering achieved with TNF vectors alone, with IND+TNF returning the single best set of clusters, measured by the Pr and Rc statistics described. This may indicate that the IND and TNF features are complementary, allowing for a more complete separation between sequences from different genomes: in regions where the profiles of one feature overlap between two genomes, the use of multiple feature types could allow for sequences from these regions to be distinguished if the profiles of the other feature type(s) remain distinct. Further comparison of IND, TNF and the other feature vector types will provide a more complete understanding of which vectors most reliably produce good clustering.

The equal split in the sequences between the three species in Dataset 1, as well as the considerable evolutionary distance between them, were deemed to be the causes of the similarity observed between Pr and Rc values obtained from clustering with this dataset. For each feature type and combination, at each mean sequence length of the data clustered, these values were found to be very closely matched. This effect was a product of the clusters in the data being produced of largely equal size: if the three clusters are equally sized, at 20,000 reads each, and each species is represented by 20,000 reads in the dataset, then the precision and recall statistics for each cluster will be identical.

The decrease in clustering quality observed with decreasing mean sequence length in Dataset 1 suggested that shorter sequencing reads are not well-suited to this kind of clustering analysis. It has been observed previously that short sequence lengths can be particularly limiting to clustering with genomic signature features (Abe, Kanaya et al. 2003; Huson, Auch et al. 2007; Martin, Diaz et al. 2008; Saeed and Halgamuge 2009), and these results added further weight to that conclusion. Reads from *Illumina*, SOLiD and Ion Torrent

platforms, all with typical lengths below 400 bp, would be difficult to cluster effectively using the methods applied here, especially considering the additional levels of complexity associated with true sequencing data compared to the more straightforward composition of Dataset 1.

The construction of Dataset 1 with a variety of mean sequence lengths allowed the influence of this variable on clustering to be studied, while the equal proportions of sequences from different species, and the small number of different organisms used allowed a simple analysis of the quality of clustering obtained for each feature set. The large phylogenetic distance between the three species used also increased the likelihood that a good separation between species would be achieved in clustering.

However, as was stated when the dataset was introduced, Dataset 1 was too simplistic to be truly informative in the investigation of feature vectors, beyond a basic identification of the trends in the comparison. The species contained in the dataset were too arbitrary and distinct, and too evenly represented in the data.

The dataset was also relatively small, when compared to the 100,000s of sequencing reads now obtained as a matter of course from a standard high-throughput sequencing experiment.

Finally, the process of incrementally cutting sequence fragments of a randomly determined length from the genomic DNA sequences of the three species used was an extremely simplistic means of modelling true sequencing data, which failed to take into account the intricacies of the ‘real-world’ datasets: sequencing error profiles, sequencing noise, average read lengths, length distributions and variable coverage of the original sequence.

The clustering analysis performed with Dataset 1 provided a platform for the comparison of the sequence features and their combinations, but the conclusions that could be drawn from the results were limited. In order for a more detailed understanding of the potential of each feature vector type to be reached, a closer representation of true sequencing data was needed for use in further investigations.

Feature vectors identified from analysis of Dataset 1, as possible candidates for further use, included TNF, IND + TNF, OFDEG + TNF and IND + OFDEG + TNF

vectors.

simLC

The levels of success achieved in clustering of Dataset 1 were not reached when the same features and methods were applied to the more complex dataset, *simLC.* The results detailed in Tables 2.4-2.6 and in Fig. 2.7(i)-2.7(xv) indicated that the isolation of reads from the genome of a single species was beyond the scope of the approach taken here, when the species contributing to the dataset are so closely related and disproportionately-represented.

Instead, a more successful approach to the clustering of a more complex dataset such as this one appeared to be to separate the data along broader taxonomic lines into a smaller number of clusters than the total number of individual contributing genomes. The hybrid classification system detailed in Fig. 2.6 made interpretation of the quality of this clustering much easier, allowing for a more in-depth analysis and comparison of the results obtained with each feature type and combination.

Although no species-specific clusters could be produced, sequencing reads from CPH and the two predominant proteobacterial species RP and BR1 were separated particularly well with TNF feature vectors (Fig. 2.7(iv)) when the dataset was grouped in this way. Co-clustering was observed between reads from these predominant species and those from other species of the same phylum. Another cluster was also produced that contained very few reads from any of the three best-represented species. These results indicated that the grouping and separation of *simLC* according to the genome from which reads had originated was relatively successful with these features.

As with Dataset 1, TNF features appeared to be the common factor in the vectors that produced the most accurate clustering results, with GC content also providing a good basis for separation of the data, at least along the broad taxonomic lines described here. IND and OFDEG features were shown to produce negligible specificity in grouping of the dataset unless combined with one of the aforementioned feature types. This indicated that GC and TNF features were responsible for the majority of the distinction made between reads from different genomes during clustering.

Excepting those sets of clusters derived from IND, OFDEG and IND+OFDEG

feature vectors, which were found to provide results no better than could be expected from clustering at random, a typical set of clusters produced from *simLC* could be described as consisting of: two clusters containing between them all or almost all of the sequences from RP and BR1, and a large fraction of the sequences from other proteobacterial species in the dataset; one or two clusters containing between them all or almost all of the sequences from CPH, with the remainder of sequences in these clusters belong to the Proteobacteria and non-Proteobacteria classes; and one or two clusters containing, almost exclusively, sequences from the two non-specific classes, Proteobacteria and non-Proteobacteria.

The clustering of sequences from RP and BR1 together with sequences from other proteobacterial species indicated that sequences from related organisms were being grouped together.

These results appeared to confirm that some separation of sequences according to species of origin was possible using most of the feature vector types investigated. For datasets such as *simLC* that are derived in unequal proportions from many closely related organisms, the accurate grouping of all sequences derived from one species into an individual cluster appears likely to be outside the scope of these methods. However, the results of clustering interpreted with a hybrid, summarised classification of the data showed that sequences could be grouped along much more broad taxonomic lines. Such a broad separation could allow some clusters to be enriched with the sequences from organisms represented much more highly in the dataset.

In some of the sets of clusters summarised in Fig. 2.7 all or almost all CPH reads were grouped together into a single cluster, that is, with Rc ~100%. However, these sequences constituted only a minority within the cluster as a whole (with Pr <<100%) A similar pattern was observed for RP and BR1, where the majority of reads from these species were clustered into one or two clusters, as described previously.

It may be that the lack of precise clustering - the production of clusters returning high Pr statistics - was a product of the closely related nature of the organisms contributing to the dataset. The tendency for the clustering together of sequences from proteobacterial species - including RP and BR1 - and,

separately, of non-proteobacterial species appeared to support this.

If a greater number of clusters were to be generated using the same methodology, some of the clusters formed might accurately and individually group together all of the sequences present from a single organism. Finding the optimal number of clusters to maximise this effect, however, is not trivial. This is especially true where little or no prior knowledge of the composition of the dataset under investigation is available.

When applied to a dataset that might typically be obtained from sequencing of an infected tissue sample, containing sequences from a relatively small number of different species that are not closely related, even a separation along the broad lines observed here could result in a fairly accurate clustering of the data. For example, if sequencing data generated from a plant tissue sample infected with a fungal pathogen was clustered, to a resolution comparable to that observed here, a separation of the sequences by phylum would still be sufficient to produce groups containing the host and pathogen sequences separately.

simLC was a more complex synthetic sequencing dataset than Dataset 1, discussed previously. Containing 97,255 sequences - after filtering - derived from 112 microbial species, *simLC* was ~50% larger than Dataset 1 and the product of sequences from many more organisms, which were more closely related than *A. thaliana*, *A. fumigatus* and *E. coli*. The degree of representation of each of the organisms in the data was also much more varied in *simLC*, with almost half of all the sequences originating from only three of the species. This variation in the representation of organisms within the dataset was a much more realistic simulation of the properties expected of a true sequencing dataset from a mixed sample, where it is unlikely that any two organisms would be found to contribute equal numbers of sequences.

The sequences contained within *simLC* were actual sequencing reads, taken in differing proportions, from experiments aimed at sequencing each of the species involved and combined to form the dataset. This ensured a more accurate simulation of true sequencing data, as each read may carry sequencing errors and any variation in length of reads was a product of natural variation in the sequencing process, rather than an arbitrary and simplistic random variation, introduced artificially as in Dataset 1.

However, because the sequencing reads that comprise *simLC* originated from Sanger sequencing experiments, the length distribution of the reads and the errors introduced into the reads during sequencing were the product of this sequencing method, and could not be assumed to accurately represent those of a high-throughput sequencing platform.

Two of the limitations of Dataset 1, described previously, were applicable also to *simLC*. Firstly, the mean length of the sequences, ~933 bp, was greater than that of most high-throughput sequencing read datasets currently produced. Secondly, the size of the dataset (97,255 sequences) was relatively small when compared to the 100,000s of individual reads typically obtained from a single sequencing experiment.

Additional concerns also existed over the complexity of the dataset and the short evolutionary distance between many of the species from which the dataset is derived. Given that the aim of this project was to determine whether any of the sequence features investigated were suitable for clustering of sequencing reads generated from infected tissue, which would commonly contain only a small number of relatively unrelated species in proportions great enough to be relevant post-sequencing, the composition of the *simLC* dataset from 112 microbial species may be too complex for the results of the clustering experiments described here to be truly informative.

Feature vectors identified here as possible candidates for further use were TNF, GC + TNF, OFDEG + TNF and IND + OFDEG + TNF vectors.

Limitations of the sequence features

Analysis of GC content is a very basic method of characterising a sequence, and effective separation of sequences from different species using this feature relies upon the overall GC content of the genomes being distinct, an assumption that cannot be made for all combinations of species. In relation to the organisms represented in *simLC*, GC features appeared to be sufficient to group the data along broad taxonomic lines. This may have been the result of so many species being present in the dataset across a taxonomic range.

However, GC content is likely to prove a less effective sequence feature when applied to a dataset derived from species with a similar genomic GC content profile, or multi-cellular eukaryotic species with regional variations in their genomic GC content profile. While the GC content of prokaryotic genomes remains consistent throughout the full sequence, the existence of isochores in the genomes of higher organisms, meaning that the GC content varies over whole regions many kbp in size, may reduce the effectivity of this feature in application to more complex species.

Characterising sequences from multiple isochores by GC content increases the likelihood of overlaps occurring between the GC profiles of sequences from the genomes of distinct species and subsequently of these sequences being grouped together incorrectly during clustering. Clustering by GC content may also result in sequences originating from the same genome being clustered separately due to the existence of these isochores. The brief analysis of this effect presented here suggested that the presence of isochores in the genome of *A. thaliana* did not have a significant effect of sequence grouping. It is possible that a stronger effect on clustering may be observed for sequences from other species.

Similar limitations also apply to the tetra-nucleotide frequency profiles of genomes: two genomes, from species that are not closely related, could conceivably have very similar TNF distributions. This would be likely to lead to difficulties in separating sequences from these species in a dataset. However, because the TNF profile of a sequence/genome consists of many individual frequency values, the likelihood of such circumstances arising, especially in two organisms represented in the same dataset, are much reduced compared to the

single-variable GC content feature.

The regional variation in GC content of eukaryotic genomes may also introduce some variation in oligonucleotide frequency across the whole genome. This effect has been compensated for in the past by measuring the relative abundance of oligonucleotides for each sequence, normalising the frequency of the oligonucleotide against those of its constituent nucleotides, a method that has been shown to provide a signature pattern consistent across the genome (Gentles and Karlin 2001; Simmons 2008). This representation of TNF features was not used in the investigations discussed here.

Coding regions of the genome have been shown to exhibit a slightly different inter-nucleotide distance (IND) profile from that of the genome as a whole, due to the existence of a reading frame of tri-nucleotide codons in these regions (Afreixo, Bastos et al. 2009). Beyond this, little investigation has been made into IND and OFDEG features in the context of the challenges associated with eukaryotic genomes, and the effect of the existence of isochores on the consistency of these features is not well understood.

Application of all four feature types to datasets containing sequences from eukaryotic organisms, beyond the simplistic case presented in Dataset 1, will provide a basis for determining the potential of each feature for effective clustering of this kind of data.

While the overall computational time required to generate OFDEG statistics for a dataset was shown to be considerably larger than the equivalent for the other three feature types, this limitation may be mitigated in large part through parallelisation of the feature generation process. The increased time requirement for OFDEG features is a consequence of the multiple nested sampling steps involved in their calculation. It would be relatively simple to run these individual sampling and re-sampling steps in parallel on multiple separate processing units, before collating the error values produced in each of these steps for computation of the gradient of their regression.

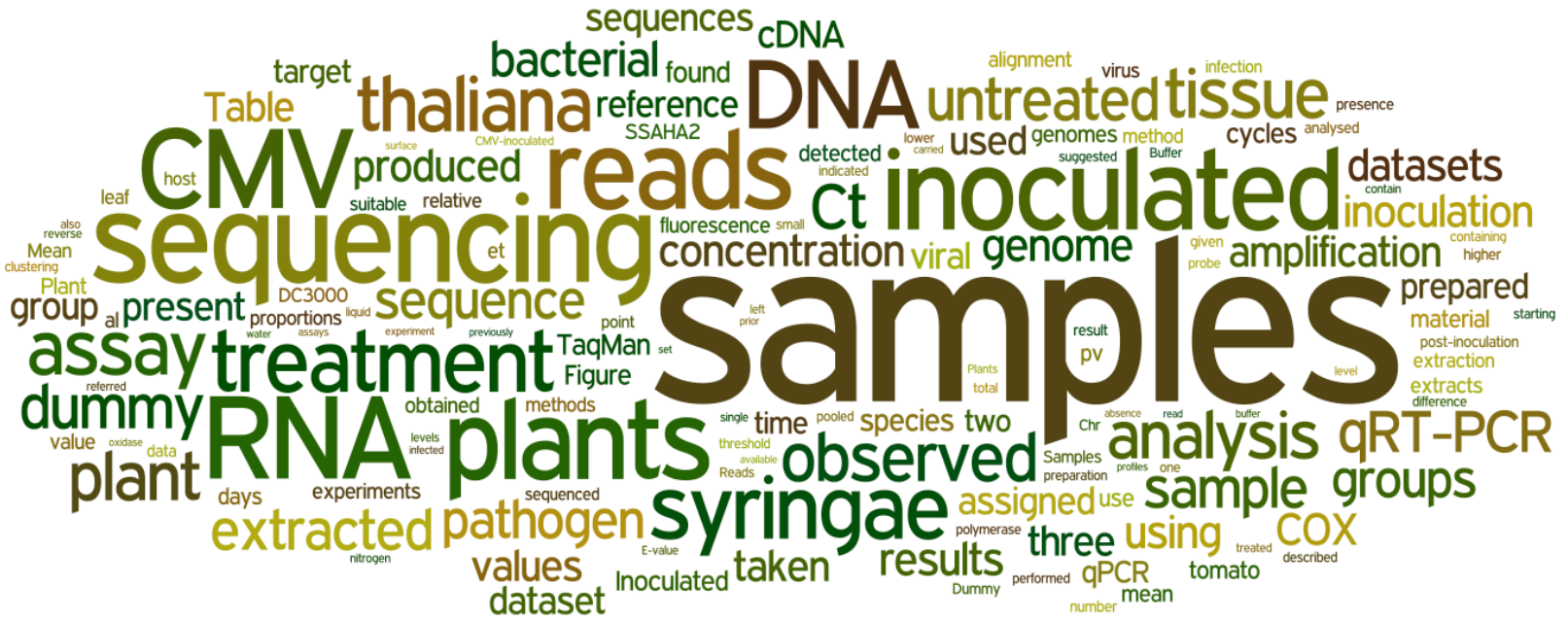
If this parallelisation of OFDEG generation were carried out, and feature generation performed on an array of many processor cores, the ‘wall clock’ time required to produce these features could be reduced by a huge degree to a level comparable with that of the other types discussed here.

Taking these factors into consideration and based on the results of feature comparisons with the *simLC* dataset, TNF features were determined to be the most suitable for separation of sequences.

Conclusion and future work

As discussed previously, both datasets used in this chapter were synthetic and were not expected to perfectly mimic the characteristics of a 'true' high-throughput sequencing dataset. In order for the potential of the features to be fully evaluated, another dataset should be found that more closely resembles such data, while retaining the advantages provided by a synthetic dataset: prior knowledge of the source of every sequence in dataset such that any clustering performed on the data can be assessed quantitatively.

An effort to develop such a dataset, through 454 GS FLX sequencing (*Roche/454 Life Sciences*) of tissue from a plant with a fully sequenced genome, infected with a pathogen with a genome that has also been sequenced fully, are described in the next chapter.


sequences
cDNA
bacterial
target
found
DNA
alignment
virus
infection
Table
thaliana
reference
untreated
tissue
presence
SSAHA2
CMV
produced
reads
used
genomes
method
cycles
analysed
detected
inoculated
datasets
host
leaf
suitable
relative
Ct
inoculation
sequencing
fluorescence
concentration
viral
genome
amplification
Mean
et
samples
prepared
Plant
group
present
DC3000
sequence
material
starting
assay
treatment
TaqMan
Figure
pv
post-inoculation
extraction
dummy
RNA
plants
obtained
methods
extracts
qRT-PCR
plant
observed
species
two
analysis
value
time
groups
days
experiments
assigned
use
sample
extracted
pathogen
syringae
COX
values
taken
results
three
using
dataset
Inoculated
qPCR
tomato
mean

3

Preparation and analysis of high-throughput sequencing data from a host-pathogen system with fully available reference genome sequences

Abstract

A pair of sequencing datasets were prepared from Arabidopsis thaliana *plants inoculated with the bacterium* Pseudomonas syringae *pv.* tomato *DC3000 or* Cucumber mosaic virus*. The aim of these experiments was to investigate whether this approach could produce a dataset suitable as a basis for comparison of sequence features and methods of clustering analysis. As the full genome sequence was available for the plant host and both pathogens, the reads generated in sequencing by 454 GS FLX could be mapped to a contributing genome, providing insight into the relative proportion of reads derived from each species in the sample and potentially allowing the quantitative evaluation of the results of any clustering analysis performed on the dataset. The datasets produced from samples prepared with both pathogens were found to contain a negligible number of reads originating from the pathogen genome, preventing their use in any further clustering analysis and suggesting that this sequencing approach may not be suitable for the preparation of data for such analysis.*

Introduction

In the previous chapter, an assessment of sequence feature vectors was described. The features were used to cluster two synthetic datasets that were developed to simulate the data produced by high-throughput DNA sequencing of multi-species samples. It was recognised that these datasets were not ideal resources for predicting or evaluating the performance of these feature vectors.

In one case, this unsuitability stemmed from the simplicity of the dataset, where the sequences used from each species were in equal proportion and derived from species that were very well separated in evolutionary distance. In the other, the dataset contained sequences that were too long and from too many different species, that were too closely related, for the scope of methods under investigation here. Finally, both datasets were composed of fewer than 10^5 individual reads, placing them at the lower size limit of a high-throughput sequencing dataset, which can typically consist of millions of reads. A more stringent evaluation of sequence feature vectors could be carried out with a larger dataset.

A more suitable dataset for the evaluation of these features might be obtained by the actual sequencing of a sample containing multiple species, where all species present have the full sequence of their genome available. The use of such a dataset would have the combined advantages of providing the user prior knowledge of the species from which each sequence contained in the dataset has originated, while also removing the uncertainty of how any methods tested on the dataset will perform when applied to other datasets that have been produced in true sequencing experiments from a natural combination of species.

The aim of the work described in this chapter was to prepare a pair of sequencing datasets from a fully sequenced plant host infected with fully sequenced pathogen species, to investigate the nature and proportions of the sequencing reads produced, and to determine whether these datasets would be suitable for use in the evaluation of sequence clustering methods. By inoculating individuals belonging to a host plant species with a fully sequenced genome, with a pathogen with a fully sequenced genome and preparing the infected tissue that is produced for sequencing, a dataset of sequencing reads

belonging to either the host species or the pathogen should be produced. As the genome of each species used in production of the dataset is fully available, a reference database can be built from these sequences and (sequencing error and noise notwithstanding) used to assign an origin to each individual read produced in sequencing by mapping each one to one of the reference sequences.

Once the reads have been assigned, the proportion of the dataset originating from each species can be determined, and its suitability for use in the evaluation of clustering methods can be assessed.

Arabidopsis thaliana ecotype Col-0 (*A. thaliana*, genome size ~119 Mbp, mean GC content ~36%) was chosen as a suitable host plant. *A. thaliana* is a very well-studied plant *(Initiative 2000)* that is fast-growing and has a small genome (~119 Mbp in five chromosomes *(Swarbreck, Wilks et al. 2008)*) in contrast to those of most of the other plant species that have had their genomes fully sequenced (e.g. genome size of the tomato plant, *Solanum lycopersicum* = ~781 Mbp in 12 chromosomes, according to the most recent NCBI Genome assembly (http://www.ncbi.nlm.nih.gov/genome/assembly/243988/)). These properties made it a sensible candidate for use in these experiments: the short maturation period reducing the time between planting seeds and the individuals being ready for inoculation, and the small genome reducing the size of the reference database to be produced and therefore the time required to assign sequencing output to it.

Pseudomonas syringae pv. *tomato* DC3000 (*P. syringae* DC3000, genome size ~6.4 Mbp + ~140 kbp in two plasmids, mean GC content ~58%) is a bacterial pathogen of *A. thaliana (Whalen, Innes et al. 1991)*, whose interactions with the host plant have been extensively studied *(Soylu, Brown et al. 2005; Thilmony, Underwood et al. 2006; Kim, Kim et al. 2008; Rico, McCraw et al. 2011)*. As a result of this, the interaction between *A. thaliana* and *P. syringae* pv. *tomato* DC3000 has been established as a model system, and the genome of the bacteria has been fully sequenced *(Buell, Joardar et al. 2003)*. This made it suitable for use as a bacterial pathogen in this series of experiments.

Cucumber mosaic virus (CMV, genome size 6.45 kbp, mean GC content ~41%) is a widespread plant pathogen, which is known to infect a wide range of plant

species including *A. thaliana (Sosnova and Polak 1975)*. It can be transmitted easily between plants mechanically or by seed infection, and has a genome that has been fully sequenced, composed of three single-stranded RNAs *(Rizzo and Palukaitis 1988; Rizzo and Palukaitis 1989; Owen, Shintaku et al. 1990)*. As a pathogen of *A. thaliana* with a fully sequenced genome, CMV was a suitable choice of pathogen for use in these experiments.

Two sequencing datasets were prepared, from *A. thaliana* tissue infected in one case with *P. syringae* pv. *tomato* DC3000 and in the other with CMV. In addition to these two datasets, two control datasets were also prepared for each pathogen treatment, one from untreated plants and another from “dummy inoculated” plants.

Dummy inoculated plants were treated identically to plants that were inoculated with viral or bacterial material, but in the absence of any pathogenic material. For example, where true inoculations were carried out using bacteria suspended in water or virus-infected plant material homogenised in phosphate buffer, dummy inoculated plants were treated using only water or buffer. Sampling from these dummy inoculated plants acts as a control to ensure that any differences observed between inoculated and untreated samples were the result of the presence of the pathogen, rather than a product of the inoculation process itself.

Plants were harvested at an appropriate time point after inoculation, and DNA or RNA extracted from bacterial- and viral-treated plants respectively. DNA or cDNA prepared from the extracted RNA was sequenced by 454 GS FLX (*Roche*/*454 Life Sciences*). The sequencing reads produced were then mapped to the genomes of *A. thaliana* and the appropriate pathogen. The proportions of the datasets that originated from each species are reported and discussed, in the context of the feature comparison and clustering analysis to be performed in the remainder of the project.

Materials and Methods

Inoculation of plants

Arabidopsis thaliana ecotype Col-0 plants were grown for at least 18 days after sowing before any treatment was carried out.

For each pathogen, three treatment groups were established: untreated, dummy inoculated and pathogen inoculated plants (for brevity, sometimes referred to as UT, DI, and *P. syringae* or CMV according to pathogen). Plants belonging to each treatment group were grown and treated in separate growth chambers, under identical conditions, to prevent cross-contamination between groups.

- **Viral inoculation**

Fresh leaf material infected with *Cucumber mosaic virus* (CMV) was homogenised in phosphate buffer (0.04M NO_2HPO_4, 0.027M KH_2PO_4 in distilled water) to release sap, with Celite (powdered diatomite) added to increase abrasion of the leaf surface and improve the likelihood of viral inoculation.

This mixture was rubbed onto the surface of a single lowest-level leaf on each target plant, and left for ~2 mins, before washing with distilled water.

Plants referred to as ‘dummy inoculated’ in relation to viral inoculation were treated identically, but rubbed with a mixture of phosphate buffer and Celite only.

Plants referred to as ‘untreated’ were left untreated.

- **Bacterial inoculation**

Target *A. thaliana* plants were sealed individually in clear polythene bags for ~18 hours prior to inoculation, to facilitate the uptake of bacteria into leaves and, after inoculation, resealed in the bags for ~72 hours.

Plants referred to as ‘untreated’ were not sealed in bags and were left untreated.

- **Spraying protocol**

Cultured *Pseudomonas syringae* pv. *tomato* DC3000 (referred to as simply *P. syringae*, unless otherwise specified) was suspended in distilled water until turbid. Each target plant was individually removed from its polythene bag, and

bacterial suspension sprayed onto the plant using a small, sterilised, plastic atomiser.

Plants referred to as ‘dummy inoculated’ in relation to bacterial spraying inoculation were treated identically, but sprayed only with distilled water using a different, sterilised atomiser before being resealed in a polythene bag.

Plants referred to as ‘untreated’ were not sealed in bags and were left untreated.

- **Rubbing protocol**

Each plant was removed from its bag and cultured *P. syringae* from a plate of growth medium was rubbed gently onto the underside of a single, lowest-level leaf using a sterile gloved hand.

Plants referred to as ‘dummy inoculated’ (in relation to bacterial rubbing inoculation) were treated identically, but rubbed in the absence of any cultured bacteria before being resealed in a polythene bag.

Tissue sampling

Tissue samples (100 mg) were taken from leaves of *A. thaliana*. Where a specific leaf was used as the point of inoculation (i.e. in viral inoculation and bacterial rubbing inoculation), samples were taken from systemic leaves, separate and distinct from the inoculated leaf and from a higher (younger) point on the plant stem. This helped to ensure that any pathogen material detected during the analysis of these samples was the result of systemic infection of the plant, rather than residual material remaining on the surface of the treated leaf.

Tissue samples taken from viral treatment groups were stored at -80℃ until needed for RNA extraction. Tissue samples taken from bacterial treatment groups were stored at -20℃ until needed for DNA extraction.

Extraction of RNA from viral treatment groups

RNA was extracted viral treatment group samples, for analysis by quantitative reverse transcription-coupled polymerase chain reaction (qRT-PCR).

Extraction of RNA was achieved using RNEasy Kit (*Qiagen*), following manufacturer’s instructions for plant tissue samples, including all optional steps and using the buffers provided. Briefly, 100 mg plant tissue was homogenised in

Buffer RLT+β-mercaptoethanol (450 µl) by shaking vigorously with glass beads and applied to a QIAshredder column, then centrifuged to remove tissue debris. Ethanol (0.5 x sample volume) was added to the sample, which was then applied to a RNEasy spin column and centrifuged to bind RNA to the column. A DNase digestion (Appendix D of RNEasy Kit handbook) was performed on the immobilised sample, before washing with a series of buffers (700 µl RW1, 2 x 500 µl RPE). Extracted RNA was eluted in 2 x 30 µl RNase-free water and stored at -80 ℃ until needed.

Two slightly different methods were used in RNA extraction. In method LqN, tissue samples were frozen in liquid nitrogen before and during homogenisation, until Buffer RLT was added. In method NLqN the use of liquid nitrogen was omitted.

Extraction of DNA from bacterial treatment groups

DNA was extracted from bacterial treatment group samples, for analysis by quantitative polymerase chain reaction (qPCR), to assess the relative concentration of *P. syringae* DNA present in the tissue samples taken.

Extraction of DNA was achieved using DNEasy Kit (*Qiagen*), following manufacturer's instructions for small plant tissue samples, including all optional steps (excepting those related to tissue disruption) and using the buffers provided. Tissue (100 mg) was homogenised in 400 µl Buffer AP1 (provided with DNEasy Kit) by shaking vigorously with glass beads, RNase added and the sample incubated for 10 minutes at room temperature to allow RNA digestion. Buffer AP2 (130 µl) was added and the sample applied to a QIAshredder column and centrifuged to remove tissue debris. A third buffer, 1.5 x sample volume of AP3+ethanol, was added and the sample applied to a DNEasy spin column and centrifuged to immobilise DNA. The sample was washed with Buffer AW (2 x 500 µl) and DNA eluted in 2 x 50 µl Buffer AE and stored at -20 ℃ until needed.

TaqMan assays

The presence and relative concentration of pathogenic material in a sample of extracted DNA or RNA was determined by quantitative polymerase chain reaction (qPCR) or quantitative reverse transcription-coupled polymerase chain reaction (qRT-PCR) respectively. These analyses were performed using a

TaqMan assay for each pathogen, and another assay to detect the presence of cytochrome oxidase.

Cytochrome oxidase subunit 1 (COX) is one part of a ‘housekeeping’ enzyme, vital for the metabolism of cells and as such the COX gene (contained in the mitochondrial genome) is highly conserved between species and consistently active in the great majority of cells. Consequently, an assay for this gene in plants is applicable as an assay to detect plant material across many species (Boonham, Laurenson et al. 2009). In this experiment, fluorescent activity observed from the COX-specific assay demonstrates the presence of plant genetic material in the extracted RNA or DNA.

First described by (Holland, Abramson et al. 1991), a TaqMan assay is a set of biochemical reagents that combine to form a test specific to a target DNA sequence, where the level of fluorescence emitted from a sample under analysis is proportional to the number of copies of the target sequence present in the sample.

The assay set consists of four reagents: two primers, which straddle or ‘bookend’ the target sequence; a probe, consisting of the target sequence, a fluorophore and a ‘quencher’ group that absorbs the fluorescence emitted from the fluorophore when in close proximity; and a DNA polymerase with 5’ exonuclease activity (Taq polymerase).

Assay sequences:

- ***Cucumber mosaic virus***

Forward primer: GCTTGTTTCGCGCATTCAA

Reverse primer: GAGGCAGRAACTTTACGRACYGT

Probe: FAM-TTAATCCTTTGCCGAAATTTGATTCTACCCGT-BHQ1

- ***Pseudomonas syringae* pv. *tomato* DC3000**

P. syringae forward primer: GTGAAACTGCATTCTTCCATGTG

P. syringae reverse primer: TTGCGTCCTGGCGTTGT

P. syringae probe: FAM-CCGGTGGCAGATCCTCTCCATACCA-BHQ1

- **Cytochrome oxidase subunit 1**

COX forward primer: CGTCGCATTCCAGATTATCCA

COX reverse primer: CAACTACGGATATATAAGRRCCRRAACTG

COX probe: VIC-AGGGCATTCCATCCAGCGTAAGCA-TAMRA

COX assay sequences originally published in (Boonham, Laurenson et al. 2009).

qPCR and qRT-PCR

To assess the relative concentrations of cytochrome oxidase and CMV RNA present, samples of RNA extracted from viral treatment group plants were analysed by qRT-PCR. Similarly, DNA extracted from bacterial treatment group samples were analysed by qPCR, to determine the relative concentrations of COX and *P. syringae* DNA present.

As in standard PCR, commonly used to quickly produce many copies of a single target sequence, in qPCR and qRT-PCR a pair of primers are used to allow amplification of the target sequence, doubling the copy number for every thermal cycle of the reaction until insufficient primers remain or the cycling is halted and amplification ceases. As new copies of the target sequence are produced in this amplification, a probe binds to its reverse complement on one strand of the target sequence. As the reverse complement of this strand of the target sequence is produced by the polymerase, the probe is removed nucleotide-by-nucleotide - a process of ‘overwriting’ achieved by the 5’

exonuclease activity of the Taq polymerase. As the probe is degraded by the Taq polymerase, the quenching group is removed from the proximity of the fluorophore, and the subsequent increase in emitted fluorescence can be observed by a fluorimeter.

As each probe sequence has a single fluorophore attached, the observed increase in fluorescence remains proportional to the increase in copy number of the target sequence as long as sufficient probe remains available for binding to newly produced target sequences (assuming the primers are truly target-specific i.e. no non-specific amplification is taking place alongside the desired reaction).

In qRT-PCR a reverse transcription step is included before thermal cycling, to produce double-stranded cDNA copies of the RNA present in the sample. This allows the subsequent qPCR analysis to progress correctly.

qPCR and qRT-PCR results are represented as C_t values. The C_t value of an experiment is the number of amplification cycles required for fluorescence emitted from the TaqMan assay to reach a given threshold value. This threshold value will be reached in fewer cycles with increasing initial concentration of target sequence, which is assumed to correspond to initial concentration of pathogenic DNA/cDNA here. A decrease in C_t value of ~3.3 cycles corresponds to an approximately tenfold increase in starting concentration of target sequence. The threshold value from which Ct values were calculated was set at 0.2, the default value for the software provided by Applied Biosystems for use with the equipment.

Analysis of extracted RNA by qRT-PCR

Moloney Murine Leukemia Virus (M-MuLV) Reverse Transcriptase was used to prepare cDNA from extracted RNA, using the TaqMan assay primers also used in target sequence detection during qRT-PCR analysis. Samples were prepared for analysis in 96-well plates, each well containing 25 µl prepared as detailed in Table 3.1.

Each sample was analysed in duplicate for each of the two assays, for the detection of COX and CMV. Positive and negative control samples were prepared to ensure that a presence or absence of assay activity from a sample can be attributed only to a presence or absence of the target sequence in the extracted RNA sample. Positive controls were prepared with RNA extracted from a sample known to be infected with CMV, while negative controls were prepared as in Table 3.1 but without the addition of an RNA extract.

Prepared samples were cycled using an ABI 7900HT Fast Real-Time PCR System with the thermal cycling conditions detailed in Table 3.2.

Table 3.1 *Reagents and volumes used in preparation of samples for qRT-PCR analysis.*

Reagent (stock concentration)	**Volume (final concentration)**
Buffer A (10X Stock Solution, *Life Technologies*)	2.5 µl (1x)
MgCl2 (25 mM)	5.5 µl (5.5 mM)
dNTPs (2.5 mM each)	2 µl (200 µM each)
Forward Primer (7.5 µM)	1 µl (300 nM)
Reverse Primer (7.5 µM)	1 µl (300 nM)
TaqMan Probe (5 µM)	0.5 µl (200 nM)
AmpliTaq Gold DNA Polymerase (5 U/µl, *Life Technologies*)	0.125 µl (0.625 U)
M-MuLV Reverse Transcriptase (20 U/µl, *Fermentas*)	0.05 µl (1 U)
Extracted RNA sample	1 µl
Water	11.375 µl

Table 3.2 *Thermal cycling conditions for qRT-PCR analysis of extracted RNA samples.*

Phase	**Number of cycles**	**Temperature**	**Time (mins:secs)**
1	1	48 ℃	30:00
2	1	95 ℃	10:00
3	40	95 ℃	0:15
		60 ℃	1:00

Analysis of extracted DNA by qPCR

Extracted DNA samples were prepared for qPCR analysis in 96-well plates, each well containing 25 µl prepared as detailed in Table 3.3. Samples were prepared in duplicate and with controls as described for RNA samples previously. Positive controls were prepared with DNA from a sample known to contain both assay targets, while negative controls were prepared as below, but without the addition of a DNA extract.

Prepared samples were cycled using an ABI 7900HT Fast Real-Time PCR System with the thermal cycling conditions detailed in Table 3.4.

Table 3.3 *Reagents and volumes used in preparation of samples for qPCR analysis.*

Reagent (stock concentration)	Volume (final concentration)
Buffer A (10X Stock Solution)	2.5 µl (1x)
MgCl2 (25 mM)	5.5 µl (5.5 mM)
dNTPs (2.5 mM each)	2 µl (200 µM each)
Forward Primer (7.5 µM)	1 µl (300 nM)
Reverse Primer (7.5 µM)	1 µl (300 nM)
TaqMan Probe (5 µM)	0.5 µl (200 nM)
Taq Polymerase (Gold, 5 U/µl)	0.125 µl (0.625 U)
Extracted DNA sample	1 µl
Water	11.375 µl

Table 3.4 *Thermal cycling conditions for qPCR analysis of extracted DNA samples.*

Phase	Number of cycles	Temperature	Time (mins:secs)
1	1	50 ℃	2:00
2	1	95 ℃	10:00
3	40	95 ℃	0:15
		60 ℃	1:00

Preparation of cDNA for sequencing from extracted RNA

A 10 µl aliquot of each of 20 RNA samples from the three viral treatment groups was pooled and prepared for sequencing by reverse transcription to cDNA. cDNA was synthesised from each of these 200 µl pooled samples using the First-Strand cDNA Synthesis Kit (*Fermentas*) according to the manufacturers instructions for first- and second-strand cDNA synthesis.

Quantification of total DNA

DNA content of samples was quantified using a Qubit fluorimeter (*Invitrogen*), in preparation for sequencing of extracted DNA/RNA. DNA/cDNA samples (1 µl) were quantified using the 'high specificity dsDNA' kit, containing a set of standards for calibration of the fluorimeter and reagents for preparation of samples for detection, for use with the fluorimeter as per the manufacturer's instructions.

Preparation of extracted DNA and RNA for sequencing

It was estimated that a total of ~500 ng of DNA or cDNA for each sample would be sufficient for sequencing by 454 GS FLX (*Roche*/*454 Life Sciences*). A 15 µl aliquot was taken from each of 15 DNA extracts from untreated plants and pooled, while 10µl aliquots were taken and pooled from each of 15 DNA extracts from plants in the dummy inoculated and *P. syringae* inoculated treatment groups to give ≥500 ng total DNA for sequencing for each group. These volumes were chosen based on the concentration in extracts taken from each treatment group (results not shown here). The quantity of DNA present in these pooled samples was also determined, to ensure sufficient DNA was present for sequencing. Results of this quantification analysis, indicating total DNA present in the pooled samples, are given in Table 3.5.

The concentration of cDNA synthesised from pooled RNA extracts was also determined by fluorimeter analysis. cDNA synthesis was repeated for all three treatment groups, and a third time for CMV-inoculated samples, due to insufficient cDNA yield from the first round of cDNA synthesis.

cDNA was purified in preparation for 454 GS FLX DNA sequencing, according to the manufacturer's instructions for sample preparation. cDNA yield after purification of the combined products of these multiple rounds of synthesis are given in Table 3.5.

Table 3.5 *Amount of DNA sequenced from each treatment group, according to results of quantification analysis with Qubit fluorimeter. *ds-cDNA purified from the combined product of two rounds of double-stranded cDNA synthesis from pooled RNA samples. **ds-cDNA purified from the combined product of three rounds of double-stranded cDNA synthesis pooled RNA samples*

Treatment Group	DNA/cDNA sequenced (ng)
Untreated Plant DNA	672
Dummy Inoculated Plant DNA	588
***P. syringae* Inoculated Plant DNA**	924
Untreated Plant cDNA	327*
Dummy Inoculated Plant cDNA	376*
CMV Inoculated Plant cDNA	359**

Pooled DNA samples from bacterial inoculation groups were determined to contain over 500ng DNA for sequencing. After purification, total cDNA yields available for sequencing were below 500ng. While acknowledging concerns over the relatively small amount of cDNA available for sequencing, no further cDNA was prepared in view of the purity of the cDNA and constraints on available resources.

The six samples were prepared for sequencing according to the manufacturer's instructions and sequenced in two separate sequencing runs, corresponding to the two pathogens used, and each sample was sequenced on one region of a single 454 GS FLX plate.

Assignment of sequencing reads to reference genomes

After sequencing was completed, reads generated from each sample were aligned against two reference genomes using *SSAHA2* (Ning, Cox et al. 2001). Sequences used were the complete genomes of *Arabidopsis thaliana* Col-0 (NCBI Genome accession numbers NC_003070, NC_000932 and NC_001284) and of the appropriate pathogen for each database - either *Pseudomonas syringae* pv. *tomato* DC3000 (NC_004578, NC_004632, NC_004633) or *Cucumber mosaic virus* (NC_002035). These reference genomes included the *A. thaliana* organellar genomes and two plasmids found in *P. syringae* pv. *tomato* DC3000, as well as the chromosomal genomes, and all three CMV genomic RNA sequences.

SSAHA2 is a sequence alignment program, designed to allow fast mapping of short DNA sequences to a user-defined reference database. The software compiles a hash table of oligonucleotide words (13-mers in the default settings for *454* sequencing reads) from the sequences in the reference database and holds it in the system memory. This ready availability of the whole hash table allows *SSAHA2* to rapidly find exact matches, to seed alignments between query sequences and a reference sequence and return these as 'hits' between the two sequences (Ning, Cox et al. 2001). A newer software package, *SMALT*, has recently been released that the developers promise improves on the performance and ease of use of *SSAHA2* (available from http://www.sanger.ac.uk/resources/software/smalt/), although at the time of writing, no scientific report has yet been published detailing this software.

A sequencing read was assigned to a reference sequence if an alignment was found between the two sequences that did not exceed an expectation value (E-value) cutoff of 1×10^{-4}. This expectation value is a measure of the likelihood that this alignment could occur by chance between two unrelated sequences. *SSAHA2* ranks alignment hits between a query and a reference sequence in order of alignment quality, based on the Smith-Waterman score (Waterman and Smith 1981), with the hit returning the highest score ranked first. Where multiple hits were found for a single read here, this ranking based on S-W score was used to assign the read to a reference sequence.

Reads for which no alignment was found were placed in two groups: reads for which no hits were found for either reference genome and reads for which no alignment with a sufficiently low E-value were found.

SSAHA2 database generation and analysis was performed according to the developers instructions, using the built-in settings for 454 GS FLX sequencing reads (using the *-454* tag on execution), which specifies the following options: *-kmer* 13; *-skip* 3; *-seeds* 2; *-score* 30; *-cmatch* 10; *-ckmer* 6 (detailed in the user manual for *SSAHA2*).

Processing of *SSAHA2* results (in SAM format) and E-value-limited assignment of sequencing reads to a reference genome were carried out using the *perl* script 'SAMseqAssigner.pl', reproduced in Appendix A.

Results

Comparison of bacterial inoculation techniques

Two commonly used bacterial inoculation techniques were compared, to determine which most efficiently produced infection in *Arabidopsis thaliana* plants for dataset generation. Eight plants were inoculated with *P. syringae* pv. *tomato* DC3000, using two alternative methods of rubbing and spraying the bacteria onto four plants each.

Samples were taken from each plant at 17 and 21 days post-inoculation, and analysed by qPCR, to determine concentration of *P. syringae* DNA present. Results are given in Table 3.6.

Table 3.6 *Mean threshold fluorescence cycle (C_t) values for* P. syringae *pv* tomato *DC3000 and cytochrome oxidase (COX) assay of inoculated* A. thaliana *tissue samples. Lower C_t values indicate a higher initial concentration of assay target DN9A sequence in the sample prior to amplification in qPCR. Samples marked as 'not detected' possessed a starting concentration that was insufficient to be detected in the forty thermal cycles over which amplification occurred.*

Inoculation Method	**Rub**		**Spray**	
Assay	*P. syringae*	COX	*P. syringae*	COX
Dummy Inoculated Samples Day 17	not detected	33.5759	not detected	33.7736
***P. syringae* Inoculated Samples Day 17**	34.8489	33.5480	30.6720	33.5957
Dummy Inoculated Samples Day 21	not detected	32.9825	not detected	32.5474
***P. syringae* Inoculated Samples Day 21**	32.4631	32.2923	30.1573	32.7670

In the sets of samples taken at both time points, the C_t values measured with the *P. syringae* assay from samples prepared by spraying inoculation were found to be lower than those measured from samples inoculated by rubbing protocol.

As a lower mean C_t value indicates a greater starting concentration of target sequence, it was concluded that the concentration of bacterial DNA obtained from samples inoculated by spraying protocol was consistently higher than

those from samples inoculated by rubbing. At both sampling time points, the C_t values measured with the COX assay were similar between samples prepared with both inoculation methods, suggesting that the differences in C_t values observed with the pathogen assay were the result of differences in starting concentration of *P. syringae* DNA, as opposed to differences in overall starting concentration of all DNA.

From the higher yields observed from samples prepared by spraying inoculation, it was concluded that this was the more efficient method for inoculation of the *A. thaliana* plants with the cultured bacteria. This method was used to inoculate plants used in all subsequent experiments.

Determination of optimal tissue sampling time

The composition of a sequencing dataset is determined in part by the relative proportions of host and pathogen DNA/cDNA present in the sequenced sample. As the pathogen genome is considerably smaller than that of the plant host, in order to try to ensure that the pathogen is represented by as many reads as possible in a sequencing dataset produced from the samples, the level of pathogen present in the samples should be maximised. As such, tissue samples should be taken at an appropriate time point post-inoculation when levels of pathogen material in the tissue are at their highest.

To estimate the optimal sampling time post-inoculation to maximise the level of pathogen material present in tissue samples, three tissue samples were taken from inoculated plants at three-day time intervals (where day 0 was the point of inoculation). DNA, for *P. syringae*-inoculated plant samples, and RNA extracts, for CMV-inoculated plant samples, from these samples were analysed by qPCR/qRT-PCR. Results are given in Table 3.7 and Figure 3.1.

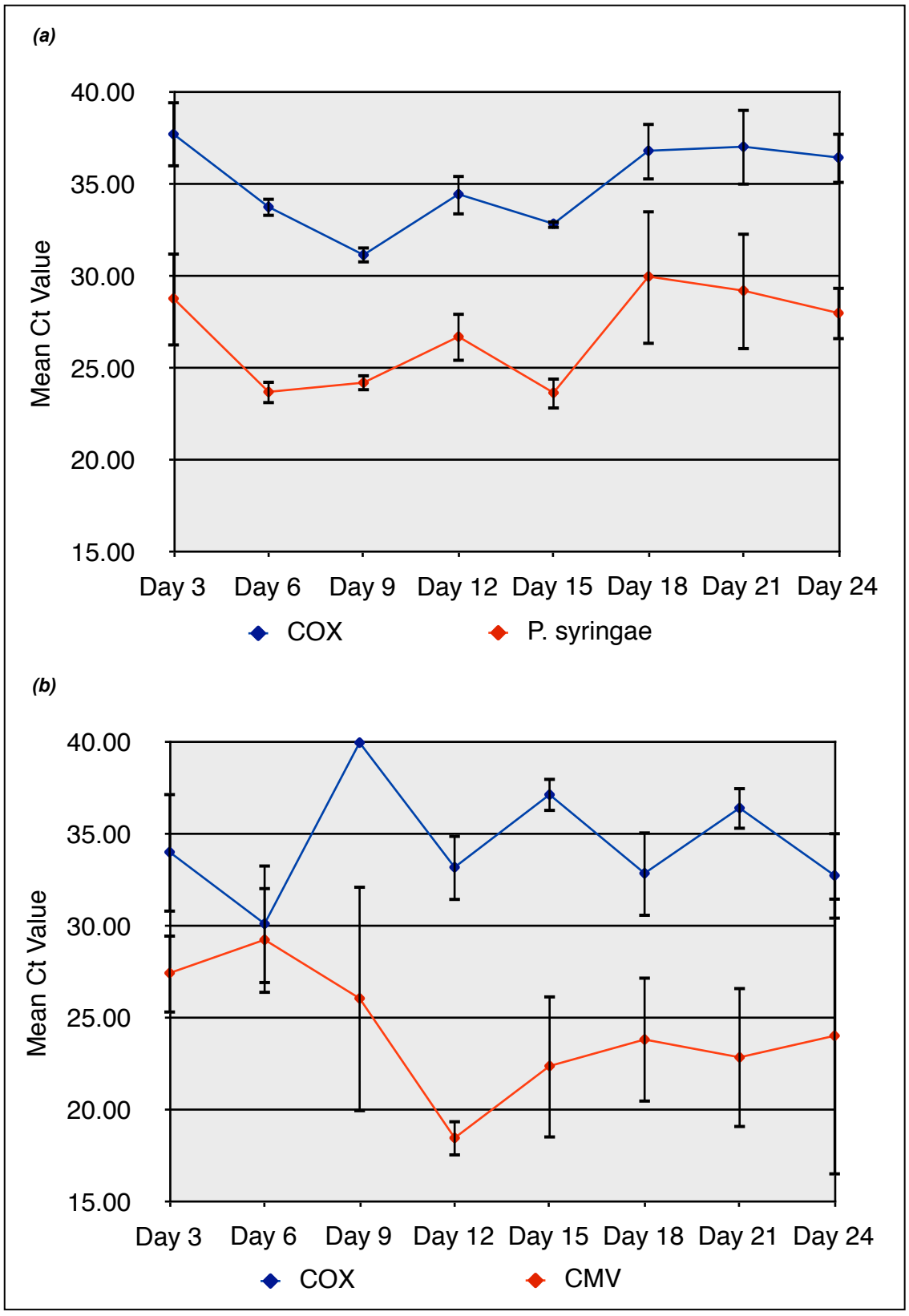


Figure 3.1 *Mean C_t values resulting from qPCR (DNA) or qRT-PCR (RNA) analysis with cytochrome oxidase- and pathogen-specific TaqMan assays, measured from plant tissue samples taken every three days over 24 days. Values for DNA samples (with* P. syringae *assay) are shown in (a), with values for RNA samples (with CMV assay) in (b). Standard error bars are given for each mean C_t value.*

Table 3.7 *Mean threshold fluorescence cycle (C_t) values for* P syringae *pv* tomato *DC3000 and CMV TaqMan assay of inoculated A thaliana tissue samples taken over 24 days. Lower C_t values indicate a higher initial concentration of pathogenic DNA/cDNA in the sample prior to amplification in qPCR.*

Sampling Time (days post-inoculation)	Mean Ct Value - *P. syringae*	Standard Error - *P. syringae*	Mean Ct Value - CMV	Standard Error - CMV
3	28.81	2.61	27.47	2.21
6	23.73	0.67	29.28	2.95
9	24.23	0.47	26.09	5.81
12	26.73	1.37	18.50	1.01
15	23.68	0.92	22.41	3.95
18	30.00	3.72	23.85	3.44
21	29.23	3.24	22.88	3.85
24	28.01	1.48	24.05	3.78

A correlation was observed between the mean C_t values derived from the *P. syringae* and the COX assay. This indicated that the variation observed in *P. syringae* assay C_t values could mostly be attributed to variation in the absolute DNA yield from the tissue samples taken at each time point, rather than variation in the starting concentration of *P. syringae* DNA present. This correlation suggested that the concentration of *P. syringae* DNA remained fairly consistent throughout the course of the experiment.

The same correlation was not observed between mean C_t values of COX and CMV in the qRT-PCR results. The amplification profiles of the COX assay were poor (compared to the exponential amplification expected in successful qPCR/ qRT-PCR analysis) and inconsistent (see Figures 3.1 and 3.2), making it very difficult to observe any pattern in the results. It was suggested that this poor amplification may have been the result of degradation of RNA prior to analysis. This degradation was most likely to have taken place during extraction, as tissue and extracted RNA were kept frozen at all other times before analysis. If the breakdown of RNA could be prevented prior to analysis, it was predicted that a repeat of this time course would be more informative.

A more identifiable pattern was observed in the mean C_t values derived from

the CMV assay. If the poor amplification observed with the COX assay was in fact due to degradation during extraction, the relative absence of this effect with the CMV assay may have been due to protection of the viral RNA by the protein capsid during the early stages of extraction.

It was estimated that the optimal time point post-inoculation for harvesting tissue samples was 15 days and 12 days for plants inoculated with *P. syringae* and CMV respectively. The choice of time point for sampling of *P. syringae* inoculated was made as day 15 post-inoculation, based on the lowest mean C_t value observed throughout the time course. The lack of stable COX assay results, to compare the CMV values with, led to the time point for harvesting of viral treatment groups to be chosen as day 12, the point at which the lowest absolute mean C_t value was observed with the CMV assay.

Study of RNA degradation in samples extracted using liquid nitrogen

The RNA samples used in the analyses described so far were extracted using the method NLqN. For examples of the poor amplification observed for COX RNA in these samples, see Table 3.7 and Figures 3.1 and 3.2.

In order to test the hypothesis that the poor COX amplification was the result of RNA degradation during extraction, and subsequently adopt the most effective method of extraction, a second protocol was devised (method LqN), using liquid nitrogen to keep the plant tissue frozen during the early stages of the extraction process when the tissue was homogenised, and extracts prepared by these two methods were compared. Freezing the tissue minimises degradation of the RNA by RNases prior to stabilisation with the addition of Buffer RLT, which was predicted to be the principal factor influencing the poor quality of RNA observed in the samples previously. Other than this use of liquid nitrogen, the two extraction protocols were identical.

RNA was extracted using method LqN from four untreated *A. thaliana* plants, and qRT-PCR analysis performed on these samples. Figure 3.3 shows a comparison of the mean C_t values observed from samples extracted by the two methods.

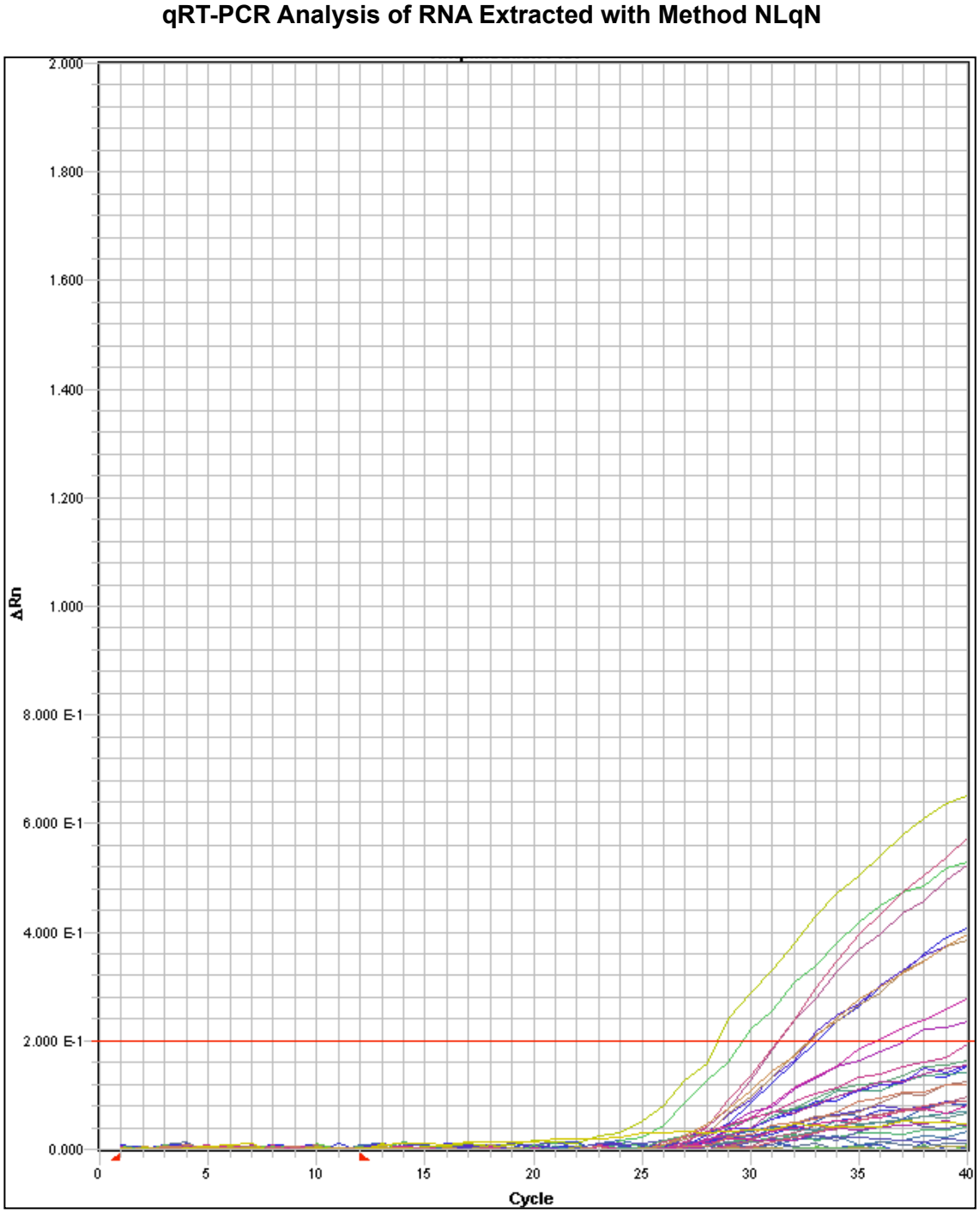


Figure 3.2 *Example of qRT-PCR amplification profiles observed with COX assay of RNA samples extracted from* A. thaliana *tissue using NLqN method.*

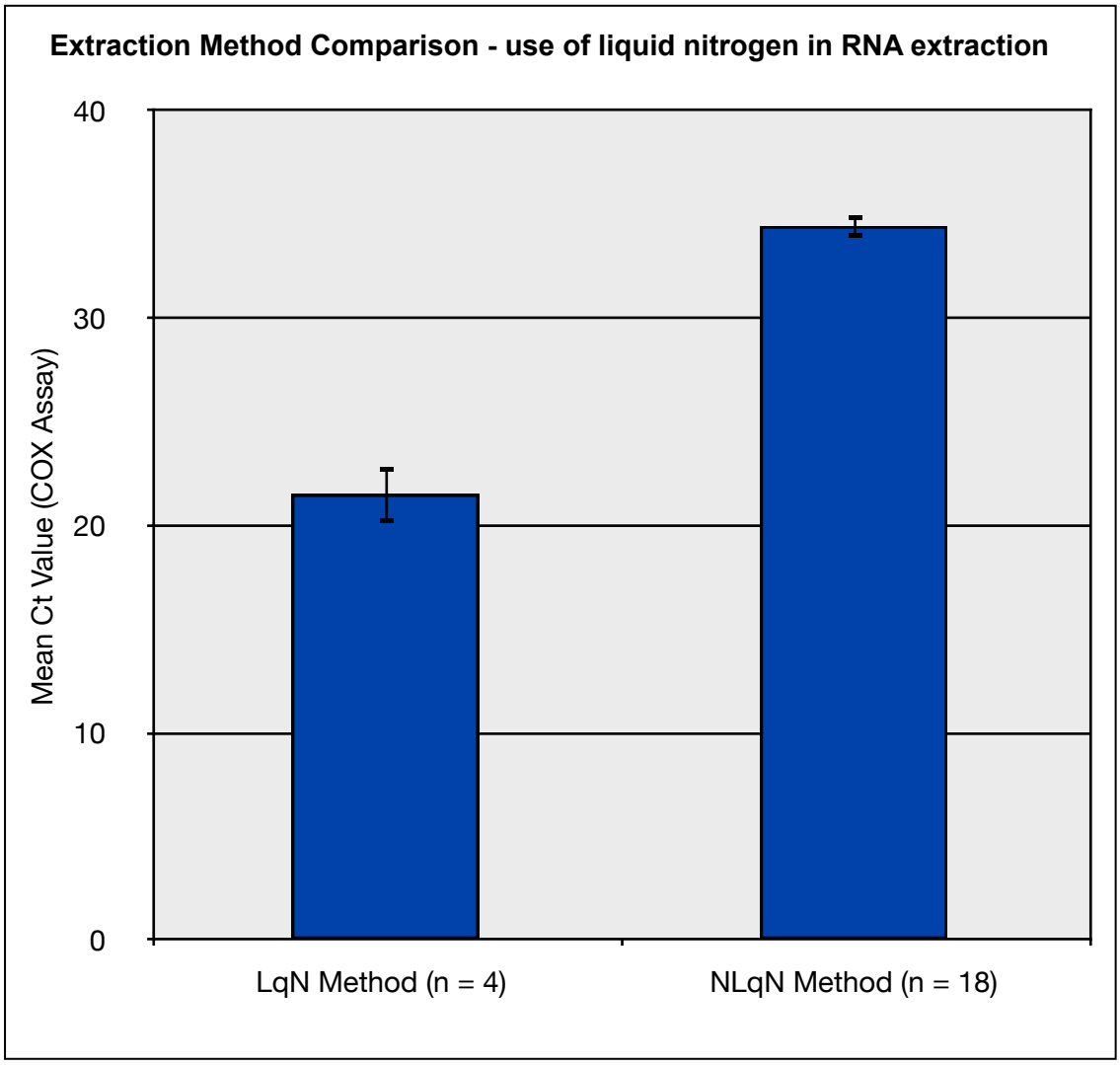


Figure 3.3 *A comparison of average Ct values observed from qRT-PCR analysis with cytochrome oxidase assay of RNA extracted from plant tissue samples using methods with (right) and without (left) liquid nitrogen. Error bars show the standard error of the mean values given.*

The COX assay-derived C_t values observed from RNA samples extracted by method LqN were consistently lower and showed less variation than those from samples where no liquid nitrogen was used during extraction. These results supported the hypothesis that the poor amplification observed with the COX assay (Fig. 3.2) was a result of the degradation of RNA during the extraction process, and that this effect could be reduced by ensuring that tissue samples remained frozen at all times during homogenisation.

Due to the consistently improved qRT-PCR results observed from samples extracted by method LqN, this protocol was used for all further RNA extractions.

Latent CMV infection in *A. thaliana* plants

In preliminary qRT-PCR experiments, low-level amplification was observed with the CMV assay in RNA extracts from untreated and dummy inoculated plants. As precautions had been taken to prevent cross-contamination between treatment groups during sample prepartion, it was hypothesised that the low-level amplification of CMV in samples from those plants not inoculated with the virus could be attributed to a latent infection of the plants, probably due to seed contamination.

To further test this hypothesis, a newly grown batch of plants was treated in the three separate viral treatment groups as described previously. After four days, tissue samples were taken from three plants in each treatment group. RNA extracts were taken and analysed by qRT-PCR. The results are illustrated as amplification profiles of the CMV assay in Figure 3.4. Pairs of traces corresponding to CMV inoculated samples are marked with an asterisk. The C_t values of the three CMV inoculated samples ranged between 20-27 cycles, whilst those of untreated and dummy inoculated samples ranged between 33-40 cycles.

