Uncertainty estimation for QSAR models using machine learning methods

By

Christina Maria Founti

A study submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy

The University of Sheffield

Information School

September 2019

Acknowledgements

In what seems like the end of a four-year long race there are many people I would like to thank for helping me reach the finish line.

First and foremost, I want to thank my supervisors Professor Val Gillet and Jonathan Vessey for the guidance, unlimited patience and support over the last four years.

I am very thankful to Dr. Dave Evans and the computational chemistry and chemoinformatics group at Eli Lilly, including Prashant Desai and Suntara Cahaya from the US division, for their useful advice and kindness during my 6-month secondment. I would also like to extend my thanks to the D3i4AD consortium administration team for making this secondment possible. Additional thanks go to my colleagues at Lhasa Limited for their kindness and help during my short placement.

I would also like to express my gratitude to Professor Peter Willet and the Sheffield Chemoinformatics group for their helpful advice and constant encouragement. Particular thanks to my colleagues who have been a great source of inspiration and motivation: Matt, Lucy, Antonio, Giammy, Jess and James.

Thank you to all my friends in the department who have made these four years a lot more pleasurable and fun: Mengdie, Sukaina, Soureh, Sally, Wasim, Marc and Alex. I'll miss our badminton matches, coffee meetups and dinners. I'm also very appreciative of previous and current members of the iSchool admin team for always shedding the light in any glooming question related to all the paperwork.

I'm very thankful to my extended family and friends located at various places in the world who provided me with moral support even from afar. Special thanks go to my housemates and the friends I've met in UK who have helped me stay sane at difficult times and feel like family by now. Particularly: Serena, Alexandros, Raquel, Barby, Chiara, Leandro, Roxana, Hannuun and Elli.

More than anyone I would like to thank my parents for their love, support and encouragement in everything I do. Without their help I would never have made it until here. Words are not enough to express my gratitude and thanks to Lorenzo for his unlimited support from the first day in this journey.

Finally, this work would not have been possible without the financial support of the BBSRC, the European Union's Seventh Framework Programme, Lhasa Limited and Eli Lilly.

Abstract

Providing safe, timely and affordable treatments is a major challenge addressed by big pharma. An important computational technique that is established in risk assessment as an alternative method to animal testing is Quantitative Structure-Activity Relationship (QSAR) modelling. In drug discovery, QSAR models are utilised to predict the properties of new compounds, thus reducing the number of tests required and associated risks of potential side effects leading to high costs and drug attrition. Yet, their value is limited in the absence of information regarding the reliability of their predictions.

The current research contributes to the understanding of limitations associated with uncertainty estimation methods for QSAR models and their implications on the validation of Absorption, Distribution, Metabolism and Excretion (ADME) models. The aim of this thesis is to investigate the value of machine learning algorithms in the estimation of errors in QSAR models and report on their performance for different ADME endpoints.

The study focuses on the evaluation of error models as a method for identifying poorly predicted compounds and estimating the uncertainty of QSAR predictions. Assessment of the models takes into account the correlations of the error estimates to the actual prediction errors and the magnitude of the error estimates in relation to the experimental error. The error models are then integrated in the conformal prediction framework for the estimation of compound-specific prediction intervals. For this purpose, a new normalisation method that combines error models and applicability domain features is defined. The results of the assessment suggest that the performance of error models is influenced by the quality of the QSAR model and the presence of measurement bias in the modelled ADME data. It is shown that considering different types of features in the error models provides a flexible approach for optimising not only the efficiency of prediction intervals but also ensuring that they are correlated to the actual prediction error.

Table of Contents

Table of Figures

List of Tables

Chapter 1 Introduction

In the present age of Big Data, machine learning algorithms have emerged as an indispensable tool for the exploration of complex data across many domains. In chemoinformatics, machine learning algorithms are applied to mine chemical databases and identify trends in experimental data that may be exploited for chemical property prediction, particularly, in the study of Quantitative Structure-Activity Relationships (QSAR) and Absorption, Distribution, Metabolism and Excretion (ADME) property prediction. These techniques are of primary interest in the pharmaceutical industry where they are used to guide drug development, but also in the regulation of chemicals for the assessment of risk. Despite their widespread use, the use of machine learning algorithms has been criticised due to the lack of reliability estimates for their predictions. A range of approaches for the estimation of reliability in QSAR predictions are available, yet, there is no consensus on a single best approach: one method involves estimating the errors of a prediction model using a second model, i.e., an error model.

This thesis aims to investigate the performance of error models and assess whether these are useful for the estimation of uncertainty in physicochemical and ADME property models. The focus is on evaluating error models in the detection of poorly predicted compounds and the estimation of uncertainty in QSAR predictions with confidence. The objectives of this study are threefold: 1) to assess the predictive performance of regression error models, 2) to evaluate the utility of error models in the detection of prediction error outliers and 3) to evaluate the utility of error models as methods in confidence estimation. The contents of each chapter are outlined below.

Chapter 2 provides a brief introduction to chemoinformatics and defines basic concepts such as molecular similarity and the representation of molecules. It also discusses the use of chemoinformatics techniques that are applied to support the drug discovery process.

Chapter 3 introduces the theoretical assumptions of QSAR modelling, the main components of a model and the guidelines for the development of a standard modelling workflow. It discusses the importance of statistical techniques for the purpose of model optimisation and validation, and introduces the standard measures used to assess model performance. The main methods for estimating the reliability of QSAR predictions using the concept of the applicability domain is reviewed and an introduction to confidence estimation methods applied in QSAR is provided.

Chapter 4 describes the datasets that were used to build the underlying QSAR models and error models that were investigated the experimental chapters that follow.

Chapters 5 to 7 describe the experimental work carried out in this thesis and report the results of the investigations. Chapter 5 describes the QSAR modelling workflow and presents the underlying models that were used as a base in the investigations of the following chapters. Chapter 6 presents the error models that were built for the estimation of prediction errors of the underlying models' predictions. The error estimates are analysed and evaluated for their ability to rank predictions based on their true accuracy. The size of the error estimates is also assessed while taking into account the experimental error of the data. This is done by applying an information theoretic framework, which requires that measurements and predictions are represented as probability distributions. In Chapter 7 conformal prediction is applied to estimate prediction intervals using the estimates of error models and their results are analysed to evaluate the utility of error models in confidence estimation.

Finally, Chapter 8 summarises the conclusions of this work, outlines the limitations and makes suggestions for future investigations.

Chapter 2 Chemoinformatics

2.1 Introduction

Chemoinformatics is focused on the development of computational methods that address chemical problems and facilitate decision-making processes in the chemical and related industries. The emergence of the term dates to the late 1990s and is linked to the technological advances of the time that resulted in increasing chemical data generated by the industry and the research community (Engel, 2006). Yet, important statistical techniques had already been developed for the study of organic chemical structures from as early as the '30s (Fujita & Winkler, 2016).The invention of computers in the 1940s meant that they would be available for research by the 1950s. During that decade, a lot of work focused on documentation, such as the development of methods for archiving, processing and centralising collections of data of the Chemical Abstracts Service (CAS) (Powell, 2000; Willett, 2008). The storage and retrieval of chemical information in databases were among the first challenges addressed, which helped establish fundamental concepts and methods for the representation of chemical compounds in a machine-readable format by the end of the decade. The next three decades were followed by the development of computational methods for the analysis of substructures in chemical databases and their extension to more advanced applications, such as the development of structure-activity relationships and chemical expert systems for automated structure elucidation and computer-aided synthesis (Engel, 2006; Leach & Gillet, 2007). By the early 1990s, advancements in biotechnology and high-throughput screening technologies contributed to the development of molecular modelling and structure-based virtual screening techniques, which created new opportunities for the discovery and development of new drugs (Lavecchia & Di Giovanni, 2013; Powell, 2000).

This chapter is a brief introduction to the basic concepts of molecular similarity and molecular representation in chemoinformatics. It also discusses how chemoinformatics has contributed to the development of the modern drug discovery process and illustrates the types of problems that chemoinformatics techniques aim to address.

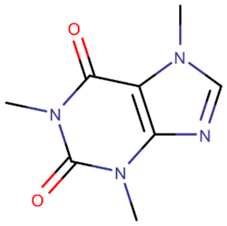
2.2 Similarity-property principle

A concept that is widely used in chemoinformatics applications is that molecules with similar structures are likely to exhibit similar chemical properties (Johnson, Basak, & Maggiora, 1988). This is more widely known as the similarity property principle (Maggiora, Vogt, Stumpfe, & Bajorath, 2014) and its implications extend to the interactions of molecules in biological systems and their ability to bind to biological targets. As a result, the concept of molecular similarity is widely applied, for example, in similarity searching to search for molecules that are similar to a known compound with a desired property; or in quantitative structure-activity relationships to predict the activity of new compounds that are structurally similar to other compounds that have been tested. However, the definition of similarity is not trivial as it depends on the molecular representation selected to investigate a structure-activity relationship (Nikolova & Jaworska, 2003). A main limitation of the similar property principle is that it not continuous, as it is too simple to explain more complex, chemical interactions (Bender & Glen, 2004). This introduces discontinuities in the structure-activity relationship caused by molecules with similar structures but significantly different bioactivity values (Maggiora, 2006). These discontinuities are referred to as activity cliffs and, recently, they have been exploited to understand the limitations of structure-activity relationships and how these may be best utilised in the structural optimisation of new compounds(Cruz-Monteagudo et al., 2014; Stumpfe, Hu, Dimova, & Bajorath, 2014)).

2.3 Molecular representation

In chemistry, molecular structure is encoded using three main formats: structure diagrams, molecular formulas and systematic names. Structure diagrams are the most frequently used and information rich representation of the three, as they illustrate the topological arrangement of atoms and bonds in the structure using chemical symbols and lines. This representation enables chemists to estimate molecular properties (e.g. electrostatic) empirically or theoretically based on their intuition of inter- or intra-molecular interactions of the atoms using pencil and paper. Chemical diagrams also represent information that is often implicit to the structure and can do this via formalisms, e.g. d-, l- stereochemistry, which may be easily interpreted by a chemist but more difficult to interpret by a machine. Molecular formulas provide a summary of the atom count and atom types but can be ambiguous, as the same molecular formula may be used to describe more than one chemical structure (Table 2-1). More than one chemical name may be available to describe a single structure and even though a systematic, unambiguous name of a structure may be defined, it may only be intuitive to chemists.

Table 2-1. Example of chemical nomenclature for caffeine

Graph	
Systematic Name	1,3,7-trimethyl-2,3,6,7-tetrahydro-purine-2,6-dione
Chemical Name	1,3,7-trimethylpurine-2,6-dione 1,3,7-trimethylxanthine Caffeine
Molecular formula	$C_8H_{10}N_4O_2$

2.3.1 Chemical representation in computers

Various techniques that translate chemical representations to machine-readable formats have been developed, which facilitate the storage and processing of chemical information in computers. The methods that are more widely used are mathematical graphs, connection tables and linear notations; although these are used mainly in the representation of small organic molecules. More specialised representations and notations are available for more complex structures, such as proteins, polymers, mixtures and inorganic molecules.

Mathematical graphs and connection tables. Mathematical graphs may be used to define two-dimensional structural diagrams where atoms and bonds are represented, respectively, by nodes and edges (Engel, 2006). The graphs preserve the topology of the structure and atom information but also strictly adhere to the valence connectivity rules that apply to chemical structure diagrams. The properties of mathematical graphs have been utilised in the development of search algorithms for the identification of substructures and isomorphic structures in chemical databases as well as the development of indices that summarise the molecules' topology in structure-activity relationship studies.

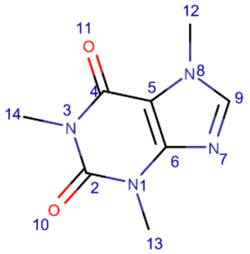
A connection table may encode two- or three-dimensional structural information of a molecule in tabular form. A simple connection table consists of two main sections whereby the first is a list that enumerates all the atoms present in the structure and the second is a table that enumerates the atoms' bonds to other atoms (Leach & Gillet, 2007). More detailed connection tables contain additional properties, such as the atoms' coordinates, bond order, stereochemistry centres or atom charge. The most widely used connection table representations are MDL's Molfile and the SDF format.

Linear notations. Linear notations translate connection tables into a string representation following a set of rules. This representation was developed to support fast search, transmission and compact storage of large collections of molecules in chemical databases. One of the earliest linear notations used, from the mid-60s until the 80s, was the Wiswesser Line Notation (WLN). The WLN notation represented molecular structures as a string of letters; and each letter represented a structural fragment. It was then replaced by the Simplified Molecular Input Line Entry System (SMILES) notation, which was easier to interpret even by non-experts in the field (Xu & Hagler, 2002).

The SMILES strings represent atoms by their atomic symbols; with uppercase characters denoting that they are aliphatic and lowercase characters that they aromatic. Hydrogen atoms are supressed, as in all representations, unless they belong to functional groups, which are enclosed in brackets. The notation also uses special characters to encode charges, bond order, chirality and isomers. Rings represented by attaching a number to the atoms of a ring-opening bond and branches are enclosed in parentheses (Leach & Gillet, 2007; Weininger, 1988). An important requirement for the storage and retrieval of the correct structures in database systems is that there should correspond a unique and unambiguous string representation to each molecular structure (Xu & Hagler, 2002). This is done through canonicalization, which assigns atoms of a molecular graphs a unique order. The Morgan algorithm is a well-known method for applying canonicalization, which orders the atoms by calculating their connectivity value over a number of iterations (Leach & Gillet, 2007). Atoms are ordered by descending connectivity values and ties are dealt with by taking into account the atomic number and the bond order. Canonical SMILES that are unique for each molecule are generated in a similar way using the CANGEN algorithm (Leach & Gillet, 2007).

The International Chemical Identifier (InChI) was developed by the International Union of Pure and Applied Chemistry (IUPAC). It was mainly developed for the purpose of establishing a universal identifier and to address unresolved issues of SMILES strings regarding stereochemistry and tautomers. Each unique, chemical identifier is a canonical alphanumeric string that encodes the structural information of the structure in 'layers' of substrings i.e., constitution, charge, fixed hydrogens, stereochemistry, isotopes and the reconnected layer. As standard InChI strings increase in length and become more complex as molecule size increases, they can be replaced with standard InChI keys, which are compact chemical identifiers generated by hashing (Heller, McNaught, Pletnev, Stein, & Tchekhovskoi, 2015).

Table 2-2. The connection table representation for caffeine in MDL Molfile format generated using Marvin Sketch (ChemAxon, 2016) and linear notations retrieved from DrugBank (Wishart et al., 2018).

Connection table	Mrv16b2109051917443D 14 15 0 0 0 0 999 V2000 -4.8401 -1.7046 0.0000 N 0 0 0 0 0 0 0 0 0 1 0 0 -6.3465 -1.3844 0.0000 C 0 0 0 0 0 0 0 0 0 2 0 0 -6.8224 0.0802 0.0000 N 0 0 0 0 0 0 0 0 0 3 0 0 -5.7919 1.2247 0.0000 C 0 0 0 0 0 0 0 0 0 4 0 0 -4.2856 0.9045 0.0000 C 0 0 0 0 0 0 0 0 0 5 0 0 -3.8097 -0.5601 0.0000 C 0 0 0 0 0 0 0 0 0 6 0 0 -2.2697 -0.5601 0.0000 N 0 0 0 0 0 0 0 0 0 7 0 0 -3.0397 1.8097 0.0000 N 0 0 0 0 0 0 0 0 0 8 0 0 -1.7938 0.9045 0.0000 C 0 0 0 0 0 0 0 0 0 9 0 0 -7.3770 -2.5288 0.0000 O 0 0 0 0 0 0 0 0 0 10 0 0 -6.2678 2.6893 0.0000 O 0 0 0 0 0 0 0 0 0 11 0 0 -3.0397 3.3497 0.0000 C 0 0 0 0 0 0 0 0 0 12 0 0 -4.3643 -3.1692 0.0000 C 0 0 0 0 0 0 0 0 0 13 0 0 -8.3287 0.4004 0.0000 C 0 0 0 0 0 0 0 0 0 14 0 0 1 2 1 0 0 0 0 2 3 1 0 0 0 0 3 4 1 0 0 0 0 4 5 1 0 0 0 0 5 6 2 0 0 0 0 1 6 1 0 0 0 0 8 9 1 0 0 0 0 7 9 2 0 0 0 0 7 6 1 0 0 0 0 5 8 1 0 0 0 0 2 10 2 0 0 0 0 4 11 2 0 0 0 0 8 12 1 0 0 0 0 1 13 1 0 0 0 0 3 14 1 0 0 0 0 M END  12 11 O N8 5 9 4 14 3 N N 7 6 2 N1 O 10 13
Canonical SMILES	CN1C=NC2=C1C(=O)N(C(=O)N2C)C
InChI	InChI=1S/C8H10N4O2/c1-10-4-9-6-5(10)7(13)12(3)8(14)11(6)2/h4H,1-3H3
InChI Key	RYYVLZVUVIJVGH-UHFFFAOYSA-N

2.3.2 Molecular descriptors

A descriptor is a numerical representation of one or more features in the molecule's structure and may be the output of logical or mathematical operations applied to the molecular representation or an empirical estimate from experimental data. More than 5,270 molecular descriptors, which are only included in the latest version of DRAGON software(Todeschini & Consonni, 2009), have been developed to convey the chemical information that is present in a molecular structure and have a wide scope of application. However, only few descriptors may be relevant to a specific structure-property relationship, while others may contain redundant information. The relevant descriptors are identified using descriptor selection techniques and are encoded as a set of numerical values for each molecule, i.e., a descriptor vector, that can be processed by computational algorithms to make predictions or perform similarity search. An overview of description selection methods is provided in the following chapter.

There are different ways of organising the molecular descriptors. Based on scope, descriptors may be classified into global or local, as they describe a property of the whole molecule or a property of its structural components, i.e., fragments, respectively. Fragment descriptors may also be combined to compute the property of larger fragments or the whole molecule (Kubinyi, 1993b; Leach & Gillet, 2007), an important technique in the development of QSAR. Based on the dimensionality of the structural information encoded, they may be classified into: a) 0D, which include molecular weights and counts of atoms and bonds, b) 1D, which consist of functional and fragment counts, c) 2D, which are calculated from two-dimensional molecular graphs, d) 3D, which are generated from the three-dimensional structure and structural conformation of the molecule and e) 4D, which take into account multiple structural conformations. A more detailed description is provided below on 1D and 2D descriptors that are widely used in the study of QSARs.

Constitutional. The simplest and most rapid descriptors to calculate for a molecule are the counts of atoms, bonds and rings. These can be calculated from the 2D connection table. The number of heteroatoms, hydrogen bond donors and acceptors are indicators of the overall binding capacity of a molecule, as they may be utilised to form intramolecular or intermolecular interactions. The number of multiple bonds, rotational bonds and aromatic rings carry information about bonding capacity, flexibility and the overall volume of the structure. These descriptors have a low discriminating power among molecules and are frequently used along with other types.

Topological indices. The representation of molecules as topological graphs makes it possible to encode their structural information in matrices from which topological indices (TIs) that describe the size, connectivity and shape of the molecule may be derived (Leach & Gillet, 2007; Winkler, 2002). TIs were first introduced in work of Wiener in the late 1940s and later evolved into molecular descriptors for structure-property relationships in diverse datasets (Katritzky & Gordeeva, 1993). The first generation TIs are integer indices and characterise the molecules' branching and composition. A characteristic example is Wiener's index, also referred to as the path number, which is defined as the sum of bonds for all atom pairs and was designed to correlate with the boiling point of alkanes. A limitation of first generation TIs is that the same value may be calculated for more than one molecule (Balaban, 1995), thus, not allowing their structures to be differentiated. Second generation TIs consist of real numbers and take into account the degree of atom connectivity, i.e., the number of atoms attached to the bond. The best known example is Randić's molecular connectivity index, which encodes the sum of the bond connectivities of all atoms (Leach & Gillet, 2007). This was later generalized by Kier and Hall's valence connectivity indices, which include longer atom paths and heteroatoms with additional valences. Another example is Balaban's average connectivity index, which encodes the presence of cyclic structures (Katritzky & Gordeeva, 1993). Third generation indices are derived from complex matrix operations and include Kier and Hall's electrotopological indices (E-state), which encode information on atom valence and sigma electrons, thus, encoding the influence of atoms that are more distant (Winkler, 2002) .

Fingerprints. Fingerprints represent molecular structure as bitstrings that encode the presence or absence of substructures in a molecule by setting the bits to 0 or 1, respectively. These were originally developed for fast screening and similarity searches in chemical databases but have been redefined for use as descriptors in QSAR.

They may be generated following the definition of substructure dictionaries or rules to identify all the possible patterns of atoms in a structure and are classified into three, main fingerprint systems: structural keys, hashed fingerprints and circular fingerprints.

Structural keys are bit string dictionaries that encode specific functional groups, ring systems, heteroatoms or other structural features. This means that other structural features that are present in the molecules will not be encoded. They are defined by applying substructure searches in compound libraries. As the structural features are assigned to specific bit positions, the information encoded in the set bit positions may be used to aid the interpretation of structure-property results and identify important structural features (Leach & Gillet, 2007). Two well-known dictionary systems are the MDL and the BCI structural keys. The MDL Molecular ACCess System (MACCS) dictionary contains the definition of 960 keys, which encode the nature of the atoms, bonds and the atom environment of specific structural features. Another MDL set of 166 keys encodes the atom properties.(Durant et al., 2002). Focused dictionaries that encode special features of interest may also be defined using the BCI fingerprint system, which allows the definition of structural keys by the user based on a specific or non-specific definition of atoms and bonds (Cereto-Massagué et al., 2015; Leach & Gillet, 2007).

Hashed fingerprints encode all the structural features that are present in the molecule. First, an algorithm exhaustively generates linear paths through the structure of molecules; then, a hashing algorithm encodes these paths into the bits of the fingerprint representation. Unlike structural keys, they are not interpretable as the set bit positions cannot be mapped to the structural features (Leach & Gillet, 2007). Daylight fingerprints are generated by producing exhaustive lists of linear paths with varying length for every structure. The presence of these paths is encoded in a molecular fingerprint of fixed length that may be sparsely populated, i.e., fewer bits are set to 1. As information density increases with molecular size and complexity, the fingerprint typically requires optimization through the process of folding ("Daylight Theory Manual," 2011). For Unity fingerprints, paths of varying size in a structure are identified and encoded in different fingerprint regions. Certain fragments can also be encoded in the form of ASCII strings. Unity fingerprints have a size of 988 bits out of which 928 bits capture the presence of the defined paths in the compound, while the rest encode specific atom types, rings and their frequency of occurrence (Wild & Blankley, 2000).

Unlike structural keys and hashed fingerprints that were developed for substructure screening applications, circular fingerprints were developed to encode important structural features for SAR studies (Rogers & Hahn, 2010). Furthermore, instead of relying on predefined structures, circular fingerprints encode structural information on the local neighbourhood of atoms in the molecule (Glen et al., 2006). Extended connectivity fingerprints (ECFPs) treat atoms as centres of concentric neighbourhoods and encode information about the connectivity of atoms and their environment by examining a neighbourhood within a specified radius. ECFPs are derived by assigning identifiers to all atoms using a variation of the Morgan algorithm and a hashing function. Depending on the required detail of their application, ECFPs can be optimized for the neighbourhood radius, fingerprint size and frequency of occurrence for each identifier. The functional class fingerprint (FCFP) is a variation of ECFP that encodes features of pharmacophores(Rogers & Hahn, 2010). This allows for atom types that are identified as halogens, hydrogen bond donors or acceptors, aromatic or ionisable in a structure to be encoded using a non-specific identifier that is representative of its functional class.

Atom-Pair descriptors. An atom pair is defined as the shortest path containing two non-hydrogen atoms and is measured by the number atoms that are present in the bonded path (Leach & Gillet, 2007). It may also be extended to include a wider range of atoms. The information encoded in atom pair descriptors includes atom type, hybridization, structural environment and substructure size. Atom pair descriptors were originally developed to encode structural features from high dimensional representations for SAR studies without the requirement of complex mathematical transformations but are also widely used in similarity searching (Carhart, Smith, & Venkataraghavan, 1985).

Physicochemical properties. Different to the topological descriptors which are calculated directly from the molecules' structure, physicochemical descriptors are estimated, empirically, from available experimental data (Katritzky & Gordeeva, 1993). Widely used molecular descriptors include properties such as lipophilicity and molar refractivity, which are described below. Other physicochemical descriptors include molecular weight, molecular volume and molecular surface area may be estimated using fragment-, atom- or property-based methods.

The logarithm of the octanol-water partition coefficient, LogP, is one of the most experimentally accessible properties that is used to model lipophilicity with many available implementations for its estimation (Mannhold, Poda, Ostermann, & Tetko, 2009). It is directly associated to other properties, such as aqueous solubility and membrane permeability (Kubinyi, 1993a). Well-known methods for the estimation of LogP include: The CLogP program, which calculates LogP values by adding the empirical LogP values of core fragments and, then, applies a set of correction factors for intermolecular interactions. Ghose and Crippen's method estimates LogP as the sum of its atom contributions, which are estimated from a regression model that consists of the contribution of 115 atoms types to the lipophilicity of approximately 8.3 thousand compounds. A variation of this method is based on a linear function of the number of atoms and correction factors for rigid, non-lipophilic carbons and intramolecular hydrogen bonds (Hou & Xu, 2003). Another approach for the estimation of LogP by Moriguchi et al. uses a structure-property regression model based on the number of hydrophobic and hydrophilic atoms as descriptors (Moriguchi, Hirono, Liu, Nakagome, & Matsushita, 1992).

Molar refractivity is a widely used physicochemical parameter, which is an expression of the overall polarizability of a molecule. In QSAR studies, it has been correlated with other additive properties, like lipophilicity, molar volume and steric bulk. These have shown that molar refractivity may be used to predict the binding of a structure to a polar surface or its steric hindrance to the binding site of a receptor. Available methods for its estimation are mainly atom- and fragment-based (Kubinyi, 1993a).

2.4 Chemoinformatics in drug discovery

In the early 1990s, the appearance of new computational tools and automated processes in the pharmaceutical industry introduced changes in the traditional drug discovery pipeline (Figure 2-1). Delays caused at the early stages of drug discovery, particularly in the identification of biologically active molecules against a disease target, severely slowed down the entire process and increased the cost of drug development candidates (Xu & Hagler, 2002).

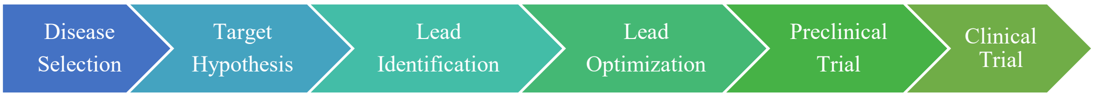


Figure 2-1. Traditional drug discovery pipeline

New strategies implemented were aimed at making the synthetic and screening processes more efficient for the discovery of lead compounds. These included the deployment of high throughput screening (HTS) technologies, which had been previously been used only as initial screens for potential drug candidates in the companies' existing records, in a process referred to as 'hit-to-lead'. The hit-to-lead optimisation process is iteratively applied and involves the screening of a chemical library, the validation and prioritisation of the hits identified, as well as the application of structure-activity relationships for the identification of a lead series (Duffy, Zhu, Decornez, & Kitchen, 2012).

The emergence of combinatorial chemistry technologies permitted the synthesis and testing of thousands of molecules in parallel, thus, significantly reducing the amount of time to produce new compounds. The application of these techniques resulted in an explosion of assay data and demanded the development of new, data-driven methods to deal with the analysis of the increasingly large amounts of data (Bajorath, 2018). However, this came with the realisation that the majority of hits obtained from parallel screens were not useful, as they did not have suitable absorption, distribution, metabolism, excretion and toxicology (ADMET) properties. The problem was addressed with the development of chemical-diversity based methods in chemoinformatics at the time. These were focused on the design of diverse compound libraries for screening experiments with the aim of increasing the number of hits and identifying new lead series (Xu & Hagler, 2002).

Nevertheless, the implementation of computer-aided technologies in the drug discovery process has not reduced the time or the costs for the development of a new drug nor have they been able to predict failures due to clinical toxicity (Ekins et al., 2019). The average time taken to move a drug from the early stages of research to the market is 10-12 years with an average cost estimate of 2.56 billion dollars (DiMasi, Grabowski, & Hansen, 2016). Thus, there is interest in identifying compounds that are likely to fail early, while the cost of failure is still low. Computational techniques may be applied prior to experimental testing to eliminate molecules with low predicted bioactivity or poor ADMET and toxicity properties that would make them more likely to fail during clinical trials. Other techniques, such as virtual screening and molecular modelling, may be used to study the drug-target interactions and guide the optimisation process(Firdaus Begam & Satheesh Kumar, 2012).

The study of molecular structure features and their association to a biological phenomenon, also known as structure activity relationship (SAR), is a computational tool that is used for the prediction of active structures. When the correlation of features and measured activity is good and an accurate numerical prediction may be derived then this may be referred to as a QSAR.

The role of QSAR methods as an alternative to assay-based methods for the prediction of the biological activity and ADMET properties of compounds has been recognised (ECHA, 2016). QSAR predictions for compounds with no available experimental data may be used as an initial screen of large chemical libraries to support the prioritisation of compounds with low risk property profiles for testing. Further to their predictive role, QSAR models may also be used to interpret mechanisms of bioactivity based on the structural features of molecules. Examples of such models have been applied to identify novel structures and structural analogues (Guha & Jurs, 2004), as well as to augment data where measurements in experimental data are unavailable (Papa, Kovarich, & Gramatica, 2009). Finally, they provide a solution towards the reduction of animal testing in toxicological studies and constitute an alternative, non-testing method for the assessment of risk in chemical substances for regulatory purposes (REACH) (Tetko et al., 2008).

2.5 Conclusions

This chapter has introduced the field of chemoinformatics and presented a fundamental concept that has been exploited for the development of many chemoinformatics techniques applied in drug discovery, namely the similarity-property principle. However, the principle is merely an abstraction and may be invalidated by considering alternative methods of molecular representation. Yet many descriptors have been developed to encode structural features and physicochemical properties of molecules, which facilitate the analysis and organisation of compound collections using computational methods. The implementation of chemoinformatics in drug discovery and drug development aims to aid the discovery of new chemical entities and their optimisation into useful leads and drug candidates. In particular, QSAR studies are widely applied to guide decision making at the early stages of drug discovery, i.e., in screening experiments, but also to optimise the biological activity and ADMET properties that are responsible for the failure of drugs during clinical trials. A detailed account on the development and application of QSAR methods is provided in the following chapter.

Chapter 3 Developing a QSAR model

3.1 Introduction

This chapter introduces the main concepts of QSAR and the requirements for the development and validation of a QSAR model. First, it outlines the available techniques for descriptor selection, then it describes the main theory of state-of-the-art machine learning algorithms and validation methods. The last section of this chapter introduces the concept of the applicability domain and the various ways that this has been applied to evaluate the reliability of future predictions.

