Extending the Graphical Representation of four KEGG Pathways for a Better Understanding of Prostate Cancer Using Machine Learning of Graphical models

Adel Abdullah M Aloraini

Ph.D. Thesis

The University of York
Department of Computer Science

March 2011

Abstract

This thesis shows a novel contribution to computational biology alongside with developed machine learning methods. It shows how the graphical representation of KEGG pathways can be refined using machine learning of graphical models. The focus mainly is on a set of graphical models called Bayesian networks. Throughout this thesis , different ways of learning Bayesian networks are discussed. The work is based on Affymetrix gene expression microarray profiles and penalised Gaussian linear models. Penalisation in linear models includes choosing the most important parents and estimating the associated coefficients simultaneously using **L1**-regression. The sparse dataset that is generated from Affymetrix microarray technology is the key point in this thesis when learning Bayesian networks. Thus, the work in this thesis can be viewed as developing robust methods to avoid overfitting that usually associated with gene expression datasets and contributing to invoke more details about a well known discrepancy in KEGG pathways. So, the problem we have is to learn from a *large number of candidates, small samples*,($p \gg n$), and for such problem the goal is to apply model selection methods that hopefully achieve an accurate prediction , interpretable models, and stable models. The prediction and the most powerful predictors can be improved by using methods that trade-off between bias and variance. Also, providing which predictors are meaningful rather than using all predictors will provide interpretable models, and finally by choosing the most important predictors, a small change in the data will not result in large changes in the subset of predictors which consequently gives the stability to the models that are learnt.

Contents

List of Tables

List of Figures

Acknowledgements

All praise is due to Allah the Almighty who has given me the strength and the ability to complete my thesis. My warmest gratitude to my supervisor Dr.James Cussens for being helpful whenever I needed. I really appreciate his patience and interest in my work. Dr.James has been more than supervisor for his continuous support; during my PhD and in the field study I have been in abroad. I would like also to thank my assessor Dr.Daniel Kudenko for his support during the meeting we have held through all my PhD. Dr.Alastair Droop and Dr.Karim El Sawy have been very friendly during the time I have spent at York Centre for Complex Systems Analysis(YCCSA) and specially the experience I have in R-package from them. I would like to record a special respect to my wife for being so patient and emotionally close to me during all the time I have spent away from home. Finally, a deep respect to my parents for being capable of awaiting the end of my PhD and looking forward to seeing me a righteous son.

Declaration

This thesis has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree other than Doctor of Philosophy of the University of York. This thesis is the result of my own investigations, except where otherwise stated. Other sources are acknowledged by explicit references.

I hereby give consent for my thesis, if accepted, to be made available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed .....(candidate)

Date .....

Some of the material contained in this thesis has been published:

1. Adel Aloraini,James Cussens and Richard Birnie. Extending KEGG Pathways for a Better Understanding of Prostate Cancer Using Graphical Models. In *Proceedings of the 3rd International Workshop on Machine Learning in Systems Biology(MLSB)*, Slovenia , September 2009.

2. Adel Aloraini,James Cussens and Richard Birnie. Extending Prostate Cancer KEGG Pathways Using Machine Learning of Graphical Models. In *Proceeding of the 3rd International Conference on Bioinformatics and Systems Biology (BSB 2010)*,China,July 2010.

CHAPTER 1

Introduction and Motivation

.This chapter gives an overview of the field of Artificial Intelligence in microarray analysis (Section 1.1). It discusses the validity of machine learning when sparse datasets are presented, especially from microarray technologies. Section 1.2 points out one of the main difficulties that exists when dealing with microarray datasets, the dimensionality of gene expression datasets and the overfitting problem that is associated with this. Section 1.3 gives a detailed account of the motivation and the research hypothesis behind the work in this thesis, which is based on a well known graphical representation for molecular interactions, the Kyoto Encyclopedia of Genes and Genomes (KEGG), and how machine learning of graphical models can be used to invoke more detailed knowledge about molecular interactions in KEGG. Finally, a compact view of the thesis chapters is given in Section 1.4.

1.1 Introduction

Recently, Artificial Intelligence (AI) has become an interdisciplinary field with medicine and biology. The huge amount of data that is available from modern technologies in medicine and biology is the key to AI success in developing medical treatments and tracking how cellular systems work inside the body. One of the prominent AI branches that can be used to develop therapies and discover knowledge about genomic interactions is the discipline of machine learning. Machine learning has been used intensively as a tool to reveal and discover complex molecular biological interactions that could not have been found manually or might have taken a long time to be discovered in the biological laboratory.

One of the most successful technologies, which transformed 20th century genetics into 21st century genomics, is microarray technology. The microarray has become one of the main tools in wet-labs that conduct molecular experiments. Microarrays are used to measure the expression level of thousands of genes simultaneously. In the face of such vast amounts of gene expression data, powerful methods are needed to extract useful knowledge and make sense of this data for further analysis. In the past, no more than a few genes could be studied at a time. However, the human body has more than 20,000 genes; to study each gene one at a time would take an extremely long time, but with the rapid development of high throughput technologies, such as the microarray, it has become possible to look at all the cell genes in one go. The result of using microarrays is rather complex data, which a biologist cannot analyse without using powerful tools to help pre-process, reveal and visualise important findings from such data. Therefore, machine learning and statistical methods are found to provide useful solutions for the genomic and proteomic era.

One of the hot topics within both the biology and medicine communities is how to treat cancer. The treatment of cancer has brought different disciplines together to contribute to improving human life. At the present time, cancer is stopped or treated by using chemotherapy, which is sometimes effective to stop the spread of cancer, but also results in side-effects to the healthy cells inside the body. Therefore, efforts are being made to find advanced therapies that can be used to target the cancerous cells alone, rather than the whole body, which will result in preventing an adverse effect to other cells in the body. The study of cancer using clinical factors (such as age, weight, etc.) has not been found enough on its own to diagnose and treat cancer. Therefore, DNA-microarray technology is being used to look at the low level causes of cancer. As an example, a healthy cell and a cancer cell can be compared using DNA-microarrays to look at the differences and the similarities between these two cells and make a decision on which genes should be targeted and diagnosed as causes of cancer. The result of microarray experiments, which are gene expression profiles, is generally massive data, which fits perfectly to machine learning algorithms. The ability of machine learning to search and find different hidden information from gene expression datasets has already been proven. One example is finding interactions at a low level between thousands of genes, resultant from microarray experiments.

1.2 On the Dimensionality of Gene Expression Microarrays

One of the main challenges in dealing with microarray data is the dimensionality of the data. Gene expression data usually has thousands of genes as variables and they are measured in only a few samples. Therefore, getting reliable information from such large dimensional datasets is hard. Algorithms that can robustly deal with a big feature space are required. Thus, one

of the main motivations in this thesis is to develop a machine learning algorithm that not only works in high dimensional space, but also with small sample sizes. One problem that is often encountered when dealing with microarray gene expression data is overfitting, which occurs when thousands of genes and small sample sizes are used for learning. This problem will be investigated throughout the thesis. We will show how the difficulty of dimensionality can be reduced using natural prior knowledge from molecular biology interaction resources such as KEGG pathways (Kanehisa & Goto 2000; Kanehisa et al. 2010).

1.3 Motivation

In this section, a discussion of the state-of-the-art in the research hypothesis area will be given which covers the discussion of a known discrepancy in the KEGG pathways. This section will also discuss the origin of the gene families represented in KEGG pathways and how machine learning of graphical models can increase the biological complexity in the KEGG signalling pathways by adding extra information about how gene families interact with one other. Finally, the formalisation of the proposed method will be given, as well as the evaluation used in this thesis.

1.3.1 KEGG (Kyoto Encyclopedia of Genes and Genomes)

The amount of genome sequence data, which has increased in the last few decades, is at the core of understanding life as a molecular system. It also helps greatly in developing medical and pharmaceutical applications. The Kyoto Encyclopaedia of Genes and Genomes (KEGG), a knowledge-based method developed in 1995, aids our understanding of the high-order systematic behaviour of cells and organisms, based on genomic and molecular systems (Kanehisa & Goto 2000; Kanehisa et al. 2010).

KEGG is a graphical representation that is used for analysing:

- Gene functions (KEGG GENES): a group of gene categories for all the completely sequenced genomes and some partial genomes, with up-to-date annotations of gene functions.
- KEGG LIGAND: consisting of chemical building blocks for endogenous and exogenous substances.
- KEGG PATHWAYS: represent molecular relationships and reactions networks. These networks can be grouped as follows:
    1. The set of chemical reactions (metabolism) that happen in living organisms. These processes allow organisms to grow, reproduce and respond to their environment.
    2. Genetic information processing, such as DNA replication.

3. Environmental information processing, such as signalling molecules and interactions.
4. Cellular processes, such as the growth and death of cells.
5. Human diseases, such as cancer. Figure 1.1 shows an example of how genes react with one another in the context of the cancer signalling pathway shown in KEGG. The interpretation of notations used in Figure 1.1 is shown in Figure 1.2. As can be seen from Figure 1.2, KEGG pathways mostly represent protein-protein interactions, but some are at the gene expression level.

- KEGG BRITE: a collection of hierarchical classifications showing knowledge of various aspects of biological systems. In comparison with KEGG PATHWAY, KEGG BRITE provides many different types of relationships not shown in KEGG PATHWAYS, so that KEGG BRITE can be viewed as a global picture of KEGG PATHWAYS. Figure 1.3 shows a hierarchical diagram of the different kinds of representations available in KEGG.

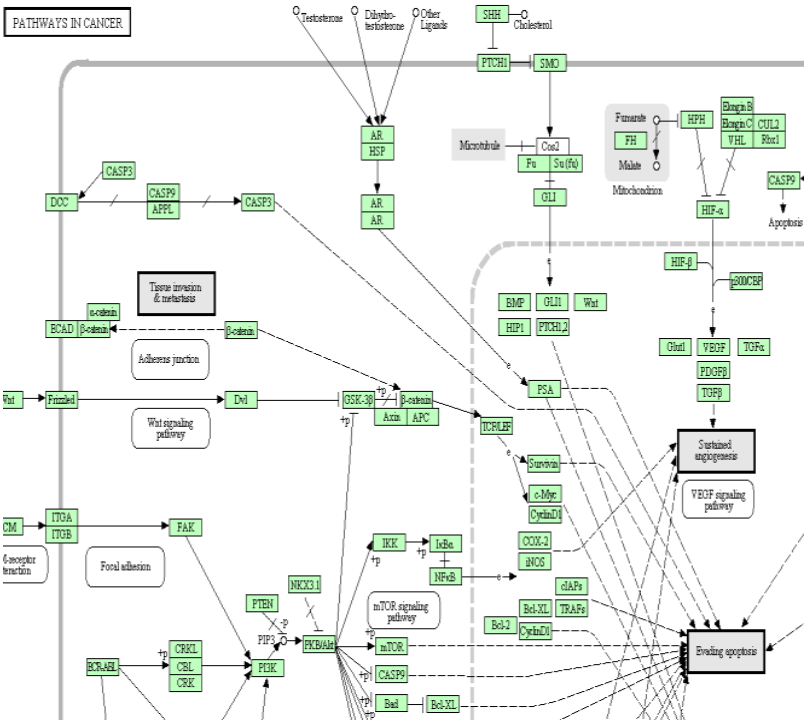


Figure 1.1: Part of cancer pathways as shown in KEGG [Kanehisa Laboratories, 2009].

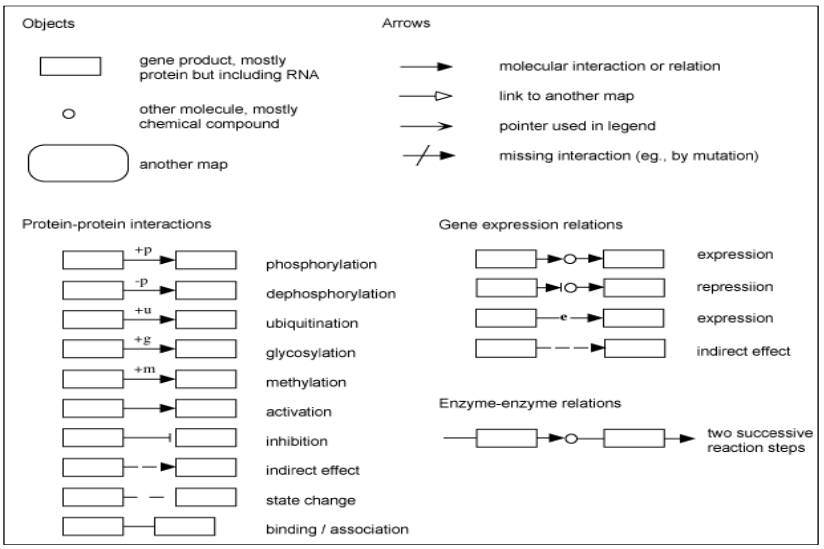


Figure 1.2: Notations in KEGG pathways diagrams[Kanehisa Laboratories, 2009].

Traditionally, biologists have attempted to understand cellular processes using a reductionist approach, which looks at the effects of manipulating a small number of genes in a living organism. KEGG pathways aid as a means of visualising the interactions between genes, mostly at the protein level, in the form of cell signalling pathways. The PATHWAY component in the KEGG database provides generic representations of cell signalling pathways. For example, in the WNT signalling pathway, depicted in Figure 1.4, KEGG shows that WNT proteins interact with Frizzled proteins (FZD). There are 19 WNT proteins (WNT3, WNT5, WNT7,...) in the WNT family and 10 FZD proteins (FZD1, FZD2, FZD7,...) in the Frizzled family listed in the KEGG database.

A gene/protein family in KEGG PATHWAYS is a group of genes/proteins that are grouped together. The groupings are based on different criteria. One reason is gene duplication, a process by which a chromosome or a segment of DNA is duplicated, resulting in an additional copy of a gene, which evolves through mutations to create new different functional genes that share important characteristics. For example, similar sequences of DNA building blocks (nucleotides) (Zhang 2003). A well-known mechanism after gene duplication is *neofunctionalization* in which one of the duplicates keeps the inherited functions, while the other continues to evolve for new functions (Tirosh & Barkai 2007). Another reason for grouping genes/proteins together in one family is that the proteins produced from genes in the same family work together in the same processes, which are needed for living organisms.

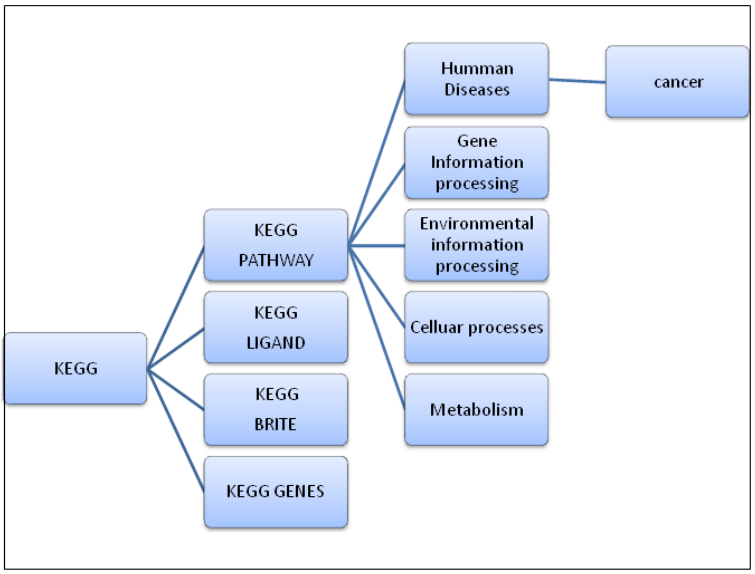


Figure 1.3: A hierarchical diagram illustrating KEGG components

1.3.2 Research hypothesis

KEGG pathways provide a useful level of abstraction for understanding the overall structure of cell signalling pathways. Each signalling pathway gives an overview of how gene families react with each other at the protein level or at the gene expression level. For example, the interaction between the WNT family and the Frizzled family is at the protein level, as shown in Figures 1.2 and 1.4. However, each gene/protein family can have several genes, for reasons mentioned previously. For example, in the WNT-signalling pathway, shown in Figure 1.4, the WNT family that appears in the upper left-hand corner has 19 proteins, including WNT5a, WNT6 and WNT10. Moreover, some gene families exist in all four pathways, which provides an opportunity in the future to link the pathways together to provide a better understanding of how a disease, for example cancer, develops. As a result, treatment could be limited to genes rather than the whole body. This means that if we can find the genes that are responsible for inducing and stimulating cancer to grow, we can target them, instead of applying chemotherapy to the whole body. In addition, the better we understand the cellular reaction, the better we understand the disease.

Furthermore, there is an increasing availability of gene expression data on the whole genome level, in the form of DNA-microarray experiments, coupled to mathematical and computational techniques that can take account of the relationships between large numbers of genes. Using these techniques will potentially enhance our understanding of the higher order molecular systems that regulate cellular growth.

it is well known that a gene produces a protein via a transcriptional step called Ribonucleic acid(RNA). It has been studied extensively in (Webb & Westhead 2009) that it is relevant to use gene co-expression to indicate potential functional linkage at the protein level. Moreover, the level of transcriptional regulation of protein complexes has been examined in (Jansen et al. 2002) and it was found that the presence of certain interactions between protein complexes is directly associated with the coherency of their expressions at the transcriptional levels. Therefore, the interaction of genes at the gene expression level can indirectly enhance our understanding of how protein-protein interaction might occur. Also, since protein-protein interactions in the same pathway are most often co-expressed, then finding a more detailed picture of how the genes interact with each other at the gene-expression level will possibly lead to understanding the interaction of these genes at the protein level. Hence, the work in this thesis will be as an indirected learning methods,via gene expression signatures, to understand most of the unknown protein-protein interaction between families represented in the KEGG PATHWAYS in addition to understand the detailed interaction between gene families that are already represented in KEGG, at the gene expression level.

This thesis looks at the prostate cancer disease networks that are part of human disease, in the KEGG PATHWAYS, (Figure 1.3). The focus is on research by (Birnie et al. 2008), in which microarrays were used to compare gene expression patterns between prostate cancer samples and benign controls to identify genes that have significantly different gene expression signatures in their stem cells from those in committed basal cells. Samples from cancer and non-cancer were used in this study. The gene expression signatures described in (Birnie et al. 2008) were found to be enriched for genes from four main KEGG PATHWAYS, JAK-STAT signalling, WNT signalling(Figure 1.4), the cell-extracellular matrix interaction pathway and the focal adhesion signalling pathway. It is increasingly apparent that studying small numbers of genes in isolation does not provide sufficient understanding of the higher order systemic processes that regulate cell growth. Thus, we are becoming interested in finding methods that provide a picture of how genes inside gene families might interact/co-express with each other and also how they interact with those around them, based on transcriptional gene expressions. In general, KEGG pathways only show a higher level of interaction between gene families. More precisely there is no existing mechanism to access the specific connections between gene families that underlie the generic connections represented in the KEGG signalling diagrams. For example, the WNT signalling pathway in Figure 1.4 shows that the WNT family/component directly interacts with Frizzled, but it does not show which member of the WNT family interacts with which in the Frizzled family or how genes inside the WNT family interact with each other.

Discovering the low level interaction between gene families will provide new insights into genome

evolution, because for example neofunctionalization resultant from gene duplication is believed to involve novel functionality for the new duplicated genes (Tirosh & Barkai 2007).

Thus, the main contribution of this thesis is to extend and refine the representation of the four KEGG PATHWAYS mentioned in (Birnie et al. 2008), to include more details and additional knowledge about the molecular representations and reaction networks. This is done by using machine learning of graphical models on gene expression data. Hence, this thesis contributes to computational biology, along with developing machine learning methods. The focus is mainly on a set of graphical models called Bayesian networks, and throughout this thesis, we present and discuss different ways of learning Bayesian networks. Since microarray gene expression profiles are used when learning graphical models, an overfitting problem is a concern. Gene expression datasets often have more genes than samples ($p >> n$) which makes it difficult to learn meaningful cellular graphs for KEGG PATHWAYS. Therefore, we show extensively how it is possible to overcome the overfitting problem when sparse datasets are used. For this purpose, penalised Gaussian linear models are used. Penalisation in linear models includes choosing the most important parents and estimating the associated coefficients simultaneously using **L1***-regression. Thus, another view of the contribution of this thesis is the development of robust methods to avoid the overfitting which is usually associated with gene expression datasets. Finally we evaluated the generated models based on prediction accuracy using leave-one-out-cross validation(LOOCV).*

1.4 Thesis Structure

This section gives a synopsis of all chapters covered in the thesis.

- **Chapter 1:** gives an introduction to the field of interest. It shows the use of machine learning in microarray data analysis, along with the potential problems arising when using small sample gene expression datasets. The last section focuses on the motivation behind this thesis and a detailed discussion of the discrepancy in KEGG pathway representations is covered.

- **Chapter 2:** shows background materials for techniques related to the problem of interest. It presents a general discussion of different machine learning algorithms that are used in microarray data analysis. It also considers how machine learning of graphical models is used to infer cellular systems, and different sets of graphical models are discussed. It then shows how machine learning is used to learn a well known set of graphical models, Bayesian networks, and is concerned with learning the structure of Bayesian networks and parameter estimation. This chapter also shows some work related to supervised and unsupervised learning algorithms in cancer and also for graphical models of cellular systems.

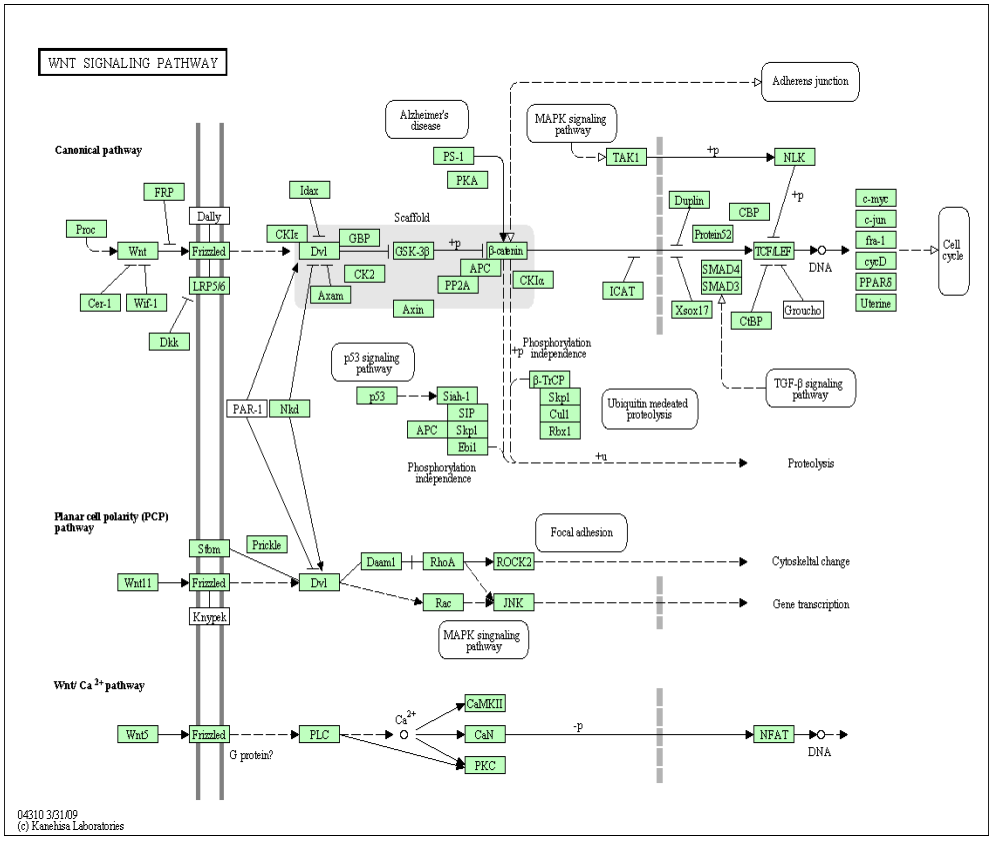


Figure 1.4: Wnt signalling Pathway [Kanehisa Laboratories, 2009]

- **Chapter 3:** has a broad discussion of the two types of cancer that are known to be the most prone to metastasis (spreading), breast cancer and prostate cancer. It gives also a detailed discussion about cell communication and Wnt-signalling pathway. At the end of the chapter, we stress how genomic diagnosis is more robust than using clinical factors such as age, gender, etc. in diagnosing cancer.

- **Chapter 4:** goes into detail about microarray technology. It shows different types of microarray technologies. It also gives a detailed discussion of how gene expression profiles are normalised using statistical methods. Finally, it shows how the gene expression datasets that are used in this thesis are normalised.

- **Chapter 5:** is an experimental chapter in which the existing tools for learning Bayesian networks are discussed. It highlights the advantages and disadvantages of using some existing tools and introduces the new direction of how we will learn Bayesian(*causal*) networks from gene expression datasets, which will lead to a more detailed representation of four KEGG pathways.

- **Chapter 6:** is about the main contribution of this thesis. It shows that the assumption of multivariate normal distribution can be used to learn linear Gaussian models. It also shows how co-expression networks can be learned by pair-wise correlation. This chapter highlights the drawback that is encountered when co-expression networks are inferred. It discusses how AIC and BIC scoring functions are used to learn models with *less* overfitting from small sample sizes. It then introduces more severe methods against overfitting, which were developed and could successfully learn meaningful causal networks from sparse gene expression datasets.
- **Chapter 7:** gives a summary of all chapters included in the thesis. It points out the limitations of the current work and looks at future work that is related to the context of this thesis.

CHAPTER 2

Background

This chapter covers the background of the field of machine learning , graphical models, and their applications to inferring cellular networks, including reviewing the literature. Section 2.1 gives an introduction to Artificial Intelligence in medicine and biology in the context of cancer and therapy development and the potential solutions machine learning might offer. Different machine learning algorithms are discussed in Section 2.2, which discusses supervised learning algorithms and their applications for class discovery in gene expression datasets, as well as unsupervised learning algorithms with their applications in gene expression datasets from microarrays. In Section 2.3, graphical models in general are discussed. This includes the graphical representation for each set of graphical models and the dependency properties encoded in each set of graphical representations. Section 2.4 details how to learn a Bayesian network, in addition to the problems usually encountered when learning from datasets. Section 2.4.2 explains how to estimate the parameters present after the graph is learned from the data. In Section 2.4.3, inference in Bayesian networks is discussed, including exact inference and approximate inference. In Section 2.5, a broad survey of machine learning in cancer diagnosis and inferring cellular networks is shown.

2.1 Introduction

From the earliest beginnings of the modern computer, scientists hoped to create an *electronic brain* with all the modern technological requirements. Scientists and doctors were captivated by the potential such a technology might have in medicine; using the ability of intelligent systems, such as machine learning, to store and process vast amounts of knowledge. The ambition was that it would become *a doctor in a box* to assist and help clinicians and biologists with tasks

like diagnosis and genomic analysis (Coiera 2003). Such motivations made it possible to create a small community of computer scientists and health-care professionals who initiated a research programme for a new discipline called Artificial Intelligence in medicine (AIM). An early definition was: *Medical Artificial Intelligence is primarily concerned with construction of AI programs that perform diagnosis and make therapy recommendations* (Fentiman 1998). Since then, Artificial Intelligence in medicine and biology has become increasingly popular, as scientists realise the complexity of making certain decisions to treat particular diseases. Furthermore, the use of machine learning and data mining as tools in medical diagnosis and biology labs has become important, since the advantages of genomic technology, such as the Affymetrix microarray became known. One practical use of machine learning is to reveal knowledge from vast genomic data from microarray platforms.

Cancer is a critical disease, leading to death if not treated in its early stages. The disease is very common and the second highest cause of death. In this chapter a broad survey of how machine learning is used in bioinformatics in the context of cancer will be given. Related work in different machine learning algorithms concerned with inferring cellular networks will also be covered.

2.2 Machine Learning for Microarray Analysis

One of the main uses of microarray experiments is to find and infer meaningful relationships between genes. In this section, we will look closely at how machine learning can offer useful methods for this purpose.

2.2.1 Supervised learning

In supervised learning algorithms, the data that is used has known classes and so the result is a classifier that can later be used in predicting the classes for an unknown sample. Decision trees, neural networks, naive Bayes, and support vector machines (SVM) are well known supervised learning algorithms used in different applications.

2.2.1.1 Class discovery and prediction

In class discovery and prediction in respect of microarray data analysis, the aim is to predict the class for a new sample. For example, it is possible to find out whether a gene based on its transcriptional expression belongs to a malignant cell or a benign cell based on the training gene expression that is used to train a classifier or a predictive model. In this section will look at SVMs (Vapnik 1998), which are one of the main supervised learning algorithms used in class discovery and prediction for microarray gene expression data, and often used in gene expression data analysis.

2.2.1.2 Support vector machines (SVMs)

A support vector machine (SVM) is a supervised learning algorithm that uses points to train a classifier which can then be used to predict the class of unknown future samples. In SVM, the classes of the training samples are determined by constructing a separable hyperplane between training data points, which separates the data points into two classes. Thus, it is a binary classification method which aims to find the optimal hyperplane in the *feature space* that can separate the data points into two classes. The advantage of SVMs is that they can work with large dimensional data, such as gene expression datasets, by mapping to a higher dimensional feature space, in which a separable hyperplane can be constructed. The mapping is made using a *kernel* function that has a mapping (Φ) from input space to feature space and that the aim to find the optimal separable hyperplane. Figure 2.1 shows how the training data is mapped to a higher dimensional space. A hyperplane is then fitted to separate the data points into two classes, in a way that minimises the prediction error for a new, unknown class of future samples.

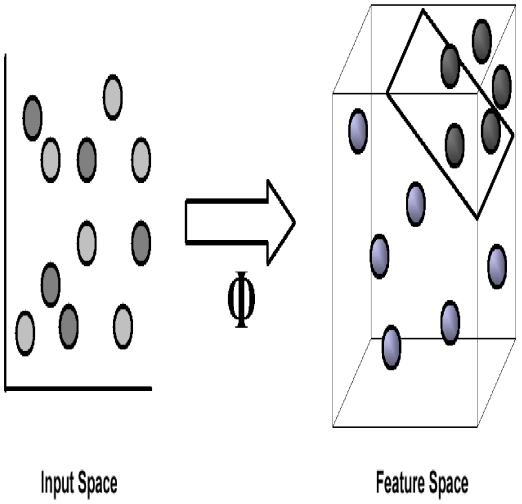


Figure 2.1: A kernel function is used to project the input data to a higher dimensional space where the hyperplane is constructed (Newton 2001).

2.2.2 Unsupervised learning

In contrast to supervised learning algorithms, unsupervised learning algorithms use data without prior knowledge of the classes and try to predict the classes through learning. Self organising maps, Bayesian networks and clustering algorithms are the ones of the main unsupervised learning algorithms used in gene expression analysis. In this section, we present a discussion about clustering, which is one of the main unsupervised learning algorithms used in microarray data analysis.

2.2.2.1 Clustering

Clustering algorithms are used in microarray data analysis to find the groups of genes that have similarity in function. A typical example of this can be seen when different microarray experiments are conducted across different conditions and a clustering algorithm is used to group the genes with similar gene expressions over the experiments. A variety of clustering techniques have been applied to microarray data and here we will describe two of the most widely used techniques, partitioning clustering algorithms and hierarchical clustering algorithms.

Partitioning algorithms: in partitioning algorithms, a dataset is partitioned into **K** clusters, based on the similarity between data points in the dataset. The similarity is measured using a distance measure such as *Euclidean distance* (2.1).

$$d(i,j) = \sqrt{(x_{i1} - x_{j1})^2 + (x_{i2} - x_{j2})^2 + ... + (x_{ip} - x_{jp})^2} \tag{2.1}$$

Some properties of Euclidean distance and any metric in general are:

- d(i,j)$\geq$0, where *i,j* are two points from the dataset.
- d(i,i)=0
- d(i,j)=d(j,i)
- d(i,j)$\leq$ d(i,k) + d(k,j)

Hierarchical clustering: the idea of hierarchal clustering is to form a tree in which the root takes all possible clusters and the leaves form every single data point each in its own cluster, in between there are different layers of clustering, (Figure 2.2). The nodes in the tree can be viewed as different stages of clustering. The uppermost node (the root) contains all the data points in one cluster. Then, the nodes in the second layer separate the data points to different clusters based on the similarity between these data points. The next layer spearates the clusters more until each data point has its own cluster which forms the leaves of the tree. In contrast to **partitioning clustering** algorithms, where data is partitioned into a particular cluster in one step, the data in hierarchical clustering algorithms is partitioned in a series of steps.

There are two types of hierarchal clustering: *agglomerative methods* and *divisive methods*. There are three kinds of agglomerative methods: *single linkage*, *complete linkage* and *average linkage*. *Single linkage clustering*, also called nearest neighbour technique (NN), is one of the simplest agglomerative hierarchal clustering algorithms. The distance between each cluster in the single linkage method is defined as the distance between the closest points in two clusters, (Figure 2.3). *Complete linkage clustering* is also known as the farthest neighbour clustering method. The distance in complete linkage clustering is defined as the farthest distance between

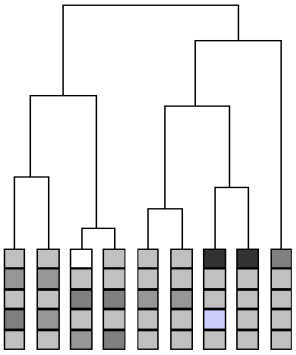


Figure 2.2: Hierarchical clustering (Newton 2001).

two points in two clusters, (Figure 2.4). The distance in *average linkage clustering* is the average distance between all points in two clusters, (Figure 2.5).

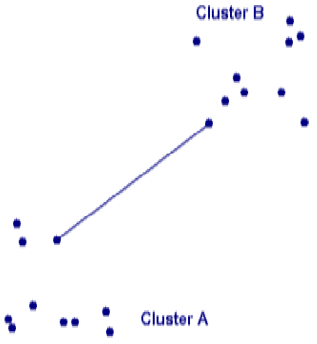


Figure 2.3: Single linkage clustering.

2.2.2.2 Inferring cellular networks

Clustering algorithms provide a technique for discovering genes that are co-expressed. However, as well as finding the similarity in expression between genes, it is also possible to infer the transcriptional regulation between genes and construct a meaningful interaction network between genes based on gene expression values. There are various graphical models techniques that can be applied to inferring gene networks and discovering interactions between genes. In the next section, we present a broad discussion of the most commonly applied graphical models in microarray data.

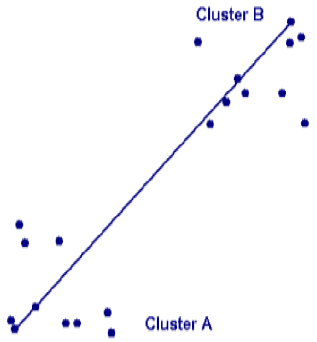


Figure 2.4: complete linkage clustering.

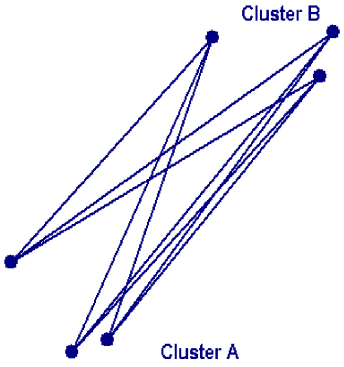


Figure 2.5: Average linkage clustering.

2.3 Overview of Graphical Models

In this section, an overview of different sets of graphical models, found to be the most commonly used representation of variable relationships in a graph, will be given. The four sets of graphs are : Markov networks (also called full conditional graphs), Bayesian networks, dependency networks, and co-expression networks.

2.3.1 Markov networks

Markov networks, also called full conditional models (Figure 2.6), are undirected graphical models. For Markov networks, if the dataset is assumed to follow a normal distribution $\sim N_p(\mu, \Sigma)$ with mean μ and covariance matrix Σ, and the covariance matrix Σ is invertible (Σ^{-1}, called a precision matrix), the value $-k_{ij}/\sqrt{k_{ii}k_{jj}}$ in the precision matrix is *the partial correlation coefficient* between variables i and j. Therefore, it holds for $i, j \in V$ with $i \neq j$ that:

$$X_i \perp X_j | \mathbf{X_{rest}} \Leftrightarrow k_{ij} = 0.$$

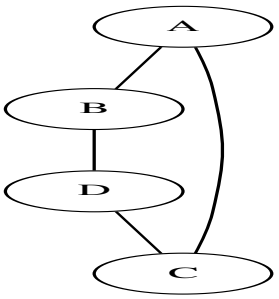


Figure 2.6: an example of Markov Network.

Where $\mathbf{X_{rest}}$ is all variables except (X_i, X_j). Thus, X_i is conditionally independent from X_j given all other variables ($\mathbf{X_{rest}}$) which is equivalent to the element k_{ij} in Σ^{-1}=0.0. This relation is used to define Gaussian graphical models (GGMs) (Lauritzen 1996; Edwards 2000) and the edges of GGMs are interpreted as non-zero partial correlations. The problem with GGMs is that they only estimate the full conditional relationships accurately when the number of samples exceeds the number of variables. However, the case in gene expression profiles, for example, is that the number of genes (variables) exceeds the number of samples, $p \gg n$. Therefore, the correlation matrix Σ does not have a full rank and hence cannot be inverted (Schäfer & Strimmer 2005a). Different studies suggest ways to estimate GGMs in a $p \gg n$ situation; for example, (Schäfer & Strimmer 2005b) suggests a linear shrinkage regularisation method.

2.3.2 Dependency networks

Markov networks, introduced in the previous section, are related to a set of graphical models named *dependency networks* (DNs) (Heckerman et al. 2000). DNs (Figure 2.7) are constructed by mapping a variable Y to its parents $\mathbf{X}$.

The subset $(X_{x_1,x_2,..x_i}) \in \mathbf{X}$ that predict Y will be connected to Y by a directed edge. Dependency networks are used because of their computational advantage over Markov networks and Bayesian networks when the graph is learnt from data. However, dependency networks are not useful representations of causal relationships (Heckerman et al. 2000) and also do not define a joint probability distribution due to the cyclicity in dependency graphs. Several different methods are proposed in the literature to learn the structure of dependency networks from data. One example is to estimate the DNs using linear regression with penalised coefficients (Meinshausen & Buhlmann 2006). Ridge regression and lasso are two examples of such penalised methods.