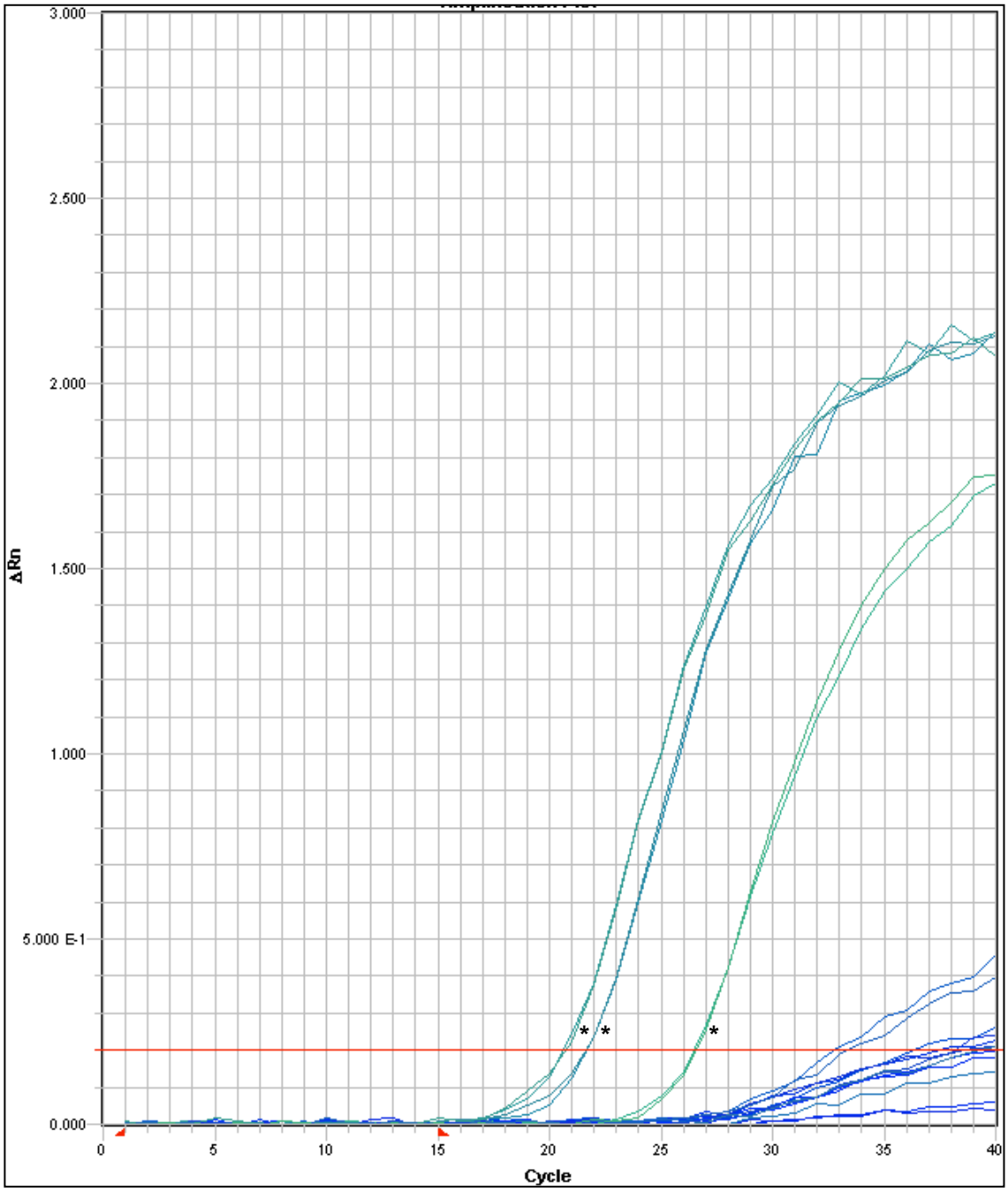


Figure 3.4 *Amplification profile of fluorescence observed with CMV-specific TaqMan assay of three tissue samples taken from each treatment group four days post-inoculation. Fluorescence increases proportionally with an increase in concentration of the pathogen-specific assay target sequence. Traces observed from CMV inoculated samples are marked with an asterisk (*), while those from untreated and dummy inoculated samples are left unmarked above.*

Two conclusions were drawn from the amplification profiles illustrated in Fig. 3.4.

Firstly, the marked difference between C_t values obtained from CMV-inoculated samples and those from untreated and dummy inoculated plants indicated that viral inoculations had been successful. The amplification profiles of the three CMV-inoculated samples indicated that the concentration of CMV material in these RNA extracts was ~3-6 orders of magnitude greater than in the untreated and dummy inoculated samples, only four days after inoculation.

Secondly, the low-level amplification of the CMV assay observed again in these untreated and dummy inoculated samples added further weight to the hypothesis that the *A. thaliana* plants used in these experiments carried a latent CMV infection. Some amplification was detected in all samples, although the level of fluorescence did not exceed the threshold during the set number of amplification cycles in some cases

Assuming that no non-specific amplification occurred with the assay, no fluorescence should be observed from the assay probe in the results of an qRT-PCR experiment if the starting concentration of assay target sequence in the sample is zero. No CMV was detected in negative control samples run in the qRT-PCR experiments, where reactions were prepared in the absence of an RNA sample. This lack of activity indicated that any fluorescence observed from the assay in the experimental samples could not be attributed to non-specific amplification.

Due to the relatively negligible level of CMV infection observed in untreated and dummy inoculated plants, it was considered unlikely that it would adversely affect the quality of any sequencing results obtained from these *A. thaliana* plants. Even in the event that samples from these plants were sequenced deeply enough for this low-level infection to be detected, the coverage of the CMV genome that could be expected from samples of those plants inoculated with the virus would be so much greater as to easily distinguish the samples from this treatment group.

qPCR analysis of DNA extracts in preparation for sequencing

DNA was extracted 15 days post-inoculation, with 25 tissue samples taken from 14 plants in each bacterial treatment group, giving 75 samples in total for sequencing. To test for cross-contamination in the samples from different treatment groups, these DNA extracts were analysed by qPCR. The results of these experiments can be found in Table 3.8 and Figure 3.5.

COX assay results were consistent between all three treatment groups, with mean C_t values ranging between ~30-35 cycles. The samples taken from untreated plants exhibited no activity with the *P. syringae* assay. Fluorescence from the *P. syringae*-specific TaqMan assay exceeded the threshold value in only a single replicate of one sample of dummy inoculated plants, at 39.05 cycles. This detection during the final cycle of the experiment indicated only a very low concentration of *P. syringae* DNA and was most likely the result of a pipetting error or airborne contamination between wells in the plate during preparation.

The mean C_t of samples from bacteria-inoculated plants with *P. syringae*-specific assay was 26.72 cycles, indicating that inoculation of plants was achieved as intended. No cross-contamination was observed in untreated or dummy inoculated samples.

These results suggested that the DNA samples extracted from these plants were suitable for sequencing.

However, the relative concentrations of DNA from *P. syringae* and *A. thaliana* could not be estimated accurately from the qPCR results described here. In order for such an estimation to be made, a means of calibration would be required that would allow for the C_t values observed here to be translated into more informative estimations of the concentration of assayed sequence present within the sample. The most common approach taken to calibration is to produce a standard curve of C_t values observed from q(RT-)PCR analysis of several samples of known concentration of target DNA/RNA. Using the curve extrapolated from these standardised C_t values, experimental values obtained from q(RT-)PCR of samples can be converted to an estimate of the starting concentration of target DNA/RNA in the sample.

The lack of such a means of estimating the concentrations of plant and

bacterial/viral DNA or RNA in the extracted samples is a major limitation on the interpretation of the results given here and elsewhere in this work.

Table 3.8 *Mean C_t values observed in qPCR analysis with COX- and* P. syringae *pv.* tomato *DC3000-specific TaqMan assays of DNA extracted from plant tissue samples from three bacterial treatment groups.*

Treatment Group	Mean Ct (COX Assay)	Mean Ct (P. syringae Assay)
Untreated	32.0266	not detected
Dummy Inoculated	31.0688	39.97
P. syringae Inoculated	*35.4511*	26.72

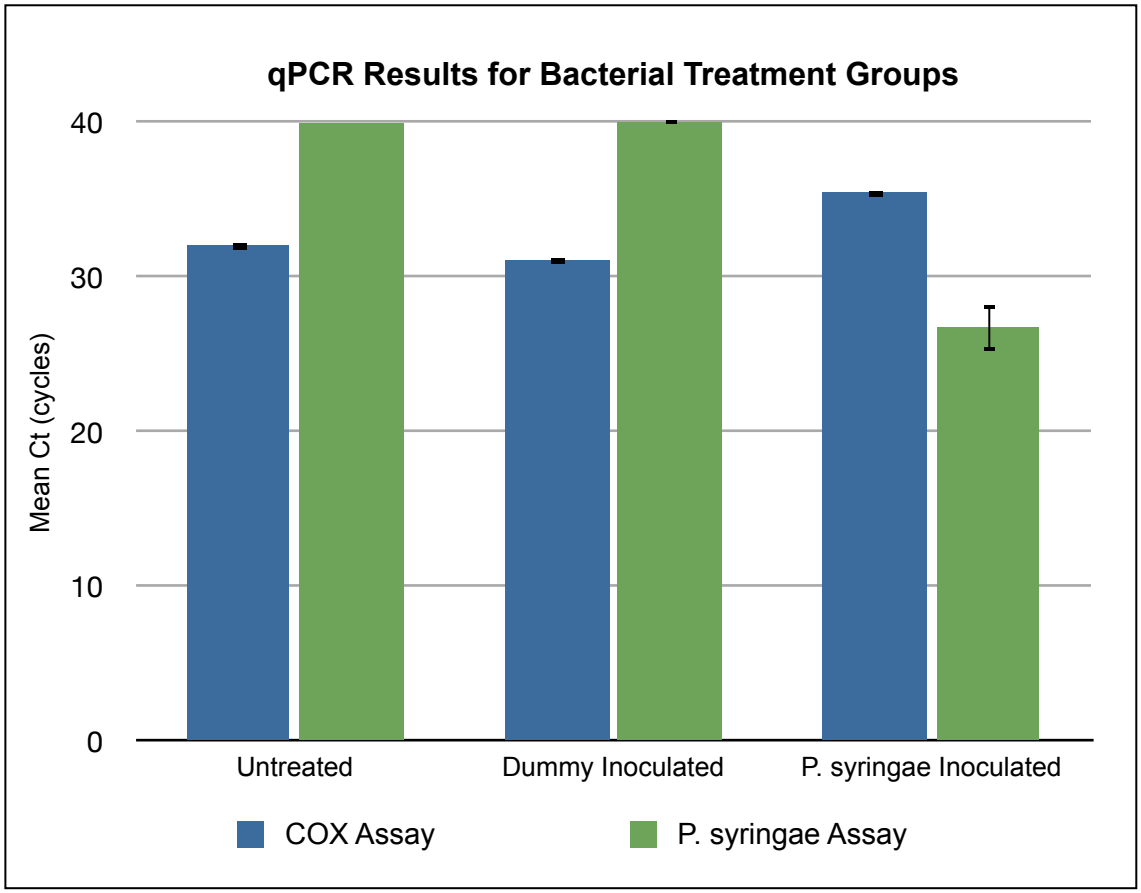


Figure 3.5 *Mean C_t values observed in qPCR analysis with COX- and* P. syringae *pv.* tomato *DC3000-specific TaqMan assays of DNA extracted from plant tissue samples from three bacterial treatment groups.*

qRT-PCR analysis of RNA extracts in preparation for sequencing

RNA was extracted 12 days post-inoculation, with 40 tissue samples taken from from 17 plants in each viral treatment group, giving 120 samples in total. To test for any cross-contamination between plants and samples, and for the low-level CMV infection identified previously, these RNA extracts were analysed by qRT-PCR. The results of these experiments can be seen in Table 3.9 and Figure 3.6. Figures 3.7-3.9 provide a comparison of CMV assay amplification profiles of samples from each of the three viral treatment groups.

The qRT-PCR results from analysis of untreated and dummy inoculated samples provided further evidence that the *A. thaliana* plants used in these experiments carried very low levels of CMV (Figs 3.7 and 3.8). When analysed with CMV TaqMan assay, untreated samples returned a mean C_t value of 33.09 cycles, while samples from dummy inoculated plants gave a mean C_t of 38.22 cycles. At 23.06 cycles, the mean C_t of these virus-inoculated plant samples was ~10 cycles lower than that of untreated plant samples and ~15 cycles lower than that of dummy inoculated plants. These results indicated a starting concentration of CMV RNA in CMV-inoculated samples ~10^3 x higher than that in untreated samples and ~10^4-10^5 x higher than that in dummy inoculated samples.

The results obtained from CMV-inoculated samples suggested that the viral inoculations performed on plants in this treatment group were successful, confirming what was previously indicated by the results of an equivalent analysis, of samples taken four days post-inoculation.

A difference of approximately five cycles was observed between the mean C_t values obtained from untreated and dummy inoculated samples, 12 days after inoculation. With the former set of samples also exhibiting several amplification profiles stronger than were observed from any of the dummy inoculated samples, these results suggested that untreated plant samples contained a higher average starting concentration than from those taken from dummy inoculated plants.

As the dummy inoculation process was the only difference in treatment between these two groups, it was concluded that the difference in the levels of latent CMV in the tissue samples was a consequence of this treatment.

The full inoculation method involved rubbing the surface of a single leaf with a mixture of homogenised infected plant tissue and Celite (powdered diatomite) in a phosphate buffer. Rubbing with Celite causes abrasion to the leaf surface, improving the likelihood of uptake of virus. The dummy inoculation methodology involved rubbing the leaf surface with a mixture of Celite and phosphate buffer only. As such, though no uptake of virus would have occurred, the abrasion caused by the Celite still damaged the leaf surface. Systemic wound response to this kind of leaf tissue damage is well documented in plants (see (Sun, Jiang et al. 2011) for a recent review of this effect), and such a response was predicted to be the cause of the lower levels of CMV infection observed in dummy inoculated plants. The tissue damage sustained to the inoculated leaf triggered a rapid systemic response that increased the plants resistance, resulting in a decrease in the titre of CMV present in the tissue samples taken for analysis.

Although CMV was detected in all the samples from viral treatment groups, the considerably greater concentrations of viral RNA detected in the CMV inoculated samples indicated that the inoculations performed had been successful. As discussed previously, the levels of viral RNA detected in samples from untreated and dummy inoculated plants were several orders of magnitude lower than in those from CMV inoculated plants, and, with this difference taken into consideration, alongside constraints on time and resources, it was concluded that these samples were suitable for sequencing.

If the level of CMV material present in the untreated and/or dummy inoculated RNA extracts was great enough to contribute to the reads produced in sequencing, the proportion of CMV reads in these datasets should be significantly smaller than in those generated from the CMV-inoculated samples, in correspondence with the difference observed in these qRT-PCR results.

The difference, observed by qRT-PCR analysis, in levels of CMV present in untreated and in dummy inoculated plant samples may also be reflected in the sequencing results obtained from these treatment groups, with fewer CMV reads being generated from dummy inoculated plant RNA than untreated plant RNA.

Unfortunately, due to the lack of a means of calibration, it was not possible to

quantify these differences between levels of CMV present in samples from different treatment groups and between the levels of viral and plant cDNA within each sample. As such, specific estimations of relative concentrations and subsequent predictions of the likely proportions per species of sequencing reads produced from the samples were not possible.

Table 3.9 *Mean C_t values observed in qRT-PCR analysis with COX- and CMV-specific TaqMan assays of RNA extracted from plant tissue samples from three treatment groups.*

Treatment Group	Mean Ct (COX Assay)	Mean Ct (CMV Assay)
Untreated	26.6986	33.09
Dummy Inoculated	25.9026	38.22
CMV Inoculated	26.717	23.06

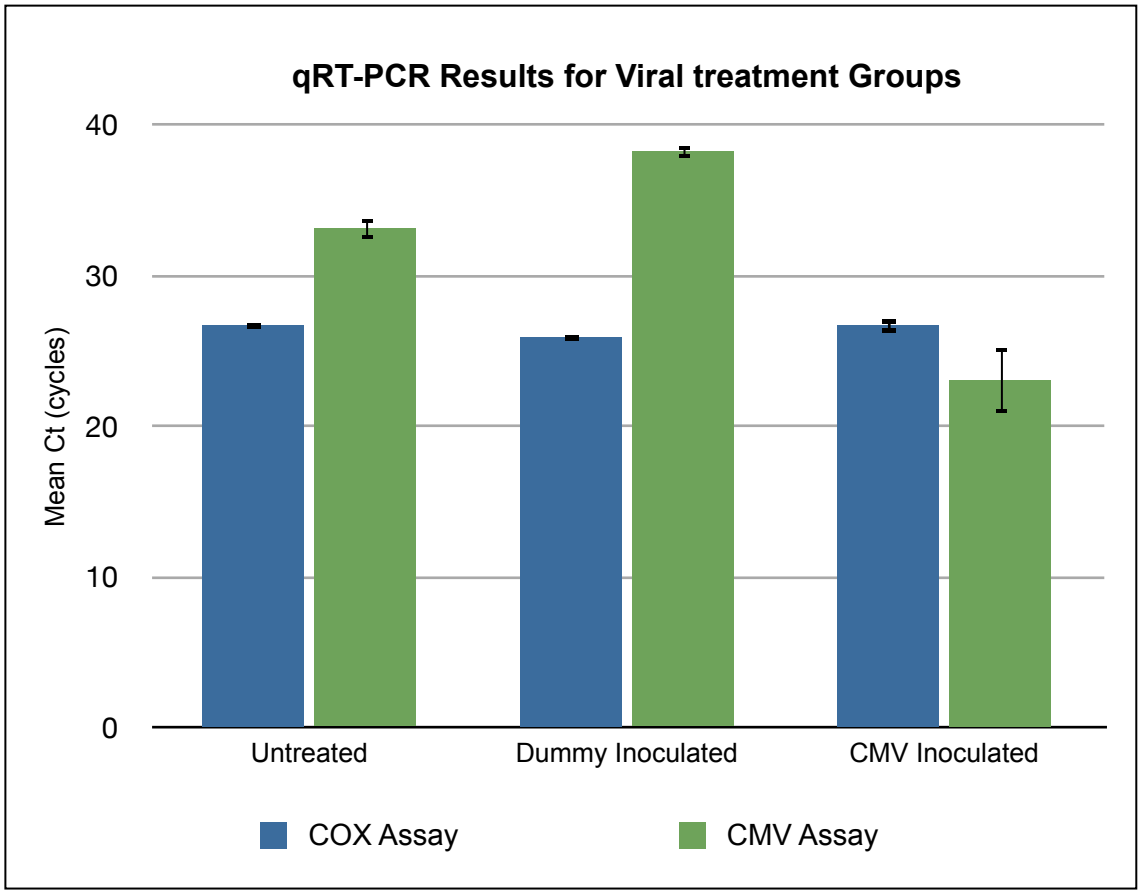


Figure 3.6 *Mean C_t values observed in qRT-PCR analysis with COX- and CMV-specific TaqMan assays of RNA extracted from plant tissue samples from three treatment groups.*

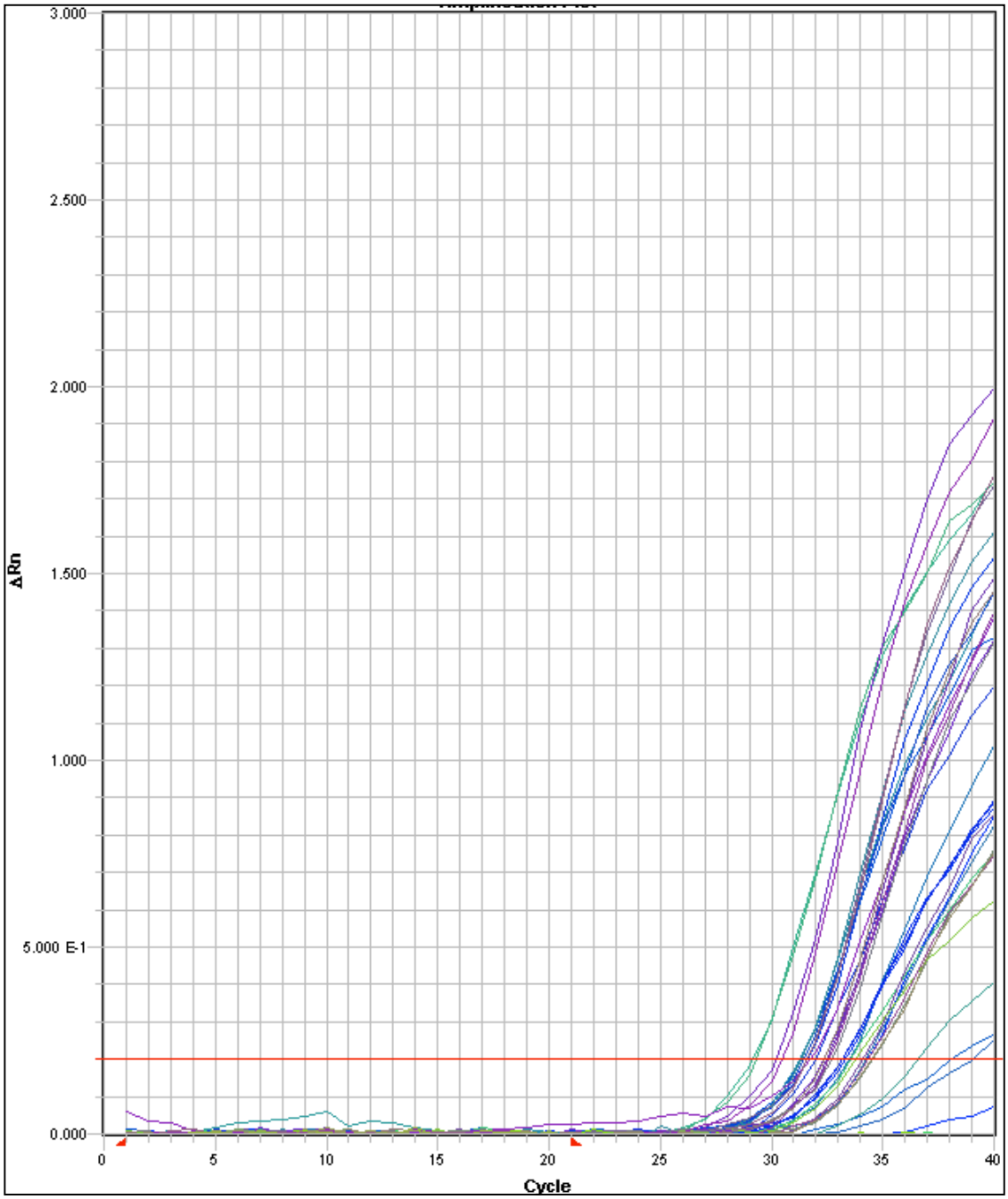


Figure 3.7 *Amplification profile of fluorescence observed in qRT-PCR analysis with CMV-specific TaqMan assay of RNA extracted from plant tissue samples of untreated plants, to be used in sequencing.*

qRT-PCR Analysis of Dummy Inoculated Plant Samples

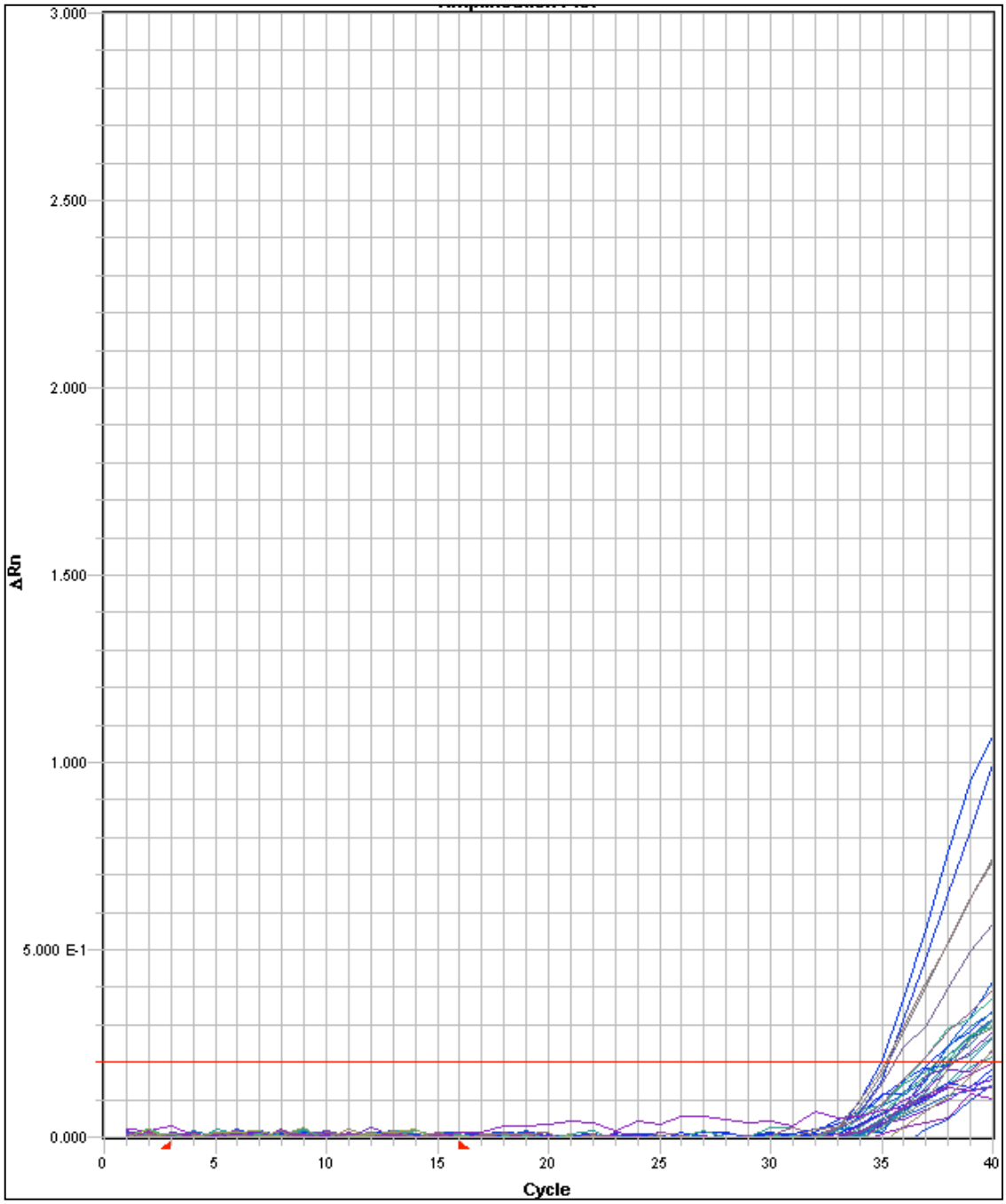


Figure 3.8 *Amplification profile of fluorescence observed in qRT-PCR analysis with CMV-specific TaqMan assay of RNA extracted from plant tissue samples of dummy inoculated plants, to be used in sequencing.*

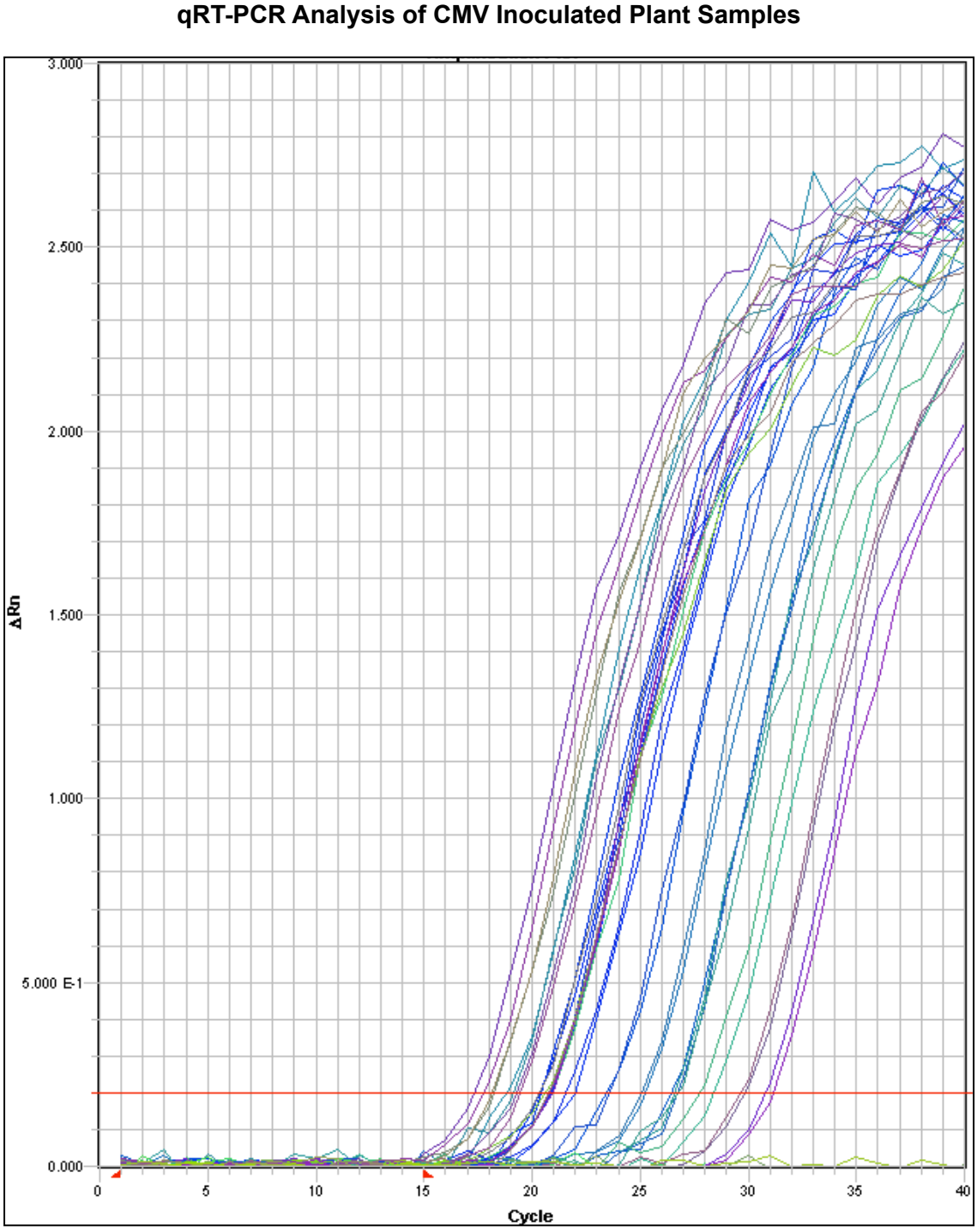


Figure 3.9 *Amplification profile of fluorescence observed in qRT-PCR analysis with CMV-specific TaqMan assay of RNA extracted from plant tissue samples of CMV inoculated plants, to be used in sequencing.*

Results of high-throughput DNA sequencing - read breakdown

- **Bacterial treatment groups**

Table 3.10 provides a breakdown, by reference sequence, of assignments of sequencing reads from the three bacterial treatment groups, produced by mapping to the *A. thaliana* and *P. syringae* pv. *tomato* DC3000 genomes with *SSAHA2*.

As illustrated in Figure 3.10, the relative proportion of reads assigned to each genome remained broadly consistent between the three samples. The dataset (defined as the set of all reads returned from sequencing of the DNA sample) of 106,294 reads from untreated plants was found to contain 92,209 reads (86.75%) assigned to *A. thaliana*, 905 (0.85%) to *P. syringae*, and 13,180 (12.40%) reads that could not be assigned to either genome. The corresponding proportions found in the dummy inoculated (115,614 reads) and the *P. syringae* inoculated (116,874 reads) datasets were 103,247 (89.30%) and 97,612 (83.52%) reads assigned to *A. thaliana*, 297 (0.26%) and 369 (0.32%) reads assigned to *P. syringae*, and 12,070 (10.44%) and 18,893 (16.17%) reads left unassigned respectively.

The read assignments detailed in Table 3.5 also displayed a consistency between datasets in the proportion of reads assigned to individual reference sequences. This pattern can be better identified in Figure 3.11. Sequencing reads from some chromosomes of *A. thaliana* appeared to be over-represented, and the same pattern of representation of each of the plants five chromosomes was observed in the *SSAHA2* results from each of the three datasets.

The relative proportions of *A. thaliana*, *P. syringae* and unassigned sequencing reads given in Figure 3.10 for each bacterial treatment group were broadly similar. Most notably, the dataset of reads obtained from *P. syringae* inoculated plants did not contain more reads mapped to the bacterial genome. In fact, the dataset containing the most reads assigned to *P. syringae* was found to be that derived from untreated plants (905 reads compared to 369 from *P. syringae* inoculated plants and 297 from dummy inoculated plants).

Table 3.10 *Number of sequencing reads from each treatment group assigned to reference* A. thaliana *and* P. syringae *pv.* tomato *DC3000 genome sequences. Reads were assigned using* SSAHA2*, with an additional E-value cutoff threshold of 1 x* 10^{-4}*. Also given are numbers of sequencing reads from each treatment group that were not assigned to any of the reference sequences, either due to the lack of an alignment with a suitably low E-value being found, or the lack of any alignment being found.*

Reference Sequence	Untreated Plant Reads	Dummy Inoculated Plant Reads	*P. syringae* Inoculated Plant Reads
***A. thaliana* Chr. 1**	16286	18289	17293
***A. thaliana* Chr. 2**	15100	16828	16056
***A. thaliana* Chr. 3**	20449	22093	22349
***A. thaliana* Chr. 4**	12643	13939	13939
***A. thaliana* Chr. 5**	17709	19594	19087
***A. thaliana* Chloroplast**	7214	9543	5822
***A. thaliana* Mitochondrion**	2808	2961	3066
***P. syringae* pv. *tomato* DC3000**	888	297	368
Plasmid pDC3000A	7	0	0
Plasmid pDC3000B	10	0	1
Unassigned (E-value above cutoff)	1214	1462	1246
Unassigned (no alignment found)	11966	10608	17647
Total	106294	115614	116874

Figure 3.10 *Proportion of sequence reads from each bacterial treatment group that were mapped to the reference genome of A. thaliana or P. syringae pv. tomato DC3000 with SSAHA2. The proportion of reads left unmapped is also shown. A more detailed breakdown of these reads assignments is provided in Table 3.5.*

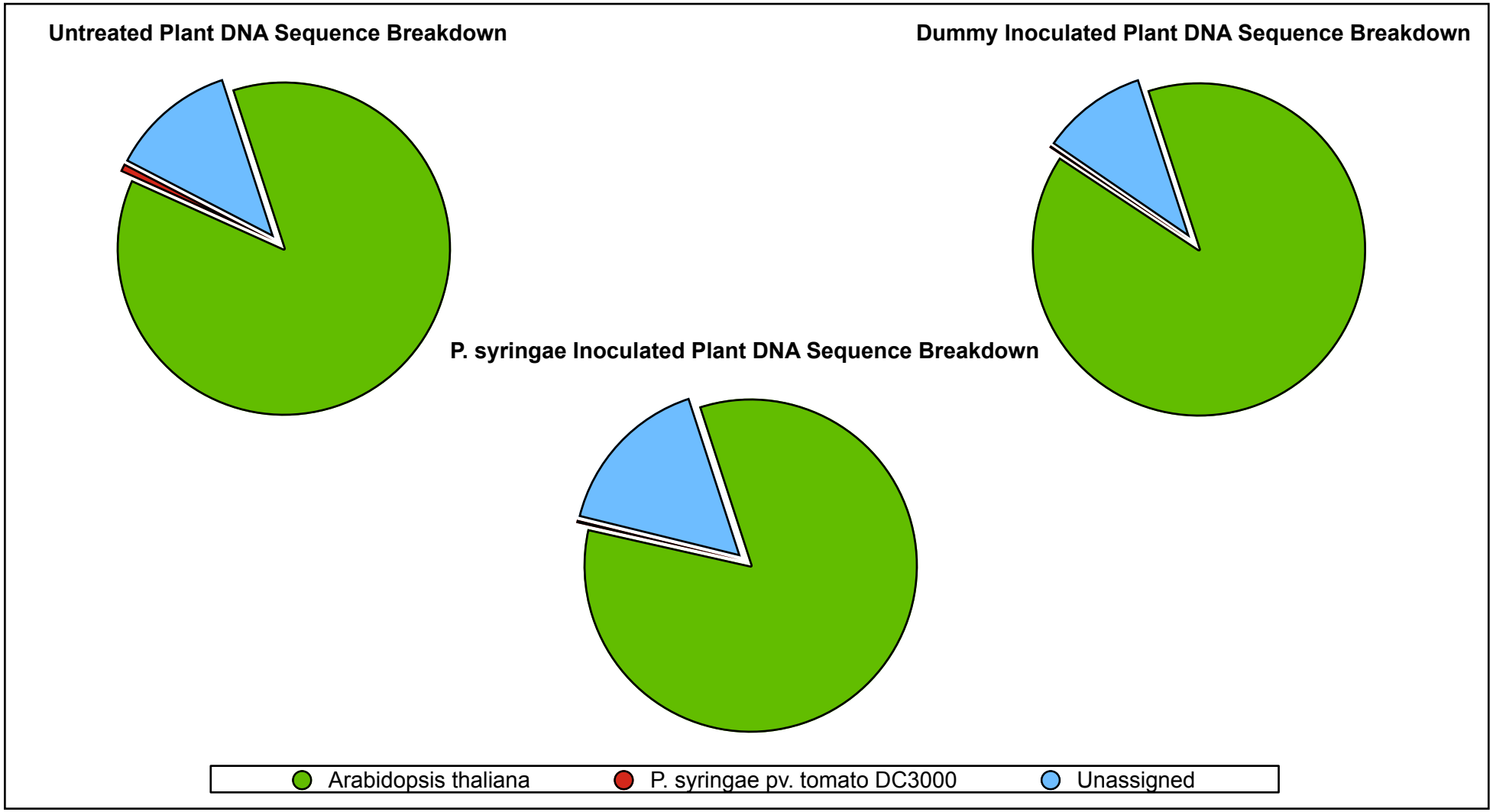
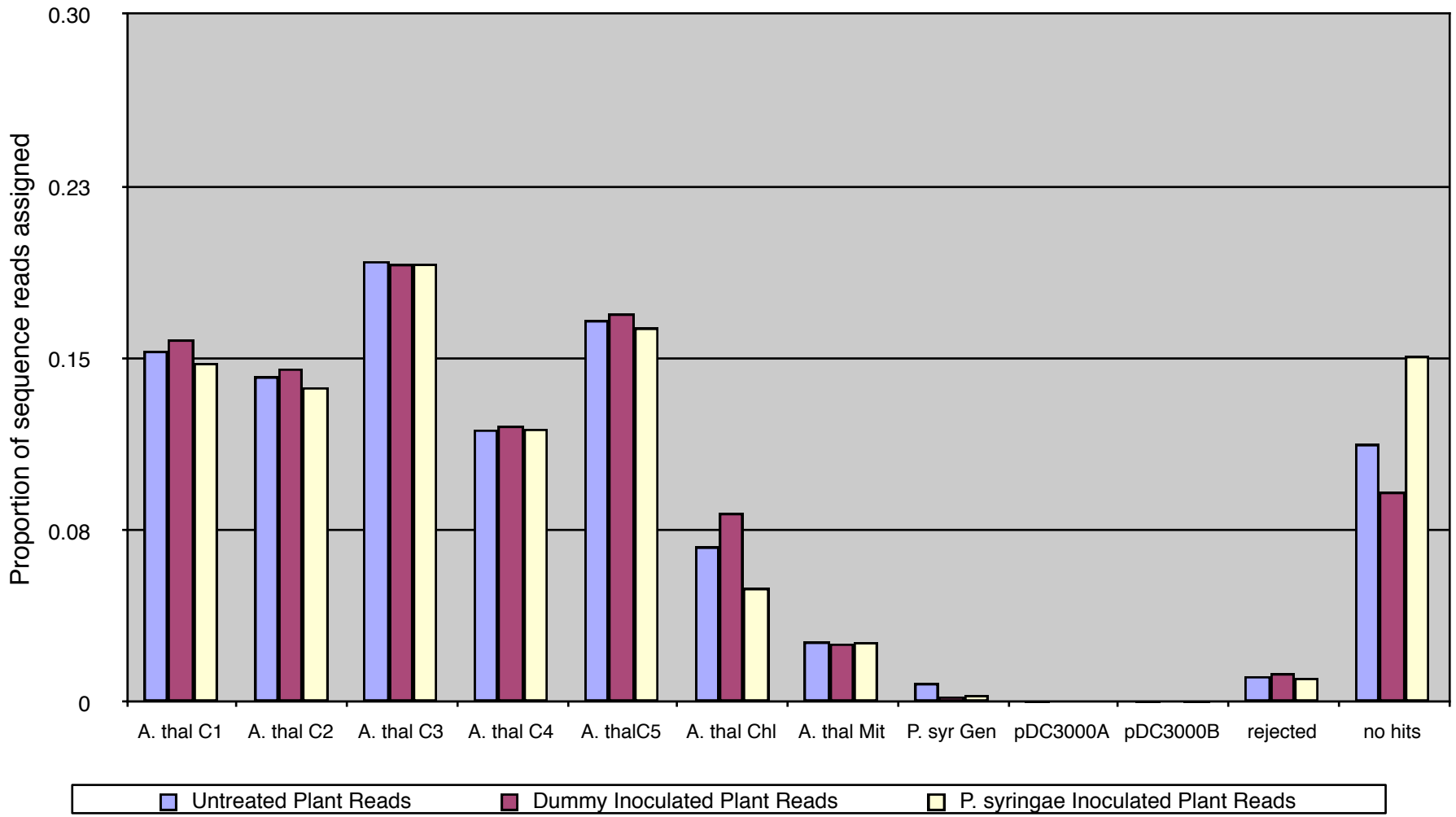

Untreated Plant DNA Sequence Breakdown
Dummy Inoculated Plant DNA Sequence Breakdown
P. syringae Inoculated Plant DNA Sequence Breakdown
Arabidopsis thaliana
P. syringae pv. tomato DC3000
Unassigned

Figure 3.11 *Proportion of sequence reads from each bacterial treatment group that were mapped to each reference sequence with* SSAHA2. *The proportion of reads left unmapped is also shown, with reads with assignments rejected based on expectation value labelled as 'rejected' and those where no hits were found with SSAHA2 labelled as 'no hits'.*


0.30
0.23
0.15
0.08
0
Proportion of sequence reads assigned
A. thal C1
A. thal C2
A. thal C3
A. thal C4
A. thalC5
A. thal Chl
A. thal Mit
P. syr Gen
pDC3000A
pDC3000B
rejected
no hits
Untreated Plant Reads
Dummy Inoculated Plant Reads
P. syringae Inoculated Plant Reads

- **Viral treatment groups**

Table 3.11 provides a breakdown of sequencing read assignments to the genomes of *A. thaliana* and CMV, mapped by *SSAHA2*. No reads from the datasets obtained from untreated and dummy inoculated plants were assigned to the CMV genome, while 273 (0.20%) reads from CMV inoculated plants were assigned to the viral genome. 21,738 (17.46%), 24,704 (18.63%), and 25,482 (18.93%) reads remained unassigned from the untreated, dummy inoculated, and CMV inoculated datasets respectively. As Figure 3.12 shows more clearly, the ratio of plant, virus and unassigned sequencing reads remained broadly consistent between the three datasets, with the key difference that only the data obtained from CMV inoculated plants contained any reads that were assigned to the CMV genome.

Similarly to the data obtained from DNA sequencing of the bacterial treatment groups, the proportions of reads assigned to individual reference sequences remained consistent between viral treatment groups. This consistency of spread between reference sequences can be seen in Figure 3.13, with an over-representation of reads assigned to chromosome 3 of *A. thaliana* being particularly prominent. In each of the three datasets described in Table 3.11, over half of all sequencing reads produced for each sample were assigned to this chromosome.

Table 3.11 *Number of sequencing reads from each treatment group assigned to reference* A. thaliana *and* Cucumber mosaic virus *genome sequences. Reads were assigned using* SSAHA2*, with an additional E-value cutoff threshold of 1 x* 10^{-4}*. Also given are numbers of sequencing reads from each treatment group that were not assigned to any of the reference sequences, either due to the lack of an alignment with a suitably low E-value being found, or the lack of any alignment being found.*

Reference Sequence	Untreated Plant Reads	Dummy Inoculated Plant Reads	CMV Inoculated Plant Reads
***A. thaliana* Chr. 1**	2105	2332	2159
***A. thaliana* Chr. 2**	11096	11386	11052
***A. thaliana* Chr. 3**	68682	72117	75463
***A. thaliana* Chr. 4**	947	1281	792
***A. thaliana* Chr. 5**	1409	1769	1075
***A. thaliana* Chloroplast**	13050	12957	10720
***A. thaliana* Mitochondrion**	5503	6064	7579
CMV RNA 1	0	0	131
CMV RNA 2	0	0	1
CMV RNA 3	0	0	141
Unassigned (E-value above cutoff)	10752	13280	12233
Unassigned (no alignment found)	10986	11424	13249
Total	124530	132610	134595

Figure 3.12 *Proportion of sequence reads from each viral treatment group that were mapped to the reference genome of* A. thaliana *or CMV with* SSAHA2. *The proportion of reads left unmapped is also shown. A more detailed breakdown of these reads assignments is provided in Table 3.6*.

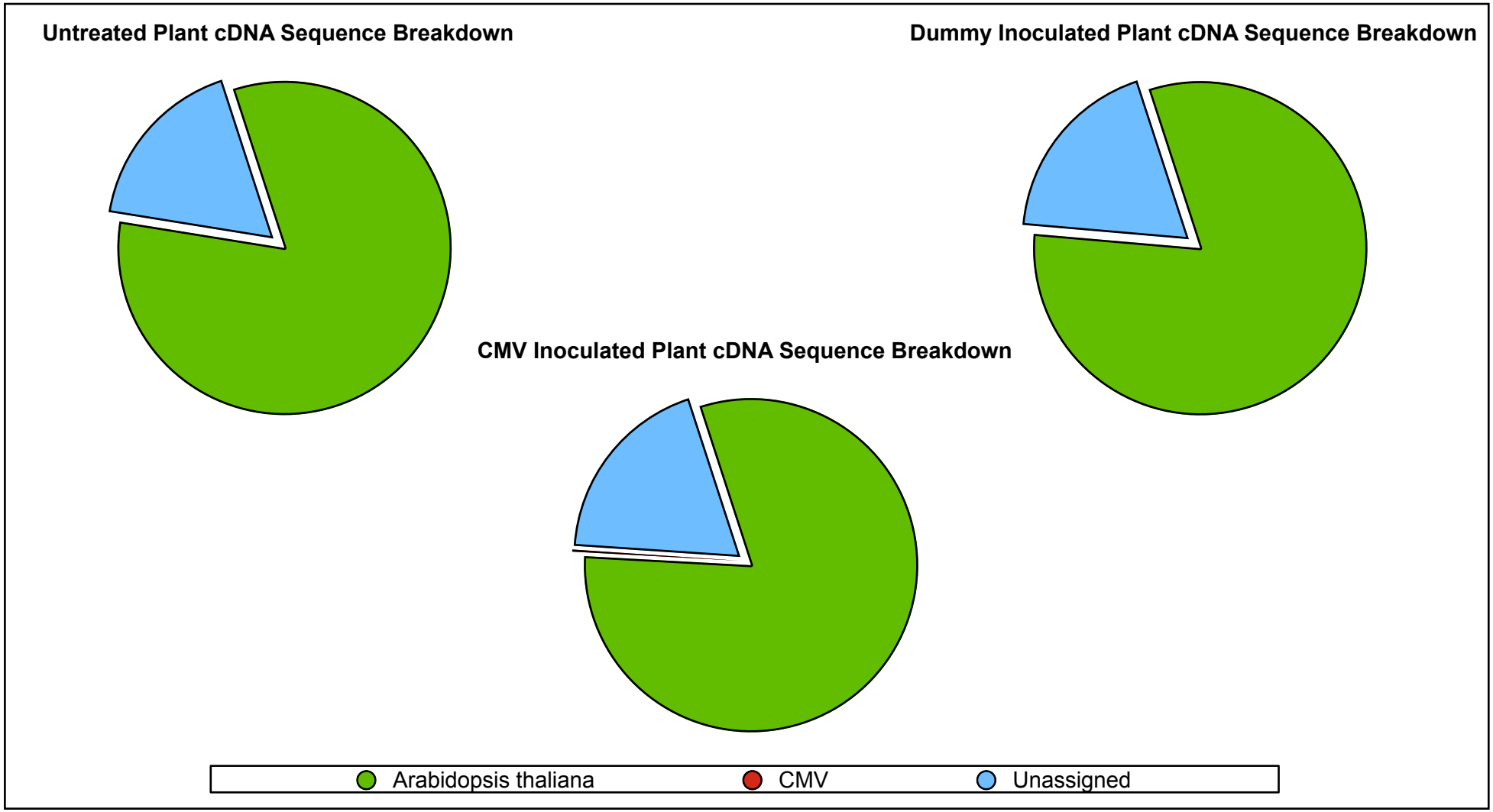
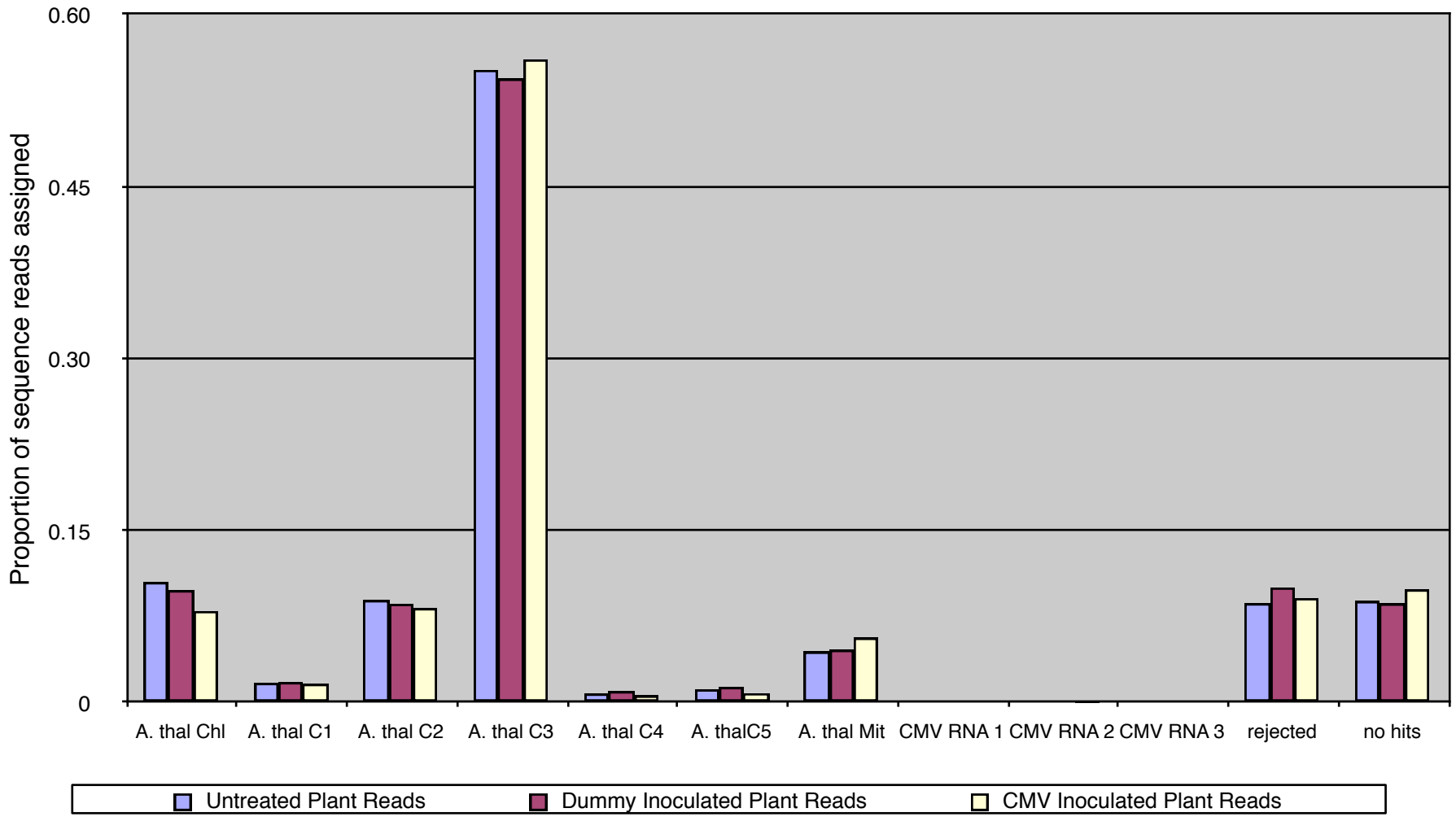

Untreated Plant cDNA Sequence Breakdown
Dummy Inoculated Plant cDNA Sequence Breakdown
CMV Inoculated Plant cDNA Sequence Breakdown
Arabidopsis thaliana
CMV
Unassigned

Figure 3.13 *Proportion of sequence reads from each viral treatment group that were mapped to each reference sequence with* SSAHA2. *The proportion of reads left unmapped is also shown, with reads with assignments rejected based on expectation value labelled as 'rejected' and those where no hits were found with SSAHA2 labelled as 'no hits'.*


Proportion of sequence reads assigned
0.60
0.45
0.30
0.15
0
A. thal Chl
A. thal C1
A. thal C2
A. thal C3
A. thal C4
A. thalC5
A. thal Mit
CMV RNA 1
CMV RNA 2
CMV RNA 3
rejected
no hits
Untreated Plant Reads
Dummy Inoculated Plant Reads
CMV Inoculated Plant Reads

Discussion

Datasets produced from bacterial treatment groups

Results of previous qPCR analyses had confirmed the presence of *P. syringae* DNA in only the samples from plants inoculated with the bacteria. This suggested that the assignment of reads in sequencing data from untreated and dummy inoculated samples to this genome, which accounted for fewer than 1% of the total reads produced, was likely to be the result of chance alignment or sequencing error producing reads that were mapped to the bacterial genome over that of *A. thaliana*.

Given that the proportion of sequencing reads from *P. syringae* inoculated samples that were mapped to the bacterial genome was of the same order of magnitude as in the other two datasets, it could not be concluded that the presence of these reads was due to bacterial inoculation of the plants in this treatment group.

Instead, it was most likely that the reads that did map to the *P. syringae* DC3000 genome in the *SSAHA2* analysis originated from bacteria that were naturally present in all of the samples e.g. on the surface of the leaf tissue harvested for extraction. Reads originating from the genome of other bacterial species would be more likely to map to the bacterial *Pseudomonas* genome than to that of *A. thaliana*, and these were the only reference sequences available during mapping. Further investigation of the alignments between these reads and the *P. syringae* genome, comparing the score and degree of sequence identity observed to that obtained in alignment to a broader range of reference sequences, e.g. all available bacterial genome sequences or the NCBI *nt* database of all non-redundant publicly available nucleotide sequences, would allow greater confidence in this prediction. If these reads were still found to map to a bacterial reference in these less-specific databases, but with a greater degree of sequence identity, it could be concluded that other bacteria present in the sample were their probable source.

In a dataset of sequencing reads, the constituent sequences of the original sample should be represented in broadly similarly proportions to their relative abundance, although some bias in the proportions of reads produced has been shown to exist (Morgan, Darling et al. 2010). The extent of sequencing of

material present in relatively small proportions in the sample is largely dependent on how ‘deep’ the sequencing goes - that is, how large-scale the experiment is in terms of number of parallel sequencing reactions carried out.

These results indicated that the level of *P. syringae* DNA present in the inoculated sample was insufficient to be detected by sequencing performed on this scale.

The three datasets produced from bacterial treatment groups did not contain a sufficient number of bacterial sequencing reads, in proportions great enough to be suitable for use in clustering analysis as intended.

Datasets produced from viral treatment groups

While the relative proportions of reads assigned to the reference genome of *A. thaliana* and reads left unassigned were similar between the datasets produced from the three viral treatment groups, one important difference existed between these datasets and those from the bacterial treatment groups discussed previously. In this case, the only dataset that was found to contain reads that were mapped to the CMV genome was that derived from CMV-inoculated plants, despite the qRT-PCR results indicating that samples from all three treatment groups had contained CMV RNA (and subsequently cDNA) in some quantity.

These results suggested that the scale of sequencing performed here was just large enough for the detection of CMV present at the levels observed in the samples from virus inoculated plants, but not for the detection of the virus when present in the lower titres described for untreated and dummy inoculated samples.

It is interesting to note that no reads were assigned by chance or sequencing error to the CMV genome in these datasets, which was believed to be the case with *P. syringae*-assigned reads in the data obtained from bacterial treatment group samples.

The relative quantities of *A. thaliana* (~81%) and CMV (~0.2%) sequencing reads present in the data obtained from virus-inoculated samples were too disproportionate, and the total number of CMV reads assigned by *SSAHA2* too small, for the dataset to be useable for the clustering analysis experiments discussed throughout the other sections of this work.

Conclusion and future work

The aim of the experiments described here was to produce a pair of datasets containing high-throughput sequencing reads derived from a natural system containing species with fully sequenced genomes, and to assess the suitability of these datasets for use in the appraisal of a range of clustering methods.

Despite qPCR/qRT-PCR analysis results indicating the presence of pathogen material in considerable concentrations in the samples prepared for both *P. syringae* and CMV, the sequencing datasets produced were not found to contain reads originating from these species in sufficient numbers to be suitable for use in the evaluation of clustering methods.

The very low quantities of pathogen sequencing reads produced in both cases suggested that these sequences were only represented in very low concentration in the sequenced samples. The short length of the pathogen genomes, relative to that of the plant host, means that, even if sequence from these pathogen genomes was present in a high copy number in these samples, it might constitute only a small fraction of the overall sequence present. As such, sequencing on a larger scale would be require for more reads from these genomes to be produced. This deeper sequencing would not affect the overall proportions of host and pathogen sequences present in the datasets.

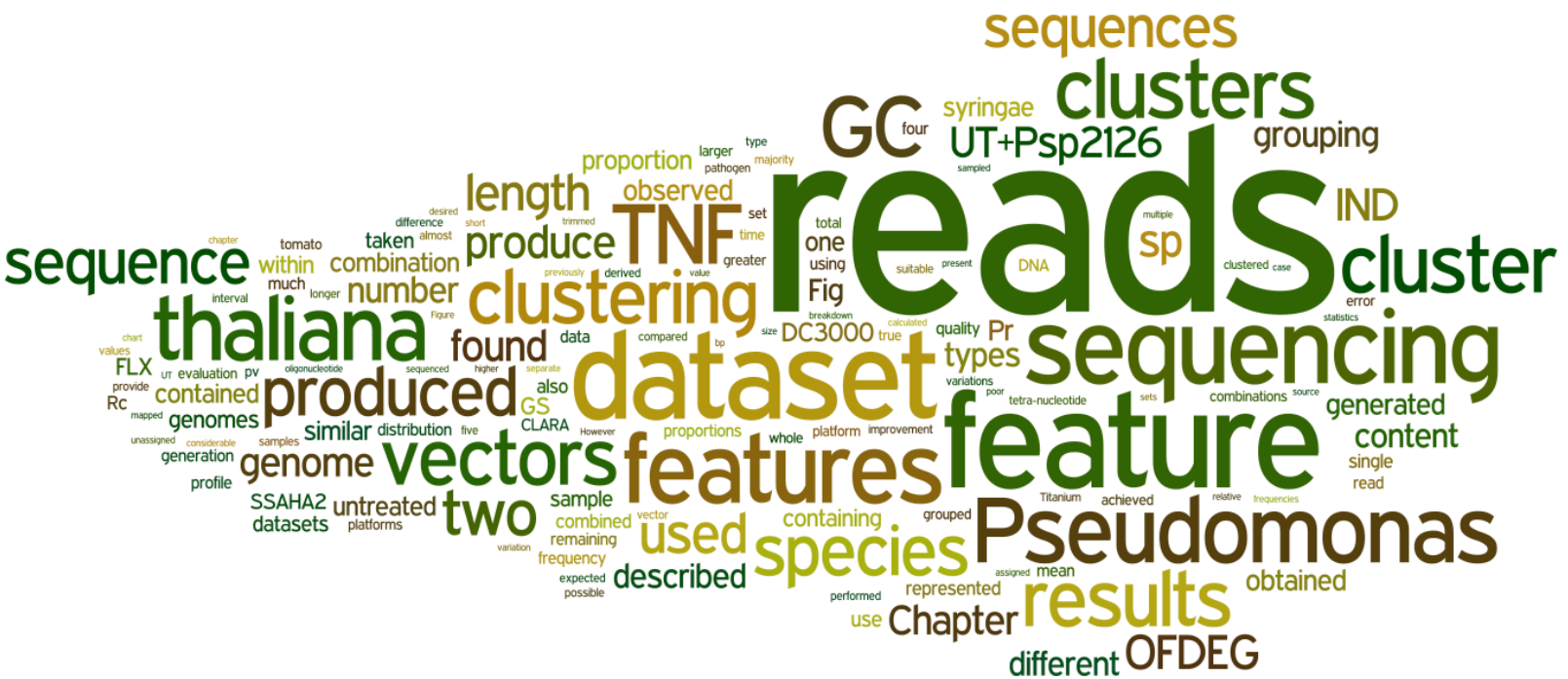
It was predicted that sequencing of the viral treatment group samples on a larger scale would result in the presence of CMV sequencing reads in the data produced from the untreated and dummy inoculated samples, as a product of the low-level infection observed in these samples in qRT-PCR analysis. However, based on the relatively small number of reads assigned to CMV in data from plants that were inoculated with CMV, and the vast predicted discrepancy in the concentration of CMV RNA between this sample and those from untreated and dummy inoculated plants, the additional costs required for such deep sequencing render such further investigation unfeasible.

For a more suitable dataset to be prepared, samples containing higher levels of pathogen would be required. This could be achieved by repeating the process, using a virus of a higher titre, comparing a range of different pathogen options, or by repeating the time course experiment, to determine the optimal point to harvest tissue samples to maximise the relative level of pathogen material

present in the samples. The use of the improved method for RNA extraction in such a time course would make the results more informative.

The results obtained here may indicate that the approach of directly sequencing infected tissue may not be suitable to produce multi-species datasets that can be analysed by clustering analysis, which tends to be problematic when applied to highly disproportionate data. However, personal correspondence has suggested that this may not be the case for every combination of host and pathogen, as other datasets have been produced that are thought to contain reads belonging to the genomes of both host and pathogen species. Some of these datasets were used later in this work (see Chapter 5).

As no suitable dataset was produced here for evaluation of sequence feature vectors and clustering methods, alternative approaches to obtaining such a dataset were considered. The next chapter details the steps taken to develop a dataset containing a mixture of sequencing reads from a host and a bacterial pathogen, by combining reads produced from separate sequencing experiments in more appropriate proportions.


reads
sequences
clusters
GC
four
syringae
UT+Psp2126
grouping
IND
sp
cluster
sequence
length
TNF
observed
proportion
produce
clustering
thaliana
found
dataset
sequencing
feature
produced
vectors
features
Pseudomonas
two
used
species
results
Chapter
OFDEG
different
obtained
described
contained
genome
genomes
similar
SSAHA2
datasets
untreated
platforms
sample
combined
remaining
frequency
containing
grouped
represented
mean
DC3000
Pr
types
quality
content
generated
single
read
Titanium
achieved
combinations
tetra-nucleotide
proportions
platform
improvement
however
CLARA
GS
also
number
combination
within
taken
tomato
much
longer
interval
chapter
difference
desired
short
trimmed
set
time
greater
total
one
using
Fig
suitable
present
DNA
clustered
case
error
statistics
breakdown
calculated
size
bp
compared
data
Figure
higher
separate
sequenced
oligonucleotide
values
chart
FLX
UT
evaluation
pv
provide
Rc
mapped
unassigned
considerable
samples
generation
profile
five
distribution
variation
vector
expected
possible
assigned
performed
use
relative
frequencies
sets
source
poor
variations
previously
derived
value
majority
pathogen
larger
type
sampled
multiple
almost

4

A comparison of genomic signature features applied to the clustering of true sequencing reads by species of origin

Abstract

The capacity of four genomic signature sequence features to group short DNA sequencing reads according to their species of origin was compared once more. In this case, the dataset used as a basis for this comparison was an amalgamation of true sequencing reads produced in separate experiments from untreated Arabidopsis thaliana *and* Pseudomonas *sp. 2_1_26. Consistent with previous analyses, it was found that tetranucleotide frequency distribution vectors provided the most successful separation of reads by species of origin. Using these features it was possible to group together >90% of* Pseudomonas *reads into a single cluster, with these reads accounting for ~60% of the cluster. Other groups were formed containing almost exclusively reads from* A. thaliana*. It was predicted that, applied to reads from more up-to-date sequencing platforms, with improved average read lengths, this approach could produce even more accurate grouping of comparable datasets. Alternative methods of clustering may also allow for improvement in the quality of clustering achievable with use of the same feature vectors.*

Introduction

An evaluation of the power for four different DNA sequence composition features to separate short DNA sequences into groups based on their species of origin was described in Chapter 2. Using two different datasets as a platform, the quality of sequence clustering obtained with the use of each feature type and combination was compared.

When the limitations of these datasets were considered, it was concluded that a more suitable dataset could be produced. The first dataset used was too simple, containing fewer sequences than would be expected in a high-throughput sequencing dataset, from species evolutionarily distant from one another and represented in equal proportion. Conversely, the second dataset (*simLC* (Mavromatis, Ivanova et al. 2007)), was held to be too complex, containing too many species that were too closely related and represented in a wide range of proportions. The mean read length was also longer than is representative of most high-throughput sequencing platforms currently in widespread use.