3.2 Overview of QSAR

A QSAR model is a mathematical or statistical function that describes the dependence of the measured property, or biological activity, on the structural features of the molecules in a dataset (Kubinyi, 2002). The model is derived by applying statistical techniques to the dataset and may be used to predict the property or activity of untested molecules. Predictions are obtained as numerical estimates using regression methods or as labels using classification methods.

According to Fujita & Winkler (2016) QSAR modelling methods are divided into classical approaches and machine learning approaches. Classical approaches use simple, linear regression models to predict the relationship between the structure and biological activity of molecules. An example is Hansch analysis, which estimates the bioactivity of molecules additively as the sum of the molecules' electronic, hydrophobic and steric parameter contributions (Hansch, 1969). Free-Wilson analysis uses a similar approach where the bioactivity of a molecule is estimated by summing the bioactivities of the structural fragments (Free & Wilson, 1964). Due to their theoretical basis, these methods produce interpretable models that are able to explain which structural features contribute the most to the observed response (Fujita & Winkler, 2016). However, the scope of classical QSAR methods is local and the models' applicability domains are restricted to series of congeneric compounds with little structural variation.

Machine learning approaches use statistical algorithms to learn nonlinear structure-activity relationships between independent variables and the experimental response of large datasets. There are two broad machine-learning approaches: the frequentist, which treat experimental observations as random, repeatable events and infer an optimum model based on a maximum likelihood parameter estimate; and the Bayesian, which treat model parameters as random variables and learn the model from prior data (Varnek & Baskin, 2012). Frequentist methods report predictions as point estimates while Bayesian methods explicitly report the uncertainties of their predictions, which are useful in decision-making (Sahlin, 2015). Although Bayesian methods are computationally demanding, recent improvements in computational resources and the optimisation of Bayesian algorithms, such as neural networks, have increased their use in QSAR (Ma, Sheridan, Liaw, Dahl, & Svetnik, 2015). However, because these methods rely on large numbers of descriptors and often lack mechanistic transparency, obtaining interpretative models is not always feasible. Frequently used alternatives are nonparametric methods based on ensemble trees, such as Random Forest (RF), and kernel methods, such as Support Vector Machines (SVM). Note, that, only the former may produce uncertainty estimates for individual predictions, directly. Section 3.4 provides a more detailed account of the algorithms applied in this thesis.

The key components in the development of QSAR models are the quality of the data, the representation of structural features and the modelling algorithm, which determines how the similar property principle is applied. To achieve high accuracy for a QSAR model it is required that the data is of high quality, as model performance is limited by the accuracy of the experimental data. The molecular structures are typically represented as numerical descriptors or fingerprints indicating the presence or absence of structural features, however, these need to be relevant to the modelled property. The selection of important variables may be guided by expert knowledge or automated descriptor selection methods, which may also be embedded in the modelling algorithm itself. Prior to training the modelling algorithm, it may also be required that the descriptor ranges are scaled so that they contribute proportionally to the model. The modelling algorithm is then optimised using methods such as cross-validation and validated on external data, if available.

3.3 Descriptor selection

The representation of chemical data poses a challenge in the development of accurate QSAR models. As discussed in the previous chapter, the information present in a chemical structure may be encoded in numerical form as molecular descriptors or as fingerprints. Machine learning methods are efficient in handling large numbers of descriptors, yet, the risk of overfitting due to added noise in the form of non-relevant descriptors (Topliss & Edwards, 1979) and descriptors carrying redundant information is high (Danishuddin & Khan, 2016; Hawkins, 2004). Therefore, it is required that feature selection and dimensionality reduction techniques are applied for the removal of unnecessary variables.

In QSAR, the aim of feature selection is to identify the descriptors that drive the prediction of the target variable. Ideally, this process is implemented during cross-validation, or by descriptor sampling using resampling methods, to avoid introducing descriptor bias into the model (Tetko et al., 2008). Feature selection techniques may be applied as filters on the original set of descriptors, as wrappers to the modelling workflow, or they may be embedded in the modelling algorithm. Filter methods are independent of the modelling algorithm and have the advantage that they are easy to apply prior to model development. The descriptors are filtered using a relevance score that is based on, for example, the correlation with the target variable, the distance between the nearest neighbours in descriptor and target space (Robnik-Šikonja & Kononenko, 2003) or mutual information with other descriptors (Danishuddin & Khan, 2016). Wrapper methods make use of the model's error to evaluate descriptor relevance by sampling different subsets of descriptor combinations. These methods are model-specific and their effectiveness is influenced by the nature of the modelling algorithm and the number of descriptor subsets (Chrysostomou, Chen, & Liu, 2008; Danishuddin & Khan, 2016). Embedded feature selection methods are built-in to the modelling algorithm and, thus, specific to the algorithm's underlying assumptions.

An example of a filter is the Variable Importance in Projection (VIP) score that is obtained from the partial least squares algorithm. A VIP score represents the amount of variance that an individual descriptor explains in the model (Abdi, 2010). As the average of the VIP scores' sum of squares is equal to 1, the minimum threshold of 1 is, typically, applied for descriptor selection (Tran, Afanador, Buydens, & Blanchet, 2014). However, other methods for deriving robust thresholds have been suggested (Akarachantachote, Chadcham, & Saithanu, 2014) due to the sensitivity of this criterion to the underlying data distribution and its lack of theoretical justification.

Variable importance scores may also be obtained from Random Forests. These may be computed as the permutation accuracy importance score (Strobl, Boulesteix, Kneib, Augustin, & Zeileis, 2008), which measures the change in prediction accuracy caused by the random permutation of a variable; or the mean decrease in impurity (Menze et al., 2009), which measures the change in the prediction variance attributed to each variable.

Dimensionality reduction techniques, also referred to as feature extraction, aim to simplify the complex structure of a high dimensional space by applying linear or nonlinear transformations to the features (Jindal & Kumar, 2017). The result is a low dimensional space, typically of two to three variables, that preserves the most important information in the data without deteriorating the model's performance. Principal Component Analysis (PCA) applies a linear transformation to the independent variables to generate a new set of orthogonal, i.e., uncorrelated variables. Although PCA is efficient for data with linear underlying structures, it cannot handle data with more complex, nonlinear structures. These are effectively addressed using nonlinear dimensionality reduction techniques based on manifold learning (Gaspar, Baskin, & Varnek, 2016).

3.4 Modelling algorithms

The theory underlying four algorithms with widespread use in QSAR is outlined below with a focus on regression. These algorithms are representative examples of four machine-learning families, namely dimensionality reduction methods, nearest neighbour methods, kernel methods and ensemble tree methods.

3.4.1 Partial Least Squares

Partial Least Squares (PLS) regression is based on the principles of PCA and multiple linear regression. First, the PLS algorithm transforms both the independent (X) and dependent (Y) variables into a common set of orthogonal X-scores, also known as latent variables, while accounting for most of the covariance between X and Y. This step is followed by applying linear regression between the latent variables, T, and Y for the estimation of Y for new compounds. Equation 3.1 describes the decomposition of the original variable matrix, X, as a product of T and their weights (loadings), P. In Equation 3.2, the estimate of the independent variable vector, $\hat{Y}$, is obtained as the product of the new latent variables, the regression weights, B, and the weight matrix of X, C.

$$X=TP^T \qquad 3.1$$

$$\hat{Y}=TBC^T \qquad 3.2$$

It is generally the case that the first few latent variables encode most of the variation present in the original variables. As a result, high dimensional data may be reduced into a significantly smaller number of variables that contain most of the information present in their original representation. Although two to three latent variables are frequently selected in methods such as PLS, the optimal number of latent variables may be selected using cross-validation.

3.4.2 Nearest Neighbours

The K-Nearest Neighbours (KNN) algorithm is an instance-based learner that does not require learning of the mapping function. In regression, the algorithm memorises the training data and makes predictions for new data by calculating the weighted average response of their nearest neighbours in descriptor space. Nearest neighbours are usually determined in Euclidean space, although other metrics may be used based on domain knowledge about the training data distribution (Chomboon, Chujai, Teerarassammee, Kerdprasop, & Kerdprasop, 2015). The original version of the algorithm uses uniform weights that assign equal contributions to all neighbouring values. However, different weighting schemes may be introduced based on domain knowledge about the training set distribution in descriptor space (Anava & Levy, 2016). The simplest variation of the KNN uses a distance-based function, i.e., inverse square distance, to assign a higher contribution to the values of the neighbours closest to the test compound (Mitchell, 1997; Nigsch et al., 2006). Optimisation of the KNN algorithm is simple as it requires the parametrisation of a single parameter K, which is the number of nearest neighbours. For a small K, predictions will be biased towards the estimates of their local neighbourhood while for large K, where model predictions are estimated from overlapping neighbourhoods, the predictions will converge to the data mean. The selection of K may involve making a trade-off based on the intended purpose of the model. For example, a smaller K value may benefit the accuracy of a predictive model, while a mechanistic model aiming at the description of an overall trend may benefit from setting a larger K value (Altman, 1992).

The algorithm is sensitive to the presence of outliers and sparse regions in the data that cause the performance of KNN to deteriorate. Limitations of the algorithm include its inability to deal with unscaled descriptors, high dimensional data and skewed data distributions.

3.4.3 Support Vector Machine

The support vector machine (SVM) algorithm was originally introduced for the binary classification of labelled data. The algorithm uses a kernel function to map the training data into a high dimensional descriptor space where a linear separating hyperplane exists between the two classes. Although more than one hyperplane may exist, the optimal hyperplane is the one that yields the maximum separation of the two classes, thus, minimising the error of classification. A margin is defined by the ε-insensitive loss function, which controls the level of noise in the data that is tolerated by the model. The size of parameter ε determines the number of data points that will be used to fit the regression function. Another parameter, C, increases model complexity and corresponds to a larger number of support vectors and a harder margin that applies greater penalties to predicted output with large errors. These data points are known as the support vectors and consist of the training data that are closest to the hyperplane (Ivanciuc, 2007).

In SVM regression, the objective is to find a function that maximises the deviation ε for all training data from the experimental response (Ivanciuc, 2007). This involves introducing slack variables that account for the deviation of data points from each side of the linear hyperplane. During fitting of the regression function, the prediction errors of training data located inside the margin are set to zero, while the prediction errors of training data outside the margin are proportional to their distance from the boundaries.

The use of kernels in SVM makes the algorithm efficient in high dimensional spaces because of a mathematical property known as the kernel trick. Instead of evaluating the mapping function for every data point, the kernel trick allows its replacement by a dot product, which is easily computed between the test instances and each support vector. Consequently, rather than being trained on the data descriptors, the SVM algorithm is trained on the pairwise dot products of the data. Predictions are also obtained as dot products of the test data and the training data (Mahé, 2006).

A more complex separating hyperplane may be constructed using a nonlinear kernel. Standard nonlinear kernels applied in SVM include the polynomial, the sigmoid and the radial basis function. Kernels that are specific to the chemical domain are also available such as the graph, Tanimoto, pharmacophore and matched-molecular pair kernels (Lavecchia, 2015).

The radial basis function (RBF) is a Gaussian kernel function φ with $\varphi(x) = e^{-\gamma||x_i - x_j||^2}$ and $\gamma = \frac{1}{2\sigma^2}$, where x_i is the test feature vector, x_j is a support vector and σ controls the shape of the hyperplane (Ivanciuc, 2007). It is commonly used in regression due to its efficiency in computing the dot product in high dimensional descriptor space.

3.4.4 Random Forest

Random forest (RF) is widely used, particularly in the field of QSAR, for its ability to build robust models and its built-in mechanisms for descriptor selection and internal validation. The algorithm is an ensemble tree method, which yields stable and accurate predictions by averaging the predictions of many unstable, random decision trees.

A decision tree (DT) algorithm sequentially applies conditional rules to distribute the data into internal nodes and leaf nodes. During the growing phase, the decision tree learns the conditions that, if applied to the attributes of the training data, minimise the mean squared error of the tree. At each partition, the data are split into internal nodes or leaf nodes. Internal nodes consist of data with large variation in the target response, which need to be considered for further splitting by applying additional conditional rules. The growing phase ends when all data have been distributed into leaf nodes with low variation in the target response. The process is represented in the form of a directed graph in which the root node at the top contains the full dataset and branches out to layers of nodes that consist of the partitioned data. Each branch represents a conditional rule that is applied to the distribution of a single descriptor. In prediction, the conditional rules that have been learned during the training of the tree are applied on test data and the prediction for a given instance of the test data may be obtained as the average response values of the training data in the leaf node that it is placed in.

Overfitting of the RF to the data is avoided by applying different stopping rules, i.e., on the response variation in the nodes, or by a process called pruning. Stopping rules are applied on each individual node during the DT growing process using the information impurity minimisation criterion, i.e., mean square error, to stop the splitting when the target response variation threshold is reached. Pruning is applied retrospectively to all trees in the forest. First, the DT is fully grown by recursively partitioning the training data and, then, the nodes where the response variation exceeds the target variation are pruned.

In RF, each tree is grown on a bootstrap sample of the training data and using a random subset of descriptors. A bootstrap sample is a subset of the training data obtained by random sampling with replacement. Repeatedly drawn bootstrap samples result in a fraction of the training set not being sampled, which is referred to as the out-of-bag sample. The out-of-bag sample may be used to assess the predictive performance of the RF in parallel with the training process, thus, reducing the requirement of a separate validation set. Predictions for unseen data are made by averaging the individual tree predictions.

3.5 Model validation

Following model development, evidence of the model's ability to generalise on unseen data is required to confirm that it is useful. The most rigorous form of validation is done using external data but considerations such as the model's applicability domain and the underlying experimental assumptions of the training data need to be made. That is to say, the descriptor space of the external data should be representative of the descriptor space of the training set and that their experimental protocols and environmental constraints should match.

In the absence of an adequate external test set, a random sample that is representative of the training set may be held out for model validation, typically, 10-20% of the training set size. As performance estimates will vary for different random samples, repeated holdout samples may be obtained to calculate a robust estimate of the average performance of a model (Consonni, Ballabio, & Todeschini, 2010; Raschka, 2018). However, the performance estimates obtained may not reflect the model's performance on unknown data. Other rational sampling methods, such as diversity sampling and time-based data splitting may yield more realistic estimates of the models' prospective performance (Golbraikh & Tropsha, 2002b; Martin et al., 2012; Sheridan, 2013a).

Cross-validation is particularly useful for small datasets where holding out a set of compounds from the training set significantly reduces a model's performance. N-fold cross-validation involves partitioning the training data into N folds and holding out one fold in each iteration, until all folds have been excluded. The model is trained at each iteration on N-1 folds, while the remaining fold is used for evaluation. The model's performance is then reported as the average performance across the N folds. Increasing the number of folds, N, yields more accurate error estimates for the individual folds but also increases the variance of the average model estimate. As a result, large datasets where each fold may contain a small, insignificant number of compounds, e.g., less than five percent, will produce unreliable estimates and, therefore, it is suggested that leave-one-out cross-validation methods are avoided (Golbraikh & Tropsha, 2002a). Cross-validation is the gold standard for both the optimisation and the validation of QSAR models (Tetko et al., 2008). During model optimisation, the best parameters are identified by minimising the average model error across all folds. However, this estimate is biased to the data and overoptimistic of the model's prospective performance (Chirico & Gramatica, 2011; Krstajic, Buturovic, Leahy, & Thomas, 2014). An unbiased estimate of model performance may be obtained by applying nested cross-validation methods, which permit the estimation of the average model error that accounts for the optimal model parameters for alternative data splits (Filzmoser, Liebmann, & Varmuza, 2009; Krstajic et al., 2014). Nested cross-validation consists of two loops: the inner loop, which is used to fit and optimise the model at each iteration of the outer loop, which is used to estimate the average model error over the different data splits represented by each fold. Repeated nested cross-validation methods, which involve adding an external loop whereby the data are randomised at each iteration of nested cross-validation, have also been used to produce an interval estimate of the model's error that may be used to assess whether additional data need to be collected or additional descriptors need to be investigated (Krstajic et al., 2014).

The coefficient of determination (R^2), the root-mean-squared error (RMSE) and the median absolute error (MAE) metrics are calculated on training data to assess the regression models' fit. Respective measures may be calculated to assess the predictive performance of QSAR models on cross-validation or external test data, but in this case they are referred to as the predictive squared correlation coefficient Q^2 and the root-mean-squared error in prediction (RMSEP). Corrections in the calculation of Q^2 have been proposed by several groups resulting to a total of five Q^2 variants (Chirico & Gramatica, 2011; Roy et al., 2012; Schüürmann, Ebert, Chen, Wang, & Kühne, 2008; Shi et al., 2001). Detailed evaluation of the available Q^2 metrics by Todeschini et al. revealed that only one introduced by Consonni, Ballabio, & Todeschini (2009) was suitable for the reliable evaluation of QSAR models (Table 3-1). Yet, it requires that the test data is within the model's applicability domain and, thus, their values are within the representative value range of the training set. The other variants of Q^2 were shown to be either sensitive to transformations applied to the data, to overestimate the predictive ability or were not well correlated with the RMSEP estimate of external or holdout data (Todeschini, Ballabio, & Grisoni, 2016). The definitions of three main performance measures for the evaluation of model fit and the external predictive ability of QSAR regression models on a validation set of size N are provided in Table 3-1, where y_i, $\hat{y}_i$ are the observation and the prediction, respectively, of each validation compound and $\bar{y}_{tr}$ is the mean observed response of the training set. The RMSEP and Q^2 are calculated for external or holdout data.

Table 3-1. Performance measures for the evaluation of regression models for prediction

Metric	Reference
$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_{tr})^2}$	(Alexander, Tropsha, & Winkler, 2015)
$RMSEP = \sqrt{\frac{\sum_{i=1}^{N_{ext}}(y_i - \hat{y}_i)^2}{N_{ext}}}$	(Consonni et al., 2010)
$Q^2 = 1 - \frac{\frac{(\sum_{i=1}^{N_{ext}}(y_i - \hat{y}_i)^2)}{N_{ext}}}{\frac{(\sum_{i=1}^{N_{tr}}(y_i - \bar{y}_{tr})^2)}{N_{tr}}}$	(Consonni et al., 2010)

High confidence may be placed in a model's predictions if these are obtained in the absence of trends in the model's error and predictions with large residual errors. Particularly in the case of local models, the predictions with residual errors that exceed a specified threshold may indicate the presence of compounds that are dissimilar to the training compounds, i.e., novel compounds, with distinct structural characteristics or modes of action (Dearden, Cronin, & Kaiser, 2009). To ensure high model accuracy, outliers that exceed 2-fold of the experimental error value (Keefer, Kauffman, & Gupta, 2013) or 3-fold of the standard deviations from the residual error mean (Dearden et al., 2009), during internal validation, are usually excluded prior to rebuilding the model. In the case of global models that consist of many diverse structures, however, there is interest in detecting novel structures and incorporating them in the model to extend the model's applicability domain and the model's ability to predict new compounds accurately.

The model's errors should also be of similar size to the experimental assay error (Eriksson et al., 2003). Regression based on the minimisation of least square errors relies on the assumption that the model's errors are random and normally distributed with a mean close to zero and constant variance, i.e., homoscedastic. The presence of trends in the models' residual errors or significant departure of the residual error distribution from normality indicates the presence of systematic errors (Cortes-Ciriano, 2016; Roy, Ambure, & Aher, 2017), which may be attributed either to the presence of measurement bias in the data or the lack of important variables in the model. Systematic errors may be detected by analysis of the residual errors with the use of residual error plots (Dearden et al., 2009).

3.6 Domain of applicability

In practice, the accuracy of a QSAR prediction for an untested compound can only be determined retrospectively and following experimental measurement. Therefore, even if a model has been validated, it may still produce inaccurate predictions, particularly, when the test data are dissimilar to the training data and are located in regions of the chemical, descriptor or response space that are not well represented in the model, such as activity cliffs (Keefer et al., 2013; Maggiora, 2006). This is partly attributed to the bias of medicinal chemistry datasets in certain regions of the chemical space. In fact, in-house datasets are integrated into public datasets to bias the applicability domain (AD) of the latter and improve the accuracy of QSAR models (Tetko, Bruneau, Mewes, Rohrer, & Poda, 2006).

To be useful, QSAR predictions need to be accompanied by a statement regarding their confidence so that users of the model may be warned if a prediction is not reliable (Netzeva et al., 2005). In QSAR, the confidence in a model's predictions is assessed by taking into account the relevance of the query compound to the chemical space of the training set (Sheridan, Feuston, Maiorov, & Kearsley, 2004), which is also known as the model's AD. Compounds with high structural similarity to the training data are associated with reliable predictions made by interpolation inside the model's AD, where the QSAR is valid; while predictions made by extrapolation are expected to be less reliable (Sahigara et al., 2012).

Many methods for defining the AD have been described in the literature but there has been little agreement as to how a model's AD is optimally defined. From a higher perspective, AD methods can be divided into novelty detection methods (Mathea, Klingspohn, & Baumann, 2016) and error estimation methods (Toplak et al., 2014).

3.6.1 Novelty detection methods

Novelty detection methods rely entirely on the input variables of the training and test data and may be applied prospectively to the model's use to obtain qualitative estimates of the predictions' reliability. The objective of these methods is to classify compounds as inside or outside the model's AD by applying empirical thresholds on the basis of their structural or molecular similarity to the training data. The boundaries are then used to distinguish between high confidence and low confidence predictions. For example, range based methods apply thresholds to each individual descriptor to form an N-dimensional bounding box or define the smallest possible space by applying geometrical boundaries to the descriptor value ranges of the training set (Sahigara et al., 2012).

Continuous reliability estimates, which facilitate the application of more flexible, user-defined thresholds, are obtained from distance- and density-based methods. Distance-based methods calculate the distance between the test compound and a reference point in the training data; and evaluate the confidence in a prediction based on a defined statistical threshold. The reference point may be defined as a) the mean of a single or K nearest neighbours in the training set, b) the mean of all training set compounds or c) a cluster of compounds (Stanforth, Kolossov, & Mirkin, 2007). Equivalent methods based on similarity measures are defined when the structural representation of molecules is based on binary fingerprints rather than numerical descriptors.

Distance-based methods

Distance is typically calculated using the Euclidean, Mahalanobis or Manhattan distance metrics in multidimensional descriptor space (Jaworska, Nikolova-Jeliazkova, & Aldenberg, 2005; Sahigara et al., 2012). The Euclidean distance is the most frequently used metric and it is applied on previously standardised data. The Mahalanobis distance accounts for correlated descriptors and its calculation is similar to the Euclidean distance but also requires calculating the covariance matrix. The Manhattan distance is more suitable for non-continuous numerical descriptors (Jaworska et al., 2005). The definition of the distance metrics for two compounds x and y in N-dimensional descriptor space is provided in Table 3-2.

Table 3-2. List of measures used to calculate the distance between compound x and y

Euclidean	$D_E = \sqrt{\sum_{i=1}^{N}(x_i - y_i)^2}$
Manhattan	$D_{MN} = \sum_{i=1}^{N}\lvert x_i - y_i\rvert$
Mahalanobis	$D_{MN} = \sqrt{(X-Y)^T S^{-1}(X-Y)}$

S: covariance matrix X, Y: descriptor vectors of x and y
T: transpose of matrix

The leverage is another related distance metric, which is used to assess the influence of training data points on the model's fit, if excluded. This metric is proportional to the Mahalanobis distance when the reference point is the centroid of the training set, and it assumes that the multivariate descriptor distribution is normal.

Training set compounds with a high leverage are more influential on the model's performance; they stabilise the model (Jaworska et al., 2005) and are considered to extend the coverage of chemical space (Gadaleta, Mangiatordi, Catto, Carotti, & Nicolotti, 2016). The leverage of test data is useful for identifying compounds that are extrapolated by the model (Netzeva et al., 2005; Tropsha, Gramatica, & Gombar, 2003; Worth et al., 2005). The leverages of the data are derived from the calculation of the hat matrix, H, and model extrapolations are identified by comparing the leverage values to a warning threshold. The warning leverage is associated with the amount of noise in the prediction and is set to 3(p+1)/n, where p is the number of descriptors and n is the size of the training set. The parameters required for the calculation of the leverage of a compound i with a descriptor vector x to the training set with descriptor matrix X are provided in Table 3-3.

Table 3-3. Parameters for the calculation of leverage for a compound i

Hat matrix	$H = X(X^T X)^{-1}X^T$
Leverage for i	$h_{ii} = x_i^T (X^T X)^{-1} x_i$
Warning leverage	$h^* = 3(p+1)/n$

A user-defined threshold is applied to assess whether a test compound lies inside or outside the boundaries of the AD. Five strategies for defining distance-based thresholds on the training data were investigated by Sahigara et al. (2012). The thresholds suggested by the authors are the following: 1) the maximum distance to the centroid of the training set; 2) double; or 3) triple the average distance to the centroid; 4) the 95th percentile of the training set distances; and 5) the sum of the average distance and the standard deviation of the distances multiplied by an arbitrary factor z. The fifth strategy was found to be stricter than the other four as it integrates information about the density of the local neighbourhoods of the query compound and, thus, excludes more compounds from the

AD compared to the other methods. It was concluded that, generally, the results obtained by applying the same threshold on different metrics may differ and that the choice of threshold needs to account for a trade-off between the number of training set compounds to be excluded and the improvement in model performance.

Density-based methods

Density methods evaluate the AD as the average of a Gaussian distribution in the multivariate descriptor space using kernel density estimation (KDE). KDE estimates the probability density of higher dimensional data as a function of known parametrical distributions (Netzeva et al., 2005). The representation of the AD using the probability density distribution of the data makes it possible to identify the highest density region occupied by a (user-defined) fraction of data in descriptor space. The potential of each compound is calculated using a known kernel function, for example, Gaussian, and statistical cut-off values may then be applied to define the AD threshold. Compounds with a smaller potential than the AD threshold are considered to be outside the AD (Sahigara et al., 2012). These compounds are easily identified visually in low confidence regions of the density distribution and correspond to predictions that are extrapolated by the model with high uncertainty.

Despite their simplicity in defining the AD of a model, novelty detection methods are inefficient in high-dimensional spaces (Mathea et al., 2016; Netzeva et al., 2005) and are unable to explain the poor accuracy of a model's predictions inside the AD. Reliability estimates with higher correlation to prediction accuracy may be obtained by applying consensus approaches, which combine various AD metrics, including algorithm-specific reliability estimates. Examples of these approaches are implemented in the works of Dragos, Gilles, & Alexandre (2009), Sheridan (2012, 2013b, 2015) and Yun et al. (2017) and are discussed in the following section. A consensus approach that involves the integration of multiple AD definitions during model development has also been suggested by (Hanser, Barber, Marchaland, & Werner, 2016). The authors propose that the various AD methods contribute different types of information to the modelling process, all of which need to be taken into account to evaluate a new prediction. They categorise the methods based on the following three elements of information that a fully described AD should have: the relevance of the test data to the training data; whether the amount of training data is enough to yield an accurate model; and whether the model may yield confident predictions (Table 3-4).

Table 3-4. Elements addressed in the definition of a model's AD according to Hanser et al. (2016)

Layer	Element	Method
1	Relevance to training set	Range, Similarity/Distance
2	Sufficiency of data	Density (D2NN)
3	Confidence	Density

3.6.2 Confidence estimation based on model error

Confidence estimation methods are applied retrospectively to QSAR modelling as they rely on the model's output and algorithm-specific reliability estimates (Mathea et al., 2016; Sahigara et al., 2012; Sushko et al., 2010). In machine learning, the confidence in a prediction is reported in the units of the endpoint, in the form of a prediction error estimate (Toplak et al., 2014) or an interval estimate.

As mentioned before, the uncertainty estimates of the model's individual predictions are not always directly available when using machine learning algorithms, however, these may be obtained by introducing additional techniques into the modelling process. For example, resampling methods may be applied to build an ensemble of models (Sahlin, Jeliazkova, & Oberg, 2014), whereby uncertainty is estimated as the variation of a prediction across the ensemble. Another technique uses the model's residual errors to build a model that may be used to estimate the errors of future predictions, i.e., an error model (Sheridan, 2013b). These may then be used to calculate confidence intervals for the model's future predictions, i.e., prediction intervals. Other techniques may be used to estimate the prediction intervals, such as conformal prediction (Eklund, Norinder, Boyer, & Carlsson, 2012) or modified ensemble tree algorithms (Feng, Svetnik, Liaw, Pratola, & Sheridan, 2019; Meinshausen, 2006; Zhang, Zimmerman, Nettleton, & Nordman, 2019).

Prediction intervals are associated with a probability that the future measurement will be included within the defined value range (Willink, 2012). In statistical inference, a prediction interval (PI) is a range of values that contains the future observation, y_{obs}, with a certain degree of confidence. A PI should not be confused with a confidence interval (CI) for the prediction as a CI only accounts for the uncertainty of the model's estimates, $u(y_{pred}\,)$; while a PI also accounts for the uncertainty of the future observation, $u(y_{obs}\,)$. As a result, the PI for a future observation will always be wider than a CI of a prediction. For a confidence level of 95%, a PI is interpreted as follows: "It is estimated that at least 95% of the calculated PIs are correct, i.e., they contain the future measurements". The PIs that represent the uncertainty of predictions from linear regression are calculated parametrically (Table 3-5). The assumption is made that the data are independent and identically distributed (IID) and the model's errors are approximately normal or follow a t-distribution. Both types of intervals are calculated using Equation 3.3, but, as seen in Table 3-5 the error margin (EM) of the PI for a new prediction,$\hat{y}_i$, contains an additional parameter.

$$\mathrm{CI} = \hat{y}_i \pm t_{\left(\frac{\alpha}{2},n-2\right)}\,\mathrm{EM} \qquad 3.3$$

Where t: critical value at (1-α) % confidence

Table 3-5. Error margins used to calculate intervals in linear regression and sampling.

Interval type	Error Margin (EM)	Application
CI of the observation mean	$s_e\sqrt{(\frac{1}{n}+\frac{(x_i-\underline{x})^2}{\sum_i^N(x_i-\underline{x})^2})}$	Linear regression
	$s\sqrt{\frac{1}{n}}$	Sampling
PI of the individual prediction	$s_e\sqrt{(1+\frac{1}{n}+\frac{(x_i-\underline{x})^2}{\sum_i^N(x_i-\underline{x})^2})}$	Linear regression
	$s\sqrt{(1+\frac{1}{n})}$	Sampling

Other known nonparametric methods for prediction interval estimation include bootstrap resampling, mean-variance estimation, the delta approach and the Bayesian approach (Kümmel, Bonate, Dingemanse, & Krause, 2018). Methods for PI estimation have also been developed for neural networks, which may also be applicable for nonlinear machine learning models (Rasmussen & Hines, 2003).

3.6.2.1 Resampling

The standard deviation from an ensemble of models built with resampling is reportedly the best estimator of accuracy of a model's prediction (Kaneko & Funatsu, 2014; Tetko et al., 2008). It is considered the gold standard for the estimation of prediction errors as it is able to distinguish well between small and large errors (Tetko et al., 2008).