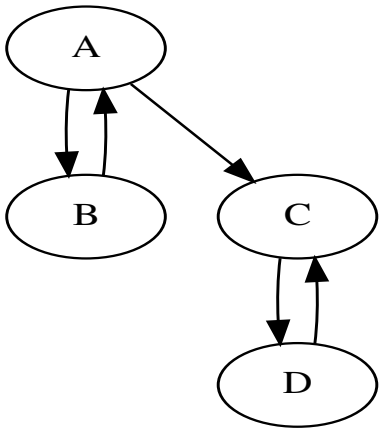


Figure 2.7: an example of Dependency Network.

2.3.3 Bayesian networks

A Bayesian network (Figure 2.8) is a probabilistic model where the structure is a directed acyclic graph (DAG) that encodes the dependences between a set of random variables. The individual random variables are nodes of a DAG, which explains the dependency structure.

Each node in the graph is explained by a local probability distribution (LPD) and over all the nodes the joint distribution $p(x)$ can be defined as follows:

$$p(x) = \prod_{v \in V} p(x_v | x_{pa(v),\theta_v}) \tag{2.2}$$

Where θ_v denotes the parametrisation of the LPD and $x_{pa(v)}$ is the set of parent states. The DAG structure implies an ordering of the variables. The parents of each node are those nodes that make the child node independent of all non-descendant nodes in the graph representation. The factorisation of the joint distribution in (2.2) is a property of Bayesian networks, which allows us to decompose the graph into a set of families. Therefore, when it comes to learning a Bayesian network from data, it is possible to decompose the structure learning for each family individually if *an ordering is given between variables*. Thus, for each variable we seek the best predictor parents separately and then join them to form a directed acyclic graph. For example, to learn a Bayesian network from gene expression data, we can take each gene and search for

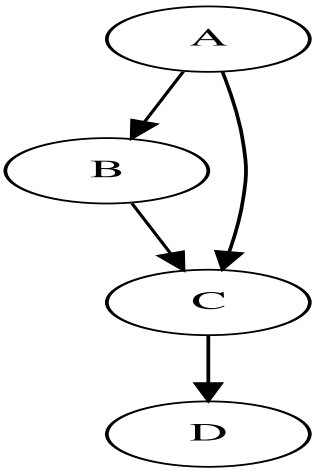


Figure 2.8: An example of a Bayesian network

the best parents for it in a decomposed manner if a prior knowledge is given about the possible causal parents for this gene, finally joining all the sub graphs for each family together to give a full picture of a cellular graph. However, the interpretation of a Bayesian network when learning a graph from gene expression data is always important.
Learning a causal acyclic network is similar to learning a Bayesian network. Causal networks can be interpreted as Bayesian networks when the *Causal Markov assumption* holds. Thus, given the values of a variable's immediate causes, it is independent of its earlier causes. When the causal Markov assumption holds, the causal network satisfies the Markov independencies of the corresponding Bayesian network (Friedman et al. 2000). The interpretation of causality from Bayesian networks has received a great deal of treatment in the literature (Heckerman et al. 1997; Pearl & Verma 1991; Spirtes et al. 2000). Another well known method for inferring causality networks is *intervention*, which uses some methods which force genes to be 'knocked out'. Another way of making causal graphs *but using observations only* is by using causal prior knowledge to guide the learning algorithm when learning the parents for each gene. This also implies that if *causal* background knowledge is used to learn a Bayesian network, then it will also hold that the resultant graph can be interpreted either as a causal or Bayesian network.

It is also worth knowing that Markov networks and Bayesian networks represent different sets of independences between variables. For example if we have a Bayesian network such as: $A \rightarrow B \leftarrow C$, this graphical representation implies that A is independent from C, $A \perp C$.

However, this sort of conditional independence relationship cannot be expressed using a Markov network.

2.3.4 Co-expression networks

Co-expression networks are one of the simplest representations of variable relationships used in gene co-expression relationships. The idea is based on the following: if two variables show similar behaviour, for example, expression values in the gene expression profiles, they are supposed to follow the same regime. Co-expression networks (also called relevance networks) are worked out by computing a similarity score for each pair of variables. If the similarity is above a certain threshold, the two variables are connected to each other in the graph, otherwise they remain unconnected.

The similarity between two variables can be calculated in different ways. One example is correlation coefficients (r) (2.3). Given that the data is drawn from a multivariate normal distribution, zero correlation between two variables corresponds to statistical independence and an unconnected edge in the graph between the two variables. Correlation networks can be easily interpreted and accurately estimated even in situations with *large variables, small sample size* (Markowetz & Spang 2007). Measuring the correlation between two variables using (2.3) will only concern the linear relationship of independence between any pair variables. Other flexible similarity measures, like mutual information, can be used as a non-linear measure of independency (Butte et al. 2000). Co-expression measurements produce undirected graphs, unless certain prior knowledge, such as the KEGG database, is used for directionality.

$$\mathrm{r} = \frac{\sum_{i=1}^{n}(Y_i - \bar{Y})(X_i - \bar{X})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}} \tag{2.3}$$

2.4 Machine Learning of Graphical Models

The first part of this section discusses how a graph is learned and how the parameters for the learned graph are estimated. Then, it dicusses the inference in machine learning of graphical models. The second part covers related work for learning graphical models from gene expression microarray data and briefly about the diagnosis of cancer using supervised learning algorithms.

2.4.1 Learning graphical models

To learn a graphical model, two parts have to be taken into account: firstly, learning the structure of the graph, and secondly, learning the parameters for the graph. There are different ways to learn the topology of a graph for a given problem. One way is from domain knowledge. In this method, the variables that are needed in the domain problem have to be determined. The rela-

tionships between these variables also have to be designated, and the last step is then to induce the parameters for the constructed network. However, it is not always possible to construct a graphical model from domain knowledge, for example, constructing a causal graph from observing the behaviour of genes in the lab is expensive. Another way to learn a graph is from data. To do this, a structure for the graph is needed and also a complete dataset(it is possible to learn from incomplete data but is harder)to estimate the parameters for the structure. However, most real-world problems have unknown structures and sometimes suffer from incomplete datasets. The following section shows how a Bayesian network is learned from data, assuming a complete dataset, and then how the parameters are inferred for the learned structure.

2.4.1.1 Learning Bayesian networks from data

When data is used to learn Bayesian networks, firstly we need to learn the structure of the Bayesian network if it is unknown. Secondly, we learn the parameters for this network, using conditional probability tables (CPTs). However, learning the structure of Bayesian network is an NP-hard problem. To illustrate this, Table 2.1 shows how difficult it is to search all possible Bayesian networks for different sets of variables, as the number of graphs grows exponentially with the number of variables. Even with a small number of variables, there are many possible Bayesian networks, Directed Acyclic Graphs (DAGs), in the search space and searching all DAGs to find the best graph that fits the data is difficult.

Table 2.1: The corresponding DAGs for each set of variables

Number of variables in DAG	Number of possible DAGs
1	1
2	3
3	25
4	546
5	29281
6	3781503
7	1.138.779.265
8	78.370.2329.343
9	1.213.442.454.842.881
10	4.175.098.976.430.598.100

Chickering et al (Chickering et al. 2004; Chickering 1996) shows that finding the highest-scoring Bayesian network is NP-hard, regardless of the size of the data, when a consistent scoring criterion ,such as BDe score function, that favours a model with fewer model parameters is used. Therefore, learning Bayesian networks from data is largely based on heuristics or *moderately* greedy search algorithms. However, if a combination of data and domain knowledge is possible, complete search algorithms can be used, as the prior knowledge will shrink the search space to an admissible search , constrained by data and prior knowledge, such as that from KEGG, as we

will show.

There are many algorithms designed for searching the most probable structure for a network and in this section we will discuss some of them. In general, there are two main approaches for learning Bayesian networks from data, *search and score algorithms* and *constraint-based algorithms*. Search and score algorithms attempt to identify the networks that maximise/minimise a score function, which expresses how well a network fits the data. The K2 algorithm and Genetic algorithms are two greedy search examples of such methods of learning (Cooper & Herskovits 1991; Larranaga et al. 1996). Constraint-based algorithms start by assuming that all variables in the network are dependent on each other. Then, an estimation is taken from the data of whether certain conditional independencies between the variables exist (Spirtes et al. 2000). The PC algorithm is an example of the constraint-based approach.

K2 algorithm: The K2 algorithm (Cooper & Herskovits 1991) assumes that an ordering of the variables in the dataset exists. When it searches the possible parent nodes πs for a node x_i it considers only those parents that come before x_i in the dataset. The best parents for each x_i are subject to maximising the following score function:

$$\mathrm{f}(i, \pi_i) = \Pi_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_i j + r_i - 1)!} \Pi_{k=1}^{r_i} \alpha_{ijk}! \tag{2.4}$$

where:
π_i is the set of parents of node x_i.
q_i = $|\phi_i|$.
ϕ_i is the list of all possible instantiations of the parents of x_i in dataset D.
r_i = $|V_i|$.
V_i is the list of all possible values of the attribute x_i.
α_{ijk} is the number of cases (i.e. instances) in D, in which the attribute x_i is instantiated with its k_{th} value, and the parents of x_i in π_i are instantiated with the j_{th} instantiation in ϕ_i.
$N_{ij} = \sum_{k=1}^{r_i} \alpha(ijk)$. That is, the number of instances in the database in which the parents of x_i in π_i are instantiated with the j^{th} instantiation in ϕ_i.

K2 heuristically searches for the most probable Bayesian network from a given dataset of variables. This algorithm starts with a node without parents and then adds each parent that is likely to increase the probability of a structure being the correct structure. The K2 algorithm adds new parents, until adding a new parent does not increase the score function. (Ferrazzi et al. 2007) used the K2 algorithm to learn dynamic Bayesian networks from gene expression data and provided a tool that we will discuss and use in the thesis.

The PC algorithm Another approach to learning a Bayesian network is using *constraint-based algorithms*. One example is the PC algorithm (Spirtes et al. 2000) which starts by connecting all variables to each other. In other words, it assumes that all variables are dependent on each other and then verifies the conditional independencies for each pair, for all possible orders:

N=0: X_1 is independent from X_2, $X_1 \perp X_2$.
N=1: X_1 is independent from X_2, given X_3, $X_1 \perp X_2|X_3$.
N=2: X_1 is independent from X_2, given X_3 and X_4, $X_1 \perp X_2|X_3, X_4$.
N=3: ...

The PC algorithm follows the verification by using a statistical test. It starts with a given undirected graph. The null hypothesis is then made (H_0) that X_1 is independent from X_2, $X_1 \perp X_2$. After that a statistical test is used to show whether H_0 is rejected, and X_1 and X_2 are dependent (the alternative hypothesis H_1). Chi-square (χ^2) and the degree of freedom (df) are used for testing the independencies in PC.

2.4.2 Parameter estimation

The second part after learning the structure of a graph, using for example Bayesian networks, is to learn the parameters of such graphs. One example of a parameter estimation method is the maximum likelihood estimator (MLE), a *frequentist estimator*, which is considered to be a non-Bayesian estimator. The MLE estimator is used to obtain the conditional probability table (CPT) for each node in the graph. In this section, an example will be given to show how the parameters can be estimated from data after the graph has been learned. For a simple binary Bayesian network structure ($X \longrightarrow Y$) the parameterisation for this graph consists of the following parameters:

θ_{x_0}, θ_{x_1} specify the probability of two values of X.
$\theta y_0|x_1$, $\theta y_1|x_1$ specify the probabilities of Y, given X= x_1,($Y|X = x_1$).
$\theta y_0|x_0$, $\theta y_1|x_0$ specify the probabilities of Y, given X= x_0,($Y|X = x_0$).

Since the sample data has two variables, then each example can be given as $< x[m], y[m] >$, where m is the sample size. As a result, the likelihood function is:

$$L(\theta : D) = \Pi_m P(x[m], y[m] : \theta) \tag{2.5}$$

For the simple Bayesian network here the equation can be written as:

$$L(\theta : D) = \Pi_m P(x[m] : \theta) P(y[m]|x[m] : \theta) \tag{2.6}$$
$$= \Pi_m P(x[m] : \theta)) (\Pi_m P(y[m]|x[m] : \theta))$$

The first term from (2.6) can be calculated straight from the data since it does not depend on any other variables. The second term in 2.6, however, will be decomposed further as follows:

$$\Pi_{m:x[m]=x^0} P(y[m]|x[m] : \theta_{Y|x^0}), \Pi_{m:x[m]=x^1} P(y[m]|x[m] : \theta_{Y|x^1}) \tag{2.7}$$

where Y=y^0, y^1

Thus, to maximise, for example the parameter $\theta_{Y_{=y^1}|x^0}$, we say that the maximum likelihood for this parameter is:

$$\theta_{y^1|x^0} = \frac{M[x^0, y^1]}{M[x^0, y^1] + M[x^0, y^0]}$$
$$= \frac{M[x^0, y^1]}{M[x^0]} \tag{2.8}$$

Where $M[x^i, y^i]$ denotes how many times this example been encountered in the dataset.

In general, to learn the parameters for a Bayesian network with structure *G* and parameters θs, given a dataset consist of samples $D_1, D_2, ..., D_m$ the maximum likelihood function is given as follows:

$$L(\theta : D) = \Pi_m P_G(D[m] : \theta)$$

$$L(\theta : D) = \Pi_m \Pi_i P(x_i[m]|\mathbf{pa_i}[m] : \theta)$$

$$L(\theta : D) = \Pi_i \Pi_m P(x_i[m]|\mathbf{pa_i}[m] : \theta) \tag{2.9}$$

Another way of estimating the parameters for a learned graph is by using a Bayesian approach that gives prior probability distribution over θs, before observing the data. Thus, the parameters are considered as random variables and we use Bayes theorem (2.10) to update θs to get the posterior probability distribution for each θ:

$$P(\theta|d) = \frac{P(d|\theta)P(\theta)}{P(d)} \tag{2.10}$$

2.4.3 Inference

After learning the structure and estimating the parameters of a graph, it can be used to make inferences. As an example , if we have a causal network that represents the relationship between flu and high temperature as $Flu \rightarrow highTemp$ and each variable in this graph has two values, either True or False, then we can assign conditional probability distribution (CPT) to this graph; thus, we have a prior probability for *Flu* and CPT for *highTemp*. So, for a new patient, based on their temperature, we can infer whether they have flu or not.

This kind of inference is called *exact inference* as it is simple to use Bayes theorem to find the probability of having flu, given the patient's temperature. The most common exact inference methods are: variable elimination; clique tree propagation; and recursive conditioning. However, in most cases doing exact inference is computationally complex and known to be an NP-hard problem (Cooper 1990). Therefore, an approximate inference is used instead. The most common approximate inference algorithms are: stochastic MCMC simulation; generalised belief propagation; and variational methods.

2.5 Related work

The field of machine learning is not new in cancer research. Many algorithms have been proposed in cancer detection and diagnosis in the last 20 years(Joseph A. Cruz 2006). Machine learning algorithms are used in detecting and classifying tumours via X-ray and CRT images, and in the classification of malignancies from proteomic and genomic data, using Affymetrix microarrays. The latest PubMed statistics show that more than 1,500 papers have been published on the subject of machine learning and cancer. The majority of these papers are concerned with how machine learning algorithms are used to identify, classify, detect or distinguish tumours and other malignancies. A few papers are concerned with using machine learning to find the probability of developing cancer before its occurrence (*susceptibility*). Nearly half of the papers that have been published, are on predicting the likelihood of redeveloping cancer after removing it(*recurrence*). About 43% of the papers have been about predicting life expectancy and tumour-drug sensitivity after the diagnosis of the disease (Joseph A. Cruz 2006). Since breast cancer and prostate cancer are the most frequent diseases to occur, machine learning algorithms have been applied extensively to these diseases. Figure 2.9 shows that the strongest preference has been made for using

machine learning in breast cancer and then in prostate cancer.

Artificial neural networks (ANN), decision trees (DT), graphical models and support vector machines (SVMs) are the most commonly algorithms being used in cancer. ANN and DT for instance, have been used to classify benign tumours from malignant tumours. Bayesian network learning algorithms have attracted huge attention for constructing gene regulatory networks. Research has also been conducted to compare three different algorithms, ANN, C4.5 and logistic regression (Delen et al. 2005). In this study, the three algorithms were to predict the survivability rates of breast cancer patients. The results indicate that the decision tree (C4.5) has the best predictor with 93.6% accuracy. SVMs have also been used for the diagnosis and prognosis of breast cancer and (Zafiropoulos et al. 2006) shows that SVM algorithms have achieved high values of accuracy (96.61%).

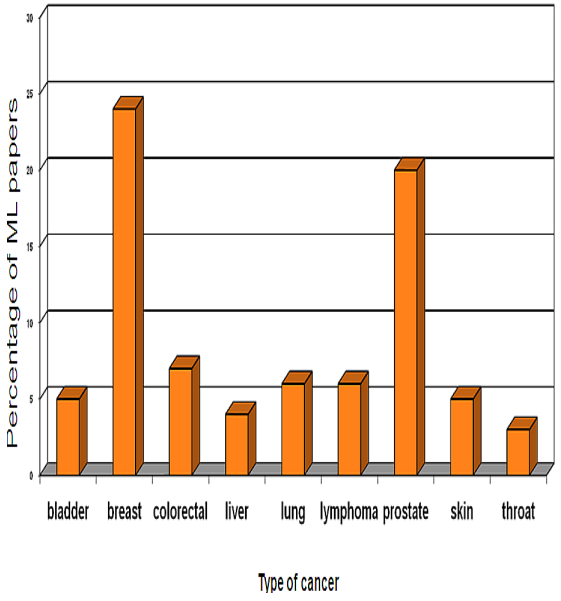


Figure 2.9: The percentage of machine learning applied in different types of cancer(Joseph A. Cruz 2006).

Moreover, graphical models learning algorithms are being used in detecting and diagnosing cancer from genomic datasets. Generally, using graphical models to learn different kinds of networks from gene expression data are referred to as inferring cellular networks. One simple approach to learn a regulatory network is to find the genes that are statistically correlated to each other in a pair-wise approach. If two genes are correlated then they influence each other. The set of graphical models is called *co-expression networks* (Wolfe et al. 2005). In a biological sense, if two genes show similar gene expression profiles, then they hint at a co-regulation relationships (Markowetz & Spang 2007). However, research has emphasised that in gene expression

profiles, where the number of genes exceeds the number of samples, $p \gg n$, it is important to find the significance of the amount of correlation, as many pairs of genes show similar behaviour in expression profiles by chance, even though they are not biologically related (Markowetz & Spang 2007).

Another set of graphical models is Bayesian networks, which give a useful representation of probability in a compact way. There are different methods to learn Bayesian networks from data. Approaches to learning Bayesian networks from gene expression data include: linear relationship learning algorithms (D'Haeseleer et al. 1999), which are referred to as learning Gaussian graphical models (Geiger & Heckerman 1994a); and non-linear relationships learning algorithms (Weaver et al. 1999). Boolean networks also a popular set of graphical models which are simple deterministic models of regulatory networks that are defined by a directed, possibly cyclic, graph. Each gene in the Boolean networks has a Boolean function, which maps the relationship between the gene and its parents. Causal acyclic graphs have also provided great help in understanding the causality between large numbers of genes. Learning a causal acyclic network is similar to learning a Bayesian network, given that causal Markov assumption holds. Thus, given the values of a variable's immediate causes, it is independent of its earlier causes.

The interpretation of causality from Bayesian networks has received a great deal of treatment in the literature (Heckerman et al. 1997; Pearl & Verma 1991; Spirtes et al. 2000). *Intervention* that makes some enforced methods, for example, gene knockout is a well known method for inferring causality networks. Another way of making causal graphs from observations is by only using causal prior knowledge to learn the parents for each gene. (Murphy & Mian 1999) shows how regulatory networks can be inferred using dynamic Bayesian networks. However, learning such networks requires time series measurements for genes at different times. (Murphy & Mian 1999) shows that most of the proposed discrete time graphical models that include Boolean networks, both linear models and non-linear models, are all special cases of a general class of models called dynamic Bayesian networks. Another set of graphical models is dependency networks (Heckerman et al. 2000). The graph of a dependency network, unlike a static Bayesian network, is potentially cyclic. When two genes in a graph are found to be good predictors of each other, then the dependency networks fit well (Aloraini et al. 2010).

CHAPTER 3

Biology of Cancer

This chapter presents a discussion about cancer in terms of how it occurs and how it is treated. It focuses on two very common types of cancer, breast cancer and prostate cancer. Section 3.2.1 gives an explanation of what breast cancer is, how it occurs and how it is treated. Section 3.2.2 details prostate cancer, how it occurs and the treatment. The chapter concludes with a detailed section about cell communication and how the treatment of cancer can be improved using signalling pathways.

3.1 What is a cancer

Cancer is a disease that is a result of uncontrolled cell growth. When a cell grows/divides uncontrollably it stops to respond to the normal signals that control the cell growth (Adami et al. 2002). Environmental factors play an essential role in cancer development but also cancer might occur due to heredity (Anand et al. 2008). The beginning of cancer, inside the cell, is dependent most commonly on a genetic mutation that occurs in the DNA (Deoxyribonucleic acid) (Figure 4.2) inside the cell. DNA mutation can happen when even a single nucleotide changes in the DNA. The genetic sequence change leads to production of a mutant protein. However, more commonly, a normal cell transforms to a cancerous one when several mutations happen inside the DNA in the cell. These mutations can disrupt the cell's growth, which in turn leads to the development of a tumour mass(`www.insidecancer.org`). Different types of cancer can occur in different parts of the body, including: breast cancer, prostate cancer, brain cancer, lung cancer and skin cancer. The cancer can be solid tumours, in which lumps are formed or liquid tumours such as leukaemia.

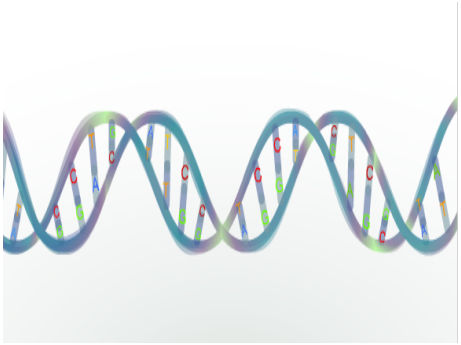


Figure 3.1: A DNA strand with the four nucleotides (lettered A, T, C,and G).

3.2 Overview of Two Types of Cancer

3.2.1 Breast cancer

Breast cancer is an uncontrolled growth of breast cells which can spread to different parts of the body. Despite the fact that breast cancer occurs when a mutation in the DNA happens, known as *genetic sequence disruption*, only 5-10% of breast cancer is inherited, while 90% of cases are known to be from environmental factors and genetic processes inside the body(`www.breastcancer.org`). Figure 3.2 shows the structure of the breast.

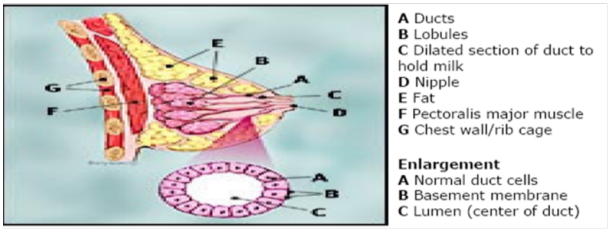


Figure 3.2: The structure of the breast inside the body(`www.breastcancer.org`).

The breast is a gland designed to make milk, which then goes through ducts to the teat. The cells in the breasts normally grow and rest in cycles, with genes inside the breast cells responsible for controlling and managing the growth of cells. The nucleus in a cell keeps cell growth under control, which causes genes to work *normally*. However, when genes develop an abnormality, they sometimes lose their ability to control the cycle of cell growth and rest. The exact causes of breast cancer are not yet known, but certain environmental risk factors have been defined(`www.breastcancer.org`):

- Age: getting older, 80% of breast cancer cases occur in post-menopausal women.

- Nulliparity: having no children or having children late in life.
- Prior history: a significant family history of breast cancer.
- Alcohol: use of alcohol is clearly linked to a slightly increased risk of getting breast cancer.
- Early menarche: menarche before age 12 is a risk factor for certain types of cancers in women, including breast cancer.
- Being overweight (especially after the menopause).

Generally, breast cancer can be considered in two main categories: *benign* cancer, sometimes called non-invasive cancer and *malignant* cancer (invasive cancer). Non-invasive cancer grows and divides abnormally inside the breast; but is only in the milk ducts in the breast (Figure 3.2) and does not spread into the surrounding breast tissue or to other parts of the body. Sometimes this is called a *pre-cancerous condition*. However, with invasive cancer, the cancerous cells are no longer confined to the breast ducts and lobules. They spread to the surrounding breast tissue and have the potential to spread to other parts of the body. Several statistical studies have reported that breast cancer is the most prevalent cancer type in many areas around the world (Parkin et al. 2000). The basic kinds of tests conducted on breast cancer patients can be defined as follows:

1. Mammogram: an x-ray examination, designed to detect breast cancer at an early stage.
2. Ultrasound scan: high-frequency sound waves are used to outline the suspicious areas of cancer.
3. Fine needle aspiration (FNA): a quick and simple procedure which is done in the outpatient clinic. Using a fine needle and syringe, the doctor can diagnose cancer cells by taking a sample of cells from the breast lump using FNA.

3.2.2 Prostate cancer

The prostate is a small gland that sits under the bladder and in front of the rectum(Figure 3.3) and is about the size of a walnut. The tube that runs through the penis that carries both urine and semen out of the body also runs directly through the prostate(`www.prostatecancerfoundation.org`).

Despite great efforts from doctors, physicians and biologists to treat prostate cancer, there is still a large chance that it will disrupt the operation of urinary, bowel and sexual functions. For example, one solution for prostate cancer is to remove the prostate from the body. As a result, the bladder is pulled downward and connects to the urethra at the point where the prostate used to sit. If the sphincter at the base of the bladder is damaged during this process or even if it is damaged during radiation therapy, some leakage of urine might start to occur. Moreover, there might be an inability to control the bladder and bowels.

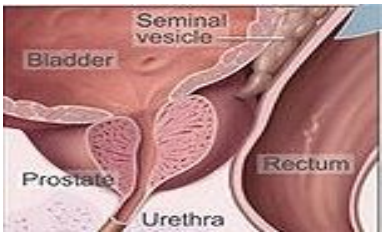


Figure 3.3: The position of prostate cancer.

Prostate cancer is the most common cancer in men. It is highly associated with different risk factors(`www.prostatecancerfoundation.org`) such as:

- Age: although only one out of 10,000 under the age of 40 will be diagnosed, with increasing age the rate becomes much higher, up to 1 in 38 men aged 40 to 59 and 1 in 15 of those aged 60 to 69 will be diagnosed.

- Race and family history: these are some of the most important factors. For example, in America, African American men are 61% more likely to develop prostate cancer compared with people of pale skin. Also, a man with a first degree relative such as a father, brother or son with a history of prostate cancer is twice as likely to develop the disease.

The risk factors mentioned above for breast and prostate cancers are used in many hospitals and labs to diagnose cancer. However, for the purpose of diagnosis, clinical factors alone are not always helpful. This is why doctors and biologists have started to pay much more attention to using other data as well as clinical data.

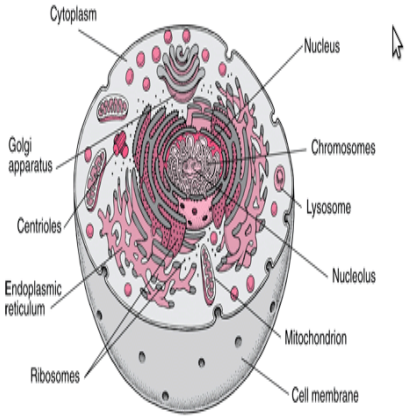


Figure 3.4: The structure of a cell inside the body.

Using signalling pathways with clinical factors for cancer treatment has emerged in the last decade. The cancer is diagnosed from the genomic point of view, based on tracking the communication of cells and genes inside each cell. Cells work together using chemical signals and therefore the signals are the core of growth, division and death for each cell. A disruption in cellular communication is a strong sign of cancer. Usually, biologists and doctors track the failure of normal signals in cancer cells by comparing them to cell signalling pathways in the normal cellular system. In the general case, the signals are transferred from outside the affected cell into its cytoplasm and then to the cell nucleus (Figure 3.4). In the next section a more detailed discussion will be given on cell communication. The discussion of cell communication in the next section is largely based on (Campbell & B.Reece 2005), and (Alberts et al. 2002).

3.3 Cell Communication

Cell-to-cell communication is essential for multicellular organisms. The cell inside the body must communicate to grow ,divide and reproduce during cell life. Understanding cell signalling helps greatly to answer important questions in biology and medicine , such as the development of cancer. The signals received by the cell either comes from another cell or from some changes in the organisms' physical surroundings. The communication between cells most often occur by chemical signals and therefore heavily depend on extracellular signals molecules which are produced by the cells. Cell-to-cell communication happens by releasing chemical signals/messages that are targeted for neighbors cells or further away cells. When the chemical messages target neighbors cells such as molecules they are called local regulators. In this section, a discussion about the main mechanisms of how cells detect, process and respond to the chemical signals will be given.

3.3.1 The stages of cell signalling

When a cell receives a signal from another cell , three stages are taken into consideration: *signal reception, signal transduction, and cellular response*. When the reception occurs at the plasma membrane(Figure 3.4), the signal transduction usually a pathway of several steps and each molecule in the pathway makes change in the next step. Finally, the last molecule in the pathway causes the cell's response.

3.3.1.1 Signal reception

The first stage of receiving a signal in the cell , a signal molecule binds to a receptor protein which causes the protein to change its shape. The targeted cell by a specific signal has molecules of a receptor protein that recognises that signal molecule. Basically, the signal molecule acts as a ligand which is a small molecule binds to a larger molecule. The process of ligand binding causes a receptor protein to change its shape. The result of this shape change will activate the receptor so that it can interact with another cellular molecule. Moreover, the ligand process might lead two or more receptor molecules to aggregate together. Most of the signal receptors are proteins

that are located in the plasma membrane.

3.3.1.2 Signal-transduction pathways

In this stage of cell signalling , multisteps pathway can occur. One major benefit of such pathway is signal amplification since if some of the molecules in a pathway transmit the signal to multiple molecules, a large number of activated molecules can happen. Signal pathways receive and send (relay) signals from receptors to cellular response in which the signal activated receptor activates another protein, which in turns activates another molecule until the protein that produces the final cellular response is activated. The molecules that receive and pass a signal from the receptor to the response mostly are proteins.

In signal transduction pathways, important protein activation/inactivation processes can happen. One important regulation in cells is protein (de)phosphorylation which is a major mechanism of signal transduction. A signal pathway starts with receiving a signal molecule that binds to a membrane receptor protein. Then, the receptor activates a relay molecule which in turn activates a protein kinase. Protein kinases are enzymes that alter other proteins by adding phosphate group to them(phosphorylation). After that, activate protein kinase transfers a phosphate from a nucleoside to active another protein kinase molecule(nucleoside is glycosylamines consisting of a nucleobase). The activation of the second protein kinase causes the phosphoralation and the activation of a third kinase. Finally, the activation of the third protein kinase phosphorylates a protein that stimulates the cell final response to the signal. The dephosphorylation of all protein kinases can be made by the removal of the phosphoate group by enzymes called phosphatases.

Another important mechanism related to protein molecules during signal transduction is protein degradation. Protein molecules are continuously synthesised and degraded in all living organism. The concentration of individual cellular proteins is determined by a trade-off between the rate of synthesis and degradation. This will lead to loss of proteins from cells(atrophy) or increase in protein content of cells(hypertrophy). In fact, degradation rates of proteins are essential to their cellular concentrations.

the transcription of genes also can be seen as a mechanism of protein activation/inactivation. The amount of gene expression released by a gene can be a sign for the amount of produced protein. Genes essentially are made up of DNAs which act as structural blocks to produce proteins. In a signalling cell life , genes are transcribed to Ribonucleic acid(RNA), and then RNA is translated to protein. Often , measuring the amount of activated protein is difficult and therefore, measuring gene expressions at the RNA level(downstream/upstream) is used to detect the activation/inactivation of proteins.

3.3.1.3 Cellular response to signals

The signal-transduction pathway stage has an ultimate goal which is the regulation of one or more cellular activates. This signalling pathway regulation will lead to active/inactive specific proteins by turning specific genes on or off.

3.4 Wnt signalling pathway

After we have given a broad discussion about cell communication and the various protein activation/inactivation processes that happen during cell signalling communication, this section will focus on an important pathway known to be involved in many biological processes, namely Wnt signalling pathway(Figure 1.4,page 21). Wnt signalling pathway has a central role and inappropriate activation of this pathway are observed in several human cancer (Spink et al. 2000). Nevertheless, many of the mechanisms involved in activation/inactivation of this pathway still unclear (Thorstensen & Lothe 2003).

In the presence of Wnt ligand(wnt signalling) , Wnt ligand binds a Frizzled(FZD) proteins and a co-receptor protein related to the low density lipoprotein receptor related protein(LRP). This in turn will activate the cytoplasmic protein dishevelled(Dsh/Dvl). Presicly how dishevelled protein is activated is not fully understood but phosphorylation by casein kinase1(CK1) and casein Kinase2(CK2) have been suggested to be partly responsible (Willert et al. 1997; Sakanaka et al. 1999; Amit et al. 2002). The activation of Dsh/Dvl will lead to the inhibition of β-catenin phosphorylation and degradation. This mechanism is not fully understood but it requires Dsh/Dvl and other signalling proteins(Axin, adenomatous polyposis coli (APC), and glycogen synthase kinase (GSK)-3β) that bind to Dsh/Dvl (Thorstensen & Lothe 2003; Alberts et al. 2002). Dsh/Dvl is suggested to bind CK1 and thereby inhibiting priming of β-catenin and this indirectly preventing GSK-3β phosphorylation of β-catenin (Amit et al. 2002). The increase in undegraded β-catenin caused by wnt signalling allows β-catenin to enter the nucleus and binds to the members of the T-cell factor(Tcf)/lymphoid enhancing factor(Lef) family(TcF/Lef1). At this stage, β-catenin acts as a co-activator which induces the transcription of the WNT target genes.

Other genes activated by β-catenin is c-myc that encodes a protein(c-Myc) which is a prime factor of cell growth and proliferation. However, if a mutation occurred to a protein called APC that binds to Dsh/Dvl via Wnt-signalling , this will inhibit the ability of c-myc proteins to bind β-catenin. Therefore,β-catenin accumulates in the nucleus and stimulates the transcription of c-myc and other target genes. This stimulation can also be observed in the absence of Wnt signalling. One major cause of this stimulation is uncontrolled cell proliferation that promotes the development of cancer.

3.5 Summary

A growing understanding of the complex signalling pathways that underlie tumour formation and progression is aiding the development of a new generation of anti-cancer drugs, targeted at specific molecular events. For example, the Wnt signalling pathway is known to participate in prostate cancer development (Birnie et al. 2008) and studying this pathway in both normal and abnormal behaviour will potentially enhance our understanding, and allowing the development of more effective drugs. Recently, databases have been created to present the latest discoveries in new interactions between pathways and the families of genes in each pathway. One well known database is KEGG (Kanehisa & Goto 2000; Kanehisa et al. 2010), a collection of manually drawn pathway maps, representing the latest knowledge on the molecular interaction and reaction networks in cancer and other diseases. Using a computerised knowledge-base makes it possible to track the latest discoveries in molecular networks and unify effort towards a better understanding of cancer from a molecular level, in addition to the clinical/environmental factors.

The success of KEGG and other projects such as the Human Genome Project have resulted in the discovery of many genes that are associated with certain diseases such as cancer. However, our understanding of molecular mechanisms is still incomplete for cancer, which is a combination of various genetic and environmental factors (Kanehisa & Goto 2000). Therefore, the analysis of cancer signalling pathways, and in particular the genes involved in these pathways, will better clarify the molecular mechanisms of cancer and help to develop new drugs and treatments in the future.

CHAPTER 4

Microarray technology and gene expression profiles data analysis

This chapter presents a discussion about microarray technology and gene expression profile data analysis. Section 4.1 begins by showing how messenger RNA (mRNA) is used to measure gene expression from Deoxyribonucleic acid (DNA). It then highlights the main throughput DNA-microarrays that are widely used to measure gene expressions. Section 4.3 discusses the Affymetrix single microarray platform in detail. It shows how the pre-processing steps are conducted to generate gene expression profiles from a single microarray (Section 4.4). In Section 4.4.2, the pre-processing steps for the prostate cancer datasets that are used in this work are given. In Section 4.4.3 and its subsections, refined pre-processing steps for the genes included in the Wnt signalling pathway are presented. Section 4.4.4 introduces the gene expression colon cancer datasets pre-processing steps that are generated from the Illumina microarray platform.

4.1 Introduction

Almost every cell in the body of an organism has the same DNA. Genes essentially are made up of DNAs that act as instructions to make molecules called proteins. Genes are expressed using two steps, first they are transcribed into RNA (Ribonucleic acid) and then the RNA is translated into the corresponding protein, Figure 4.1. There are three types of RNA: messenger RNA (mRNA), transfer RNA (tRNA) and ribosomal RNA (rRNA). Biologically, RNA is an important type of molecule, consisting of a long chain of nucleotide units. All DNA microarray platforms are being developed based on transcribed step RNA and in particular on *messenger RNA* (mRNA) as it is the most important type of RNA. The name messenger RNA suggests that it carries the

information encoded in DNA to the translation step protein, as each DNA becomes a protein.