An ideal dataset for the evaluation of features being performed here would consist of actual sequencing reads, produced from a natural mixture of species whose genomes have been fully sequenced. The subsequent assignment of each sequencing read to one of the contributing genomes (sequencing errors notwithstanding) would allow for quantitative evaluation of the performance of any clustering analysis performed with the dataset. Efforts to prepare such a dataset, from plant tissue infected separately with a bacterial and a viral pathogen, were described in Chapter 3.

Analysis of these sequencing datasets with *SSAHA2*, mapping reads to a reference database of the genomes of contributing species to establish the numbers of reads produced from each, indicated that only a negligible proportion of the reads contained in these datasets originated from the genomes of the pathogens used in preparation of the samples. This was thought to be a result of insufficient presence of pathogen genetic material in the samples, relative to the volume of host plant material.

In order to further evaluate the four feature types compared in Chapter 2 in the absence of a suitable true sequencing dataset, a compromise was found. A dataset was engineered, composed entirely of high-throughput sequencing

reads, as similar as possible to that desired initially. All sequencing reads generated from DNA extracted from untreated *A. thaliana,* as described in Chapter 3, were combined with reads from an experiment sequencing the genome of a bacterial species. The preparation and use of this dataset as a platform for further comparison of the four sequence feature types is described in this chapter.

The full set of 106,294 sequencing reads produced from untreated *A. thaliana* were used, so that the resultant dataset would resemble as closely as possible a true sequencing dataset. For this to be achieved, it was necessary that reads that did not map to the *A. thaliana* genome in the original SSAHA2 analysis were included, as these reads were representative of the 'noise' introduced into sequencing datasets by sequencing error and other species present in the sample from which DNA was extracted. The proportion of the new dataset that was accounted for by reads that were mapped to *P. syringae* DC3000 or remained unassigned by *SSAHA2* are given in Figure 4.1. For a discussion of the origins of these reads that did not map to the *A. thaliana* genome, see Chapter 3.

The *A. thaliana* sequencing reads, generated on the *454 GS FLX Titanium* platform (*Roche*/*454 Life Sciences*) in an experiment aimed at low-coverage sequencing of the genome of *Pseudomonas* sp. 2_1_26, were sampled and trimmed to match the profile of lengths of those reads produced from *A. thaliana* as described in the previous chapter.

The *Titanium* variant of 454 Life Sciences/Roche's GS FLX platform is an upgraded version of the technology that produces more reads, of a greater average length (~10^8 reads per run, at a mean length of ~450-700 bp), than those generated with the standard *GS FLX* technology used in the work described in Chapter 3 (~10^6 reads per run at a mean length ~400 bp).

The combination of these generally longer sequence reads with those derived from *A. thaliana* would produce an unrealistic dataset, where the reads originating from samples from one species have a different length profile than those from the other species. The intention of creating this dataset was that the combined sequence reads could be distinguished based on their nucleotide composition using the sequence features detailed previously. As has been

discussed elsewhere, the length of a sequence is a key factor in the effectiveness of these features. The longer the sequence, the higher the likelihood that the feature vector used will closely resemble the true feature profile of the source genome.

To prevent the introduction of such a bias towards effective clustering of the *Pseudomonas* sp. 2_1_26 (*Pseudomonas*) reads, and in an attempt to maintain consistency throughout the dataset, the randomly selected reads were 'trimmed' to an appropriate length in accordance with the sequence length profile of the reads from *A. thaliana*.

The trimmed *Pseudomonas* reads were added to those from untreated *A. thaliana* in a ratio of ~1:5 to produce a dataset, referred to as *UT+Psp2126*, with the desired proportions of 'pathogen' and 'host' reads.

The UT+Psp2126 dataset was different from the two datasets used previously, as it was derived from two evolutionarily distant species and consisted of actual sequencing reads, albeit artificially mixed, rather than artificially produced sequence fragments (Dataset 1) or Sanger sequencing fragments (*simLC*). The use of sequencing reads (after trimming) ensured an appropriate profile of lengths and the incorporation of sequencing errors during production. The dataset was intended to provide a much closer approximation to the kind that would be expected from sequencing of an infected plant tissue sample.

The quality of reads from high-throughput platforms such as the *GS FLX* is known to deteriorate as the number of nucleotide flow cycles increases during sequencing. As the ends of the *Titanium* reads from *Pseudomonas* sp. 2_1_26 were trimmed to produce a sample with the same length profile as the reads from *A. thaliana*, the bases produced in the later cycles were removed. The result of this trimming was that the overall quality of the trimmed *Pseudomonas* reads was greater than that of the reads generated from the untreated *A. thaliana* sample with which they were combined. It is very likely that this difference in quality between the two sets of reads is manifested as a difference in error profiles, with more base call errors in the reads from *A. thaliana* than in those from *Pseudomonas*. As detailed in Table 6.1 of Chapter 6, the mean Phred30 score for reads from untreated *A. thaliana* was 0.6478 (with a standard deviation of 0.1735) while that of the trimmed *Pseudomonas* sp. 2_1_26 was

0.8237 (0.2116). The Phred30 score of a read is calculated as the proportion of bases called for that read at a Phred quality score equal to or greater than 30 (confidence ≤ [1-1*10^{-3}]). Thisl discrepancy between the error profiles of reads from the two experiments, as well as the use of reads from a bacterial species that does not infect *A. thaliana*, prevented UT+Psp2126 from perfectly representing a single, true sequencing dataset, but the construction of the dataset from true sequencing reads provided a much closer approximation than any other dataset used so far in this work, or any other synthetic dataset available.

The sequenced *Pseudomonas* sp. 2_1_26 bacteria were isolated from the human gastrointestinal tract, and, although relatively closely related to *P. syringae* pv. *tomato* DC3000, this species is not a pathogen of *A. thaliana*. Despite this, these reads were chosen for use as the dataset was available as a 454 sequencing dataset, where most publicly available raw data had been produced on other massively parallel sequencing platforms. This *Pseudomonas* species was as closely related to the *P. syringae* pv. *tomato* DC3000 pathogen used previously as could be found in 454 GS FLX sequencing format. Data from other sequencing platforms could not be combined with the *A. thaliana* 454 GS FLX reads, and still produce a realistic simulation of a true sequencing dataset. Reads produced from *Illumina* and SOLiD platforms are on average much shorter, and with different error profiles to those produced on 454 sequencing platforms.

The genome of *Pseudomonas* sp. 2_1_26 has a relatively high GC content of 66.4% (Ulrich and Zhulin 2010), rendering it considerably more GC-rich than the genome of *A. thaliana*, with a mean GC content of ~36% (The Arabidopsis Genome Initiative 2000). This discrepancy between the two genomes suggested that some success could be achieved with the use of GC content in grouping the reads from each species together, and separating reads from the different species.

As before, an evaluation of the four features introduced in Chapter 1 - GC content (GC), inter-nucleotide distances (IND, Afreixo, Bastos et al. 2009), oligonucleotide frequency-derived error gradients (OFDEG, Saeed and Halgamuge 2009) and tetra-nucleotide frequency distributions (TNF, Karlin and Ladunga 1994; Teeling, Meyerdierks et al. 2004) - was performed, using a

single clustering method, CLARA. The aim of this evaluation was to compare the capability of each feature and combination of features to represent the sequences of within UT+Psp2126, and allow a successful grouping and separation of the data according to their original genomes.

Materials and Methods

Dataset preparation - UT+Psp2126

A dataset was prepared from a set of 454 GS FLX sequencing reads generated from DNA extracted from untreated *Arabidopsis thaliana* plants, as described in Chapter 3. These reads were combined with a second set generated in a separate experiment aimed at sequencing the genome of *Pseudomonas* sp. 2_1_26 as part of the Human Microbiome U54 project (HMP U54 Project, Broad Institute, broadinstitute.org).

A random sample of 19,045 sequencing reads was taken from sample SRR063796 in the NCBI Sequence Read Archive (SRA, http://www.ncbi.nlm.nih.gov/Traces/sra), These reads were chosen at random from the full set of 201,903 reads produced by sequencing of a sample of *Pseudomonas* sp. 2_1_26, isolated from the human gut and sequenced on *454 GS FLX Titanium*.

The frequency of sequence lengths was calculated at 100 bp intervals for the *A. thaliana* sequencing reads used. These frequencies were used to establish a sequence length distribution profile for the dataset as a whole. The 3' ends of the sampled *Pseudomonas* sequences were removed to produce sequence fragments that fitted the proportions of this distribution at each length interval. For each *Pseudomonas* read, a length interval was chosen at random, with the probability of the read falling into each interval equal to the proportion of *A. thaliana* reads that fell in that interval. The sequence was then 'trimmed' to a randomly determined length within the limits of that interval. The minimum read length was limited to 40 bp, consistent with the filtering built into the sequencing platform itself.

Trimmed *Pseudomonas* reads were combined with the untreated *A. thaliana* reads produced in the work described in Chapter 3. The proportions of the resulting dataset are laid out in Figure 4.1. A broad breakdown of the sequence assignments as mapped to the genomes of *A. thaliana* and *P. syringae* pv. *tomato* DC3000 by *SSAHA2*, described in Chapter 3, is included in this figure, showing the proportions of the reads from untreated *A. thaliana* that were mapped to either genome, and those that remained unassigned.

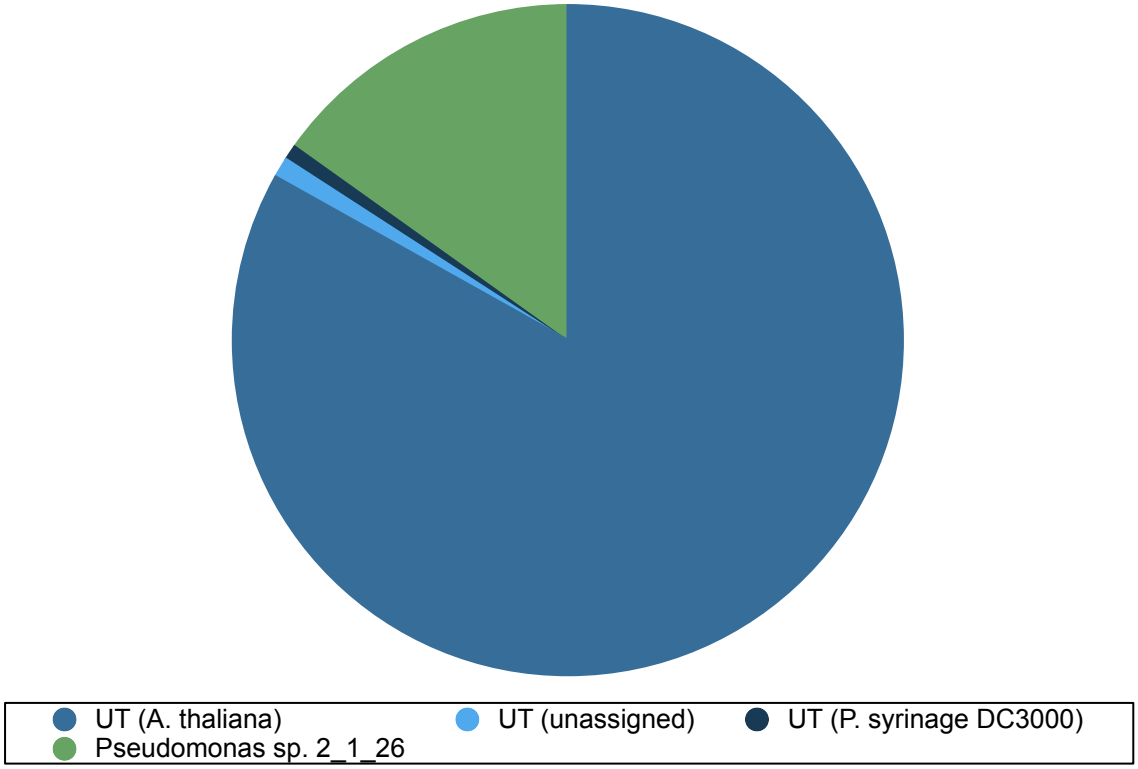


Figure 4.1 *A breakdown of UT+Psp2126, the product of a combination of reads from sequencing of DNA extracted from untreated A. thaliana plants and sequencing of Pseudomonas sp. 2_1_26, describing the proportion of sequences in the dataset that originate from each set of sequencing results. The sequences from untreated A. thaliana are further differentiated according to the genome to which they were assigned by SSAHA2 as described in Chapter 3. Those sequences assigned by SSAHA2 to A. thaliana, to P. syringae pv. tomato DC3000 and those left unassigned are represented in differing shades of blue, while reads originating from Pseudomonas sp. 2_1_26 are represented in green.*

Table 4.1 *The number of sequences in UT+Psp2126 that belong to each species. The numbers of reads that were assigned, by SSAHA2 mapping, to A. thaliana and P. syringae DC3000, and of those left unassigned in are also given.*

Sequence origin (*SSAHA2* assignment)	**Reads in dataset**
UT (*A. thaliana*)	104175
UT (unassigned)	1214
UT (*P. syringae* pv. *tomato* DC3000)	905
***Pseudomonas* sp. 2_1_26**	19045
Total	125339

Generation of feature vectors

GC, IND, OFDEG and TNF feature vectors, and combinations of these four, were generated for each sequence in the UT+Psp2126 dataset as described in the Methods section of Chapter 1.

Clustering - CLARA

Clustering of feature vectors was carried out using CLARA (Kaufman and Rousseeuw 1990), as described in the Methods section of Chapter 1.

The quality of clustering achieved is sometimes described by precision (Pr) and recall (Rc) statistics, also described in Chapter 1. For convenience, a brief summary of these two statistics is reproduced here:

- The predominant class of data within the cluster is determined as the class represented by the largest number of datapoints in the cluster.
- The precision value of the cluster is calculated as the proportion of the total datapoints contained within the cluster that belong to this predominant class.
- The recall value of the cluster is calculated as the proportion of the total datapoints belonging to this predominant class that are contained within the cluster.

Results

The scope for the four feature types, and their combinations, to allow separation and grouping of the reads in the UT+Psp2126 dataset was evaluated by clustering of the dataset into sets of two, and five clusters.

Clustering into two groups was performed to coincide with the number of species known to have been sampled and sequenced to produce the data. The data was also clustered into five groups to provide an indication of the effect of separating reads into a number of clusters larger than the number of principally contributing species. It was thought that *A. thaliana* reads, grouped together with the majority of *Pseudomonas* reads where two clusters were produced, might be further separated from the bacterial reads if the data were divided further, allowing for more successful isolation of the *Pseudomonas* sequences.

Such a separation into multiple groups per species could be successful where regional variations in genomic signature profile exist in the genome, as with isochores of varying GC content. It was hypothesised that subgroups of reads within the *A. thaliana* genome, with a signature feature profile different from other groups of reads sampled from a different region of the genome, might be clustered together with the *Pseudomonas* reads where the data was divided only into two. This effect could be avoided if the dataset was separated into a larger number of clusters.

CLARA analysis of UT+Psp2126 - two clusters

Figure 4.2(i-xv) provides a breakdown of the results of grouping of the UT +Psp2126 dataset, by CLARA, into two clusters with each feature vector type and combination of features.

As with the charts reproduced in Fig. 2.5, each chart in Fig. 4.2 here represents a single cluster in the results. The area of each chart is directly proportional to the number of sequencing reads present in each cluster.

Two clusters were generated from the dataset in accordance with the number of distinct species known to have contributed reads to the dataset. It is possible that reads from other species were also produced due to sample contamination, but any such sequences were assumed to constitute only a negligible proportion of the data.

If perfect clustering were to be achieved, the two clusters produced would each contain all sequencing reads derived from one of these species - one cluster containing 106,294 reads from untreated *Arabidopsis thaliana* and no others, and the other containing 19,045 reads from *Pseudomonas* sp. 2_1_26. Analysed using Pr and Rc statistics, these clusters would return values of 100% for both statistics.

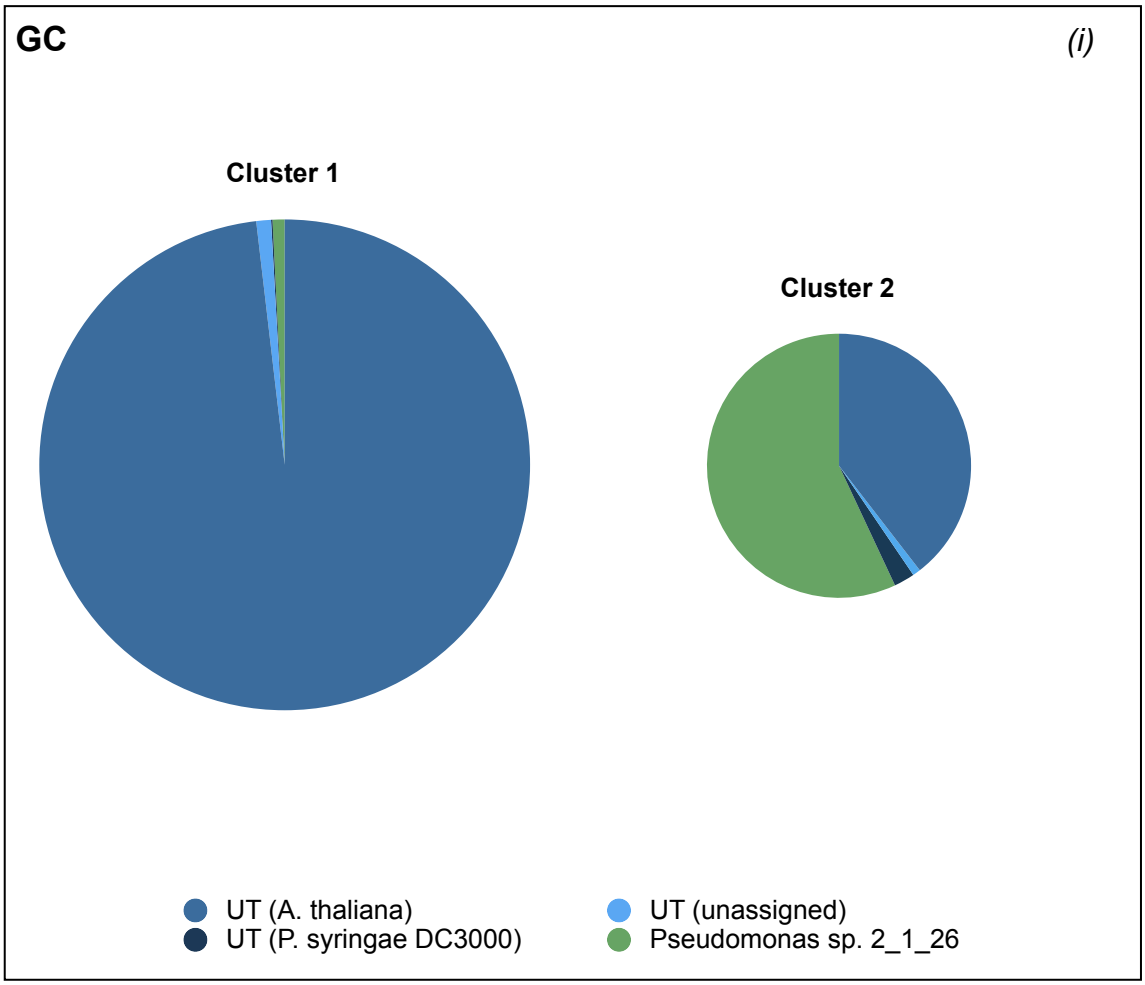
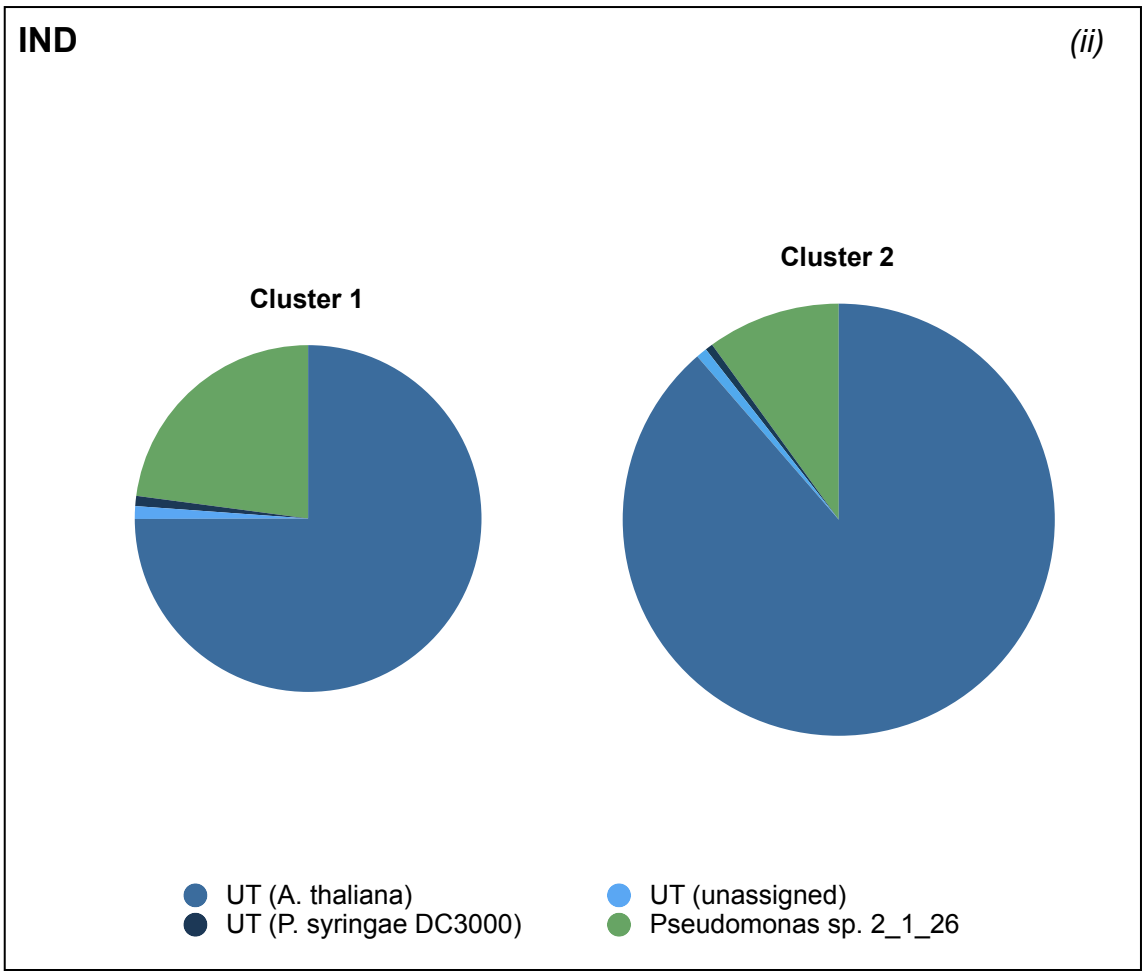
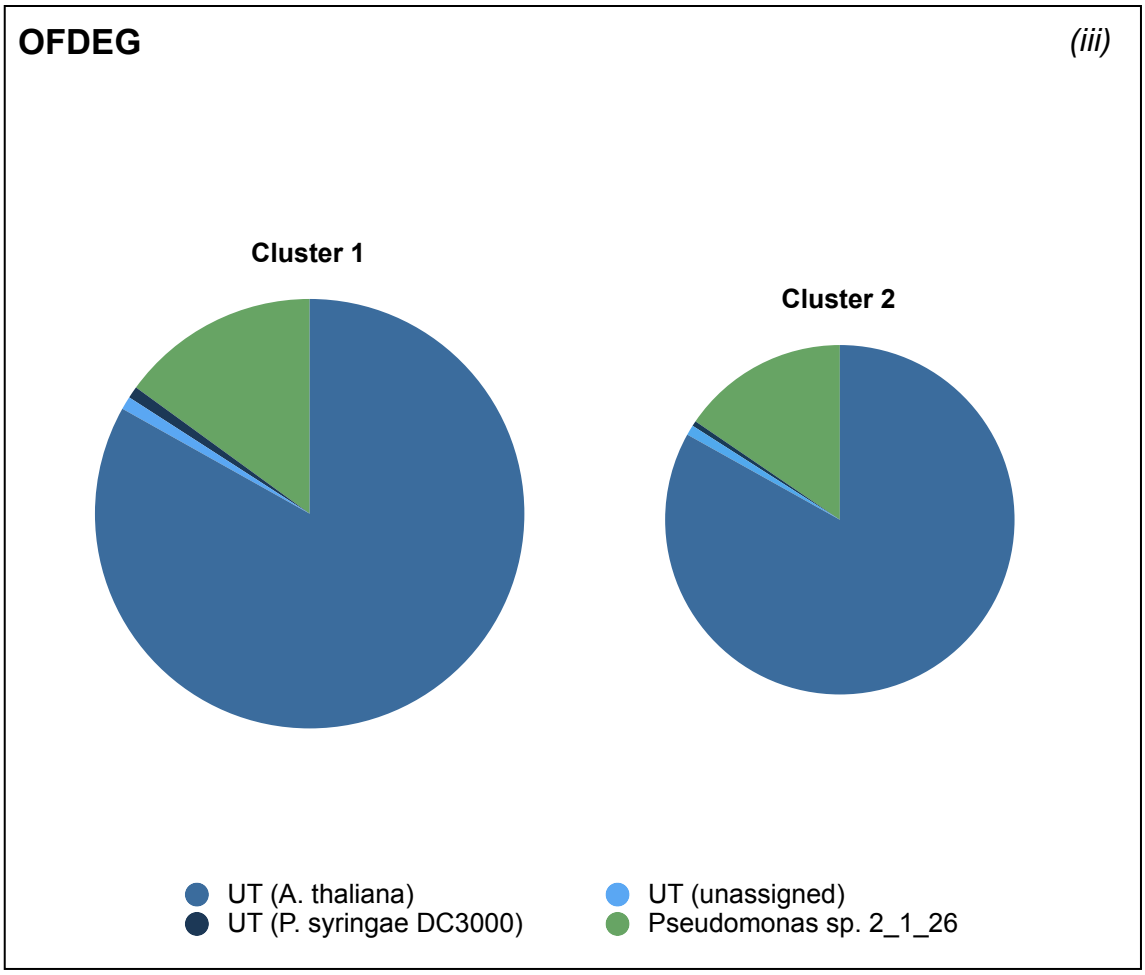
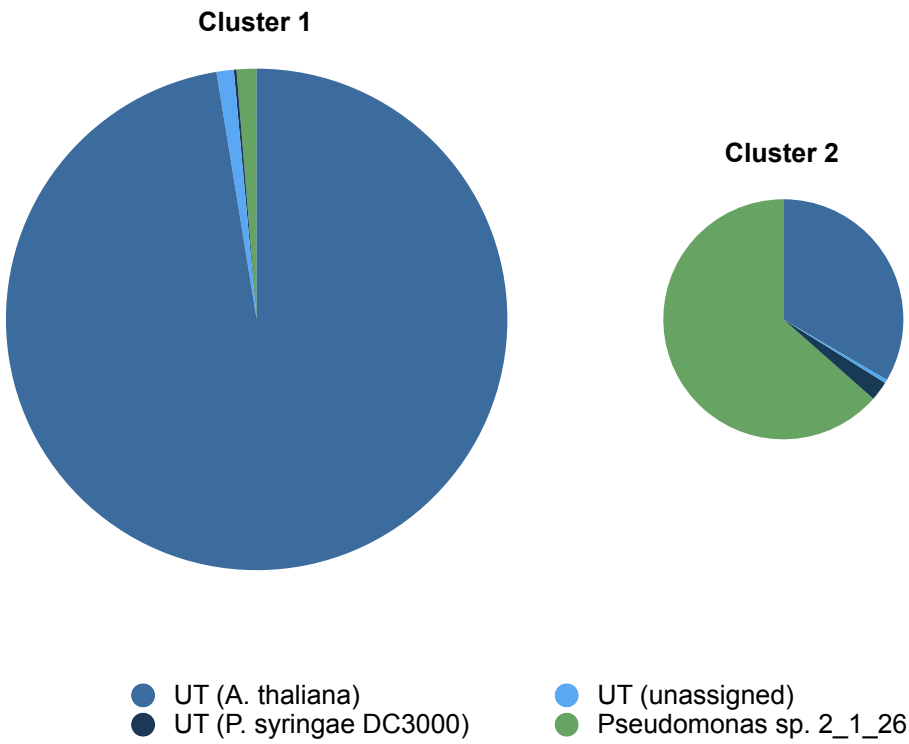


Figure 4.2(i) - 4.2(xv) *Comparative pie charts describing the distribution of sequence reads in UT +Psp2126 dataset between two clusters generated by CLARA analysis with each sequence feature and their combinations. Each set of pie charts corresponds to a feature set. The sections of each chart correspond to the proportion of sequence reads in the cluster that are derived from reads from untreated* A. thaliana *(shades of blue) and* Pseudomonas sp 2_1_26 *(green). Reads from untreated* A. thaliana *are further broken down into subsets according to their assignment to the genome of* A. thaliana *and* P. syringae *pv.* tomato *DC3000 by SSAHA2, with those sequences left unassigned also identified. The pie charts are comparable by size - the area of each chart if directly proportional to the number of sequence reads contained in the cluster that it represents.* ***Fig. 4.2(ii) - 4.2(xv) follow this page.***

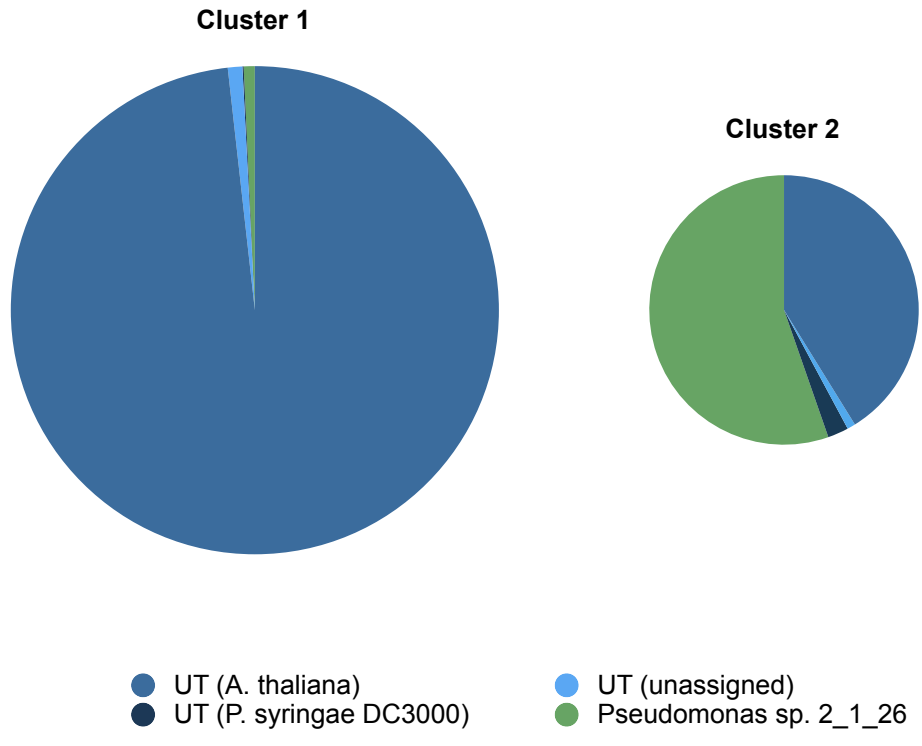

IND
(ii)
Cluster 1
Cluster 2
UT (A. thaliana)
UT (P. syringae DC3000)
UT (unassigned)
Pseudomonas sp. 2_1_26


OFDEG
(iii)
Cluster 1
Cluster 2
UT (A. thaliana)
UT (P. syringae DC3000)
UT (unassigned)
Pseudomonas sp. 2_1_26

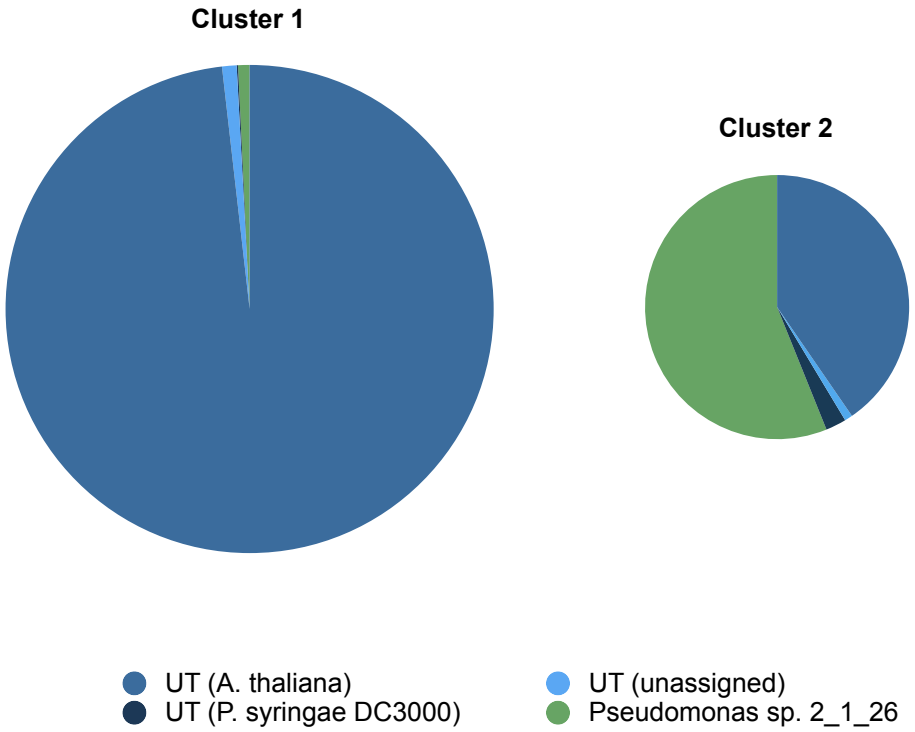
TNF *(iv)*



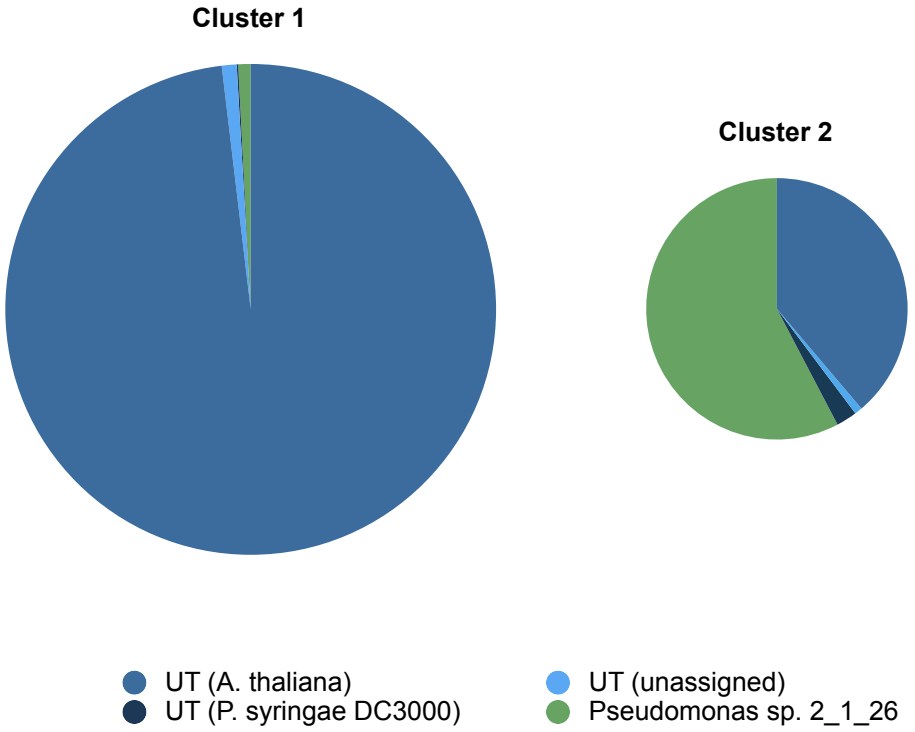
GC + IND *(v)*



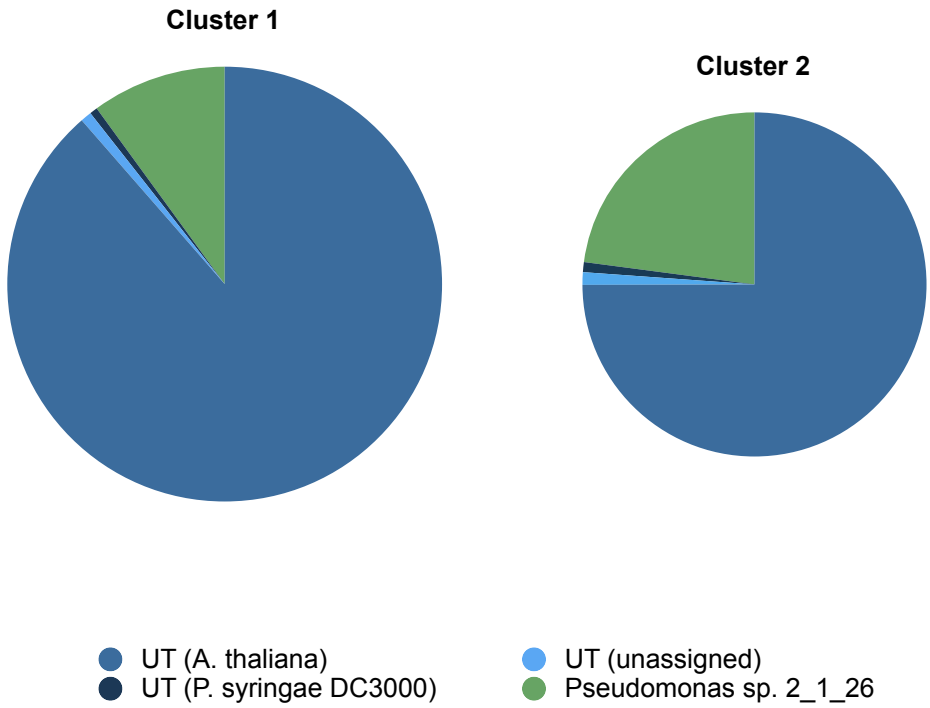
GC + OFDEG *(vi)*



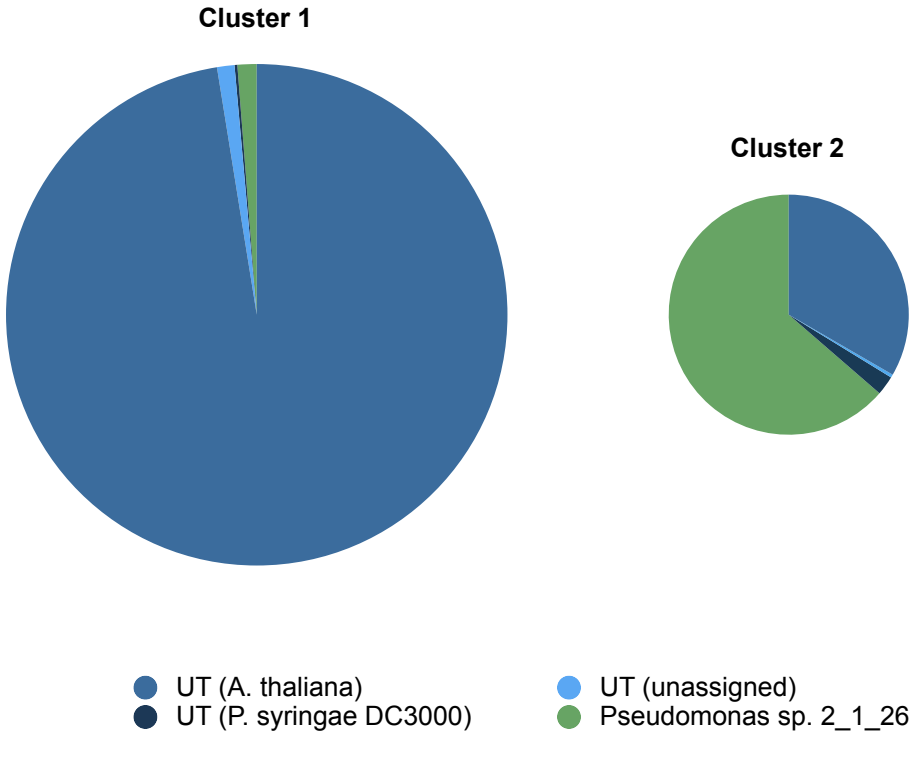
GC + TNF *(vii)*

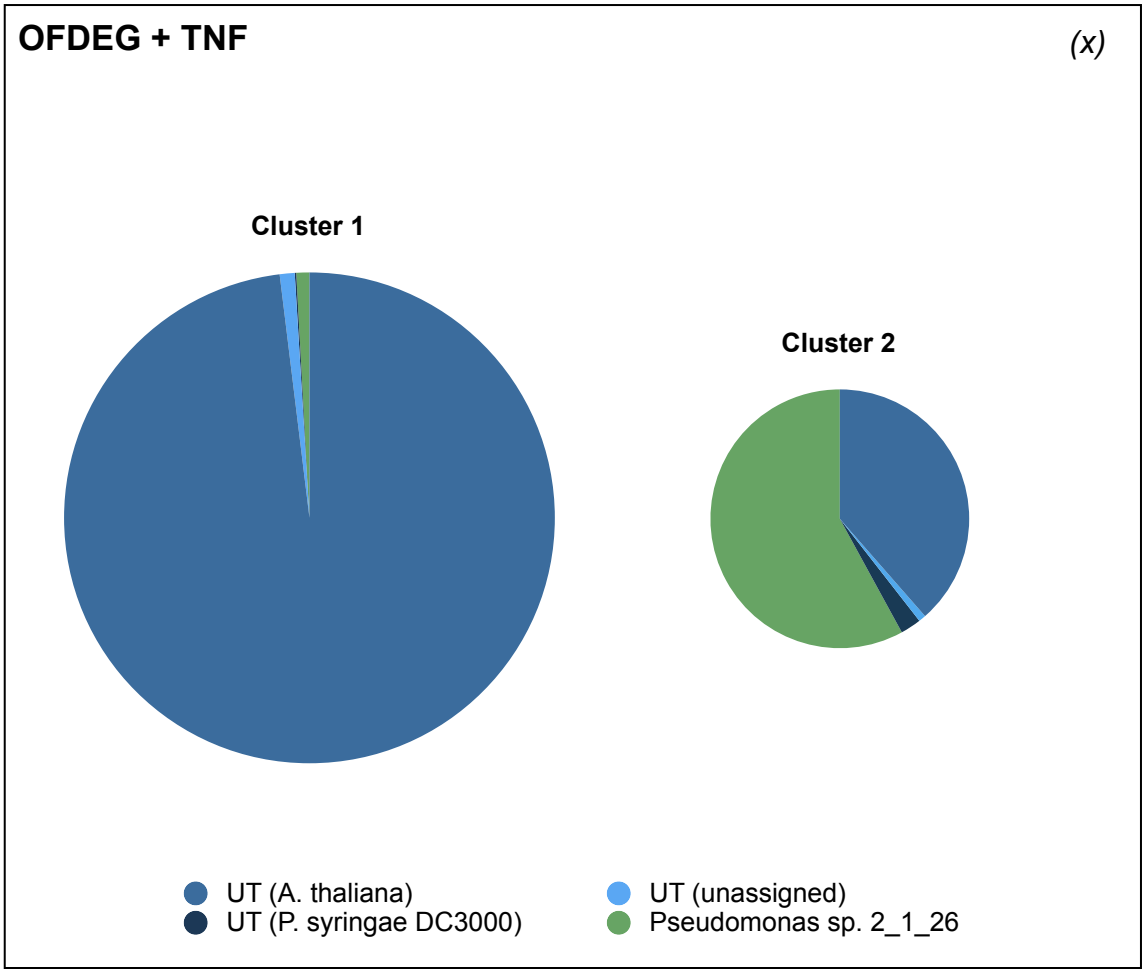


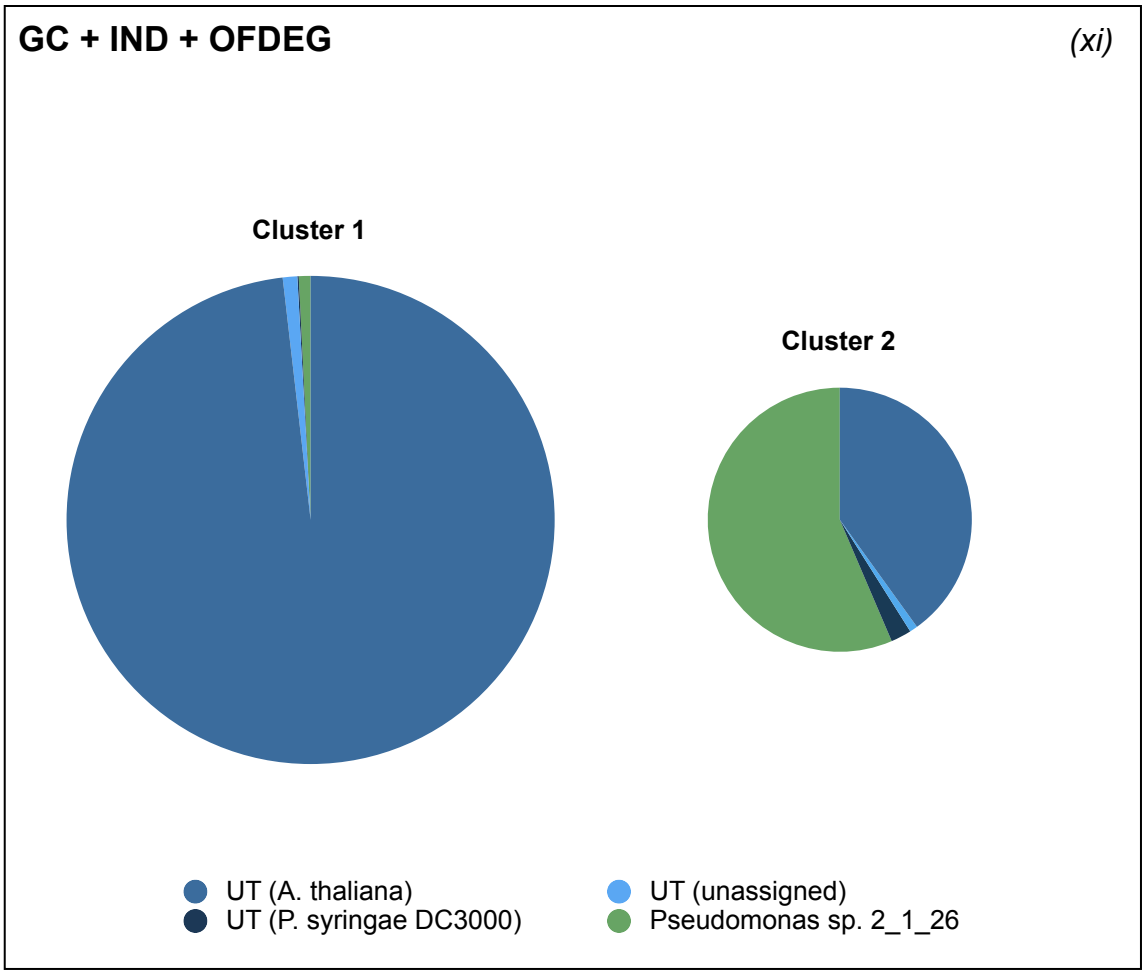
IND + OFDEG *(viii)*



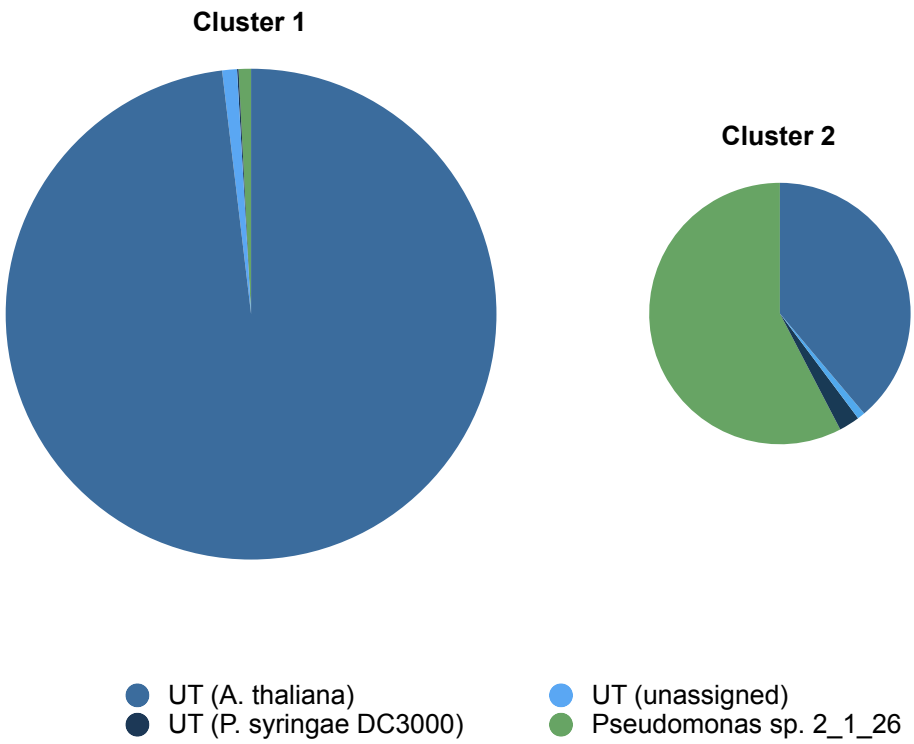
IND + TNF *(ix)*



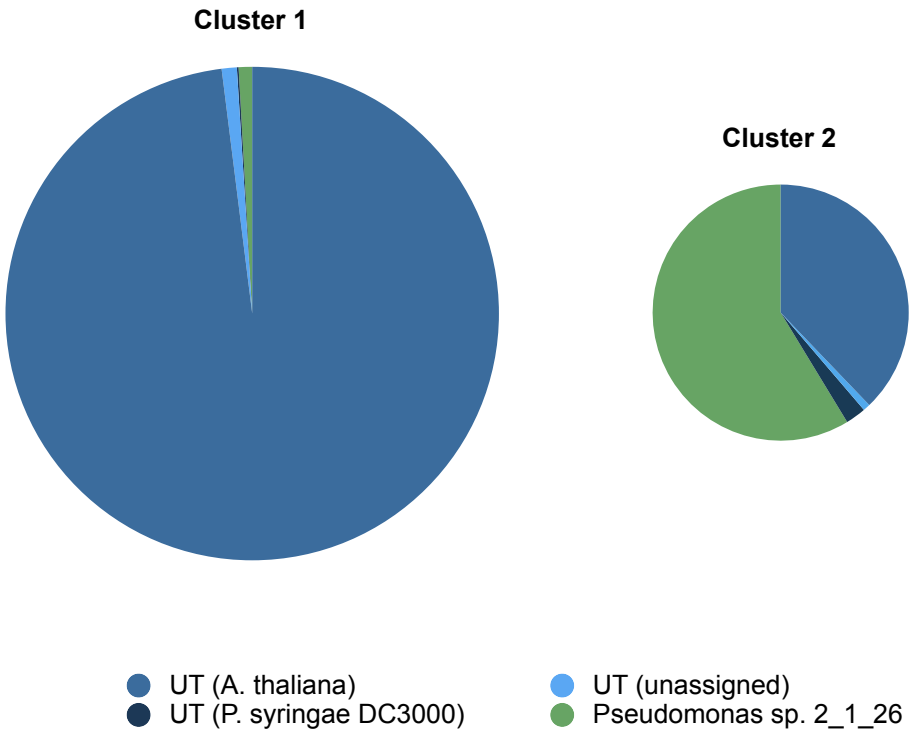

OFDEG + TNF
(x)
Cluster 1
Cluster 2
UT (A. thaliana)
UT (P. syringae DC3000)
UT (unassigned)
Pseudomonas sp. 2_1_26


GC + IND + OFDEG
(xi)
Cluster 1
Cluster 2
UT (A. thaliana)
UT (P. syringae DC3000)
UT (unassigned)
Pseudomonas sp. 2_1_26

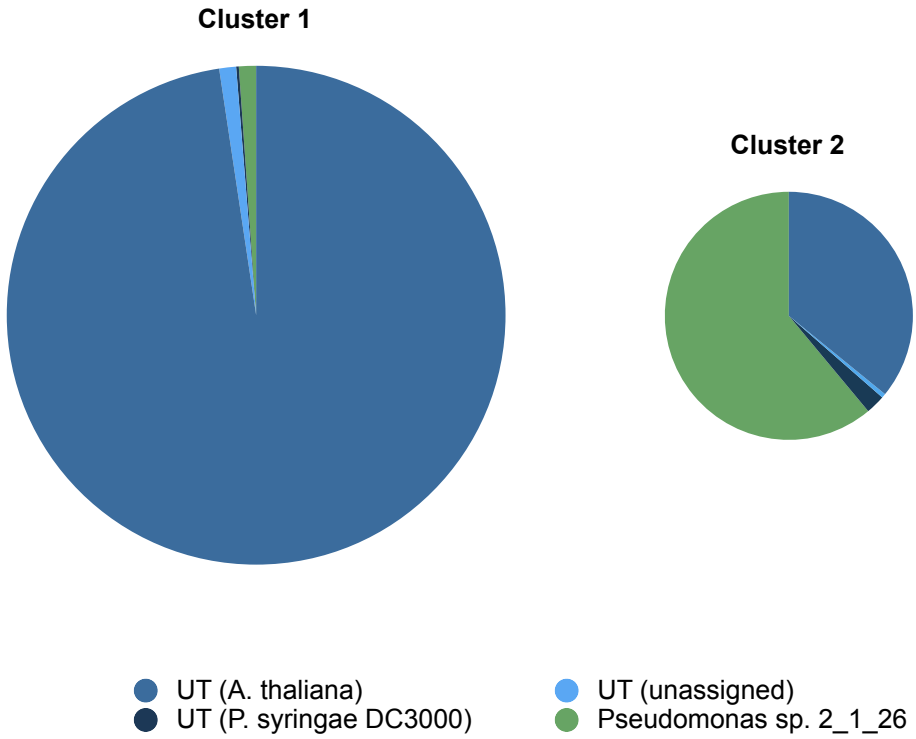
GC + IND + TNF *(xii)*



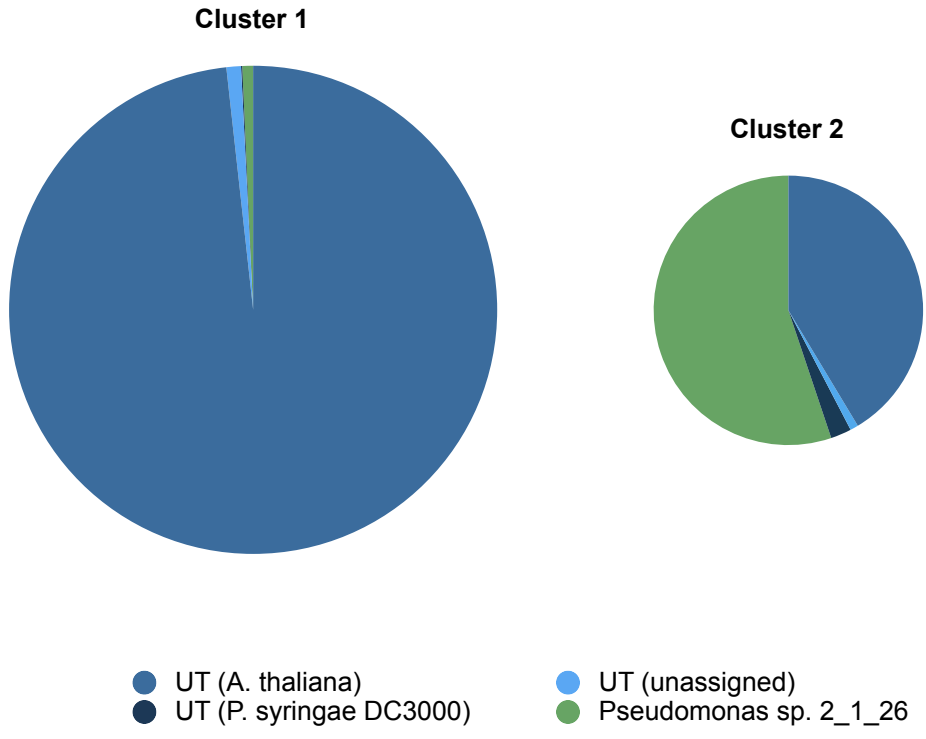
GC + OFDEG + TNF *(xiii)*



IND + OFDEG + TNF *(xiv)*



GC + IND + OFDEG + TNF *(xv)*



Coherent with clustering results obtained when applied to previous datasets, the differentiation between sequencing reads from *A. thaliana* and *Pseudomonas* sp. 2_1_26 was found to be poor in clusters obtained with IND (Fig. 4.2(ii)) and OFDEG (Fig. 4.2(iii)) feature vectors. In the results from both of these feature types, two clusters were produced, of broadly similar size and containing reads from both species in proportions similar to those of the dataset as a whole (*A. thaliana*: ~85%; *Pseudomonas* sp. 2_1_26: ~15%). Equally poor clustering was observed with IND + OFDEG vectors (Fig. 4.2(viii)).

Greater success was achieved with GC (Fig. 4.2(i)) and TNF (Fig. 4.2(iv)) feature vectors. As before, the clustering results obtained with the use of these two feature types displayed similar patterns.

In both sets of results, one of the clusters generated contained a considerably larger proportion of the total sequencing reads than the other. The larger cluster contained a major proportion of reads generated from *A. thaliana*. With GC features, this cluster contained *A. thaliana* reads at Rc 86.99% and Pr 99.21%, while with TNF features the reads were clustered at Rc 90.37% and Pr 98.72%. Conversely, the smaller cluster contained the vast majority of reads from *Pseudomonas* sp. 2_1_26 in the dataset. With GC features, these reads were clustered at Rc 96.12% and Pr 43.03%, and with TNF at Rc 93.44% and Pr 36.51%. As such, the larger cluster consisted almost entirely of reads from *A. thaliana*, while the smaller cluster, though containing the vast majority of *Pseudomonas* reads in the dataset, still mostly consisted of *A. thaliana* reads.

Excluding IND + OFDEG, discussed previously, the feature combinations were found to produce clusters similar to those produced with GC and TNF single-feature vectors. In the results from all combined feature vectors, the same pattern was observed in clustering. No combination of features was found to provide an obvious improvement on the grouping achieved with GC or TNF features used on their own.

Excluding the results from IND, OFDEG and IND + OFDEG vectors, where no notable discrimination between reads based on species could be identified, it was observed that reads generated from untreated *A. thaliana* samples, that had been assigned by SSAHA2 to the genome of *Pseudomonas syringae* pv. tomato DC3000 (labelled ‘UT(*P. syringae* DC3000)’ in Fig 4.2) tended to be

grouped into a cluster with the majority of the *Pseudomonas* sp. 2_1_26.

As the *A. thaliana* samples from which the dataset was prepared were not found to contain *P. syringae* DC3000 material when analysed by qPCR assay, the 905 reads mapped to this genome were thought to be the result of mis-sequencing of *A. thaliana* DNA (see Chapter 3).

Nevertheless, it was not surprising to find these sequences clustered with *Pseudomonas* reads here. SSAHA2 maps and assigns a sequence to a reference genome based on an alignment of the sequences, suggesting that these reads held significant sequence homology to a region of the *P. syringae* DC3000 genome. Assuming a considerable level of homology between the genomes of the two *Pseudomonas* species, as would be expected in closely related species, these homologous *A. thaliana* reads could be expected to produce features be grouped with true bacterial reads, regardless of their source.

UT+Psp2126 - five clusters

Figure 4.3(i-xv) provides a breakdown of the results obtained from grouping of UT+Psp2126 by CLARA into five clusters, with each feature vector type and combination of features. The area of each chart is directly proportional to the number of sequencing reads present in each cluster, excepting those marked with an asterisk, where the number of sequences contained in the cluster was too small to be proportionally represented in a chart.

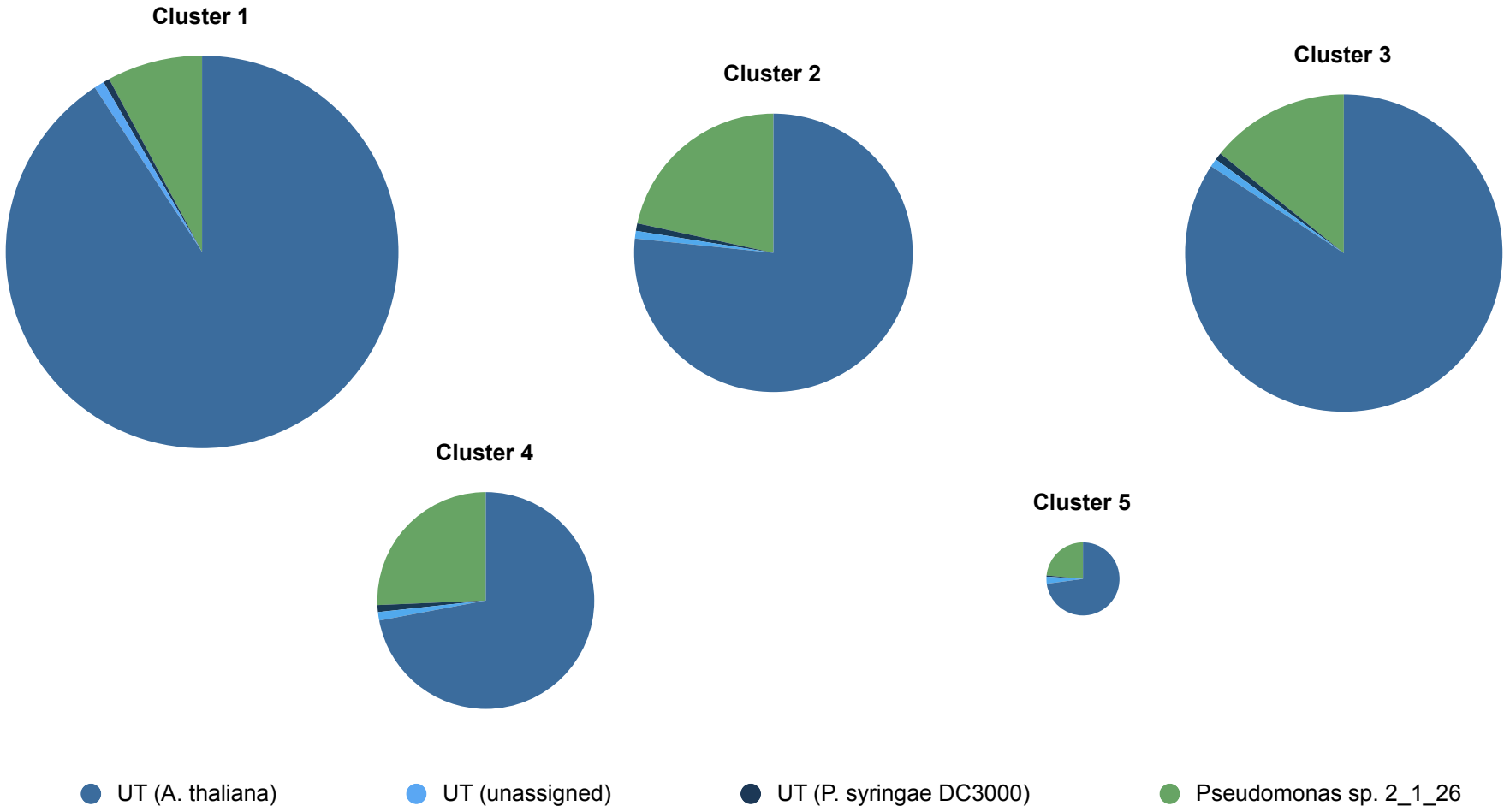
The dataset was grouped into five clusters in response to the consistent grouping of the vast majority of *Pseudomonas* sp. 2_1_26 reads into a cluster also containing reads from *A. thaliana* (in a ratio of ~3:2). It was thought that these bacterial reads might be further separated from those originating from the plant genome if the dataset were grouped into a number of clusters greater than the number of species known to have been sequenced in production of the dataset.