Ensembles may consist of hundreds of models trained using the same algorithm but different subsets of data, variables or model parameters. These may be built through the implementation of bootstrap sampling (Kaneko & Funatsu, 2014) or cross-validation methods (Baumann & Baumann, 2014; Tetko et al., 2008). Ensembles formed as a consensus of separately trained models using different algorithms have also been reported by Tetko et al.(2008).

An ensemble prediction is calculated by averaging the predictions, y_i, of the individual models, k, and its confidence is estimated by the standard deviation (STD) of the prediction mean, $\bar{y}$ (Equation 3.4).

$$STD = \sqrt{\frac{\sum_{i=1}^{k}(y_i-\bar{y})^2}{k-1}} \qquad 3.4$$

The standard deviation informs about the degree of discord in the ensemble. A large STD for a prediction, y, implies that the new data is very different to the training set and, thus, that the prediction is less reliable and that the prediction is likely to have a large prediction error. However, predictions with a large STD may still have small prediction errors. In the study of Kaneko & Funatsu (2014) it was found that ensembles based on variable sampling are more efficient in identifying large prediction errors in diverse datasets, while ensembles based on data sampling are suitable for less diverse data. However, neither of the sampling methods is able to capture the bias in the predicted value (Kaneko & Funatsu, 2014), which is introduced in the model by the data distribution. While this bias may be accounted for in AD-based metrics, it has been difficult to integrate in ensembles (Kaneko, 2018; Kaneko & Funatsu, 2014). This is mainly due to the complexity of joining the ADs of the individual k models, which are based on different variables (Kaneko, 2018).

Another ensemble method for obtaining prediction error estimates was suggested by Tetko et al. (2008)and Sushko et al. (2010) and is based on the correlation between the distribution of a test compound's predictions and the distribution of training set predictions in the ensemble. A correlation measure is defined as the maximum correlation coefficient of the ensemble predictions of the test compound with the predictions of the training set compounds. However, it is outperformed by the STD of prediction in the estimation of prediction errors (Tetko et al., 2008)

3.6.2.2 Error models

The use of error models to study the relationship of more than one AD metric and QSAR prediction errors has been investigated in the work of Sheridan (2012). Regression algorithms, such as RF (2013a) and SVM (Lapins et al., 2018) have been applied as a data-driven approach to investigate the influence of different types of variables on the model's individual prediction errors (Mathea et al., 2016; Sheridan, 2012). Classification error models have also been reported in the literature (Carrió, Pinto, Ecker, Sanz, & Pastor, 2014; Dragos et al., 2009; Klingspohn et al., 2017), though, error estimates typically involve averaging of errors within binary or multi-category classes (Carrió et al., 2014).

In the works of Dragos et al. (2009) and Sheridan (2012, 2013a) the consensus of AD metrics was shown to synergistically improve the estimation of errors in individual QSAR predictions. Dragos et al. (2009) built classification error models based on AD metrics and statistical-based metrics to distinguish between trustworthy and untrustworthy predictions of QSAR regression models. Although their approach does not produce compound-specific reliability estimates, their results suggest that the robustness of reliability estimates may be improved by using a consensus of AD metrics. Their proposed framework minimises the occurrence of inaccurate reliability estimates in order to objectively apply an optimal AD threshold. Furthermore, they show that robust reliability estimates may be obtained by resampling QSAR descriptor subsets without prior treatment of descriptor correlations, and that applying an error-based threshold on the training data improves the performance of distance-to-model approaches.

In his work, Sheridan (2012) provides evidence that the relative importance of algorithm-based reliability metrics and the similarity between the test and training set data varies across diverse datasets. A consensus AD approach was also utilised in Carrió et al. (2014). The Applicability Domain ANalysis (ADAN) approach uses a combination of six AD metrics to classify PLS and RF predictions into seven reliability categories (Carrió et al., 2014). Each category represents the number of AD rules, i.e., thresholds, satisfied for each compound and is associated with a range of prediction error estimates from validation, which are used to compute approximate confidence intervals for each category. However, linear correlation between categories is apparent only on well behaved data distributions and models with good predictive performance.

Similar results were obtained in (Sheridan, 2013a), after training a RF regression error model on the cross-validation residuals of a RF QSAR model. Later work (Sheridan, 2015), revealed that similarity was a more important variable for the estimation of prediction errors only in the case of local, less diverse datasets. Considering the low performance of error models on cross-validation data, the value of the error models when applied to holdout or external data is not guaranteed. This is further supported by validation data supplied by the author on a different study (Sheridan, 2013a), which indicates that cross-validation yields too optimistic estimates compared to other validation methods, such as the use of time-split or neighbour-split test sets.

Error models based on PLS (Wood, Carlsson, Eklund, Norinder, & Stålring, 2013), KNN and SVM (Lapins et al., 2018) algorithms have also been reported in the literature but may require optimisation, which is not required in the case of RF. Another benefit of RFs is that resampling and feature selection are embedded in the algorithm, which yield robust prediction error estimates and facilitate the identification of important AD variables. In contrast to the training of a QSAR model it is not clear as to whether error models should be optimised or not: as optimisation of the error models introduces bias to the error model it is likely to limit its predictive performance on new test data (Lapins et al., 2018).

A method for assessing uncertainty estimation techniques that are based on resampling and error modelling involves treating observations and predictions as distributions. The uncertainty estimates generated by each method are applied to convert predictions to Gaussian distributions and a likelihood-based measure is used to assess the performance of the alternative techniques (Sahlin et al., 2014; Tetko et al., 2008; Wood et al., 2013). This requires that the probability distributions of the observations are known, in other words, that information about the experimental uncertainty of the data is available. The technique that produces the highest likelihood score for test data is the one that yields the optimal prediction distributions, which are those with the smallest uncertainty estimates for the maximum amount of overlap with the observations (Sahlin et al., 2014; Wood et al., 2013).

3.6.2.3 Conformal prediction

Conformal Prediction (CP) is a confidence estimation framework with successful applications in a range of classification and regression tasks solved by machine learning. Further to its use in chemoinformatics (Ahmed et al., 2018; Svensson, Norinder, & Bender, 2017) and QSAR (Cortés-Ciriano, Bender, & Malliavin, 2015; Eklund et al., 2012; Norinder, Rybacka, & Andersson, 2016; Sun et al., 2017) , the method has been used in applications such as biomedical diagnosis (Papadopoulos, Gammerman, & Vovk, 2009), bioinformatics (Nouretdinov, Gammerman, Qi, & Klein-Seetharaman, 2012), network traffic prediction (Dashevskiy & Luo, 2008), image analysis (Lambrou et al., 2010) and facial recognition (Eliades & Papadopoulos, 2017), cyber security (Wechsler, 2015) and stock price prediction.

Definitions

CP is a nonparametric method and non-specific to the modelling algorithm, thus, it may be applied on top of any machine learning algorithm to obtain empirical uncertainty estimates for its predictions (Papadopoulos, Vovk, & Gammerman, 2011). A conformal predictor yields a prediction interval (PI) as output, which corresponds to a range of values that is expected to contain the future observation with confidence. The confidence threshold, which is set by the user, is applied on the model's error distribution from existing data rather than parametric distributions, i.e., t-distribution or z-distribution, for the calculation of prediction intervals.

Several implementations of CP that differ with respect to the model's training schedule are available: transductive (TCP), inductive (ICP) and aggregate (ACP). The original implementation is based on a TCP training schedule whereby the model is retrained following the prediction of every additional test prediction made. An ICP training schedule involves training the model only once; following partitioning of the original training set of size l into the proper training set and the calibration set of size m and q, respectively, where $q < \mathrm{m}$ and $l = \mathrm{q} + \mathrm{m}$. The ACP training schedule, involves repeatedly sampling the calibration set from the original training set to produce an average error distribution that improves the robustness of the model's error estimates. Note that the calibration data are excluded from training of the underlying algorithm and only used to infer the model's empirical error distribution.

The concepts of CP for regression tasks have been introduced by (Papadopoulos et al., 2011). Given the vector of a training sample k with $\{z_1, z_2 \ldots, z_l\} \in Z^k$, where z_i represents a compound with a descriptor vector of x_i and an observation of y_i, a conformal predictor estimates the confidence in all of the model's predictions $\tilde{y}$ for a new, test compound with a descriptor vector of x_{l+1}. The only assumption made is that all (x_i, y_i) pairs are independently and identically distributed (IID) or, at least, exchangeable.

The nonconformity, a_i, i.e., the dissimilarity of z_i to the compounds of the training sample k is given by a function of the model's error, i.e., the signed residual error or the absolute residual error, which is referred to as a nonconformity measure $A = f(z)$. The nonconformity scores of calibration data may then be ordered to produce a reference ranked list, $\{a_{1,}a_{2,} \dots a_i\}_k$,which, in essence, represents the model's empirical error distribution. The nonconformity score of each reference compound is associated with a p-value (Equation 3.5), which is interpreted as the fraction of compounds in the list that are at least as "nonconforming" as the reference (Linusson, 2017).

$$p(\tilde{y}) = \frac{\#\{i = 1, \dots, l+1 : a_i \geq a_{l+1}\}}{l+1} \qquad 3.5$$

The error of new predictions is then estimated by applying a significance level threshold $\varepsilon \in [0,1]$ to the list. The error estimate is equal to the value of a_ε that corresponds to the p-value that satisfies the condition given in Equation 3.6:

$$p(\tilde{y}) > \varepsilon \qquad 3.6$$

Thus, the calculation of the PI for all possible predictions of the model, $\tilde{y}$, at a confidence level of (1- ε) % is given in Equation 3.7.

$$PI_{\tilde{y}} = \hat{y}_i \pm a_\varepsilon \qquad 3.7$$

However, Equation 3.7 yields uniform PIs for all compounds, which assume that the model's uncertainty for every new prediction is equal. Compound-specific prediction intervals, where PIs are scaled to the uncertainty associated with the individual prediction, are obtained with the use of normalised nonconformity measures. These are defined as $A_n = \frac{f(z)}{\sigma}$, where σ is an uncertainty estimate derived from a mathematical function, error model or a reliability score obtained from the evaluation of the AD (Eklund et al., 2012; Norinder, Carlsson, Boyer, & Eklund, 2015; Norinder et al., 2016; Svensson et al., 2018). Thus, the normalised PIs for predictions i at a confidence level of (1- ε) % are calculated as in Equation 3.8.

$$PI_{\varepsilon,i} = \hat{y}_i \pm a_\varepsilon \sigma_i \qquad 3.8$$

The requirement for higher confidence yields larger intervals and, naturally, PIs are more likely to include the future observation. At a fixed confidence level, e.g., 95%, normalised PIs obtained from different normalisation functions may vary. The agreement of the different normalised nonconformity measures is assessed for a dataset by comparing the sizes of the average PIs (Bland & Altman, 2003; Johansson, Boström, Löfström, & Linusson, 2014; Rasmussen & Hines, 2003).

The validity of the PIs is guaranteed by the assumption that the data are IID, or at least exchangeable, which is a common requirement in statistical inference and machine learning methods. It is satisfied by designing the calibration and test data using random sampling methods, as they maintain the original probability distribution of the data. This assumption guarantees that future observations of the model will in fact be present in the specified

PIs at the stated probability. However, the assumption cannot be verified for new data until after the experimental measurements are made; nor may it be, strictly, satisfied in diverse, pharmaceutical datasets (Eklund, Norinder, Boyer, & Carlsson, 2015). Global QSAR models are periodically updated with predictions to account for the newly assayed compounds, although predictions outside the desirable range are commonly excluded. The diversity of compound structures is also greater in global datasets as a pharmaceutical company may work on multiple projects that focus on different molecular scaffolds represented by different probability distributions. Furthermore, temporal information may be associated with the data, since historical data is used to guide the design of new compounds. As a result, the validity of the CP models may be compromised and the uncertainty underestimated. Nevertheless, CP may still be useful even if validity is only approximate. The training schedule applied for global QSAR models is compatible with a TCP setting, where the CP model is updated with every new data prediction that is tested. However, as discussed in (Eklund et al., 2015), this setting is impractical for their development. Instead, the ICP setting, in which the model is trained once, is usually implemented due to its computational efficiency.

Evaluation of Conformal Predictors

The performance of conformal predictors is evaluated by their validity and efficiency. Validity holds if the CP confidence estimate is confirmed on test data. In other words, a CP is valid, when the percentage of PIs that do not include the future measurement, i.e., the error rate of the CP, is less than or equal to the significance level. The PIs estimated from a CP that is not valid are, thus, not useful. A conservative CP, that yields a much smaller error rate than the significance level results in large PIs. The L2-norm, which is defined in Equation 3.9 for an N-dimensional space, may be used to measure the difference of an ACP's expected error rate to the obtained error rate for the full confidence distribution or part of it (Svensson et al., 2017).

$$l^2 = \sqrt{x_1^2 + \cdots + x_N^2} \tag{3.9}$$

The efficiency of CPs is evaluated by the average PI size for a dataset. As previously mentioned, efficiency is greater at lower confidence levels where PIs are narrower than in higher confidence levels. Narrow PIs are more informative as they can be used to make decisions that require a greater precision. However, narrow enough PIs may not be easy to obtain at very high levels of confidence, e.g. 95% and, typically, setting the confidence level threshold at 80% yields a suitable trade-off between confidence and efficiency.

In the work of Eklund et al. (2012), normalised PIs were constructed by applying error models as normalisation functions. The error models, similar to AD models, are considered as an alternative method for the estimation of prediction uncertainty based on the variability of the data in feature space. However, the combination of uncertainty from different sources has not been addressed.

3.7 Conclusions

This chapter has introduced the concepts of QSAR modelling and the main steps required to build a QSAR model. These include the selection of descriptors, the optimisation of the modelling algorithm, validation of the model's performance on known data and the definition of the model's applicability domain. The main focus has been the development of regression QSAR models using supervised learning methods and the approaches for assigning confidence estimates to their individual predictions. The literature suggests that defining the model's AD is a nontrivial task; and that combining information obtained from different AD methods and error-based reliability estimation methods may be required to obtain confidence estimates. A promising method is that of error modelling using regression algorithms, which may be used to explore the relationship between reliability metrics and QSAR prediction errors. Assuming that error models are predictive, then they may be useful for the estimation of confidence in individual QSAR predictions. Conformal prediction provides a robust mathematical framework for this purpose; as it utilises calibration data to estimate the confidence in the model's predictions. Estimates are obtained in the form of PIs, whereby reliability estimates derived from the definition of ADs or error models may be used to generate compound-specific PIs. The predictive performance of error models and their utility for the purpose of confidence estimation for ADME models is investigated and discussed in greater detail in the following chapters of this thesis.

Chapter 4 Description of Datasets

4.1 Introduction

This chapter describes the datasets that are used throughout the thesis. The sections that follow provide an overview of the dataset characteristics and provide additional details regarding the pre-processing and data curation steps applied to the data. Eleven datasets were studied in total and consist of one physicochemical property dataset, namely LogD, and ten datasets that represent ADME endpoints. Each of the datasets contains experimental measurements from a single assay.

4.2 LogD dataset

The distribution coefficient, LogD, is the concentration ratio of a compound between two immiscible solvents, such as n-octanol and water. It is closely related to the partition coefficient, LogP, which quantifies the lipophilicity of a molecule in neutral form; however, the LogD yields a more realistic estimate of lipophilicity in physiological conditions as it quantifies all forms of the molecule, i.e., the ionised and neutral states (Wang et al., 2015). In ADME prediction, LogD may be used as an estimate of membrane permeability. The wide availability of reliable LogP measurements in public datasets has prompted the development of many methods for the estimation of LogP. However, these are lacking for LogD for which large datasets of experimental data are not available and the estimation is more complex (Tetko et al., 2006; van de Waterbeemd & Gifford, 2003).

The LogD dataset (CHEMBL3301363) used here was retrieved from CHEMBL (v.21) and consists of 4200 data points. Each data point corresponds to a single compound represented by its SMILES string and a single measurement of its coefficient of distribution (LogD) in a buffer solution of n-octanol and water at pH = 7.4. This dataset is a small subset (7%) of the LogD AstraZeneca dataset that has been described in several publications and is associated with an experimental assay error estimate of 0.1 LogD units (Wenlock & Carlsson, 2015; Wood et al., 2011).

Prior to the calculation of descriptors, compounds with missing measurements and molecular structures representing mixtures of compounds, inorganic molecules or salts were removed using the RDKit salt stripper node in KNIME. The remaining structures were standardised by applying the RDKit structure Normalizer node which removes salts, neutralises charged structures and resolves the structures in which the stereochemistry is not accurately represented. The canonical SMILES were then generated using the RDKit Canonical SMILES node.

Three compounds with missing measurements were identified and subsequently excluded from model training and validation. Following the data curation process and the removal of 27 structures that could not be resolved; a total of 4170 structures were used to generate canonical SMILES strings for the calculation of molecular descriptors the details of which are provided in the following chapter. The holdout data was sampled randomly for the purpose of model validation with 85% of the dataset assigned to the training set and the remaining 15% to the test set. Table 4-1 below shows the descriptive statistics of the training set and holdout test set measurement distributions.

Table 4-1. Descriptive statistics of the LogD training set and holdout test set

Data Partition	**Size**	**Range**	**Median**	**Mean**	**Standard Deviation**
Train	3574	[-1.50, 4.50]	2.37	2.19	1.19
Test	596	[-1.48, 4.50]	2.33	2.16	1.24

4.3 ADME datasets

Ten datasets representing ADME endpoints were supplied by Eli Lilly. Due to their confidential nature, the training set and test set of each dataset were provided as pre-calculated matrices of descriptors, which were calculated following the same data curation protocol described for the LogD dataset in KNIME. Details on the calculated descriptors are provided in the following chapter. All figures reported below refer to the curated data.

An external test set was provided by Lilly, which was derived from temporal ordering, i.e., time-split, and represented future data measurements. However, separate holdout test sets were generated by randomly sampling 20% of the training data. The holdout test sets were used to validate the underlying models built in Chapter 5 and the error models in Chapter 6. Both the randomly selected holdout test sets and the temporal test sets were used to validate the conformal prediction results in Chapter 7.

The provided measurements had been previously normalised at Eli Lilly to fall within the range [0, 1] by applying a log transformation, unless these were reported as a percentage or a fraction. Measurements that were reported as censored data, e.g. >1 or <0, were excluded. Repeated measurements in the data were averaged and replaced by their mean value for modelling. Table 4-2 below lists the endpoints and the descriptive statistics of the training sets and the external test sets of all ADME datasets.

Table 4-2. ADME endpoints and descriptive statistics of the training sets and external test set measurements

Dataset	Endpoint	Partition	Range	Mean	Standard Deviation	Median	Size
1	Brain-to-plasma concentration ratio	Train	[0.00, 1.00]	0.39	0.21	0.35	866
		Test	[0.04, 1.00]	0.40	0.22	0.38	230
2	Total concentration, plasma	Train	[0.00, 0.96]	0.34	0.11	0.34	928
		Test	[0.03, 0.81]	0.34	0.09	0.35	248
3	Fraction unbound protein – mouse, brain	Train	[0.00, 1.00]	0.49	0.23	0.48	1429
		Test	[0.06, 1.00]	0.51	0.23	0.50	375
4	Passive permeability	Train	[0.00, 1.00]	0.18	0.19	0.10	2089
		Test	[0.02, 0.80]	0.19	0.19	0.11	548
5	Fraction unbound protein-human, microsomal	Train	[0.0, 1.00]	0.18	0.14	0.13	2401
		Test	[0.03, 0.85]	0.22	0.16	0.15	621
6	Fraction unbound protein – mouse, plasma	Train	[0.00, 1.00]	0.38	0.20	0.37	3133
		Test	[0.00, 0.85]	0.32	0.18	0.32	824
7	Metabolic stability human	Train	[0.00, 1.00]	0.28	0.25	0.19	2959
		Test	[0.00, 1.00]	0.33	0.30	0.23	804
8	Metabolic stability-dog	Train	[0.00, 1.00]	0.33	0.28	0.25	2962
		Test	[0.00, 1.00]	0.34	0.27	0.28	803
9	High throughput solubility assay	Train	[0.02, 1.00]	0.47	0.27	0.47	12022
		Test	[0.04, 1.00]	0.47	0.27	0.46	3116
10	Metabolic stability-rat	Train	[0.00, 1.00]	0.49	0.34	0.45	22094
		Test	[0.00, 1.00]	0.44	0.33	0.36	5524

The repeats were used to estimate the error of the individual measurements using the coefficient of variation. The coefficient of variation was calculated for compounds with repeated measurements as the standard deviation of the repeats divided by their mean. The average experimental error of the datasets was estimated as the median coefficient of variation (CoV) of the compounds with repeated measurements. Table 4-3 shows the percentage of compounds with repeated measurements in the training data, which were used to calculate the CoV, and the in-house estimates of experimental error for each dataset that were provided by Eli Lilly. The latter were calculated as the MSD, i.e., the square root of the Minimum Significant Ratio (MSR) that is a statistical parameter that characterises the reproducibility of an assay's measurements in two or more experiments (Haas, Eastwood, Iversen, & al., 2013). The MSR is an estimate of assay variability and it is obtained from historical data using analysis of variance (ANOVA). Both estimates were similar for some datasets but for others, the median CoV estimates were too small and, thus, it was decided that the latter should be used. The estimates of experimental error were used in the validation of the QSAR models, error models and conformal prediction models in the following chapters.

Table 4-3. Percentage of repeated measurements in training set and experimental error estimated as the coefficient of variation (CoV) and the square root of the minimum significance ratio (MSD).

Dataset	Percentage of repeats in training set	Median CoV	MSD
1	5.9	0.072	0.133
2	5.7	0.089	0.087
3	5.9	0.030	0.061
4	6.2	0.078	0.070
5	6.8	0.074	0.044
6	8.6	0.050	0.069
7	100	0.262	0.218
8	100	0.188	0.184
9	4.2	0.027	0.133
10	11.3	0.126	0.182

4.4 Conclusions

This chapter has provided details on the datasets that were used to develop the models in the following chapters. The details provided also include information regarding data processing and data curation steps, as well as the experimental details of the data. Further information on the calculation of molecular descriptors are provided in the next chapter.

Chapter 5 Developing the Underlying QSAR Models

5.1 Introduction

Poor physicochemical and ADME properties are identified as a major cause in the high failure rates of drug candidates during drug development. There is, therefore, a need to have access to methods that can predict these properties, both, accurately and reliably. The aim of this chapter is to build QSAR regression models for the datasets introduced in Chapter 4 using different state-of-the-art machine learning (ML) algorithms and validate their performance. The main objective is to produce QSAR models that will be utilised as underlying models in the investigations of the following chapters, which focus on the estimation of errors in individual QSAR predictions using error models.

5.2 Methods

This section details the methods that were applied for the development of the QSAR models. A summary of the QSAR modelling workflow is illustrated in Figure 5-1. It covers the steps followed for data curation/preparation, the descriptor filtering process, the optimisation of the modelling algorithms and their evaluation using validation techniques. Details on the definition of applicability domains are provided wherever these are applicable. A summary of each of the steps is described first before full details being given below.

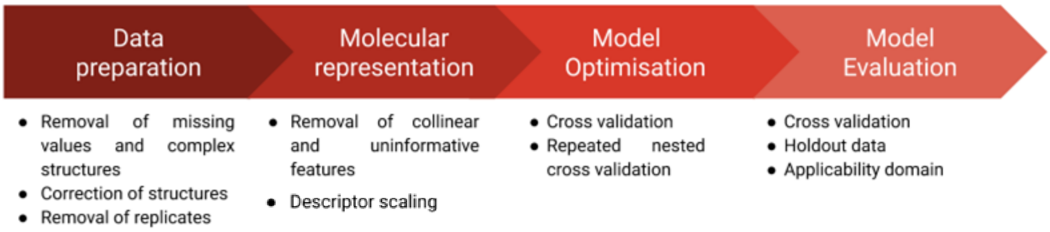


Figure 5-1. Summary of the QSAR modelling workflow

The first step was described in the previous chapter and details regarding the output of data curation were also provided.

The second step dealt with the calculation of molecular descriptors and descriptor selection. Numerical descriptors that are suitable for use in regression analysis and are calculated from the two-dimensional representation of molecules were used throughout this thesis. The presence of collinear descriptors, which carry redundant information and introduce noise to the model, was treated by applying filters based on a correlation threshold and variable importance methods. The descriptors were standardised as this resulted in improved performance for all algorithms.

The third step investigated the optimisation of four regression algorithms and the selection of the best models by means of cross-validation. The final step evaluated the average performance of the models on holdout data and their applicability domain.

5.2.1 Molecular representation and feature selection

The molecular descriptors for all datasets were calculated using the RDKit Descriptor Calculation node in KNIME and consisted of 117 constitutional, physicochemical and topological descriptors in total. The molecular descriptors of the training data were standardised using unit-variance scaling, and the mean and variance values used to scale the training data were applied to the test data, for all datasets.

Redundant information present in the descriptors was removed by excluding collinear descriptors with a pairwise correlation coefficient greater than 0.95 (Pearson's r) in KNIME. Invariant descriptors were also removed. This procedure was followed for all datasets, i.e., the LogD dataset and the ADME datasets.

As discussed in Chapter 3, feature selection may also be applied using the variable importance scores obtained from the PLS and RF algorithms of previously trained QSAR models. Different feature selection methods were investigated for the LogD dataset as described below. Feature selection was not attempted on the ADME datasets due to their large number and the limited benefits seen for the LogD data.

The PLS and RF models were trained on non-invariant (i.e., not constant), near orthogonal descriptors using default settings for both algorithms. The variable importance scores were extracted from the models with the aid of scripts using Python's *sci-kit* learn library. Descriptors were removed as having negligible contribution to the models' performance by applying a lower threshold of 1.0 to their PLS variable importance scores (Tran, Afanador, Buydens, & Blanchet, 2014) and an arbitrary lower threshold of 0.2 to the RF feature importance scores. The two descriptor subsets obtained by PLS and RF variable importance filtering were then used to train LogD models using four regression algorithms.

The Gini impurity score of each descriptor in the RF was used to generate ranking of the feature's importance. Gini impurity represents the number of times the descriptor is selected for growing the trees in the forest and is indicative of the descriptor's contribution to the minimisation of error. The end result was the compilation of four descriptor subsets (Table 5-1); which were then used to train the algorithms with default settings.

Table 5-1. Description of the five descriptor sets compiled by filtering

Descriptor set	**Details**
1	All descriptors
2	Non-invariant, near orthogonal descriptors (r ≤ 0.95)
3	Non-invariant, near orthogonal descriptors (r ≤ 0.95), Feature Importance (RF)
4	Non-invariant, orthogonal descriptors (r ≤ 0.95), Variable Importance in Projection (PLS)

5.2.2 Model optimisation

The parameters of four ML algorithms (PLS, SVM, KNN and RF) for LogD and two algorithms (SVM and RF) for the ADME datasets were optimised using the reduced subset of descriptors (Table 5-1, Descriptor set 2). The optimal parameters for each algorithm were found by implementing a grid search algorithm in a cross-validation loop. All models were built, optimised and validated using Python's *sci-kit learn* library. The gridsearch optimisation was implemented using the model selection function, while the evaluation metrics of the optimum parameters were calculated using the cross-validation and metrics functions available in *sci-kit learn*. The Q^2 values were calculated using a customised script.

A range of parameter values for each modelling method was provided as input to the grid search algorithm, which was used to compute the cross-validated MSE and coefficient of determination measures for all parameter combinations. The algorithm reported the optimum parameter combinations that were evaluated using error minimisation. To prevent overtraining the cross-validation measures of all parameter combinations were plotted to aid visual inspection of the optimum parameters and, subsequently, manual parameter selection. The parameters and the ranges of values that were investigated during the optimisation of each modelling algorithm are shown in Table 5-2.

Table 5-2. Parameters and ranges of values provided as input to the grid search algorithm

Algorithm	**Parameters**	**Range**
PLS	Number of latent variables	1 – 50
SVM	RBF kernel: Gamma Penalty C	$\gamma = (2^{-8}, 2^{-7}, 2^{-6})$ C = (0.1, 1, 2, 5, 10)
KNN	Number of neighbours Averaging weights	1 - 25 Uniform weights or Distance-based weights
RF	Number of trees Size of leaves	50 – 500 1 – 50

5.2.3 Model validation

The average performances of the QSAR models were evaluated using 7-fold cross-validation and on holdout data using the R^2, Q^2, RMSE and MAE measures (Alexander et al., 2015; Consonni et al., 2009). The measures were computed according to the definitions provided in Table 5-3.

Table 5-3. Definition of measures for the evaluation of QSAR models

$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_{tr})^2}$	$RMSE = \sqrt{\frac{1}{N_{ext}}\sum_{i=1}^{N_{ext}}(y_i - \hat{y}_i)^2}$	$MAE = \frac{1}{N_{ext}}\sum_{i=1}^{N_{ext}}\lvert y_i - \hat{y}_i\rvert$
${Q_{F1}}^2 = 1 - \frac{\sum_{i=1}^{Next}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{Next}(y_i - \bar{y}_{tr})^2}$	${Q_{F2}}^2 = 1 - \frac{\sum_{i=1}^{Next}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{Next}(y_i - \bar{y}_{ext})^2}$	${Q_{F3}}^2 = 1 - \frac{\frac{(\sum_{i=1}^{Next}(\hat{y}_i - y_i)^2)}{N_{ext}}}{\frac{(\sum_{i=1}^{Ntr}(y_i - \bar{y}_{tr})^2)}{N_{tr}}}$

y_i:measurement $\hat{y}_i$: prediction
$\bar{y}_{tr}$: mean of training set measurements $\bar{y}_{ext}$: mean of external test set measurements
$N\ or\ N_{tr}$: size of training set N_{ext}: size of external test set

The error distributions estimated from cross-validation and the holdout data were also examined for departure from normality; as a non-normal error distribution may indicate the presence of bias in the model or the lack of an important variable. Normality was confirmed visually, by inspecting the residual error plots and quantile-quantile (QQ) plots of the models' errors and, statistically, by applying the one-sample Kolmogorov-Smirnov (KS) test. A QQ plot is used to compare an empirical distribution function with another theoretical or known distribution function, e.g., Gaussian. The quantiles of the two distributions are plotted in a two-dimensional plane and the distributions may be assumed to be equal only if they overlap with the diagonal (Thas, 2010a). The two-sample KS test is used to evaluate the goodness-of-fit of two distributions. The KS test statistic is defined as the largest absolute deviation between the theoretical distribution function and the empirical distribution function (Thas, 2010b). The one-sample KS test was calculated using Python's *scipy* library.

5.2.4 Definition of the applicability domain

The AD was defined only for the LogD models using traditional AD definitions implemented in QSAR that are independent of the ML method. These were used to qualitatively assess the reliability of the models' predictions for the holdout data and were based on range-based, distance-based and density-based AD definitions (Table 5-4). The AD assessment and the calculations of the reliability estimates were implemented using the Applicability Domain Toolbox in MATLAB (Sahigara, Ballabio, Todeschini, & Consonni, 2014; Sahigara et al., 2012).