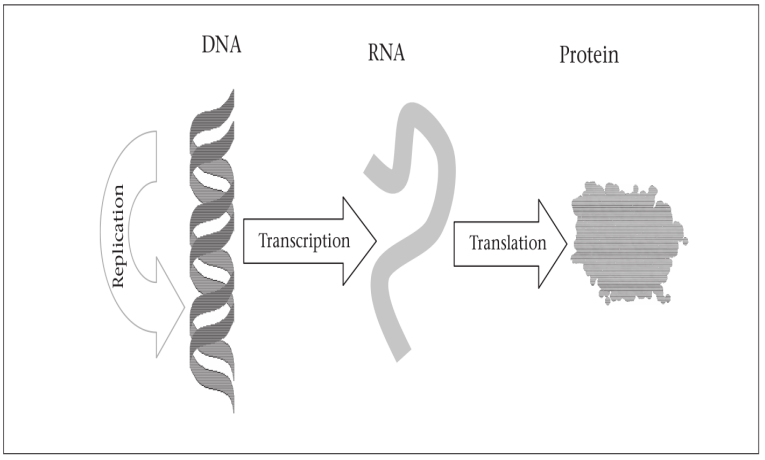


Figure 4.1: DNA is transcribed to mRNA and then translated to protein

4.2 DNA-microarrays

Deoxyribonucleic acid(DNA) is described as a nucleic acid that contains the genetic instructions for living organisms. It is a double helix(Figure 4.2) and each side helix are formed by a backbone sugar and phosphate molecule. The two helix are connected by four nucleotide bases joined weakly in the middle by hydrogen bonds. Thus, Adenine base(A) is bound with Thymine(T), whereas Guanine base(G) is bound with Cytosine(C). Therefore, the strands of the helix are complement to each other. All DNA-microarrays are built using this chemical fact of complementarity. DNA-microarrays(Figure 4.3) consist of many single strands of DNA attached to their surface that are known as probes. When the complementary strands for the probes on the surface of microarray are spread on the surface, the *hybridization* process happens which is the result of sticking each strand to its complementary on the surface of microarray. Microarrays measure the amount of hybridization at the mRNA level which is the transcriptional step before a gene translated to a protein, Figure 4.1. mRNAs carry the DNA's genetic message to the cytoplasm of a cell where proteins are made. A strand of mRNA is similar to a strand of DNA except that DNA has A,T,C,G nucleotide bases but RNA instead of Thymine(T) it has uracil (U). Therefore, the purpose of microarrays is to measure for each gene in the genome the amount of message that is carried by mRNA and if mRNA can find its complementary in the array surface then it binds naturally and sticks to a particular spot in the array.

Currently, different DNA-microarrays are used to measure thousands of genes in one go. Affymetrix one-channel microarrays, two-channel microarrays and Illumina microarrays are examples of microarray technologies currently used in many labs. In this work, the focus is on one-channel microarrays and Illumina microarrays. The comparison between different gene ex-

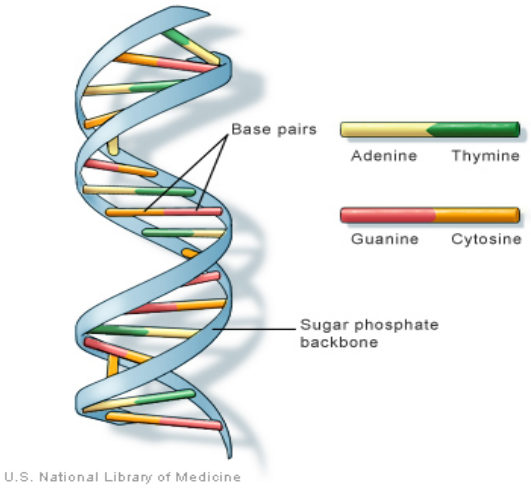


Figure 4.2: DNA is a double helix formed by base pairs attached to a sugar-phosphate backbone.

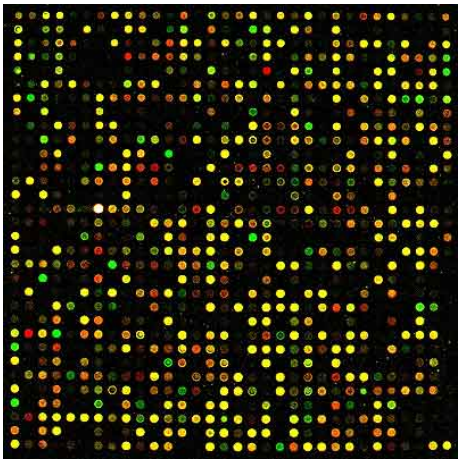


Figure 4.3: DNA-microarray

pression profiles generated from all microarray technologies are based on the amount of mRNA shown from each gene in different conditions. For example, the expression level of a gene in cancer tissue is compared to the expression level for the same gene in a non-cancer tissue.

Although DNA-microarrays are widley used in many labs for new biological discoveries , they have their limitation. The limitation of DNA-microarrays comes from mRNA measurements. Generally speaking, microarrays are used to find out the protein activation/inactivation via measuring mRNA released from a gene to the cytoplasm of a cell. This is a more stable process and possibly easier than measuring proteins in the translation step of a gene in the cytoplasm. However, for different reasons a gene might release mRNA but a defect in the cell might prevent that gene from being translated to protein , for example in cancer cells. DNA-microarrays can not detect this defect and might show an amount of gene expression at the mRNA level but still it will be questionable whether this gene is being translated to a protein. A more advance technology called protein expression analysis can be used which can tell whether a gene has been translated to a protein after an amount of mRNA is detected by DNA-microarray. However, protein identification and quantification technologies still far away from the high-throughput experimental techniques used to determine mRNA expression levels (Greenbaum et al. 2003).

4.3 Single-channel Microarrays

Single channel microarrays, of which the Affymetrix system is the most popular, are used to measure the expression levels of thousands of genes in parallel. They can be used for one-time experiments or time-series experiments, used for instance to measure a gene expression in different time under one condition. Affymetrix microarrays have been developed to generate gene expression profiles by measuring the signal intensity, the amount of hybridisation, of fluorescing molecules that are attached to DNA (which are reverse-transcribed from extracted mRNA or genomic DNA) that are bound to the complementary strand probes localised on the surface of the microarrays. Each localised probe in Affymetrix microarray is distinguished by a *probe-id*, which is used usually after the hybridisation to map each probe to a gene name. The design of Affymetrix experiments ,to obtain gene expressions, consists of three steps:(i) identifying the conditions of interest ,for example we are interested in comparing the healthy cell against a cancerous cell (ii) obtaining biological replicated of each condition, and (iii) preparing the hybridization sample. Preparing the hybridization sample includes preparing the surface of the chips by injecting the fragment of complementary DNA (cDNA) for each gene, for each condition, in each spot of the microarray (Wit & McClure 2004). However, the results of Affymetrix experiments are not in the final format for further analysis. The result of the hybridisation in each chip is a pixel file ($CELL$ file), which has *probe-sets* intensities that go through different normalisation steps. Each gene in Affymetrix microarrays is represnted with replicated probes,hence probe-set, and typically 11-20 probes are represented on a microarray for each gene. The fregment of a

gene usually 25 base pairs long and what might happen naturally is that mRNA from other genes may find parts on common with individual probe and attach itself to that probe. To overcome this problem, Affymetrix platforms have what is called mismatch probes(MM) which are obtained by changing the 13th base pair in the fregment of the gene localized in each spot in addition to the canonical probe(a section of the mRNA molecule of interest) "perfect match"(PM) for each gene. To obtain one single gene expression for any gene , the probe set has to be summarized as we will discuss in the next section.

Generally in single channel microarrays, a biologist is usually interested in a comparison between two conditions, for example, healthy genes compared to cancerous genes. However, it is not possible to use a single-channel microarray for more than one condition. Therefore, if two samples/conditions are used, the biologist needs two single-channels for each sample/condition. The number of samples and conditions are solely dependent on the number of microarray chips available. Since a single chip can measure thousands of genes in parallel, usually having many chips (samples) comparing to genes number is very expensive and therefore the resultant gene expression profiles are usually exposed to the problem of *large variables (p), small samples (n)*.

In the next section, we will go through the pre-process steps for generating gene expression profiles from single-channel microarrays. Then, we will apply it to row data (CELL files) appeared in (Birnie et al. 2008) that we will use throughout the next chapters.

4.4 Pre-processing Steps for Generating Gene Expression Profiles

Several methods have been proposed to normalise and pre-process Affymetrix raw data stored in a $CELL$ file. For example: MAS 5.0, dChip (Li & Wong 001a) and robust multichip average (RMA) algorithms (Irizarry et al. 003a) are well established methods for normalisation. Mainly , one can view the differences between MAS 5.0, dChip, and RMA algorithms in terms of probe intensities summarisation. MAS 5.0 algorithm computes the probe set intensity signal as the anti-log of a robust average of the values $log(PM_{ij} - CT_{ij})$ where CT is equal to MM when MM $<$ PM but adjusted to be $<$ PM when MM$\geq$ PM to non-negative summarised probe set signal values (Saviozzi & Calogero 2003; Wit & McClure 2004). So, a model for MAS 5.0 probe set intensity measures is :

$$\log(PM_{ij} - CT_{ij}) = \log(\theta_i + \epsilon_{ij}), j = 1, ..., J$$

Therefore, the expression value for a probe set on any array(i) is given by θ_i and an error term ϵ_{ij} that represents the variance for $j = 1, ..., J$ probe-set spots. The dChip algorithm also calculates the probe set intensity based on the difference between the perfect match(PM) probes and mismatch(MM) probes using multiplicative model:

$$PM_{ij} - MM_{ij} = \theta_i \phi + \epsilon_{ij}, i = 1, ...I, j = 1, ..., J$$

The dChip algorithm is based on the observation that the variation of a specific probe across multiple arrays could be smaller than the variance across probes within a probe set , which indicates a strong probe affinity effect (Li & Wong 001b).

On the other hand, RMA algorithm is based only on perfect match(PM) probes and uses another way of finding the correct summarised signal represented for each probe set intensity. The probe set model in RMA algorithm can be given generally as follows :

$$T(PM_{ij}) = e_i + a_j + \epsilon_{ij}, i = 1, ...I, j = 1, ..., J$$

Where T is multistep procedure conducted on probe set intensities which includes : background correction, quantile normalisation , and logs PM intensities, e_i is the log_2 scale expression value found on array i=1,...I and a_j is the log scale affinity effects for probes j=1,...J. (Irizarry et al. 003a) gives a comparison of RMA to the dChip and MAS 5.0 algorithms and shows that RMA has a better precision than MAS 5.0 and dChip. According to that , in this work we will use RMA algorithm to normalize the probe intensities data that we will use throughout this thesis. In the next section, a detailed section about the three steps of RMA algorithm will be given.

4.4.1 The normalization of probe set intensities using RMA Aglrothim

In this section , we will explain the RMA algorithm for normalisation which is based on perfect match(PM) probes , as it is well established and widely used in Affymetrix data normalisation (Barash et al. 2004; Laurent et al. 2004; Schlecht et al. 2004; Abeyta et al. 2004; Scott et al. 2004; Tsuchiya et al. 2004; Parmigiani et al. 2004; Barczak et al. 2003). It consists of three steps: background adjustment, quantile normalisation and summarisation (Bolstad et al. 2003; Irizarry et al. 003a,b).

The background adjustment/correction is used to make sure that the values obtained from the surface of a microarray when it is scanned for pixel intensities correspond to the amount of mRNA expressed. Figure 4.4 shows an example of how the background might affect the true mRNA signal.

The background corrected intensities are computed in such a way that all background-corrected values must be positive. After the background correction, base-2 logarithm is used for each background corrected value.

The second step in the RMA algorithm is quantile normalisation. Normally in the lab the experiment is replicated many times using different populations. For example, it might use different

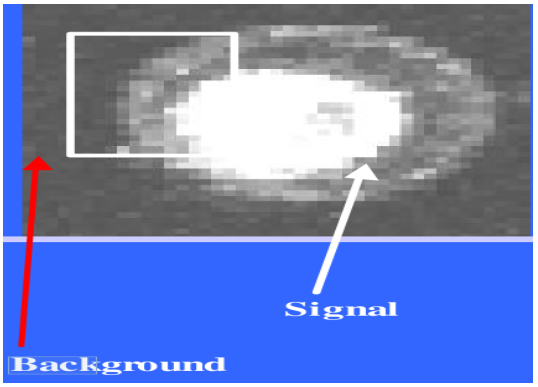


Figure 4.4: Background signals attached to the true mRNA signals in the surface of a chip (Yukhananov & Loguinov Yukhananov & Loguinov).

samples from different patients under one condition. The goal is to obtain one gene expression profile for all genes (n) across all samples (p) given one condition. Therefore, shift and scale parameters are used to remove the systematic differences between the different arrays (Wit & McClure 2004), called *normalisation across arrays*. Different methods to deal with different measurement scales from different arrays have been proposed. These methods transform the data by bringing the mean intensity for each array to some fixed quantity and scaling it to a fixed value (Kerr et al. 2000; Yang et al. 002b; Wolfinger et al. 2001). The choice of shift and scale parameters are largely arbitrary. The most common ones are the mean for shift and standard deviation for scale. The main objection to using this global method for removing systematic differences across samples is that it does not take into consideration the differences between arrays due to different environmental factors for each population used in each array (samples). For example, in an experiment using healthy patients as samples, each patient might be exposed to factors that affect their genetic profile; therefore, there might be a gene in one patient that is up-regulated but which is down-regulated in another patient. Using a global method will discretise this difference into a shifted and scaled value that might make these two patients have the same gene expression values. The global method has the intuition that given that the same genes are measured under one condition, such as healthy patients or cell lines, it is expected that the gene expressions values are similar in all cell lines under the same condition, which is not always true (Wit & McClure 2004).

Quantile normalisation is proposed to deal with the limitations encountered in the global method discussed above. Quantile normalisation includes transforming all the replicates for each gene (in the probe-level) into the same scale. The scale is attained by using the overall mean of the probe intensity distributions of all the replicates, which makes the distribution identical across arrays (Bolstad et al. 2003). Before the data is scaled using the mean of the replicates, an ar-

rangement to the probes across the arrays is made. This is done by ranking the probe values in each array incrementally and then taking the mean across the arrays. For example, if we have two vectors representing five spots and two replicates: $\mathbf{X_1} = [16, 0, 9, 11, 7]$ and $\mathbf{X_2} = [13, 3, 5, 14, 8]$. First, the rank for each probe in each array is taken $\mathbf{X_1^{ranked}} = [5, 1, 3, 4, 2]$ and $\mathbf{X_2^{ranked}} = [4, 1, 2, 5, 3]$. Then each vector is ordered $\mathbf{X_1^{ordered}} = [0, 7, 9, 11, 16]$, $\mathbf{X_2^{ordered}} = [3, 5, 8, 13, 14]$. Next, the mean for each two ordered probe across samples is taken $\bar{\mathbf{X}} = [1.5, 6, 8.5, 12, 15]$. Finally, we map $\mathbf{X_i^{ranked}}$ to $\bar{\mathbf{X}}$ and each probe quantile is normalised in each spot of the array $\mathbf{X_1^{norm}} = [15, 1.5, 8.5, 12, 6.0]$, $\mathbf{X_2^{norm}} = [12, 1.5, 6.0, 15, 8.5]$.

This method is also able to deal with non-linear relationships and by using the scale as the mean of all the distributions of all the replicates, it is hoped that the resultant distribution for each gene is a better reflection of the gene transcription (Wit & McClure 2004).

The last step for normalising gene intensities is summarisation. When the probe-levels are background corrected and normalised, each gene expression is still in the base of the probe-set in each spot. The gene measurements need to be summarised into a single number, representing a gene expression value in each spot (Irizarry et al. 003b; Bolstad et al. 2003). The probe-sets in each spot are identical pieces from the same DNA and are supposed to bind to the same complementary DNA (cDNA). Therefore, the signals or the intensities we get in the CELL file are probe-set intensities. The probe-sets are used in the hybridisation process, because more accurate results can be obtained, rather than relying on one piece of probe for hybridisation.

4.4.2 Pre-processing prostate cancer datasets from Affymetrix microarrays

This section presents the result of pre-processing the prostate cancer datasets that are used for further analysis in the chapters below. The focus on prostate cancer is based on (Birnie et al. 2008), which uses Affymetrix microarrays (Affymetrix GeneChip Human Genome U133 Plus 2.0) to generate the hybridised probes (.CELL files). The aim of this study was to identify genes that have significantly different expressions in stem cells from those in committed basal cells. Samples from both prostate cancer samples and benign controls were used. The RMA algorithm was used for normalisation and the result was four complete datasets, without missing values. Furthermore, 38 samples were used in the experiments, of which 19 samples were stem cells (SC), and 19 were committed basal cells (CB). When these four datasets were mapped to the KEGG database, the gene expression signatures were found to be enriched for genes from four main pathways: JAK-STAT signalling (Figure 4.5), Wnt signalling (Figure 1.4,page 21), the cell-extracellular matrix interaction pathway (Figure 4.6) and the focal adhesion signalling pathway (Figure 4.7).

As it becomes increasingly apparent that studying a small numbers of genes in isolation does not provide a sufficient understanding of the higher order systemic processes that regulate cell

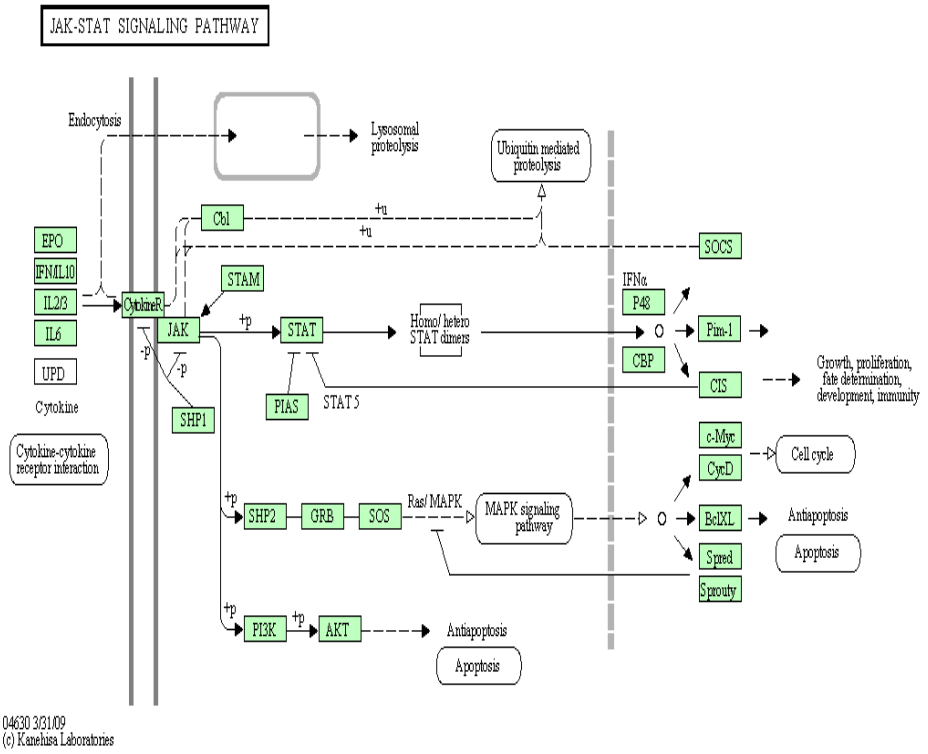


Figure 4.5: The JAK-STAT signalling pathway [Kanehisa Laboratories, 2009]

growth, researchers are becoming interested in finding methods that provide a picture of how each gene in a pathway interacts with those around it. There is no existing mechanism for accessing the specific connections between gene families that underlie the generic connections represented in the KEGG signalling diagrams. For example, the Wnt signalling pathway in Figure (1.4), page (21) shows that WNT directly interacts with Frizzled (FZD). However, there are 19 WNT and 10 FZD proteins listed in the KEGG database and KEGG does not show which member of the WNT family interacts with which in the Frizzled family.

After obtaining the result in Table 4.1, we concentrated on each pathway separately for the refining process. In this study, the result is based on the Wnt signalling pathway. The methods developed were then used to refine the other pathways.

The results in Table 4.1 were obtained by using different *Bioconductor* and *R* software packages. First, two libraries were loaded, hgu133plus2.db, which is a package corresponding to the Affymetrix microarray used in the experiments, and KEGG.db which is used for searching KEGG pathways to find the targeted pathways. When the RMA algorithm normalises the datasets, the result initially has probes that are irrelevant to the study, for example, some probes have no gene name annotations, since they are control probe IDs. Therefore, to get the probe

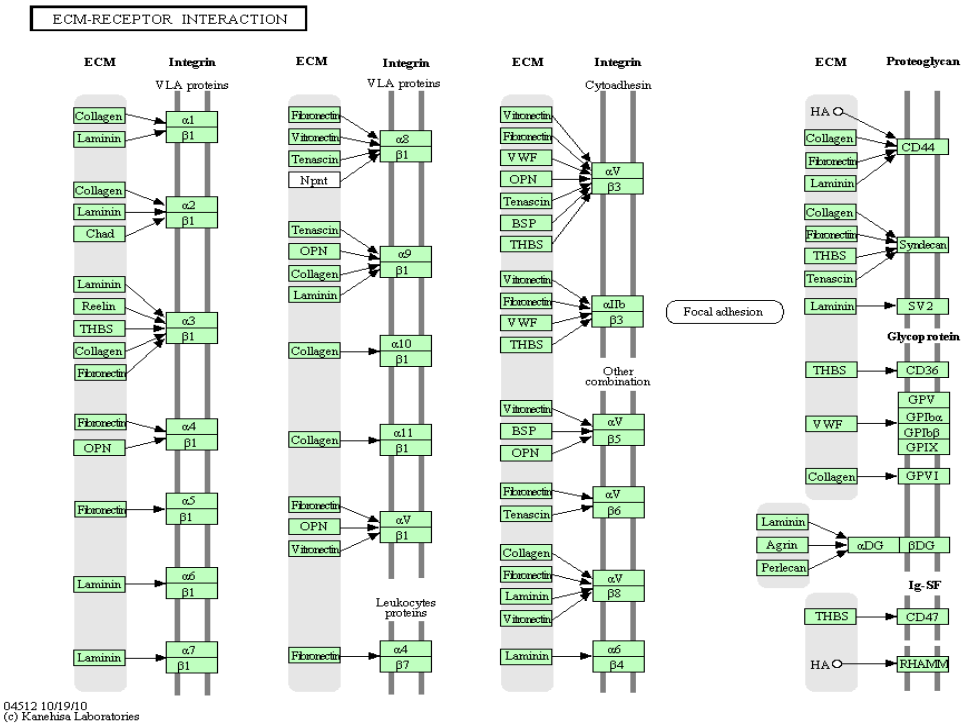


Figure 4.6: Cell-extracellular matrix interaction signalling pathway [Kanehisa Laboratories, 2009]

Table 4.1: Each pathway and its dataset

Pathway	Number of Samples	Number of Probes
Wnt signalling		451 probes
JAK-STAT	38 samples (19 SC,19 CB)	398 probes
Cell-extracellular matrix interaction		291 probes
Focal adhesion signalling		705 probes

IDs that have annotations, hgu133plus2.db was used to map the probe IDs to gene names. The KEGG.db package was then used to extract the probe IDs and gene names that are only in the four KEGG pathways we will focus on.

4.4.3 Wnt signalling pathway datasets

Since each dataset contains both cancer and non-cancer samples, it is essential to separate them further, because we want to understand how the cellular system works in cancer and non-cancer samples separately. The first thing we considered was separating the dataset into four different cell types: cancer stem cell, cancer committed basal cell, benign stem cell and benign committed basal cell. Each one was then treated individually, as shown in Table 4.2.

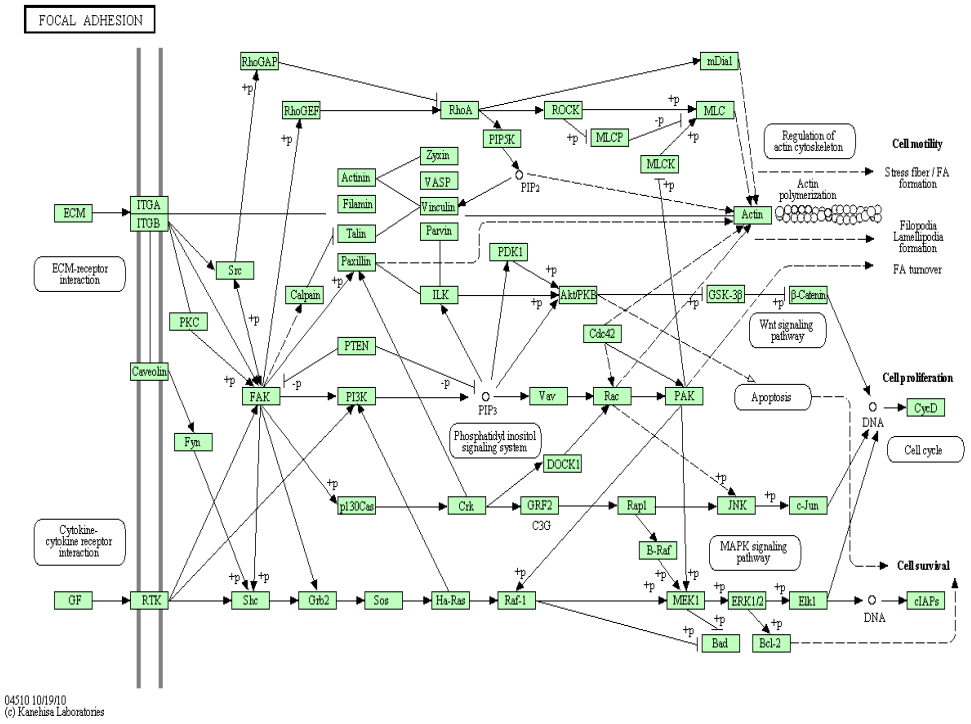


Figure 4.7: Focal adhesion signalling pathway [Kanehisa Laboratories, 2009]

Table 4.2: Wnt signalling Pathway Dataset.

Cell Type	SC dataset	CB dataset
Cancer	451 probes,13 samples	451 probes, 13 samples
Non-cancer	451 probes,6 samples	451 probes, 6 samples

4.4.3.1 Pre-processing stem cells (SC) in the Wnt signalling Pathway.

The stem cell dataset is used because it has more samples than the committed basal cell dataset. Moreover, since we are not interested in this thesis in differential gene expressions that happen between different conditions(cancer and non-cancer samples) nor between different cells(stem cells and committed basal cells) , we will focus on genes that show expressions. Genes that are unexpressed/down-expressed are discarded, Thus, there is a chance to reduce the dataset further. The question then becomes how to distinguish unexpressed/down-expressed genes from expressed or over-expressed genes. The sensitivity limits of the detection system mean that expression values < 50 are considered unexpressed/down-expressed. This corresponds to the value 5.64 in the gene expression profiles used here, since log_2 is used after the background correction method in the RMA algorithm. Thus, we can use the following logic: $IF\ gene\ value < 5.64, THEN\ it\ is\ excluded.$ Furthermore, since we are interested on genes that show expres-

sions , the gene that falls below the threshold used here(< 5.64) in more than or equal half of the samples($<= 13/2$) , will be discarded from the data. In fact, from machine learning point of view , all these pre-processing steps have been used to make *overfitting* problem less harmful as we will discuss in more details in chapter 6 . Based on this criterion, the SC dataset becomes smaller(212 probes out of 451 genes are kept in the SC cancer samples), and more importantly, contains fewer genes, which in turn reduces the possibility of overfitting or what is known as the $p \gg n$ problem, as we only have 13 samples. Therefore, when we learn a graphical model from this data in the next chapters , all the genes represented in the graph have shown expressions in the Affymetrix experiments and so the detailed KEGG pathways will be based only on the expressed values of genes used in the experiments.

We started by looking at a group of components that react together. Therefore, we extracted the upper left part of the Wnt signalling pathway, Figure 4.8, whose components are known from KEGG to react with each other. Table 4.3 shows the dataset of the 1st block from which cancer samples used in this study were taken.

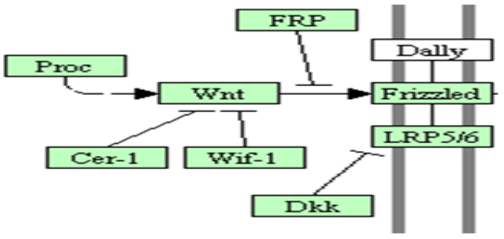


Figure 4.8: The first part of the Wnt signalling pathway.

Table 4.3: The SC dataset (cancer and non-cancer) for the 1st part of the Wnt signalling pathway.

Cancer	Non-Cancer
25 probes,13 samples	25 probes, 6 samples

4.4.4 Pre-processing colon cancer datasets from Illumina micorarrays

This section details a short experimental study carried out at King Abdullah International Medical Research Center (KAIMRC). It was based on using Illumina microarrays to explore the effectiveness of 4 treatments on 5 colon cancer cell lines (5 samples), in comparison to untreated colon cancer cell lines. The cell lines are HCT-116, HT-29, RKO, HCT-8 and Gc3/c1. The therapies used for treating colon cancer cell lines are:

- F:5-Fluorouracil + Leucovorin
- A: Interferon - Alpha

- G: Interferon - Gamma
- F+G: Interferon - Gamma + 5-Fluorouracil (F) + Leucovorin

The Illumina microarray (Illumina BeadArray) is a relatively new type of microarray for generating gene expression datasets. It has been repeatedly evolved in biological labs alongside with Affymetrix microarrays. A comparison study between these two types of microarray (Barnes et al. 2005) shows that for the purpose of spotting the differential gene expressions between two conditions, the two platforms find the same genes that are differentially expressed. However, the statistical methods for analysing Illumina data are still far from those methods used for Affymetrix and need to be improved (Xie et al. 2009).

To pre-process the data generated from the Illumina BeadArray we used Illumina BeadStudio software. The pre-processing or normalisation involves two steps: background correction and quantile normalisation.

The background correction method used in BeadStudio is called background subtraction. Background subtraction assumes that the total (S) signal (intensity) which comes from the chip for each spot has some noisy signals (B) (background signals). The true signal (T)=$S-B$. Therefore, the result will have positive and negative values, based on whether the gene is expressed or unexpressed. A normal scenario after background correction is to use log_2 for variance-stabilising transformation, but with negative values this will not be possible, a drawback which is reported in the literature (Xie et al. 2009). For the gene expression datasets that we used, the view was that the negative values should be excluded, either because they were noisy or had low expression values. We used this step, since the aim was to keep only the genes with high expression values. This was for two reasons, firstly, we only had five samples, so we needed to cut off the dimensionality of having a lot of genes and small sample sizes. The second reason was that we are interested in the most important genes; therefore, a compact picture of how high expressed genes work in different treatments vs the control (untreated) cell line can be observed. This will also allow us to use log transformation for variance stability and also more linear relationship can be observed. After the data from Illumina is background corrected and logged, quantile normalisation is used as a final step for pre-processing. We used the same cut off value ($IF\ gene\ value < 5.64, THEN\ it\ is\ excluded$) as used in Affymetrix, in order to keep the highly expressed genes in the datasets. The result is 5 complete datasets, 4 treated and 1 untreated, without any missing values.

Each treated and control gene expression profile was then mapped to the KEGG database. We restricted the search in KEGG to colorectal cancer pathways and found that a large proportion of the genes in each treatment were annotated in two well known colorectal cancer pathways, the MAPK signalling pathway and cell cycle pathway. Figure 4.9 shows how many genes are found

annotated in each colorectal cancer pathway in KEGG and we can see that a large proportion of genes are found in the MAPK and cell cycle pathways. In Chapter 6, we detail how we used the methods developed to learn refined MAPK and cell cycle pathways, based on the genes we have in each treatment and this will show how the behaviour of each treated cellular system changes after applying each treatment, compared to the control cellular system.

After we have obtained two sets of normalised datasets, we can proceed to explore and develop a method that can effectively show meaningful results. The next chapter will show the first attempt to learn the genes relationships between the genes involved in prostate cancer, shown in Table 4.3 using existing tools. Chapter 6 will show a different approach to the problem of learning graphical models, beyond the existing tools. The method developed will then be used to learn complete graphical models from prostate cancer datasets, as well as colon cancer datasets.

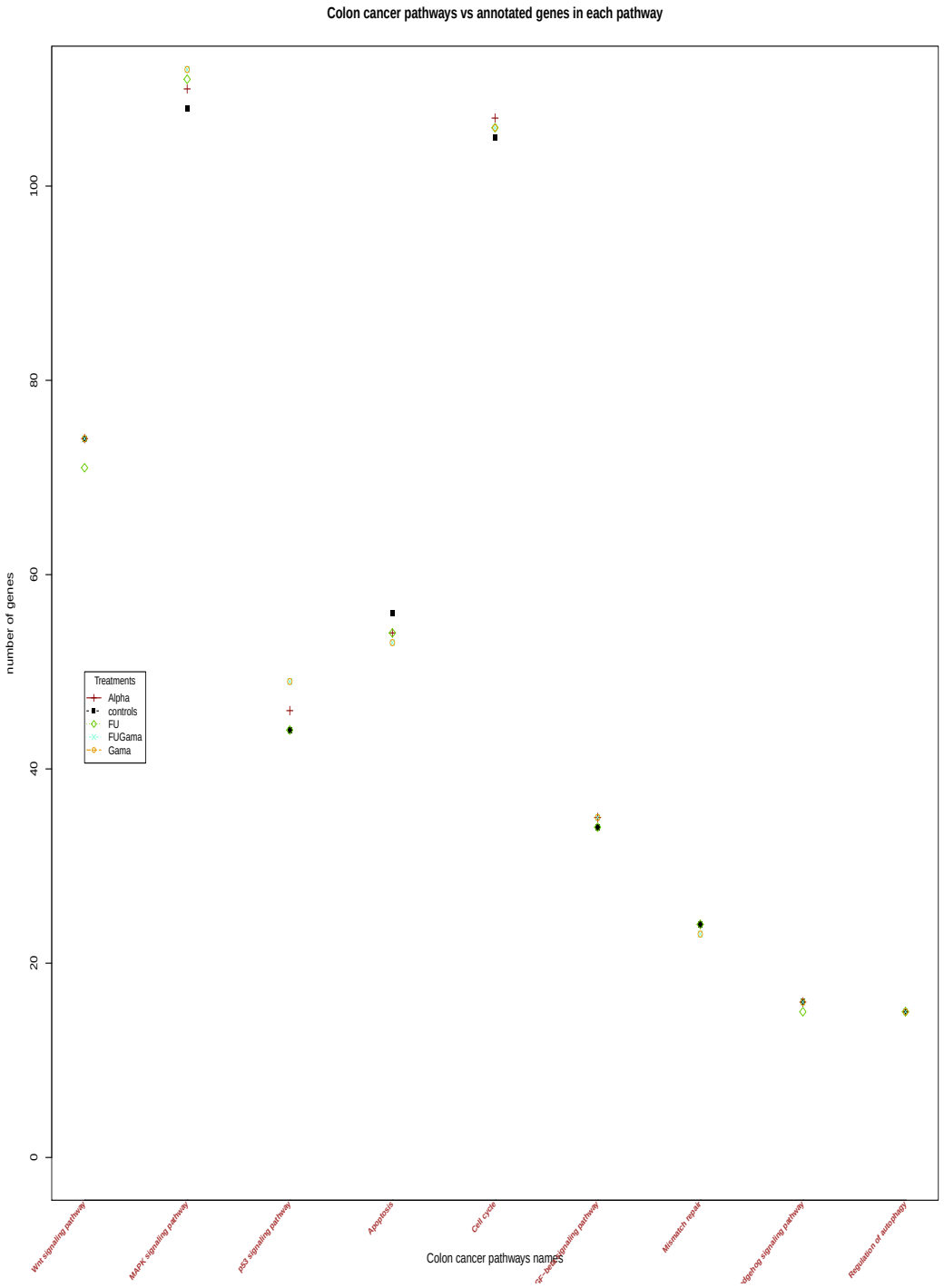


Figure 4.9: Colorectal cancer pathways and the number of genes annotated in each pathway

CHAPTER 5

Learning refined graphical models for KEGG pathways using existing tools

In this chapter, existing tools are investigated in order to learn Bayesian networks from the data in Table 5.1. Each tool was used with natural prior knowledge from the KEGG database, found to be useful in minimising the search space. Section 5.6 gives our conclusions and shows the drawbacks encountered with the tools that were appraised in this chapter.

5.1 Introduction

After focusing on the four pathways and then narrowing the focus to the Wnt signaling pathway, we took a further step to concentrate on the first part of the Wnt signaling pathway, as shown in Figure 4.8. For the dataset associated with the first part, we separated stem cells (SC) from committed basal cells (CB). Furthermore, the cancer and non-cancer samples were also separated. The work in this chapter is based on the *cancer dataset* shown in Table 4.3, which has the format shown below in Table 5.1.

This chapter shows the first work undertaken to extend the representation of the first block of the Wnt signaling pathway to involve more details about the cellular system interaction between genes. The set of graphical models used in this chapter are Bayesian networks learning algorithms from discrete and continuous data sets.

Before we introduce the first Bayesian network learning algorithm, we need to formalise

Table 5.1: Part of the cancer dataset(stem cell samples) for the first part of the Wnt signaling pathway.

219483-s-at	205990-s-at	213425-at	231227-at	....2205606-at
PORCN	WNT5a	WNT5a	WNT5a	....LRP6.1
8.86058	9.35868	8.84773	7.34806	...8.16661
8.2727	8.97303	7.91203	5.65153	...7.09921

the dataset to generate a readable network. The part of the first block in Table 5.1 shows the probe IDs that correspond to genes. Since we like to see the network showing gene names in the nodes, the probe IDs have been dropped. But this is not a straightforward task, as some genes have several different probe IDs. For example, WNT5A has three different probe IDs and if we dropped the probe IDs for this gene, this might cause confusion for the learning algorithm when it comes to representing the relationship between different WNT5A probe IDs and other genes. One way to avoid this is to give the probe IDs that correspond to one gene different notations, such as WNT5A, WNT5A.1 and WNT5A.2. Therefore, the new dataset is as shown in Table 5.2.

Table 5.2: The dataset after the probe IDs have been dropped

PORCN	WNT5a	WNT5a.1	WNT5a.2	....LRP6.1
8.86058	9.35868	8.84773	7.34806	...8.16661
8.2727	8.97303	7.91203	5.65153	...7.09921

As shown below, this method helps to evaluate the learning algorithm. For example, when the generated network contains a part like the one in Figure 5.1 we can conclude that this algorithm works in the right way, because it recognises that these genes should come together, as they are the same genes for different probes. However, doing this will not allow other genes to be examined, since once any of the wnt5a genes is chosen as a predictor, it will be enough and this will not show any improvement for any other genes outside the wnt5a family to be parents. Moreover, the reason that we did not average them to one value is that, as seen below, although they are all represent the same gene, they have different relationships with other genes. Hence, the biologist can see more details about these different probes and the interactions they hold.