For example, if the genome of *A. thaliana* contains regional variations in profile of the feature types used to compare sequences, the reads derived from these different regions might be grouped separately during clustering. This could result in co-clustering of some *A. thaliana* reads with those obtained from *Pseudomonas*, as was observed previously. Clustering into additional groups might allow for these subsets within the *A. thaliana* reads to be isolated from the *Pseudomonas* reads, increasing the precision of their grouping without adversely affecting recall within the bacterial data.

Figure 4.3(i) - 4.3(xv) *Comparative pie charts describing the distribution of sequence reads in UT/P2126 dataset between five clusters generated by CLARA analysis with each sequence feature and their combinations. Each set of pie charts corresponds to a feature set. The differently coloured sections of each chart correspond to the proportion of sequence reads in the cluster that are derived from reads from untreated* A. thaliana *and* Pseudomonas sp 2_126*. The pie charts are comparable by size - the area of each chart if directly proportional to the number of sequence reads contained in the cluster that it represents. For ease of visual interpretation, there are* ***several exceptions*** *to this,* ***in Figure 4.3 (iv), 4.3(vii), 4.3(ix), 4.3(x), 4.3(xiv) and 4.3(xv)*** *(marked with an asterisk), where the total number of sequencing reads contained in a cluster was too small to be represented proportionally in a pie chart on the same scale as the other clusters produced.*

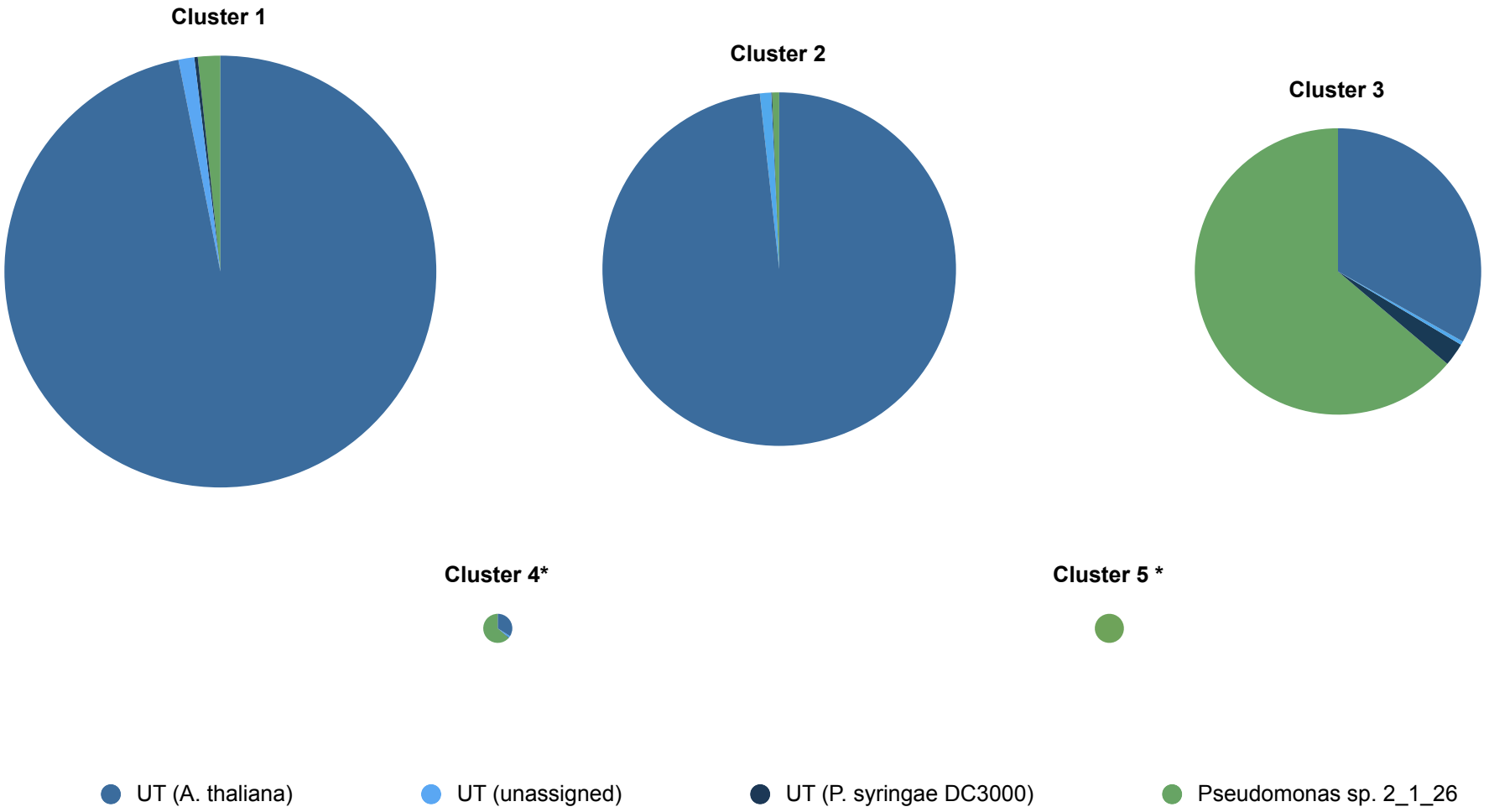
IND + OFDEG

(viii)



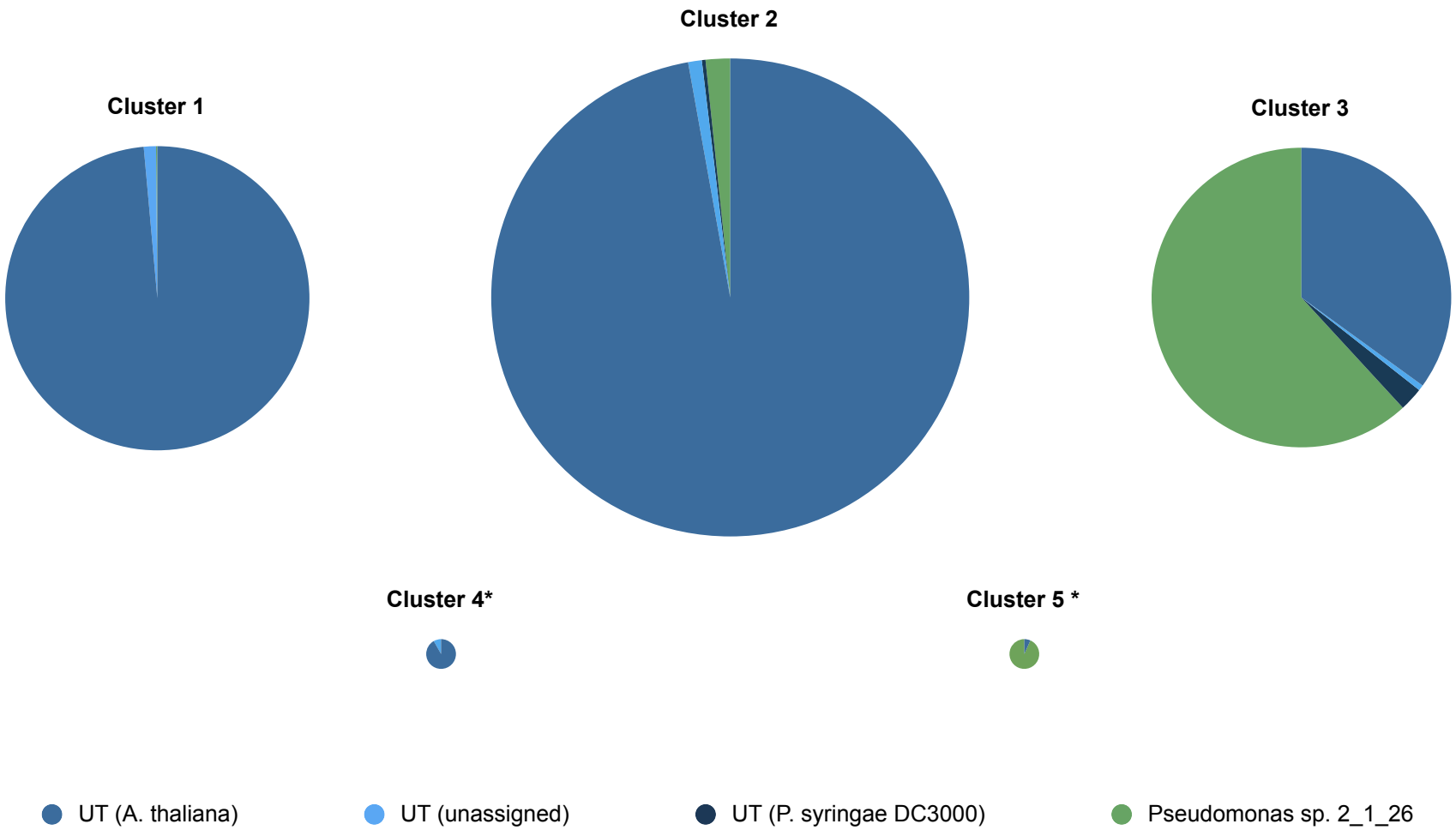
IND + TNF

(ix)


Cluster 1
Cluster 2
Cluster 3
Cluster 4*
Cluster 5 *
UT (A. thaliana)
UT (unassigned)
UT (P. syringae DC3000)
Pseudomonas sp. 2_1_26

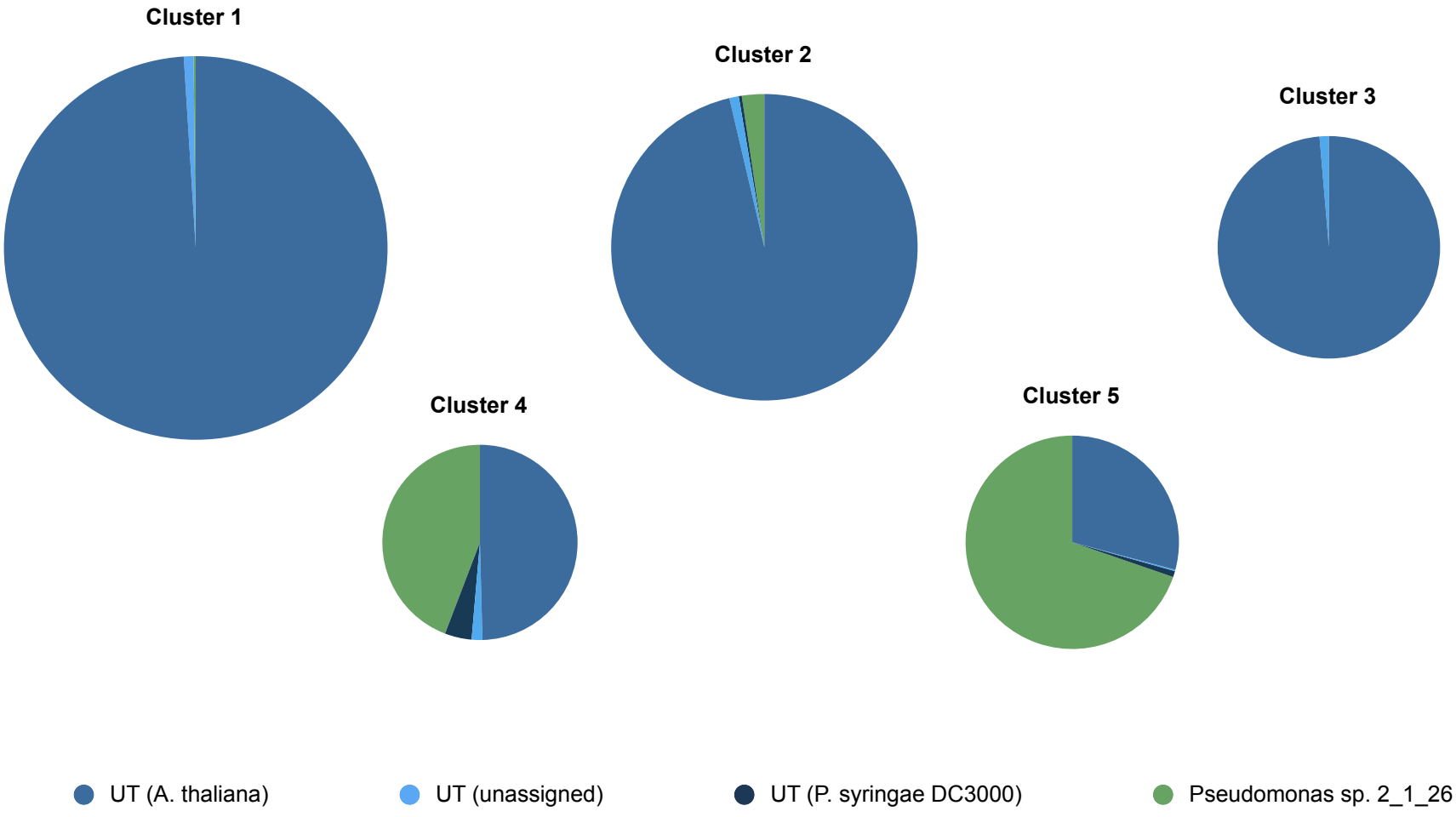
OFDEG + TNF

(x)


Cluster 1
Cluster 2
Cluster 3
Cluster 4*
Cluster 5 *
UT (A. thaliana)
UT (unassigned)
UT (P. syringae DC3000)
Pseudomonas sp. 2_1_26

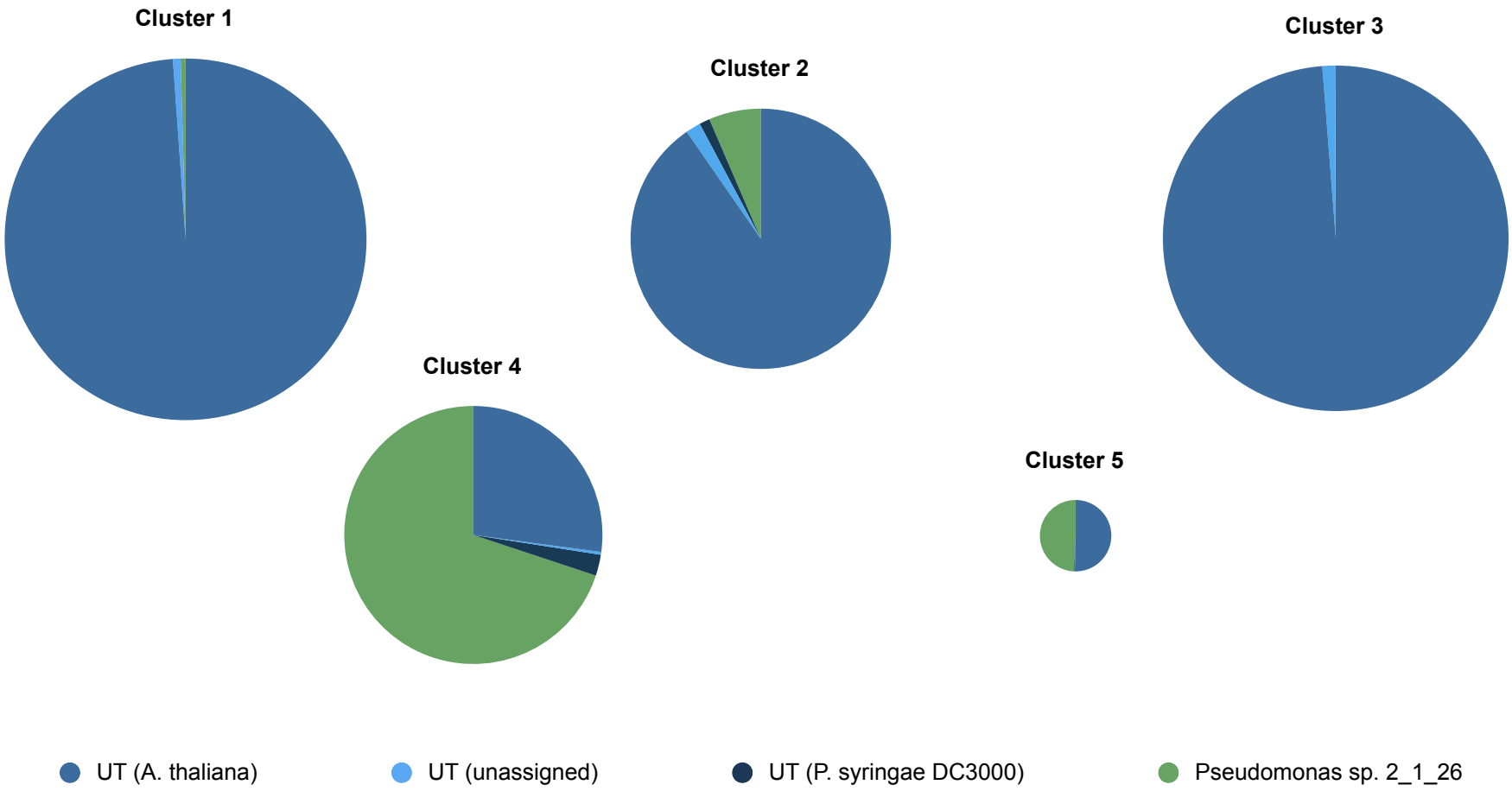
GC + IND + OFDEG

(xi)



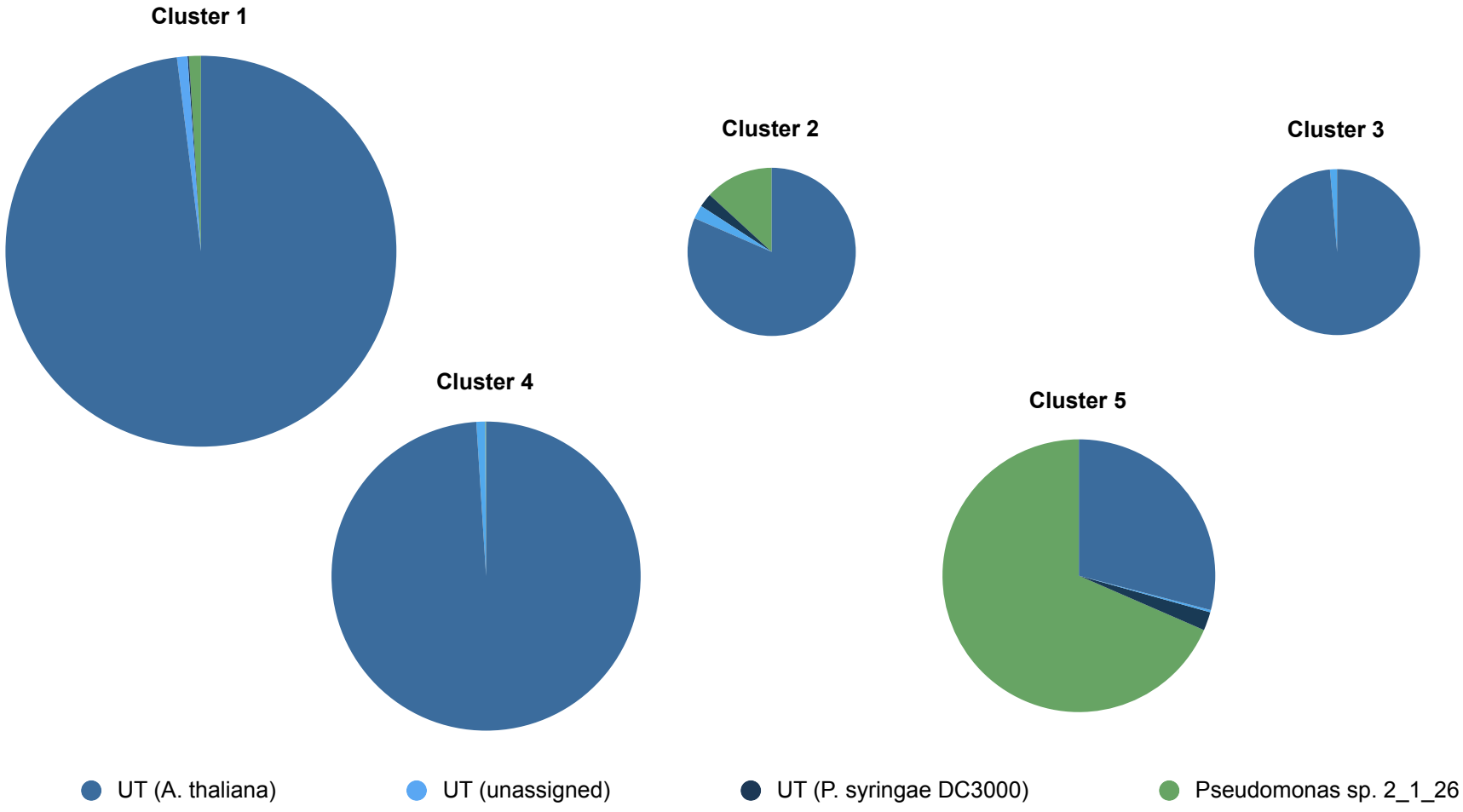
GC + IND + TNF

(xii)



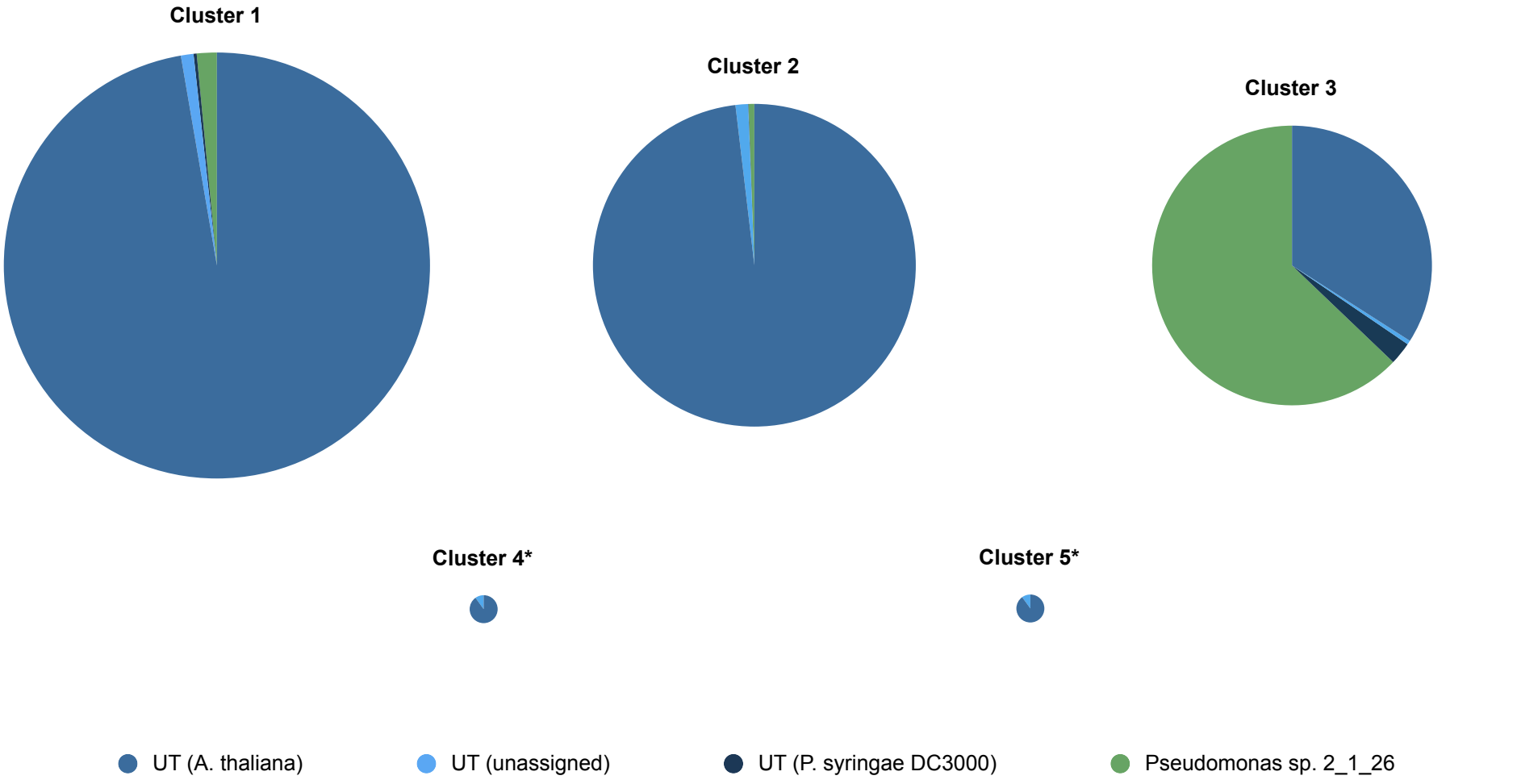
GC + OFDEG + TNF

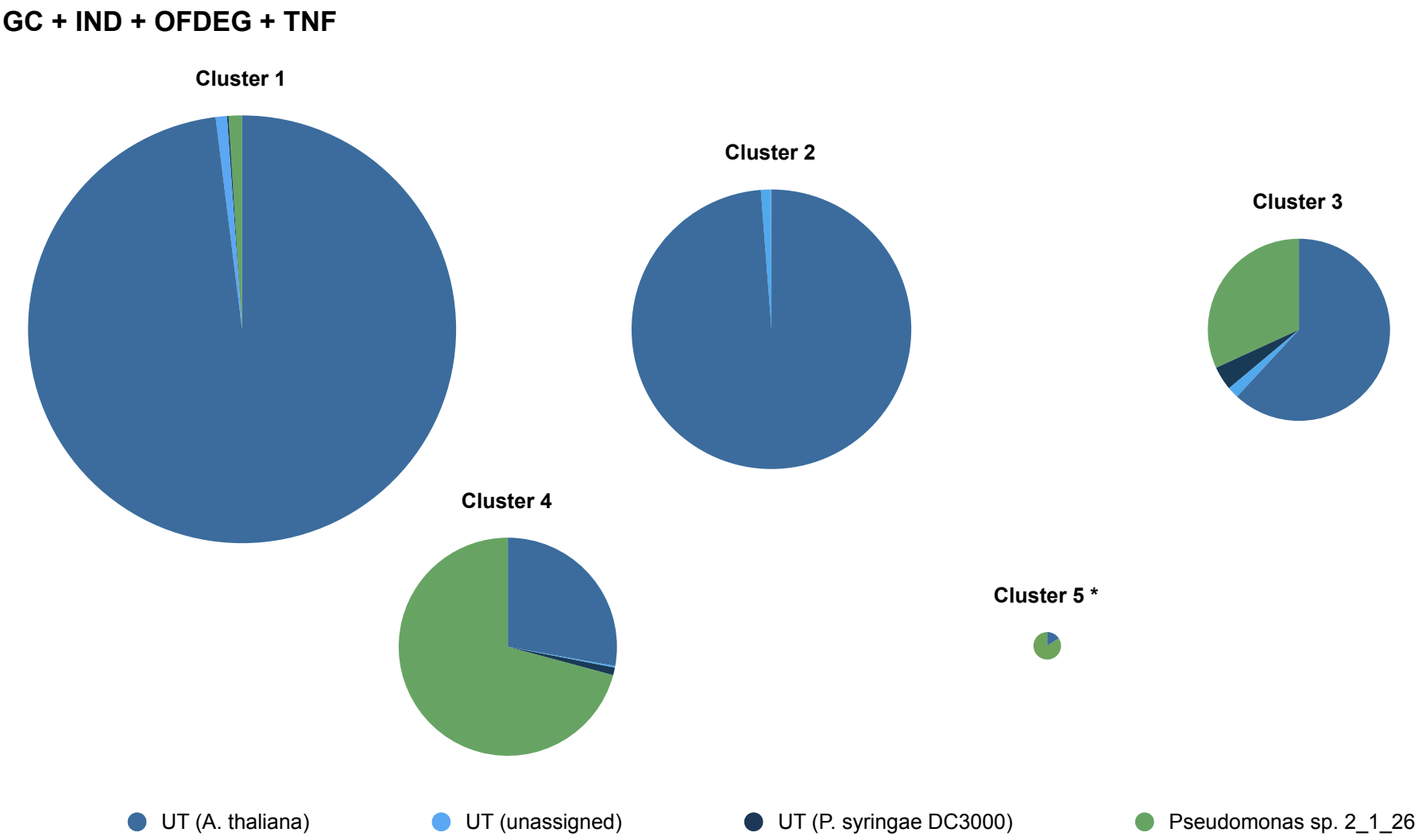
(xiii)



IND + OFDEG + TNF

(xiv)




GC + IND + OFDEG + TNF
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5 *
UT (A. thaliana)
UT (unassigned)
UT (P. syringae DC3000)
Pseudomonas sp. 2_1_26

Once again, clustering results produced with IND, OFDEG and IND + OFDEG feature vectors displayed poor separation between sequencing reads from the two species (Fig. 4.3(ii), 4.3(iii) and 4.3(viii)). As in the results from grouping into two clusters, the distribution of sequences within clusters was found to mirror that of the dataset as a whole.

Similarity with the results obtained from grouping UT+Psp2126 into two clusters was also found in the results from GC and TNF features. As before, clustering with these features was found to be relatively effective. However, a more discernible difference was observed here, between the quality of clustering achieved with GC and TNF features.

In both sets of results, one cluster was found to contain a large fraction of the total *Pseudomonas* sp. 2_1_26 reads in the dataset. With GC features this cluster grouped *Pseudomonas* reads at Rc 87.85% and Pr 68.84%, and with TNF features at Rc 93.43% and Pr 63.49%. In both cases, most of the remaining *Pseudomonas* reads were grouped into a second cluster, leaving three other clusters populated almost exclusively with *A. thaliana* reads.

In the results from TNF features, the cluster containing most of the remaining *Pseudomonas* reads, was much larger than the other clusters produced, accounting for over half of the total reads in the dataset. Of the reads in this cluster only 1.85% originated from *Pseudomonas* 2_1_26.

Of the remaining three clusters produced with TNF feature vectors, one contained only 3 reads in total, and could be effectively disregarded. The other two clusters were composed almost entirely of *A. thaliana* reads (Pr values of 99.94% and 99.91%).

In the results from GC features, the cluster containing most of the remaining *Pseudomonas* reads was much smaller than its equivalent in the TNF results. Of the reads in this cluster, 13.07% were derived from *Pseudomonas*, a greater proportion of the cluster than in the results obtained with TNF features. The distribution of reads from different species within this cluster resembled that of the dataset as a whole, indicating that this cluster was not markedly enriched with reads from either of the represented species. Although this cluster contained *A. thaliana* reads at a lower Pr than in the equivalent from TNF features, the actual number of *Pseudomonas* reads contained in the two

clusters was almost identical: the greater Pr statistic for the TNF-derived cluster was a product of the much greater size of the cluster overall.

The remaining three clusters produced with GC feature vectors were composed predominantly of reads from untreated *A. thaliana*, with Pr values of 99.74%, 98.33% and 99.98% respectively.

In coherence with the clustering results observed with a different number of clusters and different datasets, GC and TNF feature vectors were clearly observed here to provide a considerable level of enrichment for reads derived from the same species in some or all of the clusters produced.

In this case, the separation and grouping obtained with TNF feature vectors was marginally more accurate than with GC, but the difference between the two sets of results was not great. However, grouping with TNF feature vectors resulted in >93% of all *Pseudomonas* reads in UT+Psp2126 being clustered together and constituting almost two thirds of the reads in that cluster - a considerable enrichment in a single cluster.

Of the vectors of combined feature types, clusters produced with each of IND + TNF (Fig. 4.3(ix)), OFDEG + TNF (Fig. 4.3(x)) and IND + OFDEG + TNF (Fig. 4.3(xiv)) vectors were observed to be of comparable quality and hold similar properties to those produced with TNF features alone. The clusters produced with these three feature combination vectors varied slightly in size and composition, but displayed largely the same pattern in the distribution of *Pseudomonas* reads within them. None of these combinations of features yielded results that constituted an improvement in terms of levels of enrichment in clusters, relative to that of TNF features used on their own.

The remaining combinations produced results similar to those obtained with the use of GC feature vectors.

As was observed in the grouping of UT+Psp2126 into two clusters, reads that were generated from untreated *A. thaliana,* but that were mapped by SSAHA2 to the genome of *P. syringae* DC3000, tended to be grouped into clusters containing a higher proportion of *Psedomonas* sp. 2_1_26 reads.

Discussion

Several trends were identified in the clustering performance obtained with the different feature types and combinations throughout all of these comparisons. Firstly, representation of sequences with GC and TNF features was found to provide much more successful clustering than with IND and OFDEG vectors.

Throughout the range of datasets used here (both in this chapter, and in Chapter 1), with variations in the sequence length, the number of different species represented, the relatedness of those species and the proportions of representation within the dataset, tetra-nucleotide frequency feature vectors were found to consistently produce some of the most successful clustering results. In cases where TNF vectors did not produce the best clusters, then the quality of TNF-derived clusters was found to be only marginally lower than that of another feature combination.

GC content was also found to produce good clustering in the cases considered here. Often the quality of the clusters produced was comparable to those from TNF features, but was never found to excel this standard.

As discussed in Chapter 1, the single-variable nature of the GC content feature and the regional variation of GC content in isochores of the genomes of eukaryotic organisms makes it easy to foresee circumstances in which GC content would not be a suitable mechanism by which to separate reads. For example, where two or more sampled organisms from which the sequencing dataset was taken share a similar genomic GC content or, in the case of higher organisms, considerable overlap exists in the GC content of some isochores within one or more of their genomes. It is conceivable that this effect was observed in the results of clustering the UT+Psp2126 dataset with GC features, where a number of *A. thaliana*-derived reads were clustered together with the majority of *Pseudomonas* sp. 2_1_26 reads.

Similar limitations may exist for tetra-nucleotide frequencies, and the other feature types used here. Intra-genomic variations in GC content are inevitably linked to similar variations in oligonucleotide content, which will be manifested as regional variations in the oligonucleotide distribution of the genome. However, the multiple frequencies that contribute to each vector in the case of TNF features and other oligonucleotide frequency features may provide some

protection against local variability in the frequency of a single nucleotide.

With these limitations in mind, it is important to consider again the impact of the short length of the reads present in the dataset. This short length means that the source of variation in GC content of the sequences was likely to be natural short-range variation, in addition to the mid- to long-range variations observed in the mosaic structure of the *A. thaliana* genome.

This level of variation is an unavoidable obstacle associated with the short length of the reads produced by current sequencing platforms, and is the single greatest limiting factor in the kind of clustering experiments that have been performed here, for every type of sequence feature: a read can be considered as a sample taken from the whole genome, and the longer the read, the larger the sample and the higher the likelihood that feature vectors produced from this sample will accurately represent the genome as a whole. As the technology improves, producing more reads per run at a greater mean length (e.g. the 454 GS FLX Titanium platform used to produce the *Pseudomonas* sp. 2_1_26 reads used here), this limitation will begin to be addressed.

The similarity in the clustering results obtained with GC and TNF features throughout these experiments, and the lack of any significant improvement in clustering quality when vectors were used that combined these two features, indicated a redundancy between the two types of feature. The information provided by measuring the GC content of a sequence is captured in the tetra-nucleotide relative frequency distribution of that sequence, in the form of the frequencies of all G- and C-containing tetra-nucleotides.

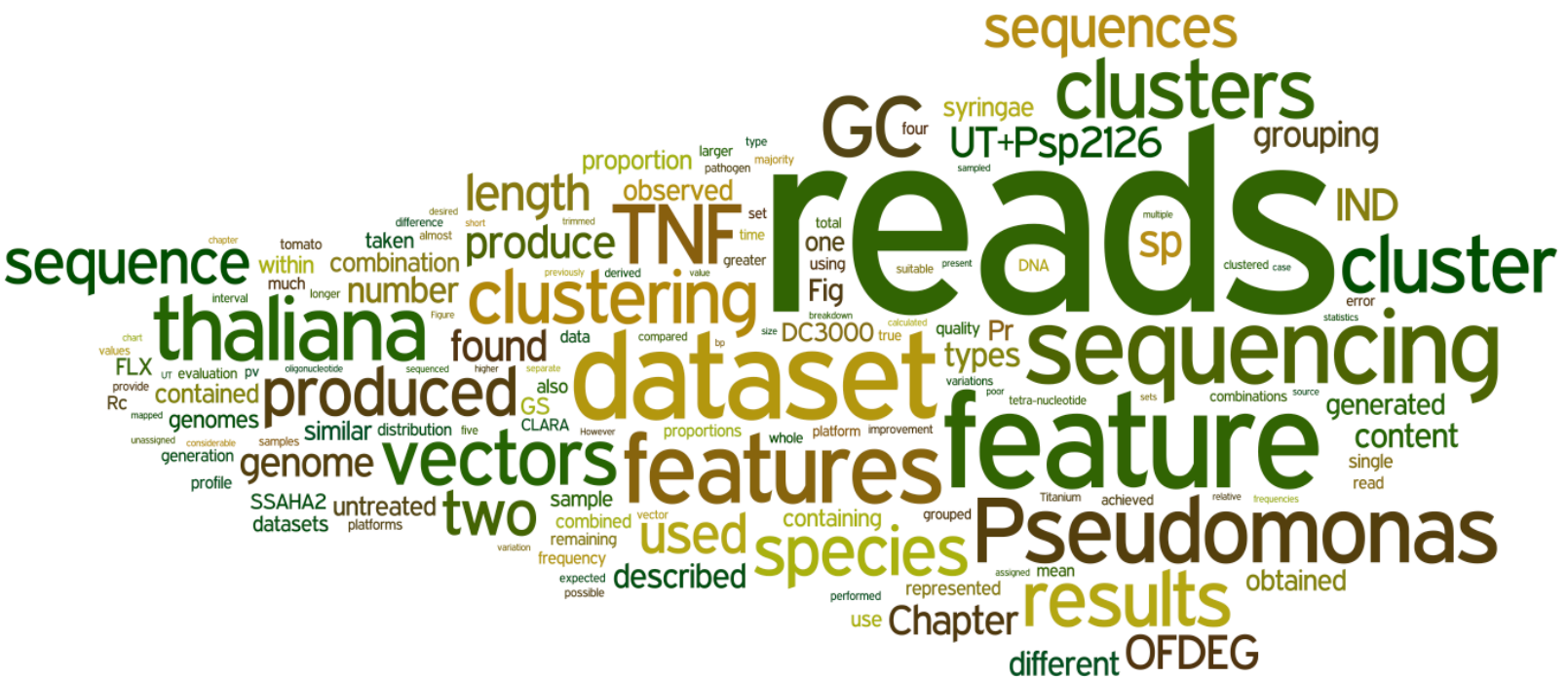
The remaining feature types, IND and OFDEG, were consistently found to produce poor clustering results from the different datasets used. It was concluded that these features, used on their own or in combination, were not suitable for clustering of sequencing reads by species of origin as desired. When used in conjunction with one of the other two feature types, a marginal improvement in clustering results was sometimes observed. For an example, see the IND+TNF and IND+OFDEG five-cluster results with UT+Psp2126 (Fig. 4.3(ix) and 4.3(x)). However, such an improvement was not consistently observed over multiple datasets/number of clusters.

Without the parallelisation of OFDEG feature generation, the time taken to

produce large numbers of these feature vectors appeared prohibitive (see results detailed in Chapter 2), especially when compared to that taken to produce the other feature types, and the relative performance of these feature types.

With these considerations made, it was concluded that vectors of TNF features on their own were the most suitable form of sequence representation for use in clustering where the desired outcome is the grouping of sequences from the same taxonomic source.

In this evaluation, features were compared using a single clustering method, *CLARA*, so that any difference in the results obtained could be fully attributed to the difference in feature vectors used to represent the sequences in the dataset. Many more clustering methods exist that could be used to separate the data, and a range of these clustering methods will be compared, using TNF feature vectors, in the next chapter, with the aim of finding an optimal combination of feature vector and clustering method for grouping of sequencing reads.


reads
sequences
clusters
GC
four
syringae
UT+Psp2126
grouping
IND
sp
cluster
sequence
length
TNF
observed
proportion
produce
clustering
thaliana
found
dataset
sequencing
feature
produced
vectors
features
Pseudomonas
two
used
species
results
Chapter
OFDEG
different
obtained
described
contained
genome
genomes
similar
SSAHA2
datasets
untreated
platforms
sample
combined
remaining
frequency
containing
grouped
represented
mean
DC3000
Pr
types
quality
content
generated
single
read
Titanium
achieved
combinations
tetra-nucleotide
proportions
platform
improvement
however
CLARA
GS
also
number
combination
within
taken
tomato
much
longer
interval
chapter
difference
desired
short
trimmed
set
time
greater
total
one
using
Fig
suitable
present
DNA
clustered
case
error
statistics
breakdown
calculated
size
bp
compared
data
Figure
higher
separate
sequenced
oligonucleotide
values
chart
FLX
UT
evaluation
pv
provide
Rc
mapped
unassigned
considerable
samples
generation
profile
five
distribution
variation
vector
expected
possible
assigned
performed
use
relative
frequencies
sets
source
poor
variations
previously
derived
value
majority
pathogen
larger
type
sampled
multiple
almost

5

A comparison of clustering methods applied to true sequencing reads represented by composition-based feature vectors

Abstract

A range of clustering methods were compared, to establish the optimal method to be combined with the feature vectors elected previously, for grouping of DNA sequencing reads by their species of origin. An overview of different types of clustering method is provided, and the challenges associated with clustering of large datasets with high dimensionality discussed. A selection of suitable methods were used to cluster the reads contained in the simulated dataset UT +Psp2126, and the results compared. The most effective approach was found to be k*-means clustering, a relatively simple partitioning method. The most successful clustering of UT+Psp2126 was not sufficient to entirely isolate all of the reads belonging to either species in the sequenced sample. However, it was predicted that an enrichment in a cluster with reads derived from a particular species was likely to prove beneficial in further analysis of the data, such as in sequence assembly and/or the study of minority or pathogen species in a sample.*

Introduction

Previous chapters have dealt with the comparison and selection of feature vectors, to represent DNA sequencing reads for clustering based on their species of origin. Following comparison by CLARA clustering performance, it was concluded that tetra-nucleotide frequency distribution (TNF) vectors provided the most effective characterisation of sequences, of the four feature types compared and their combinations.

This chapter will focus on the second element of sequence comparison and grouping: the choice of clustering method. The aim of clustering is to identify distinct groups within a dataset (Berkhin 2006). The features used to represent the data determine the lines along which it is then grouped and separated. As such, the process of feature comparison can be thought of as the selection of a feature space in which the differences between sequences from different species, and similarities between those from the same species are best manifested. The process of clustering method comparison is then the selection of a method that can best identify these groups in the feature space. The suitability of a clustering method for this task is dependent on a number of factors including dataset size and dimensionality, cluster shape and intra-cluster data distribution.

There exist a huge number of methods for data clustering, utilising a variety of approaches and targeted towards many different fields and disciplines, including text-mining (Chen, Tseng et al. 2010), pattern recognition (Wiesinger-mayr, Vierlinger et al. 2007), analysis of flow cytometry data (Sugar and Sealfon 2010), and image analysis/interpretation (Del Frate, Pacifici et al. 2007). A complete review and comparison of the many methods available is beyond the scope of the work described here. Instead, the focus is on several of the main types of clustering method and their commonly used implementations, applied to raw high-throughput sequencing data characterised using TNF vectors.

Clustering methods fall into many categories. Several of the most commonly used approaches are described briefly here. For an overview of many clustering methods, and discussion of the implementation and limitations of these methods, see (Berkhin 2006).

k-Means and other partitioning clustering approaches

Partitioning approaches to clustering are defined as those methods where the dataset is divided into a set number of groups based on some measure of clustering quality. A common theme between such methods is the requirement for the desired number of groups to be specified, with a partitioning of the data based around a central point for each cluster. This differs from other approaches, such as hierarchical clustering, which do not require the number of groups to be input.

The most well-known of the partitioning clustering methods is *k*-means (KM) clustering (Macqueen 1967). In this approach, the data is grouped into a defined number of clusters, *k*, by minimising the distance between the points in a cluster and the mean data vector of these points. Several similar algorithms exist to implement this system (Forgy 1965; Hartigan and Wong 1979; Lloyd 1982), in addition to that first published by J. MacQueen.

In the MacQueen implementation, a number of points, *k*, are chosen as initial 'centroids' for clusters in the data. The remaining points in the dataset are added to the group whose centroid it is closest to in the feature space. After all data vectors have been added to a cluster, the centroid for each cluster is recalculated as the mean vector for all points in the cluster. Each vector in the dataset is then reassigned according to these new centroids, and the centroids recalculated again. This process is repeated until no change in the centroid positions produces a better solution. By minimising the distance between each data point and its cluster centroid, this process has the effect of minimising the sum-of-squares distance within each cluster (Macqueen 1967).

Other partitioning methods include the closely related *k*-medians, fuzzy *c*-means (Bezdek 1981) and partitioning around mediods (Kaufman and Rousseeuw 1990).

Fuzzy *c*-means (FCM) clustering is a 'soft' version of *k*-means clustering, where each datapoint is associated with a set of weightings corresponding to its level of membership to each cluster produced. A datapoint that is very close to a single centroid will be heavily weighted towards that cluster, while a datapoint that is not located so close to any centroid in the feature space will have weightings that are more evenly spread between multiple clusters (Bezdek

1981).

Partitioning around mediods (PAM) uses representative points in the data - mediods rather than centroids - as the centres around which clusters are defined, based on minimising a measure of dissimilarity between points in a cluster, rather than maximising similarity as in KM and FCM clustering. For large datasets where the PAM algorithm is unsuitable due to time and memory requirements, a variant of the method, CLARA (Clustering LARge Applications), is used. In CLARA, a sample is taken from the dataset and clustered as with the PAM method, after which the remaining datapoints are grouped according to the cluster mediod that is closest in the feature space (Kaufman and Rousseeuw 1990).

As PAM and CLARA use representative points in clusters, rather than a centroid vector, the approach is more robust to outliers in clusters, which would distort the values in any such averaged vector (Kaufman and Rousseeuw 1990).

Clustering results produced by partitioning methods can be sensitive to the datapoints chosen to initialise the grouping. As the grouping is built around a moving average originating from a (usually randomly) chosen datapoint, the process of rearrangement to find a grouping is liable to settle at a local minimum if the initial centres are not located closely enough to the true cluster centres in the data. This effect may be guarded against by choosing centres manually (if approximations to the appropriate centres are known), or else performing the clustering multiple times and choosing the best/most frequently observed solution.

Cluster validity

If the true number of different populations sampled in a dataset is not known, as is likely to be the case for the kind of sequencing datasets of interest in this project, the correct number of groups into which the data should be clustered is difficult to estimate with great confidence.

The problem of predicting the correct or optimal number of clusters into which a dataset should be separated is referred to as cluster validity and has been studied for many years (for an in-depth, albeit slightly outdated, overview of this subject, see this review (Halkidi, Batistakis et al. 2001) and the references therein). Cluster validity predictions are of particular importance in partitioning

clustering where the number of groups must be defined at the outset of analysis. If a sub-optimal number of clusters is produced, points that are distinctly grouped in a dataset may be clustered together (where the defined number of clusters is too small) or arbitrarily divided into several groups (where the defined number of clusters is too large). How well-defined the groups of datapoints, and which points are grouped together, is dependent on the feature used to compare them.

This introduces an interesting question with regard to the clustering of the sequencing read datasets of interest here: is the optimum number of clusters for these datasets equal to the number of species that contribute a considerable proportion of reads in the data? Or do the reads from a single species form more than one cluster within the feature space, perhaps due to regional variations in tetranucleotide frequency within the genome (as with isochores of differing G/C content)?

The optimal number of clusters predicted by a cluster validity method is based on a purely mathematical approach, with no prior information on the 'correct' number of populations within the data assumed. As such, the results of cluster validity analysis and the desired outcome of read clustering may not be identical, as the optimal grouping determined by cluster validity may divide reads originating from a single genome into multiple clusters.

The effect of grouping the dataset into a number of clusters larger than the number of species present was investigated in Chapter 4, where the UT +Psp2126 dataset was separated by CLARA into five clusters, using a range of feature types and combinations. The results described there suggested that increasing the number of clusters in this way was most likely to result in division of reads from the *A. thaliana* host into multiple groups while the clustering of those from *Pseudomonas* sp. 2_1_26 tended to remain largely uneffected. This may be due to the difference in complexity between the plant and bacterial genomes, with regional differences in tetranucleotide frequency more likely to be present in the genome of *A. thaliana*.

A full survey of available validity methods was not appropriate here. Instead, a selection of methods were used based largely on accessibility and ease of implementation. The cluster validity methods used in this work are prediction

strength (Tibshirani and Walther 2011), the gap statistic (Tibshirani, Walther et al. 2001), and *pamk()*, an implementation of the PAM/CLARA clustering method in *R* by Christian Hennig, that returns the optimal number of clusters in a specified range.

In the *pamk()* implementation, the choice of the optimal number of clusters is based on the average silhouette width of the clusters produced in each case (part of the package *fpc*, available at http://cran.r-project.org/web/packages/fpc/index.html). The silhouette width of a cluster is a measure of how closely grouped the points in the cluster are, and how well-defined the separation is between clusters in the results of the PAM analysis (Kaufman and Rousseeuw 1990).

In the prediction strength method, the number of clusters is predicted based on cross-validation of results obtained from clustering of samples taken from the dataset (Tibshirani and Walther 2011). The same authors produced the gap statistic as a measure of cluster validity. This method measures a difference between the dispersion within clusters and that that would be expected from an appropriate standard distribution of data. The most suitable number of clusters is selected as the value that maximises this difference (Tibshirani, Walther et al. 2001).

Previous research has highlighted a tendancy for the *k*-means clustering algorithm to preferentially group data into similarly-sized clusters (Yeung, Haynor et al. 2001), behaviour that may be a recurring issue with partitioning methods, when applied to a dataset composed from classes present in unequal proportions.

Another difficulty associated with partitioning approaches, and common between many clustering methods that use Euclidean distance as a measure of the relatedness of points in a dataset, is the bias towards production of spherical/convex clusters. As grouping of data in KM, CLARA and FCM clustering is based on finding the centres (whether centroids or mediods) that minimise the Euclidean distance between the points and the centre in a cluster, the ideal shape for such a cluster will be spherical around the centre. This bias makes such methods less suitable for finding groups in the dataset that are not arranged in such a standard shape (Berkhin 2006).

Hierarchical clustering

Unlike partitioning clustering methods, hierarchical clustering does not split the dataset into a defined number of clusters, instead producing a dendrogram, a treelike framework of connections, between datapoints according to how closely they are located in the feature space. A ‘stopping criterion’ can be applied to the clustering, such that once the data has been grouped into a certain number of clusters, or the level of divergence or similarity in the groups produced has reached a certain threshold, the clustering process is ceased. However, the specification of such a criterion is not required for clustering.

Generally speaking, hierarchical clustering can operate in either a divisive, ‘top-down’ or agglomerative, ‘bottom-up’ fashion. In the former, the whole dataset is split into constituent parts based on a measure of distance between groups of points. This process is repeated in a stepwise fashion until the dataset can be split no further, that is, when each individual datapoint is in a ‘group’ of its own. A common analogy holds that ‘top-down’ hierarchical clustering can be thought of as producing a tree from the trunk (the whole dataset), outwards through a network of increasingly small branches, to the individual leaves (the datapoints themselves). ‘Bottom-up’ clustering operates in the reverse direction, from the leaves to the trunk, grouping datapoints with those closest to them, then grouping the closest groups together repeatedly until a single cluster is formed of the whole dataset (Kaufman and Rousseeuw 1990; Murtagh and Contreras 2012).

With the data clustered in this way, the results can be interpreted as any number of groups (up to the number of individual points in the dataset), by interrogating the dendrogram at the appropriate branching point. Choosing this branching point, without prior knowledge of the exact composition of the dataset, may be difficult.

Where a decision must be made on which groups to merge, the distance between groups of points can be measured in a number of ways, generally referred to as ‘linkage metrics’. Commonly used linkage metrics include, single-link, average link and complete link, corresponding respectively to the use of the distance between the two closest points in the two clusters, the mean centroids of the two clusters, and the two farthest-apart points in the clusters.

Where these linkage metrics are measured by Euclidean distance, hierarchical clustering preferentially constructs spherical clusters as described previously. Approaches to hierarchical clustering have been developed that are more suitable for identifying groups of non-standard shape in a dataset (Karypis, Eui-Hong et al. 1999; Guha, Rastogi et al. 2001).

Hierarchical clustering requires the construction of a *n* x *n* connectivity matrix (where *n* is equal to the number of points in the dataset) detailing the distance between each point in the data and every other point. For large datasets like the sequencing data that is the subject of these investigations, this connectivity matrix becomes too large to be held in system memory and as such most hierarchical clustering approaches are inappropriate for such analysis. Methods have been developed to tackle this, by producing a dendrogram from a sample of the data, with groups at each branch represented by a sample of the points contained (Guha, Rastogi et al. 2001), by reducing the size of the connectivity matrix by removing all values bar those corresponding to a set number of nearest neighbours for each point (Karypis, Eui-Hong et al. 1999), or by creating a summary of the dataset and using this for clustering (Zhang, Ramakrishnan et al. 1997).

Density-based clustering

A cluster in a dataset, represented spatially, can be thought of as a region of the feature space more densely populated with datapoints than the surrounding area. Density-based clustering methods define groups in the data based on the numbers of points located in the same region of the feature space. If the number of points in a given region of a set size is above a given boundary value, these points are grouped together along with any other points similarly closely located in the surrounding feature space.

One of the most popular density-based clustering methods, DBSCAN (Density Based Spatial Clustering of Applications with Noise, Ester, Kriegel et al. 1996), works on a neighbourhood system, where a cluster is defined as a collection of points within a set distance limit (the neighbourhood) of a preset number of other points in the feature space. The cluster consists of the neighbourhood of every point fulfilling these criteria, that is within the neighbourhood of at least one other point in the cluster.

The use of distance between one point and the next to define the boundaries of a cluster allows for clusters of any shape to be identified by density-based clustering, and provides automatic removal of outliers from the grouping (Kailing, Kriegel et al. 2004).

Density-based clustering performance tends to suffer when applied to data with a higher dimensionality than standard spatial observations in two or three dimensions (Weber, Schek et al. 1998). The TNF vectors used here to characterise DNA sequences contain 136 individual values, rendering approaches such as DBSCAN unsuitable for use in analysis of such a dataset. The methods are also not scaleable to larger datasets (Viswanath and Suresh Babu 2009). Density-based methods exist that are less sensitive to high dimensionality, which may be more suited to application to a dataset of TNF feature vectors (Hinneburg and Gabriel 2007; Viswanath and Suresh Babu 2009; Sugar and Sealfon 2010). Methods that are applicable to large datasets are not necessarily applicable to high-dimensional data, and vice versa. For example, the Misty Mountain clustering described by (Sugar and Sealfon 2010) does not scale well to higher levels of dimensionality such as the 136 dimensions of the TNF feature space.

Spectral clustering

Spectral clustering methods implement a dimensionality reduction on a connectivity matrix of the dataset, by constructing a graph of relationships between datapoints and finding a partition of this graph to group the data (Kannan, Vempala et al. 2000; Ng, Jordan et al. 2001; Tian, Yang et al. 2008; Zhang and You 2011).

These methods can be powerful when applied to relatively small datasets but, largely due to the requirement for a full connectivity matrix to be constructed to compare the datapoints are much less suitable for analysis of large datasets (Yan and Jordan 2009). Methods have been developed that are more applicable, including KASP (k-means-based approximate spectral clustering, Yan and Jordan 2009), approximating the optimal solution using a set of representative points to partition the dataset as a whole.

Model-based clustering

Model-based methods constitute a different approach to clustering. Where

many of the methods described previously operate through direct interrogation of the points in the dataset and the relationships between them, model-based clustering treats the datapoints as sampled data from one of a number of distributions in a mixed population. The aim is to fit a mixture model - a set of defined distributions with different means in the feature space - to the observed data and try to identify the points that belong to each distribution in the mixture, providing a grouping of the data (Berkhin 2006).

The selection of an appropriate model, consisting of suitable distributions that fit the observed data, allows for the number of different groups of classes in the data to be estimated - each distribution should correspond to a single group.

The selection of a mixture model from a range can be achieved by finding the best fit using the Expectation-Maximisation (EM) Algorithm (Dempster, Laird et al. 1977; Fraley and Raftery 2002). For each mixture model, the EM Algorithm calculates the probability that each datapoint belongs to each distribution, producing a weighted classification for each point and approximating the parameters of the distributions present, and then calculates a likelihood that the observed data originated from the mixture model of these distributions, providing a measure of the quality of the model.

Many examples of model-based clustering exist, one of the most commonly used being an implementation in the MCLUST software package, which builds mixture models from Gaussian distributions (Fraley and Raftery 1999). Other implementations exist, using other types of distribution (Cheeseman and Stutz 1996) and even mixtures of mixtures of distributions (Browne, McNicholas et al. 2012).

Self-organising maps

Self-organising maps (SOMs) are a type of artificial neural network first introduced by (Kohonen 1982). The map consists of a grid of nodes, onto which vectors from the dataset are placed one-by-one. Vectors are chosen at random from the dataset and placed onto a node on the grid. After the first vector has been placed, the nodes to which the remaining vectors are added are determined based on similarity between the vector to be added and those already present on the grid. Vectors are placed on the same node, or an adjacent node to those to which they show similarity. As vectors are placed, the

grid ‘learns’ the data, with distances between nodes growing and shrinking depending on the degree of similarity between the data placed onto each node (Kohonen 1982; Kohonen 1990).

Initially the grid of nodes is equally spaced, but as vectors are added and the learning process progresses, the grid warps and stretches. After the whole dataset has been incorporated, nodes containing similar vectors are held more closely together and those that are more different are further apart (Kohonen 1982; Kohonen 1990).

As in KM clustering, the similarity between a vector and those on the grid is measured as the distance in the feature space between the vector and a mean vector of those already assigned to each node. As vectors are added to the grid, these mean vectors are recalculated.

In addition to affecting the mean vector representing its node, the assignment of a vector also affects the mean vectors representing the nodes nearby. The magnitude of this effect is dependent on the distance between the nodes, and the progress through the learning process. Initially, the effect that a vector assignment has on nearby nodes is large, but as more vectors are added and the grid fits to the data, this effect is reduced (Kohonen 1982; Kohonen 1990).

For large maps composed of thousands of nodes, clusters of nodes, themselves containing groups of similar data vectors, may be identified after a dataset has been applied to the SOM, using a matrix of distances between the nodes and their nearest neighbours (Ultsch and Fabian 2005).

SOMs have been successfully applied to DNA sequence data, characterised by their nucleotide composition (Kanaya, Kinouchi et al. 2001; Abe, Kanaya et al. 2002; Abe, Kanaya et al. 2003; Abe, Sugawara et al. 2006). Recently, growing SOMs that allow for new nodes to be created if the existing node cannot adequately represent the variation in the data assigned to it, and hierarchically structured maps projected onto non-Euclidean space have been introduced for such analysis (Chan, Hsu et al. 2008; Martin, Diaz et al. 2008).

The hyperbolic hierarchically growing SOM (HHSOM) is of particular interest here, due to it’s apparent capacity to successfully group and separate sequences of a length comparable to those produced in high-throughput sequencing (Martin, Diaz et al. 2008).

Data can be grouped with an SOM in an unsupervised manner, with nodes and the distances between them allowing for groups to be identified in the data. However, the successful clustering of short (200 bp and 500 bp) sequence fragments using an HHSOM was achieved using a semi-supervised method, where longer (e.g. 50 kbp) fragments from the genomes being clustered were used to construct the SOM and the shorter fragments classified with this map after the learning process had been completed (Martin, Diaz et al. 2008). This suggests that the unguided grouping of raw sequencing reads being assessed here may prove unsuccessful with an HHSOM in the absence of a set of longer sequences for training.

Another difference between the published implementation of HHSOM and the approach taken here is in the type of features applied to the map in clustering. Martin et al utilised a variant of TNF features in their work - the *tf-ti* features modified to amplify the signal from rare oligonucleotides present in reads and mask that of oligonucleotides common throughout the dataset (Martin, Diaz et al. 2008). These features are described in more detail in Chapter 2 of this work.

The SOM-based approach to clustering is not so sensitive to the size of the dataset, as each datapoint is considered only once during clustering, and its placement onto a node is determined in the context of the current state of the SOM grid. This allows the method to scale well with an increasing number of datapoints.

Comparison of clustering methods

The clustering performance of several of the methods outlined here was compared, to find the combination of TNF features and clustering method that produced the best grouping of sequencing reads according to their species of origin. Clustering analysis was applied to the UT+Psp2126 dataset of sequencing reads obtained from untreated *A. thaliana* plant tissue and the bacterium *Pseudomonas* sp. 2_1_26, described in the previous chapter, with sequences characterised as TNF vectors.

The methods used in this comparison were chosen based on availability (generally as an implementation in *R*), and suitability for use with a large, multidimensional dataset such as UT+Psp2126 or any other dataset of raw high-throughput sequencing reads, represented by TNF features.

As mentioned previously, TNF feature vectors consist of 136 relative frequency values and as such the TNF feature space can be described as 136-dimensional. The adverse effects of high-dimensionality on density-based clustering was discussed previously, and it was concluded that the DBSCAN method was not suitable for application to this data. No suitable *R* implementation of DENCLUE (Hinneburg and Gabriel 2007), a density-based approach less sensitive to high-dimensionality, was available at the time of writing.

The size of the dataset presented a problem for hierarchical and spectral clustering. The requirement for construction of an *n* x *n* connectivity matrix of the dataset constituted too great a burden on system memory for standard implementations of these methods to be used. This is especially apparent when we consider that the UT+Psp2126 dataset is comparably small in relation to many sequencing datasets.

A variant of spectral clustering, KASP, was used in this comparison. As mentioned previously, KASP provides an approximation to spectral clustering for large datasets, by operating on a sample of representative points in the dataset to produce a solution using *k*-means clustering (Yan and Jordan 2009).

Three partitioning clustering methods were compared: KM, FCM and CLARA. Two cluster validity methods described earlier, prediction strength and *pamk()*, were used to estimate the optimal number of clusters that should be produced using these methods. Results obtained for the numbers of clusters predicted to be optimal by these methods, as well as for two clusters (in accordance with the two species known to be represented in the dataset) were compared. The gap statistic was not used to predict an optimal number of clusters as it was found to be unsuitable for application to such a large dataset.

Finally, an implementation of the HHSOM method in MatLab, kindly provided by Christian Martin and colleagues, the original developers of the system, was also used and the results compared with those from other methods.

Evaluation of cluster quality

One of the challenges associated with a comparison such as this, of a broad range of methods with differing input and output, is in finding a systematic method for evaluating the results and measuring the relative quality of the

separation of data in each case.

So far in this project, where data has been separated by a partitioning clustering method (CLARA), clustering has been assessed in large part using statistics of precision and recall. These metrics are used again here, with the aim of providing a simple and easily understood means of direct comparison between clustering by different methods and different variations of the same method.

In general terms, the ideal separation of a dataset such as that used here would produce two clusters with each wholly containing (i.e. with both precision and recall scores of 1.0) the reads derived from one of the two principal contributing species, *A. thaliana* and *Pseudomonas* sp. 2_1_26. The quality of clusters produced here was evaluated based on how closely they resembled this ideal separation.

The quality of these clusters was evaluated with a particular focus on the isolation of *Pseudomonas* reads. This view was taken based on the disproportionality of representation of the two species in the dataset: reads from *Pseudomonas* account for approximately 16% of UT+Psp2126, and effectively separating this minority of reads to isolate the ‘pathogen’ was of particular interest in this project. Where the number of clusters produced was greater than the number of species predicted to be present in the dataset (i.e. >2), this approach was taken with the aim of further isolating these *Pseudomonas* reads at the cost of dividing those derived from *A. thaliana* into multiple groups.

This may be necessary as subgroups with differing typical TNF profiles may exist within the plant reads, taken from regions of the *A. thaliana* genome with a slightly different distribution of tetranucleotides, as with G/C content in the isochores of the genome. By increasing the number of clusters produced, the reads from these sub-populations may be grouped together separately, preventing them from being erroneously included in a cluster that would otherwise be principally composed of *Pseudomonas* reads. This applies more widely to other host-pathogen datasets, where one genome is larger and more complex, and contributes a larger proportion of reads.