Compounds in the holdout test set were classified as inside or outside the domain by the different AD methods for descriptor set 1 (all, 117 descriptors) and descriptor set 2 (82 descriptors). Classification was made using AD thresholds that were calculated using the parameters specified for methods in Table 5-4.

For the leverage method, the threshold h was calculated by setting the warning leverage factor to three. The warning leverage then corresponds to three times the average leverage of p/n compounds, where p is the number of descriptors and n is the size of the training set. The threshold values for both descriptor sets are provided as h_1 and h_2.

The other distance-based thresholds were based on the Euclidean distance of the test compound to the centroid of the training set or its average distance from the 5 nearest neighbours or 9 nearest neighbours, the latter determined following the optimisation of k for values between 1-25 over 1000 iterations on validation data 20% the training set size.

The density-based threshold was applied to the probability density of the holdout test set. The probability density distribution of the training set was estimated using a Gaussian potential for each training set compound with a smoothing factor that is optimised by default. The threshold was set to the value of the 95th percentile of the training set density distribution.

Table 5-4. Domain of applicability methods investigated and thresholds applied

Method	Parameters	Threshold
Range-based		
Bounded box	-	-
Bounded box with PCA	-	-
Distance-based		
Leverage	Threshold factor = 3.0	h_1=0.098 h_2=0.066
Distance from centroid	Euclidean	d_1=15.22 d_2=12.40
Distance KNN - fixed K	Euclidean, K=5	d_1=8.30 d_2=7.54
Distance KNN - variable K	Euclidean, K_1=10, K_2=11	-
Density-based		
Potential functions	Gaussian kernel Smoothness $_1$ = 0.1 Smoothness $_2$ = 0.9	Threshold = p95

5.3 Results

The following sections present the results from the evaluation of the underlying LogD and ADME models. Detailed results from the optimisation of the PLS, KNN, SVM and RF LogD models are provided and include the results from their evaluation using 7-fold cross-validation and a single holdout test set. The evaluation of each of the ADME models was conducted using 10-fold cross validation and a single holdout test set.

5.3.1 LogD dataset

Descriptor selection by filtering, based on the exclusion of collinear descriptors (linear correlation <0.95), and the variable importance thresholds of FI (threshold=0.01) and VIP (threshold =0.1), resulted in a total of 82, 31 and 23 descriptors, respectively. The ranking agreement of the two variable importance methods was measured using Spearman's rank order correlation coefficient, which is a nonparametric correlation measure that shows the association between the ranks of two sets of data. The calculated value of Spearman's rho was -0.36, which indicates weak, negative correlation of the ranked features obtained by RF and PLS.

The ten highest ranking descriptors based on their importance computed by the RF and PLS algorithms are shown in Figure 5-2. Besides the expected high importance of the SlogP descriptor, which is an atom-based estimate of

LogP (Wildman & Crippen, 1999), other important descriptors identified by both methods were the number of hydrogen bond donors (NumLipinskiHBD), the number of acyclic oxygen atoms (MQN10) (Nguyen, Blum, Van Deursen, & Reymond, 2009) and the molecule's van der Waals surface area contributing to a predefined interval of LogP values (slogP_VSA10) (Labute, 2000). The RF algorithm gave a higher importance to descriptors based on surface area contributions to partial charge (peoe_VSA) and logP (slogP_VSA), though it is clear that the first two features contribute the most to the impurity reduction criterion of RF. On the other hand, PLS gave higher importance to molecular topological counts, such as the number of aromatic rings, rings, hydrogen bond donors and amide bonds, cyclic divalent nodes (MQN30), acyclic oxygens (MQN10) and 6-membered rings (MQN36).

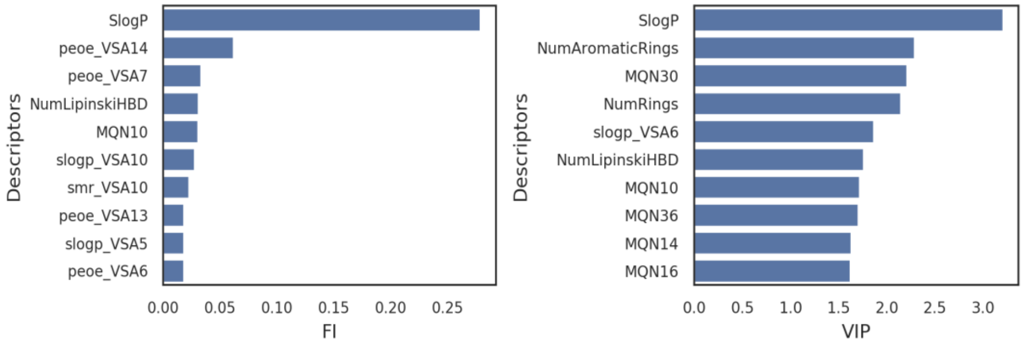


Figure 5-2. Ten most important descriptors based on their rankings from RF's feature importance scores (left) and PLS's (right) variable importance in projection scores

Given that the feature importance ranks depend on the performance of the algorithm the ranking is sensitive to model parametrisation. Consequently, although the importance rankings in Figure 5-2 illustrate the most important features for the LogD models; the robustness of the results could be investigated by resampling of the models' parameters.

The performance measures computed for the modelling algorithms trained on the full descriptor set and the three descriptor subsets are shown in Table 5-5. Filtering of redundant information and collinear descriptors is seen to improve the accuracy of all algorithms, except KNN. With subsequent application of the FI threshold the accuracy of KNN and PLS increases; while it decreases for SVM and, surprisingly, RF. On the other hand, the use of the VIP filter results in similar improvement in the performance of PLS but deterioration in the performance of the other algorithms.

While there is no value in maintaining descriptors that do not contribute to model performance, it is seen that the RF and SVM algorithms are able to yield accurate models without feature selection. Yet, their accuracy improves by excluding collinear variables that carry redundant information. As for KNN, removal of collinear descriptors followed by the FI filter yields the best results. Furthermore, despite the ability of PLS to extract the underlying latent variables; the performance of the algorithm improves with prior removal of collinear descriptors and with the application of either variable importance method.

Table 5-5. Performance of algorithms using default parameters for the four descriptor subsets

Descriptor set	Model	R^2	RMSE	R^2 (CV)	RMSE (CV)	R^2 - R^2 (CV)
1 All (117)	KNN	0.68	0.679	0.48	0.859	0.20
	PLS	0.24	1.040	0.23	1.044	0.01
	RF	0.92	0.343	0.55	0.803	0.37
	SVM	0.74	0.615	0.59	0.765	0.15
2 LC (82)	KNN	0.66	0.693	0.46	0.874	0.20
	PLS	0.25	1.032	0.24	1.038	0.01
	RF	0.92	0.333	0.56	0.786	0.36
	SVM	0.75	0.595	0.59	0.759	0.16
3 FI (31)	KNN	0.70	0.658	0.52	0.829	0.18
	PLS	0.27	1.020	0.26	1.025	0.01
	RF	0.92	0.341	0.55	0.795	0.37
	SVM	0.72	0.629	0.59	0.766	0.13
4 VIP (23)	KNN	0.67	0.686	0.46	0.875	0.21
	PLS	0.26	1.025	0.26	1.028	0.00
	RF	0.92	0.343	0.53	0.815	0.39
	SVM	0.64	0.721	0.52	0.825	0.12

It is understood that feature selection introduces bias to the model by restricting the model's applicability domain and, by doing so, limits the model's ability to generalise for new instances and identify outliers (Eriksson, 2003, Hawkins, 2004). Therefore, the decision was made to proceed with model parametrisation using descriptor set 2 that is the output of filtering collinear descriptors.

The optimisation curves are shown in Figure 5-3 and illustrate the change in the R^2 and mean squared error (MSE) for the parameter values optimised for each of the KNN, PLS, RF and SVM algorithms.

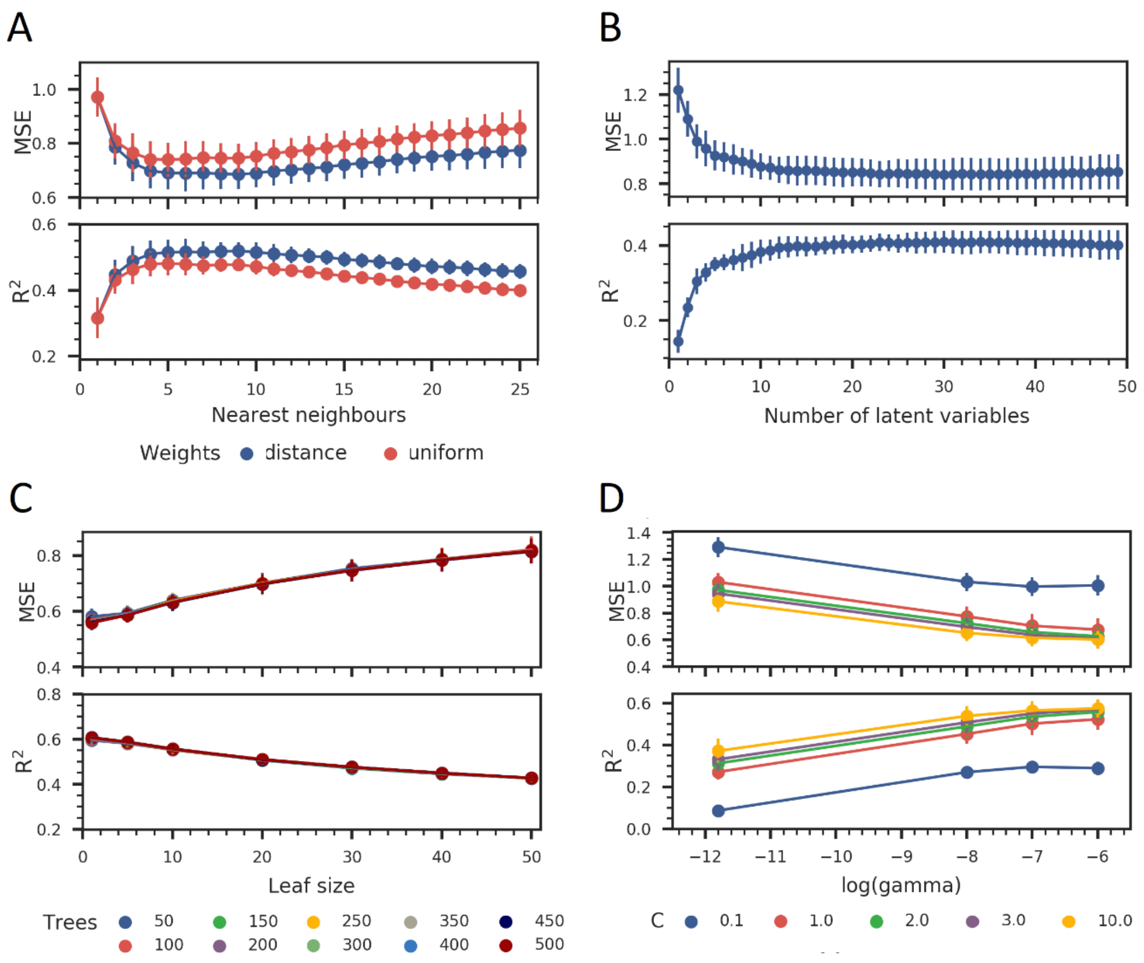


Figure 5-3. Optimisation of A) the number of nearest neighbours and weighting method of KNN, B) the number of latent variables in PLS, C) the number of trees and leaf size of RF and D) the C and gamma parameter of the RBF kernel in SVM

The best parameters found with grid search-based optimisation are provided in Table 5-6 with the metrics from the evaluation of the algorithms' performance.

Table 5-6. Optimum parameters identified by the grid search algorithm

Algorithm	**Parameters**	**R^2**	**RMSE**	**R^2 (CV)**	**RMSE (CV)**	**R^2 - R^2 (CV)**
KNN	K= 9	1.00	0.029	0.52	0.827	0.48
PLS	LV=30	0.46	0.879	0.41	0.916	0.05
RF	Leaf size=1, Trees=500	0.95	0.274	0.61	0.747	0.34
SVM	C=10, gamma= 2^{-6}	0.98	0.178	0.64	0.710	0.33

The grid search algorithm is strictly driven by error minimization and as a result it can easily lead to over-trained models and underestimate the prediction error of novel compounds. In Table 5-6, overtraining is evident from the large difference between the fitted R^2 and the cross-validated R^2 ($\Delta R^2 > 0.3$). For this reason, it was decided to identify an alternative set of parameters (Table 5-7) through visual inspection of the algorithms' optimisation curves based on the cross-validation performance measures. The parameters were selected by taking into account the rule of parsimony according to which the most generalizable model with similar performance should be chosen. This approach resulted in less optimistic performance metrics and reduced the difference between the fitted and cross-validated R^2 in the case of the RF and SVM algorithms but increased in the case of KNN and PLS.

Table 5-7. Optimum parameters identified from visual inspection

Algorithm	Parameters	R^2	RMSE	R^2 (CV)	RMSE (CV)	R^2 - R^2 (CV)
KNN	K= 5	1.00	0.029	0.50	0.846	0.50
PLS	LV=5	0.46	0.879	0.35	0.962	0.11
RF	Leaf size=5, Trees=250	0.86	0.449	0.59	0.766	0.27
SVM	C=2, gamma= 2^{-7}	0.76	0.591	0.60	0.753	0.16

In Figure 5-4, visual assessment of the final QSAR models was carried out by overlaying the cumulative distributions of the actual and predicted endpoint values of the cross-validation and holdout data. Though scatter plots are the most common way of visually assessing the quality of a regression model, the plots in Figure 5-4 clearly summarise the ability of the different ML algorithms to reproduce the LogD data distributions. Smaller distances in the predicted cumulative density distribution (CDD) of the cross-validation data from the actual CDD, in comparison to the CDDs for the holdout data, indicate that the performance of the algorithms on the cross-validation data is higher. The result for KNN and SVM are better in estimating the full data distribution, while a larger distance of the PLS and RF CDFs from the actual data distribution indicates that these algorithms performed less accurately on the tails of the distribution. In addition, PLS generated predictions outside the response value range in the cross-validation data. These results are also reflected in the ranking of the algorithms produced by the Kolmogorov-Smirnov (KS) statistic from the two sample KS test, which was applied between the actual measurements and the predictions of each algorithm for the holdout data (Table 5-8).

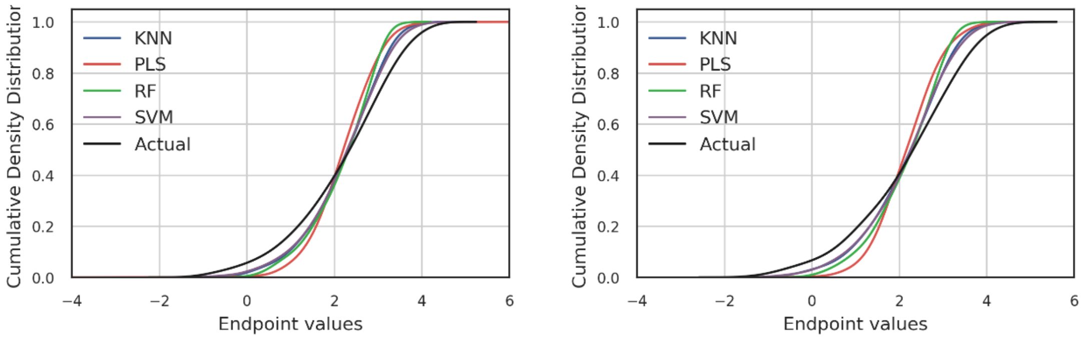


Figure 5-4. Cumulative density distributions of QSAR predictions and the actual measurements of cross-validation (left) and holdout data (right).

Table 5-8. Two sample Kolmogorov-Smirnov test for holdout measurements predictions

Model	KS statistic	p-value
KNN	0.143	0.00
PLS	0.181	0.00
RF	0.169	0.00
SVM	0.122	0.03

The model’s performance measures on cross-validated and holdout data are provided in Table 5-9. The estimates of the model’s accuracy for prospective predictions based on cross-validation are similar to the estimates of holdout data, yet slightly optimistic with a difference of 0.01 - 0.05 units. However, a limitation is that these could vary for other holdout samples and further investigation using resampling methods or other data partitioning approaches would have to be incorporated into the modelling process.

Table 5-9. Performance measures estimated from cross-validation and holdout data.

Model	Parameters	R^2 (CV)	RMSE (CV)	Q^2	RMSE (Holdout)
KNN	K= 5	0.50	0.846	0.48	0.859
PLS	LV=5	0.35	0.962	0.28	1.012
RF	Leaf size=5, Trees=250	0.59	0.766	0.55	0.804
SVM	C=2, gamma= 2^{-7}	0.60	0.753	0.57	0.784

The residual error distribution of each model was tested for departure from normality by plotting the quantiles of the residual distribution against the quantiles of the theoretical, normal distribution (Figure 5-5).

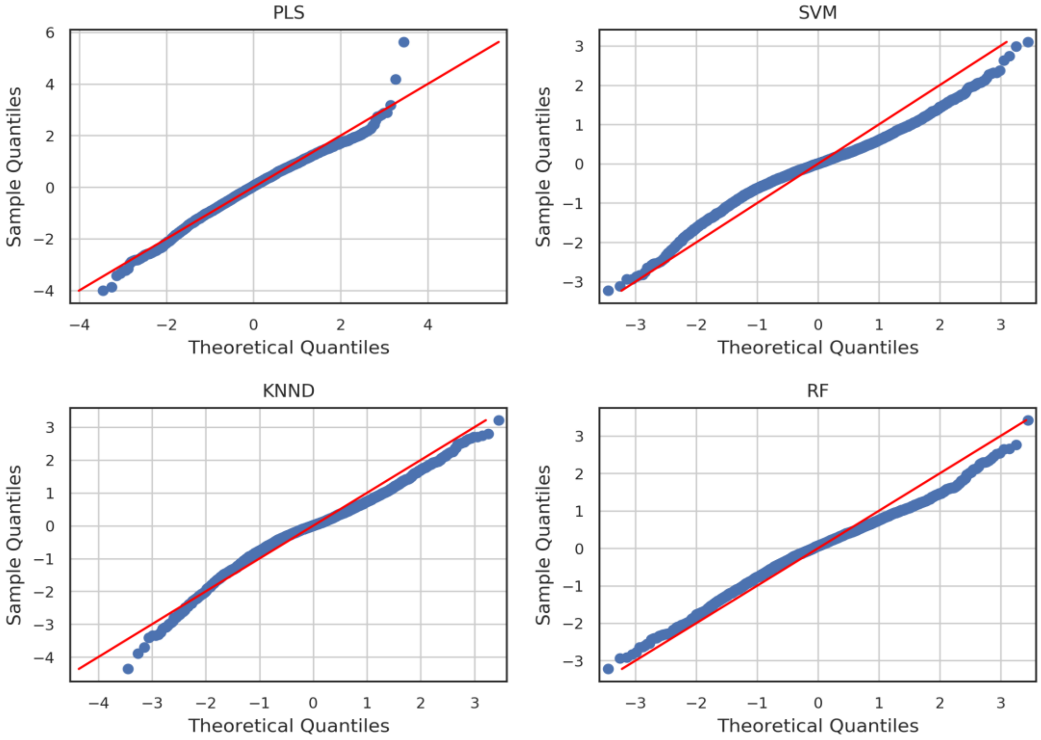


Figure 5-5. Q-Q plots of the models' errors departure from normality.

Given the absence of major trends indicating departure from normality in Figure 5-5, it is reasonable to assume that the models' error distributions are, approximately, normally distributed. The results were confirmed quantitatively by applying a one sample KS test, which produced a KS statistic of 0.5 and p=0.0 for the residuals of all models (see Appendix, A 1).

Evaluation of the reliability of the models' predictions for the holdout data using the different AD methods produced varied results. Table 5-10 shows the percentage of the holdout test set that was classified as outside the AD defined by the individual methods using the original set of descriptors calculated, descriptor set 1, and the filtered set of descriptors, descriptor set 2 (see Table 5-1 for details). The joint AD was calculated by taking into account the predictions that were classified as out-of-domain by at least four AD methods. This was done to investigate the overlap in the results produced the different AD definitions. The density method produced surprisingly different results for the different descriptor sets, classifying most of the holdout data as outside the applicability domain using descriptor set 2. The difference between the number of out-of-domain compounds identified by the range-based and distance based methods using different descriptor sets is much smaller.

Table 5-10. Statistics of the evaluation of AD for the holdout data using descriptor sets 1 and 2

Method	**Outside AD (%)**	
	Descriptor set 1	**Descriptor set 2**
Range-based		
Bounded box	0.3	0.3
Bounded box with PCA	0.8	0.8
Distance-based		
Leverage	3.3	2.5
Distance from centroid	5.1	4.9
Distance KNN - fixed K	5.5	5.2
Distance KNN - variable K	4.7	3.9
Density-based		
Potential functions	0.0	91.8
Joint AD (exc. density based)	1.3	1.7

Table 5-11 compares the mean absolute errors (MAEs) calculated when no applicability method is applied to the holdout test set, indicated as None, with the MAEs calculated for compounds classified as inside (In) and outside (Out) the AD specified using each method, respectively. Highlighted in bold are the results for which the MAE is reduced following the removal of predictions for compounds that were out-of-domain. Indicated in italics are results were the MAE increases following the application of the AD. It is seen that the most efficient AD is obtained using the distance-to-model definition with a variable number of neighbours.

Table 5-11. Mean absolute residual errors of holdout test data classified as in/out of the AD defined using different methods (calculated for descriptor set 2)

AD method		Number of compounds	KNN	PLS	RF	SVM
None	-	596	0.620	0.786	0.592	0.556
Bounding box	In	594	0.618	0.784	0.587	0.553
	Out	2	1.354	1.351	1.902	1.358
Bounding box with PCA	In	591	0.619	0.784	0.591	0.556
	Out	5	0.729	1.020	0.658	0.592
Distance from centroid	In	567	0.618	0.782	0.585	0.557
	Out	29	0.658	0.875	0.718	0.542
Distance - fixed k	In	565	0.620	*0.787*	0.587	0.555
	Out	31	0.628	0.772	0.670	0.570
Distance - variable k	In	573	0.616	**0.778**	**0.580**	**0.548**
	Out	23	0.734	0.986	0.882	0.739
Leverage	In	577	**0.614**	0.785	0.584	0.556
	Out	19	0.825	0.837	0.830	0.565
Joint	In	586	0.618	0.783	0.587	0.555
	Out	10	0.743	1.007	0.881	0.630

The AD outliers that were identified by all methods (excluding the density-based method) are highlighted in the residual plots of the holdout data for models in Figure 5-6. Although most of the AD outliers lie within the boundaries of 2 and 3 standard deviations of the residual errors, indicated by the red lines in Figure 5-6, it could be argued that these should be removed and the model retrained, however, this was not done here.

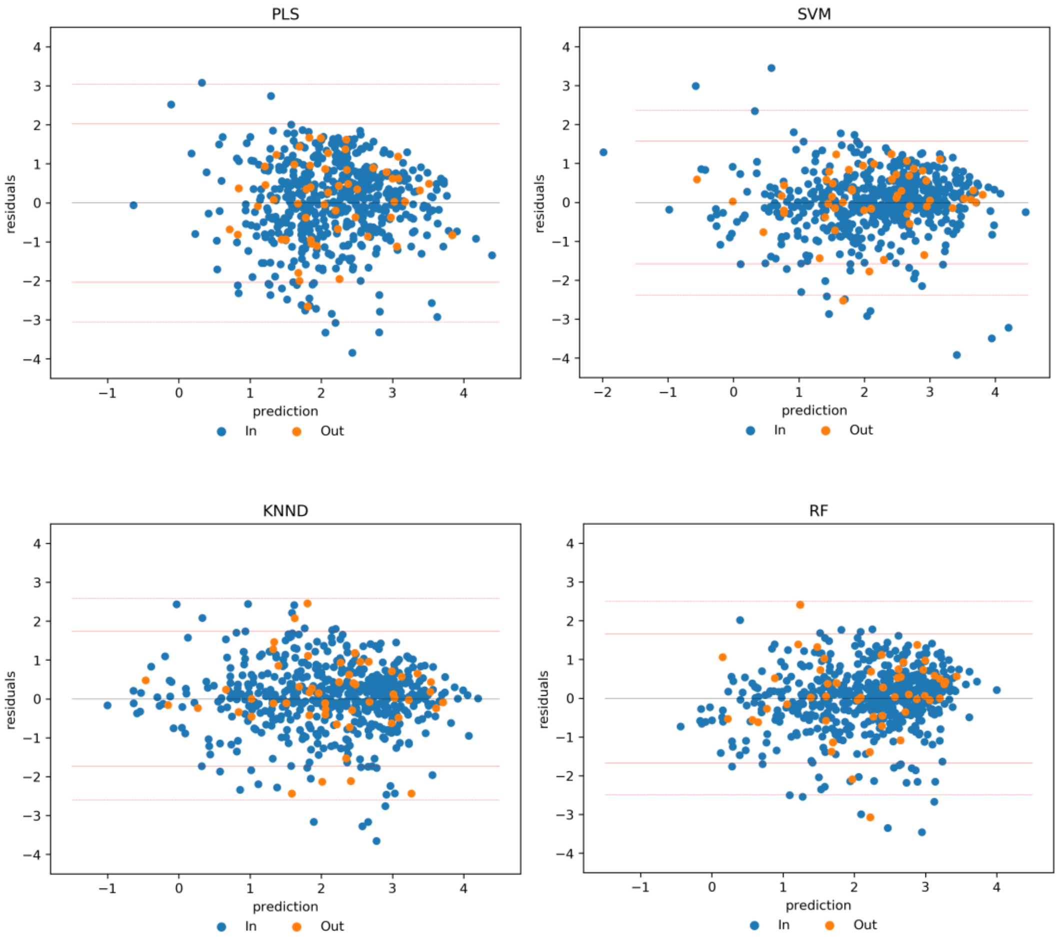


Figure 5-6. Residual plots of the model's holdout data indicating whether predictions are inside and outside the model's joint AD

There are 10 compounds that were identified as out-of-bounds by at least four AD methods representing 1.7 % of the holdout data. A total of 52 compounds were classified as out-of-domain by combining the results of all AD definitions, by at least one AD method. Surprisingly, only 1 compound was identified by all methods and this had a residual error between 1.05 - 1.5 across the four LogD models, thus, it may not be considered a residual error outlier. The distance-based methods were more successful in identifying predictions with large residuals than was the consensus of the methods, i.e., based on the joint AD, yet, many compounds that were poorly predicted by the four ML algorithms failed to be identified as out-of-domain by any method.

This highlights the fact that the assessment of the reliability of predictions using these AD definitions is rather simplistic as it does not take into account the modelled response and, practically, implies a linear relationship between the similarity of the compounds and their accuracy of prediction. In addition, the ML algorithms used to build the QSAR models are applying a more complex statistical treatment that cannot be explained by a simple range-based or density-based reliability metric. These methods may be useful in verifying the theoretical assumptions of the QSAR experiment, such as for example, testing whether a prediction is the result of an interpolation or extrapolation in the model’s AD as defined by its descriptors. However, many of the compounds that were classified as outside the AD were actually predicted well therefore these methods may not be very useful for identifying mis-predicted compounds by ML algorithms.

5.3.2 ADME datasets

Regression models based on the SVM and RF algorithms were trained using the best parameters obtained from grid search optimisation. These are reported in the Appendix (see A 2). Despite many of the models being over trained, these were chosen as they yielded models with the highest accuracy for cross-validation and holdout data. Model performance was evaluated using 10-fold cross validation and by resampling the hold-out data 10 times. The predictive performance of SVM and RF models estimated by 10-fold cross-validation is illustrated in Figure 5-7.

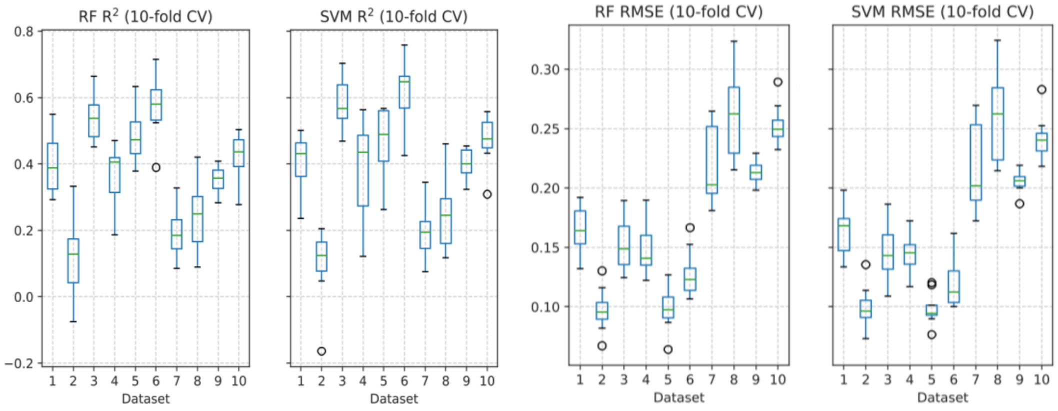


Figure 5-7. Distributions of predictive R^2 and RMSE estimates (by fold) of RF and SVM models from 10-fold cross-validation.

Both algorithms produced models with similar performance on cross-validation data. The cross-validated R^2 and RMSE values are compared with the statistics obtained on the holdout data, in Table 5-12. It is seen that the SVM models for datasets 1, 3, 6 and 10 were more accurate than the RF models by a small margin in the range of 0.004 - 0.011. The overall accuracy of the QSAR models ranges between, approximately, 10 – 30 % of the endpoint values; with the models for the largest datasets being the least accurate. Dataset 2 produced ADME models with the smallest error on cross-validation and holdout data, however, the measurements in the dataset were accumulated in a smaller range of values.

Table 5-12. Average model performance of SVM and RF algorithms estimated from cross validation and holdout data

Dataset	Model	R^2 (CV)	RMSE (CV)	R^2 (Holdout)	RMSE (Holdout)	Assay error
1	RF	0.33	0.170	0.41	0.170	0.133
	SVM	0.36	0.166	0.41	0.169	
2	RF	0.19	0.094	0.31	0.070	0.087
	SVM	0.15	0.096	0.28	0.071	
3	RF	0.55	0.151	0.59	0.142	0.061
	SVM	0.61	0.140	0.61	0.137	
4	RF	0.35	0.151	0.51	0.129	0.070
	SVM	0.36	0.149	0.53	0.126	
5	RF	0.45	0.104	0.52	0.091	0.044
	SVM	0.45	0.104	0.54	0.090	
6	RF	0.52	0.132	0.64	0.117	0.069
	SVM	0.57	0.125	0.69	0.110	
7	RF	0.20	0.221	0.51	0.183	0.218
	SVM	0.20	0.221	0.33	0.214	
8	RF	0.26	0.260	0.68	0.180	0.184
	SVM	0.25	0.260	0.45	0.230	
9	RF	0.32	0.220	0.42	0.201	0.133
	SVM	0.31	0.218	0.34	0.212	
10	RF	0.33	0.273	0.45	0.247	0.182
	SVM	0.37	0.265	0.50	0.237	

The accuracy of the models was compared to the available experimental error estimates by applying a criterion based on the 3σ rule; whereby a model with an accuracy estimate that does not exceed 3 times the assay variability estimate is considered suitable for use (Haas, 2004). The rule was applied by calculating the ratio between the average model error estimate and the experimental error estimate and setting an upper threshold of 3 (results not shown). This condition was satisfied for the models of all datasets; however, a ratio greater than 2 for the ADME models of datasets 3, 4 and 5 suggested that the error was closer to the threshold.