5.2 WEKA: Machine Learning Software

The first tool we will use is Weka (Hall et al. 2009), which is open source software with a collection of machine learning algorithms. Since we are dealing with graphical model learning algorithms, we will look at a specific algorithm in this tool. The K2 algorithm that is a Bayesian method for the induction of probabilistic networks (Cooper & Herskovits 1991). It is a heuristic search algorithm guided by a score function which adds parents to the child until its score does

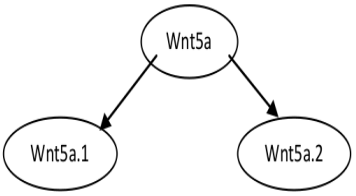


Figure 5.1: WNT5A probes interaction.

not change. The score function evaluates how well each network fits a given dataset. The search returns the network with the highest score.

Before turning to the tool, there are requirements that have to be met when the K2 algorithm is used. Firstly, the dataset used has to have ordered variables. We have to assume ordering between variables. This means that if the dataset has three variables $D = [X1, X2, X3]$, then the search space of all possible networks has a size of 8 networks(2^n), where n the number of variables, since each variable has possible parents from the variables preceding it in the order. Figure 5.2 shows the possible networks from dataset D. In this way, the search space has been

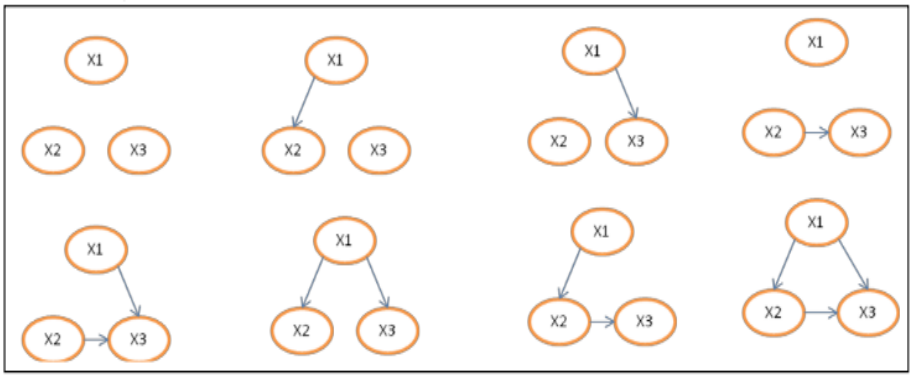


Figure 5.2: All possible networks generated by K2.

reduced to 8 networks instead of 25. Table2.1(Chapter 2,page33) shows all possible networks for different sizes of datasets, when the assumption of ordered variables is not made. However, even with ordered variables there are exponentially many Bayesian networks in the search space, if the parents set for each gene are not limited.

To meet the first assumption, we require background knowledge and the background knowledge from Figure 4.8 is used. This representation reveals the following:

1. The PORCN component indirectly affects the WNT component.
2. The Cer-1 component inhibits the WNT component.
3. The Wif-1 component inhibits the WNT component.
4. The FRP component inhibits the WNT component.
5. The Dkk component inhibits the LRP65/6 component.
6. The WNT component activates the Frizzled component.
7. The Frizzled and LRP6/5 components bind to each other.

However, after consulting the database of KEGG pathways that underlie the KEGG diagrams, it was clear that the diagram in Figure 4.8 has a limitation in its presentation. It shows that Frizzled and LRP6/5 are dependent on each other (binding to each other in biological language) but the database underlying it shows more; Frizzled activates LRP6/5 but LRP6/5 does not influence Frizzled. Therefore, we chose the database as a source of prior knowledge rather than the diagrams. Furthermore, as we are interested only in genes that are expressed, the Cer-1-component and the Wif-1 component were removed from the dataset, because they were not expressed ($genes\ value < 5.64$). Accordingly, the final prior knowledge used was:

1. The PORCN component indirectly affects the WNT component.
2. The Dkk component inhibits the LRP65/6 component.
3. The FRP component inhibits the WNT component.
4. The WNT component activates the Frizzled component.
5. The Frizzled component activates the LRP6/5 component.

Therefore, one possible partial order is: *PORCN—WNT—DKK—FRP—Frizzled—LRP5.6*. We will treat this partial order as a full order to the dataset, as is required by the algorithm. The second requirement is for the dataset to have discrete values. The dataset we have comprises continuous gene expression values and therefore, to adapt the dataset to be acceptable by the K2 algorithm in Weka, the dataset was discretised before using it. (Dougherty et al. 1995) describes different ways of choosing the best discretisation method; we made a natural discretisation, using biologists in the lab. The suggestion was to plot the histogram of the dataset as in Figure 5.3 which then tells us how to choose the best intervals.

One choice, based on the histogram in Figure 5.3 could be as follows:

$If(Gene < 5.6)THEN(\text{"}very - low\text{"});$
$Else$

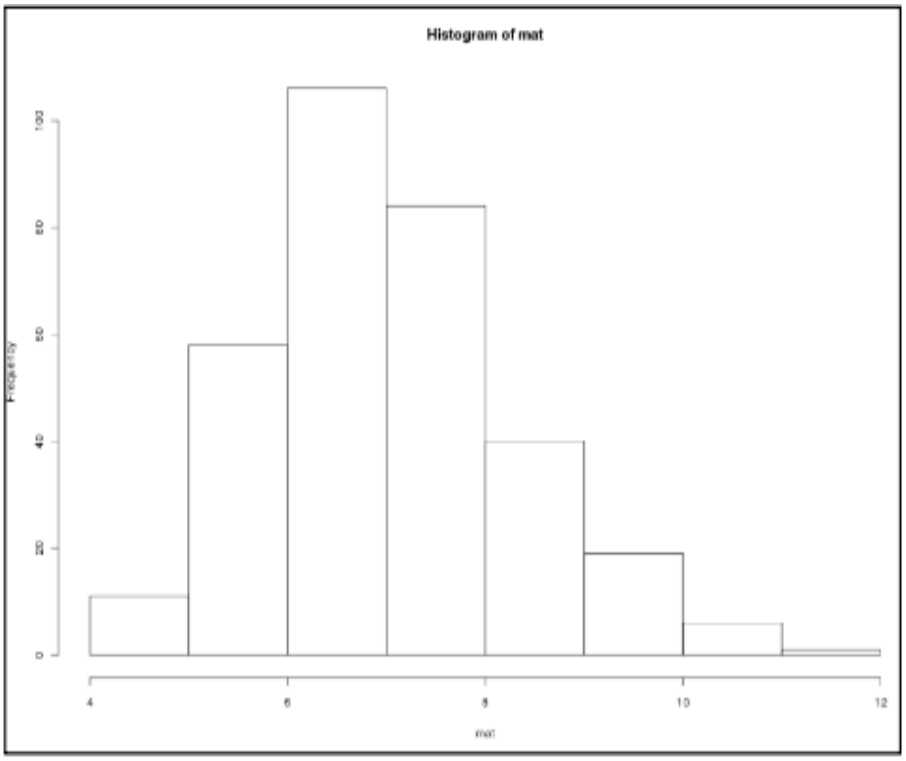


Figure 5.3: A histogram of the dataset shows in x-axis the values of gene expressions and in y-axis the frequency of each interval of values.

$If(Gene >= 5.6\ AND\ Gene <= 7.9) THEN(\text{"}low\text{"});$
$Else$
$If(Gene >= 7.9\ AND\ Gene <= 9.9) THEN(\text{"}medium\text{"});$
$Else$
$if(Gene > 9.9) THEN(\text{"}high\text{"});$

Figure 5.4 shows a snapshot of the discretised dataset and Figure 5.5 shows the resultant Bayesian network.

5.2.1 Discussion

From the network shown in Figure 5.5 we see that there are a few isolated nodes; for example, wnt3, wnt6 and wnt10b. There are several reasons for this incomplete network. Firstly, the logic we used to discretise the dataset did not fulfill expectations, because the values of the unconnected nodes do not help to find any patterns. To illustrate this, if we look at the wnt3 values, we can see that they are all low. Thus, it is very difficult to find a relationship that supports these values with any other genes and also this can be seen in genes wnt6, and wnt10b . Secondly,

PORCN	WNT3	WNT5A	WNT5A.2	WNT6	WNT10B
medium	low	medium	low	low	low
medium	low	medium	low	low	low
high	low	medium	low	low	low
medium	low	low	very_low	low	low
low	low	medium	low	low	low
low	low	medium	low	low	low
medium	low	low	low	low	low
low	low	medium	medium	low	low
medium	low	low	very_low	low	low
medium	low	low	very_low	low	low
medium	low	low	very_low	low	low
medium	low	medium	low	low	low
medium	low	medium	low	low	low

Figure 5.4: A snapshot of the discretised dataset.

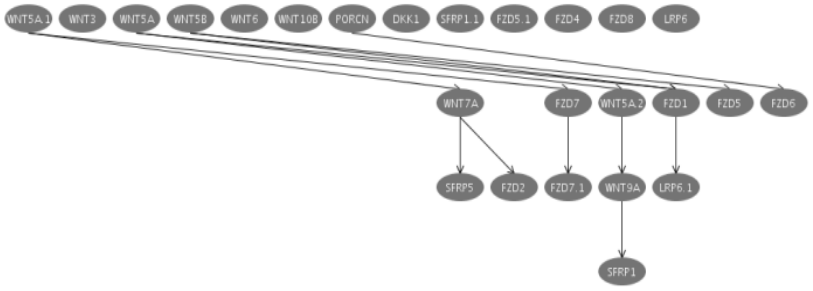


Figure 5.5: The resultant Bayesian network from the K2 algorithm.

the dataset consists of just 13 samples, which is another barrier to finding useful patterns between the genes in the dataset. There are other methods for better discretisation, for example, supervised discretisation or unsupervised discretisation might result in better networks. These methods would allow us to refine the discretisation process to accept more categories (bins or intervals); examples are, Equal Interval Width and Entropy-based partitioning (Dougherty et al. 1995). However, after searching the literature and obtaining the results from the K2 algorithm, it is clear that any kind of discretisation method will yield a loss of information, especially when the dataset is small.

Additionally, if we labelled the classes as cancer and non-cancer, it is likely that the classification information would be lost by partitioning, as a result of combining values into the same partition. The same principle can be seen in our dataset, where genes with high expression levels might be lost by combining them with normal or even low expression levels. Therefore, in the next sections the dataset is used without discretisation.

5.3 The Bayesian Network Wizard Tool

The result from the Weka tool yields loss of information because discretisation causes such problem, poor patterns between genes and, more importantly, combines different genes with different significant values into one interval. In this section, we will use the Bayesian network wizard tool (BNW) (Ferrazzi et al. 2007) to generate a Bayesian network from the dataset in its original form. Thus, the dataset will be used without discretisation. The different advantages and disadvantages of the tool during the experiments will be addressed. This tool can learn a Bayesian network from any type of data (discrete/continuous, static/dynamic). For our current purpose, we will focus on learning from continuous, static data, due to problems with discrete data which are discussed in detail above. However, we show briefly in Figure 5.6 that the resultant network from BNW tool, using discrete data, is worse than the one given by the Weka tool.

One explanation of Figure 5.6 is that no patterns were found between the genes except for two genes. Again, this was expected, since we do not have much data to support the learning. Also, discrete data cannot be used with the dynamic option in the tool, due to memory capacity. The tool shows a message that the dataset should have, at most, 20 variables to use this type of learning; however, we have 25 variables. Moreover, choosing dynamic learning requires that the measured gene expression values in the dataset should be taken at different times; for example t_1, t_2, t_3 but the measurements that we have in the dataset were taken in one-time. Therefore, the only options that satisfy the situation are continuous with static data. Using this tool requires the assumption of multivariate normal distribution. Thus, the dependency of each variable on its parents is a linear combination of the non-linear functions of the parents values, as shown in equation (5.1):

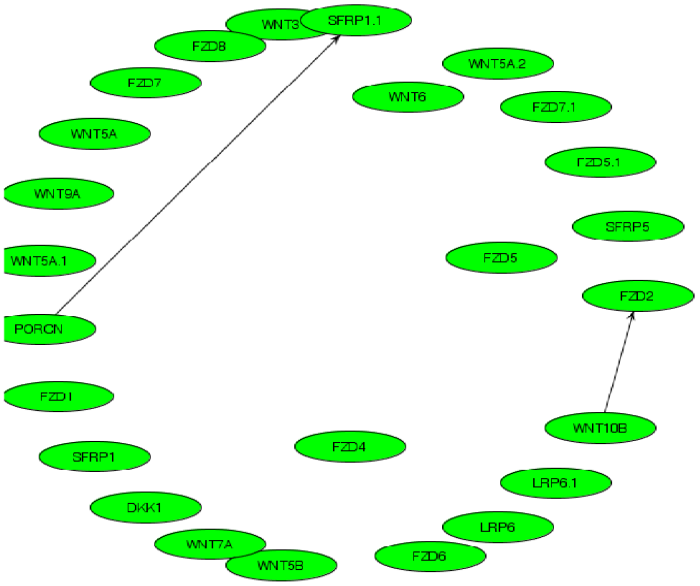


Figure 5.6: The resultant discrete Bayesian network from BNW tool.

$$\mu_i = \beta_{i0} + \sum_{j=1}^{p} \beta_{ij} \Phi(x_{ij}) \tag{5.1}$$

and $\Phi(.)$ set to $\tanh(\alpha x)$, where α is a predefined parameter.

To choose a network from the search space, a Bayesian model is used to select the best network, so the search will return the network with maximum posterior probability:

$$P(M_h|D) \propto P(M_h)P(D|M_h)$$

Where:

$P(M_h|D)$ is the probability of a Bayesian network, given the data.

$P(M_h)$ is the prior probability of the network.

$P(D|M_h)$ is the marginal likelihood.

An assumption of "equally likely for all models" is applied. Therefore, searching for a Bayesian network with maximum posterior probability is equivalent to searching for one with the maximum marginal likelihood. The scoring function of the local marginal likelihood is given in (Ferrazzi et al. 2007) as follows :

$$P(D|M_{hi}) = \frac{1}{(2\pi)^{n/2}} \frac{(det(R)_{i0})^{1/2}}{(det(R)_{in})^{1/2}} \frac{\Gamma(v_{in}/2)}{\Gamma(v_{i0}/2)} \frac{(v_{i0}\sigma_{i0}^2/2)^{v_{i0}/2}}{(v_{in}\sigma_{in}^2/2)^{v_{in}/2}} \tag{5.2}$$

Where R_{i0} = the identity matrix,R_{in}=$R_{i0} + X_i^T X_i$, v= the sample size,σ^2= the variance.

The search strategy used here is the K2 algorithm. Thus, an ordering is assumed between variables. This algorithm evaluates models of increasing complexity, as long as there is a gain in the marginal likelihood and stops when adding parents does not increase the scoring function. To reduce the risk of finding suboptimal models, a stepwise search has been implemented. The old marginal likelihood is not only compared with the marginal likelihood of the model when the new parent which increases the score is added, but also with the marginal likelihood values of the models with the new parent and when one of the old parents is removed. Figure 5.7 shows the resultant Bayesian network from the Bayesian network Wizard tool.

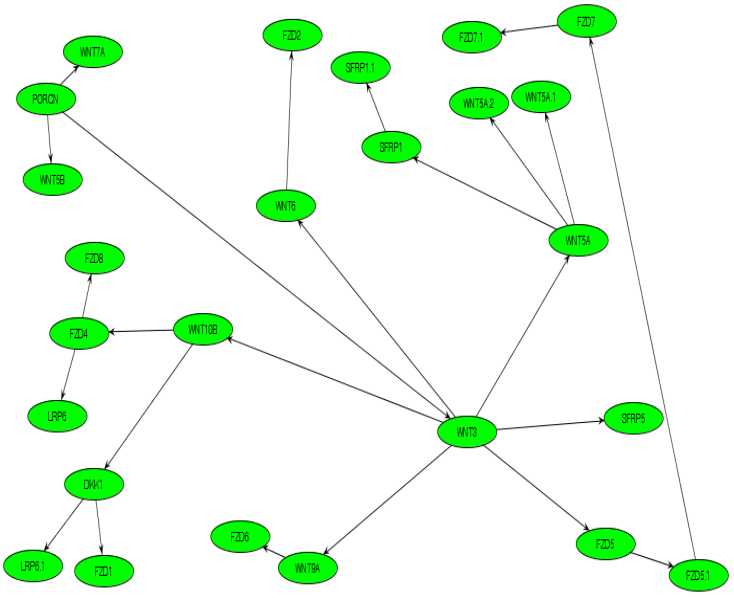


Figure 5.7: The resultant Bayesian network from the BNW tool using a continuous dataset

5.3.1 Discussion

In Figure 5.7, some genes are represented with more than one variable, because the same gene is assigned to different probes. For example, wnt5a was labelled (3 times) as wnt5a, wnt5a.1 and wnt5a.2 and wnt7a was labelled (twice) as wnt7a and wnt7a.1. Since these genes are basically the same, we expect them to be close to each other. The Bayesian network in Figure 5.7 shows that wnt5a genes and wnt7a genes are close to each other, but not lrp6 and lrp6.1. However, it seems that if there is enough data lrp6 and lrp6.1 will end up close to each other. Some further disadvantages have also been addressed:

- The dependency of each variable on its parents is a linear combination of the non-linear functions of the parents values, and $\tanh$ is used as a nonlinear function. However, the $\tanh()$ function is not necessarily the best function, even if it has shown a perfect fit with the current models. $\log_2$ function, for example, could outperform it for a non-linear transformation in another dataset.

- In setting the parameters on the scoring function (5.2), there are two sensitive parameters, v and σ^2 which are sample size and variance. These parameters greatly influence what is learned, actually changing the structure of the Bayesian network, but there is no optimal choice for setting such parameters. For example, the parameters are set to fixed values in the tool $\sigma = 1.0$ and $v = 3.0$, in order to have a large variance (Ferrazzi et al. 2007). However, there is no guarantee that this is the optimal choice for these two parameters. (Silander et al. 2007) shows a similar scoring function "BDeu marginal likelihood score", which has a single parameter, the equivalent sample size. It shows that the resultant Bayesian networks are highly sensitive to the chosen parameter value. Another example is the scoring function "BGe score function" described in (Geiger & Heckerman 1994b), which requires two parameters to be set, the sample size parameter and the prior network parameter. It is used with different settings by (Friedman 2004).

- The maximum number of parents that each child is allowed is set to 5. Although, genetic regulation networks are sparse, since for a given gene it is assumed that no more than a few dozen genes directly affect its transcription (Friedman et al. 2000), we cannot find exactly how many there are. Moreover, by setting the maximum number of parents randomly, we might miss important parents.

- The version of the K2 algorithm used in this tool is implemented on the basis of a full order assumed between the variables in the dataset. However, we know that we have partial order in the dataset. The prior knowledge that we used from the KEGG pathways database helps to order genes as one possible full order, as used in the previous tool, but some genes will have parents even if the prior knowledge does not support this. For example

the resultant graph in Figure 5.7 is based on the following order: *PROC—WNT—DKK—FRP—Frizzled—LRP5.6*. Therefore, the graph has some irrelative relationships to the prior knowledge, for example $WNT10B \rightarrow DKK1$ which not exist in KEGG database. Moreover, each component or family of genes, for example the WNT family, also has several genes that appear in the dataset, but their order remains unknown and using an arbitrary order will result in some of them being missed. For example, the WNT family has a lot of genes; if the order is "wnt3, wnt5a, wnt6", for example, we cannot check whether the data supports "$wnt6 \rightarrow wnt5a$".

5.4 The WinMine Toolkit

The WinMine toolkit is a set of tools created by Microsoft (Chickering 2002), which make it possible to build statistical models from data. The main reason for using this tool is to make use of partial ordering. In the previous tool, the dataset was assumed to have full ordering and there was no choice to invoke a partial order, based on the ordering between components. One of the features of the WinMine tool is that the variables in the dataset can be partially ordered. Figure 5.8 shows the method used to give a partial order to the tool.

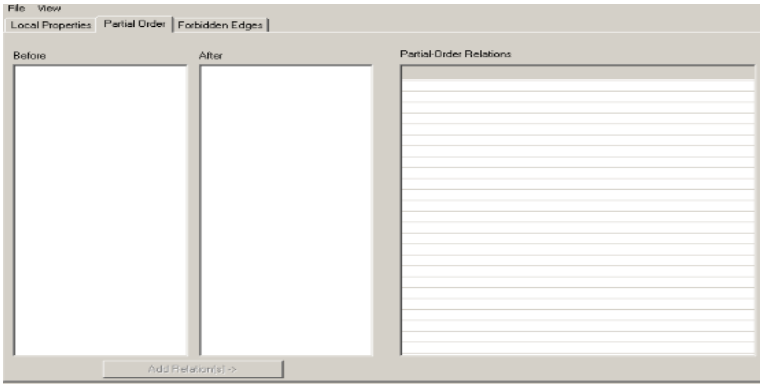


Figure 5.8: Partial order screen in the WinMine tool.

Since we did not have access to the algorithms and the scoring function implemented in this tool, the best we can do is to show the results we obtained from using this tool. The tool was given a continuous dataset and partial order, the result obtained is a graph without connections, as shown in Figure 5.9.

The result shown in Figure 5.9 is not encouraging, possibly because of the small sample size. However, since the tool does not have a reference, apart from (Chickering 2002), we are unable to provide any further reasons.

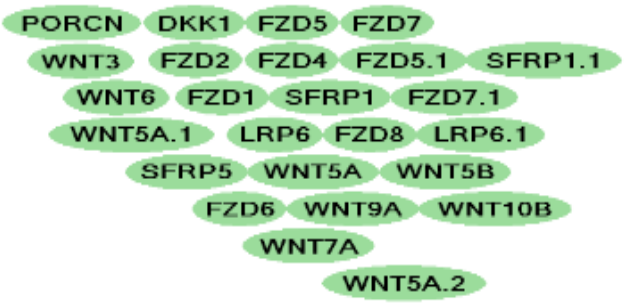


Figure 5.9: The resultant network without edges from the WinMine tool.

5.5 Deal tool for learning the Bayesian network

The Deal tool is an R package that has the ability to learn a Bayesian network from continuous, discrete or mixed data. It includes several methods for analysing data using Bayesian networks (Bøtcher & Dethlefsen 2009) and does not require any assumption about ordering between variables. However, it allows the implicit ordering of variables, as shown below.

The scoring function used in Deal is the calculation of the posterior probability using Bayes theorem for each Bayesian network as in (5.3).

$$P(BN|D) = \frac{P(D|BN)P(BN)}{P(D)} \tag{5.3}$$

$P(D)$ is a constant and does not depend on BN; therefore, it is not necessary to calculate it when comparing two networks. $P(BN)$ is the prior probability of a Bayesian Network. In Deal, all Bayesian networks are equally likely, therefore:

$$P(D|BN) \propto P(BN|D)$$

To compare two Bayesian networks, a Bayes factor is used:

$$BF = \frac{P(BN|D)}{P(BN_{new}|D)}$$

The Bayes factor is used to find the network with the highest score function when two networks are scored by (5.3).

A basic heuristic greedy search was implemented in Deal to search the search space. The network with the highest posterior probability is always preferable.

Deal allows the implicit invocation of the order to the dataset before learning. It is possible to tell the algorithm that there are variables in the dataset about which we do not have prior knowledge or which are not allowed to interact. We can use matrix^{n*2}, which bans certain directions. This means that, for example, if we attach the following matrix to the network:

$$\begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix}$$

then we are asking the algorithm to disallow: Gene1→ Gene4, Gene2 → Gene5, and Gene3 → Gene6. Therefore, the values in the matrix are the indices of the variables in the dataset. The resultant network is shown in Figure 5.10.

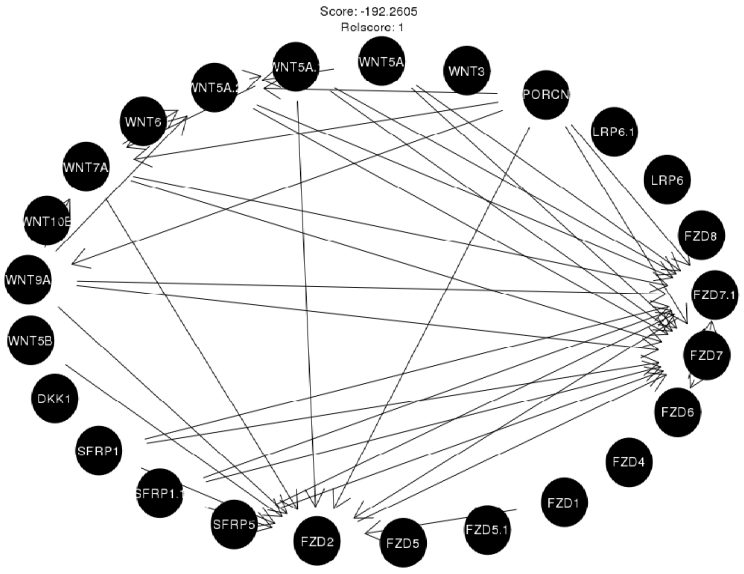


Figure 5.10: Incomplete results from the Deal package

The results in Figure 5.10 are incomplete, because the learning cannot be completed. The error message received was ”cannot allocate vector of size 20.2 mb”. This is a memory problem, which is to be expected, as the greedy search in Deal allocated all possible networks that differed even by added, removed, or reserved arrows. It then used the BF ratio to compare each one with

every other one, based on the posterior probability.

5.6 Summary

In summary, although we tried several different tools, none of them gave a satisfactory result. The problems we observed can be summarised as follows:

1. The discretisation (Weka tool) yields poor patterns between genes; or
2. the tool (BN-Wizard) assumes a full order between variables and we do not have this in the dataset; and,
3. there is no interpretable answer from the tool at all and it could not access the implemented algorithms (WinMine); and
4. the learning phase crashed before the final result, due to the implementation of the algorithm (Deal) and the memory capacity.

Another common problem among all tools is that we have a small dataset. As each tool required a substantial amount of time for analysing the results obtained from it, trying other tools for learning a Bayesian network from the dataset is time-consuming, as we are limited in terms of time. Therefore, in the next chapter we will show another direction for learning graphical models in general, which will lead to the methods developed and used throughout all the refined KEGG pathways.

CHAPTER 6

Learning linear Gaussian models

Chapter 5 looked at some of the problems with existing tools, such as loss of information from discretisation and the lack of full ordering in the dataset in Table 5.1. This chapter will show the detailed direction of a new setting to solve the problem of learning graphical models from continuous datasets. In Section 6.1, the problem of learning from sparse datasets and the goal of model selection methods are addressed. The assumption of multivariate normal distribution is explained in Section 6.2, which shows how this assumption is assessed for the cancer dataset(stem cell samples) in Table 5.1, the starting point for learning linear Gaussian models. Therefore, the work in this chapter is also based on the cancer samples of stem cells that is used in the previous chapter(Table 5.1). Section 6.4 is mainly about different variable selection methods, used to learn from normal linear regression. Section 6.5 presents more sophisticated methods which contribute to learning from gene expression datasets in this thesis. In section 6.7, an intensive evaluation of the developed methods is shown, including: learning from large and small subsets of parents for each gene, when 13 samples are used to learn from; validating the methods we developed with a bigger dataset; and showing the robustness of the developed methods using statistical tests. The last section examines to what degree the prior knowledge that is used to order variables in the dataset in Table 5.1 is captured when the prior knowledge is not used.

6.1 Introduction

The problem of learning from a large number of candidates and small samples has recently been addressed in the field of biology, because of the importance of data collection technologies. One example of this is Affymetrix microarray data analysis techniques, in which the number of genes

(predictors) to be examined exceeds the number of samples (observations). For such problems , linear Gaussian models are used intensively to address the problem of model selection and the ultimate goal is to achieve:

- $An\ accurate\ prediction$: The prediction and the most powerful predictors can be improved by using methods that trade-off between bias and variance.

- $An\ interpretable\ model$: providing which predictors are meaningful rather than using all predictors.

- $Stability$: by choosing the most important predictors, a small change in the data will not result in large changes in the subset of predictors (Hastie et al. 2009).

In this chapter, predictor, independent, regressor and parent are used interchangeably. Furthermore, gene and child are also used interchangeably for dependent variables.

Since we are dealing with continuous variables, one of the most popular continuous distributions used in graphical model learning problems is the class of multivariate Gaussian distributions, in which the relationships between variables are in linear form. Moreover, a multivariate Gaussian distribution can be generalised to encode non-linear relationships (Koller & Friedman 2009). In this chapter, we are going to focus on the linear form of Gaussian distribution and proceed to learn the linear relationships between each gene and its potential parents.

6.2 Multivariate Normal Distribution

A multivariate Gaussian distribution is a generalisation of a Normal distribution (one variable). A p-multivariate Normal distribution with mean μ and covariance matrix Σ is denoted by $\sim N_p(\mu, \Sigma)$. If a dataset has a multivariate Normal distribution, then we can model the relationship between a response and one or more regressors in a linear form. It is also known (Geiger & Heckerman 1994a) that a multivariate Gaussian distribution can be decomposed into a product of conditional distributions:

$$P(X_1, X_2, X_3, ...) = P(X_1)P(X_2|X_1)P(X_3|X_1, X_2), ... \tag{6.1}$$

where the relationship between parents and children fits a linear regression model. Thus, for each gene we seek a subset of parents that are good predictors assuming the linear model (6.2).

$$\mu_i = \beta_{i0} + \sum_{j=1}^{p} \beta_{ij} x_{ij} \tag{6.2}$$

6.2.1 Linear regression

Linear regression analyses the relationship between two or more variables. It expresses the linear relationship, for example, between two variables X and Y in linear equation:

$$Y = \beta_0 + \beta_1 x \tag{6.3}$$

Where Y is a response variable and x is a single regressor or predictor variable. β_0 and β_1 are the coefficients, and their values determine how the line is drawn between Y and x. Using linear regression allows us to find the dependency between two or more variables, where Y is the dependent variable and xs are independent variables.

An important quantity in linear regression is least squares error, which is the difference between the actual value Y and the estimated value $\hat{Y}$. Therefore, if we know the actual value of Y in equation (6.3) in advance, using x and the associated coefficients to predict Y is denoted as $\hat{Y}$ and least squares error can be used to find the prediction error by subtracting Y from $\hat{Y}$ and squaring it. The square is used for two reasons, to ensure the difference is a positive value and to make least square errors correspond to the maximum likelihood.

A quantity close to linear regression is the correlation coefficient (r) which is found by:

$$\mathrm{r} = \frac{\sum_{i=1}^{n}(Y_i - \bar{Y})(X_i - \bar{X})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})^2}}$$

The correlation coefficient is fundamental to linear regression analysis, in which it expresses the strength of the linear relationship between two variables. Two variables are said to be correlated if a change in one variable is associated with a change in the other variable. The strength of the relationship between two variables can be expressed in correlations, as in Table 6.1.

Table 6.1: Meaning of correlation coefficients between two variables X, Y

r = +1.0	The relationship is Strong-positive	As X goes up, Y always goes up
r = +0.5	The relationship is Weak-positive	As X goes up, Y tends to usually go up
r = 0.0	no correlation	X and Y are not correlated
r = -0.5	The relationship is Weak-negative	As X goes up, Y tends to usually go down
r = -0.1	The relationship is Strong-negative	As X goes up, Y always goes down

It is important to note that when $r = 0.0$ it only means that there is no *linear* relationship between X and Y. Thus, there might be a non-linear relationship between X and Y, which cannot

be captured by correlation coefficient r. However, given the data is drawn from multivariate Normal distribution, when $r = 0.0$ between X and Y, it implies that X and Y are independent.

6.3 Assessing multivariate normal distribution for the first part of the Wnt signaling pathway dataset

The first step towards the regression learning problem is to assess the assumption of multivariate Normal distribution for the dataset used. In this section, we assess to what degree the dataset for the first part of the Wnt signaling pathway in Table 5.1(Chapter 5,page 64)meets the assumption of Normality.

6.3.1 Work related to the normality of the first part of the Wnt signaling pathway dataset

The assumption of normality is used in different ways in the literature. Some researchers assume the Normality of the data for the dataset (Geiger & Heckerman 1994a). Other papers show that discretising the dataset is the best choice for dealing with the data, since a continuous dataset will allow us to find only the relationships between any child and its parents that are close to linear (Friedman 2004). However, (Ferrazzi et al. 2007) states that it is possible to map the child to its parents in a non-linear relationship, by using non-linear functions. Other research mentions that, due to the nature of gene expression profile experiments and the possibility of errors, the log transformed datasets satisfy the multivariate Normal distribution assumption (Waddell et al. 2000; Wu et al. 2003). Another way of not violating the assumption is (Parrish et al. 2009) which suggests using a family of normalising transformations, from which a transformation is selected for each gene that satisfies the assumption. However, it does not necessarily follow that, if each gene has been drawn from a Normal distribution, the genes together follow a multivariate Normal distribution, as will be shown below.

6.3.2 Normality test on the first part of the Wnt signaling pathway dataset

After reviewing the literature, it was found that different researchers have different opinions. It might be that each gene expression dataset yields different results when analysed, due to the normalisation process and the systematic differences resulting from such experiments. Therefore, we found it important to investigate the assumption of normality for the dataset set rather than, for example, assuming normality, without testing it and then linking the results with what has already been found in the literature.

A lot of statistical tests can be used to assess the assumption of Normality for a given dataset. A commonly used test is the Shapiro-Wilk test, which tests whether a sample is drawn from a Normal distribution (a null hypothesis). Another way of testing the Normality of each gene is to use a normal probability plot, which compares the cumulative distribution of the data values

W = 0.5349, p-value = 8.416e-08

Table 6.2: The Multivariate Shapiro-Wilk test.

for a variable with the cumulative distribution of a Normal distribution. To test the normality of the dataset, we used the multivariate Shapiro-Wilk test. Despite the small sample size, the result obtained was encouraging. All genes apart from one (SFRP5) experimentally passed the test shown in Figure 6.1, and 6.2 .

However, when the multivariate Shapiro-Wilk test was performed on the dataset, it failed. The result of the multivariate Shapiro-Wilk test is shown in Table 6.2.

6.3.3 Discussion

The result of the multivariate Shapiro-Wilk test indicated that the dataset was not drawn from a multivariate normal distribution. However, if we look at the Normal and multivariate normal analysis in Figure 6.1, 6.2 and Table 6.2, we can see that three results were obtained:

1. When each gene was assessed for Normality, only one of them, SFRP5, failed the test. We therefore accept that each gene was drawn from a Normal distribution.

2. When we assessed the entire dataset, the test showed a small p-value. Therefore, the dataset was not drawn from a multivariate Normal distribution.

3. By looking at the statistical value (W) of the multivariate Shapiro-Wilk test in Table 6.2, it does not have a small value, but is > 0.5 and we know that the closer it is to one, the more Normal is the dataset (Shapiro & Wilk 1965; Dudley 2010).

Based on the results above and the small samples we have (the most important factor when a statistical test is used), it is acceptable to say that the assumption of normality has approximately been assessed as positive even if the entire dataset did not pass the test. It should also be mentioned that, prior to the pre-processing steps at the beginning of this work, a lot of genes could not be improved in terms of their normality, fortunately, the different pre-processing steps removed them from the dataset. Moreover, it is widely accepted in the community to assume normality for gene expression datasets, as long as the hope is to find useful results which will help biologists to understand how complicated cellular systems work.

6.4 Variable Selection Methods for Learning a Graph

After accepting that the dataset was approximately drawn from a multivariate Normal distribution, the problem of learning a graph is essentially based on learning the best predictors of each gene based on gene expression values. In this section, different methods will be examined on finding the best co-expressed genes(predictors) for each gene(dependent/child) and then a final method for learning a graphical model from a sparse gene expression dataset(13 samples) will be

Figure 6.1: Shapiro-test and normal probability plot for the 13 cancer samples(stem cell) (1).