Materials and Methods

Generation of TNF feature vectors from sequencing reads

TNF vectors were generated for sequencing reads in the UT+Psp2126 dataset in FASTA format, using the *perl* script 'featureWriter.pl', reproduced in Appendix A.

k-Means clustering

KM clustering was performed using the *kmeans()* implementation, part of the *R* package *stats*. Unless otherwise stated, default settings were used in clustering. All *R* packages used here are available through CRAN (the Comprehensive *R* Archive Network, cran.r-project.org/).

The implementation allows for a choice of four different algorithms for clustering - Hartigan-Wong, Lloyd, Forgy and MacQueen (Forgy 1965; Macqueen 1967; Hartigan and Wong 1979; Lloyd 1982). A comparison of clustering results, with the UT+Psp2126 dataset, obtained using each of these four options showed no difference between clusters produced (results not shown), and the default algorithm (Hartigan-Wong) was used in all *k*-means analysis detailed here.

Fuzzy *c*-means clustering

Fuzzy *c*-means (FCM) clustering was performed using the *cmeans()* implementation, part of the *R* package *e1071*. Default settings were used unless otherwise stated. The *cmeans()* implementation allows for a choice between Euclidean and Manhattan distance measures for clustering of datapoints, and for the degree of fuzziness of clustering to be defined. A comparison between the two distance measures, and the effect of increasing fuzziness of clustering on the grouping produced, is discussed later.

CLARA

As in previous chapters, CLARA clustering was performed using the *clara()* implementation, part of the *R* package *cluster*. Unless otherwise stated, default settings were used in clustering. Similarly to the *cmeans()* algorithm mentioned above, *clara()* allows the user to choose a distance measure, between Euclidean and Manhattan distance. A comparison of the quality of clustering obtained with each of these two options suggested that the use of a different distance measure produced very little difference in outcome (results not shown

here). Euclidean distance, the default setting, was used in all CLARA experiments discussed here.

Cluster validity

Three cluster validity methods were used with the dataset, to estimate the optimal number of clusters to be produced with partitioning methods KM, FCM and CLARA. Such an estimation using the gap statistic (Tibshirani, Walther et al. 2001) was found to take much too long to produce a result with such a large dataset and the method was discarded.

An estimation was made with the more recent prediction strength method (Tibshirani and Walther 2011), using the *R* implementation *prediction.strength()*, and with the function *pamk()*, both implemented by Christian Hennig in the *R* package *fpc*.

The optimal number of clusters to be produced from UT+Psp2126 was estimated as 3 from the prediction strength method, and 7 from *pamk()* (results not shown).

KASP

Spectral clustering using *k*-means, KASP, was implemented using an R implementation *kasp()* detailed in (Yan and Jordan 2009) and downloaded from the authors' website (http://www.cs.berkeley.edu/~jordan/fasp.html).

The *kasp()* implementation allows for input of two variables, α and σ, that affect data sampling and cluster boundary estimation during clustering. The fraction of the dataset sampled for clustering is determined by α and σ dictates the Gaussian bandwidth kernel. Selection of optimal values for these parameters is discussed later. Beyond these parameters, default settings were used.

HHSOM

Grouping of data with a hyperbolic hierarchically growing self-organising map (HHSOM) was performed using a *MatLab* implementation, published by (Martin, Diaz et al. 2008), and kindly provided by the authors (C. Martin, by personal correspondence).

Results

Parameter selection for FCM clustering

The *R* implementation of FCM used here provided a choice of distance measure to be used in computing the similarity between datapoints, between Euclidean and Manhattan (city block) distance. To investigate the effect that this choice of distance measure might have on the effectivity of clustering of sequence reads, the clusters produced from the UT+Psp2126 dataset with each option were compared. The results are given in Table 5.1.

Table 5.1 *A comparison of the FCM clustering results obtained from grouping the UT+Psp2126 dataset into two clusters using Euclidean and Manhattan distance to measure the similarity between TNF vectors.*

		Euclidean	Manhattan
Cluster	**Organism**	**Reads in cluster**	**Reads in cluster**
1	***A. thaliana***	13254	22380
	Pseudomonas sp. 2_1_26	18282	18726
2	***A. thaliana***	93040	83914
	Pseudomonas sp. 2_1_26	763	319

The clusters produced using each distance measure were broadly similar. In both cases, the clusters were produced with high recall - they contained the vast majority of reads in the dataset belonging to either *A. thaliana* or *Pseudomonas* sp. 2_1_26 (*Pseudomonas*). In the case of the predominant class in the data, *A. thaliana*, this high recall was also reflected in a high precision value for the cluster, with 99.19% of reads in Cluster 2 generated using Euclidean distance being derived from *A. thaliana*, accounting for 87.53% of all *A. thaliana* reads in the dataset, and 99.62% of reads, accounting for 78.95% of all *A. thaliana* reads where the data was clustered using Manhattan distance.

The high recall of *Pseudomonas* sp. 2_1_26 reads in the opposing clusters of both sets of results was not reflected in a high precision for these clusters, with bacterial sequencing reads accounting only for around half of the total number of reads in these clusters (57.97% (Euclidean) and 45.55% (Manhattan)).

Beyond this general pattern within the results, the difference between the clustering produced using the two distance measures could be summarised as follows: clustering using Manhattan distance grouped a slightly larger proportion of the *Pseudomonas* reads into a single cluster in comparison with clusters produced using Euclidean distance, but at the cost of reduced precision within that cluster. The inclusion of an extra 444 *Pseudomonas* reads was accompanied by 9126 *A. thaliana* extra reads in the same cluster. With this cost in mind, it was concluded that Euclidean distance was most suitable for use in all further FCM analysis described here.

The second variable in FCM clustering that could effect the quality of clusters produced is the degree of ‘fuzziness’ used in grouping of the data. To determine the optimal value for this parameter, clustering of UT+Psp2126 was performed with a range of fuzziness values. The results are given in Table 5.2 and illustrated in Figure 5.1.

Table 5.2 *Results of FCM clustering of UT+Psp2126 dataset represented by TNF feature vectors, into three groups and with increasing degree of fuzziness used in grouping of the data. The clusters produced are broken down by sequencing reads generated from* A. thaliana *and* Pseudomonas *sp. 2_1_26.*

		Fuzziness							
Cluster	**Organism**	**1.1**	**1.2**	**1.3**	**1.4**	**1.5**	**2**	**3**	**4**
1	***A. thaliana***	61383	56628	54152	51578	49980	79553	79592	79592
	***Pseudomonas* sp. 2_1_26**	1036	985	841	658	615	242	242	242
2	***A. thaliana***	10479	10631	11493	13256	13947	26660	26673	26678
	***Pseudomonas* sp. 2_1_26**	17934	17965	18101	18285	18332	18803	18803	18803
3	***A. thaliana***	34432	39035	40649	41460	42367	81	29	24
	***Pseudomonas* sp. 2_1_26**	75	95	103	102	98	0	0	0

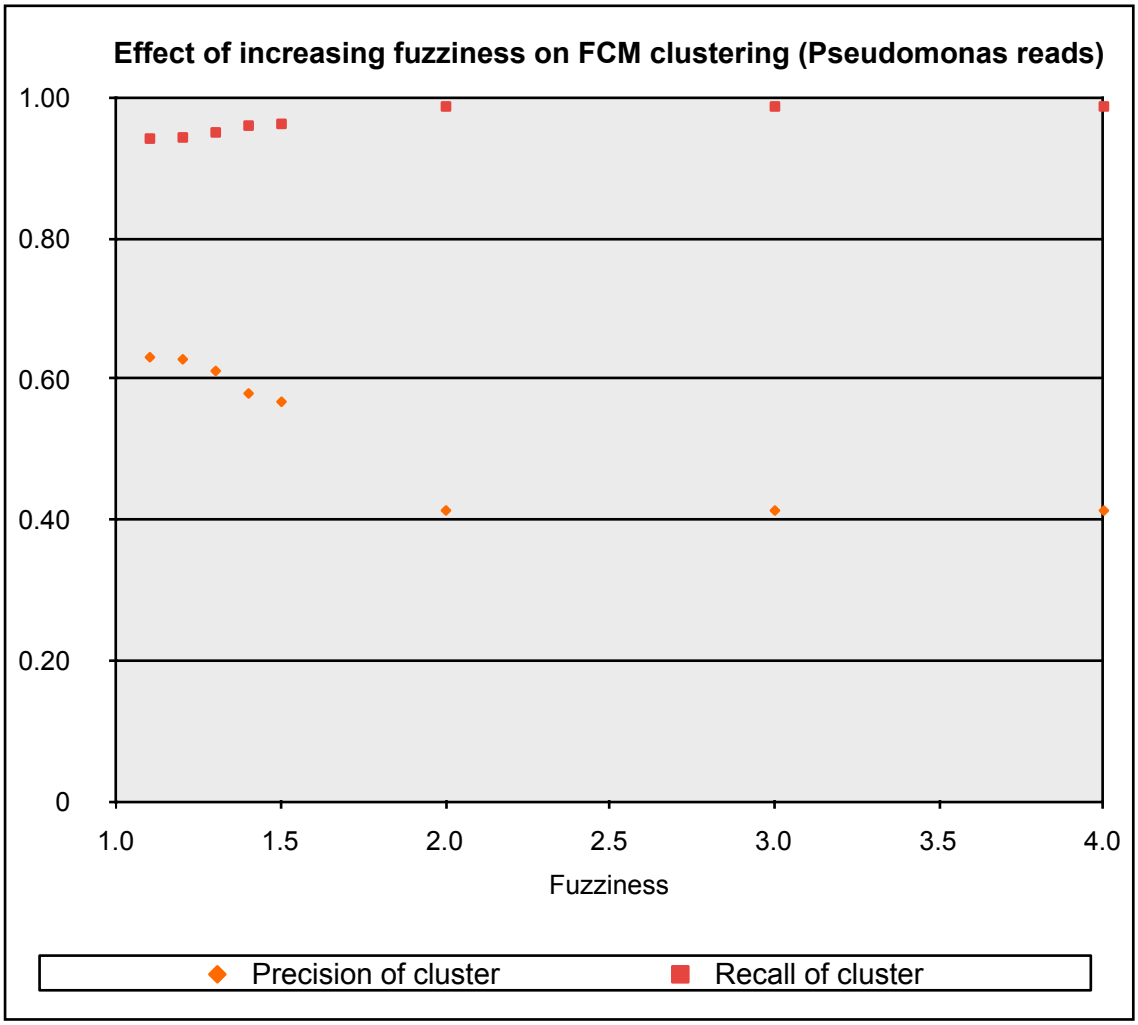


Figure 5.1 *Precision and recall values of* Pseudomonas *sp. 2_1_26 sequencing reads for the cluster with the greatest recall of these reads in results of FCM clustering into three groups, with increasing fuzziness.*

The Pr and Rc values plotted in Fig. 5.1 were derived from FCM clustering of the dataset into three groups. The statistics for the cluster with the greatest recall of *Pseudomonas* reads are shown in the figure, to illustrate the effect of varying the degree of fuzziness in clustering on the grouping of this minority class within the data. As can be seen from Table 5.1, the vast majority of *Pseudomonas* sequencing reads in the dataset were consistently grouped together into a single cluster. It is the quality of clustering achieved with this minority class within the dataset (the *Pseudomonas* sequences) that is most indicative of the success of the clustering performed.

The results indicated that the proportion of all *Pseudomonas* reads within the dataset that were contained in a single cluster (reflected in the recall value of the cluster) increased with increasing fuzziness of clustering, reaching a limit just below 99% when the degree of fuzziness was increased ≥2. However, this increased recall of *Pseudomonas* reads was accompanied by a decrease in the precision of clustering, to a limit of ~41%, caused by the increasing inclusion of *A. thaliana* reads within the cluster. The increase in recall associated with increasing fuzziness of clustering from a minimal value of 1.1, to a maximum value of 4, was from 94.17% to 98.73%, with a decrease in precision of clustering from 63.12% to 41.34%.

The cost to clustering precision, associated with a relatively modest increase in recall provided by increased fuzziness of clustering, suggested that fuzzy clustering of this data did not provide an advantage over standard, non-fuzzy partition clustering methods such as KM and CLARA.

Parameter selection for spectral clustering

The KASP spectral clustering implementation used in this work allowed for two parameters used in the grouping of data to be defined. The authors of the implementation advised that a range of values be tried for these variables, *α* and *σ*, to determine the optimal combination for the dataset concerned. The example data that accompanied the implementation used included several datasets of comparable size and complexity to the UT+Psp2126 dataset used here, with guideline values of *α* and *σ* provided for these example datasets.

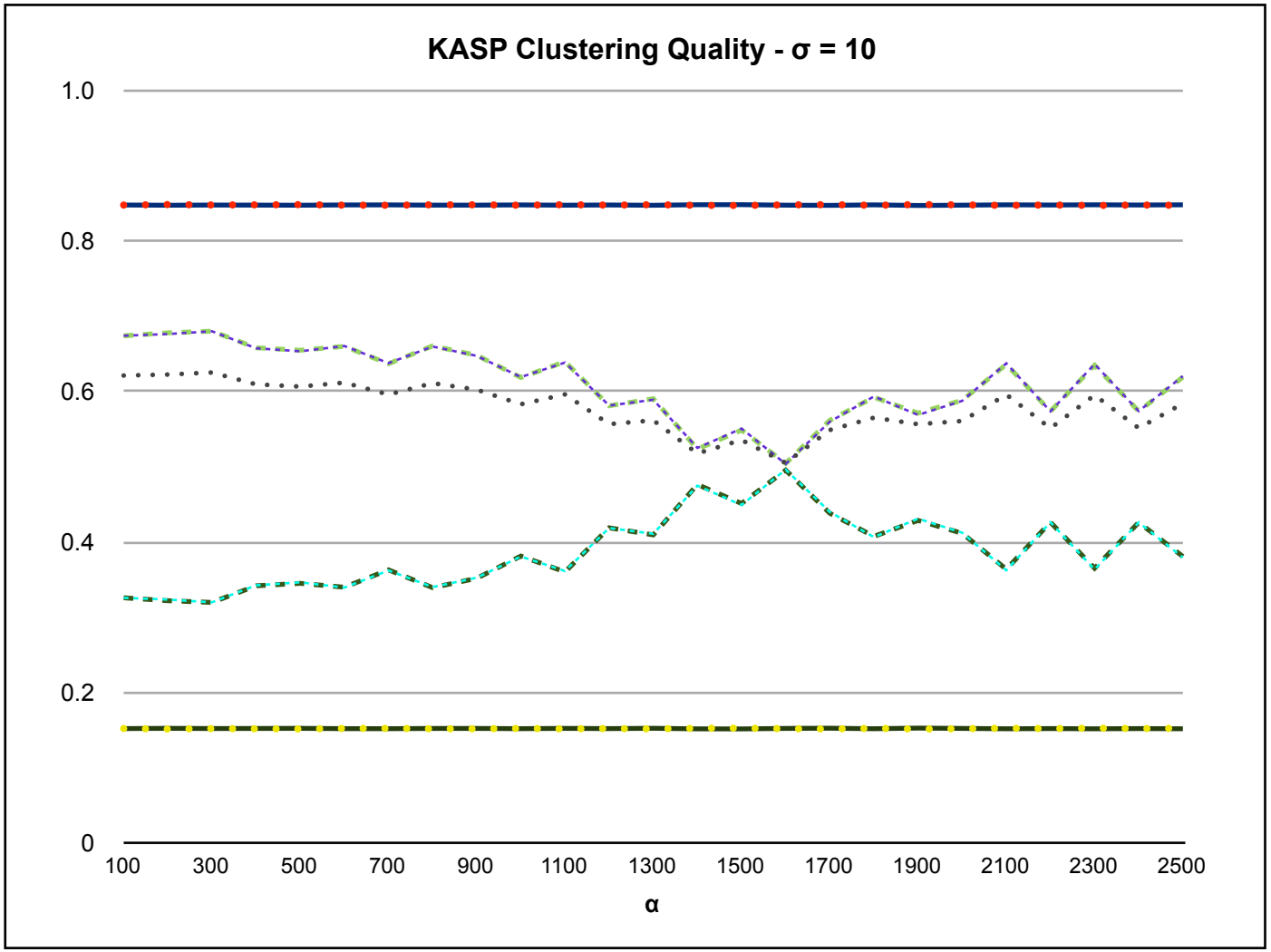
These datasets varied in three properties, the number of datapoints in the set, the number of features representing each datapoint (the dimensionality), and the number of classes within the dataset. Two of the example datasets were comparable to UT+Psp2126 (the ‘Connect 4’ and ‘Census Income’ datasets), with guideline values for *α* and *σ* of 200 and 50, and 500 and 10, respectively.

In addition to these suggested values, the authors also recommended that *σ*, the Gaussian kernel bandwidth parameter, be searched over a range to find the optimal value for a dataset. In order to estimate the optimal values of both *α* and *σ* for KASP analysis of UT+Psp2126, clustering was performed for five different values of *σ* over a range of values for *α*, the data reduction (sampling) ratio. An example of the results obtained is given in Figure 5.2. The results obtained were very similar for all values used, (*σ* = [1, 10, 20, 50, 100]), and only the results for *σ* = 10 are reproduced here.

The quality of clustering as measured by the proportion of reads successfully separated was included here as this measure was built into the *R* implementation used here. However, as the figure clearly shows, the results of KASP clustering are much better represented using the precision and recall statistics used previously, as these measures provide an evaluation of clustering independent of the relative size of the clusters produced.

The variance in quality of results as measured by this statistic (not shown in Fig. 5.2 for ease of interpretation), increased with increasing sampling ratio, as smaller samples were taken from the dataset as representative points for *k*-means clustering.

Figure 5.2 *Results of KASP spectral clustering of UT+Psp2126, over a range of values for α, the sampling ratio, with σ, the Gaussian kernel bandwidth, set at 10. The same analysis was performed, over the same range of values for α, for a selection of other values for σ yielding near identical results.*


KASP Clustering Quality - σ = 10
1.0
0.8
0.6
0.4
0.2
0
100
300
500
700
900
1100
1300
1500
1700
1900
2100
2300
2500
α
KASP Proportion Statistic
A. thaliana Pr 1
Pseudomonas Pr 1
A. thaliana Pr 2
Pseudomonas Pr 2
A. thaliana Rc 1
Pseudomonas Rc 1
A. thaliana Rc 2
Pseudomonas Rc 2

The plots of these statistics in Figure 5.2 showed that, although the overall proportion of reads correctly separated by KASP varied across a range of values for *α*, the clustering of sequencing reads by KASP remained close to random across the range of values used for *α* and *σ*.

The consistency of the precision values for both species in both clusters, ~84.8% for *A. thaliana* and ~15.2% for *Pseudomonas*, across the whole range of values for *α* showed that the variation in recall values and the KASP statistic was the product of variation in the size of the clusters produced. The precision values remained nearly identical to the proportions of sequences from each species in the dataset as a whole, suggesting that the grouping of sequences by KASP was near-random, and did not provide any significant enrichment for sequences from either species.

It may be the case that no appropriate combination of values for *α* and *σ* exists that would allow effective KASP clustering in this case, perhaps due to the sie and/or dimensionality of the dataset. However, a more thorough approach searching across a range of values for *σ* with a set value for *α* could allow a more robust conclusion to be drawn.

It is not known whether an optimal set of values for *α* and *σ*, determined for UT +Psp2126 or some other dataset, would be applicable to all sequencing datasets prepared in a similar way. This process of determining optimal parameter values could constitute an inconvenient and lengthy additional step in the clustering process, especially given the poor quality of results obtained in this analysis.

HHSOM

When originally published by Martin, Diaz and colleagues (2008), the HHSOM method was applied to a dataset of 350 microbial genomes represented with feature vectors closely related to tetra-nucleotide frequency distributions, using a hierarchical self-organising map arranged in five rings over hyperbolic space. This suggested that the size and dimensionality of UT+Psp2126 should not be limiting on the performance of the HHSOM method.

The UT+Psp2126 dataset used here, containing sequences from the genomes of only two species, differed substantially in nature from this microbial dataset. With fewer species contributing to the dataset the appropriate number of nodes onto which the data will be clustered was predicted to be smaller, corresponding to the requirement for less space to distinguish between groups of reads. To determine whether a smaller grid, constructed from nodes on fewer rings, was more appropriate for use with this dataset, results of HHSOM analysis of UT +Psp2126, on a grid of two, three and five rings were evaluated. In each case, the HHSOM was trained and tested with all UT+Psp2126 reads, represented as TNF feature vectors. The results are illustrated in Figures 5.3, 5.4 and 5.5.

Figure 5.3 *The number of sequencing reads from* A. thaliana *(blue) and* Pseudomonas *sp. 2_1_26 (green) assigned to each node of a 2-ring HHSOM trained with the dataset UT+Psp2126.*

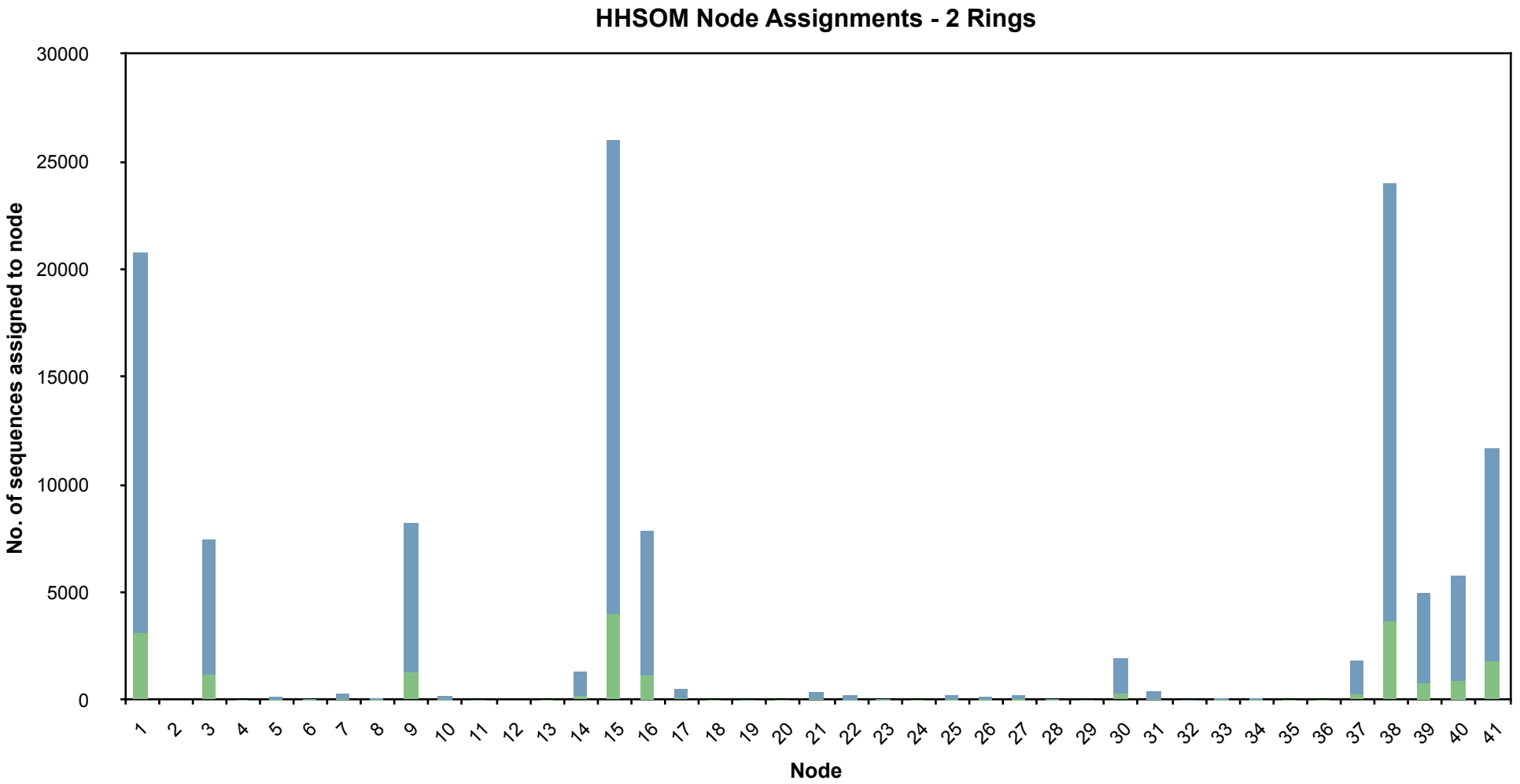

HHSOM Node Assignments - 2 Rings
No. of sequences assigned to node
30000
25000
20000
15000
10000
5000
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
Node

Figure 5.4 *The number of sequencing reads from* A. thaliana *(blue) and* Pseudomonas *sp. 2_1_26 (green) assigned to each node of a 3-ring HHSOM trained with the dataset UT+Psp2126.*

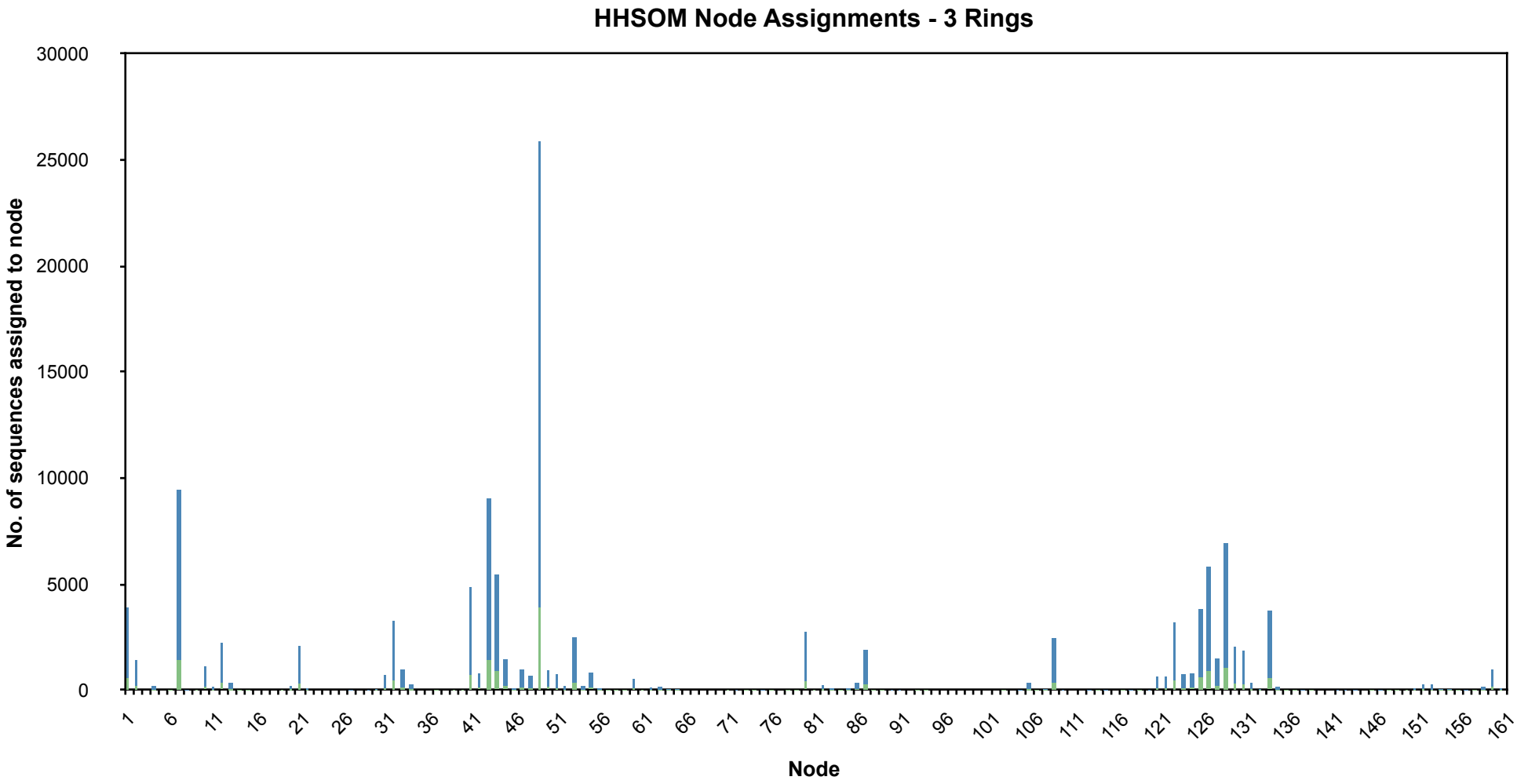
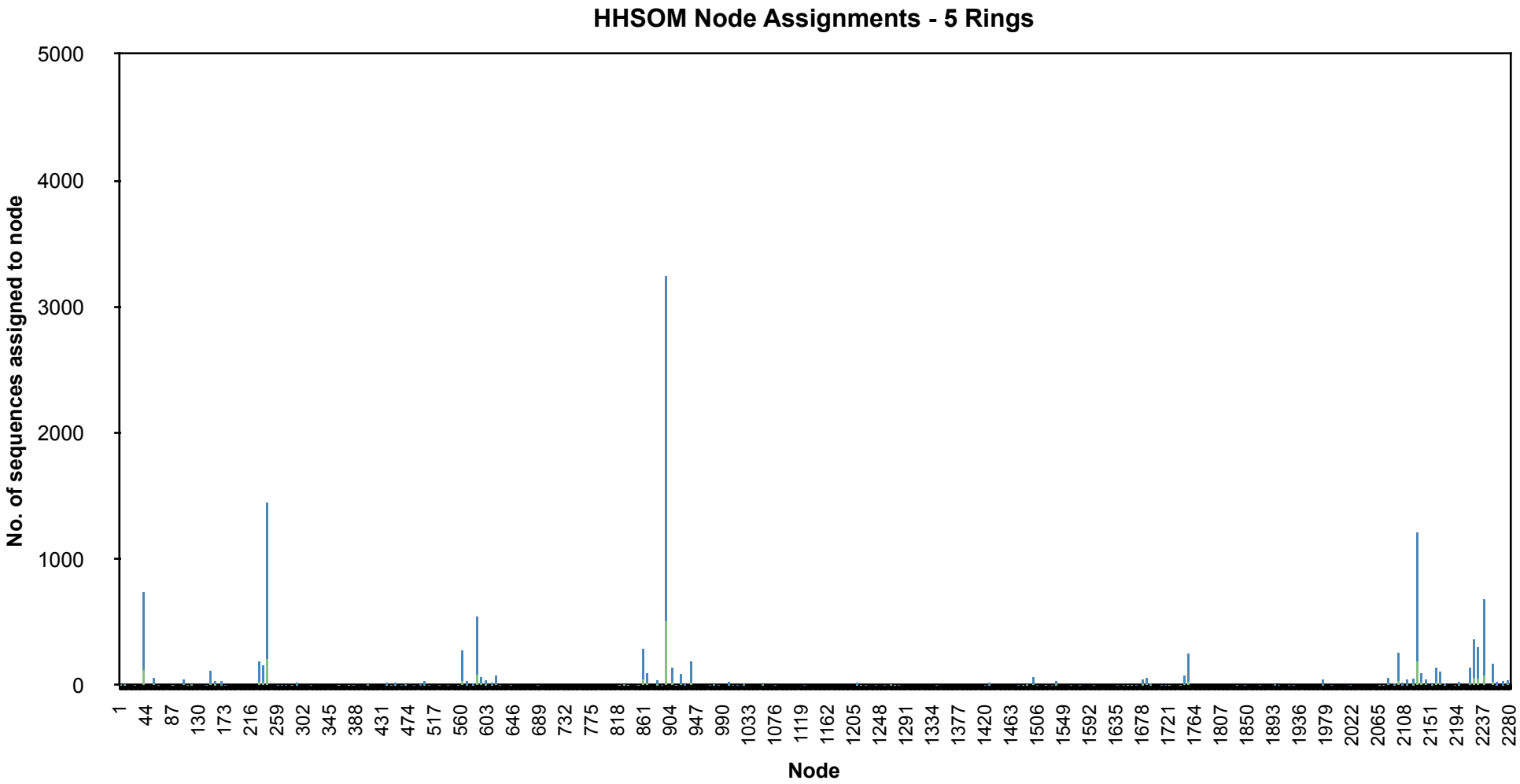

HHSOM Node Assignments - 3 Rings
No. of sequences assigned to node
30000
25000
20000
15000
10000
5000
0
1
6
11
16
21
26
31
36
41
46
51
56
61
66
71
76
81
86
91
96
101
106
111
116
121
126
131
136
141
146
151
156
161
Node

Figure 5.5 *The number of sequencing reads from* A. thaliana *(blue) and* Pseudomonas *sp. 2_1_26 (green) assigned to each node of a 5-ring HHSOM trained with the dataset UT+Psp2126.*


HHSOM Node Assignments - 5 Rings
No. of sequences assigned to node
5000
4000
3000
2000
1000
0
1
44
87
130
173
216
259
302
345
388
431
474
517
560
603
646
689
732
775
818
861
904
947
990
1033
1076
1119
1162
1205
1248
1291
1334
1377
1420
1463
1506
1549
1592
1635
1678
1721
1764
1807
1850
1893
1936
1979
2022
2065
2108
2151
2194
2237
2280
Node

From these results it was clear that, in each case, the proportions of reads from each species assigned to each node of the HHSOM remained consistent with their overall proportions in the dataset. Across the range of grid sizes/number of rings implemented, no enrichment for reads from one particular species was observed in any node or group of nodes.

As originally published, the HHSOM method was intended for use in a supervised or semi-supervised approach, with the HHSOM grid first trained either using complete genome sequences or longer (1-50 kbp) sequence fragments. Shorter sequences were then grouped together onto the nodes of the trained HHSOM, and classified according to the labels applied to the longer, training sequences. An approach similar to this, used with this data, might produce a more successful grouping than was achieved using only the short sequencing reads in the UT+Psp2126 dataset.

Supervised methods are problematic when applied to a dataset of unknown composition. It could be inefficient to train an HHSOM using longer sequences or whole genomes before classifying the sequences in such a dataset, as without specific knowledge of the species that contributed to the data, a wide range of genomes would be required for training. However, if predictions could be made about the composition of the dataset, this range of genomes could be narrowed to provide a training dataset of more manageable proportions. Alternatively, it may be feasible to develop a collection of ‘pre-trained’ maps that have been trained with sequences from common host species and groups of pathogens, with the map used to analyse a dataset chosen based on the host species and type of pathogen thought to be present in the sequenced sample.

The sequences applied to the HHSOM system were also represented differently in the published application of the method, using *tfti* vectors, a variant of the tetra-nucleotide frequency distribution that is designed to accentuate the differences between sequences (Martin, Diaz et al. 2008).

The use of *tf-ti* vectors with the dataset analysed here might also provide an improvement to HHSOM results, and indeed the results obtained from the other methods evaluated. However, the short length of sequencing reads remains a major obstacle to successful clustering.

Comparison of partitioning clustering methods

The performance of three partitioning methods, KM, FCM and CLARA, were evaluated using the UT+Psp2126 dataset, represented by TNF feature vectors. A comparison of the results obtained from each of these methods, clustering the data into two, three and seven clusters, is presented in Tables 5.3, 5.4 and 5.5 respectively.

As disagreement was observed between the two values returned from the cluster validity methods used and with the true number of species known to have contributed directly to the dataset, all three values were used in partitioning clustering of the dataset. This disagreement between cluster validity predictions was an indication that the groups in the dataset were not well-defined within the TNF feature space, an issue that had been made apparent by the imperfect separation observed between reads from each species in clusters previously produced using these features.

Table 5.3 *Precision and recall statistics for clusters produced from UT+Psp2126 using three different partioning clustering methods. The dataset was clustered into two groups. The best Pr and Rc values for each of the two species present in the dataset are highlighted. Two Rc values are highlighted for* A. thaliana *reads, as the same Rc value was recorded in clusters from both* k-*means and* CLARA.

	Method	***k*-Means**		**Fuzzy *c*-Means (fuzziness = 1.1)**		***CLARA***	
Cluster	**Species**	**Pr**	**Rc**	**Pr**	**Rc**	**Pr**	**Rc**
1	***A. thaliana***	0.9916	0.8780	0.9919	0.8753	0.9914	0.8780
	***Pseudomonas* sp. 2_1_26**	0.0084	0.0413	0.0081	0.0401	0.0086	0.0424
2	***A. thaliana***	0.4153	0.1220	0.4203	0.1247	0.4156	0.1220
	***Pseudomonas* sp. 2_1_26**	0.5847	0.9587	0.5797	0.9599	0.5844	0.9576

Table 5.4 *Precision and recall statistics for clusters produced from UT+Psp2126 using three different partioning clustering methods. The dataset was clustered into three groups. The best Pr and Rc values for each of the two species present in the dataset are highlighted.*

		***k*-Means**		**Fuzzy *c*-Means (fuzziness = 1.1)**		***CLARA***	
Cluster	**Species**	**Pr**	**Rc**	**Pr**	**Rc**	**Pr**	**Rc**
1	***A. thaliana***	0.9845	0.6148	0.9834	0.5775	0.9859	0.5486
	***Pseudomonas* sp. 2_1_26**	0.0155	0.0539	0.0166	0.0544	0.0141	0.0438
2	***A. thaliana***	0.3721	0.1001	0.3688	0.0986	0.4102	0.1185
	***Pseudomonas* sp. 2_1_26**	0.6279	0.9431	0.6312	0.9417	0.5898	0.9510
3	***A. thaliana***	0.9981	0.2850	0.9978	0.3239	0.9972	0.3329
	***Pseudomonas* sp. 2_1_26**	0.0019	0.0030	0.0022	0.0039	0.0028	0.0052

Table 5.5 *Precision and recall statistics for clusters produced from UT+Psp2126 using three different partioning clustering methods. The dataset was clustered into seven groups. The best Pr and Rc values for each of the two species present in the dataset are highlighted in yellow. Values for* CLARA *clusters 6 and 7 (highlighted in orange) contained fewer than 20 sequencing reads in total, and were discounted from the search for the highest Pr and Rc values for each species.*

		k-Means		Fuzzy *c*-Means (fuzziness = 1.1)		*CLARA*	
Cluster	**Species**	**Pr**	**Rc**	**Pr**	**Rc**	**Pr**	**Rc**
1	***A. thaliana***	0.9994	0.0952	0.9989	0.1323	0.9730	0.2900
	***Pseudomonas* sp. 2_1_26**	0.0006	0.0003	0.0011	0.0008	0.0270	0.0448
2	***A. thaliana***	0.9958	0.1492	0.9952	0.1875	0.9939	0.3594
	***Pseudomonas* sp. 2_1_26**	0.0042	0.0035	0.0048	0.0051	0.0061	0.0123
3	***A. thaliana***	0.9966	0.2236	0.9973	0.2343	0.3779	0.1024
	***Pseudomonas* sp. 2_1_26**	0.0034	0.0042	0.0027	0.0036	0.6221	0.9409
4	***A. thaliana***	0.9998	0.0624	0.9997	0.0622	0.9989	0.1895
	***Pseudomonas* sp. 2_1_26**	0.0002	0.0001	0.0003	0.0001	0.0011	0.0012
5	***A. thaliana***	0.9934	0.2786	0.9897	0.2019	0.9995	0.0586
	***Pseudomonas* sp. 2_1_26**	0.0066	0.0103	0.0103	0.0118	0.0005	0.0002
6	***A. thaliana***	0.2944	0.0622	0.2936	0.0610	0.2500	0.0000
	***Pseudomonas* sp. 2_1_26**	0.7056	0.8321	0.7064	0.8192	0.7500	0.0006
7	***A. thaliana***	0.8279	0.1288	0.8088	0.1208	1.0000	0.0001
	***Pseudomonas* sp. 2_1_26**	0.1721	0.1495	0.1912	0.1594	0.0000	0.0000

The results indicated that very similar clustering results could be obtained with the use of each of the three partitioning clustering methods and with a range of numbers of clusters. These partitioning methods produced clusters in the data much more successfully than with any of the other, non-partitioning methods investigated here.

With these three methods, grouping the reads into two clusters (Table 5.3) produced one cluster with Rc >95% for *Pseudomonas* reads, and another with Rc of almost 90% for *A. thaliana* reads.

As was observed in CLARA results in previous chapters, the clusters produced here containing the vast majority of *Pseudomonas* reads returned Pr values ~58% with all three methods, suggesting that the reads from the least-represented species in the dataset could be grouped together, but only along with a considerable number of reads from the dominant species. Nevertheless, the extent to which the vast majority of *Pseudomonas* reads were isolated in a single cluster with these methods, with the other containing *A. thaliana* reads almost exclusively, represented a marked improvement on the separation of reads achieved with any other methods investigated here.

The consistent inclusion of a relatively large number of *A. thaliana* reads in clusters containing most of the *Pseudomonas* reads indicated that some overlap existed between the TNF feature profiles of reads from the two species. There may exist multiple populations with different TNF distributions within the reads of both species, and particularly of *A. thaliana*, as eukaryotic genomes are known to exhibit a greater degree of heterogeneity. If this were to be the case, the *A. thaliana* sequences clustered with the bulk of the *Pseudomonas* reads may belong to localised regions within the *A. thaliana* genome.

Grouping the reads into more clusters than the number of species represented in the dataset allowed the investigation of the potential for these methods to isolate *Pseudomonas* reads more effectively. Splitting the data this way allows for clusters to be produced based on groups that may exist within the population of reads from one species, as described. If such an approach was successful, a cluster would be produced with a similar recall value to that observed with two clusters (~95%) for *Pseudomonas* sp. 2_1_26 reads, but with higher precision indicating that a smaller quantity of *A. thaliana* reads had also

been grouped into the cluster. Precision and recall values for three and seven clusters produced using each of the three clustering methods are given in Tables 5.4 and 5.5.

The results in both tables indicated that, while a slight improvement in the precision of grouping *Pseudomonas* reads was obtained, the major effect of increasing the number of clusters generated by KM, FCM and CLARA clustering was to divide the *A. thaliana* reads that had previously been grouped together between the additional clusters. Where the data was separated into three clusters, each of the three methods produced a single cluster with Rc ~94-95% for *Pseudomonas* reads, similarly to grouping into two clusters. These *Pseudomonas* reads accounted for ~59-63% of the cluster in each case, with the highest Rc (CLARA Cluster 2, 95.1%) being associated with the lowest Pr value (58.98%), and *vice versa* (FCM Cluster 2, Pr=94.17%, Rc=63.12%).

With each method, the other two clusters produced predominantly contained *A. thaliana* reads, at a precision of ~98-99%, effectively splitting the *A. thaliana*-rich cluster found when the data was grouped into two clusters.

Where the dataset was grouped into seven clusters, as detailed in Table 5.5, both the KM and FCM methods produced a single cluster with Rc ~82-83% for *Pseudomonas* reads, and Pr ~70%. The remaining bacterial reads were mostly grouped into one other cluster in the results, at a lower precision of ~17-19%.

The similarities observed between clusters produced by KM and FCM were perhaps not surprising, as such a low degree of fuzziness was used in FCM clustering. Excluding the fuzzy approach to grouping the data the two methods are virtually identical, so with fuzziness kept relatively minimal the results obtained from each method could be expected to be similar.

In the results produced with CLARA, two of the clusters produced contained fewer than 20 reads in total (these clusters are highlighted in orange in Table 5.5). As such, the dataset was effectively grouped into five clusters. Within these five clusters, *Pseudomonas* reads were grouped along lines similar to the grouping into two or three clusters, with Rc ~94% in one cluster at Pr ~62%. The remaining reads were divided between the other four clusters, predominantly populated by reads from *A. thaliana*.

In each of these sets of results, an increase in precision of clustering of

Pseudomonas reads was associated with a decrease in recall, and *vice versa*. *A. thaliana* reads were regularly grouped together with very little 'contamination' from bacterial sequences. This trend had been identified previously when comparing feature types for representing the sequences.

The similarity in the pattern of results observed when different numbers of clusters were produced suggested that if these methods were applied to a dataset of unknown composition (i.e. where the exact number of species represented in the data is not known), the choice of the number of clusters is not critical for successful grouping. The number of clusters chosen should be larger or equal to the number of species present, to avoid sequencing reads from multiple species being grouped together unnecessarily. If this approach is taken, a system predicting the origin of reads contained in each cluster must be introduced to allow for the groups of most interest to be identified.

Of all the clustering methods compared here, *k*-means clustering was chosen as the most suitable method for use with TNF feature vectors in further sequence clustering experiments. The clusters obtained from the use of KM were consistently of a good quality relative to the other methods used, and the method itself is easily implemented, very widely known and easily understood, making it the most suitable candidate.

Discussion

The level of accuracy achieved in the clustering described here suggested that it would be difficult to use such methods to fully isolate reads belonging to a pathogen from a sequencing dataset similar to that analysed here. However, the enrichment of particular clusters for reads from each species might improve any further analysis, such as a genome assembly performed on the clustered reads. This could allow the sequencing and characterisation of pathogens that have previously been difficult to study. An investigation into the effect of clustering on the performance of sequence assembly is described in the next chapter.

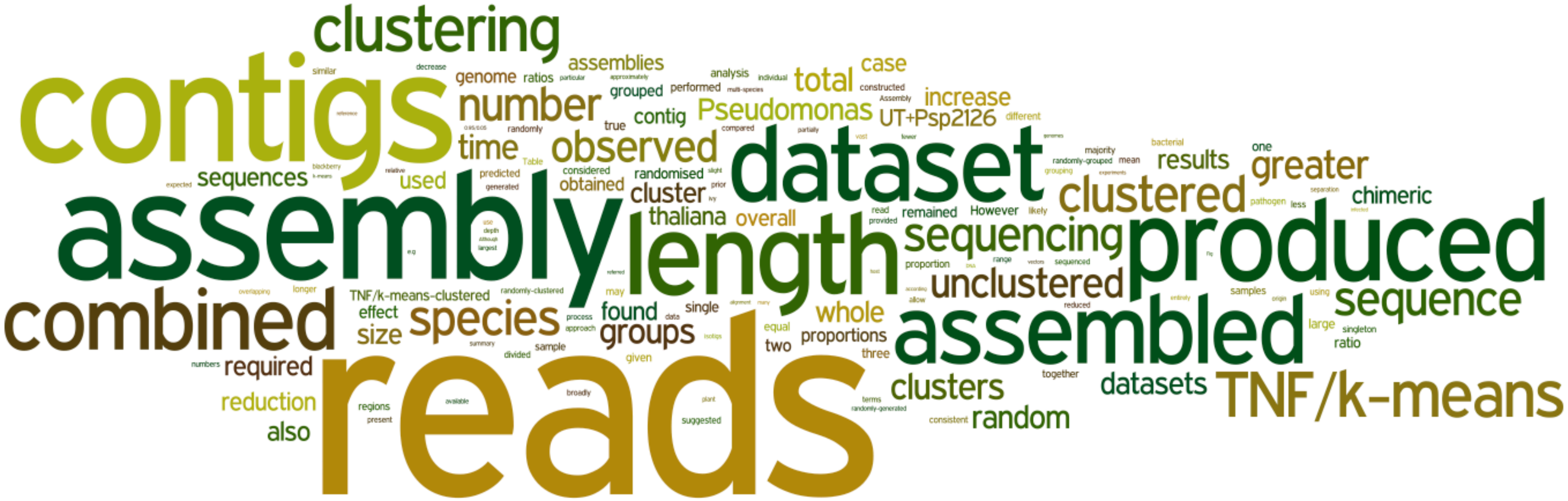
Supervised methods of clustering the data could allow predictions to be made about the identity or phylogeny of species contained in a dataset, as discussed for HHSOM clustering. The limitations associated with using reference sequences/databases to analyse sequencing datasets from environmental samples, especially those containing previously uncharacterised or poorly characterised species, have been discussed extensively previously in this work.

The considerably greater levels of success observed in clustering reads from different species using KM, FCM and CLARA, relative to the near-random grouping of reads observed with other methods investigated here, may be a consequence of the feature selection process described in previous chapters. The TNF feature vectors used here to represent sequences in the comparison of methods were selected based on their clustering performance with a range of sequence datasets, using CLARA to group the data. It is possible that this process introduced a bias favouring the selection of a feature vector type more suited for use with partitioning clustering than other methods.

However, as all clustering methods group and separate data based on measures of similarity between points, a feature type that succeeds in portraying sequences such that those originating from the same species are more similar to each other than to those from another species could be expected to produce the desired grouping of the data using any method suitable for such a comparison.

As with the range of feature vectors compared previously, the list of methods evaluated here is by no means exhaustive. As new methods are introduced,

and increases in computing power allow established methods to overcome the issues associated with large and complex datasets, many more powerful techniques may become available for such clustering analysis. Improvements to the technology used to sequence samples and produce these datasets are also likely to result in an increase in the mean sequencing read length, providing an improvement to results.


reads
assembly
contigs
dataset
length
produced
assembled
combined
clustering
clustered
TNF/k-means
number
observed
sequencing
unclustered
sequence
species
greater
groups
clusters
datasets
random
Pseudomonas
UT+Psp2126
total
increase
assemblies
case
genome
time
sequences
used
obtained
cluster
thaliana
overall
whole
found
size
effect
required
reduction
also
results
chimeric
contig
grouped
randomised
proportions
two
three
However
remained
TNF/k-means-clustered

6

A comparison of *de novo* sequence assembly performance before and after clustering reads according to a prediction of shared origin

Abstract

The effect that the clustering of sequencing reads has on the speed and quality of sequence assembly was investigated. Unguided, de novo *assembly of the simulated dataset UT+Psp2126 and three real sequencing datasets was performed before and after clustering, with groups of reads produced in clustering assembled individually and the results considered together. The data was clustered by the TNF/k-means approach chosen through the work described throughout this thesis, and at random. The results of these assemblies were compared, to establish an understanding of the effect that clustering prior to assembly has on the number of contigs constructed and the total sequence covered by these contigs. The effect of clustering on the time required for assembly was also studied. The results indicated that clustering by TNF/k-means did not adversely affect the contigs produced, and was considerably more effective than randomised clustering in terms of maintaining the assembled coverage of minority genomes in a dataset while minimising the construction of chimeric contigs. Clustering of reads was also found to reduce the time required for assembly of the datasets. This approach may be beneficial when applied to assembly of very large and complex datasets, allowing for a reduction in the time required without cost to the quality of assembly. Individual cluster assemblies may also allow for the isolation and investigation of minority/ pathogen genomes in a multi-species sample.*

Introduction

The previous work in this project was devoted to identifying the optimal combination of sequence features and clustering method for use in the sequence composition-based comparison of DNA sequencing reads produced from a multi-species sample, to cluster these reads according to their species of origin. It was determined that, of the range of features and methods compared, the combination of tetranucleotide relative frequency distribution feature (TNF) vectors and *k*-means clustering produced the most complete separation and grouping of sequencing reads. This combined approach is referred to here as TNF/*k*-means.

Using this methodology, clusters of reads could be produced that were enriched with reads originating from the genome of one particular species over the others represented in the dataset, although the complete separation of reads in such a fashion was not found to be possible.

In this chapter, the effect that the clustering of sequencing reads has on the assembly of contiguous sequences is investigated.

The vast majority of DNA sequencing experiments are performed with the aim of elucidating a stretch of sequence, from a single gene to an entire genome, many times longer than the individual reads produced by current sequencing platforms. In order for these longer sequences to be determined, the short reads produced in the initial sequencing must be combined, a process commonly referred to as *sequence assembly* (Paszkiewicz and Studholme 2010).

There exist many different software packages that can be used for assembly, each generally intended for use with reads obtained from different sequencing platforms, or targeted at experiments with specific aims. These different assembly packages are too numerous to be listed here, but a good, albeit slightly outdated, collection of many of the different options is available at (http://genome.ku.dk/resources/assembly/methods.html, Scheibye-Alsing, Hoffmann et al. 2009). A detailed review of all existing methods is outside the scope of this work, but a general methodology is shared by the majority of assembly packages. What follows is a summary of this common central process.

Sequence assembly is based on the principle of multiple sequence alignment,

where a pairwise comparison is made between every sequencing read in a dataset, to identify overlapping regions of homologous sequence between reads. These overlapping stretches are taken to indicate where the sequence from one read flows into that of the other, and the two reads can be joined together by this overlap. This can happen many times during the assembly process, with more reads being assimilated into these longer sequences according to the overlapping regions found, until all reads have been compared. The longer sequences produced in this process are known as contiguous sequences, or *contigs*.

This contig assembly can be performed using a reference genome as a scaffold, with reads and contigs being aligned to this scaffold. However, this approach requires the availability of a genome sequence from the same species, or another suitably homologous genome, to act as an appropriate scaffold (e.g. Wheeler, Srinivasan et al. 2008).

If no such reference is available, as is the case when sequencing material from the majority of species, a *de novo* assembly of reads must be performed using only the overlaps found between reads (e.g. Chaisson and Pevzner 2008; Paszkiewicz and Studholme 2010; Simpson and Durbin 2012). In such a case, the successful assembly of all or the vast majority of a whole genome relies on the depth of the sequencing carried out on a sample providing sufficient overlapping regions throughout the whole genome. The depth of sequencing is the average number of times that a given point in the target has been sequenced during the experiment. The more sequences are produced from a sample, the greater this depth will be.

The existence of large regions of repetitive and/or palindromic sequence in eukaryotic genomes complicates the assembly process, with reads originating from these regions forming loops or forks in the alignment, which must be resolved before a single consensus sequence can be reached. The presence of repetitive regions of sequence is particularly problematic for assembly when the length of these regions exceeds that of the read lengths of the sequencing platform. If a read cannot span the full length of a repeating section of the genome, the true length of this section cannot be resolved just by assembling together the reads obatined from each end. As a result, repetitive and duplicate reads tend to be removed from the analysis prior to assembly (e.g. Wang, Wong

et al. 2002), as are any sequences that are too short to contribute effectively.

Overlaps between reads are generally identified from sections of identical sequence of a given length, known as ‘seeds’, from which longer overlaps can be established. Information on the quality of sequencing reads may also be used in the assembly, to aid the judgement of a good alignment and the determination of a consensus sequence between aligned reads in a contig.

Because assembly requires every sequence in a dataset to be compared with every other sequence so that overlapping regions can be identified, the time required for assembly scales poorly as the number of reads in the dataset, and can become very large for datasets generated at the depth required for effective genome sequencing.

Where a dataset is assembled that contains sequences from more than one species, as is the case in the investigations described here, it is possible for contigs to be erroneously constructed from reads obtained from two or more different species. These contigs are referred to here as *chimeric*, and are more likely to be assembled if a large degree of identity exists between the genomes of the species sequenced in the dataset. This should be of minimal concern in the work described here due to the highly-divergent nature of the organisms that contributed to the datasets used.

The enrichment within each group of reads for sequences from one particular species, achieved by TNF/*k*-means, may be beneficial in the study of multi-species samples. Such clustering might allow contigs assembled from reads from one particular species to be isolated in the assembly results of a single cluster, and is predicted to be coupled with a decrease in time required for the dataset to be assembled.

Any separation of a dataset will reduce the time required to compute an assembly. However, arbitrarily separating reads in a dataset in order to reduce computation time is likely to be detrimental to the quality of assembly produced from the dataset as a whole, since reads that might otherwise have been combined into contigs become separated into different groups.

This risk is likely to be reduced where the reads in a dataset can be separated according to their species of origin, as reads that would used in assembly of a single contig are more likely to be grouped together than if the separation was

performed at random. This also improves the likelihood of constructing large portions of genomic sequence for specific species from the reads clustered together in each group. If coupled with a system for predicting the likely contents of each cluster (in terms of the species from which reads originated in that cluster), this could allow for isolation and increased simplicity of the study of the genome of particular, minority genomes in a multi-species or metagenomic dataset.

It is feasible that such a separation could also reduce the risk of erroneous assembly of chimeric contigs, combining reads produced from the genomes of two different species. However, given the short length of the reads of interest here, and the sequence similarity-dependent nature of the assembly process, it appears probable that two reads that would be combined in the production of a contig would also be grouped together during clustering of the dataset.

Here, the software package provided for use with the 454 GS FLX sequencing platform, based around Newbler (gsAssembler, *Roche*/*454 Life Sciences*, CT, USA) was used for sequencing read assembly. This package was chosen as it is widely-used and the datasets studied here were all produced on the *454* sequencing platform.

Several datasets of sequencing reads, both simulated and obtained from real samples, were assembled before and after *k*-means clustering of TNF vectors generated from the reads.

The UT+Psp2126 dataset introduced in Chapter 4, consisting of reads from separate experiments sequencing *A. thaliana* and *Pseudomonas* sp. 2_1_26 combined in a ratio of ~5:1, was used as a basis for species-specific investigation of the number and length of contigs, with three true sequencing datasets obtained from samples of infected plant tissue. Data was used from blackberry (*Rubus fruticosus*) predicted to be infected with a bacterial pathogen, ivy (genus *Hedera*) predicted to be infected with both a bacterial and a fungal pathogen, and tomato (*Lycopersicum solanum*) predicted to be infected with *Pepino mosaic virus* (PepMV).

Summary statistics for these datasets are provided in Table 6.1.

Table 6.1 *Summary statistics of datasets used in assembly and, where possible, the genomes that contributed to these datasets. Statistics could not be provided from unknown pathogens predicted to be present.*
1. Complete genome sequence of A. thaliana *(assembly TAIR1).*
2. Incomplete genome of Pseudomonas *sp. 2_1_26 (draft v1 from MiST 2.1 (Ulrich and Zhulin 2010)).*
3. Very little Rubus fruticosus *sequence data is publicly available. As such, all sequences available in the NCBI Nucleotide database were used to provide an estimate of the G/C content of the genome (17 sequences, 12518 bp total).*
4. As with R. fruticosus*, very little sequence data is available for organisms in genus* Hedera*. Sequences available in the NCBI Nucleotide database for* Hedera helix *(English Ivy) were used to provide an estimate of G/C content of the genome (223 sequences, 190798 bp).*
5. Complete genome sequence of Solanum lycopersicum *(assembly SL2.40).*
6. Complete genome sequence of Pepino Mosaic Virus *(NCBI accession no. NC_004067).*
**statistic derived from an incomplete genome.*
***statistic derived from a limited number of available nucleotide sequences from this species.*

Dataset	Organism	Genome Size	Genome GC content	Reads in dataset	Total size of dataset / mean read length bp	Dataset GC content (st. dev)	Dataset mean Phred30 Score (st. dev)
UT+Psp2126	***A. thaliana*** [1]	119.67 Mbp	0.36	125339	27.96 Mbp / 223 bp (~23.7 Mbp *A. thaliana*; ~4.25 Mbp *Pseudomonas*)	0.4488 (0.1348)	0.6478 (0.1735)
	***Pseudomonas* sp. 2_1_26** [2]	6.3 Mbp*	0.66*				0.8237 (0.2116)
1	***Rubus fruticosus*** [3]	*n/a*	0.41**	111531	46.46 Mbp / 417 bp	0.5392 (0.0463)	0.7066 (0.2009)
	bacterial pathogen	*n/a*	*n/a*				
2	***Hedera*** [4]	*n/a*	0.37**	22733	5.22 Mbp / 230 bp	0.5162 (0.0482)	0.6485 (0.2139)
	bacterial pathogen	*n/a*	*n/a*				
	fungal pathogen	*n/a*	*n/a*				
3	***S. lycopersicum*** [5]	781.35 Mbp	0.35	65691	16.59 Mbp / 253 bp	0.5018 (0.0837)	0.8828 (0.1386)
	PepMV [6]	6.45 kbp	0.41				

Materials and Methods

Clustering of reads

Tetranucleotide relative frequency distribution vectors, generated from reads as described in Chapter 2, were clustered by *k*-means, using the Perl script 'partClustering.pl' reproduced in Appendix A.

Randomly generated clusters were produced by assigning to each read a pseudorandom integer between 1 and *k*, where *k* was equal to the desired number of clusters. These pseudorandom labels were then used to determine the reads belonging in each cluster.

Contig assembly

Sequencing reads in FASTA format, with accompanying .qual files, were assembled with Newbler (GSAssembler v2.6, *Roche*/*454 Life Sciences*, CT, USA). Files for samples 1 (blackberry), 2 (ivy), and 3 (tomato/PepMV) were kindly provided by the Plant Pathology (formerly Novel Methods) Group, Food and Environment Research Agency (FERA, Sand Hutton, York, UK). The names by which these datasets are referred to here are an indication of the host and, in the case of tomato/PepMV, pathogen species predicted by blastn/blastx searching of NCBI Genbank to be present in the sequenced samples (predicted presence of PepMV based on personal correspondence).

Analysis of assembly results

Contigs produced and the reads used by the assembly were summarised using the Perl script 'contigInfo.pl', reproduced in Appendix A. This script operates on the '454Contigs.ace' and '454ReadStatus.txt' files produced in the results of a genomic DNA assembly.

Where isotigs were produced from assembly of cDNA, 'contigInfo.pl' was used to summarise the reads used and isotigs produced, operating on the file '454Isotigs.ace' instead of the equivalent '454Contigs.ace'. A summary of contigs was obtained from the file '454ContigGraph.txt' also produced in the results of an assembly. In correspondence with the contigs detailed in the '454Contigs.ace' file for a genomic DNA assembly, only those contigs ≥100 bp in length were summarised here for cDNA assemblies.

Results

UT+Psp2126

The UT+Psp2126 dataset was assembled before and after clustering into two groups of reads, by TNF/*k*-means analysis and by random assignment. Where reads were assigned at random, one set of groups was produced in approximately equal proportion (i.e. in proportions of 0.5 of the whole dataset), and a second set grouped into the same proportions as those generated by TNF/*k*-means.

These randomly generated clusters of reads were produced to allow any effects on assembly that were specific to TNF/*k*-means clustering to be distinguished from those caused by the difference in the number of reads available for assembly when clusters were grouped separately. Clusters produced at random in equal proportion were included to provide an impression of the assembly results that could be expected if the dataset were divided in such a way as to provide the maximum reduction in the number of reads considered for assembly in each analysis and thus provide the greatest reduction in time required for assembly.

After clusters had been produced, these sets of reads were assembled individually. A comparison of the contigs produced is given in Table 6.2.

The statistics presented in Table 6.2 were intended to provide an indication of the overall sequence correctly assembled into contigs in each case. The combined total length of all contigs produced in assembly is given, as well as the combined total length of contigs constructed entirely from reads from a single species - *A. thaliana* and *Pseudomonas* - and chimeric contigs assembled from a mixture of reads from these two species. The combined length of all contigs produced from reads derived from a single species (labelled *NCCL* in Table 6.2) provided a measure of the total sequence that was assumed to be correctly assembled from the dataset. Counts are also provided of the number of individual reads assembled and partially assembled, and of the number of singleton reads, that could not be assembled into longer stretches of sequence.

Table 6.2 *Details of contigs produced from assembly of UT+Psp2126 dataset as a whole, and separated into two clusters. The combined number of reads assembled and partially assembled (APA), and the number of singleton reads left unassembled are given. A breakdown is included, of contigs constructed exclusively from reads obtained from sequencing of* A. thaliana *(marked 'All* A. thaliana*'),* Pseudomonas *sp. 2_1_26 (marked 'All* Pseudomonas*'), and of those constructed from a combination of reads obtained from both species (marked 'Chimeric'). A summary of contigs produced from two randomly-generated clusters of reads is also included. The total sequence covered by these contigs is included, as is the combined length of all non-chimeric contigs (NCCL), which provides a measure of the total sequence successfully assembled.*

	Un-clustered	Clustered with TNF/*k*-means			Clustered at random with same proportions as TNF/*k*-means groups			Clustered at random with ~1/2 reads in each group		
Metric	**All reads**	**Cluster 1**	**Cluster 2**	**Total**	**Cluster 1**	**Cluster 2**	**Total**	**Cluster 1**	**Cluster 2**	**Total**
Contigs	604	117	504	621	142	346	488	191	196	387
Combined length (bp)	407408	56997	355467	412464	138765	300003	438768	205203	209681	414884
Mean length (bp)	675	487	705	664	1028	817	899	1074	1070	1072
APA	27683	4167	23345	27512	6021	19955	25976	12670	12258	24928
Singletons	91072	24938	65907	90845	23872	69450	93322	47279	47396	94675
All *A. thaliana*	*493*	*23*	*487*	*510*	*131*	*319*	*450*	*181*	*187*	*368*
Combined length (bp)	*359216*	*18213*	*348320*	*366533*	*110275*	*251644*	*361919*	*170741*	*188178*	*358919*
All *Pseudomonas*	107	92	16	108	3	42	45	9	5	14
Combined length (bp)	46130	38227	6936	45163	638	17336	17974	3685	1679	5364
Chimeric	*4*	*2*	*1*	*3*	*1*	*6*	*7*	*1*	*4*	*5*
Combined length (bp)	*2062*	*557*	*211*	*768*	*27852*	*31023*	*58875*	*30777*	*19824*	*50601*
NCCL (bp)	405346	56440	355256	411696	110913	268980	379893	174426	189857	364283

Where the UT+Psp2126 dataset was assembled without first being split into clusters of reads, contigs were produced from *A. thaliana* and *Pseudomonas* reads in proportions roughly equal to those of the sequencing reads in the dataset. A large proportion of reads (72.66%) were left unassembled, and four chimeric contigs were produced accounting for a total of ~2 kbp of sequence.