If a model has an average error value that is smaller than the assay variability estimate this suggests that the model is more accurate than the assay method. This is observed for the average SVM and RF error estimates obtained from holdout data for datasets 2 and 7, and the RF model of dataset 8. These three datasets were also the most difficult to model and performed less well in cross-validation. The large difference between the R^2 and RMSE values of the models in cross-validated and holdout data, suggests that the models perform better on the holdout data. This is also observed between the accuracy estimates obtained from cross-validation and holdout data for all datasets but dataset 1 and it is attributed to the presence of measurement bias in the data, which is discussed below.

The low RMSE values for the models of dataset 2 and dataset 5 are partly attributed to the narrow data distributions rather than high predictive performance. The implications of this would have been clear if the normalised RMSE had been used for the assessment of accuracy, which would allow comparison of accuracy across data with different values and distributions. (Normalised RMSE is used in the following chapters to compare the accuracy of QSAR models and error models). Poor performance was also observed for datasets 7 and 8 (average predictive R^2 <0.3 and average RMSE > 0.2), which have wide distributions and high variation in the response.

The metabolic stability datasets (7, 8, 9 and 10) were the largest in size and, thus, model accuracy was expected to be higher. However, they proved to be the least accurate models as they consisted of greater assay error and variation in the response distribution.

Figure 5-8 illustrates the shape of the ADME models' error distributions obtained from the signed residual errors of cross-validation data. A striking observation is that the range of the residual error distributions is very large; particularly, for the larger datasets, i.e., datasets 8, 9 and 10. The presence of heavy tails on the residual error distributions is a consequence of measurement bias in datasets 3, 4, 5, 6, 7 and 8 and shows that normality of the error distributions cannot be assumed. This was also confirmed by visual inspection of the residual error histograms (see Appendix, A 4). Although the theoretical assumption of normally distributed errors is important in regression analysis, it may not be strictly followed when applied to real data nor is it a requirement for non-parametric methods and ML algorithms. Model errors that are non-Gaussian distributed may also indicate the lack of important variables in the models, however, additional descriptors were not investigated in this thesis and the presence of measurement bias was quite clear, after inspecting the endpoint distributions. ML algorithms generally perform well close to the dataset response mean and are less accurate as the distance from the mean increases (see Appendix, A 2 and A 3), therefore the errors will also be larger where the data is sparser. This is a problem of imbalanced data and is easily addressed in categorical data for classification by techniques such as oversampling or under-sampling; but the treatment of continuous data that are unevenly sampled is less straightforward.

Excluding outliers in the endpoint value range from model training is not a suitable option as this would represent a reduction of the models' applicability domain and result in the deterioration of the models' performance on external data, thus, outliers in the data have been retained.

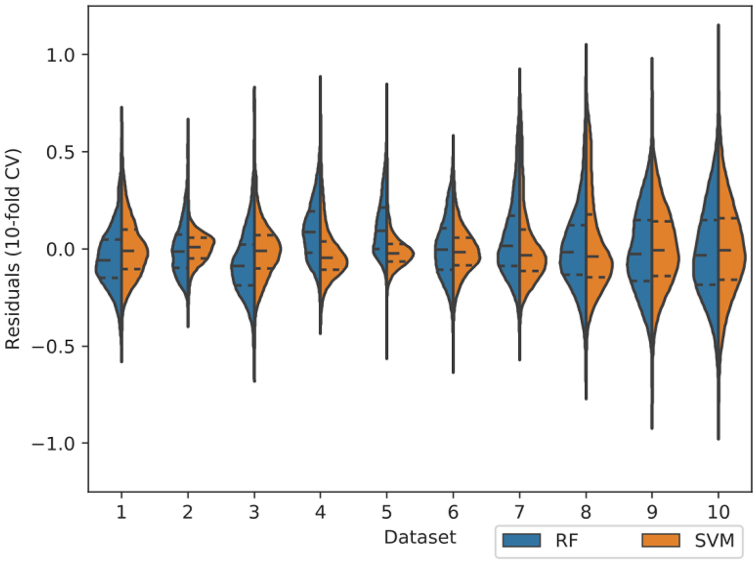


Figure 5-8. Residual error distributions of RF and SVM models from 10-fold cross-validation

The distributions of the RF signed residuals differ from the SVM signed residuals on several datasets (Figure 5-8); as they are shifted to positive values, indicating that the RF has overestimated. This is more obvious for the distributions of datasets 2, 4, 5 and 7.

The reliability assessment based on the AD was not applied to these datasets. However, alternative methods for the assessment of prediction reliability using error models are investigated in the following chapter.

5.4 Discussion

Models of reasonable predictive performance have been obtained for the LogD and ADME datasets using ML algorithms without the requirement of complex tuning. Nevertheless, the use of the automated gridsearch algorithm for model optimisation produced models that were overtrained.

From the LogD models, and with regard to descriptor selection, it was found that treatment of highly collinear descriptors improved the accuracy of PLS, RF and SVM algorithms but not in the case of KNN. The accuracy of PLS improved with the subsequent application of the PLS-based and RF-based feature importance methods; however, the predictive performance of the PLS model was very low. In contrast, there was no improvement in the performance of the RF and SVM models following descriptor selection using feature importance methods. The highest improvement following the application of RF’s feature importance method was observed for KNN. It was also seen that cross-validation estimates were generally more optimistic than the holdout data. However, more robust estimates of the model’s prediction error may be obtained using nested cross-validation or by applying resampling techniques. The results showed that SVM and RF algorithms consistently provided more accurate predictions than PLS and KNN. Analysis of the models' residual errors showed that approximate normality may be assumed. Minor deviations from normality in the residual errors are observed either as an effect of the modelling algorithm, thus, indicating the model's lack of an important variable; or measurement bias in the data.

The reliability of the LogD predictions was evaluated using applicability domain methods that do not take account of the modelled response or the model's errors and are, therefore, non-model specific. A main drawback of these approaches is that they cannot efficiently identify compounds that are poorly predicted by ML algorithms. Consequently, predictions with high accuracy were considered unreliable, while many predictions with low accuracy were considered to be reliable. Yet, the distance-based applicability domains were more efficient than the range-based approaches for all modelling algorithms. Although the evaluation of the model's applicability domain is concerned with the theoretical assumptions that are implicit in the QSAR model, rather than the statistical assumptions of the modelling algorithm, it is important to consider both aspects in the detection of unreliable predictions to avoid spending resources on testing compounds that are accurately predicted by the QSAR model.

The RF and SVM algorithms produced models of similar overall performance for the ADME datasets. From the average performance of the models and the error distributions of the models, it became clear that, although the models were able to predict the mean response of the dataset well, they were less accurate in predicting instances that were further away from the response mean. This revealed that some of the datasets were biased; the measurements were either restricted to a short range of values or the measurement distributions were severely skewed, i.e., imbalanced. Data that were associated with larger experimental errors also had very broad error distributions regardless of the ML algorithm used to build the models.

5.5 Conclusions

This chapter focused on the development of physicochemical and ADME property regression models using established ML algorithms that will be further studied for the estimation of their prediction errors. The RF and SVM algorithms produced the most accurate models for all datasets, although in the case of several ADME datasets the errors were found to be non-normally distributed. Analysis of the results from the validation of the LogD models using AD definitions suggested that AD-based reliability estimates are not good indicators of the accuracy in predictions of ML algorithms as they do not take into account any information about the modelled response or the models' errors. This suggests that other methods which produce reliability estimates that correlate with the accuracy of the models' predictions need to be considered. In the following chapters, we investigate the use of error models as an alternative approach to the definition of AD for the assessment of prediction reliability and focus on the assessment of their performance.

Chapter 6 Prediction Error Estimation

6.1 Introduction

The existing approaches utilising machine learning methods for the estimation of uncertainty in individual QSAR predictions were discussed in Chapter 3. Previous studies have reported methods that are able to distinguish between accurate and inaccurate predictions; yet, their performance on ADME data is often poor (Sheridan, 2013; Toplak et al., 2014). The focus of this chapter is to investigate the performance of machine learning algorithms for the estimation of errors in QSAR models and assesses the usefulness in confidence estimation. Error models are developed with the objective of estimating the prediction errors of the underlying models built in the previous chapter. Details regarding the variables, methods and measures for the evaluation of the error models are described in the sections below.

6.2 Methods

Two error model methods are evaluated on their ability to estimate the prediction errors of individual QSAR predictions and on their ability to rank predictions based on their actual prediction errors. These are both based on RF regression but using different types of features: namely descriptor-based features and AD-based metrics. The performance of the two error based models are compared with a novelty detection method, i.e., the direct use of an AD distance-to-model (D2M), and bagged ensembles as baseline methods. Each method is applied to the underlying QSAR models that were built in the previous chapter using the LogD and ADME datasets described in Chapter 4.

The baseline methods are described first followed by the error model methods. The two different methods used to evaluate performance are then described.

6.2.1 Binned D2M-based model

As discussed in Chapter 3, distance-to-model methods rely on the assumption that compounds with a greater degree of extrapolation from the model's AD will be predicted less accurately than those which are closer to the AD. This section describes the development of the baseline D2M error models that were used to evaluate the error models for the LogD dataset. These were built using the D2M indices and the cross-validation errors of the underlying models. First, the D2M indices were binned and the mean prediction error of each bin was calculated;

then, linear regression was applied to the mean D2M and mean error of each bin. Binning was applied in KNIME using the Auto-binner node and linear regression was run using the linregress function using Python's *scipy* library.

The D2M indices of the training data were calculated using the descriptors of the underlying LogD models built in the previous chapter. The descriptors are described in the Methods section of Chapter 5. The D2M index of each training set compound was calculated as its mean Euclidean distance to its three nearest neighbours, which is defined in Chapter 3. The Euclidean distances were calculated using the distance function in Python's *scipy.*

The D2M indices were ordered and then distributed into non-overlapping bins of equal frequency. The prediction error estimate of each bin was derived as the mean cross-validation error of the predictions assigned to each bin. The performance of the linear regression model was investigated for different numbers of bins, specifically 10, 20, 50 and 100 bins. The D2M indices of the holdout data were calculated as the mean distances to their three nearest neighbours in the training set and used to assign prediction error estimates to the LogD predictions obtained from their corresponding bins.

6.2.2 Bagged ensembles

Resampling techniques are widely applied in statistical inference for the estimation of uncertainty. In machine learning, ensemble methods implement resampling techniques to construct multiple models by sampling the training data or the data variables. The uncertainties of the individual predictions are estimated as the standard deviations of the predictions in the ensemble. Several studies in QSAR have suggested that the standard deviation of ensemble predictions correlates well with prediction accuracy (Kaneko & Funatsu, 2014; Tetko et al., 2008). Here the standard deviation of ensemble predictions is used as a benchmark for the performance of the LogD and ADME error model estimates.

Ensembles were constructed for both the LogD and ADME datasets by applying bootstrap sampling on the data and the features of the underlying models that were reported in the previous chapter. Instead of using the cross-validation errors, as was done for all other error estimation methods in this chapter, the out-of-bag data of the bootstrap samples, i.e., the data that were omitted due to sampling with replacement, were used. The size of the ensembles was varied between 10, 100 and 1000 models. The parameters of the ensemble models applied were the parameters that were previously identified from the (visual) optimisation of the underlying models in Chapter 5. Prediction error estimates were directly obtained as the standard deviations of the ensemble predictions for out-of-bag predictions and the holdout test data.

6.2.3 Regression error models

According to Sheridan (2004), a single reliability method may not sufficiently explain the prediction errors of all compounds, nor is it necessary that it encodes their molecular structure in the same way as the QSAR model. Thus, supervised learning algorithms may be used to explore the non-linear relationships between several AD-based reliability metrics and the prediction errors of the QSAR models. Specifically, the RF algorithm has the functionality of identifying important variables from a wide range of descriptors and contains an in-built validation method based on out-of-bag estimates. In Sheridan's work (2013), RF error models were used to estimate the errors of RF QSAR models using the structural similarity of the test compound to its first nearest neighbour, the QSAR prediction and the standard deviation of the QSAR prediction across the ensemble as model features. In this study, RF error models are built for the underlying QSAR models that were developed in Chapter 5 and to predict the errors in holdout data. These RF error models were built using QSAR descriptors and AD-based metrics as features.

Descriptor-based error models were generated by training the RF algorithm on the cross-validation errors and the descriptors of the underlying models built in Chapter 5. The parameters of the RF algorithm were set to a node size of 10 and 200 trees.

The AD-based error models were trained on the cross-validation errors of the underlying models built in Chapter 5 and features which combine AD-indices with predictions of the underlying model, similar to the approach suggested by Sheridan. The features included the mean and standard deviation of the D2M index, the prediction of the underlying model, and the standard deviation of the QSAR prediction, if the underlying algorithm was a RF. The D2M was calculated as the average Euclidean distance of the test compound to its three nearest neighbours in the training set using the nearest neighbour algorithm in Python's *sci-kit learn* library. The standard deviation of the distance-to-model metric was included as an additional descriptor to capture the local variation in the test molecules' D2M metric space.

Two distance-weighted error metrics described in Sheridan (2013) and Keefer, Kauffman, & Gupta (2013), which capture the continuity of the QSAR relationship in the local neighbourhood of the test molecule, were also considered. These are defined in Equation 6.1 and Equation 6.2 below. The first, wRMSD1, is the weighted difference between the QSAR prediction for the test molecule, $\hat{y}_M$, and the observations, y_i, of its k=3 nearest neighbours in the training set. The second, wRMSD2, is the weighted difference of the prediction for the k nearest neighbours of the query molecule M, $\hat{y}_k$, and their observations, y_i. The weights are given by the inverse of the distance between the query molecule and the i^{th} neighbour. However, neither of these were found to improve the model's performance and they were, therefore, abandoned.

$$wRMSD1 = \sqrt{\frac{\sum_{i=1}^{k} w_i^2 (\hat{y}_M - y_i)^2}{\sum_{i=1}^{k} w_i^2}} \qquad 6.1$$

$$wRMSD2 = \sqrt{\frac{\sum_{i=1}^{k} w_i^2 (\hat{y}_k - y_i)^2}{\sum_{i=1}^{k} w_i^2}} \qquad 6.2$$

All error models were trained on the absolute residual errors obtained from cross-validation. For the LogD error models, these were obtained from 7-fold cross-validation; while for the ADME error models there were obtained from 10-fold cross-validation. Although other functions of error were tried on several datasets (signed error, logarithm, square root, square) in preliminary experiments; they did not improve the error models' performance (results not shown); thus, subsequent error models were based only on absolute errors.

6.2.4 Evaluation of error models using correlation

The regression error models obtained were initially assessed for their predictive performance on cross-validation and holdout data, using the standard regression metrics of R^2 and the RMSE, defined in Chapter 4. The estimates obtained from the ensemble models and the regression error models were also assessed using the linear and rank correlation of their estimates with their actual errors. These evaluations were conducted using Pearson's correlation coefficient, r, and Spearman's rank correlation coefficient, rho, respectively. While Pearson's r is a parametric method and requires that the data are normally distributed, Spearman's rho (ρ) is non-parametric as it is applied to the ranks of the data. Although residual errors are, generally, assumed to follow a normal distribution this may not be the case for the obtained prediction error estimates. The calculation of Pearson's r for two variables x and y is given in Equation 6.3:

$$Pearson's\ r = \frac{covariance_{xy}}{s_x s_y} \qquad 6.3$$

where s_x, s_y are their respective standard deviations. Spearman's ρ is calculated using the same equation but by replacing the continuous variables with their ranks. The values of correlation coefficients range between -1 and +1, which indicate a perfect negative and a perfect positive correlation, respectively. Pearson's r and Spearman's ρ were calculated in KNIME using the linear and rank correlation nodes, respectively.

The ranking agreement between regression error model estimates and the benchmark, i.e., the standard deviation of the ensemble, was also assessed using Spearman's correlation coefficient. The ranking agreement of all error estimates is also assessed using Kendall's coefficient of concordance, W. Kendall's W is a non-parametric correlation coefficient that is used to express the level of agreement between more than two ranking methods. The values of Kendall's coefficient of concordance range between 0 and 1; with 0 indicating complete disagreement between all the ranks obtained from all error estimation methods and 1 indicating their complete agreement. The calculations of the coefficient were applied with a custom script in Python using Equation 6.4, where columns refer to the ranks of the individual ranking methods.

$$Kendall's\ W = \frac{Variance\ of\ column\ totals}{Maximum\ variance\ of\ column\ totals} \qquad 6.4$$

6.2.5 Evaluation of error models using Kullback-Leibler divergence

Measurements were converted to distributions by assuming that they represent the means of Gaussian distributions with a standard deviation estimated by the experimental error. For the datasets where repeated measurements were available; compounds with repeats were considered to be less reliable than compounds with a single measurement since repeats may either be the outcome of retesting due to a suspected systematic error, i.e., environmental or assay failure, or measurements that have accumulated from multiple projects over time. The opposite could also be argued however: that average estimates of two or more repeated measurements are more reliable than estimates from a single measurement. Yet, repeated measurements were only available for the ADME datasets, where in most datasets they represented a small fraction of the data with the exception of two datasets where repeats were available for all compounds (see Chapter 4). For compounds with repeats, the standard deviation of the measurement was substituted by the propagated error σ_i of the measurement, $\sigma_i = \sqrt{\sigma_{assay}^2 + \sigma_{meas,i}^2}$, where σ_{assay}^2 represents the assay variability and $\sigma_{meas,i}^2$ the variance of the individual measurement. This was done so that the sizes of the error estimates represent the reliability of the measurements in a consistent manner. The point predictions of the QSAR models were converted to distributions using the prediction error estimates obtained from error models as estimates of their standard deviation.

The divergence between the measurement and prediction distributions of a compound was measured using the KLD score. This was calculated using Equation 6.5

$$D_{KL}(S_{p_i}, S_{q_i}) = \left[\frac{(\mu_{pi} - \mu_{qi})^2}{2{\sigma_{qi}}^2} + \frac{{\sigma_{pi}}^2}{2{\sigma_{qi}}^2} + \ln\frac{\sigma_{qi}}{\sigma_{pi}}\right] - \frac{1}{2} \qquad 6.5$$

where μ_{pi} is the measurement value with a standard error of σ_{pi} i.e., the measurement error and $\mu_{qi}$is the predicted value obtained from the QSAR model with a prediction error estimate of σ_{qi} . The value of D_{KL} is unbounded, non-negative and equal to zero only when the two distributions fully overlap; therefore, the smaller the value of the KLD score the higher is the overlap of the two distributions. The order of the values increases rapidly when the distance of the means becomes increasingly larger than the standard deviation of the prediction distribution, but also when the magnitude of the standard deviations of the distributions varies. The calculation of the KLD metric and a demonstration of its behaviour for the relative differences between the residual error, the prediction error estimate and the measurement error estimate are provided in the Appendix (B 1 and B 2).

The mean KLD of the N measurement and prediction distribution pairs of a dataset is calculated by averaging the KLD of the compounds (Equation 6.6).

$$KL_{AVE} = \frac{1}{N}\sum_{i=1}^{N} D_{KL} \qquad 6.6$$

A mean KLD value that is close to zero indicates that there is high overlap between the model's prediction distributions and measurement distributions. Error models that produce error estimates of the same order as the experimental assay and the true prediction errors, on average, will have lower KLD scores.

The individual KLD scores were calculated by substituting the standard deviation of the prediction distributions with error estimates from the AD-based error models, descriptor-based error models and the standard deviation of the ensemble predictions. Standard deviation estimates were only available for the ensemble models for the KNN, SVM and RF algorithms in the case of LogD.

The KLD scores using a uniform estimate based on the cross-validation RMSE of the QSAR models were used as a baseline for assessing the average performance of the error models. The experimental error estimates, i.e., MSD, and the uniform prediction error estimates that were used for the calculation of the KLD scores of the uniform, baseline error estimates are provided in Table 6-1.

Table 6-1. Comparison of the assay variability estimate and the models' cross-validation error used to assign uniform uncertainty estimates to the measurement and the prediction distributions, respectively.

Dataset	**MSD**	**CV RMSE**	
		RF	**SVM**
1	0.133	0.170	0.166
2	0.087	0.094	0.096
3	0.061	0.151	0.140
4	0.070	0.151	0.149
5	0.044	0.104	0.104
6	0.069	0.132	0.125
7	0.218	0.221	0.221
8	0.184	0.260	0.260
9	0.133	0.220	0.218
10	0.182	0.273	0.265

6.3 Results

6.3.1 Overall performance of error models

The overall performance of the error models was evaluated with respect to the accuracy of the error estimates and their correlation to the actual prediction errors and compared to the predictive performance of the underlying LogD and ADME models that were built in the previous chapter. The boxplots in the left of Figure 6-1, represent a total of 96 estimates of the descriptor-based and AD-based RF error models' predictive performance for the two underlying ADME models and the four underlying LogD models on cross-validation and holdout data. The error models perform poorly and are not predictive, with an average squared correlation coefficient that is close to zero. The negative R^2 values indicate that some models are unsuitable for use as they are unable to explain the variability of the errors and, thus, produce random output (Kvalseth, 1985). This is more prominent when AD-based features are used to train the error models, although the large variation in the R^2 values suggests that there may be cases where the error estimates may be useful.

Normalised RMSE values were also used to facilitate the comparison of accuracy obtained across different models and datasets; as shown in the right of Figure 6-1. The normalised RMSE of each QSAR model was calculated by dividing the model's RMSE with the range of observed values; while the normalised RMSE of each error model was obtained by dividing the error model's RMSE with the range of the true residual errors. The normalised RMSE was calculated for both the cross-validation and holdout data. On average, the accuracy of descriptor-based error models is similar to the accuracy of the QSAR models; but marginally more accurate than the AD-based error models.

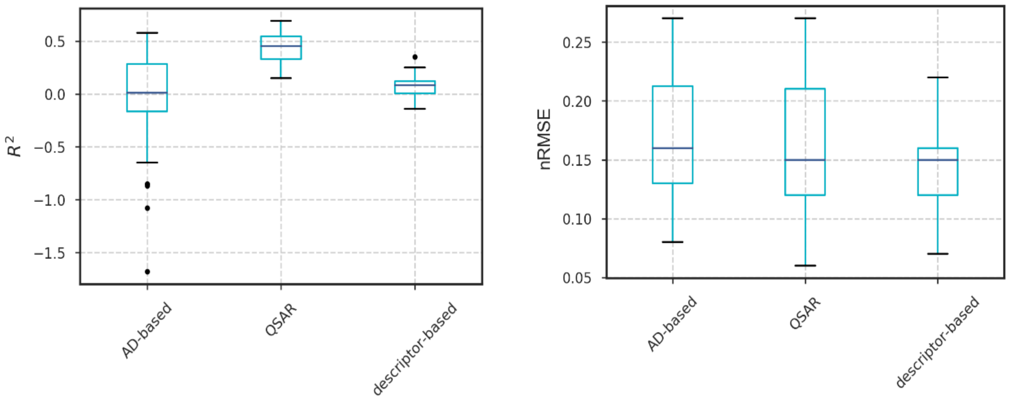


Figure 6-1. Average performance of the error models relative to performance of the underlying QSAR models measured by the R2 (left) and the normalised RMSE (right)

6.3.2 Performance of error estimates for LogD models

6.3.2.1 Binned D2M-based model

The strength of the underlying assumption that prediction error increases with increasing distance from the model's descriptor-based AD is first evaluated for the PLS, KNN, RF and SVM LogD underlying models.

The cross-validation errors of the underlying RF model are plotted against the D2M indices of the training data on the left of Figure 6-2. The red line illustrates the linear relationship between the mean D2M and the mean errors of the binned data using 10 bins of equal frequency. The black line shows the same relationship following the removal of 26 statistical D2M outliers in the D2M range of 11 – 40 (not shown), i.e., data points with a D2M index greater than 3 times the standard deviation of the D2M indices of all compounds. On the right, the same lines are plotted against the linear regression of the median distances and median errors of the bins indicated by the hashed line, with and without the D2M outliers. The regression of the medians resulted in a better fit than the linear regression of the means.

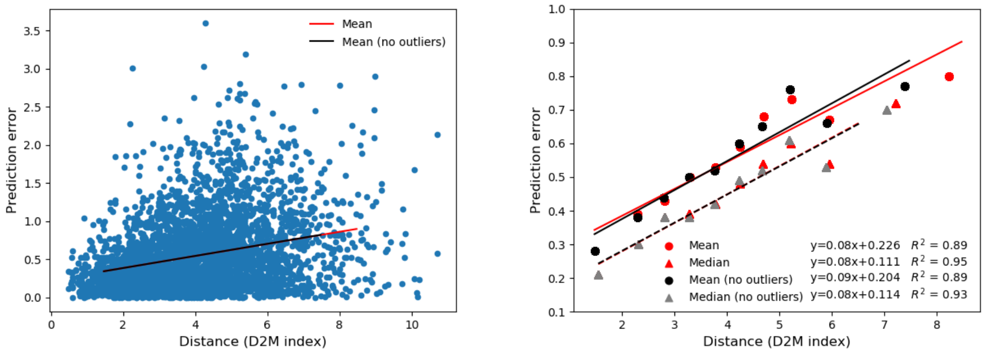


Figure 6-2. Linear regression line of the mean distances and errors plotted against the original D2M indices of the training data and the RF cross-validation errors following the removal of the 26 D2M outliers (left). Linear regression of the mean (red circle: outliers included; black circle: outliers excluded) and median estimates (red triangle: outliers included; grey triangle: outliers excluded) of the D2M bins (right).

The prediction error estimates obtained by averaging the cross-validated errors of the four LogD QSAR models in each D2M bin are provided in Table 6-2, without the removal of outliers.

Table 6-2. Mean distances (D2M) and absolute error (AE) of bins used in the binned D2M-based error model for KNN, PLS, SVM and RF predictions (bins=10, bin size=357) without the removal of outliers.

Bin	D2M		KNN AE		PLS AE		SVM AE		RF AE	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD
1	1.51	0.42	0.36	0.31	0.71	0.53	0.27	0.23	0.34	0.26
2	2.36	0.16	0.49	0.42	0.80	0.63	0.35	0.36	0.46	0.38
3	2.88	0.14	0.51	0.42	0.84	0.59	0.38	0.31	0.50	0.37
4	3.39	0.15	0.60	0.50	0.82	0.58	0.46	0.41	0.58	0.45
5	3.88	0.13	0.62	0.54	0.79	0.61	0.46	0.40	0.58	0.45
6	4.37	0.14	0.68	0.56	0.84	0.63	0.53	0.44	0.64	0.52
7	4.84	0.14	0.79	0.62	0.88	0.62	0.61	0.50	0.67	0.53
8	5.38	0.15	0.83	0.62	0.90	0.63	0.65	0.52	0.76	0.56
9	6.13	0.28	0.81	0.62	0.80	0.58	0.65	0.55	0.69	0.54
10	8.55	3.82	0.91	0.75	0.96	0.72	0.77	0.64	0.84	0.62

The high variation in the residual errors of the bins, which is indicated by the increasingly large standard deviations, illustrates the uncertainty associated with the binned error estimates. Assuming a linear relationship would require that the variation of the errors remains constant across the bins; however, this is not the case as the standard deviation of the error is seen to increase. The increasing standard deviation of the binned error estimates also results in overlapping values across neighbouring bins. Higher variation in the binned estimates is observed for the binned D2M-models of the PLS and KNN errors.

Strong correlations between the mean D2M and prediction error means of the RF, SVM and KNN models were also obtained for the binned cross-validation data, when increasing the number of bins from 10 to 100 (not shown).

The prediction error estimates of the holdout data were assessed bin-wise; by evaluating the agreement of the bin estimate to the average true error of the predictions assigned to the bin. In Figure 6-3, the average error estimate is plotted against the average true error of each bin for all four algorithms. It is seen that the performance of the method deteriorates with increasing number of bins for each algorithm at different rates. This makes sense, as increasing the number of bins will result in fewer compounds in each bin and the average binned estimates will be less accurate. This is more obvious for bins with higher D2M and prediction error estimates, which have greater variance. The binned models worked best for the SVM algorithms for up to 20 bins, and also performed reasonably well for the KNN and RF algorithms using 10 bins.

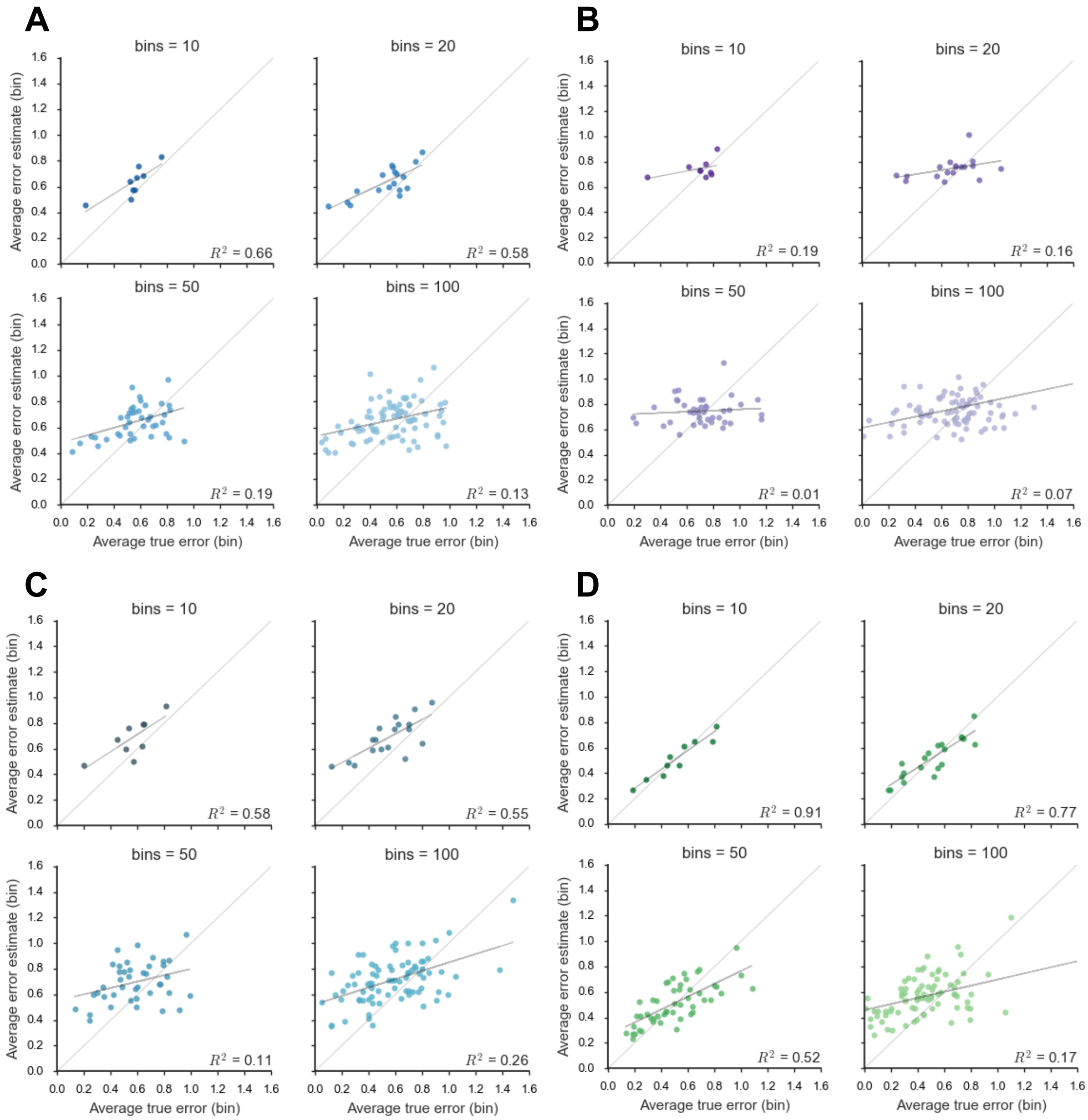


Figure 6-3. Effect of bin number on the predictive performance of linear D2M error models for A: KNN, B: PLS, C: RF,D: SVM holdout predictions.