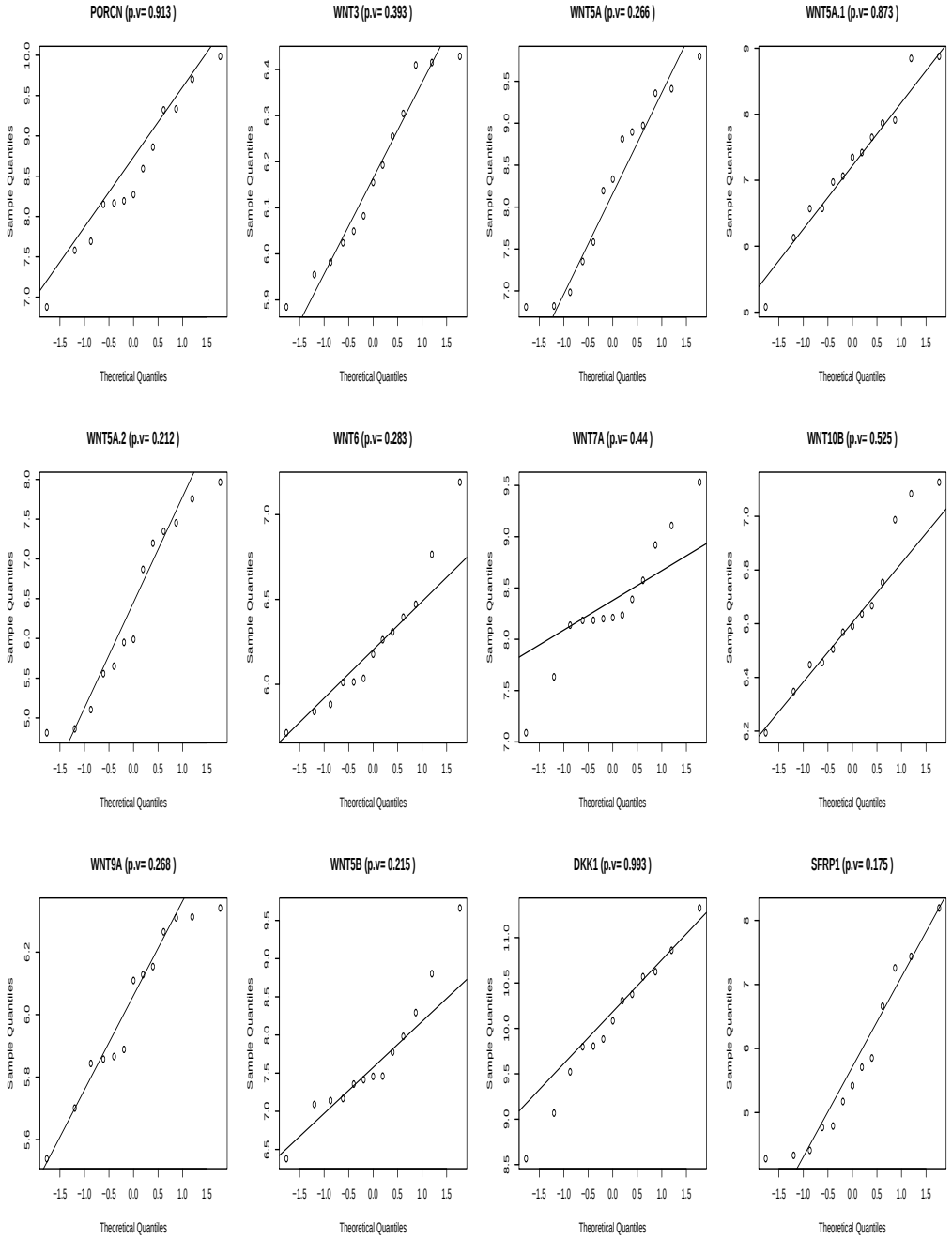
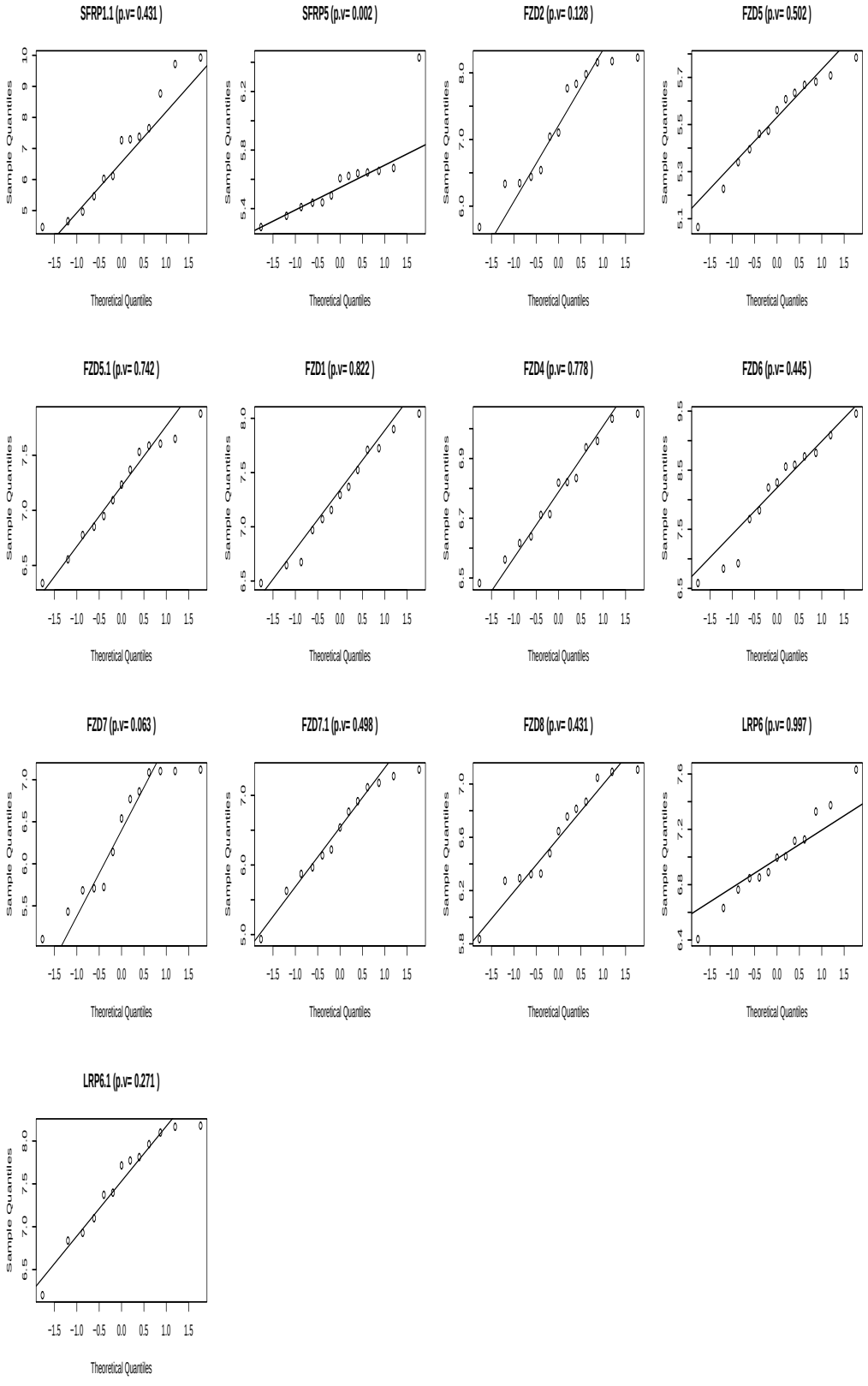


Figure 6.2: Shapiro-test and normal probability plot for the 13 cancer samples(stem cell) (2).



proposed. Research in this area states that there is no progress in subset selection in regression as long as the estimation of regression coefficients is concerned (Hesterberg et al. 2008; Miller 2008). Therefore, we examine different methods for variable selection and parameter estimation as well as examining the advantages and disadvantages of each method.

6.4.1 Learning a co-expression graph

As the problem of learning a graph is now based on a multivariate Normal distribution assumption, we can represent the dependencies in a linear regression relationship. We can also think about the correlation coefficients as a score function for subset selection(6.2.1) in linear regression. If a vector $\mathbf{X}$ is normally distributed $\sim N_p(\mu, \Sigma)$, then any two variables $x1, x2$ in $\mathbf{X}$ that are uncorrelated $r(x1, x2) = 0.0$,are independent $x1 \perp x2$.
Therefore, the correlation coefficient can be used here to determine a small subset of parents that have the most co-expressed relationships with the child. If a parent has a high correlation with a child, we conclude that this parent has a strong linear relationship and therefore is a good predictor for the child. Thus, finding the best subset of parents will be controlled by a correlation coefficient (Markowetz & Spang 2007) and parents with high correlations with the child will be included in the model. The literature shows that, if the dataset has fewer than 30 samples, using the correlation is not enough and it is important to find how significant the correlation is (Hair et al. 1998). For that reason, we use a statistical test, with correlation coefficients as a score function, to choose the best parents for each gene since we have only 13 samples . A t-test implemented in the R package is used and the p-value is set to 0.05. The null hypothesis in the test is that the correlation $r(x1, x2) = 0.0$ and therefore the alternative hypothesis $r(x1, x2) \neq 0.0$. After the test was done, each child had a set of parents that are significantly high correlated with it.

In learning a graph using correlation coefficients, we put some natural constraints on the genes/gene families (components) in the dataset using the same prior knowledge used in Chapter 5 from the first part of the Wnt signaling pathway in KEGG, shown in Figure 4.8 as follows:

1. The PORCN component indirectly affects the WNT component.
2. The Dkk component inhibits the LRP65/6 component.
3. The FRP component inhibits the WNT component.
4. The WNT component activates the Frizzled component.
5. The Frizzled component activates the LRP6/5 component.

Therefore, all possible subsets of parents for each gene in the search space are going to be relatively small. For example, when searching for possible parents for WNT3, the parents that

are allowed to be in the subsearch space for gene WNT3 come from the PORCN component and the WNT component (the genes come from the same family of WNT3, since the target *initially* is also to find how the genes in each component interact with each other). Thus, based on the prior knowledge from the KEGG pathway, the genes that are highly correlated will be represented by a directed arrow, if KEGG shows any relationship. However, if two genes are highly correlated but KEGG does not show any prior knowledge for them , then the relationship will be represented by an undirected arrow.

As we have emphasised in Section 1.3.2 that the work in this thesis is based on transcriptional interactions but the relationships in KEGG mostly are based on protein-protein interaction. Therefore, the resultant graph holds relationship between nodes that represent gene expression values which can be an abstract level to understand how protein-protein interaction happen in the cell.

The network resulting from the experiment is shown in figure 6.3.

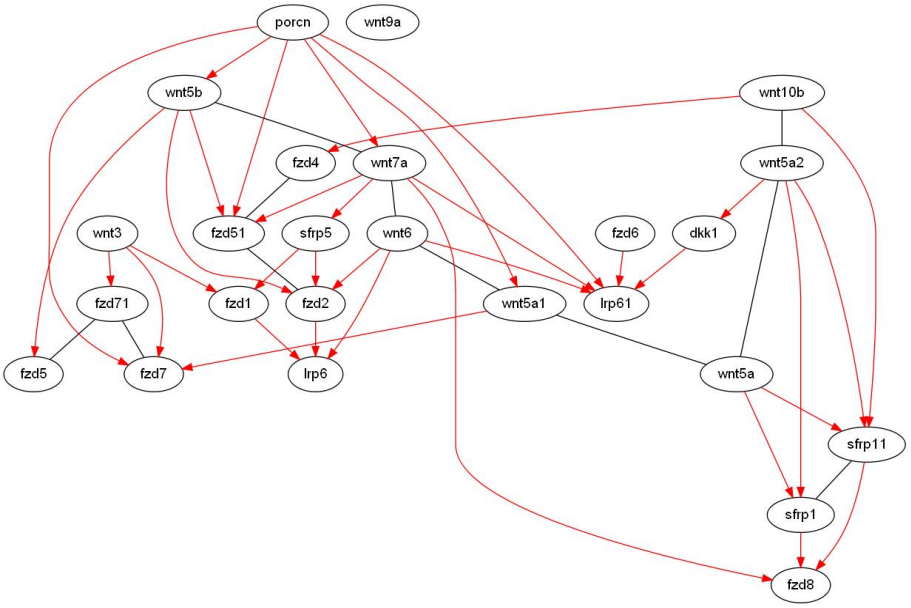


Figure 6.3: Coexpression Network using correlation coefficients and t-test.

6.4.1.1 Discussion

The network in Figure 6.3 shows the undirected and directed relationships between genes based on gene expression values and the prior knowledge retrieved from KEGG. The score function used here has restricted the child to having the best parents in the dataset, which is effective in terms of reducing the variance, and minimizing the bias at the same time. However, doing so might lead the learning to miss some of the parents. To illustrate this, consider the fzd8 node in

Figure 6.3. Based on the pair-wise correlation, sfrp1, sfrp11 and wnt7a are the parents of fzd8 and the residual sum of squares (RSS) for this sub-model is 0.4080789. Suppose that we added one more parent from the set of parents which were removed by the significant test, say wnt5a and wnt5a.1. The error for the sub-model is going to be smaller. Thus, generally, the correlation coefficient can find that X1 is a parent of X2, while X3 is not, but cannot find that X1 and X3 together might be parents of X2.

To illustrate the general concept here we take Exclusive-OR (XOR) logical table in 6.3 as an example. In a XOR logical table, 'A' values alone cannot determine the output values and also 'B' values alone cannot determine the output values. The only way to get the right logical value is to take A and B together as an input to get the correct output.

Table 6.3: Exclusive-OR logical table.

Input A	Input B	Output
0	0	0
0	1	1
1	0	1
1	1	0

Another explanation of the drawbacks encountered in co-expression network is that adding parents to reduce RSS is not a good choice either, since the model might suffer from overfitting. Therefore, learning a graph using only correlation coefficients is not enough to determine the best subset of co-expressed parents for each gene.

6.4.2 Learning a graph based on the penalised goodness-of-fit

After we have addressed the potential problems arising from using RSS and correlation as a measure of how well the model fits the data, the next step is to consider all possible parents from KEGG, for the child and then choose the best subset among them, using different criteria. This is done by constructing all the possible candidates in the search space exhaustively and then using an additive score function that penalises the more complex models. In this work the score functions that we use are a version of Akaike Information Criterion (AIC) (6.4) and Bayesian information criteria (BIC) (6.5). AIC is given by:

$$\mathrm{AIC} = n \log(\mathrm{RSS}/n) + 2p \tag{6.4}$$

Where $2p$ discourages the overfitting ($p = length(\beta_i) \neq 0.0$), p is the number of non-zero coefficients. AIC tries to find the model that best explains the data, with a minimum of free parameters. The best subset of parents for each gene returned by AIC has the smallest AIC.

BIC is like AIC in that it is possible when estimating a model's parameters using maximum likelihood estimation to increase the likelihood by adding more parameters, which could lead to overfitting. BIC resolves this problem by adding a penalty to the number of parameters in the model and the penalty for the complexity of the model is stronger than AIC. BIC is given by :

$$\mathrm{BIC} = n\log(\mathrm{RSS}/n) + p\log(n) \tag{6.5}$$

The best subset of parents is the one with the smallest BIC. BIC also tends to penalise complex models more heavily, giving preference to simpler models in the search space (Hastie et al. 2009). The potential problem, when no prior knowledge is used in the search space to find a reasonable model, is that the number of models in the search space is super-exponential in the number of observations (genes) and the possible subsets of parents grows very quickly with the number of possible parents. Therefore, prior knowledge from KEGG is an important factor to reduce the complexity of the search space. Several search algorithms have been proposed and in this study we will examine how AIC and BIC are introduced in stepwise regression, all subset regression, ridge regression and lasso estimate search strategies, along with the advantages and disadvantages of each search algorithm.

6.4.2.1 Adjusted R^2 score function for subset selection

Adjusted R^2 is a generalised version of coefficient of determination R^2. It is given as follows :

$$Adj - R^2 = 1 - \frac{SSE/(n-p)}{TSS/(n-1)} \tag{6.6}$$

Where :

$$TSS = SSE + SSR \tag{6.7}$$

and,

TSS: the total sum of squares=$\sum_{i=1}^{n}(y_i - \overline{y})^2$.

SSE: the sum of squared errors=$\sum_{i=1}^{n}(y_i - \hat{y_i})^2$.

SSR: the sum of squares regression=$\sum_{i=1}^{n}(\hat{y_i} - \overline{y})^2$, where y_i is the actual value , $\overline{y}$ is the mean, and $\hat{y_i}$ is the predicted value.

In case that the best subset of parents for a particular gene is $\emptyset$, the mean is used ($\overline{y}$) , therefore $\hat{y} = \overline{y}$ in SSR.

Thus : $\forall \hat{y_i} = \overline{y}$

$$SSR = \sum_{i=1}^{n} (\hat{y_i} - \overline{y})^2 = 0.0, \rightarrow TSS = SSE$$

and, as a result

$$Adj - R^2 = 1 - 1 = 0$$

This means that Adjusted R^2 ignores the fact that some genes without parents might have a higher score than if some parents are included.

6.4.2.2 Stepwise regression

The first attempt for variable selection based on penalised goodness-of-fit is stepwise regression. The search begins either by assuming that all parents are possible for a particular gene, *backward stepwise search*, and then removing each predictor in turn until the score function does not change or becomes worse; or it assumes that the gene does not have any parents, and justifies this by adding each predictor in turn, until the score function does not change or becomes worse *Forward stepwise search*. The search can also add and remove predictors at the same time *Forward-backward stepwise search* (Efroymson 1960). However, each stepwise search-score algorithm returns a different model and all are a form of greedy search. Greedy search does not guarantee to find the global optimal model, since the best predictor is chosen regardless of the future effect and therefore the result of the search in stepwise search algorithms does not guarantee optimal results. Greedy search generally lacks the concept of exploration before the exploitation, which means that even if the search space is complete (the best optimal model exists) the greedy search might not visit all the models and therefore misses the most consistent model. Therefore, in the next section we use a complete search method "all-subset selection" using a branch-and-bound algorithm that guarantees to visit each candidate in the search space.

6.4.2.3 All-subset selection

All-subset selection is an expanded version of stepwise regression in which all models in the search space can be visited; prior knowledge from KEGG will restrict the search space and make the exploration manageable. To increase the effectiveness of the exhaustive search, a branch-and-bound algorithm (Land & Doig 1960) that makes a complete search faster than a brute exhaustive search is used.

In this search method, one step more has been taken before the search starts. Since the AIC and BIC score functions rely heavily on RSS, for each family, the set of parents are examined by RSS. If a set of parents in the regression model leads to RSS = 0.0, then the AIC/BIC score is going to return $= -\infty$, which in turn gives unbeatable result. This is because the size of the used dataset (13 samples, 25 observations) is likely to lead to overfitting. Therefore, each family is examined before the search starts. If RSS = 0.0 the correlation coefficient (r) is used to drop

parents with a small correlation with the child gene. The resulting models from search-score-AIC and search-score-BIC are shown in Figures 6.4 and 6.5 respectively.

6.4.2.4 Discussion

The model chosen by BIC is sparser than the one by AIC, which is sometimes preferable, since we would like to see the global picture of the co-expression between genes. This in turn helps to read the graph more easily. However, there is no clear choice between AIC and BIC in general, except for the following: given a search space that includes the true model, the probability that BIC will get the correct model is 1.0 as $N \rightarrow \infty$. However, as $N \rightarrow \infty$, AIC tends to choose more complex models. On the other hand, for finite samples, BIC often chooses models that are simple, because of its penalty for complexity (Hastie et al. 2009).

The final result that we obtained looks at which genes from one component co-express or react with which in another component. However, the results also show how genes in each gene family react with each other. We observed that some genes in each gene family is dependent on each other, where the undirected arrow is invoked, or one dependent on an another, where a directed arrow is used. Therefore, the network contains cyclic families, which in turn leads to a set of graphical models called *dependency networks* (Heckerman et al. 2000). The prior knowledge that is used shows that the graph we have is a collection of directed and undirected relationships. Therefore, another view that needs more investigation is that the graphs in Figures 6.4 and 6.5 have directed co-expressed relationships between blocks, undirected/directed or both between genes inside each block, and acyclic relationships between blocks. This set of graphical models is called Chain Graphs (Roverato & Rocca 2006; Aloraini et al. 2010).

However, the small dataset used in the experiments will not be a great help to investigate dependency and chains graphs, because if we seek the best possible parents for a gene and we include its family, a gene might have more than 13 parents from which to search the best combination. For example, there are 19 possible parents for FZD8, if the family of Frizzled is included in the search, and 11 possible parents for FZD8 without its family. This is in cancer samples, but in non-cancer samples the situation is worse since there are only 6 samples from which to learn. Therefore, we paid attention to learning the directed relationships between families rather than between families and within gene families. For example, we are keen to see which gene from the WNT family co-expresses with which gene in the Frizzled family which might lead to understand how protein-protein interaction between these two families happens. However, we will not look at how Frizzled genes might work together, as this needs bigger datasets, and even with the new settings, there are still some problems.

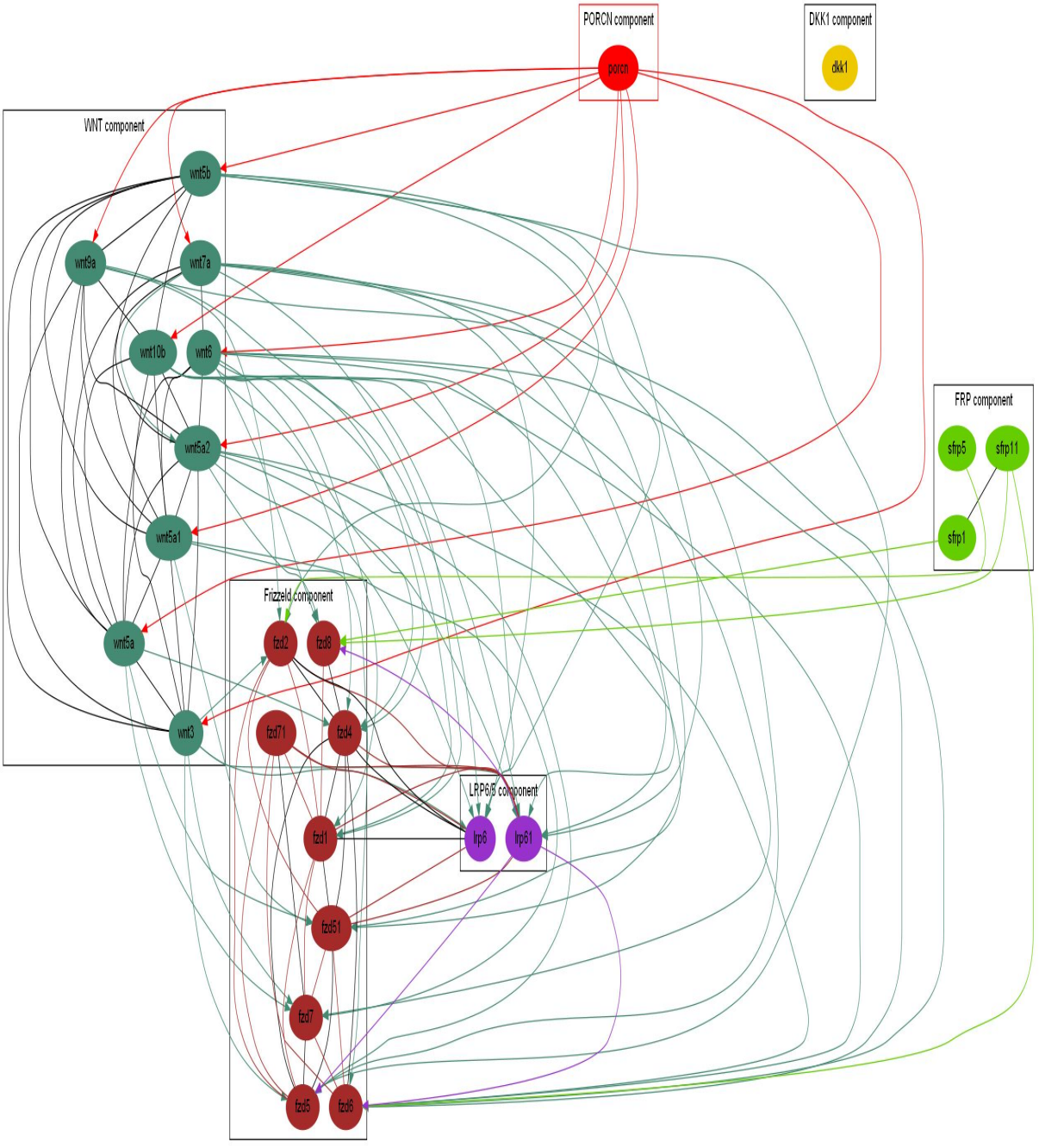


Figure 6.4: The resultant graph from the search-score (AIC) method in normal linear regression.

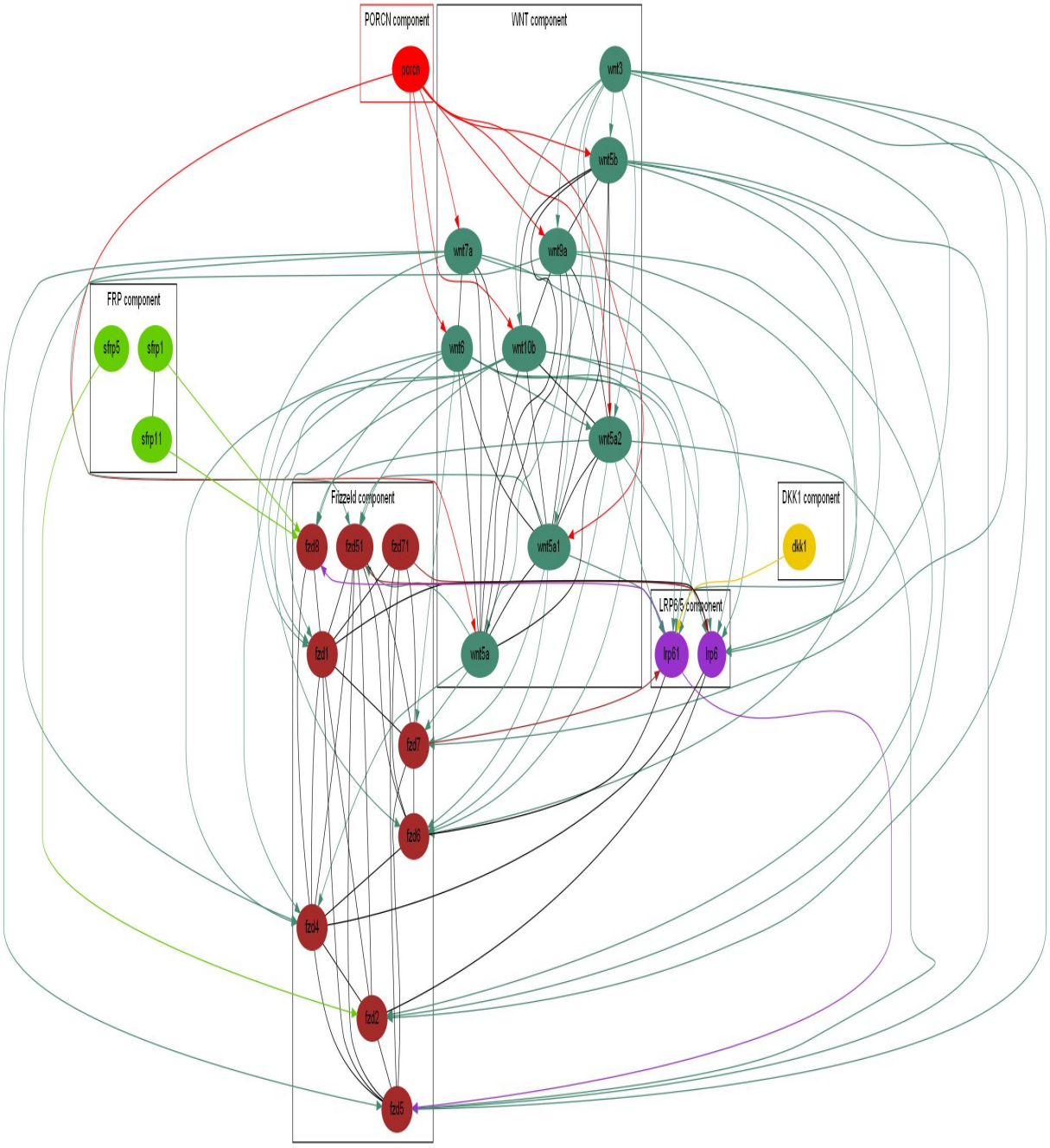


Figure 6.5: The resultant graph from the search-score (BIC) method in normal linear regression.

A substantial problem in all-subset selection used in this study is that it focuses on variable selection but not on the coefficients. The current approach is concerned about variable selection and only estimates the coefficients using least squares/Maximum Likelihood after the predictors have been determined. The advantage of using more continuous methods to estimate the best predictors, along with the coefficients, is that they will result in fewer predictors for each gene. It will also lead to the stability of the model in which a small change in the data will not result in large changes in the subset of predictors. In the following sections, we will introduce shrinkage methods that take *extra* care with parameter estimation, and proceed to show how the learning is improved in two ways:

- When the resultant graphs are evaluated before and after the shrinkage methods are used.
- When the parents from the same gene family are invoked to learn the best subset of parents for each gene and when only the predictors from other families are used.

6.5 Shrinkage Methods for Learning a Graph

Variable selection by stepwise regression considers either adding or removing predictors from the regression model and this is beneficial, since the resultant model is usually interpretable. The result from such a greedy search is usually unstable, as any small change in the data might cause one variable to be chosen instead of another (Hesterberg et al. 2008). All subset regression on the other hand, avoids this problem, because it considers each model in the search space and returns the best model found. However, all-subset regression only takes care to choose the best predictors, after that, it estimates the coefficients using the standard least squares. Shrinkage methods are more continuous, and overcome the problem of exhibiting a high variance and the increase of prediction errors when a discrete approach is used.

6.5.1 Ridge regression

Ridge regression is a shrinkage method that takes $extra$ care in adding parents or predictors and estimating the coefficients in a more robust way (Tikhonov & Arsenin 1977). Ridge regression shrinks the regression coefficients by imposing a penalty on their size. Therefore, the ridge coefficients minimise a penalised residual sum of squares.

$$\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{i=1}^{p} \beta_j^2 \right\}$$

Where $\lambda \geq 0$ is a complexity parameter that controls the amount of shrinkage. If $\lambda = 1.0$, then no parents are added. In contrast, if $\lambda = 0.0$ ridge regression returns an ordinary least squares. No penalty is applied to the intercept (β_0) and β_0 equals $\overline{y} = \sum_1^N y_i/n$ and also x_{ij} is centred *reparametrization*. An equivalent way to write the ridge problem is:

$$\hat{\beta}^{ridge} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2, subject\ to \sum_{j=1}^{p} \beta_j^2 \leq s$$

If $s = 0.0$, no parents are added. In contrast, if $s = 1.0$ ridge regression returns an ordinary least squares, where s is a shrinkage factor that controls the size of the βs in the regression equation.

Although Ridge regression carries out the variable selection and least estimate coefficients in a continuous way, simultaneously, often the resultant model includes all possible predictors that are allowed, but typically with smaller coefficients than they would have under ordinary least squares (Hastie et al. 2009). This is because the imposed penalty ($\sum_{i=1}^{p} \beta_j^2$) will shrink coefficients towards zero, but not exactly zero. Therefore, no variable is ever excluded from the model (except when some coefficients cross zero for smaller values of λ).

6.5.2 Lasso

Lasso is a shrinkage method in which many coefficients are 'shrunk' to zero. This is because the penalty for large coefficients in lasso is very severe, being the sum of the absolute values of the regression coefficients $\sum_{j=1}^{p} |\beta_j|$ (in contrast to ridge regression $\hat{\beta}^{ridge}$ where the penalty is less strict $\sum_{j=1}^{p} \beta_j^2$). The lasso estimate $\hat{\beta}^{lasso}$ for the regression coefficients for a particular complexity parameter λ is:

$$\hat{\beta}^{lasso} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{i=1}^{p} |\beta_j| \right\} \tag{6.8}$$

or equivalently, (where s is determined by λ):

$$\hat{\beta}^{lasso} = \underset{\beta}{\operatorname{argmin}} \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2, \text{ subject to } \sum_{j=1}^{p} |\beta_j| \leq s \tag{6.9}$$

No penalty is applied to the intercept (β_0) and $\beta_0 = \overline{y} = \sum_1^N y_i/n$, and the x_{ij} are centred. Figure 6.6 shows how the lasso estimate for each coefficient varies for candidate parents for gene FZD7, as the complexity parameter varies from 0.0 to 1.0.

6.5.2.1 Related work

The lasso estimate has been used in a lot of research, as a technique for regularisation, variable selection, or covariance selection for high-dimensional data, in which the number of predictor variables is much larger than the number of samples($p \gg n$). (Friedman et al. 2007) shows the use of the *glasso* method, graphical lasso, in which an imposed penalty is introduced to the inverse matrix $\sum^{-1}$ to increase its sparseness. The solution of glasso shows that $i \perp j$ given all other variables if $i, j^{th} = 0.0$ in $\sum^{-1}$. (Peng et al. 2009) proposed a computationally efficient

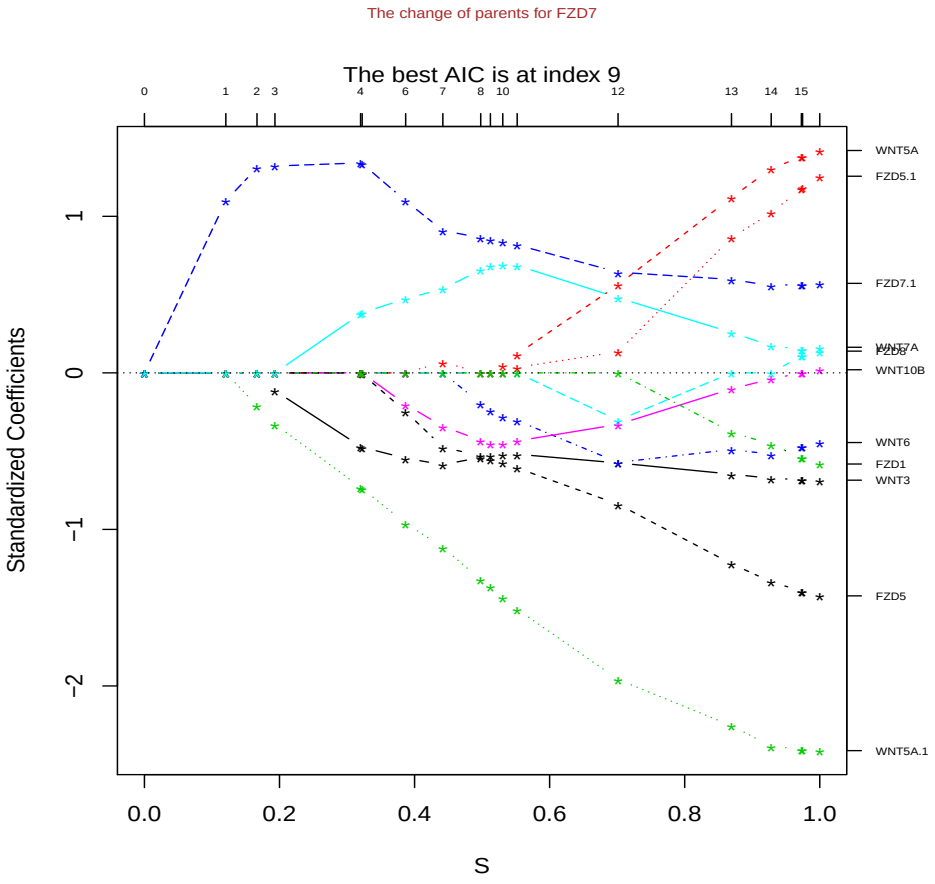


Figure 6.6: This Figure shows how the lasso estimate for each coefficient varies for candidate parents for gene FZD7, as the complexity parameter varies from 0.0 to 1.0.

approach for selecting non-zero partial correlations using **L1**-penalty under $p \gg n$ called Sparse PArtial Correlation Estimation (*space*) where most variable pairs are conditionally independent . The advantages of *space* over *glasso* is that in *space* prior knowledge is used in form of weighted nodes according to their importance. Also, the complexity of *space* is min($O(np^2, O(p)^3)$ while in *glasso* is $O(p^3)$ and therefore, the *space* is much faster than *glasso*. Two drawbacks are observed from the previous studies when lasso estimate is used. Firstly, the same imposed penalty for all nodes is used. However, each node might have different set of possible parents and therefore choosing the best subset of parents for each node using the same penalty is not an optimal choice as the nodes with more possible subset of parents might need *stronger* penalty than the nodes with few possible parents. Secondly, lasso estimate in the mentioned papers has been found leads to undirected graphs and usually no natural background knowledge is used. Undirected graphs are not always useful specially when the inference aims to discover the causality between variables in a graph. As far as ascertainable, the only work used L1-regularization methods for a directed acyclic graph is by (Niculescu-Mizil & Murphy 2007) using *big* datasets.

In our work, a more constrained work is achieved using lasso estimate using small dataset. The lasso estimate solution will be more constrained based on a natural prior knowledge from KEGG database and feature ranking selection method and hence the resultant graph will be shown in the context of directed relationships. Also, in addition to the traditional cross validation method to choose the most probable regularizer (s), we show how AIC , and BIC score functions are used to choose s when small dataset and natural prior knowledge are used. Moreover, we will show that the lasso-AIC/BIC estimate solution corresponds to a global optimal solution even when a small-scale dataset is used. Thus , it is guaranteed to choose the best subset of parents for each gene found in the search space.

6.5.2.2 Using lasso and penalised goodness-of-fit for learning a graph

This section shows how AIC and BIC are used to evaluate the goodness-of-fit for each model using the same data as used previously in the first part of the Wnt signaling pathway (stem cell , 13 cancer samples). For each gene, the best subset of parents will be learnt and evaluated along with the coefficients being estimated using *lasso*. Using lasso to estimate the coefficients will cause some of the coefficients to be zero. Therefore, the number of predictors for each gene will be relatively small. As it is always preferable to inject prior knowledge into the search space, to minimise the complexity of the search, we put some natural constraints on the genes in the dataset from KEGG, as before. In lasso, choosing the best model is subject to choosing the best value for the tuning parameter(s in this case). The normal way is to use leave-one-out cross-validation (LOOCV) to choose the best value of s. In this work, AIC and BIC score functions are also used to choose the best value for s and then a comparison between the three methods will be shown.

6.5.2.3 The best value of (s) using AIC and BIC

The best subset of parents for each gene is going to be chosen according to the most appropriate value of the tuning parameter s (the regularisation parameter) which is chosen by the score function: AIC or BIC. This is done by scoring all models that are returned by different values of s. The model with the smallest AIC or BIC will be chosen as the best model and indirectly this shows the best value of s. One more step was taken before the search started, since AIC and BIC score functions rely heavily on the residual sum of squares errors (RSS), again if a set of parents in the regression model leads to RSS = 0.0, then the AIC/BIC score is going to return $= -\infty$, which in turn gives a unbeatable score. This is because the size of the used dataset (13 samples, 25 observation) is likely to lead to overfitting. Therefore, each family is examined before the search starts and the correlation coefficient (r) is used to drop parents with a small correlation with the child gene if RSS = 0.0. Figures 6.7 and 6.8 show the best value of s determined by AIC and BIC for the best subset of parents for genes WNT9A and WNT3,respectively.

6.5.2.4 The optimality of the lasso solution

The lasso solution is derived using a *less greedy search* (Hesterberg et al. 2008). Following (Hesterberg et al. 2008), we have shown experimentally in this work that the lasso solution corresponds to a globally optimal solution, in terms of choosing the best subset of parents, although the search-score algorithm that is used here, does not visit every model, every combination from all possible subset of parents for each gene, in the search space. Figure 6.9(a) shows the lasso solution when the AIC score function is used to score each subset of parents resulting from various tuning parameter values (s). In Figure 6.9(a), AIC does not score every possible subset of parents for each gene resulting from each single value of s (100 values between 0 and 1), but only those subsets of parents that show up (their coefficients $\neq 0.0$) when a particular value of the parameter makes change. For example, the change of parents subsets for WNT5B in lasso in Figure 6.9(a) shows that there are only 15 models (subsets of parents) that have been scored (15 indices as in the 3rd axis (top axis)). To show that the change of the subset of parents for each gene is only for those that have been scored by AIC, we have plotted the path of each single value of s (100 values between 0 and 1) vs the change of parents in Figure 6.9(b) and we can see that the values that do not change the subset of parents in Figure 6.9(b) are discarded by lasso-AIC in Fig 6.9(a). This explains why, sometimes the setting of the problem helps to reduce the complexity of the search, and instead of constructing all possible candidate parents for each value of s and then using AIC or BIC to score each model (100 models) in the search space, we have only scored the models returned from lasso, for example, 19 and 15 models in Figure 6.9(a).