Where assembly was performed on reads clustered by TNF/*k*-means, 621 contigs were produced in total between both groups of reads, a slight increase on the 604 produced when the dataset was assembled as a whole. These contigs were shorter on average than those assembled when the dataset was considered as a whole, but the total combined length of these contigs was ~5 kbp longer.

In total, 17 more *A. thaliana* contigs were assembed from reads clustered by TNF/*k*-means, with a slight increase (~7 kbp) observed in the total length of these contigs. The combined length of *Pseudomonas* contigs assembled from this grouping was ~1 kbp less than from UT+Psp2126 considered as a whole, with one more contig produced from TNF/*k*-means-clustered reads.

Assembly after TNF/*k*-means clustering also resulted in the construction of one fewer chimeric contigs, with the 3 such sequences produced constituting only 768 bp in total.

Assembly of reads clustered at random in the same proportions as the groups produced by TNF/*k*-means produced fewer contigs overall. However, the combined length of the 488 contigs produced in total from these groups was ~31.3 kbp greater than that of those assembled from the dataset considered as a whole, and ~26.3 kbp greater than those produced from TNF/*k*-means-clustered reads.

The combined length of contigs produced entirely from *A. thaliana* reads randomly grouped in this way was ~2.7 kbp greater than from those assembled from the whole dataset considered together, but ~4.6 kbp less than from those produced from assembly of reads grouped by TNF/*k*-means. A considerable reduction was observed in the combined length of *Pseudomonas* contigs produced in this case, with ~18 kbp of sequence assembled in total. This constituted a reduction of >60% in the combined length produced, relative to where reads were left unclustered, or clustered by TNF/*k*-means.

As such, the increase in total length of contigs observed in assembly of these randomly grouped reads, compared with that of reads clustered by TNF/*k*-means, could be almost entirely attributed to the construction of 7 chimeric contigs at a total combined length of 58,875 bp.

Similar results were observed from the assembly of reads divided at random into equally sized groups. In this case, a smaller number of contigs were produced overall (387), but the combined length of these contigs was once again observed to be greater than in the case of unclustered reads (an increase of ~7.4 kbp) and reads clustered by TNF/*k*-means (~2.4 kbp).

A reduction was observed in the number of *A. thaliana* (368) and *Pseudomonas* contigs (14) produced. In this case, the combined length of these groups of contigs was found to be shorter than of the equivalent sets assembled from unclustered and TNF/*k*-means-clustered reads. Once again, the greater overall combined contig length of these assemblies was accounted for by the size of the chimeric contigs produced. Between these two randomly produced groups of reads, 5 chimeric contigs were assembled with a combined length of 50,601 bp.

In all of these assemblies, both clustered and unclustered, the numbers of reads assembled and left unassembled remained broadly consistent. Marginally fewer reads were assembled or partially assembled where the dataset was grouped by TNF/*k*-means (a reduction of 171), while randomised grouping was associated with a decrease of 1,707 reads assembled/partially assembled in the case of groups proportional with those generated by TNF/*k*-means and 2,755 in the case of reads grouped in equal proportion. Conversely, the number of singleton reads increased by similar margins in each case.

The total length of sequencing reads in the dataset that were produced from *A. thaliana* was ~23.7 Mbp, approximately 19.8% of the total length of the genome. The 19,045 reads from sequencing of *Pseudomonas* sp. 2_1_26 constituted ~4.25 Mbp in total, or approximately 67.5% of the predicted length of the genome (from draft v1 of the genome, see Table 6.1).

According to the Lander-Waterman theory for predicting coverage in an assembly project (Lander and Waterman 1988), the expected number of individual mapped regions along a genome can be modelled as a function of

the number of reads, the read and genome length and the length of overlap required to join two reads together to form a contig. This approach is designed for single genome assembly projects, and several variations have been proposed including those for metagenomic applications (Hooper, Dalevi et al. 2010) and assembly of paired-end reads (Wendl 2006). Although the UT +Psp2126 dataset contains sequences obtained from more than one species, the Lander-Waterman theory should still be applicable to the two constituent sets of reads, considered separately. Such an approach would not be possible where the numbers of reads and length of genome were not known for the individual organisms that contributed to the dataset.

Following the Lander-Waterman theory as described in (Lander and Waterman 1988), the total number of mapped regions (contigs + mapped singletons) on the genome of *A. thaliana* was predicted to be 90,348. Of these, the predicted number of contigs (those mapped regions containing multiple, overlapping reads) was 13,554, leaving 76,794 predicted singleton reads that would map to the genome of *A. thaliana*. The total length of mapped regions on the genome was predicted to be ~21.6 Mbp. Assuming that the mean length of those singleton reads predicted was equal to that of the all the reads in the dataset, 223 bp, the predicted total length of singleton reads was (76,794 * 223) = ~17.1 Mbp. After this length of predicted singletons mapped is subtracted from the total predicted mapped length, the predicted total length of assembled contigs from *A. thaliana* by this method was (21.6 - 17.1) = ~4.5 Mbp.

The same calculations for *Pseudomonas* reads yielded a prediction of 4654 contigs and 6299 singletons. *Pseudomonas* contigs were predicted to have a combined length of ~1.7 Mbp.

In both cases, the observed amount of sequence assembled from reads of each genome was considerably smaller than predicted by the Lander-Waterman method. The numbers of contigs assembled were also markedly fewer than predicted. This discrepancy remained across all assemblies of the data, regardless of clustering conditions.

In the case of sequencing reads obtained from *A. thaliana*, analysis by alignment to a reference database containing the full genome (see Chapter 3) had already suggested that not all of the reads could be assigned to the plant.

As such, the predictions of mapping all of these reads were likely to over-estimate the level of coverage that could be achieved. A similar investigation was not undertaken for *Pseudomonas* sp. 2_1_26 reads, which were assumed to be completely derived from the bacterial genome.

The significant overestimation of assembly could also be a result of the presence of many reads originating from the same or repetitive regions of the genome. If this were the case, depth of coverage would not be uniformly spread across the genome (as assumed in the Lander-Waterman model), but instead would be concentrated at points of high coverage.

Without further investigation into the actual genomic mapping of all reads (rather than *de novo* assembly investigated here), the reasons for this inconsistency between expected and observed sequence assembly can only be speculated upon.

Unfortunately, the information required for predicting coverage and mapping by this method (total reads from each species and genome lengths) were not available for the other datasets considered here, so similar estimations of assembly were not made.

In order to better observe the effect of randomly clustering the dataset, and to distinguish the effect of TNF/*k*-means clustering compared to this randomised division of reads, UT+Psp2126 was assembled in randomly generated groups across a range of size ratios. At intervals of 0.05 of the total number of reads in the dataset, sets of two clusters were produced at random from UT+Psp2126 and assembled. The results of these assemblies, combined between the two clusters, are plotted in Figures 6.1 and 6.2, across the range of size ratios produced. Also plotted in the figures are statistics obtained from assembly of reads clustered by TNF/*k*-means, at the appropriate distance along the axis of size ratios.

Figure 6.1 *The effect of increasing disproportionality of split between clusters on assembly of UT+Psp2126. The combined length of contigs is plotted, for the unclustered dataset, and for reads derived from the* A. thaliana *genome only, assembled from reads clustered at random into two groups. The total non-chimeric contig length (NCCL) is also given. The statistics are plotted against the proportion of reads contained in the largest cluster. Values are included for assembly of clusters produced randomly in proportions of 0.95/0.05, 0.9/0.1, 0.85/0.15, ..., 0.55/0.45, 0.5/0.5, and for assembly of the dataset as a whole. The equivalent statistics are included for assembly of reads clustered by the TNF/k-means approach, a the appropriate point along the x-axis Error bars are included for assembly of randomly-generated clusters, given as standard error from the mean for three repeats of clustering the dataset at random.*

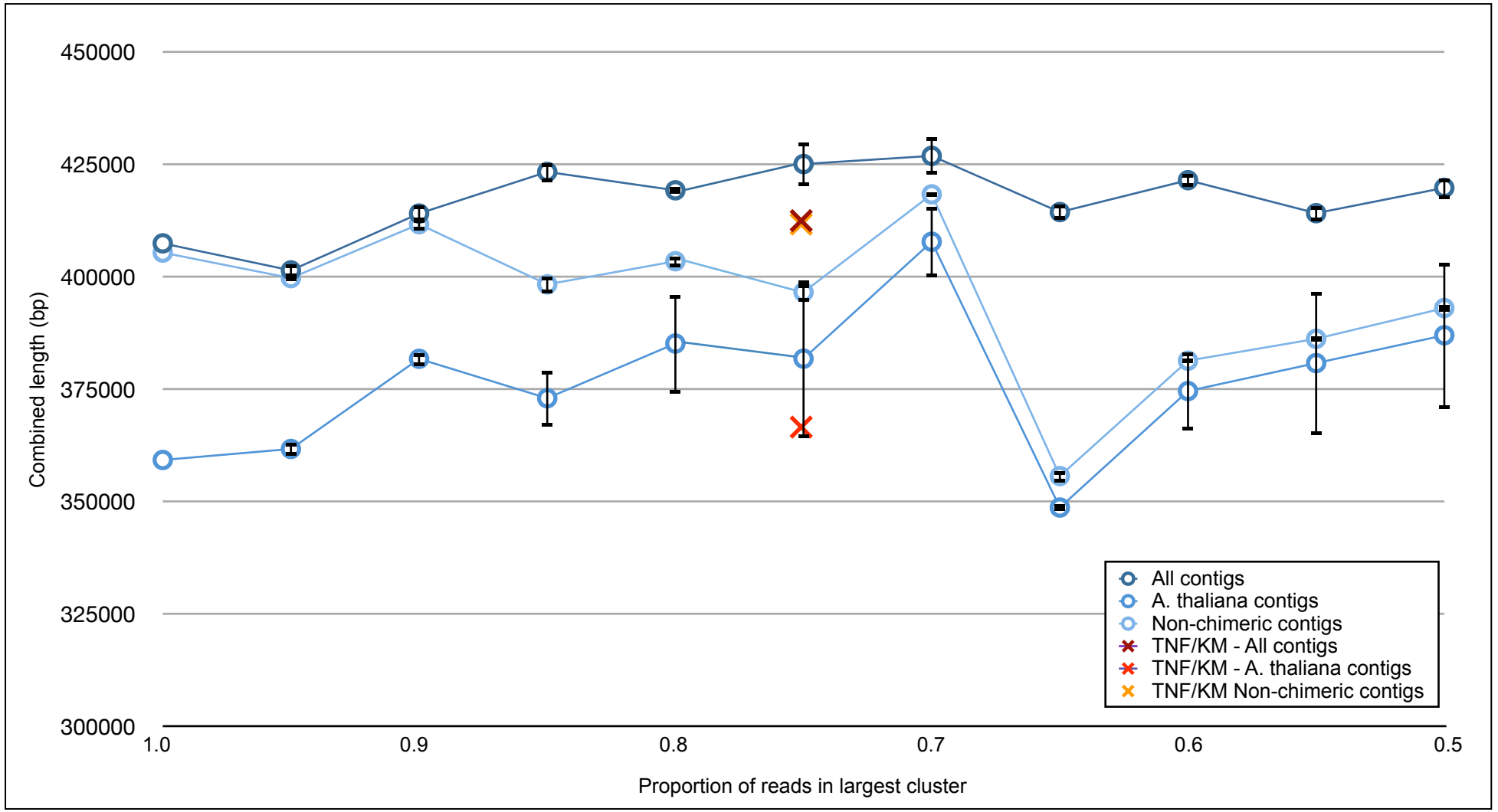
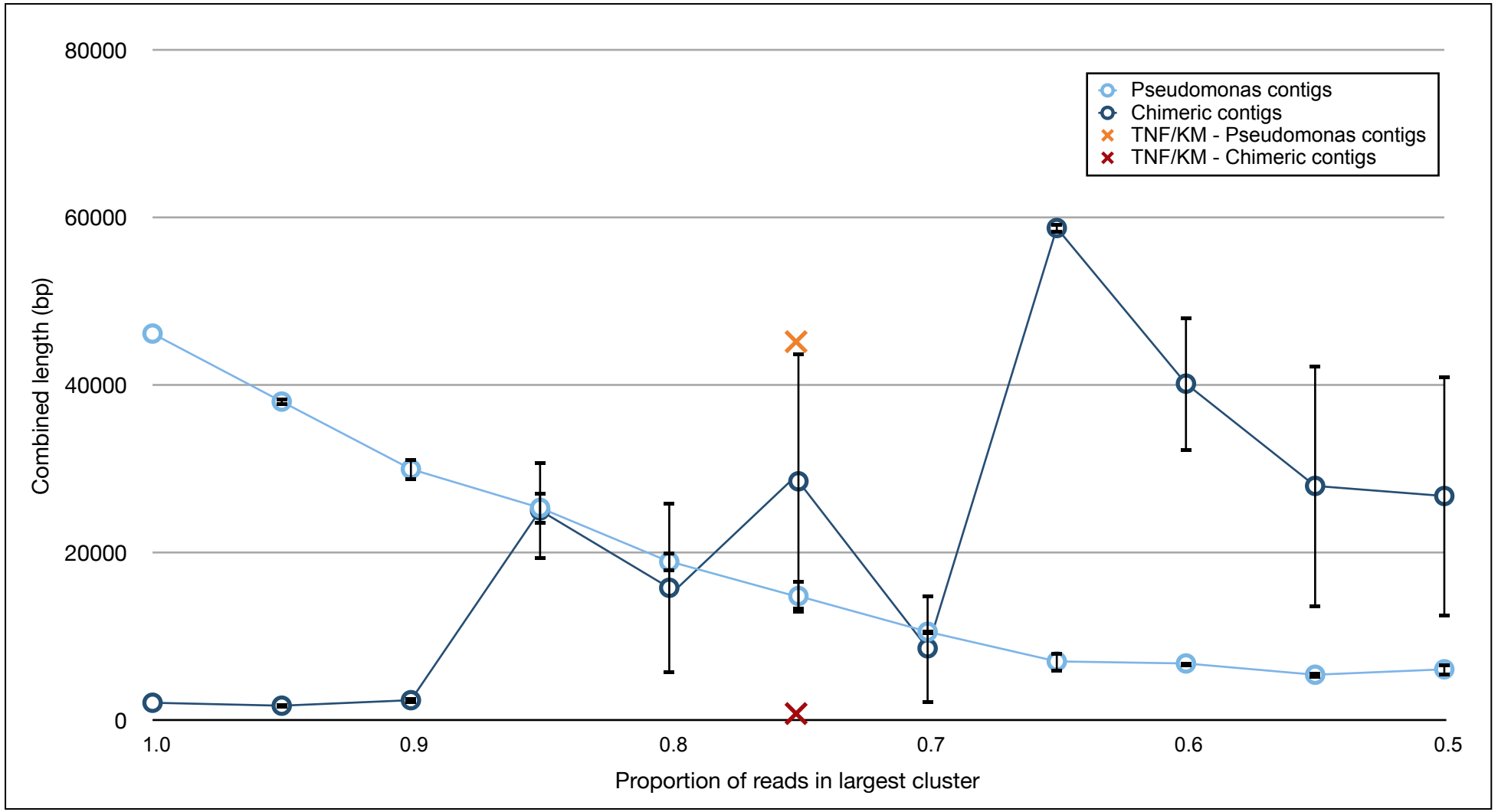

450000
425000
400000
375000
350000
325000
300000
Combined length (bp)
1.0
0.9
0.8
0.7
0.6
0.5
Proportion of reads in largest cluster
All contigs
A. thaliana contigs
Non-chimeric contigs
TNF/KM - All contigs
TNF/KM - A. thaliana contigs
TNF/KM Non-chimeric contigs

Figure 6.2 *The effect of increasing disproportionality of split between clusters on assembly of UT+Psp2126. The combined length of contigs from* Pseudomonas *reads only, and of chimeric contigs, produced from reads from both species, are given as assembled from reads clustered at random into two groups. The statistics are plotted against the proportion of reads contained in the largest cluster. Values are included for assembly of clusters produced randomly in proportions of 0.95/0.05, 0.9/0.1, 0.85/0.15, ..., 0.55/0.45, 0.5/0.5, and for assembly of the dataset as a whole. The equivalent statistics are included for assembly of reads clustered by the TNF/k-means approach, a the appropriate point along the x-axis. Error bars are included for assembly of randomly-generated clusters, given as standard error from the mean for three repeats of clustering the dataset.*


80000
60000
40000
20000
0
Combined length (bp)
1.0
0.9
0.8
0.7
0.6
0.5
Proportion of reads in largest cluster
Pseudomonas contigs
Chimeric contigs
TNF/KM - Pseudomonas contigs
TNF/KM - Chimeric contigs

Across the range of size ratios, with one exception at 0.95/0.05, the combined length of all contigs was found to be greater from assembly of randomly clustered reads than from assembly of the unclustered dataset. This increase reached a peak at a size ratio of 0.7/0.3, where the combined length of all contigs assembled was ~25 kbp greater than from the unclustered data.

The combined length of *A. thaliana* contigs was found to increase where the dataset was randomly grouped at size ratios from 0.95/0.05 to 0.7/0.3, relative to those assembled from the unclustered data. A considerable decrease was observed between a ratio of 0.7/0.3 and 0.65/0.35, at which point the combined length of *A. thaliana* contigs fell below that assembled from unclustered reads. The combined length was found to increase steadily from this point, for assembly of clusters of proportions from 0.65/0.35 to 0.5/0.5.

In contrast to this variation in combined length of *A. thaliana* contigs around that obtained from unclustered reads, the combined length of contigs assembled entirely from *Pseudomonas* reads was found to decrease steadily as the proportionality of randomised clustering was increased (see Fig. 6.2). *Pseudomonas* contigs assembled from unclustered reads formed a total length of ~46 kbp. This combined length was found to decrease steadily with each cluster size ratio, falling to ~5 kbp where the dataset was divided in equal proportions of 0.5/0.5.

Conversely, the combined length of chimeric contigs produced in assembly was found to increase from a relatively negligible 2 kbp for unclustered reads, peaking at ~59 kbp at a size ratio of 0.65/0.35 before decreasing again for the remaining randomised clustering ratios. This combined length of chimeric contigs exhibited a greater degree of variation between repeats of random clustering than the other statistics used to describe the assemblies.

The variation in combined chimeric contig length was reflected in the total combined length of non-chimeric contigs (Fig. 6.1), which remained close to the overall combined length of all contigs from assembly of unclustered reads and reads clustered randomly at a size ratio of 0.95/0.05 and 0.9/0.1, but thereafter failed to increase alongside this overall measure of combined contig length. Where the length of chimeric contigs was removed, the combined length of assembled sequences was found to remain broadly consistent with that of

unclustered reads, for random cluster size ratios from 0.95/0.05 to 0.7/0.3, before decreasing by ~50 kbp at 0.65/0.35, as was observed in the combined length of *A. thaliana* contigs. In fact, these length statistics were very similar at these size ratios, because the combined length of *Pseudomonas* contigs constituted a very small proportion of the overall length at these ratios.

The combined length of all contigs produced from reads clustered by TNF/*k*-means was found to be ~12.4 kbp less than that observed with reads clustered randomly at a ratio of 0.75/0.25 (Fig. 6.1). This randomised clustering ratio interval was extremely close to the 0.7501/0.2499 split of reads produced by TNF/*k*-means clustering.

Although the overall combined contig length from this TNF/*k*-means-clustered assembly was smaller than that observed from randomly clustered reads, the combined length of chimeric contigs was considerably smaller (Fig. 6.2), giving a greater combined length of non-chimeric contigs in assembly of these clusters than in the equivalent randomly clustered reads. The combined length of *Pseudomonas* contigs was also observed to be markedly larger than for randomly divided reads, and only slightly reduced (by ~1 kbp) relative to those assembled when the dataset was considered as a whole.

Where reads were clustered by TNF/*k*-means, some distinction could be made between the assemblies performed on the individual clusters. The vast majority of *A. thaliana* sequence assembled from the dataset under this clustering (348,320 bp, or 95.03%) was obtained from a single cluster. The same effect was observed for assembled *Pseudomonas* sequence, where 38,227 bp (84.64%) was produced from reads contained in the opposing cluster from this large fraction of plant contig length. This separation of contigs was not observed where the dataset was divided at random. In these cases, the proportions of *A. thaliana* and *Pseudomonas* sequence assembled from each cluster remained broadly consistent with the overall proportions of reads contained in each group, with little differentiation between the combined length of contigs from each species.

The combined length of *A. thaliana* contigs assembled from reads clustered by TNF/*k*-means was ~15 kbp less than from randomly clustered reads at a size ratio of 0.75/0.25.

The marked decrease in combined length of *A. thaliana* contigs and the associated increase in chimeric contigs visible where UT+Psp2126 was randomly grouped at a size ratio of 0.65/0.35, compared to the assembly at a ratio of 0.7/0.3, was difficult to explain. It may be that the reduction in the size of the largest of the two groups produced fell below some threshold value for the mean coverage of *A. thaliana* provided by these reads, resulting in a sudden fall in the total sequence that could be produced for this genome. The subsequent rise in combined length of *A. thaliana* contigs as the ratio of cluster sizes approached parity could then be explained as a result of increasing mean coverage of the plant genome in the smallest cluster. It is difficult to draw any firm conclusions on this without further investigation into the specific coverage within the dataset, and the nature of the contigs being produced.

Sample 1 - blackberry + suspected bacterial pathogen

The blackberry dataset of 111,531 reads was assembled as a whole, after TNF/*k*-means clustering, and after randomised division of reads into two groups, as before. The dataset was clustered into two groups, as this was the number of species predicted to be present in the sequenced sample - the blackberry plant host and a bacterial pathogen. A summary of the contigs produced in these assemblies is given in Table 6.3.

A larger number of contigs were assembled in total from reads clustered by TNF/*k*-means (143), and clustered at random in proportions equal to those obtained from TNF/*k*-means (136) and in a ratio of 0.5/0.5 (127), than were constructed from the unclustered dataset (119 contigs). Clustered reads were also assembled to cover a greater total sequence across these contigs, with TNF/*k*-means-clustered contigs returning a combined length of ~110 kbp, randomly generated clusters of the same proportions as these producing clusters of total length ~138.5 kbp and equally proportioned randomly divided clusters ~136 kbp, compared to a combined length of ~94.5 kbp in contigs assembled from unclustered reads.

The combined numbers of assembled and partially assembled were similar for each assembly, with ~102,000 reads used in each case. The randomly grouped assemblies returned a greater number of singleton reads than in the assembly of unclustered reads and reads clustered by TNF/*k*-means.

Table 6.3 *Details of contigs produced from* de novo *assembly of the blackberry dataset as a whole, and separated into two clusters. The number of reads assembled, the combined number of reads assembled and partially assembled (APA) and the number of singleton reads are given. Statistics are included for reads clustered at random and with TNF/*k-means.

	Un-clustered	Clustered with TNF/*k*-means			Clustered at random with same proportions as TNF/*k*-means groups			Clustered at random with ~1/2 reads in each group		
Metric	**All reads**	**Cluster 1**	**Cluster 2**	**Total**	**Cluster 1**	**Cluster 2**	**Total**	**Cluster 1**	**Cluster 2**	**Total**
Contigs	119	103	40	143	92	44	136	60	67	127
Combined length (bp)	94686	88338	22203	110541	86845	51692	138537	66113	70024	136137
Mean length (bp)	796	858	555	773	944	1175	1019	1102	1045	1072
APA	102574	75571	27227	102798	74639	27211	101850	51038	50999	102037
Singletons	6224	4225	1983	6208	5094	2207	7301	3521	3543	7064

Sample 2 - ivy + supected bacterial and fungal pathogens

The ivy dataset of 22,733 reads was assembled as a whole, and after TNF/*k*-means and random clustering, as before. In this case, three groups were produced, to reflect the three species thought to be present in the sequenced sample: the ivy plant host, a fungal and a bacterial pathogen. A summary of the contigs produced in these assemblies is given in Table 6.4.

Here, as observed with the blackberry dataset, the total numbers of contigs assembled from clustered reads were greater than those produced from the unclustered ivy dataset (97): assembly of reads clustered by TNF/*k*-means (104), at random in the same proportions as produced by TNF/*k*-means (143), and at random into equally proportioned clusters (167). The combined length of contigs produced from clustered reads was also found to be greater than that from contigs assembled from the unclustered dataset. In the case of clusters produced by TNF/*k*-means, this combined length of contigs was ~57 kbp, approximately 3 kbp greater than obtained from unclustered reads, while assembly of randomly generated clusters of the same proportions as the three produced by TNF/*k*-means produced contigs of total length ~87 kbp, and randomly assembled clusters of equal proportions ~103 kbp.

As observed with the previous datasets, the number of reads assembled or partially assembled remained broadly consistent between the different methods of clustering. In this instance, fewer singleton reads remained in the assembly of clustered reads - 3212 singletons remained from unclustered assembly, while the smallest number of singletons in a clustered assembly, where the dataset was divided at random into clusters of identical size to those generated by TNF/*k*-means, was 2255.

Table 6.4 *Details of contigs produced from* de novo *assembly of ivy dataset as a whole, and separated into three clusters. The number of reads assembled, the combined number of reads assembled and partially assembled (APA) and the number of singleton reads are given. Statistics are included for reads clustered at random and with TNF/*k-means.

	Un-clustered	Clustered with TNF/*k*-means				Clustered at random with same proportions as TNF/*k*-means groups				Clustered at random with ~1/3 reads in each group			
Metric	**All reads**	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Total**	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Total**	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Total**
Contigs	97	99	3	2	104	84	36	23	143	59	52	56	167
Combined length (bp)	53954	55051	1606	446	57103	50289	24078	12904	87271	35409	35793	32159	103361
Mean length (bp)	556	556	535	223	549	599	669	561	610	600	688	574	619
APA	17144	13487	1732	712	15931	14868	2480	765	18113	5966	6256	5805	18027
Singletons	3212	1981	603	70	2654	2089	103	63	2255	987	588	922	2497

Sample 3 - tomato + *Pepino mosaic virus*

The tomato/PepMV dataset of 65,691 cDNA reads was assembled as a whole, and after TNF/*k*-means and random clustering into two groups, the number of groups corresponding to the predicted number of species contributing to the sequences in the dataset. A summary of the isotigs and contigs produced in these assemblies is given in Table 6.5.

Once again, a similar pattern was observed in these results as in those discussed previously. After reads were clustered with TNF/*k*-means, a slightly larger number of contigs were assembled (82) than from the unclustered dataset (67). Assembly of randomly grouped reads produced 73 contigs where the groups produced were of the same sixze as those generated by TNF/*k*-means, and 71 where the dataset was divided into equally proportioned groups.

The combined length of the contigs produced from the dataset when reads were left unclustered was 56.6 kbp. As with the other datasets investigated, the assembly of clustered reads produced contigs of greater combined length than where the dataset was assembled as a whole. After reads were clustered with TNF/*k*-means, the assembled contigs covered 60.5 kbp, while assembly following the randomised grouping of reads produced contigs of combined length 73 kbp and 73.4 kbp where reads groups were of the same size as in TNF/*k*-means and of equal size respectively.

The mean length of contigs produced from randomly grouped reads was also greater than that of unclustered reads and reads clustered by TF/*k*-means. A small reduction in mean contig length was observed where reads were clustered by TNF/*k*-means, relative to the unclustered data.

The number of isotigs assembled in each case correlated with the numbers of contigs produced. However, the mean length of these isotigs was greatest where the dataset was assembled without first being clustered.

The nuber of reads assembled and partially assembled remained consistent across each set of assemblies, but a slightly greater number of singleton reads remained in the clustered assemblies (~5.3 - 5.4 kbp in contrast to 5 kbp for unclustered assembly).

Table 6.5 *Details of contigs and isotigs produced from* de novo *assembly of tomato/PepMV dataset as a whole and separated into two clusters. The number of reads assembled, the combined number of reads assembled and partially assembled (APA) and the number of singletons are given. Statistics are given for reads clustered at random and by TNF/k-means.*

	Un-clustered	Clustered with TNF/*k*-means			Clustered at random with same proportions as TNF/*k*-means groups			Clustered at random with ~1/2 reads in each group		
Metric	**All reads**	**Cluster 1**	**Cluster 2**	**Total**	**Cluster 1**	**Cluster 2**	**Total**	**Cluster 1**	**Cluster 2**	**Total**
Contigs	67	64	18	82	22	51	73	35	36	71
Combined length (bp)	56637	39232	21323	60555	29084	43972	73056	36126	37350	73476
Mean length (bp)	845	613	1185	738	1322	862	1001	1032	1038	1035
Isotigs	73	69	19	88	23	59	82	36	40	78
Mean length (bp)	1599	656	1625	865	1588	1121	1252	1548	1418	1441
APA	60231	15460	44491	59951	18470	41483	59953	30079	29912	59973
Singletons	5003	4634	762	5396	1773	3526	5299	2605	2695	5286

Speed of assembly

The time taken for completion of assembly for each dataset as a whole and in clusters is detailed in Table 6.6. Assembly was performed on a single 2.2 GHz AMD Opteron 6174 CPU with 512 KB memory.

The total time required for assembly of each dataset was reduced when clusters of reads from the dataset were considered separately. A greater reduction in time required for assembly was observed where reads were clustered at random into equally sized groups, than when clustered with TNF/*k*-means.

Table 6.6 *CPU time in seconds taken for Newbler* de novo *assembly of each dataset, before and after clustering both with TNF/k-means and at random.*

Dataset	Whole Dataset	Clustered with TNF/*k*-means				Clustered at random			
UT+Psp2126		**Cluster 1**		**Cluster 2**	**Total**	**Cluster 1**		**Cluster 2**	**Total**
	444.428 s	77.744 s		315.001 s	392.745 s	182.493 s		180.199 s	362.692 s
Blackberry R34r5		**Cluster 1**		**Cluster 2**	**Total**	**Cluster 1**		**Cluster 2**	**Total**
	6386.751 s	3384.738 s		1732.481 s	5117.219 s	1746.831 s		1728.347 s	3475.178 s
Ivy 10		**Cluster 1**	**Cluster 2**	**Cluster 3**	**Total**	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Total**
	252.471 s	221.847 s	21.479 s	0.838 s	244.164 s	49.468 s	51.874 s	48.148 s	149.490 s
Tomato/PepMV 1		**Cluster 1**		**Cluster 2**	**Total**	**Cluster 1**		**Cluster 2**	**Total**
	390.477 s	60.585 s		278.356 s	338.941 s	110.175 s		137.725 s	247.900 s

Discussion

UT+Psp2126

In previous chapters it had been observed that clustering of reads by TNF/*k*-means analysis produced groups in the data that were enriched with reads from a single species. Here, this enrichment was manifested in the disproportionate number and combined length of contigs assembled from *A. thaliana* reads in one cluster and *Pseudomonas* reads in the opposing cluster. These proportions were considerably divergent from the ~5:1 ratio of *A. thaliana* and *Pseudomonas* reads in the dataset as a whole.

Although the total combined length of *A. thaliana* contigs remained broadly consistent between all assemblies performed with UT+Psp2126, with the mean length of these contigs even found to increase where the dataset was divided at random, a distinctive effect was observed on the number and combined length of *Pseudomonas* contigs. Those *Pseudomonas* contigs constructed from reads grouped by TNF/*k*-means provided coverage for the bacterial genome that was comparable to that produced from the dataset without clustering, whereas a considerable drop in *Pseudomonas* sequence assembly was observed when UT+Psp2126 was grouped at random.

The randomised clustering of reads was also found to result in a marked increase in erroneous assembly of chimeric contigs, compared to those from unclustered and TNF/*k*-means-clustered reads. Although the assembly of randomly grouped reads produced a greater overall combined length of all the contigs assembled, the combined length of ‘pure’ contigs assembled from reads derived from a single species was found to be less than in unclustered or TNF/*k*-means-clustered assembly.

Considering the detrimental effect on the contigs assembled from randomly clustered reads, especially on those contigs constructed exclusively from *Pseudomonas* reads, sequence assembly from groups of reads produced by TNF/*k*-means was found to be preferable to random clustering. In fact, the overall assembly of TNF/*k*-means-clustered reads produced results almost identical to those from assembly of the whole dataset considered at once, with only a slight reduction in the combined length of *Pseudomonas* contigs. This is best illustrated in Figs. 6.1 and 6.2, where the improvement in assembly with

the TNF/*k*-means approach, compared with that of randomly generated clusters of the same size, is especially apparent. The overall length of the three chimeric contigs assembled from these clustered reads was also found to be lower than that observed from unclustered reads, which may suggest that the clustering of reads prior to assembly reduces the likelihood of such contigs being assembled.

The differences between assembly of TNF/*k*-means-clustered reads and of reads randomly grouped in the same proportions as these clusters - the greater combined length of *Pseudomonas* contigs and the reduced length of chimeric sequence produced with TNF/*k*-means clusters - could be attributed entirely to the method of clustering used, as all other conditions for assembly remained identical.

The increase in mean length of *A. thaliana* contigs in both cases of randomised clustering was difficult to explain without further investigation of the specific nature and sequence of these contigs. Further investigation of both the contigs assembled, and the mapping of these sequences and other reads to the genomes of both species would allow for the true coverage to be determined. As well as allowing for a greater understanding of the effects of grouping reads on *de novo* assembly, this would also provide insight into the reasons for the discrepancy between expected and observed coverage of the genomes by the contigs produced.

Further study of these results is required for a full understanding of the effects of randomised clustering to be gained. A repeat of these analyses, using a range of other assembly software packages would also allow for the impact of the choice of approach to assembly to be assessed and accounted for.

True sequencing datasets

Assembly of the three real sequencing datasets could not be assessed in the same, species-specific detail as with UT+Psp2126, as the prior knowledge of the origin of each read was not available in these cases. However, a number of trends were observed throughout the results that suggested that clustering of reads prior to assembly might have produced similar effects with these datasets.

In each case, the numbers of reads assembled and partially assembled, and the number of singleton reads, were found to be broadly consistent in assembly

of the dataset before and after clustering. This indicated that the differences observed in the number and length of contigs produced in assembly were the result of the differences in the clustering of the data, rather than an inconsistency in the total number of reads used in each assembly. This pattern was also observed in the analysis of UT+Psp2126, although a slight decrease in assembled and partially assembled reads and a slight increase in singleton reads was observed with assembly after randomised grouping in that case.

Another trend common between the analysis of all three true sequencing datasets was the assembly of a greater combined length of contigs after randomised clustering into groups, for both size ratios used. Assembly after clustering by TNF/*k*-means also produced a greater combined contig length than where the dataset remained unclustered, for all three datasets, but this increase was not as great as for randomly grouped reads.

Where UT+Psp2126 was clustered at random, a considerable increase in the assembly of chimeric sequence was observed, which had the effect of producing an increase in the overall combined length of contigs constructed. It may be that a similar effect could account for the increase in combined contig length observed with the three true sequencing datasets discussed here. However, without further investigation, or knowledge of the origin of the reads used in each of these assemblies, the conclusions that could be drawn from these analyses was limited. The consistencies between the results from each dataset and from UT+Psp2126 suggested that TNF/*k*-means clustering may have been effective with true sequencing datasets as well as the simulated dataset, and as such further exploration was desirable.

In order to more fully evaluate the effect of prior clustering of reads on the nature of contigs produced in assembly, the reads and the contigs produced from the true sequencing datasets used here could be analysed by alignment to a reference database of sequences belonging to the species represented in the dataset, or closely related species. This would allow a better understanding of the effectivity of the clustering in grouping together reads originating from the same species, and of how effectively these reads were assembled together into non-chimeric contigs.

Effect of clustering on speed of assembly

As was expected, the clustering of datasets prior to assembly reduced the overall time required for assembly of reads to be completed. The greatest reduction, in terms of absolute time required, was observed with assembly of sample 1, the blackberry dataset that consisted of ~111,000 reads. Here, a reduction of ~20% was observed in the time required for assembly when the dataset was first clustered by TNF/*k*-means, or a reduction of 21:10 (mins:secs) from 106:27 to 85:17 in absolute terms.

The reduction could be compounded further by performing the assembly of each cluster of reads in parallel using multiple processors, rather than one after another. Given sufficient processors, if performed in parallel the assembly of a whole dataset could be expected to take little more real 'wall clock' time than the time required for assembly of the largest cluster produced.

The effect of the size of clusters on the speed of assembly was illustrated by the use of randomly generated, equally sized groups of reads. Assembly of these groups required less time than for TNF/*k*-means-clustered reads, with the time required for assembly of sample 1 reduced by 48:32, or ~45.5%. This was due to the greater reduction in size, in terms of number of sequences, between the largest cluster produced at random and the dataset as a whole.

As the time required for an assembly to be performed is proportional to the square of the number of sequences in the dataset, the time required for several clusters of reads to be assembled increases with the square of the size of the largest cluster. The largest randomly produced cluster contains approximately 1/*k* of the total reads in a dataset, where *k* is the number of clusters produced, while the largest cluster produced with TNF/*k*-means can be of any size from 1/*k* up to that of the dataset as a whole, which results in a greater time requirement for a TNF/*k*-means-clustered assembly.

However, the similarity in contigs produced from assembly of a whole dataset and from clusters of reads produced with TNF/*k*-means, suggested that the improvement in CPU time requirement associated with clustering reads in this way comes at very little cost in terms of quality of the assembly results. The reduction in *Pseudomonas* contig number and mean length and the increase in chimeric contig number and mean length observed where UT+Psp2126 was

randomly clustered suggested that, while a greater reduction in CPU time requirement was obtained from random clustering, this approach had a much greater impact on the assembly results, which was unlikely to be desirable.

Conclusion and future work

The capability, provided by TNF/*k*-means clustering, to assemble a large proportion of the overall sequence covered for each species in the dataset in separate groups may prove to be beneficial for future analyses of multi-species samples. The isolation of the vast majority of *Pseudomonas* contigs from the vast majority of *A. thaliana* contigs in separate cluster assemblies opens up the possibility of not only reducing the time required for computation of dataset assembly, but also simplifying the study of pathogens/minority species in multi-species samples.

The reduction in the time required for assembly, combined with an enrichment in contigs produced from a single species in a given cluster, suggested that the TNF/*k*-means approach may be advantageous, especially in analysis of very large sequencing datasets. If the number of contigs obtained, as a whole and from assembly of reads from each species represented in the dataset, is not considerably adversely affected by clustering, as was suggested by the results obtained here, the time required to assemble a dataset can be reduced effectively using this method. The approach also provided the advantage of isolating contigs from a particular species, in the assembly results from an individual cluster.

Although the reductions observed where reads were clustered by TNF/*k*-means were not large as a proportion of the time required for unclustered assembly, the reduction in real terms for sample 1, where assembly took >20 minutes less time to complete, suggested that this approach could be very useful if it was not associated with a deleterious effect on the assembled sequences. Such a reduction in time would be most beneficial for the increasingly large datasets being produced as high-throughput sequencing technologies advance. The ever-increasing numbers of individual reads being produced from massively parallel sequencing platforms are resulting in a huge increase in the time required for assembly of contigs from these datasets. Under such conditions, even a 20% reduction could cut the overall time required for assembly by hours

The UT+Psp2126 dataset cannot be thought truly representative of a true sequencing dataset, as it was constructed from reads obtained from two separate sequencing experiments. As such, the conclusions that can be drawn from a comparison of the assembly results obtained from this dataset and the three true sequencing datasets are limited. However, the common trends observed between the results from each dataset do suggest that some success might have been achieved in the clustering and contig assembly of these reads according to their species of origin.

To allow for firm conclusions to be drawn from the results described in this chapter, further investigation of the assemblies and datasets would be required. An alignment-based analysis of those datasets containing unknown pathogens and poorly characterised host species could provide a prediction of the relative proportions of sequence present from each species and further insight into the possible effects of clustering. Mapping the reads in these datasets to a reference database of genomic sequences from the host species and related organisms, and to bacterial or fungal sequences thought to be related to those present would allow these proportions to be predicted.

Similarly, by comparing the assembled contigs to a database such as that described above (or a larger database of sequences such as the NCBI *nt* database of non-redundant nucleotide sequences), firmer conclusions might be drawn regarding the success of assembly before and after clustering, and the similarity/difference between these different assemblies assessed in a more quantitative manner. Without this information, the current results can only hint at the possible effects of clustering on assembly through observations of similar trends in the summary statistics of the datasets used and assemblies performed.

The Lander-Waterman model used to make predictions of coverage and assembly in UT+Psp2126 is well-established and easily understood, but a method designed for use with metagenomic datasets (e.g. Hooper, Dalevi et al. 2010) may have been more appropriate for use here, and with the other datasets considered, where no predictions could be made due to the lack of information about their contents and origin.

Once again, the existence of a true sequencing dataset containing reads from species with a fully sequenced genome, as was aimed to be produced in the work described in Chapter 3, would allow a considerably greater insight into the type of analysis discussed here. The availability of such a dataset for analysis would remove the limitations associated with the use of simulated sequencing datasets such as UT+Psp2126 in making predictions about the clustering and assembly of true sequencing data, while simultaneously eliminating the uncertainty associated with drawing conclusions from data of largely unknown or poorly characterised origin.


sequences
sequencing
reads
species
pathogen
methods
et
al
used
genomes
dataset
datasets
host
data
approach
sequence
clustering
may
produced
microbial
comparison
frequency
length
grouping
analysis
discussed
published
genome
sample
clusters
example
platforms
oligonucleotide
investigation
features
technologies
associated
composition
obtained
work
systems
available
longer
suggested
samples
increase
Chapter
vectors
metagenomic
likely
representation
proportions
SOMs
method
distances
previously
patterns
applied
sequenced
number
content
viral
interest
feature
contain
cluster
time
Life
high
map
successful
developed
short
assembly
DNA
derived
introduced
CA
material
understanding
emphasis
allow
Torrent
However
genomic
mean
CT
Afreixo
alignment-free
aims
relative
advances
required
results
new
similar
characterisation
USA
distinguish
means
effect
known
information
using
present
signature
produce
infected
project
order
current
containing
tissue
techniques
Technologies
separating
general
made
evaluation
Ion
bacterial
according
platform
become
combination
true communities
range
SOM
high-throughput
technology
studies
increases
HHSOM
read
large
van
profiles
lkb
GC
including
measure
compared
training
successfully
different
use

7

Discussion and future directions

Abstract

In this chapter, the work reported in this thesis is discussed within the context of the wider research environment. Suggestions are made regarding the future directions that the research may take, and how impending developments in sequencing technology might impact upon it.

Discussion

The development of new techniques of sequence comparison, the introduction of new sequencing platforms and the continuing increases in computational power and storage capacity are bringing about rapid changes in the fields of metagenomics and biological sequence analysis.

To date, most metagenomics research has been into microbial datasets more complex than the host-pathogen systems of most interest here. The emphasis on microbial communities in the wider literature can be attributed to the difficulty encountered in culturing most bacterial species (Costello, Lauber et al. 2009; Dick, Andersson et al. 2009; Qin, Li et al. 2010; Rodriguez-Brito, Li et al. 2010), and in understanding the interactions and relationships that prevail within these populations.

While the alignment-free approach to grouping sequences taken in this project is similar to some the methods used in other studies in the literature, the data of interest here is different in both general composition and complexity. The emphasis of this project was on environmental samples containing fewer species in total than a typical microbial metagenome, with a view to isolating and identifying minority constituents (i.e. pathogens in a host system).

The datasets produced here, to allow a quantitative evaluation of clustering, reflected this difference in focus. Datasets have been published previously with a controlled composition, either through the combination of reads from individual experiments, sequencing a variety of microbial species (Mavromatis, Ivanova et al. 2007) or through sequencing of a number of microbial species, mixed together artificially in roughly equal proportions (Morgan, Darling et al. 2010). These datasets may provide a good insight into the performance of many methods used to study complex metagenomic datasets, but are not wholly appropriate for use here.

The efforts made here to prepare a dataset of sequencing reads from an *in vivo* mixture of species represented a different approach (see Chapter 3), using organisms with fully-sequenced genomes in combination as naturally as possible, to determine the relative proportions of sequences obtained for each species following sequencing.

The results obtained from this approach suggested that the ratio of host to

pathogen material extracted from these systems was too disproportionate for the datasets to be useful in evaluating the methods under investigation. This disproportionality suggested that, for some combinations of host and pathogen species, the general approach taken here of sequencing infected tissue and grouping the reads obtained may not be successful in many cases. As discussed previously, other multi-species datasets have been produced, including those used in the work described in Chapter 6, that contain sufficient reads from both species to be suitable for this approach. Further investigation would be required to determine the proportions of reads required from each species to render such an approach worthwhile.

However, the requirement for alignment-free methods of analysis remains. The improvement of current sequencing platforms, and the introduction of new platforms is leading to the production of datasets containing ever more reads, often of a longer length. For example,the Ion Torrent platform (Rothberg, Hinz et al. 2011, *Life Technologies*, CT, USA) is capable of sequencing millions of reads, each ~200 bp in length (increasing to ~400 bp by the end of 2012), in a fraction of the runtime required by previous high-throughput technologies. Single molecule sequencing technologies developed by *Pacific Biosystems* (CA, USA) are capable of producing reads of length exceeding the 1kb mark, with a mean sequence insert length >3 kbp, which is circularised and sequenced repeatedly in each read produced (http://www.pacificbiosciences.com/products/smrt-technology/smrt-sequencing-advantage/).

There are many benefits associated with these improvements in sequencing technology. Genomes are being sequenced at ever greater coverage (the Ion Torrent platform can sequence a typical 45 Mbp bacterial chromosome at ~25x coverage), and environmental communities being sequenced at a greater depth, allowing for the detection and characterisation of more and more species.

The huge volume of data produced is also associated with an increased burden on the systems for storage and computation of sequencing reads. Alignment-based analysis methods do not scale well with the number of sequences for comparison, so the motivation for developing alignment-free approaches such as those discussed here remains as powerful as ever. This is especially true for

metagenomic studies, which require an efficient means for identifying reads from different species in the dataset to take full advantage of the benefits offered by such technological advances.

The major obstacle to successful clustering of sequencing data is the short average length of the reads produced from current high-throughput technologies. However, advances in sequencing technology are already leading to an improvement in these read lengths. The reads generated by the platforms manufactured by *Roche*/*454 Life Sciences* (CT, USA) fall at the longer end of the length spectrum, with reads of up to 1 kbp produced by the most up-to-date *Titanium* system. As the mean length of reads increases, the effectivity of clustering methods applied to these datasets should improve accordingly, and many of the limitations observed in this project may soon no longer be relevant.

Conventionally, metagenomic sequences have been classified, either through alignment or composition-based methods, after assembly into contigs. This approach has the advantage of considerably increasing the length of the sequences for classification. In the case of alignment-based methods of analysis, this increased length provides a larger range along which to build an alignment and subsequently greater confidence in any regions of similarity found along this length. Where sequence composition features have been used, the increased length of assembled contigs provides a greater sampling space for the calculation of these composition features, improving the likelihood of effectively approximating the true feature profile of the original genome of a sequence, and the probability of two sequences originating from the same genome producing similar profiles. The assembly of sequences also reduces the pool size for comparison, which is likely to reduce the time and computational power required for further downstream analysis of the data.

However, as discussed the process of sequence assembly can be time-consuming for large datasets - an issue that is predicted to be exacerbated by further advancements in sequencing technology resulting in larger and larger numbers of reads being produced.

Given the considerable limitations to the grouping and separation of sequences observed in this project with the short length of unassembled reads, it may be that the additional step of assembly prior to any further grouping or classification

would be beneficial in spite of the computational burden of the task.

The results presented in Chapter 6 suggested that clustering of reads prior to assembly may reduce the time required for assembly while having little effect on the overall results of this assembly. There was also some evidence to suggest that such a process of clustering raw reads prior to sequencing could reduce the likelihood and extent of chimeric sequence assembly, combining reads originating from multiple genomes into a single contig.

If these effects are genuine, it is thought that clustering could prove valuable as a first step towards assembly and further analysis and classification of sequences, providing both an improvement in assembly time and accuracy, as well as a prediction of shared origin for reads contained in the groups produced. As discussed in Chapter 6, these results require considerable further investigation before any confidence can be placed in these suggestions.

The advanced sequencing technology being developed by *Oxford Nanopore Technologies* promises a massively-increased read length with no theoretical limit, and the capability to produce these sequences at high speed in a process known as ‘strand sequencing’ (Lieberman, Cherf et al. 2010). When these systems become widely-available, whole genome sequencing is likely to undergo another revolution, shifting the focus of research further towards the informatics associated with storing, handling and analysing sequence data. Strand sequencing is likely to be hugely beneficial in the generation of whole chromosome and genome sequences, but it does not offer the same depth of sampling associated with the use of massively-parallel sequencing methods to generate millions of individual reads from many different sequence fragments, which may limit its suitability for use in metagenomic analyses.

Future directions

Sequence features

As discussed previously, the four feature types evaluated in this project are not the only available means for representation of DNA sequences by their composition. Other characteristics of sequences have been used in the past, including average mutual information profiles (Bauer, Schuster et al. 2008) and chaos game representation (Jeffrey 1990; Deschavanne and Giron 1999; Almeida, Carriço et al. 2001; Joseph and Sasikumar 2006), although it has been suggested that this form of sequence representation is superseded by oligonucleotide frequency distributions (Goldman 1993) and as such may not provide further improvement on the clustering achieved already.

The ratio between the observed and expected oligonulceotide frequencies in host and virus genomes has been used to study similarities in the signature patterns of these sequences (Barrai 1990; Pride, Wassenaar et al. 2006), while a measure of information content of sequences has been used more recently to distinguish phage genomic material from that of bacterial species (Bohlin, van Passel et al. 2012). The potential of this method for separation of pathogen and host sequences is discussed in more depth later.

The representation of tetranucleotide frequencies as *tf-ti* vectors for use with the hyperbolic hierarchically-growing self-organising map HHSOM (Martin, Diaz et al. 2008) aims to increase the signal from informative features in the distribution, by amplifying the frequency of rare features while reducing that of features common throughout the sequences in the dataset, so as to highlight the differences between sequences.

An evaluation of all of these feature types, similar to that carried out for the features investigated here, would provide a more exhaustive survey of the options available for characterising sequences.

Recently, Carlos Bastos, Vera Afreixo and colleagues, who introduced inter-nucleotide distances as a method for comparing and grouping DNA sequences, published results of an investigation into inter-di-nucleotide distances (Bastos, Afreixo et al. 2011). This group identified the potential for inter-nucleotide distances to be used as genomic signature features and, although these features were not found to be useful in the work described in this thesis, some

investigation might be made into whether inter-dinucleotide distances could be used in a similar way.

Clustering methods

The range of clustering methods compared in Chapter 5 was by no means a complete collection of the techniques available for grouping and separating sequence reads.

Other methods that have been or could be applied to the grouping of nucleic acid sequences include; the hierarchical clustering methods CHAMELEON (Karypis, Eui-Hong et al. 1999) and CURE (Guha, Rastogi et al. 2001; Qian, Shi et al. 2002), both designed for use with large datasets; the density-based clustering algorithm DENCLUE (Hinneburg and Gabriel 2007), which is designed to overcome the difficulties associated with applying such methods to high-dimensional data such as the oligonucleotide frequency vectors used here; and model-based approaches, using implementations such as MCLUST (Fraley and Raftery 1999; Fraley and Raftery 2002) to identify groups in the data by fitting distributions to the data. An evaluation of the clustering performance of such methods, with the kind of sequencing data of interest here would provide a further understanding of the best approach to take to grouping and separating reads.

In addition to the HHSOM (Martin, Diaz et al. 2008) implemented here, several other variants of the SOM have been applied to the problem of separating and grouping sequences according to their species of origin. The emergent SOM (ESOM, Ultsch and Fabian 2005) and growing SOM have both been used in metagenomic analysis of a microbial community (Chan, Hsu et al. 2008; Dick, Andersson et al. 2009). These studies have been published in addition to studies using standard SOMs in similar analysis of DNA sequences (Kanaya, Kinouchi et al. 2001; Abe, Kanaya et al. 2002; Abe, Sugawara et al. 2006).

These maps have been applied to microbial communities of many more species than in the datasets focussed on in this work, and the methods require further investigation. The difficulty of using such methods lies in the need for a training dataset to prepare a map for use as a classifier of sequencing reads. As discussed in the conclusion to Chapter 5, sequences as short as those typically obtained from high-throughput sequencing have only been successfully

clustered using SOMs previously trained with a set of longer sequences. This approach is inefficient where the component species in a sequenced sample are not known and may prove to be unfeasible for clustering of raw sequencing reads from multi-species samples.

Rather than training a new map for each sample sequenced, based on a prediction of the species present, it may be possible to build a range of SOMs trained with sequences from the genomes of host species and their common pathogens. Such an approach has become popular in the development of microarray systems for the detection of pathogens, and as the number of available genomes and genomic sequences increases, so too should the range of host and pathogen species for which SOMs could be prepared.

Datasets

With further time and resources available, another attempt could be made to produce the kind of dataset that was aimed for in the work described in Chapter 3. That host and pathogen material was extracted in such unequal proportions in the sequenced samples was unexpected, and may suggest that such an approach to the isolation and identification of pathogen sequence reads is neither efficient nor cost-effective. However, such an approach, of directly sequencing DNA/RNA extracted from infected tissue to produce data for clustering analysis is still considered worthwhile. Investigation of samples prepared from other combinations of host and pathogen species, especially with viral species of a known high titre, will provide a much better understanding of the limits of this methodology.

The dataset used for further feature comparison, UT+Psp2126 despite being composed from true sequencing reads, does not provide the ideal platform for feature comparison. One limitation of the UT+Psp2126 dataset used here was the use of a bacterial species, *Pseudomonas* sp. 2_1_26, that is a human pathogen and not a pathogen of the host plant species, *A. thaliana*.

This combination does not model the selective pressures that may exist between the host and pathogen, which may have an effect on the relationship between their signature feature patterns. This issue could be addressed if a real dataset was used, or if sequencing reads from two more appropriate species were used to construct the dataset.

Sequence assembly

As new sequencing technologies and methods are introduced, and as current platforms are improved, both the number and mean length of reads are generally predicted to increase. As the number of reads in a dataset increases, a reduction in assembly time will become more beneficial.

The largest datasets analysed here contained in the order of 10^5 reads, and the time required for assembly of these datasets was relatively short. However, modern sequencing platforms are capable of generating many times more reads in a single sequencing experiment. For example, the *Ion Proton System* from Life Technologies (CT and CA, USA) aims to produce datasets in the order of 10^8-10^9 reads, a figure likely to increase as the technology is upgraded in the near future. Such an increase in dataset size makes the benefits associated with successful clustering of a multi-species dataset all the more important.

As discussed, the general increase in mean length of reads produced in sequencing is likely to improve the effectiveness of this approach, as clustering of reads based on sequence composition features becomes easier.

Identifying clusters of interest

Most of the clustering methods surveyed here were unsupervised, that is, they group the data according to the features provided, without access to any prior information about the classes present. Consequently, the clusters produced have no labels asssociated with them, providing a prediction of their contents.

In the case of the sequence clustering of interest here, grouping of a sequencing dataset derived from a sample of unknown composition (e.g. a sample of tissue infected with an unknown pathogen) would produce a number of clusters. One or more of these clusters could be expected to contain a large proportion of the sequences derived from the pathogen genome, while others may contain very few pathogen sequences at all. The challenge, if the aim is to isolate these pathogen sequences, is to predict which of these clusters are of interest.

One approach to doing so could be to produce a summary of each cluster to allow a comparison between them and a prediction of which clusters are of most interest. For example, if the mean GC content of the sequences contained in each cluster was compared (accompanied by some measure of the intra-

cluster variance), prior knowledge of the GC content of the host genome, or assumptions about the type of pathogen present, could allow the identification of the cluster(s) most likely to contain the bulk of the pathogenic sequences from the dataset.

Similarly, the availability of a database containing oligonucleotide relative frequency profiles of available genomes/sequences would allow the mean profile to be displayed for each cluster and a prediction made regarding its contents.

Virus-host genomic signature co-evolution

The work in this project has revolved around the ability to group and separate DNA sequences according to their species of origin, by using certain signature feature patterns conserved throughout the genome to characterise and compare these sequences.

Published research has established that viral species, reliant on host machinery to replicate and maintain their genome, display similar oligonucleotide frequency patterns in their genomes to those of the host itself (Barrai 1990; Pride, Wassenaar et al. 2006; Simmons 2008). If the methods described in this work were applied to sequencing data taken from a sample of tissue infected with a viral pathogen, it is likely that such similarity in signature feature patterns between genomes would have a diminishing effect on their resolving power. The issue is further complicated by the existence of endogenous retroviral sequences in host genomes (Lower, Lower et al. 1996), which may be difficult to distinguish from the true viral sequencing reads present in a sample containing a host and viral pathogen.

Further investigation, applying the clustering techniques used here and/or other similar approaches to sequencing data obtained from a virus-infected sample is required, in order for the effects and consequences of the similarities between viral and host sequences to be understood fully. It is possible that, even if analysis via oligonucleotide relative frequency patterns cannot resolve host and virus sequences so successfully, other means of comparison may be applicable.

For example, recent results indicate that a measure of information capacity can be used to distinguish phage and prokaryote sequences (Bohlin, van Passel et

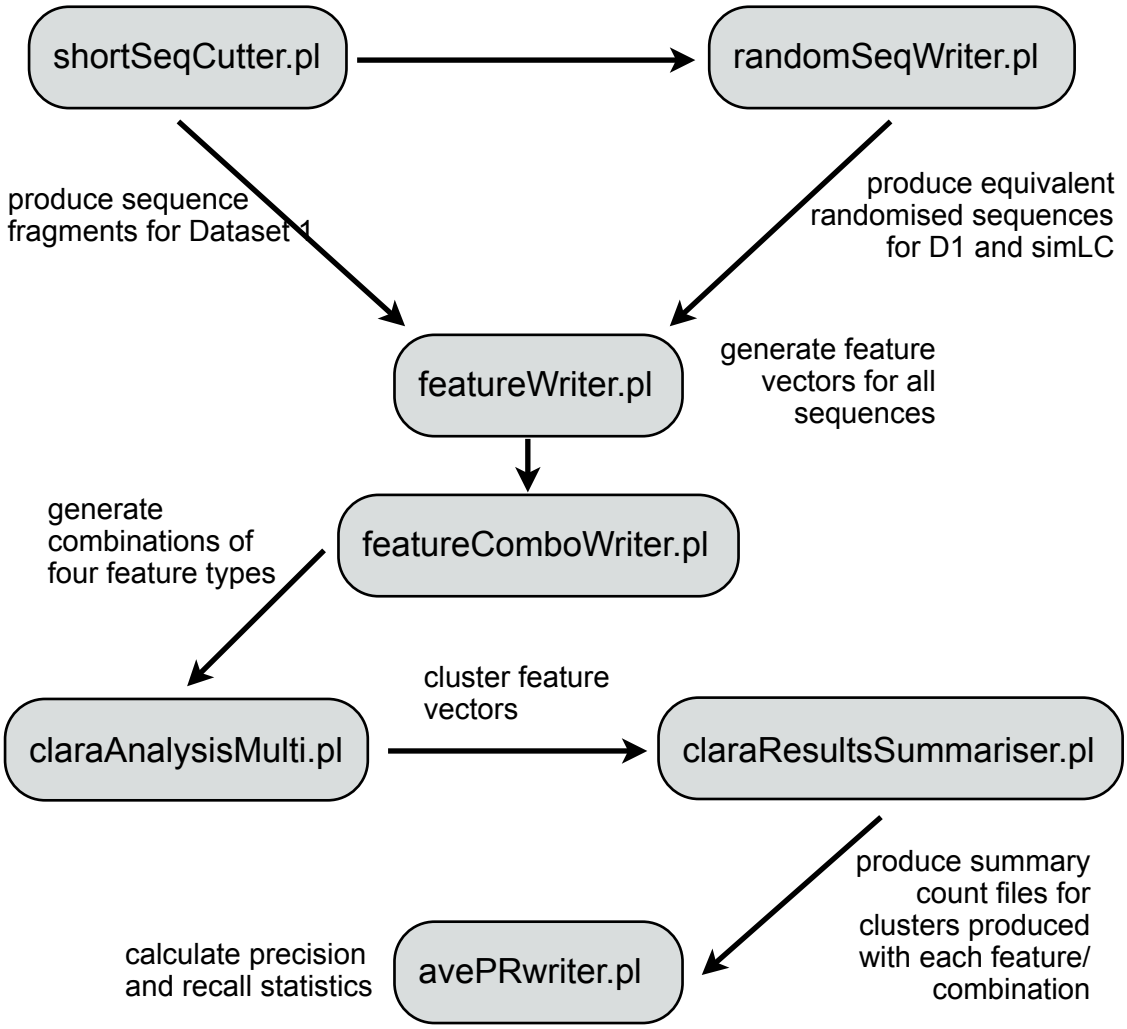
al. 2012). This method of characterisation exploits the highly compact nature of the viral genome, requiring high information content per nucleotide in the sequence. Given the relative profligacy of eukaryotic genomes, such a characterisation might be successfully applied in a complementary fashion alongside oligonucleotide frequency vectors when clustering sequences from a host-virus dataset, but the presence of endogenous retroviral sequences in eukaryotic genomes may make this distinction more difficult.

Appendix A

Annotated reproductions of *perl* scripts used in the project. Scripts are given in chronological order of use throughout the project.

- Appendix A-1: a flowchart of the use of scripts in Chapter 2
- Appendix A-2: shortSeqCutter.pl (Chapter 2)
- Appendix A-3: randomSeqWriter.pl (Chapter 2)
- Appendix A-4: featureWriter.pl (Chapter 2)
- Appendix A-5: featureComboWriter.pl (Chapter 2)
- Appendix A-6: claraAnalysisMulti.pl (Chapter 2)
- Appendix A-7: claraResultsSummariser.pl (Chapter 2)
- Appendix A-8: avePRwriter.pl (Chapter 2)
- Appendix A-9: SAMseqAssigner.pl (Chapter 3)
- Appendix A-10: partClustering.pl (Chapter 6)
- Appendix A-11: contigInfo.pl (Chapter 6)
- Appendix A-12: randomSeqFetcher.pl (Chapter 6)

Appendix A-1: Use of *perl* scripts in Chapter 2.