The rank correlation of the binned error estimates to the errors of the individual predictions was also evaluated at different binning levels for both the cross-validation experiment and the holdout data (Table 6-3). The ranking ability of the binned models improves with increasing number of bins for the holdout data; albeit at a faster rate in the case of the PLS in relation to the KNN, RF and SVM models. As in the cross-validation data, the increase in the ranking correlation coefficient for the binned PLS models is likely attributed to the presence of large residual errors, which are greatly underestimated by binned models with a small number of bins. As the number of bins increases, the binned error estimates for predictions with larger D2M become more accurate, thus, improving the ranking ability of the binned model for predictions with large prediction errors yet reducing the overall linear correlation of the average binned error estimates with the average binned true errors.

Table 6-3. Rank correlation between the mean binned estimates assigned to the individual predictions and their absolute errors at different binning levels

Binning level			**10**	**20**	**50**	**100**
Spearman's ρ	**PLS**	**CV**	0.10	0.12	0.15	0.20
		Holdout	0.13	0.22	0.35	0.45
	KNN	**CV**	0.27	0.27	0.28	0.30
		Holdout	0.31	0.34	0.38	0.45
	SVM	**CV**	0.28	0.29	0.30	0.32
		Holdout	0.33	0.34	0.41	0.48
	RF	**CV**	0.32	0.32	0.33	0.35
		Holdout	0.31	0.32	0.36	0.45

Given that prediction errors in the binned D2M-model are described by the mean and standard deviation of the binned error estimates; the prediction error of each bin may be represented by a Gaussian distribution as suggested by Tetko et al. in (2008). This assumption may be used to assign individual prediction error estimates to the compounds of each bin that are drawn from a Gaussian distribution, $N\ (\mu, \sigma)$, where μ is the mean and σ is standard deviation of the error estimates in each bin.

6.3.2.2 Bagged ensembles

The performance of the standard deviations of the ensembles' predictions was used as a benchmark for the correlation of error estimates to the actual prediction errors. Listed in Table 6-4, below, are the calculated values of Pearson's and Spearman's rank correlation coefficients between the ensemble predictions' standard deviation and the actual prediction error calculated on out-of-bag data and holdout data for the KNN, SVM and RF ensembles.

Table 6-4. Linear and rank correlation between standard deviation estimates and absolute errors of bagged ensembles

Ensembles		Pearson's r		Spearman's ρ	
		Data sampling	Feature sampling	Data sampling	Feature sampling
KNN	OOB	0.60	0.03	0.52	0.02
	Holdout	0.31	-0.02	0.28	0.01
SVM	OOB	0.61	-0.03	0.58	0.01
	Holdout	0.28	0.05	0.33	0.06
RF	OOB	0.81	0.12	0.71	0.12
	Holdout	0.33	0.00	0.29	0.15

The results indicate that there is moderate to strong correlation between the RF, SVM and KNN ensemble estimates obtained by data sampling to the actual errors on OOB data, but weak correlation on the holdout test data. Feature resampling generally resulted in random correlations between the estimates and actual prediction errors, indicating that the estimates are not useful for ranking purposes.

6.3.2.3 Regression error models

The statistics of the descriptor-based and AD-based RFs' performance on the estimation of the QSARs' prediction errors are provided in Table 6-5. As indicated by the low R^2 correlation coefficients, the correlation of the error estimates to the prediction errors is very poor or random. The RMSE values of the error models suggest that the descriptor-based RF error estimates are generally less varied than the AD-based RFs, and result in estimates with weak correlations to the prediction errors of the QSAR models. An increase in correlation of the PLS error estimates of the descriptor-based error model and the prediction errors is observed.

Table 6-5. Predictive performance of the descriptor-based and AD-based RF error model on 10-fold CV and holdout data of the four LogD QSAR models

Underlying Model	Descriptor-based RF				AD-based RF			
	CV		Holdout		CV		Holdout	
	R^2	RMSE	Q^2	RMSE	R^2	RMSE	Q^2	RMSE
SVM	0.17	0.447	0.18	0.500	0.04	0.480	0.02	0.547
RF	0.14	0.455	0.12	0.512	0.05	0.479	0.01	0.542
PLS	0.28	0.498	0.32	0.527	0.04	0.575	0.09	0.609
KNN	0.14	0.528	0.07	0.572	0.07	0.546	0.01	0.593

Comparison of the underlying LogD and the RF error models' normalised RMSEs calculated on cross-validation and holdout data in Table 6-6 suggest that error models are less accurate relative to their respective underlying models. The only exception is the RF error model for PLS, which underestimates the residual error outliers of the underlying model.

Table 6-6. Accuracy of the underlying QSAR models relative to the accuracy of the RF error models calculated as the normalised RMSE

Underlying Model	**QSAR**		**Descriptor-based RF**		**AD-based RF**	
	CV	**Holdout**	**CV**	**Holdout**	**CV**	**Holdout**
SVM	0.099	0.127	0.123	0.128	0.132	0.139
RF	0.056	0.131	0.135	0.148	0.142	0.157
PLS	0.172	0.174	0.088	0.137	0.102	0.158
KNN	0.116	0.146	0.121	0.156	0.125	0.162

Table 6-7 shows the rank correlation coefficients between the estimates of the error models and the actual prediction errors of the four LogD models where it is seen that descriptor-based error models yield error estimates with higher correlation to the residual errors of the underlying models compared to the AD-based error models. Stronger correlation is obtained for the estimates of the PLS errors in the case of descriptor-based error models; while in the case of AD-based error models higher correlation, on average, is seen for the KNN errors.

On holdout data, the rank correlation of the descriptor-based error estimates was stronger than the ensemble based error estimates to the actual errors. However, the AD-based error estimates were more weakly correlated than the ensemble estimates in the case of the SVM and RF models.

Table 6-7. Rank correlation coefficients between the estimates of the descriptor-based and AD-based RF error models and the actual prediction errors of the four LogD models

Underlying Model		**Pearson's r**		**Spearman's ρ**	
		Descriptor-based RF	**AD-based RF**	**Descriptor-based RF**	**AD-based RF**
PLS	**CV**	0.54	0.25	0.49	0.21
	Holdout	0.57	0.29	0.50	0.30
KNN	**CV**	0.37	0.30	0.35	0.31
	Holdout	0.29	0.28	0.29	0.28
SVM	**CV**	0.41	0.25	0.38	0.25
	Holdout	0.44	0.28	0.43	0.30
RF	**CV**	0.38	0.27	0.38	0.25
	Holdout	0.34	0.24	0.36	0.24

The ability of the error models to identify poorly predicted compounds was tested by applying thresholds to the error estimates of the holdout data and calculating the accuracy of the predictions above and below the thresholds. Table 6-8 shows that the error models are not very efficient in identifying large prediction errors. Specifically, using descriptor-based error estimates to filter the predictions of all underlying algorithms resulted in the removal of predictions with higher accuracy and increased the error of the remaining holdout data except in the case of KNN predictions. In the case of the KNN LogD model, improvement in the accuracy of the holdout data was trivial after removing 20% of the predictions with the largest error estimates. The AD-based error estimates were more effective for filtering out poor predictions in the case of the KNN and SVM models but less effective for RF. Nevertheless, using the error estimates of ensemble methods resulted in the highest accuracy of holdout predictions of all four models.

Table 6-8. Accuracy of the LogD holdout test set predictions above and below the thresholds of the 80th and the 90th percentiles on the error estimates calculated as the mean absolute error (MAE)

Underlying Model	Error model	MAE (All predictions)	Threshold = p80		Threshold = p90	
			Below	Above	Below	Above
PLS	**AD-based**	0.786	*0.806*	*0.709*	*0.799*	*0.673*
	Desc-based	0.786	*0.807*	*0.703*	*0.799*	*0.676*
KNN	**AD-based**	0.620	0.607	0.673	0.601	0.796
	Desc-based	0.620	0.619	0.625	0.617	0.647
	Ensemble	0.620	0.555	0.881	0.597	0.830
SVM	**AD-based**	0.556	0.546	0.594	0.547	0.635
	Desc-based	0.556	*0.573*	*0.487*	*0.562*	*0.505*
	Ensemble	0.556	0.514	0.721	0.527	0.809
RF	**AD-based**	0.592	0.590	0.596	0.600	0.516
	Desc-based	0.592	*0.615*	*0.498*	*0.598*	*0.532*
	Ensemble	0.592	0.546	0.773	0.559	0.880
	SD	0.592	*0.602*	*0.577*	*0.610*	*0.484*

Analysis of the correlations above suggested that the descriptor-based error estimates were better at ranking predictions based on their actual prediction errors than the AD-based error models or the ensemble error estimates, however, the opposite was observed here. In addition, similar rank correlation coefficients were obtained of the ensemble estimates and AD-based estimates to the actual prediction errors but there is a clear difference between the improvement of accuracy using ensemble and AD-based estimates to remove prediction error outliers. The results indicate that moderate correlation between the error estimates and the prediction errors may not be suggestive of the error model's ability to detect large prediction errors.

6.3.2.4 Evaluation using Kullback-Leibler divergence

The quartiles of the KLD distributions obtained for the predictions of PLS, KNN, SVM and RF LogD models, whereby their uncertainty is estimated using error models, are provided in Table 6-9. The KLD distributions are illustrated in the Appendix (A 9), with a KLD cut-off value of 20. For each underlying model, the differences observed in KLD scores are attributed to the estimates of the error estimation methods, as the experimental error and the residual errors remain constant. The minimum value of the KLD distributions for the uniform estimates is greater than 1 due to the difference between the experimental error estimate and the cross-validation RMSE estimate. The fold-difference of the estimates ranges between 7 and 9; with an experimental error estimate of 0.1 and the cross-validation RMSE estimates between 0.753 and 0.962 (see Chapter 4). The KLD distributions of the binned D2M estimates are shifted to larger values, as the error estimates assigned to individual predictions are average binned estimates. Error models with a median KLD score smaller than the baseline of the respective underlying models indicate that at least half of the error model estimates are more informative than the uniform error estimates and are shown in bold in Table 6-9.

Table 6-9. Quartiles and mean of the KLD distributions calculated using the uniform and variable estimates from the ensembles, AD-based and descriptor-based error models. The scores calculated using the error estimates from a single RF are provided in parentheses.

Error model	**Underlying model**	**Min**	**1st quartile**	**Median**	**3rd quartile**	**Max**	**Mean**
Uniform	**KNN**	1.64	1.66	1.77	2.19	10.98	2.16
	PLS	1.77	1.82	2.00	2.44	9.77	2.32
	RF	1.54	1.58	1.71	2.13	11.75	2.09
	SVM	1.53	1.55	1.66	2.04	15.10	2.07
Binned D2M (10 bins)	**KNN**	2.63	3.70	4.08	4.72	23.36	4.51
	PLS	3.42	3.89	4.18	4.79	15.31	4.59
	RF	2.39	3.62	3.93	4.57	19.76	4.37
	SVM	2.63	3.70	4.08	4.72	23.36	4.51
AD-based	**KNN**	0.06	1.05	**1.58**	3.79	86.87	4.16
	PLS	0.24	1.23	2.36	5.55	72.97	5.23
	RF	0.16	1.04	1.73	4.11	124.37	4.79
	SVM	0.23	0.98	**1.49**	3.90	335.48	4.78
Descriptor-based	**KNN**	0.28	1.10	**1.55**	3.44	64.25	3.75
	PLS	0.26	1.32	2.26	5.38	85.89	4.78
	RF	0.32	1.07	**1.64**	3.72	65.36	3.91
	SVM	0.34	1.00	**1.49**	3.18	144.09	3.89
Ensemble	**KNN**	0.00	1.03	1.79	4.55	293.78	4.43
	RF	0.00	1.16	4.65	15.53	440.15	16.17
	RF	(0.66)	(1.51)	(1.76)	(2.23)	(18.43)	(2.28)
	SVM	0.01	0.83	2.40	6.74	217.26	7.45

The mean KLD scores for the variable error estimates were all larger than the mean KLD score of uniform error estimates and indicated that there is no improvement in their use. However, this is because there are predictions with large residuals and small prediction error estimates, which result in very large KLD scores. Interestingly, the mean KLD scores suggest that the binned D2M estimates for the RF and SVM models are more informative than the estimates of the AD-error models and the ensembles. Similarly, the binned D2M estimates for the PLS model are more informative than the estimates obtained from the descriptor-based and AD-based error models. Based on the median KLD scores, error estimates of the descriptor-based and AD-based models indicate that they result in higher overlap between the measurement and prediction distributions than the ensemble estimates in at least half of the data. For example, the maximum KLD of 293.78 for the ensemble-based error estimate for KNN predictions is attributed to a residual error of 2.45 and a prediction error estimate of 0.101.

6.3.3 Performance of error estimates for ADME models

The prediction errors of the underlying RF and SVM ADME models were estimated using the standard deviation of their ensembles and the error estimates derived from the RF regression error models. The RF regression error models were trained using the descriptor-based and AD-based features described in the Methods section.

6.3.3.1 Bagged ensembles

Bagged ensembles derived from data sampling produced error estimates with moderate to strong correlation with the actual prediction errors of the out-of-bag (OOB) predictions on average. However, these were not representative of the correlations on holdout data. Table 6-10 shows the linear and ranked correlations between the standard deviation of the predictions that were derived from bagged RF and SVM ensembles and their actual prediction errors.

Table 6-10. Linear and rank correlation coefficients between the bagged error estimates from data sampling and actual errors of the ADME ensembles

Dataset			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Pearson's r	**SVM**	**OOB**	0.47	0.58	0.78	0.76	0.47	0.68	0.61	0.47	0.33	0.63
		Holdout	0.00	0.20	0.17	0.23	0.32	0.24	0.23	0.00	0.14	0.17
	RF	**OOB**	0.90	0.92	0.92	0.92	0.91	0.92	0.93	0.92	0.91	0.90
		Holdout	0.14	0.32	0.32	0.42	0.51	0.37	0.32	0.03	0.23	0.14
Spearman's ρ	**SVM**	**OOB**	0.35	0.07	0.40	0.32	0.14	0.28	0.57	0.54	0.26	0.55
		Holdout	0.09	0.08	0.23	0.26	0.20	0.25	0.29	0.00	0.15	0.18
	RF	**OOB**	0.84	0.80	0.86	0.85	0.79	0.86	0.86	0.89	0.85	0.87
		Holdout	0.15	0.08	0.35	0.49	0.55	0.34	0.32	0.05	0.26	0.17

RF ensembles yielded error estimates with strong linear and rank correlations to their residual errors for OOB data. However, there are large differences in the linear and rank correlations for individual datasets which are likely due to the assumptions of Pearson's r not being met. Pearson's r assumes that the error estimates and the actual errors are normally distributed. However, inspection of the histograms in Chapter 5 suggests that the error distributions are skewed due to the presence of residual error outliers. Therefore, subsequent discussion of the results focuses on analysis of the rank correlations. However, for holdout test data, weak and moderate rank correlation of the error estimates to the prediction errors were obtained for four (datasets 3, 6, 7 and 9) and two (datasets 4 and 5) datasets, respectively. Weak to moderate rank correlation was also observed for the error estimates of SVM ensembles to the prediction errors of the OOB data. Weak correlations were also obtained for the estimates for the prediction errors of holdout data. The weakest rank correlations were observed for the error estimates of bagged ensembles for the holdout data of datasets 2 and 8, which were also the ADME models with the poorest overall performance.

In Table 6-11, the bagged ensembles from feature sampling have produced random estimates with very weak or no correlation to the actual errors of OOB and holdout data. Weak correlations obtained for the holdout test data between the estimates of both SVM and RF ensembles and their respective prediction errors for dataset 5 and the RF ensemble estimates for dataset 4 are likely the result of the skewed error distributions of these models (see Figure A 4 in the Appendix).

Table 6-11. Linear and rank correlation coefficients between the bagged error estimates from feature sampling and actual errors of the ADME ensembles

Dataset			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Pearson's r	**SVM**	**OOB**	0.07	0.03	-0.12	-0.13	0.19	-0.14	-0.10	0.03	0.00	-0.01
		Holdout	-0.08	-0.03	-0.16	-0.10	0.30	-0.19	-0.02	-0.03	0.01	-0.01
	RF	**OOB**	0.00	0.12	0.09	0.09	0.14	0.02	0.04	0.00	-0.03	0.02
		Holdout	-0.05	0.11	0.07	0.17	0.28	-0.05	-0.03	0.02	-0.02	-0.03
Spearman's ρ	**SVM**	**OOB**	0.06	0.05	-0.05	-0.11	0.17	-0.06	-0.09	0.01	0.00	-0.01
		Holdout	-0.06	-0.03	-0.03	-0.09	0.20	-0.09	0.00	-0.01	-0.01	-0.02
	RF	**OOB**	-0.01	0.11	0.09	0.13	0.13	0.01	0.05	0.01	-0.02	0.01
		Holdout	-0.11	0.01	0.02	0.24	0.29	-0.04	0.00	0.01	-0.03	-0.02

6.3.3.2 Regression error models

In section 6.3.1, the descriptor-based error models and the AD-based error models were found to have similar average performance for the underlying SVM algorithm; while AD-based error models were less accurate for the estimation of RF errors, on average. The cross-validated and holdout performance metrics of the error models on the ADME datasets are provided in

Table 6-12 and Table 6-13, for the descriptor-based and AD-based error models, respectively. As a general observation, it is seen that the performance of regression error models is very poor and accurate error estimates for ADME predictions cannot be obtained.

With regards to the descriptor-based error models; similar accuracy estimates are observed in

Table 6-12 for both the RF and SVM models and the estimates obtained for cross-validation and holdout data are in good agreement. The R^2 values for datasets 3, 4, 6, 7, 9 and 10 indicate that the error estimates are weakly correlated to the actual errors. Error estimates with the strongest correlation and highest accuracy to the actual errors of the underlying RF and SVM models are observed for dataset 5, which also yields the best performing ADME models. The AD-based error models resulted in higher R^2 values on cross-validation data; but only for underlying RF models. In comparison to the descriptor-based error models; the AD-based error models were less accurate for the underlying RF models. However, they had similar accuracy to the descriptor-based error models for the underlying SVM models.

Table 6-12. Predictive performance of the descriptor-based RF error model on the 10-fold CV errors of RF and SVM ADME models

Dataset	RF				SVM			
	CV		Holdout		CV		Holdout	
	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
1	-0.03	0.161	-0.05	0.153	-0.08	0.159	0.08	0.163
2	-0.06	0.077	0.10	0.120	-0.07	0.074	0.09	0.119
3	0.10	0.139	0.08	0.125	0.08	0.147	-0.01	0.116
4	0.18	0.148	0.16	0.189	0.09	0.119	0.10	0.162
5	0.35	0.078	0.24	0.115	0.23	0.076	0.25	0.121
6	0.09	0.153	0.10	0.147	0.01	0.152	0.08	0.133
7	0.16	0.183	0.07	0.181	0.12	0.202	0.12	0.193
8	-0.07	0.200	-0.05	0.214	-0.14	0.209	-0.04	0.216
9	0.11	0.163	0.07	0.165	0.08	0.083	0.16	0.145
10	0.13	0.161	0.05	0.159	0.08	0.145	0.03	0.162

Table 6-13. Predictive performance of RF error model trained using AD-based descriptors on the 10-fold CV errors of RF and SVM ADME models

Dataset	RF				SVM			
	CV		Holdout		CV		Holdout	
	R^2	RMSE	R^2	RMSE	R^2	RMSE	R^2	RMSE
1	0.39	0.213	-0.85	0.202	-0.09	0.155	-0.21	0.160
2	0.30	0.086	-0.15	0.134	-0.09	0.081	-0.12	0.131
3	0.39	0.184	-0.60	0.165	0.06	0.150	-0.07	0.119
4	0.58	0.166	-0.06	0.211	0.08	0.119	0.11	0.162
5	0.53	0.129	-1.08	0.191	0.28	0.083	0.09	0.131
6	0.37	0.264	-1.68	0.253	-0.01	0.170	-0.14	0.148
7	0.55	0.243	-0.46	0.241	0.04	0.213	-0.01	0.203
8	0.45	0.251	-0.85	0.268	-0.22	0.224	-0.14	0.232
9	0.37	0.217	-0.65	0.220	0.03	0.087	0.08	0.151
10	0.41	0.225	-0.87	0.222	0.09	0.129	0.09	0.144

In Figure 6-4 and Figure 6-5. Comparison of the error model and the underlying SVM model RMSE values on cross-validation (left) and holdout data (right) , the accuracy estimates of the error models are plotted against the accuracies of the respective, underlying models on cross-validation and holdout data in units of RMSE. These results are also provided in tabulated format in the Appendix (A 5 and A 6). Figure 6-4, shows that, in general, ADME models with higher accuracy result in more accurate error models. On cross-validation data, the descriptor-based error models were more accurate than the underlying RF models for most datasets; particularly when the underlying models had larger errors (7, 8, 9, 10). The AD-based error models were less accurate than the respective ADME models on, both, cross-validation and holdout data.

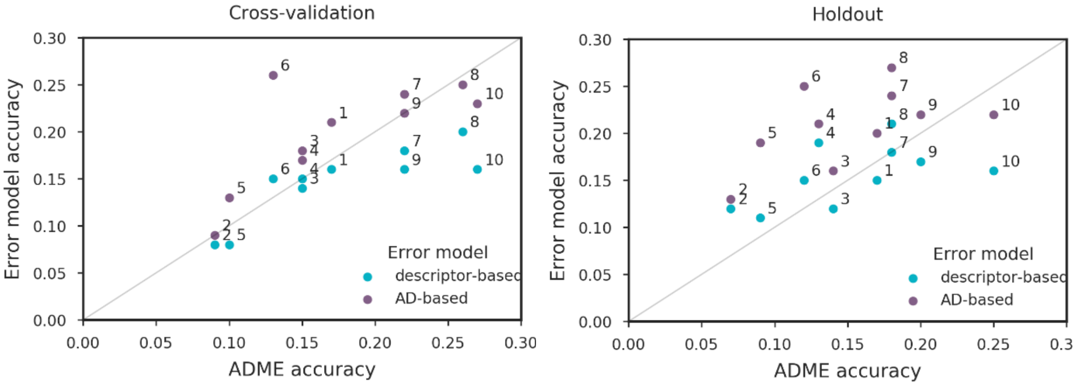


Figure 6-4. Comparison of the error model and underlying RF model RMSE values on cross-validation (left) and holdout data (right)

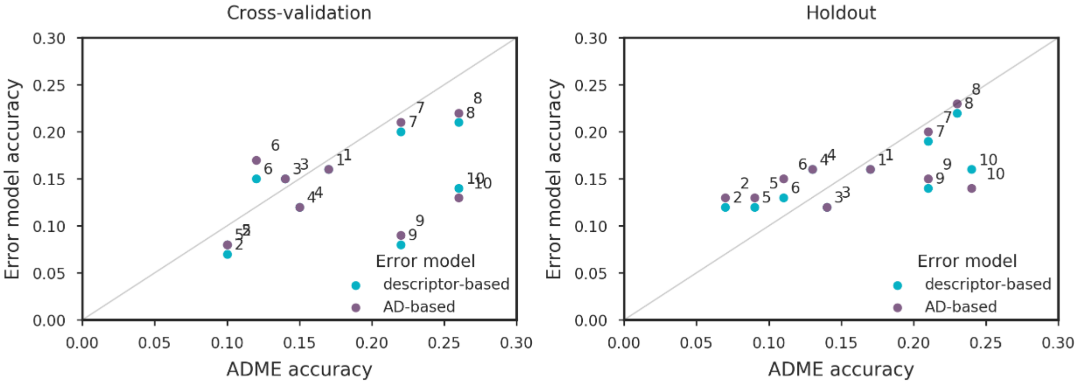


Figure 6-5. Comparison of the error model and the underlying SVM model RMSE values on cross-validation (left) and holdout data (right)

More accurate error models were obtained when the underlying algorithm was an SVM. The accuracy of the RF error models is higher than their respective ADME models, particularly when the latter have larger errors. In Figure 6-5. Comparison of the error model and the underlying SVM model RMSE values on cross-validation (left) and holdout data (right) , the average accuracy estimates of both error models are in good agreement on cross-validation and holdout data; and overlapping estimates are obtained for three datasets (1, 3 and 4).

Despite the poor predictive performance of the error models; the estimates obtained could still be useful for ranking predictions on their actual prediction errors. In Table 6-14 and Table 6-15 the rank correlations of the error model estimates to the actual prediction errors for the ADME models indicate the presence of weak to moderate correlations.

Although the AD-based error models are less accurate for the estimation of RF errors for the LogD data set in section 6.3.1, it is seen that their error estimates for RF predictions are more strongly correlated than the estimates from the descriptor-based error models even when the underlying models are poor. However, this is likely attributed to the standard deviation of the RF predictions being included as a variable in the AD-based models; which is known to correlate with the residual errors of RF predictions (Sheridan, 2012; Tetko et al., 2008). This is confirmed by the results in Table 6-16, where it is seen to exhibit very similar ranking performance to the AD-based model. In addition, the results obtained from cross-validation are similar to the results on holdout data when the underlying models have good predictive performance; such as in the case of the SVM models of datasets 3, 4, 5, 6, 7 and the RF models of datasets 1, 3, 4, 5, 6, 7, 9.

Table 6-14. Linear and rank correlation coefficients between the estimates of the descriptor-based RF error model and the actual prediction errors of RF and SVM ADME models

Dataset			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Pearson's r	**SVM**	**CV**	0.01	0.07	0.36	0.34	0.49	0.26	0.05	0.39	0.29	0.30
		Holdout	0.29	0.38	0.27	0.34	0.51	0.29	0.05	0.37	0.37	0.19
	RF	**CV**	0.13	0.12	0.37	0.44	0.60	0.33	0.15	0.44	0.34	0.37
		Holdout	0.22	0.33	0.35	0.41	0.52	0.32	0.05	0.30	0.34	0.24
Spearman's ρ	**SVM**	**CV**	0.00	0.09	0.32	0.38	0.44	0.24	0.41	0.06	0.26	0.30
		Holdout	0.26	0.27	0.37	0.38	0.47	0.27	0.38	0.10	0.40	0.19
	RF	**CV**	0.16	0.15	0.37	0.50	0.57	0.31	0.48	0.13	0.32	0.38
		Holdout	0.19	0.27	0.43	0.47	0.55	0.31	0.37	0.07	0.35	0.24

Table 6-15. Linear and rank correlation coefficients between the estimates of the using AD-based RF error model and the actual prediction errors of RF and SVM ADME models

Dataset			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Pearson's r	**SVM**	**CV**	0.07	0.02	0.37	0.35	0.54	0.26	-0.05	0.31	0.25	0.34
		Holdout	0.17	0.11	0.27	0.40	0.48	0.18	0.02	0.15	0.29	0.32
	RF	**CV**	0.64	0.61	0.66	0.77	0.74	0.64	0.71	0.76	0.62	0.65
		Holdout	0.34	0.40	0.29	0.44	0.48	0.35	0.05	0.24	0.43	0.30
Spearman's ρ	**SVM**	**CV**	0.09	0.00	0.35	0.43	0.55	0.25	0.37	-0.03	0.25	0.38
		Holdout	0.19	0.13	0.28	0.48	0.48	0.18	0.22	0.01	0.31	0.37
	RF	**CV**	0.51	0.41	0.55	0.64	0.63	0.50	0.62	0.56	0.53	0.55
		Holdout	0.31	0.25	0.37	0.50	0.56	0.37	0.31	0.08	0.45	0.32

Table 6-16. Linear and rank correlation coefficients between the standard deviation of the RF predictions and their actual prediction errors

Dataset			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Pearson's r	**RF**	**CV**	0.61	0.55	0.65	0.70	0.73	0.60	0.68	0.62	0.56	0.58
		Holdout	0.32	0.33	0.33	0.48	0.55	0.36	0.29	0.07	0.46	0.33
Spearman's ρ	**RF**	**CV**	0.55	0.45	0.59	0.65	0.64	0.51	0.62	0.54	0.53	0.55
		Holdout	0.34	0.18	0.38	0.54	0.59	0.36	0.36	0.10	0.46	0.34

The diversity of each dataset was computed as the variance of its pairwise distance matrix using the Euclidean distance metric for the cross-validation and holdout data. In Figure 6-6 and Figure 6-7, the rank correlation coefficients of the error models' estimates are plotted against dataset diversity for the underlying RF and SVM models respectively. It can be seen that the ability of error models to rank predictions is better in datasets that are more diverse. The linear trend is more prominent on holdout data than cross-validation data. In Figure 6-6, the AD-based error estimates have higher rank correlation to the actual errors of RF ADME models on cross-validation data than on holdout data regardless of the diversity in the data.