Figure 6.7: The best value of s determined by AIC for WNT9A.

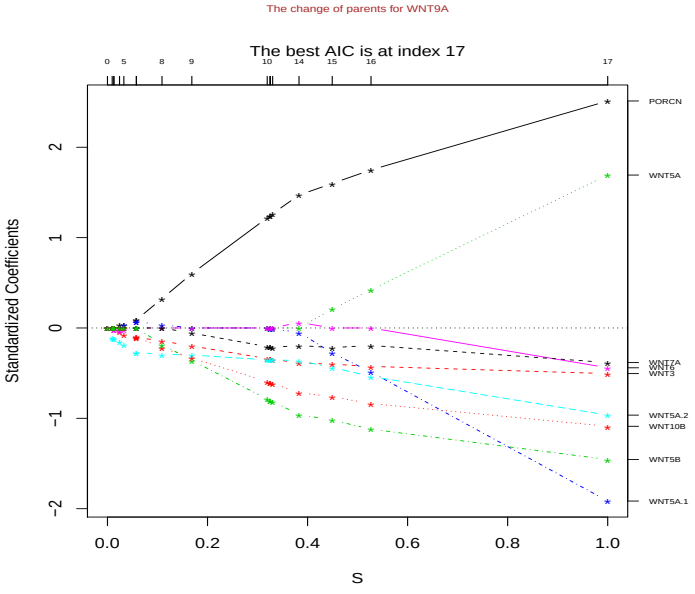


Figure 6.8: The best value of s determined by BIC for WNT3.

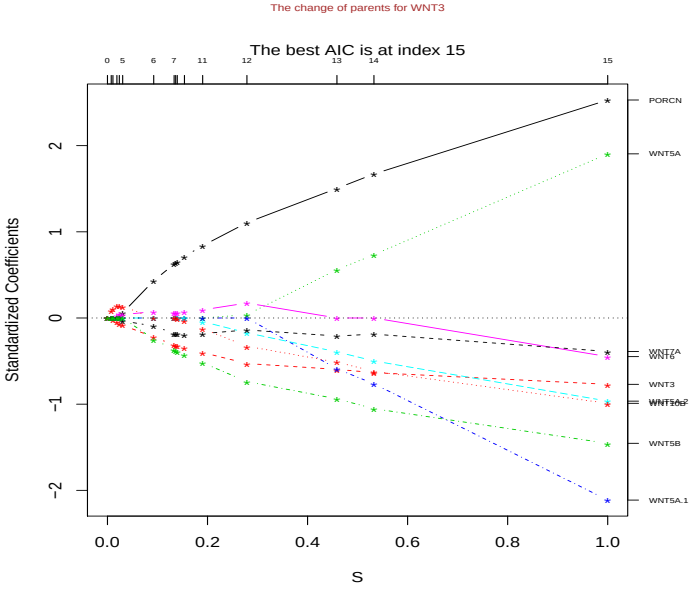
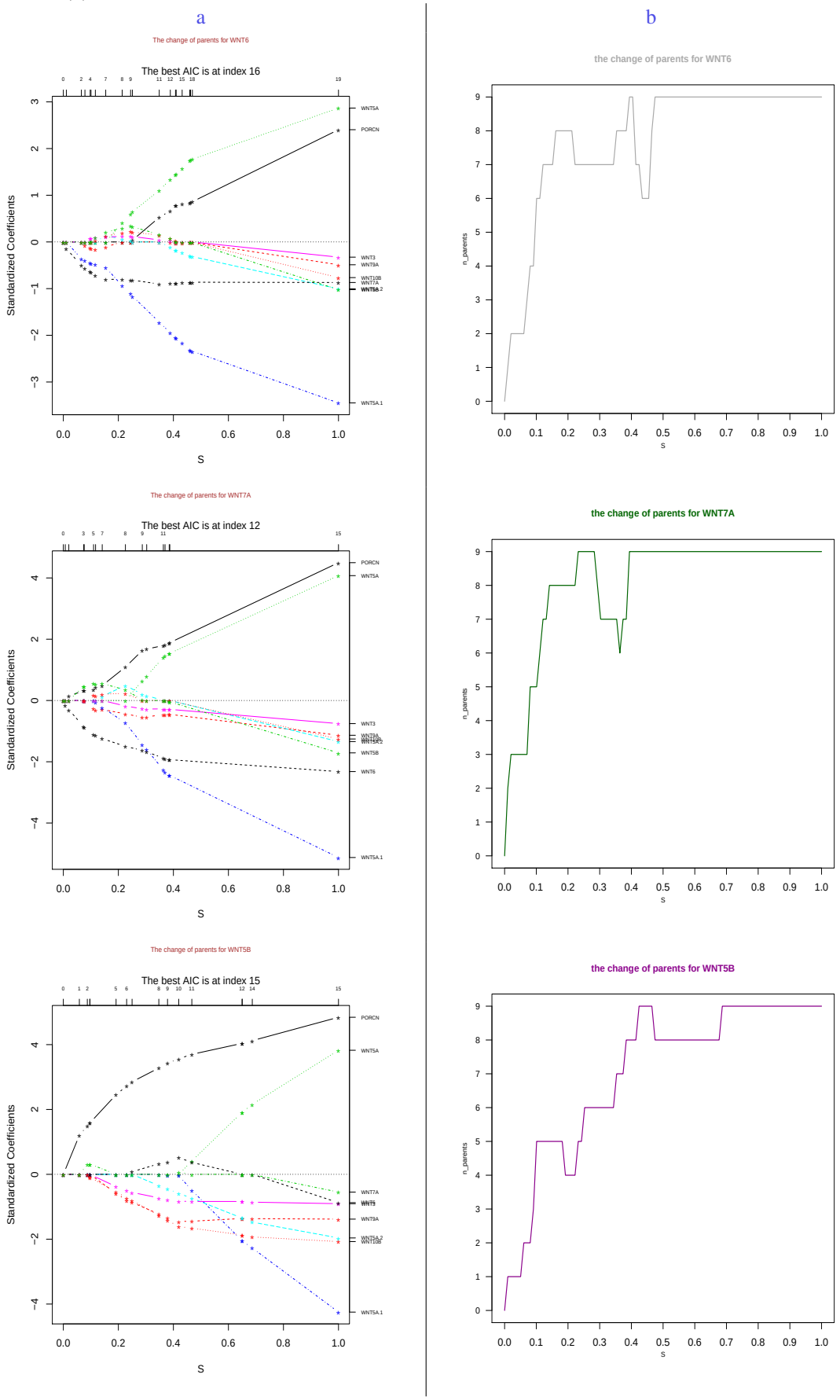


Figure 6.9: The change of parameter values (a) vs the change of parents for each value of the parameter (b).



6.5.2.5 The best value of s using cross-validation

Another way of determining the best value of s is by using cross-validation, in which each value of s is evaluated by leave-one-out cross validation. The mean of the prediction errors for each value of s is used as a score for each value of s. Figure 6.10 shows the best value of the tuning parameter(s) ≈ 0.121 which corresponds to the smallest error 0.17930 from the leave-one-out LOOCV for gene FZD1. The bottom part of 6.10 shows the best parents for FZD1 after the best value of s indicated by LOOCV.

After we have selected how the best value of s is chosen for each subset of parents for each

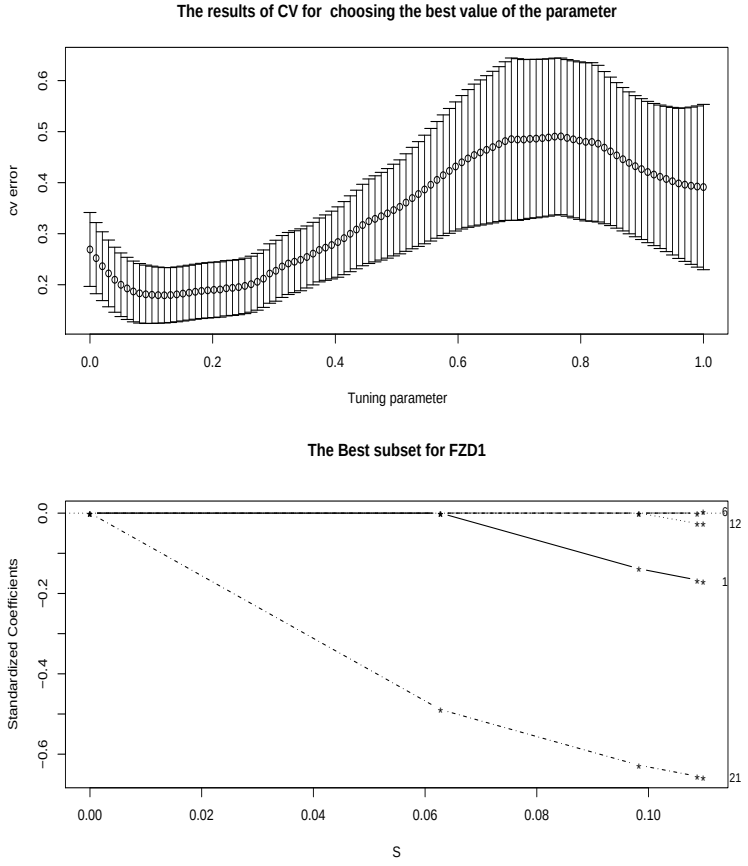


Figure 6.10: The tuning parameter(s) is chosen by LOOCV(top graph) based on the prediction accuracy for each value of s, and then the chosen s value used to find the best subset of parents for FZD1(bottom graph).

gene using AIC, BIC and LOOCV, the results for the three methods, lasso-AIC, lasso-BIC and lasso-LOOCV, are shown in Figures 6.11, 6.12 and 6.13. All the graphs contain directed cycles

and therefore they can all be considered as *dependency networks* (Heckerman et al. 2000). When we allow the genes within a component/gene family to be parents of a gene from the same component/gene family , a cycle can arise. In Figure 6.13 we see that in the WNT family, WNT6 is a good predictor for WNT5B and also WNT5B is a good predictor for WNT6. This is because no prior knowledge is shown in KEGG between genes within the same family.

However, since we are learning from a small dataset, the risk of unreliable results is high. If we are trying to find the best predictors for genes in the Frizzled family for example, we have to consider families that have relationships with Frizzled (11 genes) and genes from within Frizzled family (8 genes) as possible parents, which makes 19 parents in total.

In the next section, we will show new results, based on prior knowledge from KEGG that is between gene families but not within families . We then show how the learning improved when only the relationship between families was considered.

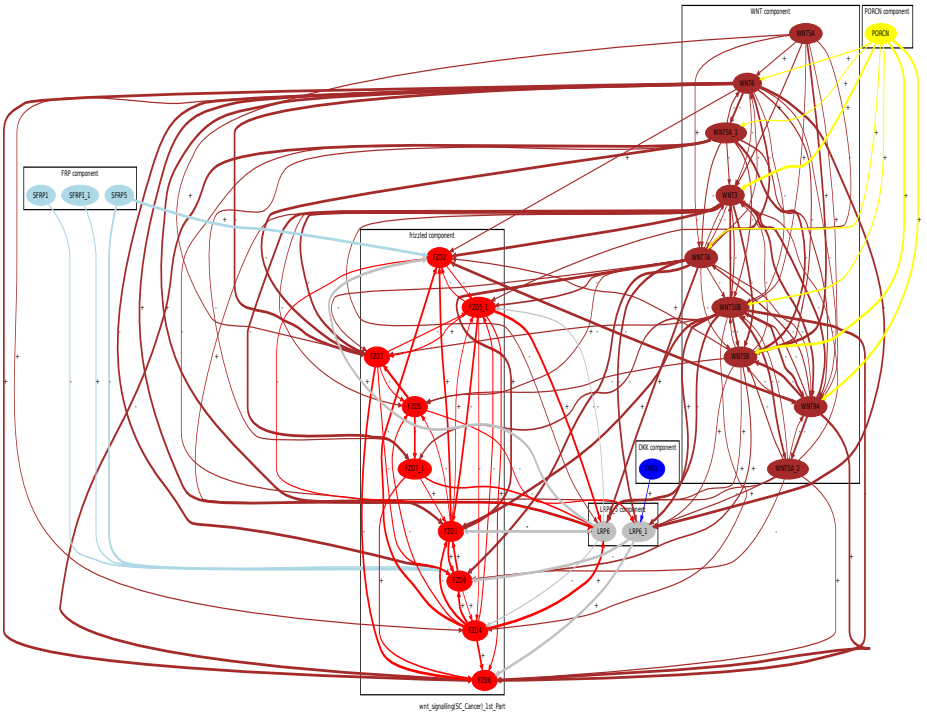


Figure 6.11: The graph resultant from lasso-AIC.

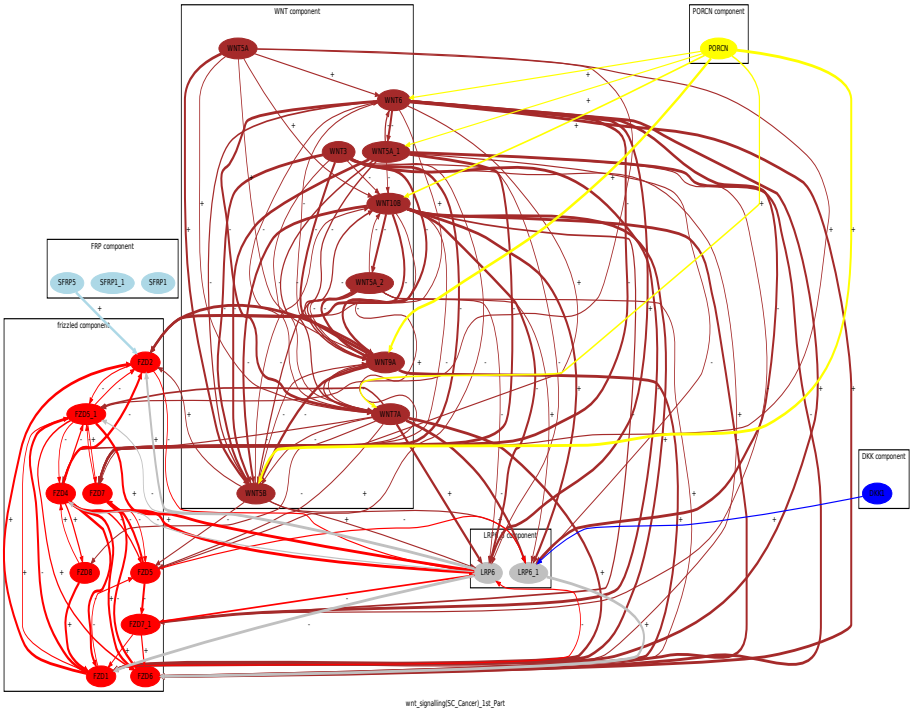


Figure 6.12: The graph resultant from lasso-BIC.

6.6 Learning a Graph Based on More Constraints on the Prior Knowledge

In the previous sections, we have shown how different score functions in the setting of lasso and normal regression were used to learn graphs based on natural prior knowledge gained from KEGG pathways. For each gene, inside each gene family, all subsets of parents come from either a gene family that has a biological effect (inhibition,...,etc.) or from within the same family of the gene under consideration. As shown, the small dataset we have might cause overfitting if a gene has a lot of possible parents, as 19 parents vs 13 samples in Frizzled for example, might remove good parents just because RSS becomes ≈ 0.0. The previous setting of the prior knowledge is important given a *big* dataset, as it reveals more unknown knowledge about the KEGG pathways and the co-expression relationships between genes in order to get more information about how the low-level protein-protein interaction occurs. However, we want to sacrifice important additional knowledge for the sake of getting more accurate results. In other words, we want to minimise the bias and reduce the variance at the same time and hopefully improve the overall prediction accuracy when well established score functions such as AIC and BIC are used to learn subset of parents for each gene in the search space. Therefore, in the next learning step,

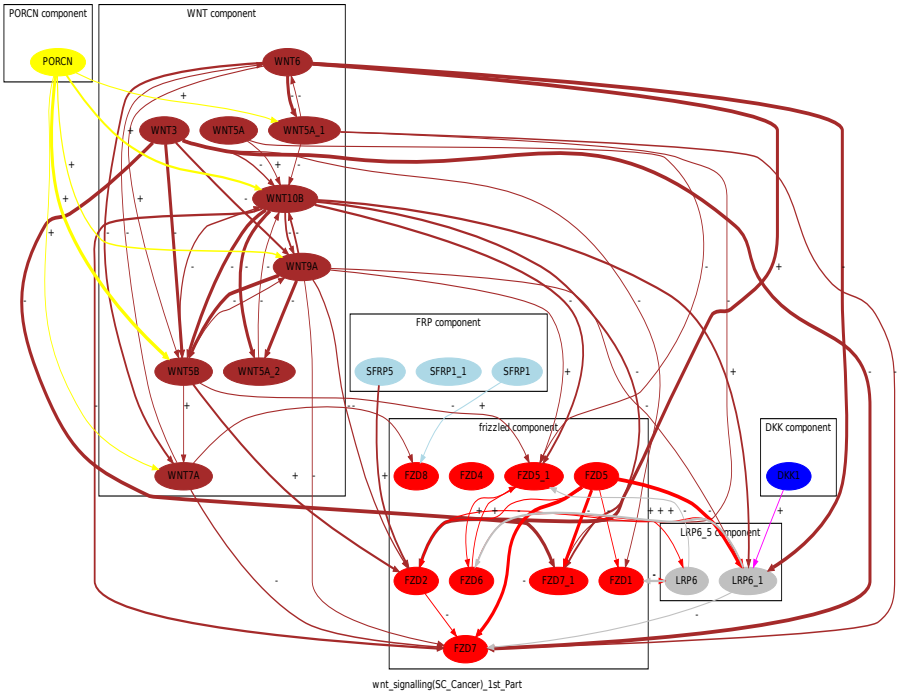


Figure 6.13: The graph resultant from lasso-LOOCV.

we will put more constraints on the prior knowledge from KEGG and use only the relationships between gene families but not within each gene family. This in turn, will reduce all the possible subsets of parents for each gene (typically each gene will not have more than 12 possible parents.). Figures 6.14 and 6.15 show part of the graphs after applying lasso-BIC and lasso-LOOCV respectively when the new imposed constraints are used.

After the new setting for the prior knowledge from KEGG, all graphs (lasso-AIC, lasso-BIC and lasso-LOOCV) are acyclic graphs and therefore a *Bayesian network* can fit here. In the next section, we carry out intensive evaluation experiments to show how the learning improved after the lasso estimate was used and also the reliability of the constrained prior knowledge.

6.7 Evaluation

One usual way of testing the performance of learned models is to use a separate data that has not been used in training or generating the graphs. Another way is by doing physical experiments in the lab using intervention methods (Friedman et al. 2000). However, since holding a spare data for testing or doing physical experiments in the lab is expensive, a robust method called cross-validation is used, in which the data is iteratively split to train and test (K-fold). The average of the $error_{1 \to N}$ is used as a final error for the evaluation. In this work, leave-one-out cross-

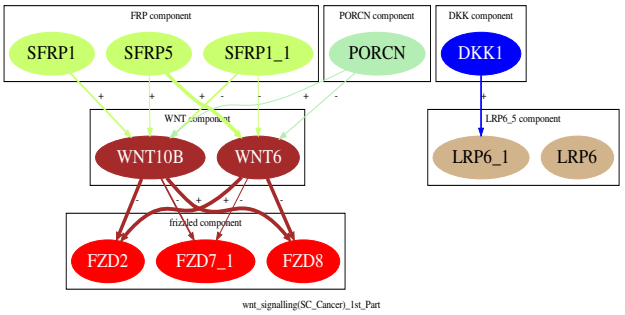


Figure 6.14: The Bayesian network from lasso-BIC, after the new constraints that takes only the relationship between gene families.

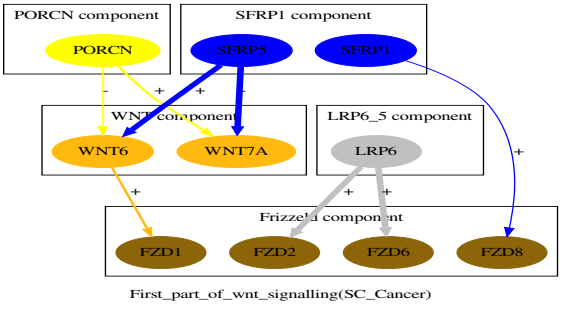


Figure 6.15: The Bayesian network from lasso-LOOCV, after the new constraints that takes only the relationship between gene families.

validation (LOOCV) is used, in which $k = n$. In the normal regression, one sample is reserved each time and AIC and BIC score functions are used in the training, to choose the best subset of parents for each gene. Then, the least estimates coefficients βs for the best subset of parents which have been determined are used to predict on the reserved sample. As we have used leave-one-out-cross-validation, we have tested each gene 13 times and each time, we used 12 samples for training and we tested on the reserved one, recording the prediction error. Finally, the average of $error_{1 \to N}$ is used as a final error for the prediction accuracy. The evaluation is done before the constraints on the prior knowledge from KEGG are made and also after the constraints are used. Figure 6.16 shows the prediction accuracy for AIC and BIC score functions in the two different sets of prior knowledge.

The same procedure is used to evaluate the graphs from lasso-AIC, lasso-BIC and lasso-LOOCV for the two sets of prior knowledge. In the lasso work, LOOCV is used as well as AIC and BIC to choose the best parent each time a sample is reserved. In lasso-LOOCV there are two layers of cross-validations. The first is to choose the value of s (internal LOOCV, Section 6.5.2.5) and the second is when a subset of parents for a gene is evaluated (external LOOCV). Figure 6.17 shows the final prediction accuracy for lasso-AIC, lasso-BIC and lasso-LOOCV for the two different sets of prior knowledge. After conducting all the experiments and by looking at the different final errors from the different methods, we can see that using lasso with AIC, BIC and LOOCV gives a better result than the normal regression before using the constraints on the prior knowledge and also after introducing it. This explains how it is difficult to learn from a small sample size, which is the usual case in gene expression profiles. One important explaination in why lasso-score functions methods have succeeded, because using lasso makes estimating the coefficients in a more robust way than the normal regression. Moreover, the results show that using lasso-AIC, lasso-BIC and lasso-LOOCV with the constraints on the prior knowledge gives a better result than when lasso is used without constraints (Figure 6.17).

6.8 A Comparison Experiment between Lasso-score Functions and the Baseline Method

After showing that lasso with any score function, given constraints in the prior knowledge, works better than the normal regression, and lasso without constraints, the next aim is to evaluate this result using a different method not used in the experiments. One way to test how good the co-expressed parents chosen by the proposed method (the lasso-score functions with constraints on the prior knowledge), is to compare it to a baseline method. The baseline predictive accuracy means that for each gene we reserve $k = i$ and predict on the average of the training expression levels $(n - i)$ for this gene which will show if the subset of parents chosen for that gene by

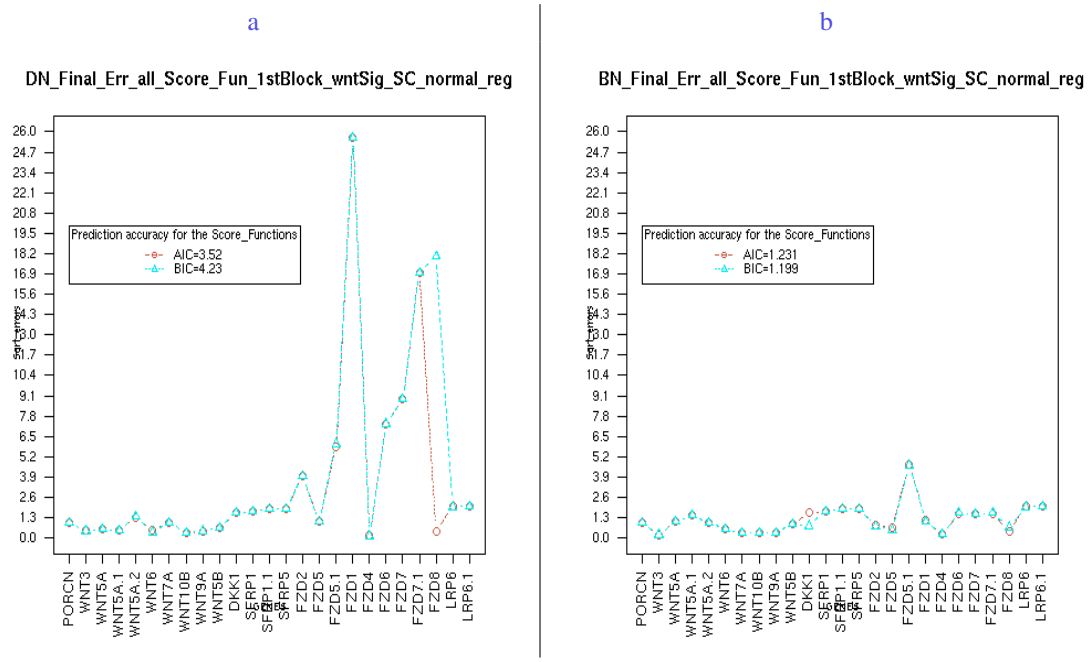


Figure 6.16: The final prediction accuracy for AIC and BIC in the normal regression when genes from the same family and other families are used in the subset of parents(a), and when only the genes from other families are used in the subset of parents(b).

the proposed method in the previous section are meaningful(truly co-expressed parents) or if the gene is better without parents (using mean predictive accuracy is better than any subset of parents). The result in Figure 6.18 shows that the final prediction accuracy based on the average of training gene expression levels obtained from the baseline method is better than any lasso method proposed in Figure 6.17b. This *initially* means that the proposed method using lasso estimate in the previous section performs worse than simply predicting on the average of the training expression levels. This means that the gene is better without any parent and therefore the data and the prior knowledge used are meaningless. Another explanation is that the data used is really small (13 samples) and it is therefore hard to learn from. We have taken the latter assumption and tried to investigate it more in the next section.

6.9 Learning Bayesian Networks Based on Feature Selection and Lasso Estimate

In the previous section, we found that the baseline method suggested that each gene is better without parents. In this section, we take more detailed steps to learn the *most* important parents for each gene and then again compare the results with the baseline. This is because the small dataset we have might need to be more constrained on the subset of parents chosen for each

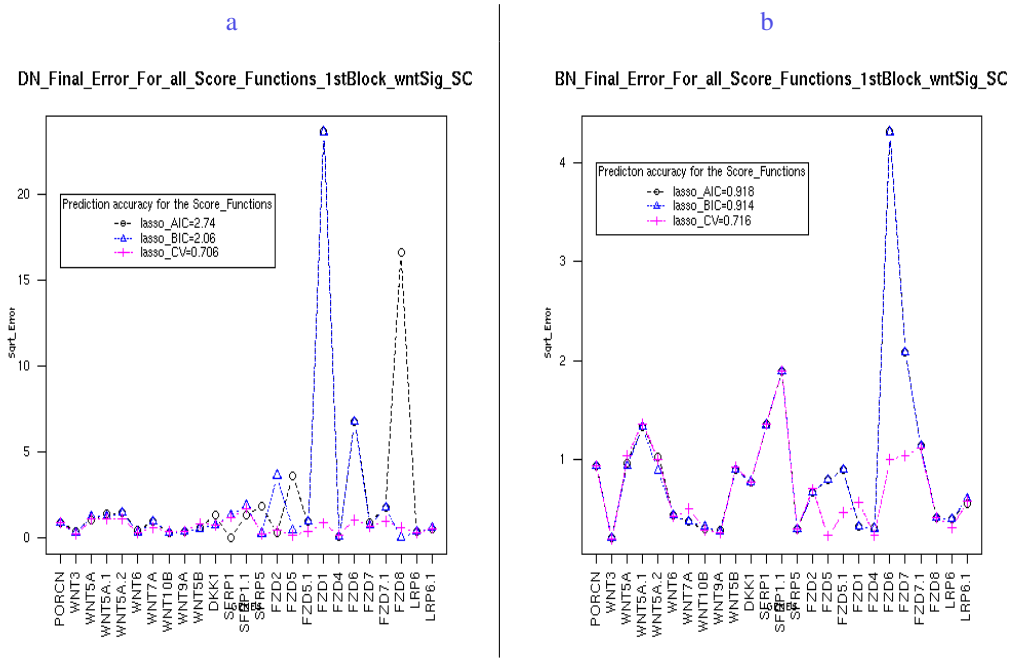


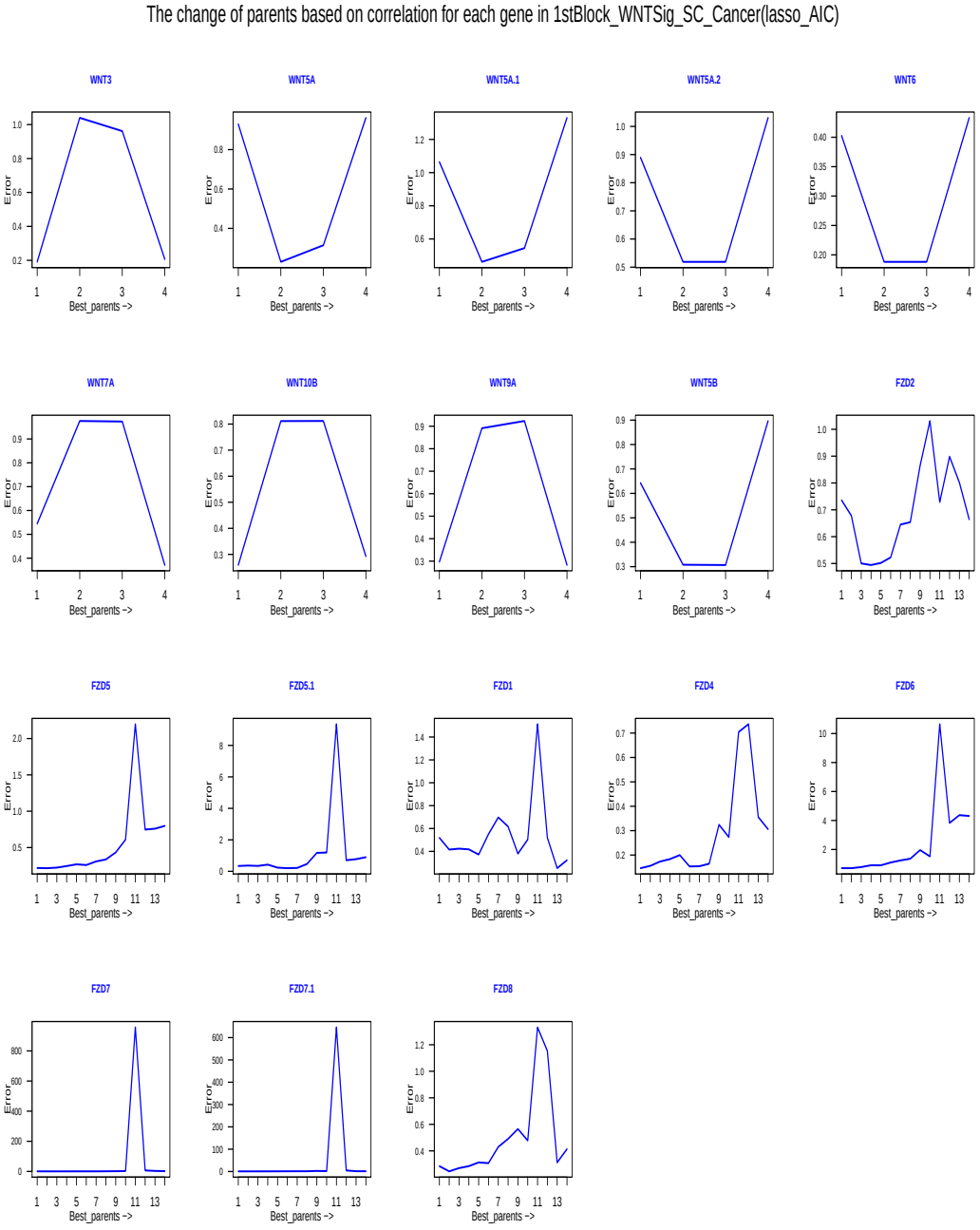
Figure 6.17: The final prediction accuracy for AIC-lasso, BIC-lasso and LOOCV-lasso when genes from the same family and other families are used in the subset of parents(a), and when only the genes from other families are used in the subset of parents(b).

gene. Therefore, the subset of parents for each gene will be ranked according to the correlation coefficients incrementally, from the highest to the lowest, called feature ranking (Guyon 2008). Following this, leave-one-out cross validation (LOOCV) is used to test the prediction error each time we remove a parent from the set of parents, as shown below. The advantage of using the feature selection method here with lasso estimate , rather than the lasso estimate only, is that when a subset of parents is examined using only the lasso estimate, the value of s that is determined by LOOCV is used to choose the best subset of parents from all possible parents one time. However, when feature selection is used we make several choices based on the ranked possible parents. Each time we remove a parent, we test how good the parent is and find the best s for this subset of parents. Then, we remove the second best parent and we repeat the process again to see how good it is and find the best s. We repeat the process until we test only the best parent alone in the model. Therefore, there is a clear advantage for feature ranking over the lasso estimate only, which undertakes this process in one go and using all the possible parents at the same time.

We will start with all parents and each time we remove the lowest parent we test by using LOOCV and recording the prediction error. The test stops when it comes to test only the best parent for this gene. Figure 6.19 shows how the prediction error changes for each gene when a parent is removed from the subset of parents examined by LOOCV in lasso-AIC. Therefore, lasso-AIC,

Figure 6.18: Baseline prediction accuracy which is based on the average of tranining expression values.

Figure 6.19: The prediction accuracy record for each subset of parents for each gene using feature ranking for lasso-AIC.



lasso-BIC and lasso-LOOCV are all run again, but this time with an embedded feature ranking.

Algorithm 1 shows how the methods work generally with the feature ranking method.

Algorithm 1 AIC-lasso, BIC-lasso and LOOCV-lasso with feature ranking

for $i = 1$ **to** $length(Genes)$ **do**
  $Y = GENE[i]$
  $PR = OrderParents(Y, corr(Parents))$
  for $j = 1$ **to** $length(PR)$ **do**
    $SP = SearchSpaceFromlasso(Y, PR)$

    return $BestParents = min[(AICLasso(SP), BICLasso(SP), LOOCVLasso(SP))]$

    return $FinalError = LOOCV(BestParents(AIC), BestParents(BIC), BestParents(LOOCV))$

    $PR = PR[, -j]$
  end for

  return $BestParents(Y, min(FinalError))$
end for

Thus, the experiment was run again, but this time the three methods return the subset of parents for each gene that has the smallest error prediction resultant from a complex and an intensive run using feature ranking. Following this, we compared the three modified methods to the baseline and found that lasso-AIC, lasso-BIC and lasso-LOOCV work better than the baseline method (Figure 6.20).

When we look at this procedure in detail, we can see that there are two layers of feature selections. The first is 'feature ranking' which is based on correlation coefficients. The second is when a value of s is chosen by LOOCV which will return the best features based on the chosen s, as shown in Figure 6.10 Section 6.5.2.5.

After lasso-AIC, lasso-BIC and lasso-LOOCV have beaten the baseline method, the next object is to choose one of these three methods as a final product to learn a refined KEGG pathway. The usual scenario is to test the proposed methods on a bigger dataset as a matter of verification. Therefore, in the next section we will show how the methods work when a bigger dataset is used.

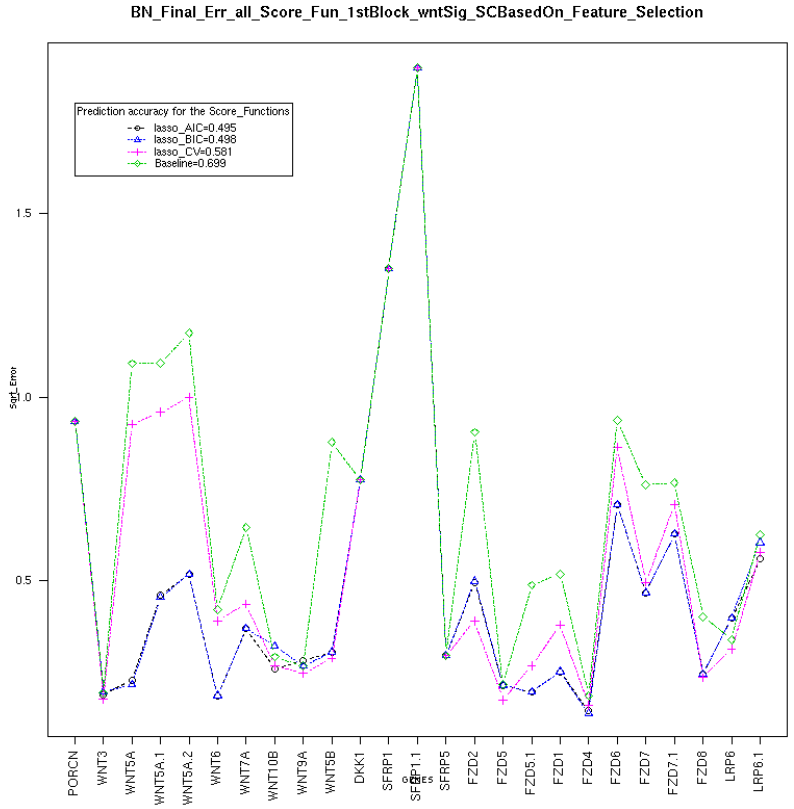


Figure 6.20: The comparison between lasso-methods and the baseline after using feature ranking.