Appendix A-2

shortSeqCutter.pl

```
#! /usr/bin/perl
use warnings;
use strict;
use Bio::SeqIO;

#enter name of input file ("[species][Gen/Chr][number].fasta") and desired average
length
my $filename = shift;
my $aveLength = shift;

#import sequences as Bio::Seq objects
my $genome = Bio::SeqIO -> new(-file => $filename, -format => "fasta");
my $genomeseq = $genome->next_seq;
my $fullseq = $genomeseq->seq;
my $fragnumber = 1;

#establish name for label of fragments and output filename
my @namesplit = split /Gen|Chr/, $filename;
my @infilesplit = split /\./, $filename;
my $orgname = $namesplit[0];
my $outfile = "$infilesplit[0]" . "_$aveLength" . "bp.fasta";

#initialise random sequence length
my $randlength = "\\w" x (int(rand(0.2*$aveLength+1)+(0.9*$aveLength)));

#write random length sequence fragments to output file
open (SHORTSEQSFILE, ">$outfile");
while ($fullseq =~ /\G($randlength)/gc) {
   print SHORTSEQSFILE (">$orgname$fragnumber\n$1\n\n");
        $fragnumber = ++$fragnumber;
        $randlength = "\\w" x (int(rand(0.2*$aveLength+1)+(0.9*$aveLength)));
}

close SHORTSEQSFILE;
```

Appendix A-3

randomSeqWriter.pl

```
#! /usr/bin/perl
use warnings;
use strict;
use Bio::SeqIO;

#compute input filename, preference for whether species name from sequence ID
string should be used in random sequence ID ([T/F]) and output filename
my $inputFile = shift;
my $namesPref = shift;
my $outfile = shift;
$namesPref="\U$namesPref";

#import input sequences as Bio::Seq objects and generate array of sequence objects
my $seq_in = Bio::SeqIO->new(-file => "$inputFile", -format => "fasta");
my ($seq,@seq_array);
while ($seq = $seq_in->next_seq) {
        push (@seq_array, $seq);
}

#calculate mean sequence length of input sequences and set as mean length for
output random sequences
my $totLength=0;
foreach $seq (@seq_array) {
        $totLength=$totLength+$seq->length;
}
my $aveLength=$totLength/@seq_array;
my $randSeqAveLength=int($aveLength);

#calculate proportions of A, C, G & T in input dataset
my $totalAP=0;
my $totalCP=0;
my $totalGP=0;
my $totalTP=0;
foreach $seq (@seq_array) {
        my $acount = (($seq->seq) =~ tr/Aa//);
        my $ccount = (($seq->seq) =~ tr/Cc//);
        my $gcount = (($seq->seq) =~ tr/Gg//);
```

```
        my $tcount = (($seq->seq) =~ tr/Tt//);
        my $aprop = $acount/$seq->length;
        my $cprop = $ccount/$seq->length;
        my $gprop = $gcount/$seq->length;
        my $tprop = $tcount/$seq->length;
        $totalAP = $totalAP + $aprop;
        $totalCP = $totalCP + $cprop;
        $totalGP = $totalGP + $gprop;
        $totalTP = $totalTP + $tprop;
}
my $aveAP = $totalAP/@seq_array;
my $aveCP = $totalCP/@seq_array;
my $aveGP = $totalGP/@seq_array;
my $aveTP = $totalTP/@seq_array;
my $check = $aveAP+$aveCP+$aveGP+$aveTP;

#find and store number of sequences to be generated for each species represented in
input dataset
my %nameHash;
my $name;
my @names;
my $seqID;
my @IDSplit;
foreach $seq (@seq_array) {
        $seqID = $seq->id;
        @IDSplit = split /(\d+)/, $seqID;
        my $name = $IDSplit[0];
        if (exists ($nameHash{$name})) {
                $nameHash{$name}++;
        }
        else {
                $nameHash{$name}=1;
                push (@names, $name);
        }
}

#generate alphabetised system for labelling random sequence 'species' if $namesPref
ne 'T'
my @alphabet = ("A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P", "Q",
"U", "V", "W", "X", "Z");
```

```
if (@alphabet < @names) {
	foreach my $firstLetter (@alphabet) {
		foreach my $secondLetter (@alphabet) {
			my $secondOrderLetter = "$firstLetter" . "$secondLetter";
			push (@alphabet, $secondOrderLetter);
		}
	}
}

#open output file and print random sequences to it
my @outfileSplit = split /\./, $outfile;
$outfile = "$outfileSplit[0]RandomSeqs.fasta";
my $randSeq;
open (OUTPUTFH, ">$outfile") or die "$!";
if ($namesPref eq "T" or $namesPref eq "TRUE") {
	foreach $seq (@seq_array) {
		$seqID = $seq->id;
		@IDSplit = split /(\d+)/, $seqID;
		my $seqLabel = shift (@IDSplit);
		my $seqNo = join "", @IDSplit;
		print OUTPUTFH (">" . $seqLabel . "Random" . $seqNo . "\n");
		$randSeq = write_random_sequence($randSeqAveLength, $aveAP,
$aveCP, $aveGP, $aveTP);
		print OUTPUTFH ("$randSeq\n\n");
	}
}
else {
	my $speciesIndex=0;
	foreach $name (@names) {
		my $index = 1;
		while ($index <= $nameHash{$name}) {
			print OUTPUTFH (">Species" . $alphabet[$speciesIndex] .
"_Random$index\n");
			$index++;
			$randSeq = write_random_sequence($randSeqAveLength,
$aveAP, $aveCP, $aveGP, $aveTP);
			print OUTPUTFH ("$randSeq\n\n");
		}
		$speciesIndex++
	}
```

```
}

close OUTPUTFH;

#subroutine for writing random sequences
sub write_random_sequence {
	my $RSAL = $_[0];
	my $AInterval = $_[1];
	my $CInterval = $AInterval + $_[2];
	my $GInterval = $CInterval + $_[3];
	my $TInterval = $GInterval + $_[4];
	my @randSeqChars;
	my $pos = 0;
	my $randLength = int(rand(0.2*$RSAL)+(0.9*$RSAL));
	while ($pos < $randLength) {
		my $dieroll = rand(1);
		if ($dieroll <= $AInterval) {
			push (@randSeqChars, "A");
		}
		elsif ($dieroll <= $CInterval) {
			push (@randSeqChars, "C");
		}
		elsif ($dieroll <= $GInterval) {
			push (@randSeqChars, "G");
		}
		elsif ($dieroll <= $TInterval) {
			push (@randSeqChars, "T");
		}
		$pos++
	}
	my $randSeq = join "", @randSeqChars;
	return $randSeq;
}
```

Appendix A-4

featureWriter.pl

```
#! /usr/bin/perl
use warnings;
use strict;
use Bio::SeqIO;
use Bio::Tools::SeqWords;
use File::Basename;
use Statistics::Descriptive;

#input FASTA filename
my $filename = shift;

# read sequences from file into Bio::SeqIO object
my $seq_in = Bio::SeqIO->new(-file => "$filename", -format => "fasta");

#input features
my @setFeatures = ("GC", "IND", "OFDEG", "TNF");
my %selectedFeatures;
my $check=0;
my $featuresInput = shift;

chomp $featuresInput;

#prepare to generate chosen feature(s)
my @featuresArray = split /\,\s/, $featuresInput;
foreach my $inputFeature (@featuresArray) {
	foreach my $setFeature (@setFeatures) {
		if ($inputFeature eq $setFeature) {
			$selectedFeatures{$setFeature}=1;
			$check++;
		}
		elsif (exists($selectedFeatures{$setFeature})) {
		}
		else {
			$selectedFeatures{$setFeature}=0;
		}
	}
}
```

```
#check that no unknown feature types have been input
my $FAlength = scalar @featuresArray;
if ($check != $FAlength) {
        die and print LOGFH ("\nUnrecognised Feature Entered.  Accepted Features are:
@setFeatures\n");
}

#create array of sequence objects
my ($seq,@seq_array);
while ($seq = $seq_in->next_seq) {
        push (@seq_array, $seq);
}

#prepare output file and open filehandle
my @nameext = split /\./, $filename;
my $specFeatures = join "", @featuresArray;
my $statsfile = "$nameext[0]$specFeatures.txt";
open (STATSFILE, ">$statsfile") or die "$!";

#initialise parameters for OFDEG generation
my $samplingDepth = 20;
my $resamplingCutoff = 5;

#initialise cutoff length value for IND calculation
my $cutoffLength = 25;

#G/C content
my %GChash;
if ($selectedFeatures{"GC"}==1) {
        foreach $seq (@seq_array) {
#count ACTG & calculate as proporotion of seq
                my $seqID = $seq->id;
                my $acount = (($seq->seq) =~ tr/Aa//);
                my $ccount = (($seq->seq) =~ tr/Cc//);
                my $gcount = (($seq->seq) =~ tr/Gg//);
                my $tcount = (($seq->seq) =~ tr/Tt//);
                my $propGC = ($ccount + $gcount)/($seq->length);
                $GChash{$seqID} = $propGC;
        }
```

```
	print "GC content done...\n";
}

#generate all possible tetranucleotides in an array
my @firstn = ("A", "C", "G", "T");
my @secondn = ("A", "C", "G", "T");
my @thirdn = ("A", "C", "G", "T");
my @fourthn = ("A", "C", "G", "T");
my ($tetraseq, @tetra_array);
foreach my $firstn(@firstn) {
	foreach my $secondn(@secondn) {
		foreach my $thirdn(@thirdn) {
			foreach my $fourthn(@fourthn) {
				$tetraseq = ($firstn . $secondn . $thirdn . $fourthn);
				push(@tetra_array, $tetraseq);
			}
		}
	}
}

#TetraNucleotide Frequencies
my %TNFhash;
if ($selectedFeatures{"TNF"}==1) {
	foreach $seq (@seq_array) {
		my $seqID = $seq->id;
		my $fragment = $seq->seq;
		my $tetranucs = (($seq->length) - 3);
		my $seq_word = Bio::Tools::SeqWords->new(-seq => $seq);
		my $tetralength = 4;
#generate a hash of tetranucleotide counts
		my $tetracount = $seq_word->count_overlap_words($tetralength);
		my %tetrahash = %$tetracount;
		my $revword;
		my %endhash;
#calculate relative frequencies, as prop of total tetranucleotides in sequence
		foreach $tetraseq(@tetra_array) {
			if (exists($tetrahash{$tetraseq})) {
				$tetrahash{$tetraseq}=($tetrahash{$tetraseq}/
$tetranucs);
			}
```

```
                        else {
                                $tetrahash{$tetraseq} = 0;
                        }
                }
#combine relative frequencies with those of their reverse complement, or doubled in
the case of palindromic sequences
                foreach $tetraseq(sort keys %tetrahash) {
                        $revword = reverse($tetraseq);
                        $revword =~ tr/[AaGgTtCc]/TtCcAaGg/;
                        $revword = uc($revword);
                        if (exists($endhash{$revword})) {
                        }
                        else {
                                if ($tetraseq eq $revword) {
                                        $endhash{$tetraseq} = $tetrahash{$tetraseq};
                                }
                                else {
                                        $endhash{$tetraseq} = $tetrahash{$tetraseq} +
$tetrahash{$revword};
                                }
                        }
                }
#add combined relative frequencies to hash of TNF vectors
                foreach $tetraseq (sort keys %endhash) {
                        push (@{$TNFhash{$seqID}}, $endhash{$tetraseq});
                }
        }
        print "Tetranucleotide Frequencies done...\n";
}

#Find length of shortest sequence
my $shortestLength = 0;
my $shortestSeq;
foreach $seq (@seq_array) {
        my $seqLength = $seq->length;
        if ($shortestLength == 0) {
                $shortestLength = $seqLength;
                $shortestSeq = $seq->id;
        }
        else {
```

```
                if ($seqLength < $shortestLength) {
                        $shortestLength = $seqLength;
                        $shortestSeq = $seq->id;
                }
        }
}


                  #
#OFDEG
my $wordSizeInput = int(0.1 * $shortestLength);
my $wordSize;
my $stepSize = $wordSizeInput;
my $startPos;
my $endPos;
my $truncSeq;
my @distanceArray;
my $dist;
my $EucDist;
my $iteration;
my $sample;
my $totalED;
my $meanDist;
my %meanDistsHash;
my $index;
my $totalMeanDist;
my $meanOfMeans;
my @MOMArray;
my @wordSizeArray;

my %OFDEGhash;
if ($selectedFeatures{"OFDEG"}==1) {
        foreach $seq (@seq_array) {
                my $seqID = $seq->id;

#create Statistics::Descriptive::Full object for regression analysis
                my $stat = Statistics::Descriptive::Full->new();
                $iteration=1;

#for set number of resamples...
                while ($iteration <= $resamplingCutoff) {
```

```
#if sequence is longer than the shortest sequence in the dataset, take sample of
shortest sequence length from the sequence
                        if ($seq->length > $shortestLength) {
                                $startPos = int(rand($seq->length - $shortestLength));
                                if ($startPos == 0) {
                                        $startPos = 1;
                                }
                        }
                        else {
                                $startPos=1;
                        }
                        $endPos = $startPos + $shortestLength - 1;
                        $truncSeq = $seq->trunc($startPos, $endPos);
                        my $truncSeqWord = Bio::Tools::SeqWords->new(-seq =>
$truncSeq);

#count tetranucleotides in sampled sequence
                        my $tetraLength = 4;
                        my $revtruncSeq = $truncSeq->revcom;
                        my $revTruncWord = Bio::Tools::SeqWords->new(-seq =>
$revtruncSeq);
                        my $tetracount = $truncSeqWord->count_overlap_words
($tetraLength);
                        my %tetrahash = %$tetracount;
                        my $revtetcnt = $revTruncWord->count_overlap_words
($tetraLength);
                        my %revtethash = %$revtetcnt;
                        my $truncSeqLength=$truncSeq->length;
                        my $totalTetranucs = $truncSeqLength-3;
                        my %mergedtethash = ();
                        foreach $tetraseq(@tetra_array) {
                                if (exists($tetrahash{$tetraseq})) {
                                        $tetrahash{$tetraseq} = $tetrahash{$tetraseq};
                                }
                                else {
                                        $tetrahash{$tetraseq} = 0;
                                }
                                if (exists($revtethash{$tetraseq})) {
                                        $revtethash{$tetraseq} = $revtethash{$tetraseq};
```

```
                        }
                        else {
                                $revtethash{$tetraseq} = 0;
                        }
                }
                foreach $tetraseq(@tetra_array) {
                        $mergedtethash{$tetraseq} = ($tetrahash{$tetraseq} +
$revtethash{$tetraseq})/(2*$truncSeqLength);
                }
                $wordSize = $wordSizeInput;

# for sub-sequence sizes up to 80% of shortest sequence length, calculate
tetranucleotide frequency distributions for subsequences starting at word-size length
and increasing by step-size, then calculate and store error as Euclidean distance
between the two distributions
                while ($wordSize <= int(0.8 * $shortestLength)) {
                        $EucDist=0;
                        $totalED=0;
                        $sample=1;

#for set number of samples at each subsequence length...
                        while ($sample <= $samplingDepth) {
                                my $subSeqStart = int(rand($truncSeq->length -
$wordSize + 1));
                                if ($subSeqStart == 0) {
                                        $subSeqStart = 1;
                                }
                                my $subSeqEnd = $subSeqStart + $wordSize-1;
                                my $subSeq = $truncSeq->trunc($subSeqStart,
$subSeqEnd);
                                my $subWords = Bio::Tools::SeqWords->new(-
seq => $subSeq);
                                my $revSub = $subSeq->revcom;
                                my $revSubWords = Bio::Tools::SeqWords->new
(-seq => $revSub);
                                my $subtetracount = $subWords-
>count_overlap_words(4);
                                my %subtetrahash = %$subtetracount;
                                my $revsubtetcount = $revSubWords-
>count_overlap_words(4);
```

```
                my %revsubtethash = %$revsubtetcount;
                my %mergedsubtethash = ();
                my $subSeqLength = $subSeq->length;
                my $subTetranucs = $subSeqLength-3;
                foreach $tetraseq(@tetra_array) {
                        if (exists($subtetrahash{$tetraseq})) {
                                $subtetrahash{$tetraseq} =
$subtetrahash{$tetraseq};
                        }
                        else {
                                $subtetrahash{$tetraseq} = 0;
                        }
                        if (exists($revsubtethash{$tetraseq})) {
                                $revsubtethash{$tetraseq} =
$revsubtethash{$tetraseq};
                        }
                        else {
                                $revsubtethash{$tetraseq} = 0;
                        }
                }
                foreach $tetraseq(@tetra_array) {
                        $mergedsubtethash{$tetraseq} =
($subtetrahash{$tetraseq} + $revsubtethash{$tetraseq})/(2*$subSeqLength);
                }
                foreach $tetraseq(@tetra_array) {
                        $dist = $mergedtethash{$tetraseq} -
$mergedsubtethash{$tetraseq};
                        if ($dist < 0) {
                                $dist = 0 - $dist;
                        }
                        $EucDist = $EucDist + $dist;
                }
                push (@distanceArray,$EucDist);
                $sample++;
        }
        foreach $EucDist(@distanceArray) {
                $totalED = $totalED + $EucDist;
        }
        $meanDist = $totalED/$samplingDepth;
        ${$meanDistsHash{$wordSize}}[$iteration] = $meanDist;
```

```
                        @distanceArray = ();
                        $wordSize = $wordSize + $stepSize;
                }
                $iteration++;
        }

        @wordSizeArray = ();
        @MOMArray = ();

#take mean of errors at each subsequence length and use these values to plot
regression and obtain OFDEG gradient value
        foreach $wordSize(sort {$a <=> $b} keys %meanDistsHash) {
                push (@wordSizeArray,$wordSize);
                $totalMeanDist=0;
                $index=1;
                while ($index <= $resamplingCutoff) {
                        $totalMeanDist = $totalMeanDist + ${$meanDistsHash
{$wordSize}}[$index];
                        $index++;
                }
                $meanOfMeans = $totalMeanDist/$resamplingCutoff;
                push (@MOMArray,$meanOfMeans);
        }
        $stat->add_data(@MOMArray);
        my @fitResults = $stat->least_squares_fit(@wordSizeArray);
        my ($gradient, $correlation, $error) = ($fitResults[1], (0-$fitResults[2]),
$fitResults[3]);

#filter out OFDEG values obtained from regressions with poor correlation
        unless ($correlation <= 0.9) {
                $OFDEGhash{$seqID} = $gradient;
        }
    }
    print "OFDEG done...\n";
}



#InterNucleotide Distances
my %INDhash;
if ($selectedFeatures{"IND"}==1) {
```

```
	my @distVector;
	my $numSeqs = 0;
	my $maxDistTotal = 0;
	foreach $seq (@seq_array) {
		$cutoffLength = 25;
		my $seqID = $seq->id;
		my $fragment = $seq->seq;
		my $seqLength = $seq->length;

# create statistics object for least-squares fit later
		my $stat = Statistics::Descriptive::Full->new();
		my $acount = (($fragment) =~ tr/Aa//)/length($fragment);
		my $ccount = (($fragment) =~ tr/Cc//)/length($fragment);
		my $gcount = (($fragment) =~ tr/Gg//)/length($fragment);
		my $tcount = (($fragment) =~ tr/Tt//)/length($fragment);
		my @counts = ($acount,$ccount,$gcount,$tcount);

#split sequence into individual characters
		my @splitSeq = split //, $fragment;
		my $pos = 0;
		my $dist = 1;
		my $Odist = "";
		my $overlap;

#calculate distances for each position in the sequence
		while ($pos < $seqLength) {
			$Odist = "";
			my $currDist = $pos + $dist;
			if (($pos+$dist) >= length($fragment)) {
			$overlap = $currDist - length($fragment);
			$Odist = 0 - $pos + $overlap;
			$currDist = $pos + $Odist;
		}
		my $posBase = $splitSeq[$pos];
		my $distBase = $splitSeq[$currDist];
		die "posBase undefined for sequence $seqID (length $seqLength)
position $pos distance $dist\n" unless (defined($posBase));
		die "distBase undefined for sequence $seqID (length $seqLength)
position $pos distance $dist\n" unless (defined($distBase));
		if ($posBase eq $distBase) {
```

```
                        if ($Odist ne "") {
                                $dist = length($fragment) + $Odist;
                        }
                        push (@distVector, $dist);
                        $pos++;
                        $dist=1;
                }
                else {
                        $dist++;
                }
        }

#count frequency of distances
        my @distribution=();
        my $maxDist;
        my $currDist;
        foreach $dist(@distVector) {
                if (defined $maxDist) {
                        if ($dist > $maxDist) {
                                $maxDist = $dist;
                        }
                }
                else {
                        $maxDist = $dist;
                }
                my $index=$dist-1;
                $distribution[$index]++;
        }
        my $COF=0;
        my @COF_array=();
        my $distCheck = 0;
        my @limitedDist;

#create distribution of distance frequencies up to cutoff
        while ($distCheck < $cutoffLength) {
                if (defined($distribution[$distCheck])) {
                        push (@limitedDist, $distribution[$distCheck]);
                }
                else {
                        push (@limitedDist, 0);
```

```
		}
		$distCheck++;
	}

#calculate observed relative frequencies
	foreach my $freq(@limitedDist) {
		if (defined $freq) {
			$freq=$freq/$seqLength;
		}
		else {
			$freq=0;
		}

#calculate cumulative observed frquencies
		$COF = $COF + $freq;
		push (@COF_array, $COF);
	}

#calculate expected frequencies and cumulative expected frequencies
	my $exptDist = 1;
	my $exptFreq = 0;
	my @exptDistribution=();
	while ($exptDist <= $cutoffLength) {
		foreach my $nucRF (@counts) {
			$exptFreq = $exptFreq+(($nucRF**2)*((1-$nucRF)**
($exptDist-1)));
		}
		push (@exptDistribution, $exptFreq);
		$exptDist++;
		$exptFreq=0;
	}
	my $CEF=0;
	my @CEF_array=();
	foreach $exptFreq(@exptDistribution) {
		if (defined $exptFreq) {
		}
		else {
			$exptFreq=0;
		}
		$CEF = $CEF+$exptFreq;
```

```
                push (@CEF_array, $CEF);
        }

#calculate difference between observed and expected frequencies and determine
largest for K-S distance
        my $currIndex=0;
        my @CFdiff_array=();
        $maxDistTotal = $maxDistTotal+$maxDist;
        while ($currIndex < 25) {
                my $CFdiff = $CEF_array[$currIndex] - $COF_array[$currIndex];
                if ($CFdiff < 0) {
                        $CFdiff = 0 - $CFdiff;
                }
                push (@CFdiff_array, $CFdiff);
                $currIndex++;
        }
        my @CFD_sorted = sort {$a<=>$b} @CFdiff_array;
        my $KSdist = $CFD_sorted[$currIndex-1];
        $INDhash{$seqID} = $KSdist;
        @distVector=();
}
print "Internucleotide Distances done...\n";
}

#print feature values to file using OFDEG filter...
if ($selectedFeatures{"OFDEG"}==1) {
        foreach $seq (@seq_array) {
                my $seqID = $seq->id;
                if (exists ($OFDEGhash{$seqID})) {
                        print STATSFILE ("$seqID\t$OFDEGhash{$seqID}");
                        if ($selectedFeatures{"IND"}==1) {
                                print STATSFILE ("\t$INDhash{$seqID}");
                        }
                        if ($selectedFeatures{"GC"}==1) {
                                print STATSFILE ("\t$GChash{$seqID}");
                        }
                        if ($selectedFeatures{"TNF"}==1) {
                                foreach my $TNF (@{$TNFhash{$seqID}}) {
                                        print STATSFILE ("\t$TNF");
                                }
```

```
                        }
                        print STATSFILE ("\n");
                }
        }
}
#...or without filter if OFDEG values have not been produced
else {
        foreach $seq (@seq_array) {
                my $seqID = $seq->id;
                print STATSFILE ("$seqID");
                if ($selectedFeatures{"IND"}==1) {
                print STATSFILE ("\t$INDhash{$seqID}");
                }
                if ($selectedFeatures{"GC"}==1) {
                        print STATSFILE ("\t$GChash{$seqID}");
                }
                if ($selectedFeatures{"TNF"}==1) {
                        foreach my $TNF (@{$TNFhash{$seqID}}) {
                                print STATSFILE ("\t$TNF");
                        }
                }
                print STATSFILE ("\n");
        }
}
close STATSFILE;
```

Appendix A-5

featureComboWriter.pl

```
#! /usr/bin/perl
use warnings;
use strict;

#input list of FASTA files to produce feature vectors from, and generate array of
filenames
my $filesListFile = shift;
my @filesList;
my $file;
open (FLFH, "<$filesListFile");
while ($file = (<FLFH>)) {
	push (@filesList, $file);
}

#prepare array of feature types and combinations
my @featuresList = ("GC", "IND", "OFDEG", "TNF");
#feature combo list - can be edited if only a selection of combinations are required
my @featCombos = ("GC\, IND", "GC\, OFDEG", "GC\, TNF", "IND\, OFDEG", "IND\,
TNF", "OFDEG\, TNF", "GC\, IND\, OFDEG", "GC\, IND\, TNF", "GC\, OFDEG\, TNF", "IND
\, OFDEG\, TNF", "GC\, IND\, OFDEG\, TNF");
my @fullFL;
push (@fullFL, @featuresList);
push (@fullFL, @featCombos);
my $feature;
my $featCombo;

#for each file...
foreach $file (@filesList) {
	my @fileSplit = split /\./, $file;
#run featureWriterMulti.pl for each feature type, to generate individual feature files -
comment out if feature files have been generated already
	foreach $feature (@featuresList) {
		my @featArgs = ("perl", "featureWriter.pl", $file, $feature);
		system(@featArgs);
	}
	my %featValuesHash;
	my $ID;
```

```
	my @values;
#read files and prepare hashes for individual feature types
	foreach $feature (@featuresList) {
		my $featFile = $fileSplit[0] . $feature . ".txt";
		open (FEATFH, "<$featFile");
		my $line;
		while ($line = <FEATFH>) {
			chomp $line;
			@values = split /\s/, $line;
			$ID = shift(@values);
			$featValuesHash{$ID}{$feature} = [ @values ];
		}
	}
#generate feature combos
	foreach $featCombo (@featCombos) {
		my @featsList = split /\,\s/, $featCombo;
		my $featsString = join "", @featsList;
		my $outfile = $fileSplit[0] . $featsString . ".txt";
		open (OUTFH, ">$outfile");
		my $OFDEG = 0;
		foreach my $feat (@featsList) {
			if ($feat eq "OFDEG") {
				$OFDEG = 1;
			}
		}
#if OFDEG is included in feature types for combination, use the IDs from the OFDEG
file, to avoid inclusion of those sequences that failed the OFDEG R^2 threshold test
		foreach $ID (sort keys %featValuesHash) {
			if ($OFDEG == 1) {
				if (exists($featValuesHash{$ID}{"OFDEG"})) {
					print OUTFH ("$ID");
					foreach my $feat (@featsList) {
						foreach my $value (@{$featValuesHash
{$ID}{$feat}}) {
							print OUTFH ("\t$value");
						}
					}
					print OUTFH ("\n");
				}
			}
```

```
                        else {
                                print OUTFH ("$ID");
                                foreach my $feat (@featsList) {
                                        foreach my $value (@{$featValuesHash{$ID}
{$feat}}) {
                                                print OUTFH ("\t$value");
                                        }
                                }
                                print OUTFH ("\n");
                        }
                }
        }
}
```

Appendix A-6

claraAnalysisMulti.pl

```
#! /usr/bin/perl
use warnings;
use strict;
use lib "/usr/lib/perl5/site_perl/5.8.8";
use Statistics::R;
use File::Basename;

#import the desired number of clusters to be produced by CLARA and a text file
containing a list of feature vector files to be clustered
my $numClusters = shift;
my $statsFileList = shift;
my @statsFiles;

#generate array of input filenames
open (LISTFH, "<$statsFileList");
while (<LISTFH>) {
        push (@statsFiles, $_);
}

#for each input file...
foreach my $statsfile (@statsFiles) {

#prepare filename for generation of output cluster files
        my @nameext = split /\./, $statsfile;
        my $name = $nameext[0];

#make new R object
        my $R = Statistics::R->new;

#start R and, if range of number of clusters has been input, determine optimal number
of clusters within range based on mean silhouette width of clusters
        $R->startR;
        if ($numClusters =~ m/\d\:\d/) {
                my $rangeUL;
                if ($numClusters =~ m/1\:\d/) {
                        die "\a1 is not a valid number of clusters!\n";
                        my @rangeSplit = split /\:/, $numClusters;
```

```
                        $rangeSplit[0] = 2;
                        $rangeUL = $rangeSplit[1];
                        $numClusters = join ':', @rangeSplit;
                        print "Finding optimal number of clusters in range $numClusters
\n";
                }

#run CLARA in R with optimal number of clusters...
                $R->send(qq`setwd(\"/var/www/html/toby\") \n library(cluster) \n
f=file(\"$statsfile\", open=\"r\") \n t=read.table(f, row.names=1) \n x=data.frame(t)
\n asw<-numeric($rangeUL) \n for (k in $numClusters) \n asw[k] <- clara(x, k) \$
silinfo \$ avg.width \n k.best<-which.max(asw) \n print(k.best)`);
                my $optClusters = $R->read;
                $optClusters =~ s/\[1\]\s//;
                print "Optimal number of clusters in range $numClusters: $optClusters
\n";
                $numClusters = $optClusters;
                $R->stopR();
        }

#...or with input number of clusters, and print sequence IDs to separate files for each
cluster.
        print "\nRunning clara with $numClusters clusters.\n";
        $R->startR;
        $R->send(qq`setwd(\"/tf/people/tah501\") \n library(cluster) \n f=file
(\"$statsfile\", open=\"r\") \n t=read.table(f, row.names=1) \n df=data.frame(t) \n
clarax <- clara(df, $numClusters) \n clusdf <- data.frame(clarax\$clustering) \n
clusSizes = 1:$numClusters \n clusSizes[1:$numClusters]=0 \n for (i in
1:$numClusters) { \n clustFile = paste("$name", "cluster", i, ".txt", sep="") \n x = which
(clusdf==i) \n y=row.names(clusdf)[x] \n clusterNames = data.frame(y) \n write.table
(clusterNames, file=clustFile, col.names=F, row.names=F, quote=F) \n clusSizes[i]
=length(x) \n } \n clusSizes=data.frame(clusSizes) \n for (j in 1:$numClusters) { \n
row.names(clusSizes)[j] = paste("clus", j, sep="") \n } \n print(clusSizes) \n`);
        my $returnedValue = $R->read;
        $R->stopR();
        $returnedValue =~ s/clus\d+\s+//g;
        my @clusterSizes = split /\n/, $returnedValue;
        shift(@clusterSizes);

#print the number of reads that were grouped into each cluster
```

```
        print "\nCluster Sizes:\n";
        my $currClust = 1;
        foreach my $clusterSize (@clusterSizes) {
                print "Cluster $currClust:\t$clusterSize Sequences\n";
                $currClust++;
        }
}
```

Appendix A-7

claraResultsSummariser.pl

```
#! /usr/bin/perl
use warnings;
use strict;
use lib "/usr/lib/perl5/site_perl/5.8.8";
use Bio::SeqIO;
use Bio::Tools::SeqWords;
use File::Basename;

#import input number of clusters produced in CLARA, a preference for whether to print to output files the total numbers of reads from each species on each cluster ([T/TRUE]/[F/FALSE]), the shared section of the cluster file names (without the features, a number and ".txt" after), and the features clustered
my $numClusters = shift;
my $totalsPrint = shift;
my $commonName = shift;
my $feats = shift;

#check that a number of clusters has been entered
unless ($numClusters =~ /\d+/) {
        print "\a\nYou must specify the number of clusters generated by clara!  Please enter number of clusters:  ";
        $numClusters = <STDIN>;
}

#check that a common name for the cluster files has been entered
if (defined ($commonName)) {
}
else {
print "\nPlease enter the common segment of file name for each cluster file.  For example, if the clusters are called \'ExampleCluster1.txt\', \'ExampleCluster2.txt\' etc, enter \'ExampleCluster\' here...  ";
$commonName = <STDIN>;
chomp $commonName;
}

#for each cluster, open an output file and read the lines from the cluster files, counting the numbers of sequences from each species in each cluster
```

```
my $i = 1;
my %clusMaxima;
my %speciesTotals;
my %clusDomSp;
my %clusPrecisions;
my $outFile = ("$commonName" . "$feats" . "ClusStats.txt");
open (OUTFILE, ">$outFile");
while ($i <= $numClusters) {
	my $clusFile = ("$commonName" . "$feats" . "cluster$i.txt");
	open (CLUSTERFILE, "<$clusFile") or die "$!";
	my @lines = <CLUSTERFILE>;
	my @speciesLabels;
	my %speciesPresent=();
	my $species;
	foreach my $line (@lines) {
		chomp $line;
		@speciesLabels = split /\d/, $line;
		$species = $speciesLabels[0];

#species defined by first six characters of ID string, allowing for differentiation
between reads from different experiments
		$species = substr $species, 0, 6;
		if (exists ($speciesPresent{$species})) {
			$speciesPresent{$species}++;
		}
		else {
			$speciesPresent{$species}=1;
		}
	}
#'flip' the %speciesPresent hash, so that the predominant species in the cluster can be
determined later
	my %speciesSeqs = reverse %speciesPresent;
	my $seqNo;
	my @seqNos;

#for each species present in the cluster, add the read counts to the respective totals
for the species - for recall calculation later
	foreach $species (sort keys %speciesPresent) {
		$seqNo = $speciesPresent{$species};
		push (@seqNos, $seqNo);
```

```
                if (exists ($speciesTotals{$species})) {
                        $speciesTotals{$species}=$speciesTotals{$species}+
$speciesPresent{$species};
                }
                else {
                        $speciesTotals{$species}=$speciesPresent{$species};
                }
        }
#if $totalsPrint was defined as 'TRUE' or 'T', print the species totals to the output files
        if (defined ($totalsPrint)) {
                if ($totalsPrint eq "T") {
                        print OUTFILE ("Cluster$i:  \n");
                        foreach $species (sort keys %speciesPresent) {
                                print OUTFILE ("$species\t$speciesPresent{$species}\n");
                        }
                        print OUTFILE ("\n");
                }
                elsif ($totalsPrint eq "TRUE") {
                        print OUTFILE ("Cluster$i:  \n");
                        foreach $species (sort keys %speciesPresent) {
                                print OUTFILE ("$species\t$speciesPresent{$species}\n");
                        }
                        print OUTFILE ("\n");
                }
        }

#find the largest number of reads for all species in the cluster
        my @sortedSNs = sort {$b <=> $a} @seqNos;
        my $maxSeqNo = $sortedSNs[0];

#define cluster species from this maximum read count
        my $clusSpecies = $speciesSeqs{$maxSeqNo};
        my $clusTotal=0;
        my $clusterName = ("Cluster " . $i);

#store maximum read count for cluster
        $clusMaxima{$clusterName} = $maxSeqNo;

#store name of predominant species for cluster
        $clusDomSp{$clusterName} = $clusSpecies;
```

```
#calculate total number of reads in cluster – for calculation of precision
        foreach $seqNo (@seqNos) {
                $clusTotal = $clusTotal+$seqNo;
        }

#calculate precision of cluster to 4dp
        my $precision = sprintf ("%.4f", $maxSeqNo/$clusTotal);
        $clusPrecisions{$clusterName} = $precision;
        close CLUSTERFILE;
        $i++;
}

#print precision and recall values to output file
print OUTFILE ("$feats\nPr\tRc\n");
foreach my $clust (sort keys %clusMaxima) {
        my $clusSpecies = $clusDomSp{$clust};
        my $recall = sprintf ("%.4f", $clusMaxima{$clust}/$speciesTotals{$clusSpecies});
        print OUTFILE ("$clusPrecisions{$clust}\t$recall\n");
}
close OUTFILE;
```

Appendix A-8

avePRwriter.pl

```
#! /usr/bin/perl
use warnings;
use strict;
use lib '/usr/local/perllibPDA';

#import name of FASTA file, features used, and number of clusters
my $fastaFile = shift;
my $feats = shift;
my @fastaSplit = split /\./, $fastaFile;
my $name = $fastaSplit[0];
my $numClusters = shift;
my $prFile = "$name" . "$feats" . "ClusStats.txt";
my $outFile = "$name" . "AvePRSummary.txt";

#read PR file for features
open (INFH, "<$prFile");
my $line;
my @lines;
while ($line = (<INFH>)) {
        push (@lines, $line);
}
close INFH;

#remove first two lines of file - these are title lines
shift (@lines);
shift (@lines);
my @precisionValues;
my $precisionValue;
my @recallValues;
my $recallValue;

#read precision and recall values from file
foreach $line (@lines) {
        my @prValues = split /\s/, $line;
        $precisionValue = $prValues[0];
        $recallValue = $prValues[1];
        push (@precisionValues, $precisionValue);
        push (@recallValues, $recallValue);
```

```
}

#calculate mean precision and recall values and standard deviations to 4dp
my $totPrecision = 0;
my $totRecall = 0;
foreach $precisionValue (@precisionValues) {
        $totPrecision = $totPrecision + $precisionValue;
}
foreach $recallValue (@recallValues) {
        $totRecall = $totRecall + $recallValue;
}
my $avePrecision = sprintf ("%.4f", $totPrecision/$numClusters);
my $aveRecall = sprintf ("%.4f", $totRecall/$numClusters);
my $pStDev = sqrt((($totPrecision*2)/$numClusters) - ($avePrecision*$avePrecision));
my $rStDev = sqrt((($totRecall*2)/$numClusters) - ($aveRecall*$aveRecall));

#print features, mean values and standard deviations to output file, amending file
rather than overwriting it
open (OUTFH, ">>$outFile");
print OUTFH ("$feats\t$avePrecision\t\($pStDev\)\t$aveRecall\t\($rStDev\)\n");
close OUTFH;
```

Appendix A-9

SAMseqAssigner.pl

```
#! /usr/bin/perl
use strict;
use warnings;
use Bio::SeqIO;

#import SAM filename, a file of keys converting accession numbers to species names,
and the file of sequences that were aligned by SSAHA2
my $samFile = shift;
my $keyFile = shift;
my $seqFile = shift;
my @SFsplit = split /Mapped\./, $samFile;

#initialise output filenames
my $outFile = $SFsplit[0] . "Assignments.txt";
my $poorAlignFile = "PoorlyAlignedSeqs.txt";
my $unalignedFile = "UnmappedSeqs.txt";

#generate Bio::Seq object for each sequence in $seqFile
my @IDarray;
my $seq_in = Bio::SeqIO->new(-file => $seqFile, -format => "fasta");
while (my $seq = $seq_in->next_seq) {
        my $seqID = $seq->id;
        push (@IDarray, $seqID);
}

#generate conversion hash for accession number -> species from $keyFile
my ($keyLine, @keyLines);
open (KEYFH, "<$keyFile");
while ($keyLine = (<KEYFH>)) {
        chomp $keyLine;
        push (@keyLines, $keyLine);
}
my ($accNo, $species, %keyHash);
foreach $keyLine (@keyLines) {
        ($accNo, $species) = split /\t/, $keyLine;
        $keyHash{$accNo} = $species;
}
```

```
#read lines from SAM file
my ($samLine, @samLines);
open (SAMFH, "<$samFile");
while ($samLine = (<SAMFH>)) {
	push (@samLines, $samLine);
}

#store reference sequence from top hit for each query sequence in SAM file, or label as
unassigned if the best alignment score is <40
my ($qName, $flag, $rName, $pos, $mapQ, $cigar, $rNext, $pNext, $tLen, $seq,
$qual, $aScore, $mScore, %alignmentHash, $rAcc, $aTag, $aType);
foreach $samLine (@samLines) {
	($qName, $flag, $rName, $pos, $mapQ, $cigar, $rNext, $pNext, $tLen, $seq,
$qual, $aScore, $mScore) = split /\t/, $samLine;
	unless (defined($alignmentHash{$qName})) {
		($aTag, $aType, $aScore) = split /\:/, $aScore;
		if ($aScore >= 40) {
			$rName =~ m/(NC\_\d+)/;
			$rAcc = $1;
			$alignmentHash{$qName} = $rAcc;
		}
		else {
			$alignmentHash{$qName} = "No significant alignment found in
reference database (p < 0.0001) -> Alignment Score = $aScore";
		}
	}
}

#open output files and print assignments (or lack of) to output file, and print
unassigned (due to score cutoff, rather than lack of hits) query sequence IDs to a
separate file ($poorAlignFile)
open (OUTFH, ">$outFile");
open (PAFH, ">$poorAlignFile");
my ($query, $refAcc, $assignment);
foreach $query (keys %alignmentHash) {
	if ($alignmentHash{$query} =~ m/NC\_\d+/) {
		$refAcc = $alignmentHash{$query};
		$assignment = $keyHash{$refAcc};
	}
```

```
        else {
                $assignment = $alignmentHash{$query};
                print PAFH ("$query\n");
        }
        print OUTFH ("$query\t$assignment\n");
}
close OUTFH;
close PAFH;

#open output file for query sequence IDs for which no hits were returned at all
open (UAFH, ">$unalignedFile");
foreach my $ID (@IDarray) {
        if (exists($alignmentHash{$ID})) {
        }
        else {
                print UAFH ("$ID\n");
        }
}
close UAFH;
```

Appendix A-10

partClustering.pl

```
#! /usr/bin/perl
use strict;
use warnings;
use Statistics::R;
use Cwd;

#input a tab-delimited feature vector text file, number of clusters, method (can be
"kmeans", "fuzzyk", "clara", "k", "f", "c"), and a fuzziness measure if using FCM
my $dataFile = shift or die "\aYou must provide a data file for clustering.\n";
my $numClusters = shift or die "\aYou must specify a desired number of clusters to be
produced.\n";
my $method = shift or die "\aYou must specify a clustering method.\n";
my $fuzziness = shift;
my @methodList = ("kmeans", "fuzzyk", "clara", "k", "f", "c");

#check input number of clusters is greater than 1
if ($numClusters < 2) {
	die "\aNumber of clusters must be greater than 1!\n";
}

#check that method entered is valid, and that only one was entered
$method = lc($method);
my $monitor = 0;
foreach my $listedMethod (@methodList) {
	if ($method eq $listedMethod) {
		$monitor++;
	}
}
unless ($monitor==1) {
	print "\aYou must enter one valid clustering method name from:\n";
	foreach my $methString (@methodList) {
		print "$methString\n";
		die;
	}
}

#check for fuzziness value if FCM was chosen method
```

```
if ($method eq "fuzzyk" || $method eq "f") {
	unless (defined($fuzziness)) {
		die "\aYou must define a degree of fuzziness (any value \> 1) for fuzzy
c means clustering.\n";
	}
}

#run appropriate method and generate output of files listing sequence IDs in each
cluster
my $outFile;
my @dataFileSplit = split /\./, $dataFile;
my $name = $dataFileSplit[0];
my $wd = getcwd();
my $path = $name;
my @pathSplit = split /\//, $path;
my $fileName = pop(@pathSplit);
my $fullFileName = join "\.", ($fileName, $dataFileSplit[-1]);
push(@pathSplit, "");
my @fullPath = ($wd, @pathSplit);
my $dir = join "/", @fullPath;
my $R = Statistics::R->new();
$R->startR;
if ($method eq "kmeans" || $method eq "k") {
	my $methFolder = "KMClusters";
	my $clusFolder = $numClusters . "C";
	$outFile = $name . "kMeans.txt";
	$R->send(qq`setwd(\"$dir") \n library(stats) \n f=file(\"$fullFileName\", open=
\"r\") \n t=read.table(f, row.names=1) \n df=data.frame(t) \n  KMX <- kmeans(df,
$numClusters, iter\.max=50) \n clusdf <- data.frame(KMX\$cluster) \n clusSizes =
1:$numClusters \n clusSizes[1:$numClusters]=0 \n for (i in 1:$numClusters) { \n
clustFile = paste("$fileName", "KMcluster", i, ".txt", sep="") \n x = which(clusdf==i) \n
y=row.names(clusdf)[x] \n clusterNames = data.frame(y) \n write.table(clusterNames,
file=clustFile, col.names=F, row.names=F, quote=F) \n clusSizes[i]=length(x) \n } \n
clusSizes=data.frame(clusSizes) \n for (j in 1:$numClusters) { \n row.names(clusSizes)
[j] = paste("clus", j, sep="") \n } \n test = "test" \n print(test) \n`);
	my $returnedValue = $R->read;
	print "$returnedValue\n";
	print $methFolder . "/" . $clusFolder . "/" . $name . "KMcluster\n";
	$R->stopR();
}
```

```
if ($method eq "fuzzyk" || $method eq "f") {
        my $methFolder = "FCMClusters";
        my $clusFolder = $numClusters . "C";
        my $fuzzFolder = $fuzziness;
        $outFile = $name . "FuzzykMeans.txt";
        $R->send(qq`setwd(\"$dir") \n library(e1071) \n library(class) \n f=file
(\"$fullFileName\", open=\"r\") \n t=read.table(f, row.names=1) \n df=data.frame(t)
\n  fuzzyCM <- cmeans(df, $numClusters, m=$fuzziness) \n clusdf <- data.frame
(fuzzyCM\$cluster) \n clusSizes = 1:$numClusters \n clusSizes[1:$numClusters]=0 \n
for (i in 1:$numClusters) { \n clustFile = paste("$fileName", "FCMcluster", i, ".txt",
sep="") \n x = which(clusdf==i) \n y=row.names(clusdf)[x] \n clusterNames =
data.frame(y) \n write.table(clusterNames, file=clustFile, col.names=F, row.names=F,
quote=F) \n clusSizes[i]=length(x) \n } \n clusSizes=data.frame(clusSizes) \n for (j in
1:$numClusters) { \n row.names(clusSizes)[j] = paste("clus", j, sep="") \n } \n test =
"test" \n print(test) \n`);
        my $returnedValue = $R->read;
        print "$returnedValue\n";
        print $methFolder . "/" . $clusFolder . "/" . $fuzzFolder . "/" . $name .
"FCMcluster\n";
        $R->stopR();
}
if ($method eq "clara" || $method eq "c") {
        my $methFolder = "CLARAClusters";
        my $clusFolder = $numClusters . "C";
        $outFile = $name . "CLARA.txt";
        $R->send(qq`setwd(\"$dir") \n library(cluster) \n f=file(\"$fullFileName\",
open=\"r\") \n t=read.table(f, row.names=1) \n df=data.frame(t) \n clarax <- clara(df,
$numClusters, samples = 50) \n clusdf <- data.frame(clarax\$clustering) \n clusSizes
= 1:$numClusters \n clusSizes[1:$numClusters]=0 \n for (i in 1:$numClusters) { \n
clustFile = paste("$fileName", "CLARAcluster", i, ".txt", sep="") \n x = which(clusdf==i)
\n y=row.names(clusdf)[x] \n clusterNames = data.frame(y) \n write.table
(clusterNames, file=clustFile, col.names=F, row.names=F, quote=F) \n clusSizes[i]
=length(x) \n } \n clusSizes=data.frame(clusSizes) \n for (j in 1:$numClusters) { \n
row.names(clusSizes)[j] = paste("clus", j, sep="") \n } \n test = "test" \n print(test) \n`);
        my $returnedValue = $R->read;
        print $methFolder . "/" . $clusFolder . "/" . $name . "CLARAcluster\n";
        print "$returnedValue\n";
        $R->stopR();
}
```

Appendix A-11

contigInfo.pl

```
#! /usr/bin/perl
use warnings;
use strict;

#input 454Contigs.ace and 454readStatus.txt files for assembly
my $inFile = shift;
my $idFile = shift;

#get all sequence IDs according to file format
#my @seqFileSplit = split /\./, $seqFile;
#my (@seqIDs, $seqID);
#if ($seqFileSplit[-1] eq "fasta") {
#	open FASTAGREP, "grep ^> $seqFile |" or die "can't fork: $!";
#	while (<FASTAGREP>) {
#		my $IDline = $_;
#		chomp $IDline;
#		$seqID = $IDline =~ s/>//;
#		push (@seqIDs, $seqID);
#	}
#}
#elsif ($seqFileSplit[-1] eq "fastq") {
#	open FASTQGREP, "grep ^@ $seqFile |" or die "can't fork: $!";
#	while (<FASTQGREP>) {
#		my $IDline = $_;
#		chomp $IDline;
#		$seqID = $IDline =~ s/@//;
#		push (@seqIDs, $seqID);
#	}
#}

#use 454ReadStatus.txt file to record status of each read
open (IDFH, "<$idFile");
my (@idLines, $idLine, $id, $status, %reads);
while (<IDFH>) {
	$idLine = $_;
	chomp $idLine;
	push (@idLines, $idLine);
```

```
}
my $junkHeading = shift(@idLines);
foreach $idLine (@idLines) {
	my @IDLsplit = split /\t/, $idLine;
	$id = $IDLsplit[0];
	$status = $IDLsplit[1];
	$reads{$id} = $status;
}
close IDFH;

my @seqInfoLines;

#grep all CO contig header lines
open COGREP, "grep ^CO $inFile |" or die "can't fork: $!";
while (<COGREP>) {
	push (@seqInfoLines, $_);
}
close COGREP;
my (%contigs, $contigID, $length, @readsUsed, @cIDs, @cReads);

my @lengths = ();
#initialise contig info hash and add lengths
foreach my $SIline (@seqInfoLines) {
	my @siSplit = split /\s/, $SIline;
	$contigID = $siSplit[1];
	push (@cIDs, $contigID);
	$length = $siSplit[2];
	push (@readsUsed, $siSplit[3]);
	$contigs{$contigID} = {"Length" => $length};
	push (@lengths, $length);
}

#create cumulative read counts
my $rInd=0;
foreach my $count (@readsUsed) {
	unless ($rInd == 0) {
		$count = $count + $cReads[$rInd-1];
	}
	push (@cReads, $count);
	$rInd++;
```

```
}

#grep lists of reads used in each contig
open AFGREP, "grep ^AF $inFile |";
my $index = 0;
my @contigReads = ();
my @foundReads = ();
while (<AFGREP>) {
        my $readLine = $_;
        chomp $readLine;
        my @rlSplit = split /\s/, $readLine;
        my $readID = $rlSplit[1];
        push (@contigReads, $readID);
        push (@foundReads, $readID);
        if ($. == $cReads[$index]) {
                $contigs{$cIDs[$index]}{"Reads"} = [@contigReads];
                @contigReads = ();
                $index++;
        }
}
close AFGREP;

my $tooShort  = 0;
my $partAss   = 0;
my $assembled = 0;
my $singleton = 0;
my $outlier   = 0;
my $repeat    = 0;
foreach $id (keys %reads) {
        if ($reads{$id} eq "TooShort") {
                $tooShort++;
        }
        elsif ($reads{$id} eq "Repeat") {
                $repeat++;
        }
        elsif ($reads{$id} eq "PartiallyAssembled") {
                $partAss++;
        }
        elsif ($reads{$id} eq "Assembled") {
                $assembled++;
```

```
		}
		elsif ($reads{$id} eq "Singleton") {
			$singleton++;
		}
		elsif ($reads{$id} eq "Outlier") {
			$outlier++;
		}
}
#print read stats
print "Number of assembled reads:\t$assembled\n";
print "Number of partially assembled reads:\t$partAss\n";
print "Number of singleton reads:\t$singleton\n";
print "Number of repeat reads:\t$repeat\n";
print "Number of outlier reads:\t$outlier\n";
print "Number of reads too short for assembly:\t$tooShort\n\n";

#calculate mean contig length
my $totalLength = 0;
my $meanLength = 0;
my $numCtgs = 0;
foreach $contigID (@cIDs) {
		foreach my $key (sort keys %{$contigs{$contigID}}) {
			if ($key eq "Length") {
				$totalLength = $totalLength + $contigs{$contigID}{$key};
				$numCtgs++;
			}
		}
}
#calculate N50 length & score
my $sumContigs = $totalLength;
my $N50length;
my $N50score = 0;
my $cumLength = 0;
my @sortedLengths = sort {$b <=> $a} @lengths;
foreach my $sLength (@sortedLengths) {
		unless ($cumLength > ($sumContigs/2)) {
			$N50length = $sLength;
			$cumLength = $cumLength + $sLength;
			$N50score++;
		}
```

```
}
$meanLength = $totalLength/$numCtgs;
#print overall contig stats
print "Cumulative length of all contigs:\t$sumContigs bp\n";
print "Mean length of all contigs:\t$meanLength bp\n";
print "N50 length of all contigs:\t$N50length bp\n";
print "N50 score of all contigs:\t$N50score\n\n";

#print species-specific contig attributes
print "Contig Info:\n\n";
my %specContigLengths;
my %specContigCounts;
my %specLengthArrays;
foreach $contigID (@cIDs) {
	my %specReads;
	foreach my $key (sort keys %{$contigs{$contigID}}) {
		if ($key eq "Reads") {
			foreach my $readID (@{$contigs{$contigID}{$key}}) {
				my $idString = substr($readID, 0, 5);
				if (exists($specReads{$idString})) {
					$specReads{$idString}++;
				}
				else {
					$specReads{$idString}=1;
				}
			}
			my $specCount = 0;
			my $specID;
			foreach my $id (keys %specReads) {
				$specID = $id;
				$specCount++;
			}
			if ($specCount == 1) {
				if (exists($specContigLengths{$specID})) {
					$specContigLengths{$specID} =
$specContigLengths{$specID} + $contigs{$contigID}{"Length"};
					$specContigCounts{$specID}++;
					push (@{$specLengthArrays{$specID}}, $contigs
{$contigID}{"Length"});
				}
```

```
                                else {
                                        $specContigLengths{$specID} = $contigs
{$contigID}{"Length"};
                                        $specContigCounts{$specID} = 1;
                                        $specLengthArrays{$specID} = ();
                                        $specLengthArrays{$specID} = [$contigs
{$contigID}{"Length"}];
                                }
                        }
                        else {
                                if (exists($specContigLengths{"Hybrid"})) {
                                        $specContigLengths{"Hybrid"} =
$specContigLengths{"Hybrid"} + $contigs{$contigID}{"Length"};
                                        $specContigCounts{"Hybrid"}++;
                                        push (@{$specLengthArrays{"Hybrid"}}, $contigs
{$contigID}{"Length"});
                                }
                                else {
                                        $specContigLengths{"Hybrid"} = $contigs
{$contigID}{"Length"};
                                        $specContigCounts{"Hybrid"} = 1;
                                        $specLengthArrays{"Hybrid"} = ();
                                        $specLengthArrays{"Hybrid"} = [$contigs
{$contigID}{"Length"}];
                                }
                        }
                }
                else {
#                       print "$key\t$contigs{$contigID}{$key}\n\n";
                }
        }
}
foreach my $contigLabel (keys %specContigLengths) {
        my @specLengths = @{$specLengthArrays{$contigLabel}};
        my @lengthsSorted = sort {$b <=> $a} @specLengths;
        my $spN50length;
        my $spN50score = 0;
        my $spCumLength = 0;
        my $spSumContigs = $specContigLengths{$contigLabel};
        foreach my $spLength (@lengthsSorted) {
```

```
                unless ($spCumLength > ($spSumContigs/2)) {
                        $spN50length = $spLength;
                        $spCumLength = $spCumLength + $spLength;
                        $spN50score++;
                }
        }
        my $meanSpecLength = $specContigLengths{$contigLabel} /
$specContigCounts{$contigLabel};
        print $contigLabel, ":\t", $specContigLengths{$contigLabel}, " bp in\t",
$specContigCounts{$contigLabel}, " contigs\tmean length:\t", $meanSpecLength, " bp
\tN50 length:\t$spN50length\tN50 score:\t$spN50score\n";
}
```

Appendix A-12

randomSeqFetcher.pl

```
#! /usr/bin/perl
use strict;
use warnings;

#collect input of ratios, fasta & quality file, and determine number of clusters
my $ratios = shift;
chomp $ratios;
my @ratios = split /,/, $ratios;
foreach my $r (@ratios) {
	print "$r\n";
}
my $numClusters = scalar(@ratios);
my $fastaFile = shift;
my $fastqFile = shift;

my %fasta;
my %fastq;
my $ID;
my $seqString;
my $qualString;
my $seqLine;
my $qualLine;
my @seqLines = ();
my @qualLines = ();

#store FASTA IDs, and sequences combined line-by-line
open (FASTAFH, "<$fastaFile");
while (<FASTAFH>) {
	if ($_ =~ />/) {
		my $seqLineCount = scalar(@seqLines);
		unless ($seqLineCount == 0) {
			$seqString = join "", @seqLines;
			$fasta{$ID} = $seqString;
		}
		$ID = $_;
		chomp $ID;
		@seqLines = ();
```

```
        }
        else {
                $seqLine = $_;
                chomp $seqLine;
                push(@seqLines, $seqLine);
        }
}
$seqString = join "", @seqLines;
$fasta{$ID} = $seqString;
close FASTAFH;

#store QUAL info
open (FASTQFH, "<$fastqFile");
while (<FASTQFH>) {
        if ($_ =~ />/) {
                my $qualLineCount = scalar(@qualLines);
                unless ($qualLineCount == 0) {
                        $qualString = join "", @qualLines;
                        $fastq{$ID} = $qualString;
                }
                $ID = $_;
                chomp $ID;
                @qualLines = ();
        }
        else {
                $qualLine = $_;
                chomp $qualLine;
                push(@qualLines, $qualLine);
        }
}
$qualString = join "", @qualLines;
$fastq{$ID} = $qualString;
close FASTQFH;

my %clusters;
my $rn;
my $ratio;
my $cRatio;
my $rClus;
my $end;
```

```
my $c2count = 0;

#randomly assign a cluster number to each ID
foreach $ID (keys %fasta) {
	$cRatio = 0;
	$rClus = 0;
	$rn = rand(1);
	$end = 0;
	foreach $ratio (@ratios) {
		unless ($end == 1) {
			$cRatio = $cRatio+$ratio;
			$rClus++;
			if ($rn < $cRatio) {
				$clusters{$ID} = $rClus;
				$end = 1;
			}
		}
	}
}

#foreach $ID (keys %fasta) {
#	foreach my $QID (keys %fastq) {
#		if ($ID =~ $QID) {
#			my $newQID = $ID;
#			$fastq{$newQID} = $fastq{$QID};
#		}
#	}
#}

#write a pair of files for each cluster
my $clusterNo = 1;
while ($clusterNo <= $numClusters) {
	my $clusCount = 0;
	my @outFasta = ("randomCluster", $clusterNo, ".fasta");
	my @outQual  = ("randomCluster", $clusterNo, ".qual");
	my $outFasta = join "", @outFasta;
	my $outQual  = join "", @outQual;
	open (OUTFASTA, ">$outFasta");
	open (OUTQUAL, ">$outQual");
	foreach $ID (keys %clusters) {
```

```
                if ($clusters{$ID} == $clusterNo) {
                        print OUTFASTA ("$ID\n", $fasta{$ID}, "\n");
                        print OUTQUAL ("$ID\n", $fastq{$ID}, "\n");
                        $clusCount++;
                }
        }
        print "$clusCount\n";
        close OUTFASTA;
        close OUTQUAL;
        $clusterNo++;
}
```

Appendix B

- Appendix B-1: a table detailing the composition of the simulated dataset simLC (Mavromatis, Ivanova et al. 2007).
- Appendix B-2: a table detailing the taxonomy of each species contributing to simLC, up to the phylum level.

Appendix B-1

A table detailing the composition of the simulated dataset *simLC*.

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Rhodopseudomonas palustris HaA2	5331656	28861	27684394	5.19	637000240	316058
Bradyrhizobium sp. BTAi1	8422430	9277	9432593	1.11	640427103	288000
Cytophaga hutchinsonii ATCC 33406	4433218	5168	4132410	0.93	637000087	269798
Moorella thermoacetica ATCC 39073	2628784	674	667732	0.25	637000167	264732
Xylella fastidiosa Dixon	2622328	601	527819	0.2	638341237	155919
Ehrlichia canis Jake	1315030	196	192979	0.14	637000097	269484
Rubrobacter xylanophilus DSM 9941	3299423	409	472860	0.14	637000248	266117
Thiobacillus denitrificans ATCC 25259	2909809	395	432598	0.14	637000324	292415

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Bacillus cereus NVH391-98	3915845	520	530628	0.13	640753006	315749
Burkholderia sp. sp.strain 383	8676277	1074	1134915	0.13	637000051	269483
Caldicellulosiruptor accharolyticus UNDEF	2788317	367	374864	0.13	640427106	351627
Chloroflexus aurantiacus J-10-fl	5193782	679	676090	0.13	641228485	324602
Clostridium beijerincki NCIMB 8052	5952522	737	774306	0.13	640753016	290402
Crocosphaera watsonii WH 8501	6285399	812	853754	0.13	638341074	165597
Ehrlichia chaffeensis sapulpa	1005812	150	137612	0.13	638341079	332415
Prochlorococcus sp. NATL2A	1842899	253	243780	0.13	637000212	59920
Psychrobacter cryopegella UNDEF	3101097	422	406224	0.13	637000227	335284

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Rhodopseudomonas palustris BisB18	5513844	699	718618	0.13	637000237	316056
Shewanella sp. ANA-3	5100729	664	672467	0.13	639633058	94122
Shewanella sp. MR-7	4546355	568	592166	0.13	637000260	60481
Silicibacter sp. TM1040	4198271	469	547605	0.13	637000268	292414
Thermoanaerobacter ethanolicus 39E	2282740	315	298534	0.13	641522655	340099
Actinobacillus succinogenes 130Z	2046146	252	263514	0.12	640753001	339671
Burkholderia ambifaria AMMD	7503613	955	966386	0.12	637000047	339670
Chlorobium limicola DSMZ 245 (T)	2761915	381	354901	0.12	642555121	290315
Deinococcus geothermalis DSM11300	3164085	415	401474	0.12	641228488	319795

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Jannaschia sp. CCS1	4404049	543	529621	0.12	637000137	290400
Kineococcus radiotolerans SRS30216	4893957	566	588484	0.12	640753031	266940
Methylobacillus flagellatus strain KT	2971517	365	369843	0.12	637000165	265072
Nitrobacter winogradskyi Nb-255	3402093	427	414571	0.12	637000193	323098
Novosphingobium aromaticivorans DSM 12444 (F199)	3561584	520	446895	0.12	640427126	279238
Pelodictyon phaeoclathratiforme BU-1 (DSMZ 5477(T))	3000217	402	384821	0.12	642555146	324925
Polaromonas sp. JS666	5898676	733	737085	0.12	637000208	296591
Pseudoalteromonas atlantica T6c	5094958	588	615147	0.12	637000216	342610
Rhodopseudomonas palustris BisB5	4892717	575	593532	0.12	637000238	316057

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Sphingopyxis alaskensis RB2256	3343420	438	415701	0.12	637000271	317655
Thiomicrospira denitrificans ATCC 33889	2201561	277	268886	0.12	637000326	326298
Trichodesmium erythraeum IMS101	7750108	977	954106	0.12	637000329	203124
Alkalilimnicola ehrlichei MLHE-1	3272789	373	374946	0.11	637000005	187272
Anabaena variabilis ATCC 29413	7105752	855	795384	0.11	646564504	240292
Anaeromyxobacter dehalogenans 2CP-C	5013479	584	590941	0.11	637000007	290397
Arthrobacter sp. FB24	5011599	570	556264	0.11	639633006	290399
Azotobacter vinelandii AvOP	5352434	650	601559	0.11	643692004	322710
Burkholderia cenocepacia AU 1054	7249477	879	833042	0.11	637000046	331271

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Burkholderia cenocepacia HI2424	8139086	956	907846	0.11	639633014	331272
Burkholderia vietnamiensis G4	8410976	992	996433	0.11	640069307	269482
Clostridium thermocellum ATCC 27405	3894953	461	435152	0.11	640069309	203119
Desulfitobacterium hafniense DCB-2	6083768	769	684554	0.11	643348537	272564
Desulfovibrio desulfuricans G20	3730232	484	436153	0.11	637000095	207559
Exiguobacterium UNDEF 255-15	2894116	377	329023	0.11	641522626	262543
Frankia sp. CcI3	5433628	645	625569	0.11	637000116	106370
Frankia sp. EAN1pec	9081415	1109	1070299	0.11	641228492	298653
Geobacter metallireducens GS-15	4011182	515	469635	0.11	637000119	269799

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Lactobacillus casei ATCC 334	2760660	362	330900	0.11	639633028	321967
Lactobacillus gasseri ATCC 33323	1950210	244	231961	0.11	639633030	324831
Marinobacter aquaeolei VT8	4647952	547	542695	0.11	639633037	351348
Methanospirillum hungatei JF-1	3544738	429	412292	0.11	637000164	323259
Nitrobacter hamburgensis UNDEF	5011522	630	592565	0.11	637000192	323097
Nitrosococcus oceani UNDEF	3522111	409	398323	0.11	637000194	323261
Nitrosomonas eutropha C71	2711982	314	313955	0.11	637000196	335283
Nitrosospira multiformis ATCC 25196	3234309	378	366210	0.11	637000197	323848
Nocardioides sp. JS614	5394058	636	632236	0.11	639633046	196162

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Pelobacter carbinolicus DSM 2380	3662252	489	425000	0.11	637000204	338963
Pelobacter propionicus DSM 2379	4466736	508	505009	0.11	639633050	338966
Pseudomonas putida F1	5925059	675	687896	0.11	640427132	351746
Pseudomonas syringae B728a	6093698	746	673596	0.11	637000224	205918
Rhodoferax ferrireducens UNDEF	4969784	599	563098	0.11	637000235	338969
Rhodopseudomonas palustris BisA53	5502424	636	648905	0.11	639279312	316055
Rhodospirillum rubrum ATCC 11170	4406557	559	519746	0.11	637000241	269796
Shewanella amazonensis SB2B	4264533	536	497450	0.11	639633057	326297
Shewanella baltica OS155	5084318	621	589065	0.11	640069330	325240

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Shewanella frigidimarina NCMB400	4782427	551	539657	0.11	637000257	318167
Shewanella putefaciens UNDEF	4577697	565	537581	0.11	640427141	319224
Shewanella sp. PV-4	4474526	524	502771	0.11	640069331	323850
Streptococcus suis 89/1591	1978218	263	234214	0.11	638341209	286604
Syntrophobacter fumaroxidans MPOB	4848841	606	561886	0.11	639633063	335543
Alkaliphillus metalliredigenes UNDEF	4410303	489	452259	0.1	640753002	293826
Bifidobacterium longum DJO10A	2375286	288	238822	0.1	638341019	205913
Brevibacterium linens BL2	4510745	542	465883	0.1	638341022	321955
Chlorobium phaeobacteroides DSM 266	3114286	359	339279	0.1	639633020	290317

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Dechloromonas aromatica RCB	4501104	537	455726	0.1	637000088	159087
Ferroplasma acidarmanus fer1	1971391	238	201766	0.1	638341092	333146
Haemophilus somnus 129PT	2008359	232	208703	0.1	637000127	205914
Lactococcus lactis cremoris SK11	2613164	301	267507	0.1	639633031	272622
Leuconostoc mesenteroides mesenteroides ATCC 8293	1976579	235	199695	0.1	639633034	203120
Magnetococcus sp. MC-1	4628740	504	479512	0.1	639633036	156889
Paracoccus denitrificans PD1222	5175736	585	556907	0.1	639633048	318586
Pediococcus pentosaceus ATCC 25745	1814631	217	189252	0.1	639633049	278197
Pelodictyon luteolum UNDEF	2364842	250	238945	0.1	637000205	319225

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Prochlorococcus marinus str. MIT 9312	1709204	183	178782	0.1	637000210	74546
Prosthecochloris aestuarii SK413/DSMZ 271(t)	2563197	282	269231	0.1	642555149	290512
Prosthecochloris sp. BS1	4444192	483	450134	0.1	642555122	331678
Saccharophagus degradans 2-40	5057531	582	538073	0.1	637000249	203122
Shewanella sp. W3-18-1	4754010	533	520544	0.1	639633059	351745
Syntrophomonas wolfei Goettingen	2845772	314	311703	0.1	637000316	335541
Thiomicrospira crunogena XCL-2	2427734	274	260697	0.1	637000325	317025
Burkholderia xenovorans LB400	9731138	1149	934846	0.09	637000053	266265
Chlorobium vvibrioforme f. thiosulfatophilum DSMZ 265(T)	1980186	199	191789	0.09	640427130	290318

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Chromohalobacter salexigens DSM3043	4404049	454	423738	0.09	637000075	290398
Enterococcus faecium DO	2848380	359	260405	0.09	638341080	333849
Lactobacillus delbrueckii bulgaricus ATCC BAA-365	1629380	195	152277	0.09	639633029	321956
Mesorhizobium sp. BNC1	4977960	567	464795	0.09	637000160	266779
Methanococcoides burtonii DSM6242	2575032	268	248743	0.09	637000161	259564
Methanosarcina barkeri Fusaro	4873766	545	468320	0.09	637000162	269797
Pseudomonas fluorescens PfO-1	6438405	730	599462	0.09	637000221	205922
Psychrobacter arcticum 273-4	2650701	327	258585	0.09	637000226	259536
Synechococcus sp. PCC 7942 (elongatus)	2695903	316	266393	0.09	637000308	1140

Taxon	Genome size	Reads used	Total size of read sequence	Estimated coverage	IMG taxon id	NCBI taxonomy id
Thermobifida fusca YX	3642249	434	347485	0.09	637000319	269800
Oenococcus oeni PSU-1	1782786	182	157278	0.08	639633047	203123
Rhodobacter sphaeroides 2.4.1	4603060	514	394254	0.08	640069327	272943
Lactobacillus brevis ATCC 367	1880818	177	143668	0.07	639633027	387344
Streptococcus thermophilus LMD-9	1842121	178	146176	0.07	639633062	322159

Appendix B-2

A table detailing the taxonomy of each species contributing to *simLC*, up to the phylum level. Details of the three most well-represented species in the dataset are emphasised.