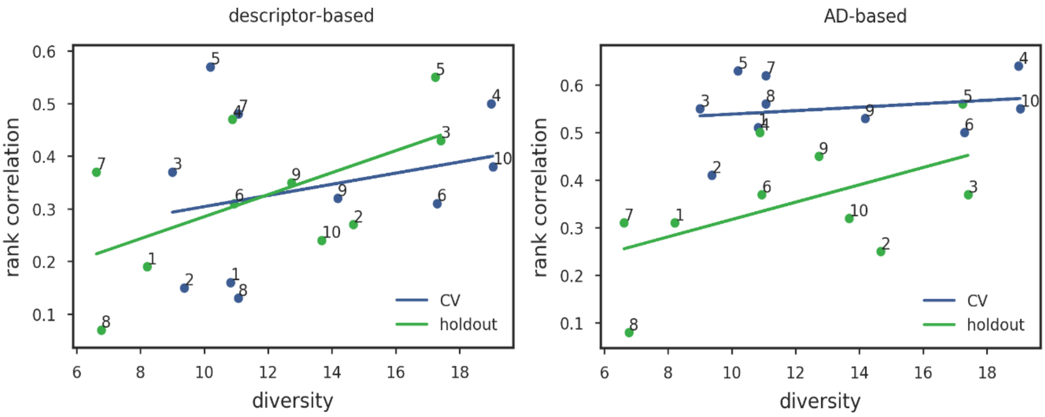


Figure 6-6. Rank correlation of the descriptor-based (left) and the AD-based (right) error models for the underlying RF ADME model plotted against the diversity of the datasets subsets

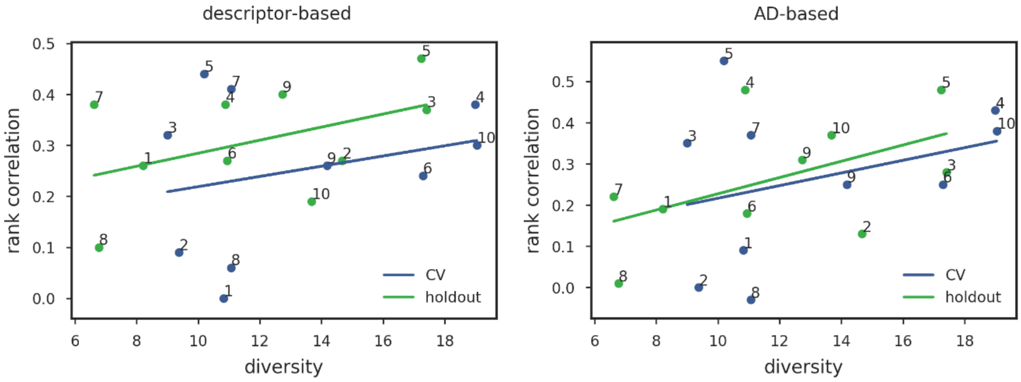


Figure 6-7. Rank correlation of the descriptor-based (left) and the AD-based (right) error models for the underlying SVM ADME model plotted against the diversity of the datasets subsets

As seen in Figure 6-8, the average performance of the error models in ranking the RF ADME predictions based on their actual errors is poorer than the performance of the standard deviation of the RF ensembles predictions. However, the error models yield similar ranking performance to the ensemble estimates in the case of the underlying SVM models' predictions.

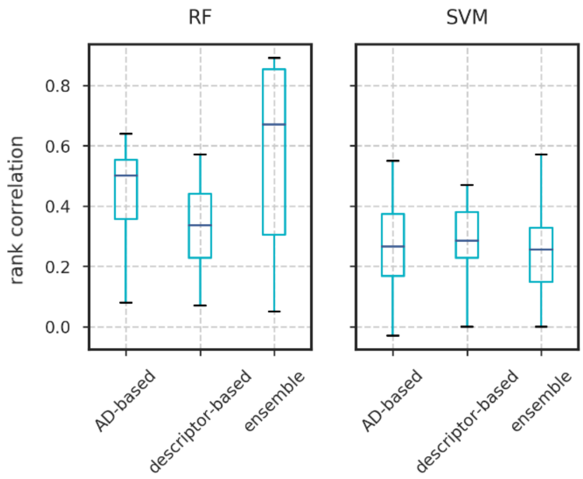


Figure 6-8. Rank correlations of error model and ensemble estimates

The ranking agreement between the AD-based and descriptor-based error models and the ensemble estimates was assessed in Table 6-17. The Kendall's W values calculated suggest that there is a strong correlation between the ranks assigned based on the error estimates, particularly in the case of the RF predictions. However, these results were calculated on the full holdout test set and, thus, it cannot be assumed that all error estimation methods assign similar ranks to poor predictions.

Table 6-17. Kendall's W for the error estimates of the two RF error models and the error estimates of the ensembles for the predictions of the RF and SVM ADME models

Dataset			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Kendall's W	**SVM**	**CV**	0.50	0.38	0.65	0.58	0.64	0.64	0.74	0.41	0.54	0.62
		Holdout	0.57	0.52	0.67	0.63	0.68	0.59	0.72	0.48	0.59	0.55
	RF	**CV**	0.69	0.69	0.79	0.84	0.90	0.75	0.81	0.66	0.71	0.74
		Holdout	0.72	0.70	0.80	0.87	0.89	0.78	0.79	0.68	0.70	0.67

Previously, there was no apparent trend between the rank correlations and improvement in the accuracy of the LogD holdout predictions was observed after applying statistical thresholds to the error models' estimates. However, the opposite is observed for the holdout data of the ADME predictions in Figure 6-9. In the figure below, the difference between the accuracy of the error models for the full holdout test set and the filtered holdout data using the 80th and 90th percentile values of the error estimates rank correlation is plotted against the rank correlation between the error estimates and the ADME predictions for each holdout test set.

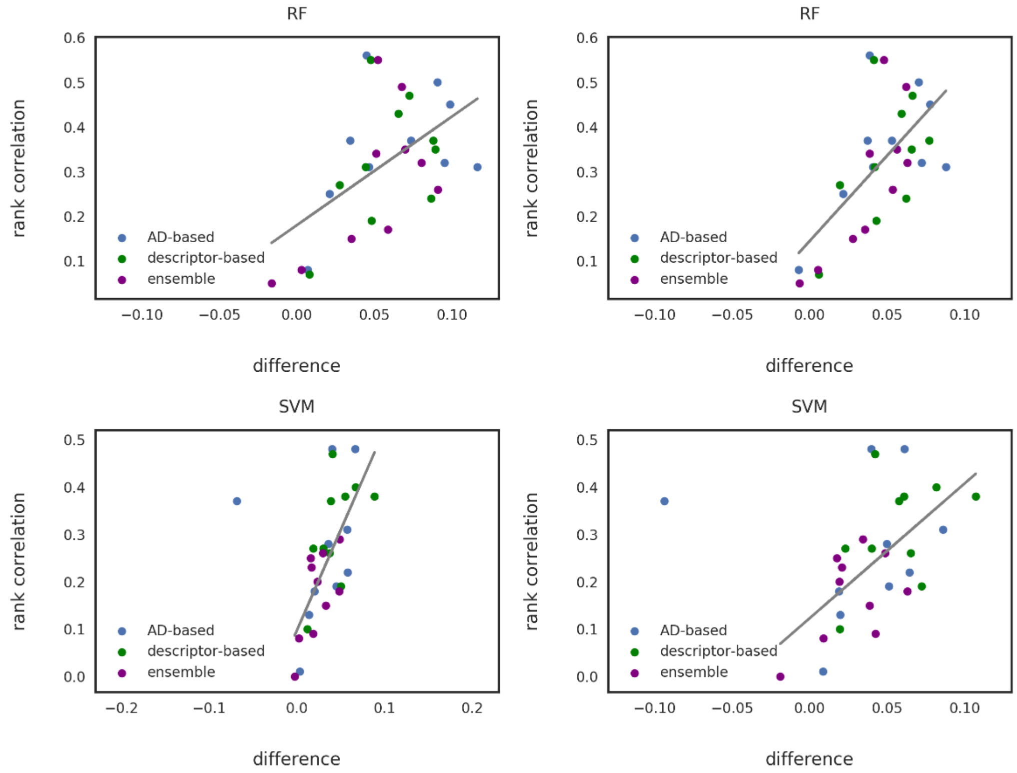


Figure 6-9. Trend showing the improvement in the accuracy of the holdout predictions after removal of the upper 20% (left) and 10% (right) error estimates

The improvement in the accuracy of the holdout test set following the application of the threshold values to the different types of error estimates is illustrated in detail for the RF and SVM ADME models in Table 6-18 and Table 6-19. The instances where applying the threshold reduced the accuracy of the holdout data are underlined and italicised.

It is seen that the removal of 20% of the holdout predictions based on their error estimates can result in double the accuracy of the error models on holdout data, such as in the case of datasets 3, 4, 5 and 9. The RF and SVM models of datasets 8 and 10 which had the largest model error did not benefit from filtering using AD-based error estimates or ensemble error estimates as it resulted in the removal of more accurate predictions, thus, increasing the error of the predictions below the threshold.

Interestingly, applying a higher threshold of p90 leading to the removal of 10% of the holdout data with the largest error estimates results in higher accuracy than the threshold of p80. In both cases, however, it is clear that the AD-based error models are more effective for identifying large prediction errors of RF models, while the descriptor-based error models are more effective in the case of SVM models. In contrast to the results of the LogD models, the ensemble-based error estimates are the least effective for filtering large prediction errors in the ADME RF and SVM models.

Table 6-18. Accuracy of the RF and SVM predictions for the ADME holdout data above and below the 80th percentile threshold value on the error estimates calculated as the mean absolute error (MAE)

Underlying model	**Error model**	**Threshold p80**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Mean Difference**
RF	AD-based	Holdout	0.111	0.053	0.112	0.126	0.070	0.101	0.205	0.217	0.154	0.224	0.049
		Below	0.070	0.032	0.059	0.056	0.031	0.063	0.117	*0.224*	0.077	0.152	
		Above	0.121	0.058	0.125	0.143	0.079	0.110	0.227	0.215	0.174	0.242	
	Desc-based	Holdout	0.111	0.053	0.110	0.126	0.070	0.101	0.205	0.217	0.151	0.224	0.048
		Below	0.068	0.034	0.051	0.060	0.028	0.059	0.128	0.211	0.086	0.162	
		Above	0.122	0.058	0.125	0.142	0.080	0.111	0.224	0.218	0.168	0.240	
	Ensemble	Holdout	0.112	0.056	0.115	0.129	0.075	0.105	0.211	0.217	0.160	0.229	0.040
		Below	0.085	0.051	0.059	0.067	0.027	0.067	0.148	*0.224*	0.107	0.193	
		Above	0.119	0.057	0.129	0.144	0.087	0.115	0.227	0.216	0.174	0.237	
SVM	AD-based	Holdout	0.111	0.060	0.099	0.126	0.077	0.093	0.205	0.213	0.154	0.342	0.027
		Below	0.066	0.046	0.063	0.060	0.037	0.073	0.147	0.210	0.097	*0.410*	
		Above	0.122	0.063	0.108	0.142	0.087	0.098	0.219	0.214	0.169	0.324	
	Desc-based	Holdout	0.111	0.060	0.099	0.126	0.077	0.093	0.205	0.213	0.154	0.211	0.044
		Below	0.074	0.042	0.061	0.071	0.037	0.063	0.116	0.201	0.088	0.161	
		Above	0.120	0.064	0.109	0.140	0.087	0.100	0.227	0.216	0.171	0.224	
	Ensemble	Holdout	0.112	0.060	0.103	0.125	0.077	0.097	0.205	0.213	0.155	0.213	0.022
		Below	0.093	0.058	0.086	0.096	0.053	0.081	0.157	*0.216*	0.122	0.165	
		Above	0.116	0.061	0.107	0.133	0.082	0.101	0.217	0.213	0.163	0.225	

Table 6-19. Accuracy of the RF and SVM predictions for the ADME holdout data above and below the 90th percentile threshold value on the error estimates calculated as the mean absolute error (MAE)

Underlying model	**Error model**	**Threshold p90**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**Mean Difference**
RF	**AD-based**	**Holdout**	0.111	0.053	0.112	0.126	0.070	0.101	0.205	0.217	0.154	0.224	0.063
		Below	0.065	0.032	0.038	0.035	0.025	0.066	0.089	0.210	0.056	0.129	
		Above	0.116	0.055	0.120	0.136	0.075	0.105	0.218	0.217	0.165	0.235	
	Desc-based	**Holdout**	0.111	0.053	0.110	0.126	0.070	0.101	0.205	0.217	0.151	0.224	0.058
		Below	0.063	0.026	0.045	0.053	0.022	0.057	0.117	0.209	0.062	0.138	
		Above	0.116	0.056	0.117	0.134	0.075	0.106	0.215	0.218	0.161	0.234	
	Ensemble	**Holdout**	0.112	0.056	0.115	0.129	0.075	0.105	0.211	0.217	0.160	0.229	0.049
		Below	0.077	0.053	0.045	0.061	0.023	0.054	0.131	*0.234*	0.070	0.170	
		Above	0.116	0.056	0.123	0.136	0.081	0.111	0.220	0.215	0.171	0.235	
SVM	**AD-based**	**Holdout**	0.111	0.060	0.099	0.126	0.077	0.093	0.205	0.213	0.154	0.342	0.031
		Below	0.060	0.040	0.049	0.065	0.037	0.074	0.141	0.204	0.069	*0.435*	
		Above	0.117	0.062	0.105	0.133	0.081	0.095	0.212	0.214	0.164	0.331	
	Desc-based	**Holdout**	0.111	0.060	0.099	0.126	0.077	0.093	0.205	0.213	0.154	0.211	0.057
		Below	0.046	0.037	0.042	0.065	0.035	0.053	0.098	0.193	0.073	0.139	
		Above	0.118	0.062	0.105	0.133	0.082	0.097	0.217	0.215	0.163	0.220	
	Ensemble	**Holdout**	0.112	0.060	0.103	0.125	0.077	0.097	0.205	0.213	0.155	0.213	0.028
		Below	0.069	0.052	0.082	0.077	0.058	0.080	0.171	*0.232*	0.116	0.150	
		Above	0.116	0.061	0.105	0.131	0.079	0.099	0.209	0.211	0.159	0.220	

6.3.3.3 Evaluation using Kullback-Leibler divergence

Table 6-20 shows the mean KLD scores of the RF ADME models for the different error estimation methods. The full KLD distributions of the RF and SVM ADME models obtained using the uniform and variable error estimates from error models are illustrated in the Appendix with a KLD cut-off value of 10 (Figures A 10 and A 11 in the Appendix). As in the results of the LogD models, the maximum KLD scores exceeded the order of two, as an effect of a large difference in the means of the distributions and prediction error estimates.

In Table 6-20 and Table 6-21, the KLD scores of the variable error estimates are generally more informative than the uniform error estimates. However, despite their larger KLD scores in comparison to the baseline it is seen that these are close to a value of one or lower, which suggests that there is some agreement in assumed measurement and prediction distributions. Lower KLD scores are observed for the AD-based and descriptor-based error model estimates of datasets 1 and 3, and the descriptor-based error model estimates of dataset 9. Although the standard deviation of the RF predictions performed less well than uniform estimates, it resulted in lower KLD scores than the other variable estimation methods in datasets 4, 5, 6 and 7. Conversely, the error estimates based on the standard deviation of the ensemble predictions resulted in high KLD scores that indicate a poor overlap of the measurement and prediction distributions. In Table 6-21, similar trends are observed in that the AD-based and descriptor-based error model estimates produced the lowest mean KLD scores for datasets 1, 3 and 9.

Table 6-20. Mean KLD of RF prediction distributions where prediction error is estimated by AD-based and descriptor-based error models, RF ensembles and the standard deviation of the prediction. The benchmark KLD score is calculated from uniform error estimates based on the model's average CV error.

Error model	**Dataset**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Uniform	1.10	0.69	2.21	1.03	0.93	0.74	0.72	0.81	1.21	0.60
AD-based RF	**0.75**	1.11	**1.16**	1.45	1.30	0.85	1.72	1.09	2.72	0.80
Desc-based RF	**0.75**	1.30	**1.07**	1.30	1.34	0.90	1.51	1.22	**1.04**	0.84
SD Ensemble	18.20	50.63	13.78	50.59	28.91	15.79	86.49	71.48	28.73	35.86
SD RF	2.44	2.24	4.20	1.15	1.17	0.76	1.41	1.35	21.83	0.80

Table 6-21. Mean KLD of SVM prediction distributions calculated for the estimates obtained from AD-based and descriptor-based error models and the standard deviation of SVM ensembles predictions

Error model	**Dataset**									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Uniform	1.18	0.58	2.50	0.98	0.95	0.70	0.80	0.83	1.32	0.60
AD-based RF	**0.85**	0.74	**1.00**	1.56	1.14	0.91	1.87	0.98	**1.19**	1.56
Desc-based RF	**0.79**	1.07	**1.05**	1.19	0.98	0.91	1.50	1.30	**0.86**	0.89
SD Ensemble	15.65	31.51	9.10	10.51	11.46	10.22	70.84	55.74	21.12	12.76

6.4 Discussion

The poor predictions of the models trained in Chapter 5 could not be sufficiently explained by defining their ADs; therefore, the use of error models that account for other variables that might be associated with the poor performance of the underlying models has been investigated. The use of a RF model to predict the errors of another RF QSAR model using similarity metrics as features was proposed by Sheridan in (Sheridan, 2013b). The AD-based RF error model utilised a selected set of features that were based on similarity and the output of the underlying RF model, thus, exploiting a special characteristic of the RF residual errors, which are correlated to the RF prediction and the prediction's variance.

In this work, the performance of RF error models was studied using descriptor-based features, i.e., the same features as the underlying model, and AD-based features, similar to those defined in the approach by Sheridan. The descriptor-based features included molecular and physicochemical descriptors, while the AD-based features consisted of the mean and standard deviation of the D2M in a local neighbourhood, which were combined with the prediction and the standard deviation of the prediction of the underlying RF model. More specifically, the performance of descriptor-based and AD-based RF error models was investigated in: 1) a well curated, LogD dataset with no residual error outliers to estimate the prediction errors of four machine learning algorithms, namely PLS, KNN, SVM and RF; and 2) ten, diverse, ADME datasets where residual error outliers were left untreated to estimate the prediction errors of RF and SVM algorithms.

The error models' performance has been compared to the benchmark performance of a binned-D2M model for the LogD data set, which assigns mean estimates to predictions assigned to the same bin; and the standard deviation of ensemble predictions obtained from data sampling and feature sampling for the LogD data set and the ADME data sets. The performance of error models was generally poor and had weak correlation to the residual errors of cross-validation data. The accuracy of the error models was higher or equivalent to the accuracy of the QSAR models, but low R^2 values indicated that the error estimates were very poorly correlated to the residual errors of the underlying models' predictions. The linear and ranking performance of the error estimates were assessed and yielded equivalent or poorer results compared to the performance of the benchmark estimates. Nevertheless, the estimates of the error models performed better in the task of identifying predictions of ADME RF and SVM models that had larger errors combined with statistical thresholds. In the LogD models, ensemble based error estimates were better in this task while the error model estimates were less effective.

These results were in agreement with the performance of other error models previously reported by Sheridan (Sheridan, 2013b). In that publication the error models were used to estimate the errors of LogD and other activity models that utilised atom pairs as descriptors instead of molecular descriptors. Furthermore, similarity instead of distance to model metrics were used to build the error models.

The descriptor-based error models performed better in ranking the residual errors of the PLS LogD model, which was the LogD model with the lowest predictive performance in Chapter 5. The average ranking performance of the error estimates for the RF and SVM LogD models, which were the best performing QSAR models, did not exceed the benchmark performance on cross-validation and holdout data. Ensemble based error estimates were better than descriptor-based error estimate in identifying poor predictions of the LogD models despite the stronger rank correlation of the latter with the actual prediction errors. Analysis based on the information-theoretic framework using KLD scores, suggested that descriptor-based estimates were actually the most informative and, that they may be useful for calculating confidence intervals that are more likely to include the measurement. Their KLD scores also indicated that, on average, the size of the prediction error estimates was comparable to the size of the actual prediction errors and the experimental error. The AD-based error models had poorer ranking performance for the LogD models but were found to be more effective in identifying large prediction errors than descriptor-based error models. However, the KLD scores for the KNN and the RF predictions indicated that their AD-based error estimates, on average, were more informative than the benchmark estimate based on uniform estimates.

The AD-based error models outperformed the ranking performance of the descriptor-based error models for underlying RF ADME models on cross-validation data. However, this is mainly due to the inclusion of the standard deviation of the underlying RF's prediction and the prediction itself as variables of the AD-based RF error model, which correlate well with the RF prediction errors. However, the former is only available for underlying RF models. The ranking performance of the error models was still poorer than the standard deviation of RF ensemble predictions, on average. Furthermore, there was no difference in the average ranking performance of the error models and the SVM ensemble error estimates. Improvement in the error models' ability to rank predictions was obtained with increasing dataset diversity. This could potentially indicate that the inclusion of residual error outliers in the error models improves their ability to distinguish predictions with large residual errors.

It was also found that the rank correlation was a useful indicator of the error estimates' ability to identify poor predictions made by the RF and SVM ADME models. Therefore, it was confirmed that AD-based error models performed best in identifying poor predictions of the RF ADME models and descriptor-based error models performed best in the case of SVM ADME models.

Analysis of the KLD scores made it clear that the RF error models yield more useful prediction error estimates than estimates based on the variation of individual predictions, as they represent direct estimates of the models' absolute residual error. While the variation of RF predictions is the best indicator of prediction error it may not always be useful as a direct estimate and may need to be adjusted by regression.

6.5 Conclusions

This chapter has investigated the performance of RF error models and other approaches for the estimation of errors in underlying QSAR models. The results suggest that error models may be of potential use in identifying novel predictions for skewed ADME datasets that are easy to model. However, it is clear that the current error models are not suitable for the direct estimation of ADME prediction errors and other methods or variables may need to be investigated. While further steps could potentially be added to the modelling process to obtain binned estimates from the error models' output it was considered that these would increase the uncertainty of the error models and become even less tractable. Furthermore, as seen in the linear binned D2M error models, binning is not suitable for the estimation of individual prediction error estimates. The results from KLD analysis suggested that the error estimates obtained may be useful for the calculation of interval estimates, if the estimates are larger and to some extent correlated with the actual prediction errors. The following chapter discusses an alternative approach for the estimation of uncertainty in individual prediction, namely conformal prediction, in the form of prediction intervals.

Chapter 7 Evaluation of Error Models using Conformal Prediction

7.1 Introduction

The CP framework facilitates the estimation of compound-specific prediction intervals that represent the uncertainty associated with the individual predictions. These are produced by normalising the models' errors with the output of any uncertainty estimation or reliability scoring method, thus, simplifying the comparison of uncertainty estimates and reliability indices to a direct comparison of prediction interval (PI) estimates. The best method among alternatives for the normalisation of the errors is the one that produces the narrowest PIs on average so that these may be informative for decision making. A study by Johansson, Boström, Löfström, & Linusson (2014) suggests that normalisation can significantly improve the efficiency of PIs particularly when the normalised PIs are strongly correlated with the actual prediction errors. Therefore, in spite of the low predictive performance of error models built in the previous chapter, their error estimates may be useful in the estimation of PIs with CP as long as they produce PIs that correlate well with the prediction errors.

The aim of this chapter is to investigate the utility of error models for the estimation of confidence in ADME regression models within the conformal prediction (CP) framework. The optimisation of the CP models' parameters is demonstrated, first, for the LogD dataset and is, then, followed for the ADME datasets. The estimates from different error models are used to generate normalised PIs for the underlying models built in Chapter 5, which are then assessed on their utility for the purpose of compound prioritisation.

7.2 Methods

Conformal prediction (CP) was applied in an aggregate conformal prediction (ACP) setting for the estimation of PIs of the LogD and ADME models built in Chapter 5. As discussed in Chapter 3, ACP infers a model's empirical error distribution from calibration data that are repeatedly sampled from the training set. The effect of the following three ACP parameters was studied on the PI estimates of the underlying models: a) the size of the ACP, i.e., the number of calibration sets repeatedly sampled, b) the sampling method and c) the method applied to normalise the error.

The effect of the ACP size and the sampling method were investigated on standard ACPs, which generate uniform PIs i.e., without normalisation. First, standard ACPs of size 10 were built to estimate uniform PIs for all compounds and assess how the PI size for each ADME model varies with respect to the level of confidence. An arbitrary threshold of 80% confidence was then chosen, which is common in CP applications within the chemoinformatics domain (Cortes-Ciriano & Bender, 2019; Norinder et al., 2016; Svensson et al., 2018) but also produced acceptable results for all models trained following confirmation. The same threshold was also applied in subsequent investigations.

The sizes of ACPs explored were 10, 100 and 1000. Larger ACPs were not studied as they were computationally expensive, particularly in the case of the larger datasets and also when applying normalisation. The sampling methods applied by the ACP to draw calibration data from the training set included bootstrap sampling, random sampling and cross-subsampling. Bootstrap and random sampling involve sampling N calibration set samples at random, each at 30% of the training set size with and without replacement, respectively. The implementation of cross-subsampling is equivalent to applying N-fold cross-validation, and involves randomly partitioning the training data into N folds with each fold representing a calibration sample draw. As a result, the size of the calibration set drawn by cross-subsampling is 10% of the training set for an ACP of size 10. The optimal size and sampling method identified were those that minimised the average PI estimates of the ACPs and these were then applied to build normalised ACPs.

Normalised ACPs were trained for the estimation of variable PIs. These were derived by normalising the models' errors using a distance-to-model (D2M) index and three error models. The distance-to-model index (D2M) was calculated for each instance as the average Euclidean distance of 5% of the nearest neighbours in the training set in descriptor space (Carrió et al., 2014). Two of the error models were built using RF and SVM algorithms and the molecular descriptors utilised by the ADME model. The third error model was an AD-based RF error model with three variables: the ADME prediction, the D2M index and the standard deviation of the distances to the 5% nearest neighbours in the training set. For the AD-based error model, the two distance-based variables introduced information regarding the density of the local neighbourhood: a small D2M and small standard deviation indicates a densely populated neighbourhood for a compound, while a large D2M indicates a sparser neighbourhood. Note, that the standard deviation of the RF ADME prediction was not included in these experiments.

The performance of the ACPs was evaluated at the set confidence level of 80% on holdout data using measures of validity and efficiency. Validity was assessed using the difference of the ACPs expected error rate that is associated with the set confidence threshold and the actual error rate on holdout data. Thus, for an 80% confidence threshold the expected error rate of the ACP is 0.2; which means that 20% of the PI estimates for the holdout data will be wrong. In other words, 20% of the PIs will not contain the true measurement. However, this error rate is only guaranteed if the CP assumption regarding the exchangeability of calibration and test data is valid. In practice, an ACP is considered valid if the difference between the expected and the actual error rate, ΔEr, is small, although a maximum threshold on the difference is not clearly defined. A negative difference indicates that the ACP fails to account for the errors of holdout data and is less desirable, while a positive difference is acceptable as the ACP accounts for all errors of the holdout data. In this work, the implications of a negative difference and a difference that is greater than 0.05 is explored. A difference greater than 0.05 is used as a threshold as it represents an ACP that fails to explain more than 5% of the holdout data and is also a standard threshold applied in significance testing.

The efficiencies of the ACPs were assessed based on their average PI size; smaller PIs are preferable as they indicate a smaller amount of uncertainty associated with the predictions. The size of the standard PIs was used as a benchmark for the efficiency of the normalised ACPs. A normalising method is optimal if it produces PIs that are, on average, more efficient than the standard (non-normalised) ACP on holdout data.

Finally, the usefulness of the PI estimates was evaluated based on their size and their correlation to the actual prediction errors. The former was done by comparing the PI estimates of the standard ACPs to the experimental error estimates of the data and the endpoint value range, which were provided in Chapter 4. The correlation of the normalised PIs to the actual prediction errors of the holdout data was calculated using Pearson's r and Spearman's ρ.

A summary of the ACP parameters investigated is provided in Table 7-1.

Table 7-1. Summary of parameters optimised during training of the aggregate CPs

Sampling	Cross subsampling
	Random subsampling
	Bootstrap
Underlying model	SVM
	RF
Normalising method	SVM error model
	RF error model
	AD-based RF model
	D2M index

ACP was implemented using Python's *nonconformist* library and necessary modifications of the original code using Python scripts that were required for the normalisation applied using the D2M index and the AD-based RF error model.

The ACP models built were based on the underlying QSAR models of Chapter 5 for the LogD and ADME data. Due to the incompatibility of the PLS algorithm with the nonconformist package, ACPs could not be obtained for the PLS LogD model.

The performance of the ACPs for the LogD models was evaluated on the same holdout data used for the evaluation of the QSAR and error models built in the previous chapters, while the performance of the ACPs for the ADME models was evaluated on separate holdout data, multiple holdout data and time-split data. The performance of the ACPs on multiple holdout data was assessed for ADME models trained using different settings than those previously described and their details are provided in the results section.

7.3 Results

In this section, the results obtained from the optimisation of standard and normalised conformal predictors for the LogD dataset are first presented and discussed. These are then followed by the results obtained from the optimisation of the conformal predictors for the ADME datasets and the evaluation of the utility of their PI estimates for the identification of novel compounds. Finally, the performance of the ACPs on repeatedly sampled holdout data and temporal test data of the ADME data is discussed.

7.3.1 LogD dataset

Standard ACPs were built for the KNN, RF and SVM LogD models and were then optimised for the ACP size, the sampling technique and the normalisation method. The conformal predictors were evaluated at 80% confidence on randomly sampled holdout data from each ADME training set.

7.3.1.1 Conformal predictor optimisation

Figure 7-1 illustrates the error rates of the standard ACPs and the size of the PIs estimated for the KNN, RF and SVM LogD models at all levels of confidence. On the left, the expected error rate, which is indicated by the black diagonal line, is plotted against the actual error rate of the ACPs for the holdout data. The expected error rate of the ACP at each confidence level, α_i, is calculated as the significance, 1- α_i, and is indicated by the black diagonal line in the confidence vs. error rate plots. The ACPs are valid at all confidence levels with small fluctuations in the expected error rates. The RF ACP yield the smallest fluctuations and are consistently in high agreement with the expected rates. The SVM and KNN ACPs yield smaller error rates than the expected error rate for confidence levels between 10-80%. This suggests that the RF ACPs are more reliable than the SVM and KNN ACPs as their error rates are closest to the expected error rates.

To the right of Figure 7-1, the size of the PI estimates for the predictions of the LogD models is plotted across all confidence levels. The narrowest PIs are obtained for the SVM LogD model, which is also the most accurate (see Chapter 5), while the RF and KNN LogD models produce PIs of similar size.

The hashed lines in Figure 7-1 mark the chosen confidence threshold of 80%, where it can be seen that valid PIs are obtained with only small differences between the actual and the expected error rates for the three models. Furthermore, the size of the PIs estimates is between 1.6 and 2.0 which is approximately 30% the LogD value range of the modelled data.