6.10 A Verified Evaluation Using a Bigger Prostate Cancer Dataset

In this section, the methods developed in the previous section will be tested on a bigger dataset not previously used in the study. The dataset consists of prostate cancer gene expression profiles resulting from (Chandran et al. 2007). The cell files from this study are pre-processed using the same techniques discussed in Chapter 5. The resultant gene expression datasets have 62 samples. We restricted the study to JAK-STAT pathway genes for a prior knowledge consideration. Therefore, the final JAK-STAT gene expression profile used has 48 genes and 62 samples. The maximum subset of parents for each gene does not exceed 14 genes. Figure 6.21 shows the final error for each method when the big dataset is used. It can be seen that AIC-lasso works better than any other methods including the baseline method. The baseline shows the worst prediction error. For further assessment, it is important to test the significant difference between the chosen method, AIC-lasso and the baseline method. Therefore, a *K-fold-cross validated paired t test* is used as a significance test.

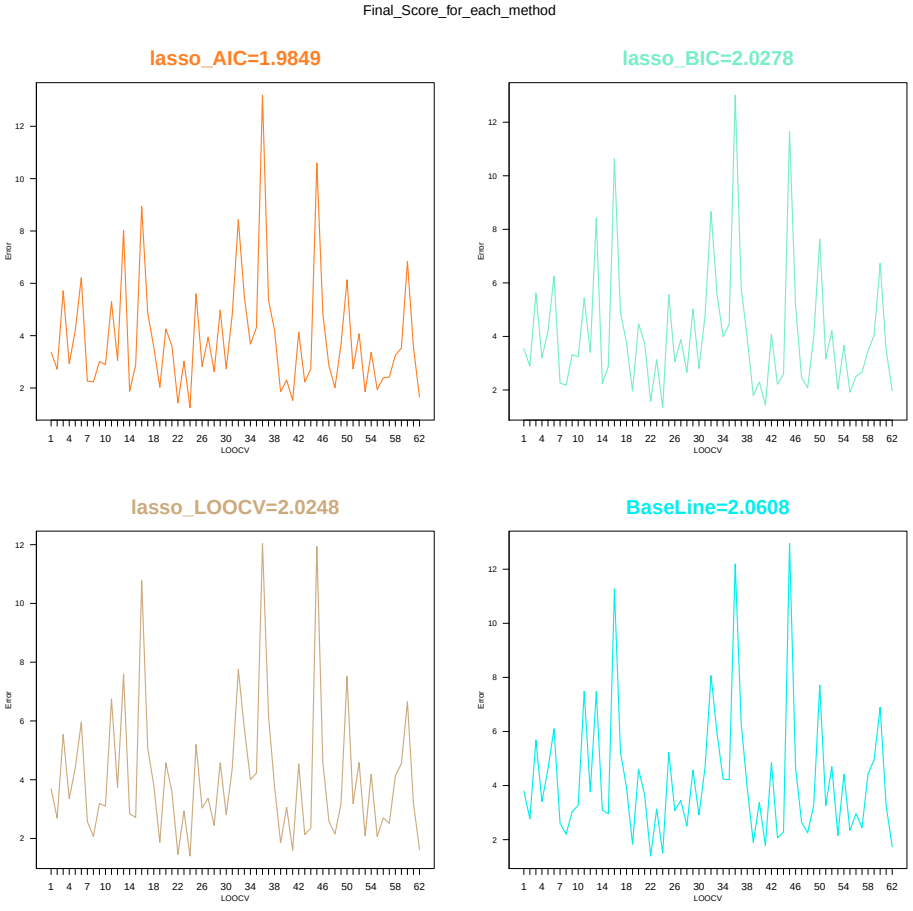


Figure 6.21: The final prediction error for each method when a bigger dataset is used for comparison.

6.10.1 K-fold-cross validated paired t test

This test is shown to be a powerful statistical test when two algorithms are compared in terms of their prediction accuracy (Dietterich 1998). The evaluation in the previous section is based on the error prediction from LOOCV. Therefore, when employing a t-test, these errors are the central points of the test. To illustrate, the comparison between AIC-lasso and the baseline method will be based on the prediction accuracy from each method. The difference between the prediction accuracy in the two methods is $p_i = p_i^1 - p_i^2$. The result is a distribution of p_i where $i = 1 \rightarrow k$. Given that $p_i^1$and p_i^2 are both *approximately* normally distributed and independent, their difference p_i is also normal (ALPAYDIN 2004). Therefore, the null hypothesis in the t-test is that: $\mu_p = \mu_{p^1} - \mu_{p^2} = 0$, and we are working towards rejecting this hypothesis (6.10)

$$\mathrm{t} = \frac{\sqrt{k}(\mu_p - 0)}{S} \tag{6.10}$$

Where,

$$\mathrm{S} = \sqrt{\frac{\sum_{i=1}^{k}(p_i - \mu)^2}{K-1}}, \mu = \frac{\sum_{i=1}^{k} p_i}{K}$$

The result of t test = 3.596106, with degree of freedom = K-1= 62-1 = 61 give p-value$<$ 0.05 from the t distribution table. Therefore, the accuracy of AIC-lasso is significantly different from the baseline method. Moreover, the dataset we used shows a low variance between each gene across samples. Figure 6.22 shows the variance for each gene across samples, for the dataset we used for further evaluation in the previous section. It shows that the variance is low except for 'SOCS3'. As mentioned above, the baseline method is about taking the mean of $n - i$ and predicting on i and therefore, given low variance we expect to see that the error resultant from the baseline is small. However, AIC-lasso has shown a better result, even in this difficult situation, and we regard this as another way of evaluating the result.

After showing that AIC-lasso has the best prediction accuracy on the bigger dataset, it is safe to use this method in our study to learn all four extended KEGG pathways in the four different cell types: cancer stem cells, cancer committed basal cells, benign stem cells and benign committed basal cells. There are 13 samples in the cancer population and 6 samples in the non-cancer population.

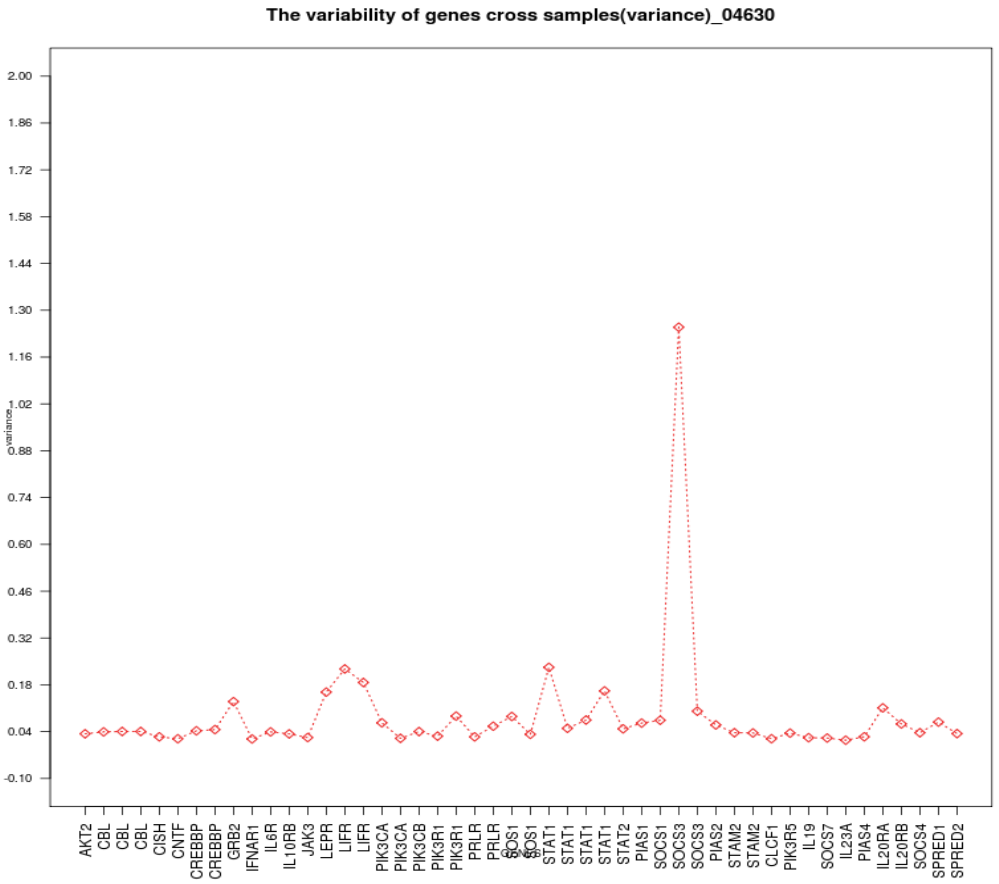


Figure 6.22: The variability of gene expression values cross samples in JAK-STAT dataset.

6.11 The Refined KEGG Pathways Using AIC-lasso with Feature Ranking

This section will show all the graphs learnt using the method developed above, AIC-lasso with an embedded feature ranking, for the first block of the Wnt signalling pathway, for the four different datasets (*SC-cancer*,*SC-non-cancer*,*CB-cancer*,and *CB-non-cancer*), the full graphs for the whole datasets of the four prostate cancer pathways, and the full graphs for the colon cancer datasets we experimented with. To ensure good interpretation and visualisation we used different methods as follows:

For some graphs : each family of genes are encapsulated in a rectangle. The thickness of a line means the amount of the directed co-expression relationship each gene has in other genes. The sign illustrates the direction of the co-expression relationship. Thus, if it is '+' the interpretation is:

- When the parent's expression level increases, the child's expression level is increased.
- When the parent's expression level decreases, the child's expression level is decreased.

If the sign is '-', the interpretation is :

- When the parent's expression level increases, the child's expression level is decreased.
- When the parent's expression level decreases, the child's expression level is increased.

6.11.1 The refined KEGG pathways for the first block of WNT KEGG pathway using AIC-lasso with feature ranking

This section shows the resultant graphs for the first block of the Wnt signaling pathway. As we have explained in the motivation of this work, all the graphs are an extended picture of how the co-expression relationships occur between gene families represented in KEGG which have been unknown previously. Understanding the co-expression relationships between genes in KEGG will lead to understand how different protein-protein activation/inactivation happens on the very low level of interaction in KEGG pathways , such as Wnt signalling pathway. mRNA-mRNA interaction between genes can be a strong evidence to undertand how the protein-protein interaction happens in KEGG pathways, given that the proteins that function in the same pathway are almost co-expressed (Webb & Westhead 2009). Figure 6.23 shows the first block of the Wnt signalling pathway for stem-*cancer* samples and Figure 6.25 shows the first block of the Wnt signaling pathway for committed basal-*cancer* samples. Figure 6.24 shows the first block of the Wnt signalling pathway for stem-*non-cancer* samples and Figure 6.26 shows the first block of the Wnt signaling pathway for committed basal-*non-cancer* samples.

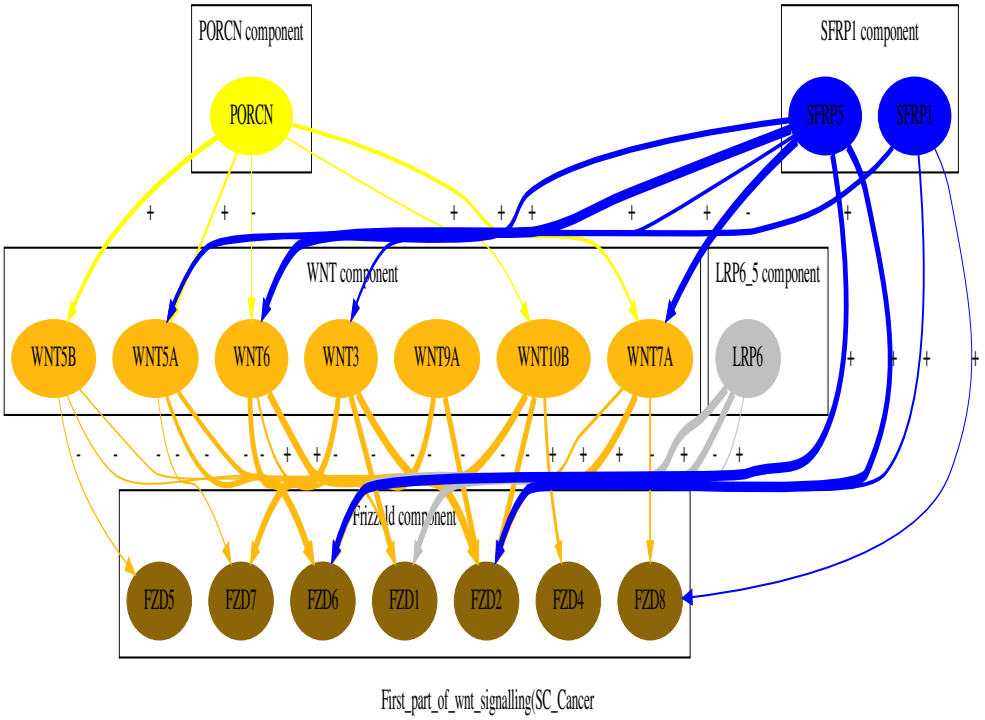


Figure 6.23: The refined 1st block of the Wnt signaling pathway for SC cancer samples.

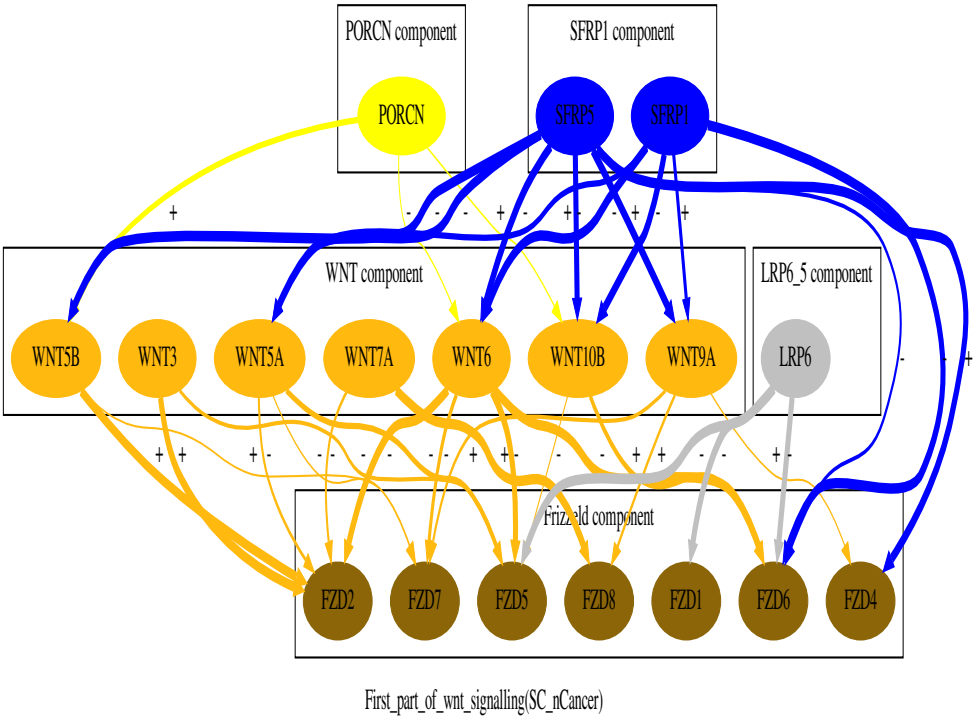


Figure 6.24: The refined 1st block of the Wnt signaling pathway for SC non-cancer samples.

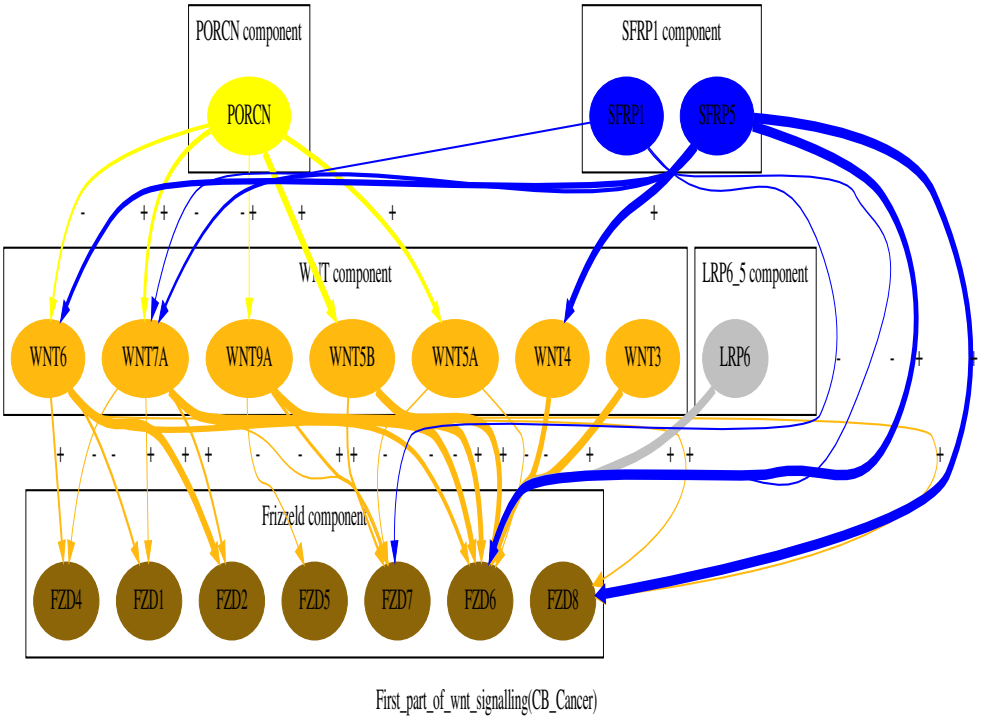


Figure 6.25: The refined 1st block of the Wnt signaling pathway for CB cancer samples.

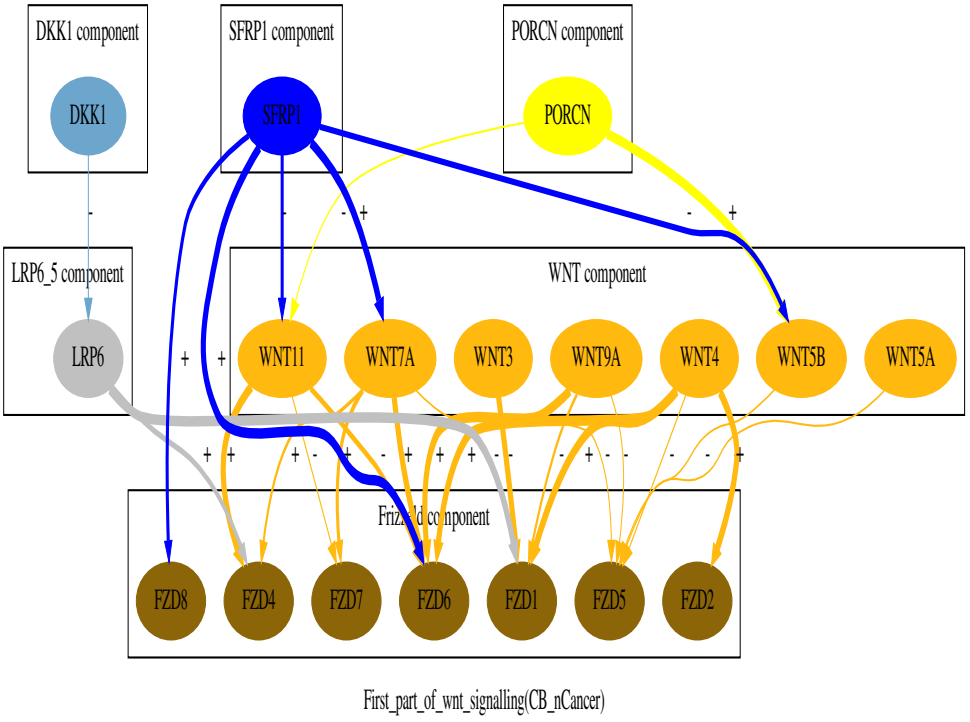


Figure 6.26: The refined 1st block of the Wnt signaling pathway for CB non-cancer samples.

For more evaluation, we have done a comparison between the resultant co-expressed graphs for cancer and non cancer samples and we show here the most important differences between the behaviour of genes in cancer and non cancer samples. We have compared the stem cell cancer graph in Figure 6.23 with the stem cell non-cancer graph in Figure 6.24 as follows:

- PORCN has more connections with the WNT family in SC-cancer than SC non-cancer.
- WNT5B in the WNT family has a higher directed relationship with FZD2 in SC non-cancer than in SC cancer.
- WNT3 in the WNT family has a higher directed relationship with the genes in the Frizzled family in SC cancer than in SC non-cancer.
- WNT7A has a high directed relationship with FZD8 in non-cancer SC, but a low directed relationship with FZD8 in SC cancer.
- WNT9A has a high directed relationship with Frizzled genes in SC cancer, but a low directed relationship in SC non-cancer.
- SFRP5 in the FRP family has a high influence on WNT7A in SC cancer. However, no influence is detected in SC non-cancer.

Also, we report here the most important differences between gene connectivity in committed basal cell graphs, cancer and non-cancer samples, that are shown in Figures 6.25 and 6.26, respectively, as follows :

- PORCN has the same amount of effect on WNT5B in cancer and non-cancer CB.
- DKK1 in the DKK family has a directed co-expression relationship with LRP6 in CB non-cancer. However, no co-expression relationship was detected in CB cancer.
- SFRP5 in the FRP family has many co-expression relationships with the WNT family in CB cancer, but no sign of any relationship between SFRP5 and the WNT family in CB non-cancer.
- WNT3 in the WNT family has a single high directed co-expression relationship with FZD6 in CB cancer and the same amount of relationship between WNT3 and FZD6 is found in CB non-cancer.
- WNT4 in the WNT family has a few high directed co-expression relationships with the Frizzled family in CB non-cancer, but only a single relationship with FZD6 in CB cancer.
- WNT11 in the WNT family has more than one high directed co-expression relationship with the Frizzled family in CB non-cancer . However, WNT11 did not appear in the WNT family in CB cancer, because this gene was unexpressed ($WNT11 < 50$) .

6.11.2 Linking the results of graphs with what is known in the literature

The results we obtained from AIC-lasso with feature selection method for the first block of the Wnt signaling pathway were verified using cross validation and a significance test. In addition to this, we linked the results obtained here with what is known in the literature about the individual interactions between each learned paired genes. We used three different resources as follows:

- Information Hyperlinked over Proteins(IHOP)(Hoffmann 2004): a network used to search millions of PubMed publication abstracts, to find if any interaction was found for a given gene with any other genes.
- Gene database: It has a section which provides the interaction between a given gene and other genes reported in PubMed publications.
- BioGRID: an online interaction repository, with references to the associated publications.

After searching the resources, we could not find any associated interactions for the datasets we have, stem cells and committed basal cells for prostate cancer and non-cancer. The resources largely report interactions between the families of genes, in the context of prostate cancer, such as the WNT-Frizzled families interaction, which we have a prior knowledge about from the graphical representation of the KEGG pathways we used. Therefore, this emphasises that the underlying connectivity between families of genes in the Wnt signaling pathway is still largely uncovered (Wang et al. 2009).

However, we have reported the same gene interactions in different samples that will be summarised here.

In (Tanaka et al. 2000), PORCN is reported to interact with WNT5B and WNT6 in human retina normal cells. We found that PORCN indirectly affects WNT5B in SC cancer and non-cancer samples (Figures 6.23 and 6.24). It also had indirect influence on WNT5B in committed basal cancer and non-cancer samples in which the amount of influence of PORCN on WNT5B is high (Figure 6.25 and 6.26). PORCN indirectly affects WNT6 in CB samples, but only in cancer samples in which PORCN has a low influence on WNT6 (Figure 6.25). In (Kim et al. 2008), it has been reported that in Hepatocellular Caric from liver cancer, that WNT3 activates FZD7. We also saw that WNT activates FZD7 in stem cell cancer samples and non-cancer samples, in Figures 6.23 and 6.24. The amount of directed co-expression relationship that WNT3 has on FZD7 is high in cancer samples, but in a negative direction. Thus, when WNT3 is up-regulated, FZD7 is down-regulated and when WNT3 is down-regulated, FZD7 is up-regulated. In stem cell non-cancer samples, WNT3 also has negative activation in FZD7, but with a low level. In (Saitoh & Katoh 2002), it has been shown that up-regulation of WNT5B, in several types of human cancer

expressing FZD5, might lead to more malignant phenotypes. We saw in stem cell cancer samples (Figure 6.23) that WNT5B has a negative sign activation on FZD5. However, in stem cell non-cancer samples in Figure 6.24, we have found a positive influence of WNT5B on FZD5, in which the up-regulation of WNT5B leads to an up-regulated FZD5 and vice versa. This connection is missed in cancer samples from committed basal cell, but exists in non-cancer samples in a negative direction (Figure6.26). In (Matsumoto et al. 2008), it has been reported that WNT9A binds to FZD4 and FZD7 during liver development. This finding is also detected in the normal stem cell and committed basal cells in the prostate sample (Figure 6.24,6.26), but in a specific direction. These graphs show that WNT9A activates FZD4 and FZD7 in a positive direction. (Lyonsa et al. 2004) reports that in kidney cell lines, WNT4 is found be bound to FZD6. In our committed basal cell non-cancer samples (Figure 6.26), we have observed in a directed manner that WNT4 has a high influence on FZD6 in a positive direction. However, in cancer committed basal samples in Figure 6.25, we observed that WNT4 has a high influence on FZD6 but in a negative direction. (Lyonsa et al. 2004) also shows that a member of Frizzled-related proteins (sFRPs), SFRP1, was found to regulate WNT4. In CB cancer samples in Figure 6.25, we found that another member of sFRPs, SFRP5, regulates WNT4 with a highly positive influence, but in prostate cancer samples. (Caricasole et al. 2003) identifies that an interaction between WNT7A and FZD5 is reported in PC12 cell lines. The activation of WNT7A leads to the expression of FZD5. The non-cancer samples of CB prostate in Figure 6.26 show this interaction. Moreover, (Caricasole et al. 2003) shows the detection of LRP6-FZD5 interaction in rat cell lines and we also detected this in stem cell non-cancer samples in Figure 6.24. Finally, (Semenov et al. 2001) shows that DKK1 is highly-affinity ligand for LRP6 in a rat normal cell line. We also detected a directed relationship between DKK1-LRP6 in a committed basal non-cancer sample in Figure 6.26, which could be related to the finding in rat normal cell lines.

We hope that such similarity in gene interactions in different cell lines can lead to more narrowed experiments, that reveal unknown discoveries, given that similar gene behaviours exist in different cell lines.

6.11.3 The full refined prostate cancer KEGG pathways using AIC-lasso with feature ranking.

After we developed the AIC-lasso method with the feature ranking method that is embedded in it for the sparse dataset, to refine the first block of the Wnt signaling pathway, we extended the application of this method to include all pathways that we are interested in. One important thing to be noted is that the results we obtained for all pathways are for the genes for which we had gene expression values. As stated above, the motivation was based on the fact that many of the

genes have been annotated in the four pathways: JAK-STAT signaling in Figure(4.5, page 56), Wnt signaling in Figure(1.4, page 21), the cell-extracellular matrix interaction pathway in Figure(4.6, page 57), and the focal adhesion signaling pathway in Figure(4.7, page 58). Therefore, there are some genes which are known to be in these KEGG pathways, but as we did not have gene expression values for them, we cannot know how they interact with each other.

Therefore, this section will show the fully refined signaling pathways for the four signaling pathways mentioned previously that have datasets in Table 6.4 and the result is 16 refined signaling pathways represent four signaling pathways from Stem cells(cancer , non-cancers samples), and Commited basal cells(cancer, non-cancer samples). Due to resolution issues with A4 page size, we show here the refined Wnt-signaling pathway for stem cells cancer samples in Figure 6.27. The rest(15 graphs) are all included in the CD attached with the thesis.

Table 6.4: Each pathway and its dataset

Pathway	Number of Samples	Number of Probes
Wnt signaling		451 probes
JAK-STAT	38 samples (19 SC,19 CB)	398 probes
Cell-extracellular matrix interaction	13 cancer,6 non-cancer	291 probes
Focal adhesion signaling		705 probes

6.12 Identifying the crucial causal relationships among genes involved in colon cancer treatments using Illumina microarray

In Chapter(4,Section 4.4.4), we obtained the normalised data for the four treated colon cancer cell lines and for the control colon cell line. In this section, we investigated how the AIC-lasso with the embedded feature selection can be used to show a more detailed picture of how the treatments have altered the behaviour of gene interactions. The work achieved here is based on two known KEGG pathways that have been found enriched with many genes in all five cell lines: the MAPK signaling pathway and the cell cycle signaling pathway. In this section we will give the results of $5 - Fluorouracil + Leucovorin$ treatment in MAPK signaling pathway genes, Figure(6.28). Likewise the control colon samples interactions found also in MAPK singaling pathway are shown in Figure 6.29.

Using the graphical representation for all treatments vs the control graph it will be possible to spot the differences and the similarities between genes behaviours and hence more inside details have been shown in these graphs. However,since this study has been done abroad , there might be more biological meanings of the results but we could not be able to discuss it with biologists

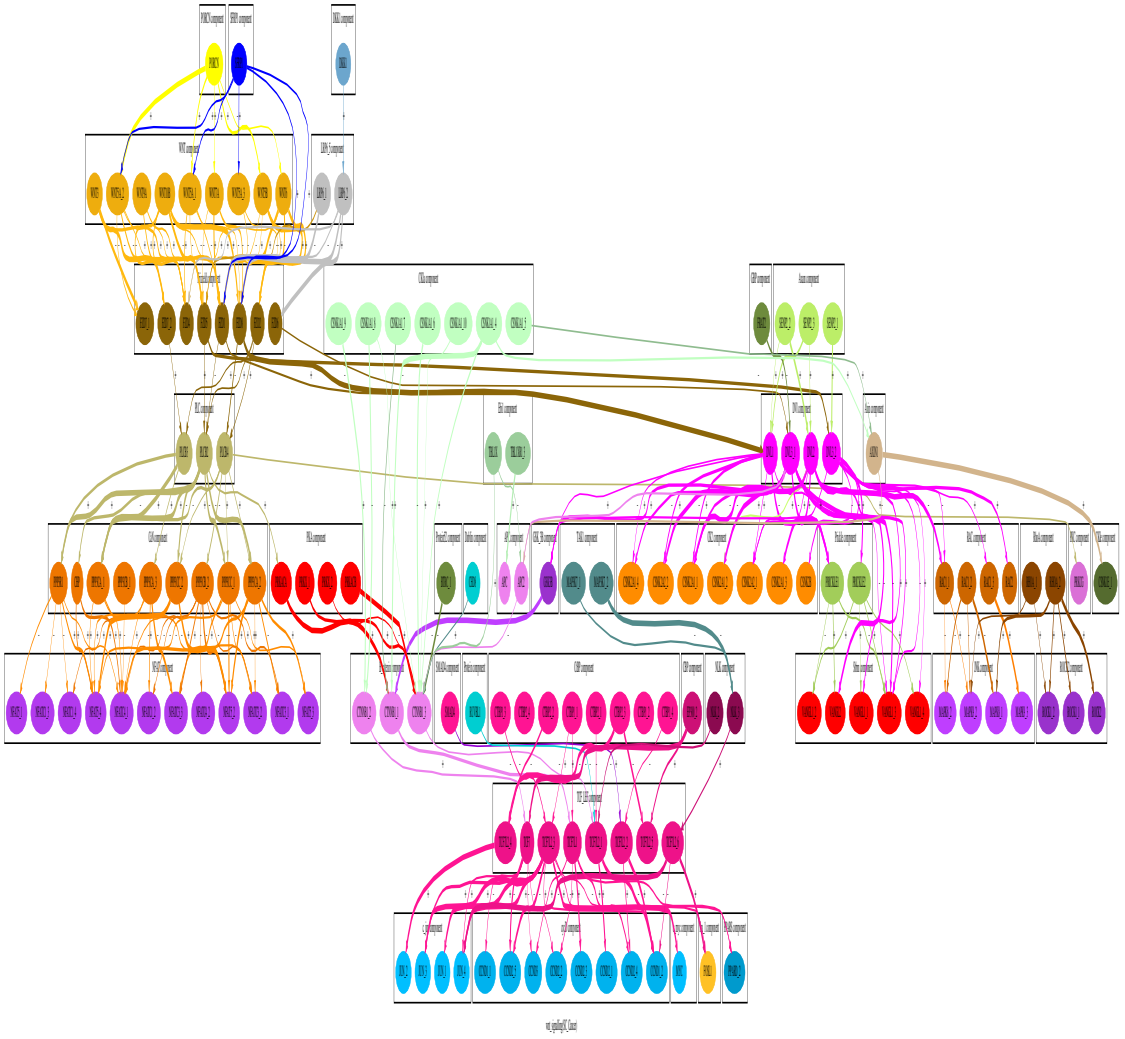


Figure 6.27: The refined Wnt signaling pathway (stem cell) cancer samples using AIC-lasso with feature selection.

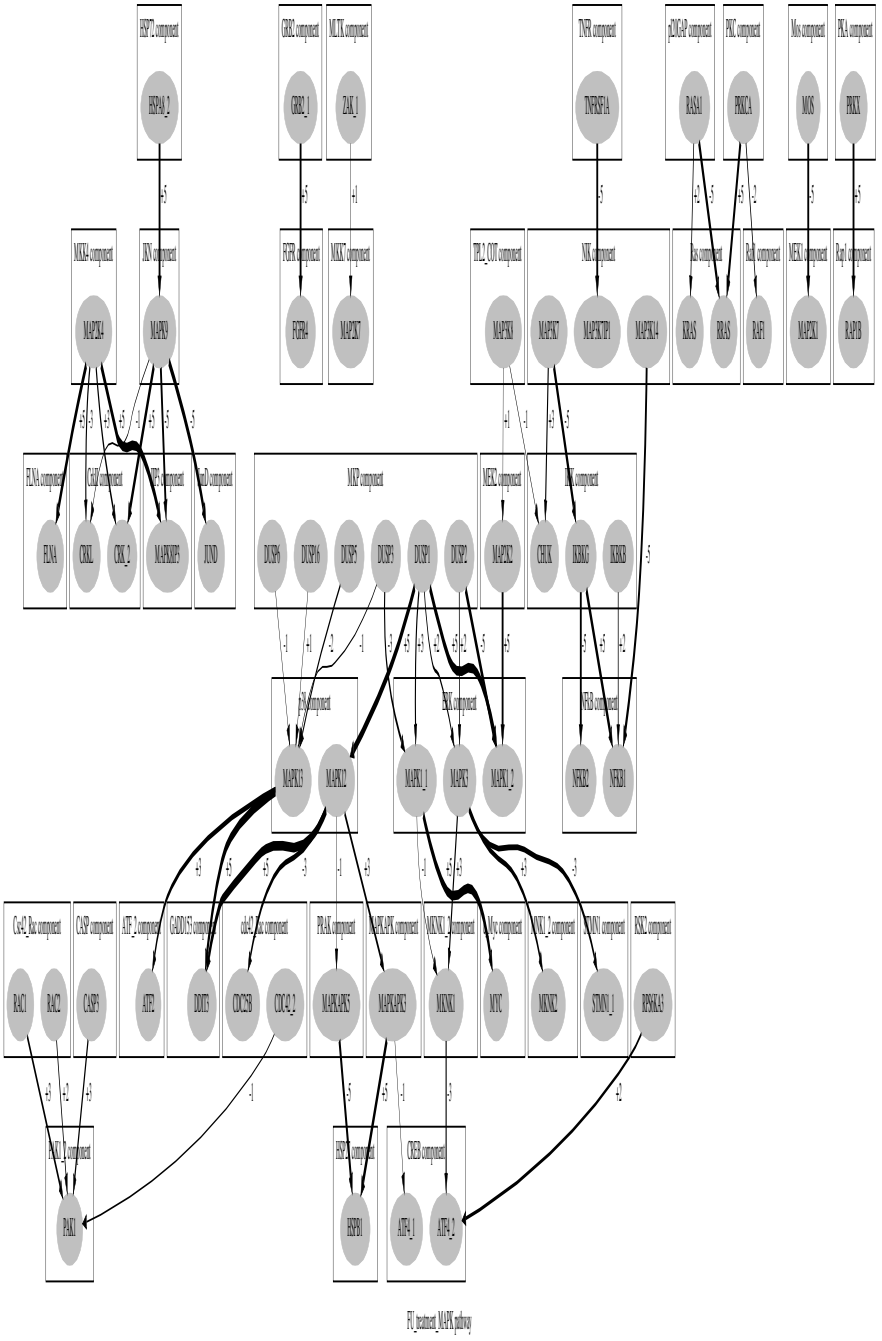


Figure 6.28: The interaction between colon cancer genes found in MAPK signaling pathway after Fluorouracil + Leucovorin treatment is applied.