Species	Genus	Family	Order	Class	Phylum
Actinobacillus succinogenes 130Z	Actinobacillus	Pasteurellaceae	Pasteurellales	Gammaproteobacteria	Proteobacteria
Alkalilimnicola ehrlichei MLHE-1	Alkalilimnicola	Ectothio-rhodospiraceae	Chromatiales	Gammaproteobacteria	Proteobacteria
Alkaliphillus metalliredigenes UNDEF	Alkaliphillus	Clostridiaceae	Clostridiales	Clostridia	Firmicutes
Anabaena variabilis ATCC 29413	Anabaena	Nostocaceae	Nostocales	Nostocales	Cyanobacteria
Anaeromyxobacter dehalogenans 2CP-C	Anaeromyxobacter	Myxococcaceae	Myxococcales	Deltaproteobacteria	Proteobacteria
Arthrobacter sp. FB24	Arthrobacter	Micrococcaceae	Actinomycetales	Actinobacteria	Actinobacteria

Species	Genus	Family	Order	Class	Phylum
Azotobacter vinelandii AvOP	Azotobacter	Pseudomonadaceae	Pseudomonadales	Gammaproteobacteria	Proteobacteria
Bacillus cereus NVH391-98	Bacillus	Bacillaceae	Bacillales	Bacilli	Firmicutes
Bifidobacterium longum DJO10A	Bifidobacterium	Bifidobacteriaceae	Bifidobacteriales	Actinobacteria	Actinobacteria
Bradyrhizobium sp. BTAi1	**Bradyrhizobium**	**Bradyrhizobiaceae**	**Rhizobiales**	**Alphaproteobacteria**	**Proteobacteria**
Brevibacterium linens BL2	Brevibacterium	Brevibacteriaceae	Burkholderiales	Betaproteobacteria	Proteobacteria
Burkholderia ambifaria AMMD	Burkholderia	Burkholderiaceae	Burkholderiales	Betaproteobacteria	Proteobacteria
Burkholderia cenocepacia AU 1054	Burkholderia	Burkholderiaceae	Burkholderiales	Betaproteobacteria	Proteobacteria

Species	Genus	Family	Order	Class	Phylum
Burkholderia cenocepacia HI2424	Burkholderia	Burkholderiaceae	Burkholderiales	Betaproteobacteria	Proteobacteria
Burkholderia sp. sp.strain 383	Burkholderia	Burkholderiaceae	Burkholderiales	Betaproteobacteria	Proteobacteria
Burkholderia vietnamiensis G4	Burkholderia	Burkholderiaceae	Burkholderiales	Betaproteobacteria	Proteobacteria
Burkholderia xenovorans LB400	Burkholderia	Burkholderiaceae	Burkholderiales	Betaproteobacteria	Proteobacteria
Caldicellulosiruptor accharolyticus UNDEF	Caldicellulosiruptor	Caldicellulosiruptor	Thermo-anaerobacterales	Clostridia	Firmicutes
Chlorobium limicola DSMZ 245 (T)	Chlorobium	Chlorobiaceae	Chlorobiales	Chlorobia	Chlorobi
Chlorobium phaeobacteroides DSM 266	Chlorobium	Chlorobiaceae	Chlorobiales	Chlorobia	Chlorobi

Species	Genus	Family	Order	Class	Phylum
Chlorobium vvibrioforme f. thiosulfatophilum DSMZ 265(T)	Chlorobium	Chlorobiaceae	Chlorobiales	Chlorobia	Chlorobi
Chloroflexus aurantiacus J-10-fl	Chloroflexus	Chloroflexaceae	Chloroflexales	Chloroflexi	Chloroflexi
Chromohalobacter salexigens DSM3043	Chromohalobacter	Halomonadaceae	Oceanospirillales	Gammaproteobacteria	Proteobacteria
Clostridium beijerincki NCIMB 8052	Clostridium	Clostridiaceae	Clostridiales	Clostridia	Firmicutes
Clostridium thermocellum ATCC 27405	Clostridium	Clostridiaceae	Clostridiales	Clostridia	Firmicutes
Crocosphaera watsonii WH 8501	Crocosphaera	Crocosphaera	Chroococcales	Chroococcales	Cyanobacteria
Cytophaga hutchinsonii ATCC 33406	**Cytophaga**	**Cytophagaceae**	**Cytophagales**	**Cytophagia**	**Bacteroidetes**

Species	Genus	Family	Order	Class	Phylum
Dechloromonas aromatica RCB	Dechloromonas	Rhodocyclaceae	Rhodocyclales	Betaproteobacteria	Proteobacteria
Deinococcus geothermalis DSM11300	Deinococcus	Deinococcaceae	Deinococcales	Deinococci	Deinococcus_Thermus
Desulfitobacterium hafniense DCB-2	Desulfitobacterium	Peptococcaceae	Clostridiales	Clostridia	Firmicutes
Desulfovibrio desulfuricans G20	Desulfovibrio	Desulfovibrionaceae	Desulfovibrionales	Deltaproteobacteria	Proteobacteria
Ehrlichia canis Jake	Ehrlichia	Anaplasmataceae	Rickettsiales	Alphaproteobacteria	Proteobacteria
Ehrlichia chaffeensis sapulpa	Ehrlichia	Anaplasmataceae	Rickettsiales	Alphaproteobacteria	Proteobacteria
Enterococcus faecium DO	Enterococcus	Enterococcaceae	Lactobacillales	Bacilli	Firmicutes

Species	Genus	Family	Order	Class	Phylum
Exiguobacterium UNDEF 255-15	Exiguobacterium	Exiguobacterium	Bacillales	Bacilli	Firmicutes
Ferroplasma acidarmanus fer1	Ferroplasma	Ferroplasmaceae	Thermoplasmatales	Thermoplasmata	Thermoplasmata
Frankia sp. CcI3	Frankia	Frankiaceae	Actinomycetales	Actinobacteria	Actinobacteria
Frankia sp. EAN1pec	Frankia	Frankiaceae	Actinomycetales	Actinobacteria	Actinobacteria
Geobacter metallireducens GS-15	Geobacter	Geobacteraceae	Desulfuromonadales	Deltaproteobacteria	Proteobacteria
Haemophilus somnus 129PT	Haemophilus	Pasteurellaceae	Pasteurellales	Gammaproteobacteria	Proteobacteria
Jannaschia sp. CCS1	Jannaschia	Rhodobacteraceae	Rhodobacterales	Alphaproteobacteria	Proteobacteria

Species	Genus	Family	Order	Class	Phylum
Kineococcus radiotolerans SRS30216	Kineococcus	Kineosporiaceae	Actinomycetales	Actinobacteria	Actinobacteria
Lactobacillus brevis ATCC 367	Lactobacillus	Lactobacillaceae	Lactobacillales	Bacilli	Firmicutes
Lactobacillus casei ATCC 334	Lactobacillus	Lactobacillaceae	Lactobacillales	Bacilli	Firmicutes
Lactobacillus delbrueckii bulgaricus ATCC BAA-365	Lactobacillus	Lactobacillaceae	Lactobacillales	Bacilli	Firmicutes
Lactobacillus gasseri ATCC 33323	Lactobacillus	Lactobacillaceae	Lactobacillales	Bacilli	Firmicutes
Lactococcus lactis cremoris SK11	Lactococcus	Streptococcaceae	Lactobacillales	Bacilli	Firmicutes
Leuconostoc mesenteroides mesenteroides ATCC 8293	Leuconostoc	Leuconostocaceae	Lactobacillales	Bacilli	Firmicutes

Species	Genus	Family	Order	Class	Phylum
Magnetococcus sp. MC-1	Magnetococcus	Magnetococcus	Magnetococcus	Magnetococcus	Proteobacteria
Marinobacter aquaeolei VT8	Marinobacter	Alteromonadaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Mesorhizobium sp. BNC1	Mesorhizobium	Phyllobacteriaceae	Rhizobiales	Alphaproteobacteria	Proteobacteria
Methanococcoides burtonii DSM6242	Methanococcoides	Methanosarcinaceae	Methanosarcinales	Methanomicrobia	Methanomicrobia
Methanosarcina barkeri Fusaro	Methanosarcina	Methanosarcinaceae	Methanosarcinales	Methanomicrobia	Methanomicrobia
Methanospirillum hungatei JF-1	Methanospirillum	Methanospirillaceae	Methanomicrobiales	Methanomicrobia	Methanomicrobia
Methylobacillus flagellatus strain KT	Methylobacillus	Methylophilaceae	Methylophilales	Betaproteobacteria	Proteobacteria

Species	Genus	Family	Order	Class	Phylum
Moorella thermoacetica ATCC 39073	Moorella	Thermo-anaerobacteraceae	Thermo-anaerobacterales	Clostridia	Firmicutes
Nitrobacter hamburgensis UNDEF	Nitrobacter	Bradyrhizobiaceae	Rhizobiales	Alphaproteobacteria	Proteobacteria
Nitrobacter winogradskyi Nb-255	Nitrobacter	Bradyrhizobiaceae	Rhizobiales	Alphaproteobacteria	Proteobacteria
Nitrosococcus oceani UNDEF	Nitrosococcus	Chromatiaceae	Chromatiales	Gammaproteobacteria	Proteobacteria
Nitrosomonas eutropha C71	Nitrosomonas	Nitrosomonadaceae	Nitrosomonadales	Betaproteobacteria	Proteobacteria
Nitrosospira multiformis ATCC 25196	Nitrosospira	Nitrosomonadaceae	Nitrosomonadales	Betaproteobacteria	Proteobacteria
Nocardioides sp. JS614	Nocardioides	Nocardioidaceae	Actinomycetales	Actinobacteria	Actinobacteria

Species	Genus	Family	Order	Class	Phylum
Novosphingobium aromaticivorans DSM 12444 (F199)	Novosphingobium	Sphingomonadaceae	Sphingomonadales	Alphaproteobacteria	Proteobacteria
Oenococcus oeni PSU-1	Oenococcus	Leuconostocaceae	Lactobacillales	Bacilli	Firmicutes
Paracoccus denitrificans PD1222	Paracoccus	Rhodobacteraceae	Rhodobacterales	Alphaproteobacteria	Proteobacteria
Pediococcus pentosaceus ATCC 25745	Pediococcus	Lactobacillaceae	Lactobacillales	Bacilli	Firmicutes
Pelobacter carbinolicus DSM 2380	Pelobacter	Pelobacteraceae	Desulfuromonadales	Deltaproteobacteria	Proteobacteria
Pelobacter propionicus DSM 2379	Pelobacter	Pelobacteraceae	Desulfuromonadales	Deltaproteobacteria	Proteobacteria
Pelodictyon luteolum UNDEF	Pelodictyon	Chlorobiaceae	Chlorobiales	Chlorobia	Chlorobi

Species	Genus	Family	Order	Class	Phylum
Pelodictyon phaeoclathratiforme BU-1 (DSMZ 5477(T))	Pelodictyon	Chlorobiaceae	Chlorobiales	Chlorobia	Chlorobi
Polaromonas sp. JS666	Polaromonas	Comamonadaceae	Burkholderiales	Betaproteobacteria	Proteobacteria
Prochlorococcus marinus str. MIT 9312	Prochlorococcus	Prochlorococcaceae	Prochlorales	Prochlorales	Cyanobacteria
Prochlorococcus sp. NATL2A	Prochlorococcus	Prochlorococcaceae	Prochlorales	Prochlorales	Cyanobacteria
Prosthecochloris aestuarii SK413/DSMZ 271(t)	Prosthecochloris	Chlorobiaceae	Chlorobiales	Chlorobia	Chlorobi
Prosthecochloris sp. BS1	Prosthecochloris	Chlorobiaceae	Chlorobiales	Chlorobia	Chlorobi
Pseudoalteromonas atlantica T6c	Pseudoalteromonas	Pseudo-alteromonadaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria

Species	Genus	Family	Order	Class	Phylum
Pseudomonas fluorescens PfO-1	Pseudomonas	Pseudomonadaceae	Pseudomonadales	Gammaproteobacteria	Proteobacteria
Pseudomonas putida F1	Pseudomonas	Pseudomonadaceae	Pseudomonadales	Gammaproteobacteria	Proteobacteria
Pseudomonas syringae B728a	Pseudomonas	Pseudomonadaceae	Pseudomonadales	Gammaproteobacteria	Proteobacteria
Psychrobacter arcticum 273-4	Psychrobacter	Moraxellaceae	Pseudomonadales	Gammaproteobacteria	Proteobacteria
Psychrobacter cryopegella UNDEF	Psychrobacter	Moraxellaceae	Pseudomonadales	Gammaproteobacteria	Proteobacteria
Rhodobacter sphaeroides 2.4.1	Rhodobacter	Rhodobacteraceae	Rhodobacterales	Alphaproteobacteria	Proteobacteria
Rhodoferax ferrireducens UNDEF	Rhodoferax	Comamonadaceae	Burkholderiales	Betaproteobacteria	Proteobacteria

Species	Genus	Family	Order	Class	Phylum
Rhodopseudomonas palustris BisA53	Rhodopseudomonas	Bradyrhizobiaceae	Rhizobiales	Alphaproteobacteria	Proteobacteria
Rhodopseudomonas palustris BisB18	Rhodopseudomonas	Bradyrhizobiaceae	Rhizobiales	Alphaproteobacteria	Proteobacteria
Rhodopseudomonas palustris BisB5	Rhodopseudomonas	Bradyrhizobiaceae	Rhizobiales	Alphaproteobacteria	Proteobacteria
Rhodopseudomonas palustris HaA2	**Rhodopseudomonas**	**Bradyrhizobiaceae**	**Rhizobiales**	**Alphaproteobacteria**	**Proteobacteria**
Rhodospirillum rubrum ATCC 11170	Rhodospirillum	Rhodospirillaceae	Rhodospirillales	Alphaproteobacteria	Proteobacteria
Rubrobacter xylanophilus DSM 9941	Rubrobacter	Rubrobacteraceae	Rubrobacterales	Actinobacteria	Actinobacteria
Saccharophagus degradans 2-40	Saccharophagus	Alteromonadaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria

Species	Genus	Family	Order	Class	Phylum
Shewanella amazonensis SB2B	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Shewanella baltica OS155	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Shewanella frigidimarina NCMB400	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Shewanella putefaciens UNDEF	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Shewanella sp. ANA-3	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Shewanella sp. MR-7	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Shewanella sp. PV-4	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria

Species	Genus	Family	Order	Class	Phylum
Shewanella sp. W3-18-1	Shewanella	Shewanellaceae	Alteromonadales	Gammaproteobacteria	Proteobacteria
Silicibacter sp. TM1040	Silicibacter	Rhodobacteraceae	Rhodobacterales	Alphaproteobacteria	Proteobacteria
Sphingopyxis alaskensis RB2256	Sphingopyxis	Sphingomonadaceae	Sphingomonadales	Alphaproteobacteria	Proteobacteria
Streptococcus suis 89/1591	Streptococcus	Streptococcaceae	Lactobacillales	Bacilli	Firmicutes
Streptococcus thermophilus LMD-9	Streptococcus	Streptococcaceae	Lactobacillales	Bacilli	Firmicutes
Synechococcus sp. PCC 7942 (elongatus)	Synechococcus	Chroococcus	Chroococcales	Chroococcales	Cyanobacteria
Syntrophobacter fumaroxidans MPOB	Syntrophobacter	Syntrophomonadaceae	Clostridiales	Clostridia	Firmicutes

Species	Genus	Family	Order	Class	Phylum
Syntrophomonas wolfei Goettingen	Syntrophomonas	Syntrophomonadaceae	Clostridiales	Clostridia	Firmicutes
Thermoanaerobacter ethanolicus 39E	Thermoanaerobacter	Thermo-anaerobacteraceae	Thermo-anaerobacterales	Clostridia	Firmicutes
Thermobifida fusca YX	Thermobifida	Nocardiopsaceae	Actinomycetales	Actinobacteria	Actinobacteria
Thiobacillus denitrificans ATCC 25259	Thiobacillus	Hydrogenophilaceae	Hydrogenophilales	Betaproteobacteria	Proteobacteria
Thiomicrospira crunogena XCL-2	Thiomicrospira	Piscirickettsiaceae	Thiotrichales	Gammaproteobacteria	Proteobacteria
Thiomicrospira denitrificans ATCC 33889	Thiomicrospira	Piscirickettsiaceae	Thiotrichales	Gammaproteobacteria	Proteobacteria
Trichodesmium erythraeum IMS101	Trichodesmium	Trichodesmium	Oscillatoriales	Oscillatoriales	Cyanobacteria

Species	Genus	Family	Order	Class	Phylum
Xylella fastidiosa Dixon	Xylella	Xanthomonadaceae	Xanthomonadales	Gammaproteobacteria	Proteobacteria

Table of Abbreviations

Abbreviation	Term	Definition
APA	Assembled + Partially-Assembled	The combined number of sequencing reads assembled and partially-assembled into contigs by *Newbler*.
BR1	*Bradyrhizobium* sp. BTAi1	A particular strain of bacterium symbiotic to the roots of plants and important in the process of nitrogen-fixation (van Rhijn and Vanderleyden, 1995). One of the three most well-represented species in the dataset simLC.
CLARA	Clustering LARge Applications	A modified implementation of the PAM method of clustering designed for use with large datasets (Kaufman and Rousseeuw 1990).
CPH	*Cytophaga hutchinsonii* ATCC 33406	A particular strain of a species of gram-negative bacterium, common in soils and able to rapidly digest cellulose (Zhu et al. 2010). One of the three most well-represented species in the dataset simLC.
DBSCAN	Density Based Spatial Clustering of Applications with Noise	A density-based clustering algorithm (Ester, Kriegel et al. 1996).
ddNTP	di-deoxynucleotide	a modified deoxynucleotide base with an additional deoxygenated group that prevents further elongation after incorporation into a strand.
DENCLUE	Fast Clustering Based on Kernel Density Estimation	A density-based approach less sensitive to high-dimensionality than DBSCAN (Hinneburg and Gabriel 2007).
EST	Expressed Sequence Tag	A short sequencing read from either end of an RNA transcript.

Abbreviation	Term	Definition
FCM	Fuzzy *c*-Means clustering	A soft partitioning clustering method, similar to *k*-means clustering, that allows for datapoints to belong to more than one group, with a weighting associated with each point for each group, denoting the degree of its membership to the group (Bezdek 1981).
GC	GC content	A sequence feature vector describing the proportion of a sequence that is made up of G and C nucleotides.
GC+IND etc.	GC content and inter-nucleotide distance vector	A feature vector containing the values of the features specified by their abbreviations.
HHSOM	Hyperbolic Hierarchically-growing Self-Organising Map	A specialised variant of the SOM, projected in hyperbolic space and arranged in rings of nodes growing from the centre of the map (Martin et al. 2008).
IND	Inter-Nucleotide Distance	A sequence feature vector consisting of a single value, the Kolmogorov-Smirnov distance between observed and expected frequencies of distances between nucleotides of the same type (Afreixo et al. 2009).
KASP	*k*-means-based Approximate SPectral clustering	An implementation of spectral clustering designed for use with large datasets, approximating the optimal solution using a set of representative points to partition the dataset as a whole (Yan and Jordan 2009).
KM	*k*-Means clustering	A partitioning clustering method, that divides data into a set number of groups by minimising the distance between each datapoint in a group and the mean vector of these datapoints.
OFDEG	Oligonucleotide Frequency-Derived Error Gradient	A sequence feature vector consisting of a single value, the gradient of the degradation of error values calculated between oligonucleotide relative frequency distributions of the sequence and sub-sequences of increasing length (Saeed and Halgamuge 2009).

Abbreviation	Term	Definition
PAM	Partitioning Around Mediods	A partitioning clustering method, similar to *k*-means/*k*-medians clustering, that groups data by minimising the total distance between the points in a group and a central, representative point in that group (the mediod) (Kaufman and Rousseeuw 1990).
Pr	Precision	A measure of the quality of clustering, calculated as the proportion of a single cluster that is accounted for by the predominant class of data within that cluster (Kelley and Salzberg 2010).
Rc	Recall	A measure of the quality of clustering, calculated as the proportion of all the data belonging to a class within the dataset that is accounted for by the predominant class of data within that cluster (Kelley and Salzberg 2010).
RPH	*Rhodopseudomonas palustris* HaA2	A particular strain of a species of gram-negative bacterium with a highly-adaptable metabolism (Larimer et al. 2004; Bell et al. 2009). One of the three most well-represented species in the dataset simLC.
SOM	Self-Organising Map	A grid of nodes that 'learns' a dataset as it is applied, with the distance between nodes increasing or decreasing according to the level of similarity between the data applied to them (Kohonen 1982).
TNF	TetraNucleotide relative Frequency distribution vector	A sequence feature vector describing the relative frequency distribution of 4-letter nucleotide 'words' in a sequence.

Bibliography

Abe, T., S. Kanaya, et al. (2002). "A novel bioinformatic strategy for unveiling hidden genome signatures of eukaryotes: self-organizing map of oligonucleotide frequency." *Genome Informatics. International Conference on Genome Informatics* **13**: 12-20.

Abe, T., S. Kanaya, et al. (2003). "Informatics for unveiling hidden genome signatures." *Genome Research* **13**(4): 693-702.

Abe, T., H. Sugawara, et al. (2006). "Novel Phylogenetic Studies of Genomic Sequence Fragments Derived from Uncultured Microbe Mixtures in Environmental and Clinical Samples." *Environment* **290**: 281-290.

Afreixo, V., C. A. C. Bastos, et al. (2009). "Genome analysis with inter-nucleotide distances." *Bioinformatics* **25**(23): 3064-3070.

Almeida, J. S., J. A. Carriço, et al. (2001). "Analysis of genomic sequences by Chaos Game Representation." *Bioinformatics* **17**(5): 429-437.

Altschul, S. F., W. Gish, et al. (1990). "Basic local alignment search tool." *Journal of Molecular Biology* **215**(3): 403-410.

Anastassiou, D. (2000). "Frequency-domain analysis of biomolecular sequences." *Bioinformatics* **16**(12): 1073-1081.

Baker, B. J., G. W. Tyson, et al. (2006). "Lineages of Acidophilic Archaea Revealed by Community Genomic Analysis." *Science* **314**(5807): 1933-1935.

Barrai, I., Scapoli, C, Barale, R and Volinia, S (1990). "Oligonucleotide correlations between infector and host genomes hint at evolutionary relationships." *Nucleic Acids Research* **18**: 3021-3025.

Basak, S. and T. C. Ghosh (2005). "On the origin of genomic adaptation at high temperature for prokaryotic organisms." *Biochemical and Biophysical Research Communications* **330**: 629-632.

Bastos, C. a. C., V. Afreixo, et al. (2011). "Inter-dinucleotide distances in the human genome: an analysis of the whole-genome and protein-coding distributions." *Journal of Integrative Bioinformatics* **8**: 172.

Bauer, M., S. M. Schuster, et al. (2008). "The average mutual information profile as a genomic signature." *BMC Bioinformatics* **9**(48).

Bell, S. G., A. B. H. Tan, et al. (2010). "Selective oxidative demethylation of veratric acid to vanillic acid by CYP199A4 from Rhodopseudomonas palustris HaA2." *Molecular Biosystems* **6**(1): 206-214.

Berkhin, P. (2006). “A Survey of Clustering Data Mining Techniques.” *Grouping Multidimensional Data* 25-71.

Bernardi, G. (2007). "The neoselectionist theory of genome evolution." *Proceedings of the National Academy of Sciences of the United States of America* **104**: 8385-8390.

Bernardi, G. and G. Bernardi (1986). "Compositional Constraints and Genome Evolution." *Journal Molecular Evolution* **24**(1-2): 1-11.

Bernardi, G., B. Olofsson, et al. (1985). "The Mosaic Genome of Warm-Blooded Vertebrates." *Science* **228**(4702): 953-958.

Bezdek, J. C. (1981). "Pattern Recognition with Fuzzy Objective Function Algorithms." Plenum, New York.

Biers, E. J., S. L. Sun, et al. (2009). "Prokaryotic Genomes and Diversity in Surface Ocean Waters: Interrogating the Global Ocean Sampling Metagenome." *Applied and Environmental Microbiology* **75**(7): 2221-2229.

Bohlin, J., E. Skjerve, et al. (2008). "Investigations of oligonucleotide usage variance within and between prokaryotes." *PLoS Computational Biology* **4**(4): e1000057.

Bohlin, J., M. van Passel, et al. (2012). "Relative entropy differences in bacterial chromosomes, plasmids, phages and genomic islands." *BMC Genomics* **13**(66): 1-12.

Boonham, N., L. Laurenson, et al. (2009). "Direct Detection of Plant Viruses in Potato Tubers Using Real-Time PCR." *Methods in Molecular Biology* **508**: 249-258.

Branney, P. A., L. Faas, et al. (2009). "Characterisation of the fibroblast growth factor dependent transcriptome in early development." *PLoS ONE* **4**(3): e4951.

Browne, R. P., P. D. McNicholas, et al. (2012). "Model-Based Learning Using a Mixture of Mixtures of Gaussian and Uniform Distributions." *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(4): 814-817.

Buell, C. R., V. Joardar, et al. (2003). "The complete genome sequence of the Arabidopsis and tomato pathogen Pseudomonas syringae pv. tomato DC3000." *Proceedings of the National Academy of Sciences of the United States of America* **100**(18): 10181-10186.

Cantarel, B. L., V. Lombard, et al. (2012). "Complex Carbohydrate Utilization by the Healthy Human Microbiome." *PLoS ONE* **7**(6): e28742.

Chaisson, M. J. and P. A. Pevzner (2008). "Short read fragment assembly of bacterial genomes." *Genome Research* **18**(2): 324-330.

Chan, C.-K. K., A. L. Hsu, et al. (2008). "Methodology Report Using Growing Self-Organising Maps to Improve the Binning Process in Environmental Whole-Genome Shotgun Sequencing." *Journal of Biomedicine & Biotechnology* **2008**

Chatterji, S., I. Yamazaki, et al. (2008). CompostBin: A DNA composition-based algorithm for binning environmental shotgun reads. *Research in*

Computational Molecular Biology, Proceedings. M. Vingron and L. Wong. Berlin, Springer-Verlag Berlin. **4955:** 17-28.

Cheeseman, P. and J. Stutz (1996). "Bayesian classification (AutoClass): theory and results." *Advances in Knowledge Discovery and Data Mining*: 153-180.

Chen, C. L., F. S. C. Tseng, et al. (2010). "An integration of Word Net and fuzzy association rule mining for multi-label document clustering." *Data & Knowledge Engineering* **69**(11): 1208-1226.

Chen, K. and L. Pachter (2005). "Bioinformatics for whole-genome shotgun sequencing of microbial communities." *PLoS Computational Biology* **1**: 106-112.

Chen, L. L. and F. Gao (2005). "Detection of nucleolar organizer and mitochondrial DNA insertion regions based on the isochore map of Arabidopsis thaliana." *Febs Journal* **272**(13): 3328-3336.

Clarke, J., H.-C. Wu, et al. (2009). "Continuous base identification for single-molecule nanopore DNA sequencing." *Nat. Nano.* **4**(4): 265-270.

Cornman, R., M. Schatz, et al. (2010). "Genomic survey of the ectoparasitic mite Varroa destructor, a major pest of the honey bee Apis mellifera." *BMC Genomics* **11**(1): 602.

Costello, E. K., C. L. Lauber, et al. (2009). "Bacterial community variation in human body habitats across space and time." *Science* **326**: 1694-1697.

Cronn, R., A. Liston, et al. (2008). "Multiplex sequencing of plant chloroplast genomes using Solexa sequencing-by-synthesis technology." *Nucleic Acids Research* **36**(19): e122.

Del Frate, F., F. Pacifici, et al. (2007). "Use of Neural Networks for Automatic Classification From High-Resolution Images." *IEEE Transactions on Geoscience and Remote Sensing* **45**(4): 800-809.

Dempster, A. P., N. M. Laird, et al. (1977). "Maximum Likelihood from Incomplete Data via the EM Algorithm." *Journal of the Royal Statistical Society. Series B (Methodological)* **39**(1): 1-38.

Desai, D. K., F. D. Desai, et al. (2012). "Factors influencing the diversity of iron uptake systems in aquatic microorganisms." *Frontiers in Microbiology* **3**: 362.

Deschavanne, P. J. and A. Giron (1999). "Genomic Signature: Characterization and Classification of Species Assessed by Chaos Game Representation of Sequences." *Mol. Biol. Evol.* **16**: 1391-1399.

Dick, G. J., A. F. Andersson, et al. (2009). "Community-wide analysis of microbial genome sequence signatures " *Genome Biology* **10**: 1-16.

Eid, J., A. Fehr, et al. (2009). "Real-Time DNA Sequencing from Single Polymerase Molecules." *Science* **323**(5910): 133-138.

Eisen, J. A. (2007). "Environmental shotgun sequencing: its potential and challenges for studying the hidden world of microbes." *PLoS Biology* **5**: e82.

Elhaik, E., D. Graur, et al. (2010). "Comparative Testing of DNA Segmentation Algorithms Using Benchmark Simulations." *Molecular Biology and Evolution* **27**(5): 1015-1024.

Ester, M., H.-p. Kriegel, et al. (1996). "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise." *Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*: 226-231.

Faust, K., J. F. Sathirapongsasuti, et al. (2012). "Microbial Co-occurrence Relationships in the Human Microbiome." *PLoS Computational Biology* **8** (7): e1002606.

Fedorova, L. and A. Fedorov (2011). "Mid-range inhomogeneity of eukaryotic genomes." *TheScientificWorldJournal* **11**: 842-854.

Forgy, E. W. (1965). "CLUSTER ANALYSIS OF MULTIVARIATE DATA - EFFICIENCY VS INTERPRETABILITY OF CLASSIFICATIONS." *Biometrics* **21**(3): 768-769.

Fraley, C. and A. E. Raftery (1999). "MCLUST: Software for Model-Based Cluster Analysis." *Journal of Classification* **16**(2): 297-306.

Fraley, C. and A. E. Raftery (2002). "Model-based clustering, discriminant analysis, and density estimation." *Journal of the American Statistical Association* **97**(458): 611-631.

Galtier, N. and J. Lobry (1997). "Relationships between genomic G+C content, RNA secondary structures, and optimal growth temperature in prokaryotes." *Journal of Molecular Evolution* **44**(6): 632-636.

Gentles, A. J. and S. Karlin (2001). "Genome-scale compositional comparisons in eukaryotes." *Genome Research* **11**: 540-546.

Gill, S. R., M. Pop, et al. (2006). "Metagenomic analysis of the human distal gut microbiome." *Science* **312**(5778): 1355-1359.

Gilles, A., E. Meglecz, et al. (2011). "Accuracy and quality assessment of 454 GS-FLX Titanium pyrosequencing." *BMC Genomics* **12**: 245.

Goldman, N. (1993). "Nucleotide, dinucleotide and trinucleotide frequencies explain patterns observed in chaos game representations of DNA sequences." *Nucleic Acids Research* **21**: 2487-2491.

Grice, E. A., H. H. Kong, et al. (2009). "Topographical and temporal diversity of the human skin microbiome." *Science* **324**: 1190-1192.

Guha, S., R. Rastogi, et al. (2001). "Cure: an efficient clustering algorithm for large databases." *Information Systems* **26**(1): 35-58.

Halkidi, M., Y. Batistakis, et al. (2001). "On clustering validation techniques." *Journal of Intelligent Information Systems* **17**(2-3): 107-145.

Handelsman, J., M. R. Rondon, et al. (1998). "Molecular biological access to the chemistry of unknown soil microbes: a new frontier for natural products." *Chemistry & Biology* **5**(10): R245-249.

Harrison, A., D. W. Dyer, et al. (2005). "Genomic sequence of an otitis media isolate of nontypeable Haemophilus influenzae: Comparative study with H-influenzae serotype d, strain KW20." *Journal of Bacteriology* **187**(13): 4627-4636.

Hartigan, J. A. and M. A. Wong (1979). "Algorithm AS 136: A K-Means Clustering Algorithm." *Journal of the Royal Statistical Society. Series C (Applied Statistics)* **28**(1): 100-108.

Hinneburg, A. and H.-H. Gabriel (2007). "DENCLUE 2.0: Fast Clustering Based on Kernel Density Estimation." *Advances in Intelligent Data Analysis VII* **4723**: 70-80.

Holland, P. M., R. D. Abramson, et al. (1991). "Detection of specific polymerase chain reaction product by utilizing the 5'----3' exonuclease activity of Thermus aquaticus DNA polymerase." *Proceedings of the National Academy of Sciences of the United States of America* **88**: 7276-7280.

Hooper, S. D., D. Dalevi, et al. (2010). "Estimating DNA coverage and abundance in metagenomes using a gamma approximation." *Bioinformatics* **26**(3): 295-301.

Huse, S. M., Y. Z. Ye, et al. (2012). "A Core Human Microbiome as Viewed through 16S rRNA Sequence Clusters." *PLoS ONE* **7**(6): e34242.

Huson, D. H., A. F. Auch, et al. (2007). "MEGAN analysis of metagenomic data." *Genome Research* **17**: 377-386.

Huson, D. H., D. C. Richter, et al. (2009). "Methods for comparative metagenomics." *BMC Bioinformatics* **10**(Supp1): S12.

IHGC, T. I. H. G. S. C. (2001). "Initial sequencing and analysis of the human genome." *Nature* **409**(6822): 860-921.

Jeffrey, H. J. (1990). "Chaos game representation of gene structure." *Nucleic Acids Reseach* **18**: 2163-2170.

Joseph, J. and R. Sasikumar (2006). "Chaos game representation for comparison of whole genomes." *BMC Bioinformatics* **7**: 243.

Kailing, K., H.-P. Kriegel, et al. (2004). "Density-Connected Subspace Clustering for High-Dimensional Data." *Proceedings of 4th SIAM International Conference of Data Mining*: 246-257.

Kanaya, S., M. Kinouchi, et al. (2001). "Analysis of codon usage diversity of bacterial genes with a self-organizing map (SOM): characterization of horizontally transferred genes with emphasis on the E. coli O157 genome." *Gene* **276**: 89-99.

Kannan, R., S. Vempala, et al. (2004). "On clusterings: Good, bad and spectral." *J. ACM* **51**(3): 497-515.

Karlin, S. and C. Burge (1995). "Dinucleotide relative abundance extremes: a genomic signature." *Trends in Genetics : TIG* **11**(7): 283-290.

Karlin, S. and I. Ladunga (1994). "Comparisons of eukaryotic genomic sequences." *Proceedings of the National Academy of Sciences of the United States of America* **91**: 12832-12836.

Karlin, S., J. Mrázek, et al. (1997). "Compositional biases of bacterial genomes and evolutionary implications." *Journal of Bacteriology* **179**: 3899-3913.

Karypis, G., H. Eui-Hong, et al. (1999). "Chameleon: hierarchical clustering using dynamic modeling." *Computer* **32**(8): 68-75.

Kaufman, L. and P. J. Rousseeuw (1990). "Finding Groups in Data." *Wiley Series on Probability and Mathematical Statistics*, Wiley Inter-Science.

Kelley, D. R. and S. L. Salzberg (2010). "Clustering metagenomic sequences with interpolated Markov models." *BMC Bioinformatics* **11**: 544.

Kim, M. G., S. Y. Kim, et al. (2008). "Responses of Arabidopsis thaliana to challenge by Pseudomonas syringae." *Molecules and Cells* **25**(3): 323-331.

Kohonen, T. (1982). "SELF-ORGANIZED FORMATION OF TOPOLOGICALLY CORRECT FEATURE MAPS." *Biological Cybernetics* **43**(1): 59-69.

Kohonen, T. (1990). "The Self-organizing Map." Proc. IEEE 78: 1464-1480.

Kosakovsky Pond, S., S. Wadhawan, et al. (2009). "Windshield splatter analysis with the Galaxy metagenomic pipeline." *Genome Research* **19**: 2144-2153.

Kumar, S. and M. L. Blaxter (2011). "Simultaneous genome sequencing of symbionts and their hosts." *Symbiosis* **55**(3): 119-126.

Lamprea-Burgunder, E., P. Ludin, et al. (2011). "Species-specific Typing of DNA Based on Palindrome Frequency Patterns." *DNA Research* **18**(2): 117-124.

Lander, E. S. and M. S. Waterman (1988). "GENOMIC MAPPING BY FINGERPRINTING RANDOM CLONES A MATHEMATICAL ANALYSIS." *Genomics* **2**(3): 231-239.

Lane, D. J., B. Pace, et al. (1985). "RAPID-DETERMINATION OF 16S RIBOSOMAL-RNA SEQUENCES FOR PHYLOGENETIC ANALYSES." *Proceedings of the National Academy of Sciences of the United States of America* **82**(20): 6955-6959.

Larimer, F. W., P. Chain, et al. (2004). "Complete genome sequence of the metabolically versatile photosynthetic bacterium Rhodopseudomonas palustris." *Nat. Biotech.* **22**(1): 55-61.

Le, S. Q. and R. Durbin (2011). "SNP detection and genotyping from low-coverage sequencing data on multiple diploid samples." *Genome Research* **21**(6): 952-960.

Li, K., M. Bihan, et al. (2012). "Analyses of the Microbial Diversity across the Human Microbiome." *PLoS ONE* **7**(6): e32118.

Lieberman, K. R., G. M. Cherf, et al. (2010). "Processive Replication of Single DNA Molecules in a Nanopore Catalyzed by phi29 DNA Polymerase." *Journal of the American Chemical Society* **132**(50): 17961-17972.

Lima, N., T. Rogers, et al. (2012). "Temporal stability and species specificity in bacteria associated with the bottlenose dolphins respiratory system." *Environmental Microbiology Reports* **4**(1): 89-96.

Liu, B., L. L. Faller, et al. (2012). "Deep Sequencing of the Oral Microbiome Reveals Signatures of Periodontal Disease." *PLoS ONE* **7**(6): e37919.

Lloyd, S. (1982). "Least squares quantization in PCM." *IEEE Transactions on Information Theory* **28**(2): 129-137.

Lorenzi, H. A., J. Hoover, et al. (2011). "TheViral MetaGenome Annotation Pipeline (VMGAP): An automated tool for the functional annotation of viral Metagenomic shotgun sequencing data." *Standards in Genomic Sciences* **4**(3): 418-429.

Lower, R., J. Lower, et al. (1996). "The viruses in all of us: Characteristics and biological significance of human endogenous retrovirus sequences." *Proceedings of the National Academy of Sciences of the United States of America* **93**(11): 5177-5184.

Macqueen, J. B. (1967). "SOME METHODS FOR CLASSIFICATION AND ANALYSIS OF MULTIVARIATE OBSERVATIONS." *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*: 281-297.

Mardis, E. R. (2011). "A decade's perspective on DNA sequencing technology." *Nature* **470**(7333): 198-203.

Margulies, M., M. Egholm, et al. (2005). "Genome sequencing in microfabricated high-density picolitre reactors." *Nature* **437**: 376-381.

Markowitz, V. M., I. M. A. Chen, et al. (2012). "IMG/M: the integrated metagenome data management and comparative analysis system." *Nucleic Acids Research* **40**(D1): D123-D129.

Martin, C., N. N. Diaz, et al. (2008). "Hyperbolic SOM-based clustering of DNA fragment features for taxonomic visualization and classification." *Bioinformatics* **24**: 1568-1574.

Martin, H. G., N. Ivanova, et al. (2006). "Metagenomic analysis of two enhanced biological phosphorus removal (EBPR) sludge communities." *Nature Biotechnology* **24**(10): 1263-1269.

Mavromatis, K., N. Ivanova, et al. (2007). "Use of simulated data sets to evaluate the fidelity of metagenomic processing methods." *Nature Methods* **4**: 495-500.

McGuire, A. L., L. S. Achenbaum, et al. (2012). "PERSPECTIVES ON HUMAN MICROBIOME RESEARCH ETHICS." *Journal of Empirical Research on Human Research Ethics* **7**(3): 1-14.

McHardy, A. C. and I. Rigoutsos (2007). "What's in the mix: phylogenetic classification of metagenome sequence samples." *Current Opinion in Microbiology* **10**(5): 499-503.

McKenna, A. H., M. Hanna, et al. (2010). "The Genome Analysis Toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data." *Genome Research* **20**: 1297-1303.

Meyer, F., D. Paarmann, et al. (2008). "The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes." *BMC Bioinformatics* **9**:386.

Miller, J. R., S. Koren, et al. (2010). "Assembly algorithms for next-generation sequencing data." *Genomics* **95**(6): 315-327.

Morgan, J. L., A. E. Darling, et al. (2010). "Metagenomic Sequencing of an In Vitro-Simulated Microbial Community." *PLoS ONE* **5**: e10209.

Murtagh, F. and P. Contreras (2012). "Algorithms for hierarchical clustering: an overview." *Wiley Interdisciplinary Reviews-Data Mining and Knowledge Discovery* **2**(1): 86-97.

Musto, H., H. Naya, et al. (2006). "Genomic GC level, optimal growth temperature, and genome size in prokaryotes." *Biochemical and Biophysical Research Communications* **347**(1): 1-3.

Nagaraj, S. H., R. B. Gasser, et al. (2007). "A hitchhiker's guide to expressed sequence tag (EST) analysis." *Briefings in Bioinformatics* **8**(1): 6-21.

Nair, A. S. S. and T. Mahalakshmi (2005). "Visualization Of Genomic Data Using Inter-Nucleotide Distance Signals." *Proceedings of IEEE Genomic Signal Processing* **408**.

Nakamura, K., T. Oshima, et al. (2011). "Sequence-specific error profile of Illumina sequencers." *Nucleic Acids Research* **39**(13).

Nasser, S., A. Breland, et al. (2008). "A Fuzzy Classifier to Taxonomically Group DNA Fragments within a Metagenome." *Proceedings of the Annual Meeting of the North American Fuzzy Information Processing Society, 2008*.

Naya, H., H. Romero, et al. (2001). "Translational selection shapes codon usage in the GC-rich genome of Chlamydomonas reinhardtii." *FEBS Letters* **501**(2-3): 127-130.

Ng, A. Y., M. I. Jordan, et al. (2001). "On Spectral Clustering: Analysis and an algorithm." *Advances in Neural Information Processing Systems* **14**: 849-856.

Nielsen, R., J. S. Paul, et al. (2011). "Genotype and SNP calling from next-generation sequencing data." *Nat. Rev. Genet.* **12**(6): 443-451.

Ning, Z., A. J. Cox, et al. (2001). "SSAHA: a fast search method for large DNA databases." *Genome Research* **11**: 1725-1729.

Oliver, J. L., P. Bernaola-Galván, et al. (2001). "Isochore chromosome maps of eukaryotic genomes." *Gene* **276**: 47-56.

Oliver, J. L., P. Carpena, et al. (2002). "Isochore chromosome maps of the human genome." *Gene* **300**: 117-127.

Oliver, J. L., P. Bernaola-Galvan, et al. (2008). "Phylogenetic distribution of large-scale genome patchiness." *BMC Evolutionary Biology* **8**:107.

Owen, J., M. Shintaku, et al. (1990). "NUCLEOTIDE-SEQUENCE AND EVOLUTIONARY RELATIONSHIPS OF CUCUMBER MOSAIC-VIRUS (CMV) STRAINS - CMV RNA-3." *Journal of General Virology* **71**: 2243-2249.

Pace, N. R. (1997). "A Molecular View of Microbial Diversity and the Biosphere." *Science* **276**(5313): 734-740.

Paszkiewicz, K. and D. J. Studholme (2010). "De novo assembly of short sequence reads." *Briefings in Bioinformatics* **11**(5): 457-472.

Paz, A., V. Kirzhner, et al. (2006). "Coevolution of DNA-interacting proteins and genome "dialect"." *Molecular Biology and Evolution* **23**: 56-64.

Peterson, J., S. Garges, et al. (2009). "The NIH Human Microbiome Project." *Genome Research* **19**(12): 2317-2323.

Pride, D. T., R. J. Meinersmann, et al. (2003). "Evolutionary implications of microbial genome tetranucleotide frequency biases." *Genome Research* **13**: 145-158.

Pushalkar, S., S. P. Mane, et al. (2011). "Microbial diversity in saliva of oral squamous cell carcinoma." *Fems Immunology and Medical Microbiology* **61**(3): 269-277.

Qin, J., R. Li, et al. (2010). "A human gut microbial gene catalogue established by metagenomic sequencing." *Nature* **464**: 59-65.

Rappe, M. S. and S. J. Giovannoni (2003). "The uncultured microbial majority." *Annual Review of Microbiology* **57**: 369-394.

Richter DC, Ott F, Auch AF, Schmid R, Huson DH (2008). "MetaSim—A Sequencing Simulator for Genomics and Metagenomics." *PLoS ONE* **3** (10): e3373. doi:10.1371/journal.pone.0003373

Rico, A., S. L. McCraw, et al. (2011). "The metabolic interface between Pseudomonas syringae and plant cells." *Current Opinion in Microbiology* **14**(1): 31-38.

Rizzo, T. M. and P. Palukaitis (1988). "NUCLEOTIDE-SEQUENCE AND EVOLUTIONARY RELATIONSHIPS OF CUCUMBER MOSAIC-VIRUS (CMV) STRAINS - CMV-RNA-2." *Journal of General Virology* **69**: 1777-1787.

Rizzo, T. M. and P. Palukaitis (1989). "NUCLEOTIDE-SEQUENCE AND EVOLUTIONARY RELATIONSHIPS OF CUCUMBER MOSAIC-VIRUS (CMV) STRAINS - CMV RNA-1." *Journal of General Virology* **70**: 1-11.

Romero, H., A. Zavala, et al. (2000). "Compositional pressure and translational selection determine codon usage in the extremely GC-poor unicellular eukaryote Entamoeba histolytica." *Gene* **242**(1-2): 307-311.

Rondon, M. R., P. R. August, et al. (2000). "Cloning the Soil Metagenome: a Strategy for Accessing the Genetic and Functional Diversity of Uncultured Microorganisms." *Applied and Environmental Microbiology* **66**: 2541-2547.

Rosani, U., L. Varotto, et al. (2011). "Massively parallel amplicon sequencing reveals isotype-specific variability of antimicrobial peptide transcripts in Mytilus galloprovincialis." *PLoS ONE* **6**(11): e26680.

Rothberg, J. M., W. Hinz, et al. (2011). "An integrated semiconductor device enabling non-optical genome sequencing." *Nature* **475**(7356): 348-352.

Runckel, C., M. L. Flenniken, et al. (2011). "Temporal Analysis of the Honey Bee Microbiome Reveals Four Novel Viruses and Seasonal Prevalence of Known Viruses, *Nosema*, and *Crithidia*." *PLoS ONE* **6**(6): e20656.

Rusch, D. B., A. L. Halpern, et al. (2007). "The Sorcerer II Global Ocean Sampling expedition: northwest Atlantic through eastern tropical Pacific." *PLoS Biology* **5**: e77.

Sachsenroder, J., S. Twardziok, et al. (2012). "Simultaneous Identification of DNA and RNA Viruses Present in Pig Faeces Using Process-Controlled Deep Sequencing." *PLoS ONE* **7**(4): e34631.

Saeed, I. and S. K. Halgamuge (2009). "The oligonucleotide frequency derived error gradient and its application to the binning of metagenome fragments." *BMC Genomics* **13**: 1-13.

Sanger, F., S. Nicklen, et al. (1977). "DNA SEQUENCING WITH CHAIN-TERMINATING INHIBITORS." *Proceedings of the National Academy of Sciences of the United States of America* **74**(12): 5463-5467.

Scheibye-Alsing, K., S. Hoffmann, et al. (2009). "Sequence assembly." *Computational Biology and Chemistry* **33**(2): 121-136.

Schloss, P. D. and J. Handelsman (2008). "A statistical toolbox for metagenomics: assessing functional diversity in microbial communities." *BMC Bioinformatics* **9**: 34.

Scholz, M. B., C. C. Lo, et al. (2012). "Next generation sequencing and bioinformatic bottlenecks: the current state of metagenomic data analysis." *Current Opinion in Biotechnology* **23**(1): 9-15.

Segata, N., L. Waldron, et al. (2012). "Metagenomic microbial community profiling using unique clade-specific marker genes." *Nature Methods* **9** (8): 811-814.

Seshadri, R., S. A. Kravitz, et al. (2007). "CAMERA: a community resource for metagenomics." *PLoS Biology* **5**: e75.

Simmons, M. P. (2008). "Potential use of host-derived genome signatures to root virus phylogenies." *Molecular Phylogenetics and Evolution* **49**: 969-978.

Simon, C. and R. Daniel (2011). "Metagenomic Analyses: Past and Future Trends." *Applied and Environmental Microbiology* **77**(4): 1153-1161.

Simpson, J. T. and R. Durbin (2012). "Efficient de novo assembly of large genomes using compressed data structures." *Genome Research* **22**(3): 549-556.

Sosnova, V. and Z. Polak (1975). "SUSCEPTIBILITY OF ARABIDOPSIS-THALIANA (L) HEYNH TO INFECTION WITH SOME PLANT VIRUSES." *Biologia Plantarum* **17**(2): 156-158.

Soylu, S., I. Brown, et al. (2005). "Cellular reactions in Arabidopsis following challenge by strains of Pseudomonas syringae: From basal resistance to compatibility." *Physiological and Molecular Plant Pathology* **66**(6): 232-243.

Sueoka, N. (1962). "ON GENETIC BASIS OF VARIATION AND HETEROGENEITY OF DNA BASE COMPOSITION." *Proceedings of the National Academy of Sciences of the United States of America* **48**(4): 582-592.

Sugar, I. P. and S. C. Sealfon (2010). "Misty Mountain clustering: application to fast unsupervised flow cytometry gating." *BMC Bioinformatics* **11**: 502.

Sun, J. Q., H. L. Jiang, et al. (2011). "Systemin/Jasmonate-Mediated Systemic Defense Signaling in Tomato." *Molecular Plant* **4**(4): 607-615.

Swarbreck, D., C. Wilks, et al. (2008). "The Arabidopsis Information Resource (TAIR): gene structure and function annotation." *Nucleic Acids Research* **36**(Database issue): D1009-1014.

Teeling, H., A. Meyerdierks, et al. (2004). "Application of tetranucleotide frequencies for the assignment of genomic fragments" *Environmental Microbiology* **6**: 938-947.

Teeling, H., J. Waldmann, et al. (2004). "TETRA: a web-service and a stand-alone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences." *BMC Bioinformatics* **5**: 1-7.

The Arabidopsis Genome Initiative (2000). "Analysis of the genome sequence of the flowering plant Arabidopsis thaliana." *Nature* **408**(6814): 796-815.

Thilmony, R., W. Underwood, et al. (2006). "Genome-wide transcriptional analysis of the Arabidopsis thaliana interaction with the plant pathogen Pseudomonas syringae pv. tomato DC3000 and the human pathogen Escherichia coli O157 : H7." *Plant Journal* **46**(1): 34-53.

Tian, S. W., H. Y. Yang, et al. (2008). "An Improved K-Means Clustering Algorithm Based on Spectral Method." *Advances in Computation and Intelligence, Proceedings* **5370**: 530-536.

Tibshirani, R. and G. Walther (2011). "Cluster Validation by Prediction Strength." *Journal of Computational and Graphical Statistics* **14**: 511- 528.

Tibshirani, R., G. Walther, et al. (2001). "Estimating the number of clusters in a data set via the gap statistic." *Journal of the Royal Statistical Society B* **63**: 411-423.

Torsvik, V., R. Sørheim, et al. (1996). "Total bacterial diversity in soil and sediment communities—A review." *Journal of Industrial Microbiology & Biotechnology* **17**: 170-178.

Toulza, E., A. Tagliabue, et al. (2012). "Analysis of the global ocean sampling (GOS) project for trends in iron uptake by surface ocean microbes." *PLoS ONE* **7**(2): e30931.

Tringe, S. G. and P. Hugenholtz (2008). "A renaissance for the pioneering 16S rRNA gene." *Current Opinion in Microbiology* **11**(5): 442-446.

Tringe, S. G., C. von Mering, et al. (2005). "Comparative metagenomics of microbial communities." *Science* **308**: 554-557.

Tse, H., A. K. L. Tsang, et al. (2012). "Identification of a Novel Bat Papillomavirus by Metagenomics." *PLoS ONE* **7**(8): e43986.

Tyson, G. W., J. Chapman, et al. (2004). "Community structure and metabolism through reconstruction of microbial genomes from the environment." *Nature* **428**(6978): 37-43.

Ulrich, L. E. and I. B. Zhulin (2010). "The MiST2 database: a comprehensive genomics resource on microbial signal transduction." *Nucleic Acids Research* **38**(suppl 1): D401-D407.

Ultsch, A. and M. Fabian (2005). "ESOM-Maps : tools for clustering , visualization , and classification with Emergent SOM." *Technical Report Department of Mathematics and Computer Science, University of Marburg, Germany*. **46**: 1-7.

Ussery, D. W., T. M. Wassenaar, et al. (2009). "Genomic Properties: Length, Base Composition and DNA Structures." *Computing for Comparative*

Microbial Genomics: Bioinformatics for Microbiologists, Springer London. **8**: 111-135.

Venter, J. C., K. Remington, et al. (2004). "Environmental genome shotgun sequencing of the Sargasso Sea." *Science* **304**: 66-74.

Vinga, S. (2003). "Alignment-free sequence comparison--a review." *Bioinformatics* **19**: 513-523.

Viswanath, P. and V. Suresh Babu (2009). "Rough-DBSCAN: A fast hybrid density based clustering method for large data sets." *Pattern Recognition Letters* **30**: 1477-1488.

Voget, S., C. Leggewie, et al. (2003). "Prospecting for Novel Biocatalysts in a Soil Metagenome." *Applied and Environmental Microbiology* **69**: 6235-6242.

Volinia, S., G. A. Calin, et al. (2006). "A microRNA expression signature of human solid tumors defines cancer gene targets." *Proceedings of the National Academy of Sciences of the United States of America* **103**(7): 2257-2261.

Voss, R. F. (1992). "Evolution of Long-Range Fractal Correlations and 1/f Noise in DNA Base Sequences." *Physical Review Letters* **68**: 3805-3808.

Wada, A. and A. Suyama (1986). "LOCAL STABILITY OF DNA AND RNA SECONDARY STRUCTURE AND ITS RELATION TO BIOLOGICAL FUNCTIONS." *Progress in Biophysics & Molecular Biology* **47**(2): 113-157.

Wang, H.-C., E. Susko, et al. (2006). "On the correlation between genomic G+C content and optimal growth temperature in prokaryotes: data quality and confounding factors." *Biochemical and Biophysical Research Communications* **342**: 681-684.

Wang, J., G. K. S. Wong, et al. (2002). "RePS: A sequence assembler that masks exact repeats identified from the shotgun data." *Genome Research* **12**(5): 824-831.

Wang, M. J., Y. Z. Ye, et al. (2012). "A de Bruijn Graph Approach to the Quantification of Closely-Related Genomes in a Microbial Community." *Journal of Computational Biology* **19**(6): 814-825.

Waterman, M. S. and T. F. Smith (1981). "Identification of Common Molecular Subsequences Identification of Common Molecular Subsequences." *J. Mol. Biol.* **147**: 195-197.

Watson, J. and F. Crick (1953). "Molecular Structure of Nucleic Acids - A Structure for Deoxyribose Nucleic Acid." *Nature* **171**: 737-738.

Weber, R., H.-J. Schek, et al. (1998). "A Quantitative Analysis and Performance Study for Similarity-Search Methods in High-Dimensional Spaces." *Proceedings of the 24rd International Conference on Very Large Data Bases*: 194-205.

Wendl, M. C. (2006). "A general coverage theory for shotgun DNA sequencing." *Journal of Computational Biology* **13**(6): 1177-1196.

Wetterstrand, K. (2012). "DNA Sequencing Costs: Data from the NHGRI Large-Scale Genome Sequencing Program. Available at: www.genome.gov/sequencingcosts." Retrieved 22/05/2012, 2012.

Whalen, M. C., R. W. Innes, et al. (1991). "IDENTIFICATION OF PSEUDOMONAS-SYRINGAE PATHOGENS OF ARABIDOPSIS AND A BACTERIAL LOCUS DETERMINING AVIRULENCE ON BOTH ARABIDOPSIS AND SOYBEAN." *Plant Cell* **3**(1): 49-59.

Wheeler, D. A., M. Srinivasan, et al. (2008). "The complete genome of an individual by massively parallel DNA sequencing." *Nature* **452**(7189): 872-U875.

Wiesinger-Mayr, H., K. Vierlinger, et al. (2007). "Identification of human pathogens isolated from blood using microarray hybridisation and signal pattern recognition." *BMC Microbiology* **7**: 1-17.

Williamson, S. J., L. Z. Allen, et al. (2012). "Metagenomic Exploration of Viruses throughout the Indian Ocean." *PLoS ONE* **7**(10): e42047.

Williamson, S. J., D. B. Rusch, et al. (2008). "The Sorcerer II Global Ocean Sampling Expedition: metagenomic characterization of viruses within aquatic microbial samples." *PloS one* **3**: e1456.

Willner, D., R. V. Thurber, et al. (2009). "Metagenomic signatures of 86 microbial and viral metagenomes." *Environmental Microbiology* **11**: 1752 -1766.

Woyke, T., H. Teeling, et al. (2006). "Symbiosis insights through metagenomic analysis of a microbial consortium." *Nature* **443**(7114): 950-955.

Wu, D., S. C. Daugherty, et al. (2006). "Metabolic complementarity and genomics of the dual bacterial symbiosis of sharpshooters." *PLoS Biology* **4**(6): 1079-1092.

Wu, H., Z. Zhang, et al. (2012). "On the molecular mechanism of GC content variation among eubacterial genomes." *Biology Direct* **7**(1): 1-16.

Yan, D. and M. I. Jordan (2009). "Fast Approximate Spectral Clustering." *15th ACM Conference on Knowledge Discovery and Data Mining (SIGKDD), Paris, France*: 907-916.

Yeung, K. Y., D. R. Haynor, et al. (2001). "Validating clustering for gene expression data." *Bioinformatics* **17**(4): 309-318.

Yilmaz, P., M. H. Iversen, et al. (2012). "Ecological structuring of bacterial and archaeal taxa in surface ocean waters." *Fems Microbiology Ecology* **81** (2): 373-385.

Yooseph, S., G. Sutton, et al. (2007). "The Sorcerer II Global Ocean Sampling expedition: Expanding the universe of protein families." *PLoS Biology* **5** (3): 432-466.

Zhang, R. and C.-T. Zhang (2004). "Isochore structures in the genome of the plant Arabidopsis thaliana." *Journal of Molecular Evolution* **59**: 227-238.

Zhang, T., R. Ramakrishnan, et al. (1997). "BIRCH: A New Data Clustering Algorithm and Its Applications." *Data Mining and Knowledge Discovery* **1** (2): 141-182.

Zhang, X. C. and Q. Z. You (2011). "An improved spectral clustering algorithm based on random walk." *Frontiers of Computer Science in China* **5**(3): 268-278.

Zoubak, S., O. Clay, et al. (1996). "The gene distribution of the human genome." *Gene* **174**(1): 95-102.