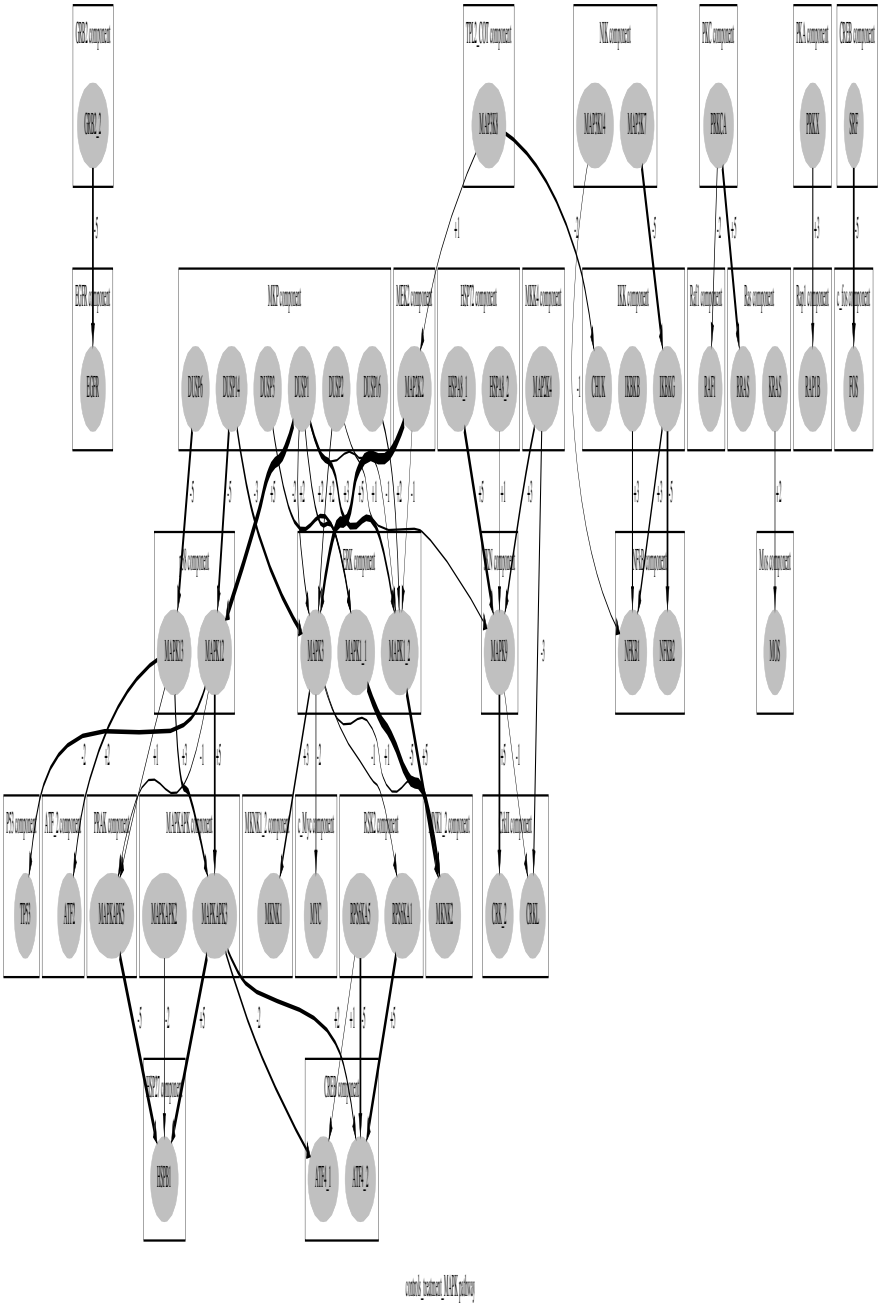


Figure 6.29: The interaction between colon control genes found in MAPK signaling pathway.

due to lack of communications. So, we consider that the main reason here is to show that the developed method worked in a smaller dataset(5 samples) than the ones we have used in my main goal(13 samples in cancer and 6 samples in non-cancer). The rest of the graphs(8 graphs) for all the treatments are all included in the CD attached with the thesis.

6.13 On learning without prior knowledge

In this section, the prior knowledge from KEGG will be evaluated on the basis of how informative it is. The idea is to use the dataset in Table 5.1 without injecting prior knowledge from KEGG, to see if the true parents represented in KEGG can be found, when no prior knowledge is used. Using lasso-methods or normal regression without KEGG information will basically lead any method to get stuck in either $-\infty$ values or overfitting issue, as explained in the previous sections, since without KEGG each gene will have 25 possible parents. Therefore, another existing method implemented in the R-package, graphical lasso (glasso), is used, to find out whether the prior knowledge from KEGG can be found when only the dataset is used, without the KEGG background knowledge. The glasso method works as follows: given the dataset has a multivariate Normal distribution with mean μ and covariance matrix $\Sigma \sim N_p(\mu, \Sigma)$, an imposed penalty is introduced to the inverse matrix $\sum^{-1}$ to increase its sparseness. The glasso solution shows that if $ij = 0.0$ in $\sum^{-1}$, then $i \perp j$ given all other variables (Friedman et al. 2007). The result of this method is depicted in Figure 6.30 with different values of λ. The results show that there are many incorrect dependences between genes, for example, when $\lambda = 0.1$ there is a dependency between PORCN and SFR1.1, PORCN and FZD5.1, but KEGG does not have these dependencies. Also, as λ increases, many dependences in KEGG are discarded by glasso. Therefore, this is evidence that KEGG background knowledge is informative and needed in finding the optimal subset of parents for each gene.

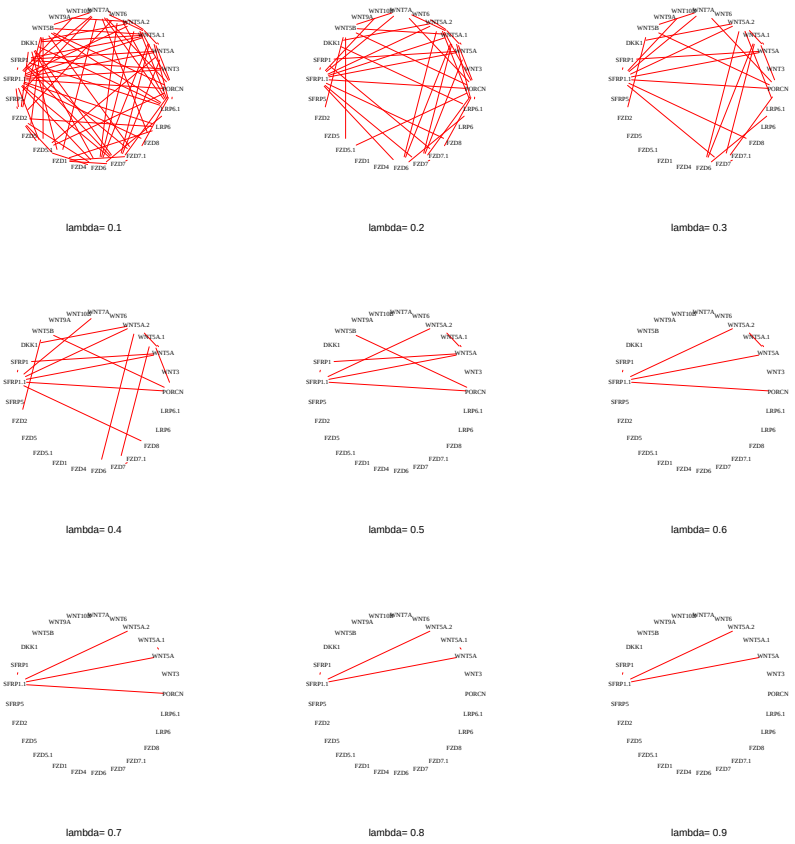


Figure 6.30: The dependences found by glasso for different values of λ.

CHAPTER 7

Conclusion and Future work

This chapter summarises the thesis, including revisiting the motivation behind the work and discussing the contribution of the thesis towards effective computational biology methods in Section 7.2. Section 7.3 reviews the chapters involved in this thesis, and at the end of this chapter, a discussion about the limitations of the thesis and possible future work is given in Section 7.4.

7.1 Introduction

Applying machine learning of graphical models has attracted many researchers, both in machine learning and computational biology, to solve the problem of finding a more detailed picture of how cellular systems work. With the existence of high throughput technologies, like microarrays, it became possible to study the whole cellular system in one or two experiments. However, almost immediately, biologists and geneticists realised that the genomic profiles resulting from microarrays are far from manually manipulatable. Therefore, there is a great need for powerful procedures, incorporating statistics and intelligent methods such as machine learning algorithms. The machine learning community find the application of machine learning in real-world problems will advance the ability to make knowledge-based systems available for computational biology problems. It also opens research questions on how machine learning methods can be used with genomic data. One of the most provoking research questions in machine learning at the present is how to develop methods in sparse datasets. This kind of data is always produced as a result of microarrays, where thousands of genes are measured in a few samples.

7.2 Motivation Revisited

With the existence of microarray datasets, robust methods are needed to reveal new knowledge that cannot be easily retrieved. The gene expression profiles resulting from microarrays are not usually in a form which can be used by machine learning algorithms, because thousands of genes, with only a few samples of each to learn from, is likely to lead to models that suffer from overfitting. Learning graphical models from gene expression microarray data is one of the most popular areas of interest among machine learning researchers. The advancement of low-level interactions visualisation knowledge-based databases, such as KEGG pathways, has helped to look at the problem of learning from gene expression profiles as a complementary step towards more understanding of how gene interactions occur under a particular condition, such as cancer. Therefore, the way we approached the problem in this thesis, was to make a contribution to what is *less* complete in the current knowledge-based databases. The motivation was based on a study by (Birnie et al. 2008) which uses cancer and non-cancer samples to highlight the differential gene expressions between them in prostate samples. This led in turn to us focusing more on where the genes that are involved in the study are annotated.

The genes are annotated in four different pathways: JAK-STAT signaling in (Figure 4.5, page 56), Wnt signaling in (Figure 1.4, page 21), the cell-extracellular matrix interaction pathway in (Figure 4.6, page 57) and the focal adhesion signaling pathway in (Figure 4.7, page 58). However, the research question that arose, is whether we can make use of machine learning to give a more detailed picture about the gene interactions in these four KEGG pathways or not.

A class of machine learning algorithms, known as graphical models learning algorithms, have been used throughout this thesis to contribute towards more understanding of cellular systems and how each gene, in each pathway, interacts with those around it. One of the main motivations for using machine learning of graphical models in this research problem is the ability to construct meaningful networks based on the injection of natural prior knowledge and the co-expression relationships between genes , which is also helpful to smooth the complexity of machine learning search algorithms when applied to small datasets, which contain thousands of variables (genes),Table (2.1,page 33). Prior knowledge from the KEGG pathways database allows us to find which family of genes in each pathway co-expresses with which other families for example , between WNT family and Frizzled family . However, the KEGG database does not show which member of each family co-expresses with which member of the other family.

The main contribution of this thesis is to reveal what is unknown in KEGG pathways. We concentrated on discovering how the members of gene families work together in the four KEGG pathways mentioned above. The prior knowledge that is obtained from KEGG allows the search

space to be reduced, for example, when searching the possible subset of co-expressed genes for each gene. We do not need to use all the variables in the dataset to find the best co-expressed genes for a particular gene, only those that the KEGG database shows that the gene in question interacts with. An example of this can be seen in the Frizzled family; the KEGG database will retrieve that the WNT family interacts with the Frizzled family, which makes the search space for each WNT member much less than the search space for WNT members without any constraints, as the goal is to find out which gene of the WNT family co-experesses which gene in the Frizzled family.

7.3 Summary of the Thesis Chapters

The chapters in the thesis have been organised to show the work from the starting point of the research problem, through the progress of the work, until we end up with a method we developed, which was used to learn from sparse datasets. The following sections give a brief summary of each chapter and highlight the main findings when necessary.

7.3.1 Chapter One: Introduction

In this chapter, the state of the art of the research problem was discussed. This included the transformation of 20th century genetics into 21st century genomic, through microarray technology. We also highlighted the main issue with microarrays when we want to use statistical methods on the output. We discussed the curse of dimensionality for the gene expression datasets that microarrays generate and how this kind of issue has given a huge consideration to statistical machine learning methods and also made use of novel methods with vital genomic tools. We then discussed the most useful machine learning techniques that are used in microarray analysis. Finally, the motivation behind the work that has been achieved was detailed.

7.3.2 Chapter Two: Background

This chapter concerned machine learning in general with much attention to machine learning of graphical models and related work in the context of microarray analysis. We discussed clustering algorithms, which are unsupervised learning algorithms, used to find the similarity between gene expression profiles, if the finding aims for example to work out which genes show similar patterns in different conditions, such as cancer and non-cancer. There was an introduction to Support Vector Machines (SVMs), that is one of the most successful supervised learning techniques, either in microarray or other datasets. The robustness of SVMs comes from the projections that the *Kernel methods* make on the data points in the space, so that a clearer separable hyperplane can be used to classify the data points into two classes resulting in less prediction errors when a new data point is used to test the classifier. In addition, we introduced graphical models which are the main class of machine learning algorithms used in the thesis . Different sets of graphical models were discussed, which were found in the literature to be the most commonly used, when a

graphical representation of genes interaction is needed. The focus is on Bayesian networks for a very biased reason, a directed co-expressed graph is needed to show how the interaction is influenced between genes based on gene expression relationships . We then discussed how Bayesian networks are learnt from data if there is no expert knowledge to construct the graphical representations between set of variables generally. We showed how the parameters are estimated after a Bayesian network is constructed, and detailed a frequentist approach, Maximum Likelihood estimation (MLE). A Bayesian approach to estimating the parameters was also briefly mentioned. Following that we highlighted the inference in Bayesian networks and a simple example was used for demonstration. At the end of the chapter, we took a broad survey of the literature to show how machine learning of graphical models is used to construct cellular systems.

7.3.3 Chapter Three: The Biology of Cancer

In this chapter, we tried to show the two most frequently diagnosed cancer types, breast cancer and prostate cancer. We detailed how the two types of cancer occur, the ways breast cancer is usually diagnosed and how they are both usually treated. We also detailed how the cell communication occurs between cells.

7.3.4 Chapter Four: Microarray Technology and Gene Expression Profiles Data Analysis

This chapter goes through microarray technology and its practical use. At the beginning of the chapter, different microarrays types are discussed and the Affymetrix microarray is discussed in detail. The purpose of this chapter is to show how the intensity files, which are the results of microarray experiments, are pre-processed in order to get readable and numerical datasets. We then discussed the detailed steps of one of the most well known pre-processing algorithms, robust multichip average (RMA), in normalising and pre-processing gene expression intensities. We then showed how RMA is used practically together with other cleaning up steps to pre-process the datasets that are used in this thesis. The RMA algorithm is used to pre-process prostate samples from the Affymetrix microarray and colon samples from the Illumina microarray.

7.3.5 Chapter Five: Learning Refined Graphical Models for KEGG Pathways Using Existing Tools

This chapter details the first attempt to learn a graphical model from the first block of the Wnt signaling pathway. Logically, we had to try existing tools to discover their advantages and disadvantages, as tools exist which are used to learn different sets of graphical models. Since we emphasised previously that Bayesian networks are going to be used in this work, we tried several different tools to learn Bayesian networks. Each tool has its advantages and disadvantages, which were noted in this chapter. The common problem seen in all the tools we used was that no robust way was found to invoke the prior knowledge gained from KEGG to simplify the search

space. As we had a small dataset, prior knowledge is very important to reduce the search space and hence less overfitting is encountered.

7.3.6 Chapter Six: Learning Linear Gaussian Models

This chapter shows the main contribution and the novel methods developed in this thesis. It shows how the problem of learning can be considered as learning families of directed co-expressed genes for each gene, and then joining them all together, to give a full graph for the dataset used in learning.

At the beginning of this chapter, we set up the problem of learning as a linear Gaussian models learning problem. This requires the assumption of multivariate Normal distribution. A multivariate Normal test, the Shapiro-Wilk, is conducted on the dataset to show to what degree the dataset meets this assumption. After that, different attempts were shown on how a Gaussian linear model is learned, starting with the simplest case, which is co-expression networks. We detailed the drawbacks from using co-expression network to learn the full co-expressed relationships between genes and then moved to show how penalised goodness-of-fit methods can be used to learn more realistic co-expressed relationships between genes. We indicated that using penalised goodness-of-fit methods can be improved using shrinkage methods. The lasso estimate, which is used in this chapter, shows the advantage of using shrinkage methods over normal regression methods and how the penalised scoring functions are used with the lasso estimate, to obtain more robust results.

In this chapter, we also conducted intensive experiments to overcome the overfitting problem. Consequently, this leads to make more constraints on the problem of learning a refined KEGG pathway, by only considering interactions between families of genes rather than involving the interaction between genes in each family. To strengthen the learning against overfitting, we also included an embedded function, named feature ranking, which gives more robust results along with the lasso estimate and the AIC scoring function, when overfitting is a concern. Moreover, the method developed showed much better results than the baseline method, which considers that each gene is best without parents. However, the results reveal that each gene has meaningful directed co-expressed genes in the dataset, and also gave the optimal parents which could be found in the sparse dataset.

Furthermore, we used a statistical test, the *K-fold-cross validated paired t test* to find out whether the method developed, AIC-lasso with feature selection, works better than the baseline method by chance or significantly. The result of the statistical test shows that AIC-lasso with feature selection works significantly better than the baseline method used in the comparison. We also had another way of comparing, this time with the prior knowledge used from KEGG. We tested

whether the prior knowledge can be captured from only the dataset without using the KEGG database or not. The result shows that using the KEGG database is important to add an extra layer of meaning to the subset of co-expressed genes that are chosen by the developed method.

7.3.7 Chapter Seven: Conclusion and Future work

This chapter summarises the thesis and draws attention to future work and the limitations of the datasets we have used.

7.4 Limitations and Future Work

In this work, we have had a chance to work with a difficult application, which always generates sparse datasets. Microarray technology has given a new direction to treat many diseases such as cancer, for example, by tracking the infected cellular systems of such diseases, then comparing them to control cellular systems that are healthy and focusing on the differences for treatments. However, in the existence of such small datasets, generated from microarrays, effective solutions still need to be further considered, if machine learning algorithms are going to be used. The effectiveness of machine learning algorithms is solely based on the amount of data which is used. To put it another way, to simulate any existing real-world problem, we need a big population to work with, which unfortunately is not the case in microarray datasets. One of the critical evaluations we have encountered in this thesis is the limitation of the datasets we have had to work with. In fact, the methods developed in this thesis were based on 13 samples.

The starting point for the motivation for this thesis was to refine the four KEGG pathways to include the following:

- The study of all genes in each pathway; and
- learning how the interaction between genes inside each gene family occurs; and
- learning how the interaction between gene families occurs; and
- linking all four pathways together, as they have common genes.

However, because of the limited samples we had, the only point we looked at is how the interaction between families occurs. We sacrificed a more detailed picture for the sake of the global picture, and more accurate results. In future projects, we intend to do more work on sparse datasets from microarrays and develop machine learning algorithms that can survive with such data limitation. This research problem has very recently received huge attention among the machine learning community. However, the way we will develop machine learning algorithms is by using the natural prior knowledge that exists in the KEGG pathways. The goal then will be to add an extra layer of information to what already exists in KEGG, and also to make sense of graphical models in machine learning in real-world problems.

The work we have done in this thesis reports a new direction in how the sensitivity of the tuning parameter in lasso-estimate can be improved using feature selection methods such as feature ranking when small datasets are used. So, in future work also we will look in more details how the sensitivity of the tunning parameter in lasso-estimate can be improved when small datasets are used in learning.

Another possible direction for future work is concerned with combining different datasets from different microarrays experiments. Finding methods to combine different datasets from different microarrays platforms is also being discussed intensively and this is because there are many resources which exist for gene expression datasets and each suffers from small sample sizes. To combine different gene expression datasets one has to be careful with the systematic differences between the different datasets. The systematic differences include the temperature of the lab when the genes are measured, the tools used to perform the experiments, such as the scanner of the gene intensities that is used, and even the experience of the biologist who conducts the experiment. All these issues have statistically prevented the datasets for one condition, but measured in different labs, from directly combining and being in one dataset. However, we believe that since the multivariate Normal distribution assumption is being *less* restricted when dealing with microarray datasets, combining different datasets from different resources to increase the sample size will also receive *less* restriction statistically, as long as the hope is to find useful information from such combinations. So that, in the future we will look at this interesting point for more accurate machine learning methods to be applied in microarray datasets analysis.

References

Abeyta, M. J., Clark, A. T., Rodriguez, R. T., Bodnar, M. S., Pera1, R. A. R., & Firpo, M. T. (2004). Unique gene expression signatures of independently-derived human embryonic stem cell lines. *Hum. Mol. Genet*, *6*(13), 601–608.

Adami, H.-O., David, H., & Dimitrios, T. (2002). *Textbook of Cancer Epidemiology*. Oxford University Press.

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K., & Walter, P. (2002). *Molecular Biology of the Cell* (4th ed.). Garland Science.

Aloraini, A., Cussens, J., & Birnie, R. (2010). Extending prostate cancer kegg pathways using machine learning of graphical models. In *Systemics and Informatics World Network,SIWN*, volume 10, (pp. 56–67).

ALPAYDIN, E. (2004). *Introduction to machine learning*. MIT press.

Amit, S., Hatzubai, A., Birman, Y., Andersen, J., Ben-Shushan, E., Mann, M., Ben-Neriah, Y., & Alkalay, I. (2002). Axin-mediated cki phosphorylation of beta-catenin at ser 45: a molecular switch for the wnt pathway. *Genes and development*, *16*(9), 1066–1076.

Anand, P., Kunnumakara, A. B., Sundaram, C., Harikumar, K. B., Tharakan, S. T., Lai, O. S., Sung, B., & Aggarwal, B. B. (2008). Cancer is a preventable disease that requires major lifestyle changes. *Pharm Res*, *25*(9), 2097–2116.

Barash, Y., Dehan, E., Krupsky, M., Franklin, W., Geraci, M., Friedman, N., & Kaminski, N. (2004). Comparative analysis of algorithms for signal quantitation from oligonucleotide microarrays. *Bioinformatics*, *20*(6), 839–846.

Barczak, A., Rodriguez, M. W., Hanspers, K., Koth, L. L., Tai, Y. C., Bolstad, B. M., Speed, T. P., & Erle, D. J. (2003). Spotted long oligonucleotide arrays for human gene expression analysis. *Genome Res*, *7*(13), 1775–85.

Barnes, M., Freudenberg, J., Thompson, S., Aronow, B., & Pavlidis, P. (2005). Experimen-

tal comparison and cross-validation of the affymetrix and illumina gene expression analysis platforms. *Nucleic Acids Research*, *33*(18), 5914–23.

Birnie, R., Bryce, S., Roome, C., Dussupt, V., Droop, A., Lang, S., Berry, P., Hyde, C., Lewis, J., Stower, M., Maitland, N., & Collins, A. (2008). Gene expression profiling of human prostate cancer stem cells reveals a pro-inflammatory phenotype and the importance of extracellular matrix interactions. *Genome Biology Journal*, *9*(2), R83.

Bolstad, B. M., Irizarry, R. A., Astrand, M., & Speed, T. P. (2003). A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, *19*(2), 185–193.

Bøtcher, S. G. & Dethlefsen, C. (2009). `deal`: A package for learning Bayesian networks. *Journal of Statistical Software*, *8*(20), 1–40.

Butte, A. J., Kohane, I. S., & Kohane, I. S. (2000). Mutual information relevance networks: functional genomic clustering using pairwise entropy measurements. In *In Proceedings of the Pacific Symp On Biocomputing*, (pp. 418–29).

Campbell, N. A. & B.Reece, J. (2005). *Biology*. The Benjamin Cummings Publishing Company Inc.

Caricasole, A., Ferraro, T., Iacovelli, L., Barletta, E., Caruso, A., Melchiorri, D., Terstappen, G. C., & Nicolett, F. (2003). Functional characterization of wnt7a signaling in pc12 cells interaction with a fzd5Â·lrp6 receptor complex and modulation by dickkopf proteins. *Journal of Biological Chemistry*, *278*(39), 37024–31.

Chandran, U. R., Ma, C., Dhir, R., Bisceglia, M., Lyons-Weiler, M., Liang, W., Michalopoulos, G., Becich1, M., & Monzon, F. A. (2007). Gene expression profiles of prostate cancer reveal involvement of multiple molecular pathways in the metastatic process. *BMC Cancer*, *7*, 64.

Chickering, D. M. (1996). Learning Bayesian networks is NP-complete. In D. Fisher & H.-J. Lenz (Eds.), *Learning from data: AI and statistics V* chapter 12, (pp. 121–130). Springer.

Chickering, D. M. (2002). The winmine toolkit. Technical Report MSR-TR-2002-103, Microsoft Research.

Chickering, D. M., Heckerman, D., & Meek, C. (2004). Large-sample learning of Bayesian networks is np-hard. *Journal of Machine Learning Research*, *5*, 1287–1330.

Coiera, E. (2003). *Guide to Health Informatics*. Arnold,London.

Cooper, G. F. (1990). The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, *42*, 393–405.

Cooper, G. F. & Herskovits, E. (1991). A Bayesian method for the induction of probabilistic networks from data. Technical Report Stanford KSL 9.

Delen, D., Walker, G., & Kadam, A. (2005). Predicting breast cancer survivability: a comparison of three data mining methods. *Artificial Intelligence in Medicine*, *34*(2), 113–127.

D'Haeseleer, P., Wen, X., Fuhrman, S., Wen, X., Fuhrman, S., & Somogyi, R. (1999). Linear modeling of mrna expression levels during cns development and injury. In *In Proceedings of*

the Pacific Symp On Biocomputing, volume 4, (pp. 41–52).

Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, *10*, 1895–1923.

Dougherty, J., Kohavi, R., & Sahami, M. (1995). Supervised and unsupervised discretization of continuous features. In *ICML*, (pp. 194–202).

Dudley, R. (2010). The shapiro-wilk test for normality.

Edwards, D. (2000). *Introduction to Graphical Modelling*. Springer.

Efroymson, M. A. (1960). Multiple regression analysis. *In Mathematical Methods for Digital Computers(eds. A. Ralston and H. S. Wilf)*, *1*, 191–203.

Fentiman, I. S. (1998). *Detection and Treatment of Breast Cancer*. Informa Healthcare.

Ferrazzi, F., Sebastiani, P., Ramoni, M., & Bellazzi, R. (2007). Bayesian approaches to reverse engineer cellular systems: a simulation study on nonlinear Gaussian networks. *BMC Bioinformatics*, *8*(S-5).

Friedman, J., Hastie, T., & Tibshirani, R. (2007). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, *0*(0), 1–10.

Friedman, N. (2004). Inferring cellular networks using probabilistic graphical models. *Science*, *303*(5659), 799–805.

Friedman, N., Linial, M., Nachman, I., & Pe'er, D. (2000). Using Bayesian networks to analyze expression data. In *RECOMB*, (pp. 127–135).

Geiger, D. & Heckerman, D. (1994a). Learning Gaussian networks. In *UAI*, volume 10, (pp. 235–243).

Geiger, D. & Heckerman, D. (1994b). Learning Gaussian networks. Technical Report MSR-TR-94-10, Microsoft Research, Redmond, WA. Revised July 1994.

Greenbaum, D., Colangelo, C., Williams, K., & Gerstein, M. (2003). Comparing protein abundance and mrna expression levels on a genomic scale. *Genome Biology*, *4*(9), 117.

Guyon, I. (2008). *Practical Feature Selection: from Correlation to Causality*. IOS Press.

Hair, J. F., Anderson, R. E., Tatham, R. L., & Black, W. (1998). *Multivariate Data Analysis*. Prentice Hall College Div. Peter's book.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explorations*, *11*(1).

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer New York Inc.

Heckerman, D., Chickering, D. M., Meek, C., Rounthwaite, R., & Kadie, C. (2000). Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research*, *1*, 49–75.

Heckerman, D., Meek, C., & Cooper, G. (1997). A Bayesian approach to causal discovery. Technical report, Microsoft research.

Hesterberg, T. C., Choi, N. H., Meier, L., & Fraley, C. (2008). Least angle and l1 penalized

regression: A review. *Statistics Surveys*, *2*, 61–93.

Hoffmann, R. (2004). A gene network for navigating the literature. *Nature Genetics*, *36*(664).

Irizarry, R. A., Bolstad, B. M., Collin, F., Cope, L. M., Hobbs, B., & Speed, T. P. (2003a). Summaries of affymetrix genechip probe level data. *Nucleic Acids Res*, *31*(4), e15.

Irizarry, R. A., Hobbs, B., Collin, F., Barclay, Y. D. B., Antonellis, K. J., Scherf, U., & Speed, T. P. (2003b). Exploration normalization and summaries of high density oligonucleotide array probe level data. *Biostat*, *4*(2), 249–264.

Jansen, R., Greenbaum, D., & Gerstein, M. (2002). Relating whole-genome expression data with protein-protein interactions. *Genome Res*, *12*, 37–46.

Joseph A. Cruz, D. S. W. (2006). Applications of machine learning in cancer prediction and prognosis. *Cancer Informatics*, *2*, 59–78.

Kanehisa, M. & Goto, S. (2000). Kegg:kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, *28*, 27–30.

Kanehisa, M., Goto, S., Furumichi, M., Tanabe, M., & Hirakawa, M. (2010). Kegg for representation and analysis of molecular networks involving diseases and drugs. *Nucleic Acids Research*, *38*, D355–D360.

Kerr, M. K., Martin, M., & Churchill, G. A. (2000). Analysis of variance for gene expression microarray data. *Journal of Computational Biology*, *7*(6), 819–837.

Kim, M., Lee, H. C., Tsedensodnom, O., Hartley, R., Lim, Y.-S., Yu, E., Merle, P., & Wands, J. R. (2008). Functional interaction between wnt3 and frizzled-7 leads to activation of the wntβ-catenin signaling pathway in hepatocellular carcinoma cells. *Journal of Hepatology*, *48*(5), 780–791.

Koller, D. & Friedman, N. (2009). *Probabilistic Graphical Models*. The MIT Press, London.

Land, A. H. & Doig, A. G. (1960). An automatic method of solving discrete programming problems. *Econometrica*, *28*(3), 497–520.

Larranaga, P., Poza, M., Yurramendi, Y., Murga, R. H., & Kuijpers, C. M. H. (1996). Structure learning of Bayesian networks by genetic algorithms: A performance analysis of control parameters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *18*(9), 912–926.

Laurent, P., Christophe, B., & Diane, M. (2004). Natural killer cells distinguish innocuous and destructive forms of pancreatic islet autoimmunity. *PNAS*, *101*(21), 8102–8107.

Lauritzen, S. L. (1996). *Graphical Models.* Clarendon Press, Oxford.

Li, C. & Wong, W. (2001a). Model-based analysis of oligonucleotide arrays: model validation, design issues and standard error application. *Genome Biol*, *2*(8), 1–11.

Li, C. & Wong, W. (2001b). Model-based analysis of oligonucleotides arrays: expression index computation and outlier detection. *Proc Natl Acad Sci*, *98*, 31–36.

Lyonsa, J. P., Muellera, U. W., Jia, H., Everetta, C., Fanga, X., Hsieh, J.-C., Barthd, A. M., & McCrea, P. D. (2004). Wnt-4 activates the canonical beta-catenin-mediated wnt pathway and binds frizzled-6 crd: functional implications of wnt/beta-catenin activity in kidney epithelial

cells. *Experimental Cell Research*, *298*(2), 369–387.

Markowetz, F. & Spang, R. (2007). Inferring cellular networks - a review. *BMC Bioinformatics*, *8*(S-6).

Matsumoto, K., Mikia, R., Nakayamaa, M., Tatsumia, N., & Yokouchi, Y. (2008). Wnt9a secreted from the walls of hepatic sinusoids is essential for morphogenesis, proliferation, and glycogen accumulation of chick hepatic epithelium. *Developmental Biology*, *319*(2), 234–247.

Meinshausen, N. & Buhlmann, P. (2006). "high-dimensional graphs and variable selection with the lasso". *The Annals of Statistics*, *34*(3), 1436–1462.

Miller, A. (2008). *Subset Selection in Regression*. Chapman and Hall.

Murphy, K. & Mian, S. (1999). Modelling gene expression data using dynamic Bayesian networks. Technical report, Computer Science Division, University of California.

Newton, J. (2001). Analysis of microarray gene expression data using machine learning techniques.

Niculescu-Mizil, M. S. A. & Murphy, K. (2007). Learning graphical model structure using l1-regularization paths. In *Proceedings of the 22nd national conference on Artificial intelligence - Volume 2*, (pp. 1278–1283). AAAI Press.

Parkin, D. M., Bray, F., Ferlay, J., & Pisani, P. (2000). Global cancer statistics. *a cancer journal for clinicians*, *55*(1), 47–108.

Parmigiani, G., Garrett-Mayer, E. S., Anbazhagan, R., & Gabrielson, E. (2004). A cross-study comparison of gene expression studies for the molecular classification of lung cancer. *Clinical cancer research*, *9*(10), 2922–7.

Parrish, R. S., III, H. J. S., & Xu, P. (2009). Distribution modeling and simulation of gene expression data. *Computational Statistics & Data Analysis*, *53*(5), 1650–1660.

Pearl, J. & Verma, T. (1991). A theory of inferred causation. In *KR*, (pp. 441–452).

Peng, J., Wang, P., Zhou, N., & Zhu, J. (2009). Partial correlation estimation by joint sparse regression models. *American Statistical Association*, *104*(486), 735–746.

Roverato, A. & Rocca, L. L. (2006). On block ordering of variables in graphical modelling. *Statistics*, *33*, 65–81.

Saitoh, T. & Katoh, M. (2002). Expression and regulation of wnt5a and wnt5b in human cancer: Up-regulation of wnt5a by tnfα in mkn45 cells and up-regulation of wnt5b by β-estradiol in mcf-7 cells. *International Journal of Molecular Medicine*, *10*(3).

Sakanaka, C., Leong, P., Xu, L., Harrison, S., & Williams, L. (1999). Casein kinase iepsilon in the wnt pathway: regulation of beta-catenin function. *Proceedings of the National Academy of Sciences of the United States of America*, *96*(22), 12548–12552.

Saviozzi, S. & Calogero, R. A. (2003). Microarray probe expression measures, data normalization and statistical validation. *Comparative and Functional Genomics*, *4*, 442–446.

Schäfer, J. & Strimmer, K. (2005a). An empirical Bayes approach to inferring large-scale gene

association networks. *Bioinformatics*, *21*(6), 754–764.

Schäfer, J. & Strimmer, K. (2005b). A shrinkage approach to large-scale covariance matrix estimation and implications for functional genomics. *Statistical Applications in Genetics and Molecular Biology*, *4*(1).

Schlecht, U., Demougin, P., Koch, R., Hermida, L., Wiederkehr, C., Descombes, P., Pineau, C., JÃ©gou, B., & Primig, M. (2004). Expression profiling of mammalian male meiosis and gametogenesis identifies novel candidate genes for roles in the regulation of fertility. *Mol. Biol. Cell*, *3*(15), 1031–1043.

Scott, L. A., Vass, J. K., Parkinson, E. K., Gillespie, D. A. F., Winnie, J. N., , & Ozanne, B. W. (2004). Invasion of normal human fibroblasts induced by v-fos is independent of proliferation, immortalization, and the tumor suppressors p16ink4a and p53. *Mol Cell Biol*, *4*(24), 1540–59.

Semenov, M. V., Tamai, K., Brott, B. K., Kuhl, M., Sokol, S., & He, X. (2001). Head inducer dickkopf-1 is a ligand for wnt coreceptor lrp6. *Current Biology*, *11*(12), 951–961.

Shapiro, S. S. & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, *52(3-4)*, 591–611.

Silander, T., Kontkanen, P., & Myllymäki, P. (2007). On sensitivity of the MAP Bayesian network structure to the equivalent sample size parameter.

Spink, K., P, P., & WI, W. (2000). Structural basis of the axin-adenomatous polyposis coli interaction. *EMBO*, *19*(10), 2270–2279.

Spirtes, P., Glymour, C., & Scheines, R. (2000). *Causation, prediction, and search* (2nd ed.). The MIT press.

Tanaka, K., Okabayashi, K., Asashima, M., Perrimon, N., & Kadowaki, T. (2000). The evolutionarily conserved porcupine gene family is involved in the processing of the wnt family. *European Journal of Biochemistry*, *267*(13), 4300–4311.

Thorstensen, L. & Lothe, R. A. (2003). The wnt signaling pathway and its role in human solid tumors.

Tikhonov, A. & Arsenin, V. (1977). *Solutions of Ill-Posed Problems*. Wiley, New York.

Tirosh, I. & Barkai, N. (2007). Comparative analysis indicates regulatory neofunctionalization of yeast duplicates. *Genome Biology*, *8*(4), 8:R50.

Tsuchiya, T., Dhahbi, J. M., Cui, X., Mote, P. L., Bartke, A., & Spindler, S. R. (2004). Additive regulation of hepatic gene expression by dwarfism and caloric restriction. *Physiol Genomics*, *3*(17), 307–15.

Vapnik, V. (1998). *Statistical learning theory*. Wiley, New York.

Waddell, P. J., Kishino, H., & Kishino, P. J. W. H. (2000). Cluster inference methods and graphical models evaluated on nci60 microarray gene expression data. *Genome Informatics*, *11*, 129–140.

Wang, H.-X., Tekpetey, F. R., & Kidder, G. M. (2009). Identification of wnt/β-catenin signaling pathway components in human cumulus cells. *Mol. Hum. Reprods*, *15*(1), 11–17.

Weaver, D. C., Workman, C. T., & Stormo, G. D. (1999). Modeling regulatory networks with weight matrices. In *In Proceedings of the Pacific Symposium on Biocomputing*, volume 4, (pp. 112–123).

Webb, E. & Westhead, D. (2009). The transcriptional regulation of protein complexes; a cross-species perspective. *Genomics*, *94*(6), 369–376.

Willert, K., Brink, M., Wodarz, A., Varmus, H., & Nusse, R. (1997). Casein kinase 2 associates with and phosphorylates dishevelled. *The EMBO*, *16*(11), 3089–3096.

Wit, E. & McClure, J. (2004). *Statistics for Microarrays: Design, Analysis and Inference*. Wiley.

Wolfe, C. J., Kohane, I. S., & Butte, A. J. (2005). Systematic survey reveals general applicability of "guilt-by-association" within gene coexpression networks. *Artificial Intelligence Applications and Innovations*, *6:227*.

Wolfinger, R. D., Gibson, G., Wolfinger, E. D., Bennett, L., Hamadeh, H., Bushel, P., Afshari, C., & Paules, R. S. (2001). Assessing gene significance from cDNA microarray expression data via mixed models. *Journal of Computational Biology*, *8*(6), 625–637.

Wu, X., Ye, Y., & Subramanian, K. R. (2003). Interactive analysis of gene interactions using graphical Gaussian model. In *ACM SIGKDD Workshop on Data Mining in Bioinformatics*, volume 3, (pp. 63–69).

Xie, Y., Wang, X., & Story, M. (2009). Statistical methods of background correction for illumina beadarray data. *Bioinformatics*, *25*(6), 751–757.

Yang, Y. H., Dudoit, S., Luu, P., Lin, D. M., Peng, V., Ngai, J., & Speed, T. P. (2002b). Normalization for cdna microarray data: a robust composite method addressing single and multiple slide systematic variation. *Nucleic Acids Research*, *30*(4), e15.

Yukhananov, R. & Loguinov, A. Introduction to microarray technology. sims2003. Summer Institute in Mathematical Studies.

Zafiropoulos, E., Maglogiannis, I., & Anagnostopoulos, I. (2006). A support vector machine approach to breast cancer diagnosis and prognosis. *Artificial Intelligence Applications and Innovations*, *204*, 500–507.

Zhang, J. (2003). Evolution by gene duplication : an update. *Trends in Ecology and Evolution*, *18*(6), 292–8.

Index

Citation Index