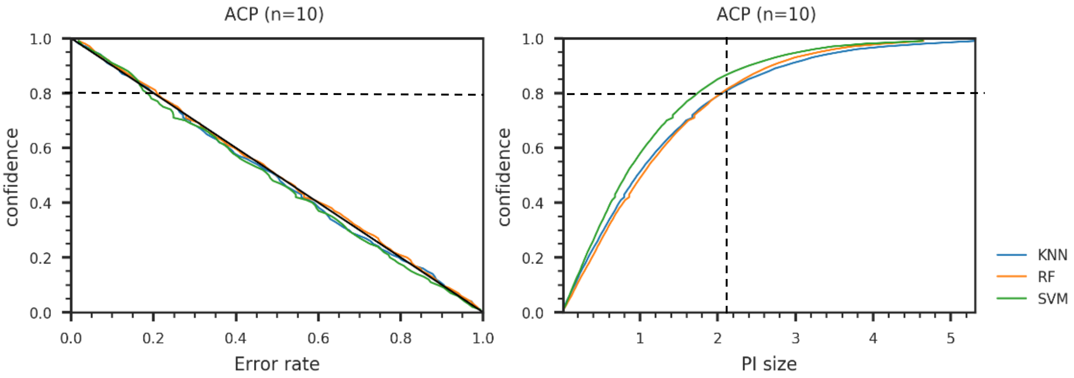


Figure 7-1. Error rate (left) and size of PIs (right) estimated by standard ACPs of size 10 for the underlying KNN, RF and SVM models of the LogD dataset on holdout data from random sampling.

7.3.1.1.1 ACP size

The results from the evaluation of the standard ACPs with different sizes on holdout data for the underlying KNN, RF and SVM models are summarised in Table 7-2. In Table 7-2, the validity of the ACPs for different sizes was evaluated at 80% confidence and is expressed as the difference between the expected and the actual error rate, ΔEr, of the conformal predictor on holdout data. The ACPs with a negative ΔEr, i.e., the actual error rate is larger than the expected error rate, are italicized. It is seen that increasing the ACP size from 10 causes the error rate of the ACP to increase in the case of the KNN and RF models, but to decrease in the case of the SVM model. The differences in the error rates are small, however, and do not exceed the set threshold of ΔEr at 0.05. An interesting observation is that the actual error rates of the ACPs for the SVM model are consistently smaller than the expected error rate of 0.2, while in the case of KNN and RF the actual error rates exceed 0.2. Increasing the ACP size improves the efficiency of the PI estimates only in the case of the RF model, as the PI size steadily decreases; however, in the case of the KNN and the SVM model the most efficient PIs are observed for sizes of 10 and 100, respectively.

Table 7-2. Difference between the expected and actual error rate, ΔEr, and the PI size of ACPs (N= 10, 100, 1000) for the underlying logD models at 80% confidence

	N	**KNN**	**RF**	**SVM**
ΔEr	**10**	0.000	*-0.008*	0.015
	100	*-0.009*	*-0.013*	0.012
	1000	-0.001	*-0.013*	0.017
PI size	**10**	2.065	2.048	1.742
	100	2.078	2.033	1.736
	1000	2.076	2.029	1.742

A small difference between the expected error rate and the actual error rate, ΔEr, of an ACP for the set confidence level suggests that the ACP yields approximately valid results. Therefore, Table 7-2 indicates that valid ACPs have been obtained for all three models with PI estimates that span over 30-35% of the LogD value range, which is 6 log units (see Chapter 3). A larger difference, whereby the actual error rate is smaller than the expected error rate suggests that the ACP performs better than expected on holdout data as it yields more PIs that are valid on holdout data than on calibration data. This is an effect of the calibration errors being larger than holdout errors at an 80% level of confidence, which decreases the error rate, as the PIs are more likely to include the measured values of the holdout data.

For example, an 80% confidence value suggests that the expected error rate is 0.2, so that for a hypothetical value of ΔEr =+0.050, the actual error rate will be 0.15, i.e., ΔEr = expected – actual = 0.2 – 0.15. This means that while 80% of the estimated PIs for calibration data are expected to contain the true measurement; 85% of the estimated PIs contain the true measurement on holdout data. In contrast, a value of ΔEr = - 0.05, suggests that the errors of the holdout set are larger than the errors of the calibration set for the specified confidence threshold and, thus, the PI estimates exclude the actual measurements of the holdout data more frequently than expected, i.e., by 5%.

Regardless of whether the difference is positive or negative, a large difference in the expected and actual error rates indicates that the theoretical assumption of exchangeability is not valid and that the ACP is not reliable for the holdout data. Therefore, an ACP is valid only for small ΔEr values. This observation highlights the importance of the error models, which were investigated in the previous chapter, being predictive, as they may be used to estimate the prospective error distribution of holdout data and test in advance whether a valid ACP may be obtained.

7.3.1.1.2 Sampling

The influence of the three sampling methods on the validity and the efficiency of the ACPs is shown in Table 7-3 for different ACP sizes. The most efficient PI estimates are in bold.

Table 7-3. Difference between the expected and actual error rate, ΔEr, and PI size of ACPs for the underlying logD models

N		Sampling	KNN	RF	SVM
10	**ΔEr**	**Bootstrap**	0.015	*-0.005*	0.014
		Cross-sampling	*-0.015*	*-0.018*	0.002
		Random	0.000	*-0.008*	0.015
	PI Size	**Bootstrap**	2.185	1.980	1.779
		Cross-sampling	**1.978**	**1.966**	**1.662**
		Random	2.065	2.048	1.742
100	**ΔEr**	**Bootstrap**	0.012	*-0.010*	0.012
		Cross-sampling	0.012	*-0.006*	0.019
		Random	*-0.009*	*-0.013*	0.012
	PI Size	**Bootstrap**	2.152	**2.001**	1.759
		Cross-sampling	2.108	2.012	**1.729**
		Random	**2.078**	2.033	1.736
1000	**ΔEr**	**Bootstrap**	0.005	*-0.006*	0.012
		Cross-sampling	0.042	0.037	0.044
		Random	*-0.001*	*-0.013*	0.017
	PI Size	**Bootstrap**	2.161	**2.000**	1.760
		Cross-sampling	2.330	2.205	1.952
		Random	**2.076**	2.028	**1.742**

The PIs obtained by cross-sampling of the calibration data are sensitive to changes in ACP size, while those obtained by bootstrap sampling and random sampling are more robust. Increasing the size of the ACP results in an increase of the PI size by approximately 0.4 when cross-sampling is applied. For an ACP of size 10, cross-sampling produces the narrowest PIs but at the cost of the increasing error rate when the underlying algorithm is KNN and RF, which is indicated by a negative ΔEr. For an ACP of size 1000, cross-sampling leads to the least efficient PIs for all models and a large decrease in the actual error rate of the ACP. The least variation between the PI estimates obtained by cross-sampling and the other sampling methods is observed for an ACP size of 100, while the highest variation is observed for the PIs from ACPs of size 1000. This is a consequence of the difference in the calibration set size in cross-sampling; the calibration data sampled at each iteration represent 10%, 1% and 0.01% of the training set data for ACPs sizes of 10, 100 and 1000, respectively. This demonstrates that applying the same threshold to an error distribution estimated from a smaller calibration set with smaller size will increase the size of the error estimate, thus, making it less accurate.

Bootstrap and random sampling yield PI estimates of similar size across all methods and the actual error rates obtained are less varied. For ACPs of size 100 and 1000, bootstrap sampling produces more efficient PIs for RF models, while random sampling produces more efficient PIs for KNN.

7.3.1.1.3 Normalisation

Different methods were tried for the normalisation of the ACPs, namely a D2M index, descriptor-based RF and SVM error models and an AD-based RF error model. These were applied for ACPs with size 10 and cross-sampling as these settings produced the most efficient PIs for holdout data for all three underlying algorithms with no normalisation. In Table 7-4, the performance of the normalised ACPs is compared to the performance of the standard ACP with the most efficient PIs underlined and in bold. The ACPs normalised using the descriptor-based RF error model, in the case of RF and SVM QSAR models, and the descriptor-based SVM error models, in the case of the KNN QSAR model, yielded the smallest difference between the expected and the actual error rate and the most efficient PIs. The large value for the average PIs for the KNN model normalised using the AD-based error model is attributed to poor error estimates produced by the model, which is trained on the residual errors of the training data rather than the cross-validation data. Note that this is the main difference in the error models trained using the Python implementation of CP and the error models trained separately in Chapter 5. As a result, the errors of the calibration data are underestimated and their normalisation with the error estimates produces very large critical values for the calculation of the PIs. Nevertheless, this suggests that the current normalisation is not suitable for the KNN model and that the error model would have to be trained on cross-validation errors to improve the efficiency of the PI estimates.

Table 7-4. Comparison of the difference between the expected and actual error rate, ΔEr, and mean PI size of standard and normalised ACPs (N=10) for the underlying RF and SVM models

	ACP	KNN	RF	SVM
	Standard	*-0.015*	*-0.018*	0.002
ΔEr	**D2M index**	0.012	*-0.018*	0.015
	RF	0.014	0.010	0.000
	SVM	0.004	0.031	0.012
	AD-based RF	0.026	0.031	0.020
PI size	**Standard**	1.978	1.966	1.662
	D2M index	2.164	2.042	1.789
	RF	2.524	<u>**2.000**</u>	<u>**1.723**</u>
	SVM	<u>**2.057**</u>	2.144	1.908
	AD-based RF	5.780	2.122	1.860

7.3.1.2 Evaluating the usefulness of normalised PIs

In the previous results, it was seen that the size of the PI estimates is approximately a third of the LogD value range which is one order of magnitude larger than the experimental error estimate of 0.1. This highlights the large prediction uncertainty of the LogD models in relation to the experimental error, which suggests that the models cannot achieve as high precision as the assay at 80% confidence.

However, as already mentioned, the sizes of the normalised PIs may be used to prioritize compounds for further testing. Large PIs indicate high uncertainty associated with the predictions suggesting that the model does not have enough data for the respective compounds; therefore, further experimental testing of similar compounds is required.

The linear and rank correlation of the normalised PIs to the actual prediction errors of the KNN, RF and SVM LogD models are shown in Table 7-5. Higher correlation is seen between the PIs obtained by normalisation with the descriptor-based error models and the errors of the underlying RF and SVM models, as well as the AD-based PI estimates with the underlying RF errors. Interestingly, normalised PIs that are estimated using SVM error models are more strongly correlated with the errors of the RF LogD model than the normalised PIs estimated using RF error models. Overall, the results suggest that PIs obtained following normalisation with the RF and SVM error models may be useful in the prioritisation of compounds.

Table 7-5. Pearson and Spearman correlation coefficients between the variable PI estimates and the actual prediction errors of the underlying models' predictions

	ACP	KNN	RF	SVM
Pearson's r	**D2M index**	0.08	0.06	0.08
	RF	-0.05	0.31	0.28
	SVM	-0.01	**0.38**	0.25
	AD-based RF	-0.11	0.27	0.12
Spearman's ρ	**D2M index**	0.06	0.01	0.05
	RF	0.06	0.33	0.24
	SVM	0.07	**0.39**	0.23
	AD-based RF	-0.29	0.28	0.12

7.3.2 ADME datasets

Standard ACPs were built for the for the RF and SVM ADME models and, as for the LogD dataset, were then optimised for the ACP size, the sampling technique and the normalisation method. Subsection 7.3.2.1 describes the results from the optimisation of the ACPs on holdout data that were randomly sampled from the training data using a ratio of 80:20. Subsection 7.3.2.2 contains the results of the ACPs applied on the full training set and the external test set that was originally supplied by Lilly.

7.3.2.1 Conformal predictor optimisation

The performance of standard ACPs applied to the underlying RF and SVM QSAR models is illustrated for all levels of confidence in Figure 7-2 and Figure 7-3, respectively. The plot shows the actual error rate and the size of the PI estimates for the holdout test set.

The confidence vs. error rate plots in the left of Figure 7-2 and Figure 7-3 indicate that for the two underlying algorithms the ACPs had smaller error rates on the holdout data at higher confidence levels. However, larger actual error rates than the expected error rate were obtained for the RF models for most datasets at lower confidence levels, particularly datasets 1 and 4, suggesting that there is less agreement between the error distributions of the calibration and holdout data. At 80% confidence, which was the chosen confidence threshold, valid PIs with small differences between the actual and the expected error rates were observed for all datasets. A closer look at the data shown in Figure 7-2, indicates that the highest observed margins for the error rates, ΔEr, were observed for the RF model of dataset 2 and the SVM model of dataset 7 with sizes of 0.010 and 0.011, respectively, which are both well below the ΔEr threshold of 0.05. These values suggest a higher than expected error rate by 1.0 % and 1.1 % for the holdout sets of dataset 2 and dataset 7, respectively.

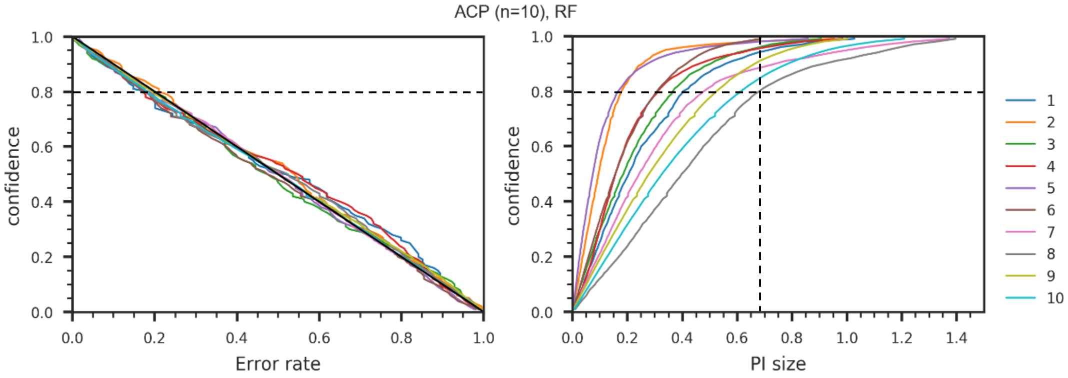


Figure 7-2. Error rate (left) and size of PIs (right) estimated by standard ACP (N=10) for the underlying RF models of the 10 ADME datasets on holdout data from random sampling.

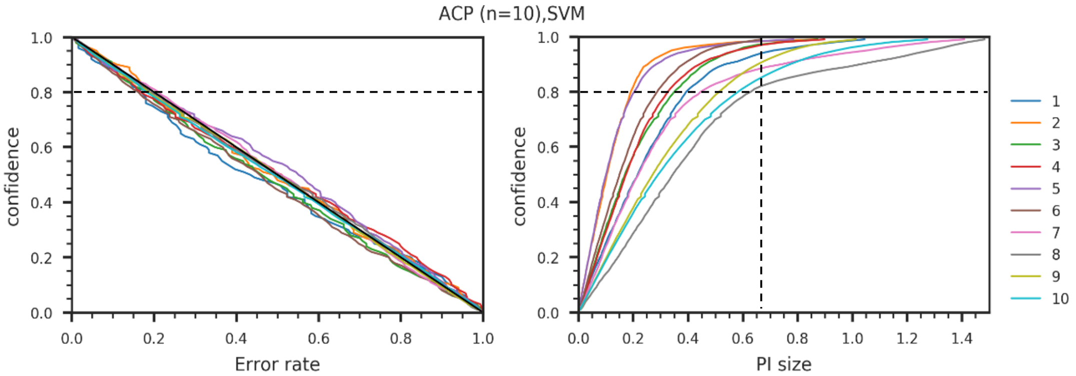


Figure 7-3. Error rate (left) and size of PIs (right) estimated by standard ACP (N=10) for the underlying SVM models of the 10 ADME datasets on holdout data from random sampling.

In the confidence vs. PI size plots in the right of Figure 7-2 and Figure 7-3, narrower PI distributions are, generally, obtained for more accurate ADME models. Levels of confidence 50% or lower, where half of the future measurements or more are likely to fall outside the PIs, are of less value; as the PIs will be too narrow and inaccurate most of the time. However, as seen in the figures above, higher confidence levels are associated with larger PIs. Furthermore, as the PIs are estimated from the underlying model's empirical error distribution, it follows that underlying models with low accuracy will also yield large PIs that are less informative at high confidence levels (see Section 7.3.2.2). This highlights that setting a suitable confidence level requires making a trade-off with the PI size by taking into account the error distribution of the calibration data.

For example, in Figure 7-2 and Figure 7-3, a 99% confidence threshold results in PI estimates with size between 0.7 – 1.4; which corresponds to 70 % - 140% coverage of the endpoints' value range. However, more precise PIs may be obtained, with a size of 0.2 or less, given an 80% confidence threshold.

7.3.2.1.1 ACP size

The performance of standard ACPs for the underlying RF and SVM models were compared for ACP sizes 10, 100 and 1000 to find the optimal size, and to the performance of an ACP of size 1, which uses a single calibration sample for the estimation of the PIs. An ACP of size 1, is equivalent to an inductive CP, therefore, it is referred to as ICP below. In Figure 7-4 validity is measured as the difference between the expected error rate and the actual error rate of the ACPs for increasing size, i.e., ΔEr. The ICPs are generally valid with smaller error rates than the ACPs, except in the case of dataset 3. However, the ICP results were not robust as the PI estimates were based on a single calibration set draw. For increasing ACP size, the actual error rate decreases for the underlying RF models of datasets 1, 5 and 6, and the underlying SVM models of datasets 1, 2 and 6, as indicated by the increase in the difference between the expected error rate and actual error rate. In fact, a difference greater than 0.05 was observed for the larger ACPs of the underlying SVM models of dataset 6. Increasing the number of calibration set draws from the training data results in greater coverage of the models' error distribution and makes it more robust.

Increasing the size of the SVM ACP from 10 to 1000 reduces the error rate of datasets 3 and 4 and improves the efficiency of PIs by 0.05.

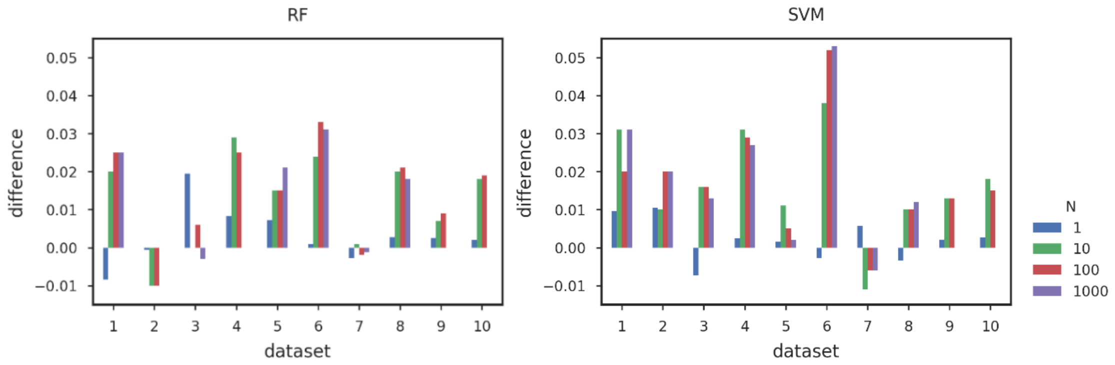


Figure 7-4. Difference between the expected and actual error rate of ACPs with different sizes for the underlying RF (left) and underlying SVM (right) models

The validity and the efficiency measures of the ICP with size 1 and the three ACPs with sizes of 10, 100 and 1000 are provided below. In Table 7-6 and Table 7-7, the validity is expressed as the difference between the expected error rate and the actual error rate of the CP on the holdout data, ΔEr. Negative ΔEr values are italicized and, as discussed earlier, indicate that the actual error rate of the conformal predictor exceeds the expected error rate. A negative ΔEr is less preferable than a positive ΔEr as it indicates that a larger number of PIs will exclude the observed holdout measurement; a positive ΔEr indicates that a smaller number of PIs will exclude the observed holdout measurement. As mentioned in the Methods section of this chapter, the difference is considered significant when the ΔEr is greater than 0.05 and such values are indicated in bold in the tables.

In Table 7-6, which shows results of the RF QSAR models, ACPs with increasing size result in a decrease in the PIs of datasets 3, 4 and 8 but an increase in the PIs of datasets 2, 5, 9. In Table 7-7, which shows the SVM QSAR model results, increasing the size of the ACP results in a decrease of the PI sizes of datasets 3, 4 and 9 and an increase of the PI size for datasets 6, 7 and 8. The size of the PI estimates of both algorithms was subject to smaller variations in datasets 1, 5, 6 and 10 for ACPs of different sizes. It is seen that rounding the PI estimates of these datasets to a precision of two shows no difference in the estimates obtained by ACPs of different sizes, while for all other datasets rounding to a precision of two results in a difference of 0.01 in the PI estimates.

Table 7-6. Difference between the expected and actual error rate and size of PI estimated for RF predictions at 80% confidence by standard ACP

Standard ACP	N	**Dataset**									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**1**	*-0.008*	*-0.001*	0.019	0.008	0.007	0.001	*-0.003*	0.003	0.003	0.002
	10	0.020	*-0.010*	0.000	0.029	0.015	0.024	0.001	0.020	0.007	0.018
	100	0.025	*-0.010*	0.006	0.025	0.015	0.033	*-0.002*	0.021	0.009	0.019
	1000	0.025	*-0.010*	*-0.003*	0.025	0.021	0.031	-0.001	0.018	0.009	0.017
PI size	**1**	0.402	0.177	0.384	0.308	0.172	0.309	0.477	0.682	0.530	0.618
	10	0.403	0.179	0.366	0.311	0.166	0.308	0.479	0.681	0.525	0.614
	100	0.400	0.184	0.364	0.309	0.166	0.306	0.480	0.676	0.529	0.614
	1000	0.403	0.182	0.362	0.308	0.168	0.307	0.480	0.673	0.528	0.613

Table 7-7. Difference between the expected and actual error rate and size of PI estimated for SVM predictions at 80% confidence by standard ACP

Standard ACP	N	**Dataset**									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**1**	0.010	0.010	*-0.007*	0.002	0.002	*-0.003*	0.006	*-0.003*	0.002	0.003
	10	0.031	0.010	0.016	0.031	0.011	0.038	*-0.011*	0.010	0.013	0.018
	100	0.020	0.020	0.016	0.029	0.005	**0.052**	*-0.006*	0.010	0.013	0.015
	1000	0.031	0.020	0.013	0.027	0.002	**0.053**	*-0.006*	0.012	0.011	0.017
PI size	**1**	0.406	0.191	0.347	0.321	0.202	0.292	0.453	0.627	0.512	0.586
	10	0.393	0.191	0.350	0.326	0.203	0.287	0.446	0.629	0.513	0.582
	100	0.393	0.196	0.345	0.318	0.202	0.290	0.449	0.632	0.510	0.581
	1000	0.395	0.195	0.343	0.321	0.202	0.291	0.450	0.636	0.511	0.581

7.3.2.1.2 Sampling

The effect of random subsampling, cross-subsampling and bootstrap sampling on the performance of the ACPs with size N=10 at 80% confidence are provided in Table 7-8 and for the RF and SVM models, respectively. The results for ACPs of size 100 and 1000 from different sampling methods are provided in the Appendix (Tables A 12 and A 13) and show similar trends to those observed for the LogD dataset.

On average, there is a smaller difference between the expected error rate and the actual error rate of ACPs with random sampling in relation to the other sampling methods. However, ACP with cross-subsampling resulted in more efficient PIs for both the RF and SVM algorithms. The PI estimates obtained from bootstrap sampling and random sampling had similar sizes; while the sizes of the PI estimates from cross-subsampling were different due to the difference in the calibration set size. This trend was similar to that observed in the LogD results. Smaller fluctuation in the PI estimates across the different sampling methods were observed for datasets 4, 5, 6 and 9 for both underlying algorithms.

Table 7-8. Difference between the expected and actual error rate and PI size of ACPs (N=10) for the underlying RF ADME models across different sampling methods

		Dataset									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**B**	0.036	*-0.020*	*-0.010*	0.022	0.021	0.033	0.017	0.018	0.015	0.023
	C	0.015	0.015	*-0.006*	0.022	0.027	0.025	*-0.002*	0.014	0.005	0.017
	R	0.020	*-0.010*	0.000	0.029	0.015	0.024	0.001	0.020	0.007	0.018
PI Size	**B**	0.405	0.183	0.355	0.311	0.163	0.309	0.491	0.679	0.527	0.618
	C	0.391	0.192	0.355	0.303	0.168	0.299	0.477	0.670	0.518	0.599
	R	0.403	0.179	0.366	0.311	0.166	0.308	0.479	0.681	0.525	0.614

B: bootstrap sampling, C: cross-subsampling, R: random sampling

Table 7-9. Difference between the expected and actual error rate and PI size of ACPs (N=10) for the underlying SVM ADME models across different sampling methods

		Dataset									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**B**	0.031	0.045	0.026	0.018	0.011	0.049	*-0.009*	0.010	0.013	0.017
	C	0.036	0.035	0.023	0.020	0.009	0.047	0.001	0.010	0.011	0.011
	R	0.031	0.010	0.016	0.031	0.011	0.038	*-0.011*	0.010	0.013	0.018
PI size	**B**	0.408	0.203	0.347	0.326	0.205	0.294	0.447	0.644	0.511	0.585
	C	0.409	0.198	0.336	0.306	0.199	0.286	0.460	0.634	0.506	0.568
	R	0.393	0.191	0.350	0.326	0.203	0.287	0.446	0.629	0.513	0.582

B: bootstrap sampling, C: cross-subsampling, R: random sampling

The influence of ACP size is on average smaller than the influence of the sampling method on the size of the PI estimates and error rates of ACPs. It was seen that ACPs with a size of 10 were able to produce valid ACPs for all datasets and the size of PIs obtained were similar to the sizes of PIs estimates from ACP with a larger size for a precision of 2. Subsequent ACPs were, therefore, trained with an ensemble size N = 10 and random sampling to further study the effect of normalisation on the utility of the PI estimates obtained for the RF and SVM ADME models.

7.3.2.1.3 Normalisation

Compound-specific PIs were estimated from ACPs of size 10 that were normalised using the D2M index and, prediction error estimates obtained from the descriptor-based RF and SVM error models and prediction error estimates obtained from the AD-based RF error models. The performance of the normalised ACPs is compared to the performance of the standard ACPs on holdout data. As discussed above, a useful normalisation method is, generally, expected to yield PIs that are more efficient than the PIs from no normalisation. The results are provided in Table 7-10 for the underlying RF ADME models and

Table 7-11 for SVM ADME models, where normalised PI estimates that are smaller than the non-normalised PIs are in bold and underlined. A large difference between the expected and actual error rate of the ACP is indicated in plain bold.

It is seen that the most efficient PIs were obtained from normalised ACPs using descriptor-based RF and SVM error models. On the other hand, AD-based error models and the D2M index resulted in the least efficient PI estimates. Normalisation generally reduced the error rates of the ACPs, which is indicated by the positive increase in the difference between the expected error rate and the actual error rate. This is because, during normalisation, large PI estimates are assigned to compounds with higher uncertainty than the calibration data, which increase the average PI size and, eventually, makes it more likely that the observed value will be included in the PI, i.e., the error rate is reduced.

Normalisation using error models did not improve the efficiency of the PIs for the RF QSAR models of datasets 2, 5 and 6, nor the efficiency of the PIs for the SVM QSAR models of datasets 2, 5, 6 and 8; as the average PI estimates were equal or larger than the non-normalised PIs. In most cases, normalisation using SVM error models produced more efficient PIs than the D2M index. For the RF QSAR models, normalised ACPs that used SVM error models as the normalising function were more efficient, particularly for datasets 2, 3, 4, 5 and 6.

Table 7-10. Comparison of the difference between the expected and actual error rate, ΔEr, and mean PI size of standard and normalised ACPs (N=10) for the underlying RF models

	ACP	Dataset									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**Standard**	0.020	*-0.010*	0.000	0.029	0.015	0.024	0.001	0.020	0.007	0.018
	D2M index	0.020	0.000	0.042	0.020	0.013	*0.030*	0.002	*0.012*	0.002	0.015
	RF	0.041	0.005	0.035	0.016	0.036	0.031	0.036	0.026	0.018	0.030
	SVM	0.015	0.005	0.026	0.004	0.034	**0.052**	0.017	0.014	0.024	0.027
	AD-based RF	0.020	-0.005	0.035	**0.056**	0.046	**0.068**	*0.042*	**0.064**	0.023	0.039
PI size	**Standard**	0.403	0.179	0.366	0.311	0.166	0.308	0.479	0.681	0.525	0.614
	D2M index	0.412	0.190	**0.356**	**0.291**	0.166	0.315	**0.465**	0.697	0.538	**0.600**
	RF	0.430	0.192	0.368	**0.290**	0.174	0.308	**0.470**	0.704	**0.523**	**0.594**
	SVM	**0.401**	0.187	0.369	**0.294**	0.182	0.309	**0.449**	**0.664**	0.531	**0.594**
	AD-based RF	0.438	0.200	0.376	**0.302**	0.182	0.324	0.507	0.799	0.566	0.622

Table 7-11. Comparison of the difference between the expected and actual error rate and mean PI size of standard and normalised ACPs (N=10) for the underlying SVM models

	ACP	Dataset									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**Standard**	0.031	0.010	0.016	0.031	0.011	0.038	*-0.011*	0.010	0.013	0.018
	D2M index	0.031	0.015	0.042	0.029	0.000	*0.014*	0.007	*0.015*	0.013	0.015
	RF	0.025	0.025	0.032	0.038	*-0.010*	**0.058**	*-0.003*	0.010	0.009	0.019
	SVM	0.047	0.015	0.026	0.022	0.025	**0.069**	0.018	0.008	0.009	0.022
	AD-based RF	0.047	**0.050**	0.035	**0.058**	0.046	**0.050**	*0.014*	0.026	0.033	0.029
PI size	**Standard**	0.393	0.191	0.350	0.326	0.203	0.287	0.446	0.629	0.513	0.582
	D2M index	0.398	0.203	0.356	**0.313**	**0.201**	0.295	0.452	0.665	0.530	0.600
	RF	0.406	0.199	**0.337**	**0.317**	0.205	0.301	0.451	0.681	**0.499**	**0.562**
	SVM	**0.383**	0.195	**0.341**	**0.308**	0.203	0.291	**0.428**	0.643	**0.498**	**0.567**
	AD-based RF	0.421	0.219	**0.343**	**0.318**	0.208	0.297	0.471	0.716	**0.501**	**0.563**

7.3.2.2 Evaluating the usefulness of normalised PIs

The efficiency of the PIs estimated by ACP is strongly determined by the accuracy of the underlying models and the quality of the modelled data. In the left of Figure 7-5, the average size of uniform, i.e. non-normalised, PIs estimated with standard ACPs is plotted against the 10-fold cross-validation RMSE of the underlying models. It is seen that models with an RMSE greater than 0.20 may produce very large PIs that span over 40 % or more of the endpoint value range of [0, 1].

In the right of Figure 7-5, it seen that large PIs are also associated with datasets with experimental errors greater than 0.1, i.e., 10% of the endpoint value range. The results suggest that obtaining PIs as low as 0.2 at 80% confidence requires that the underlying model has high accuracy with an RMSE value that is close to 0.10.

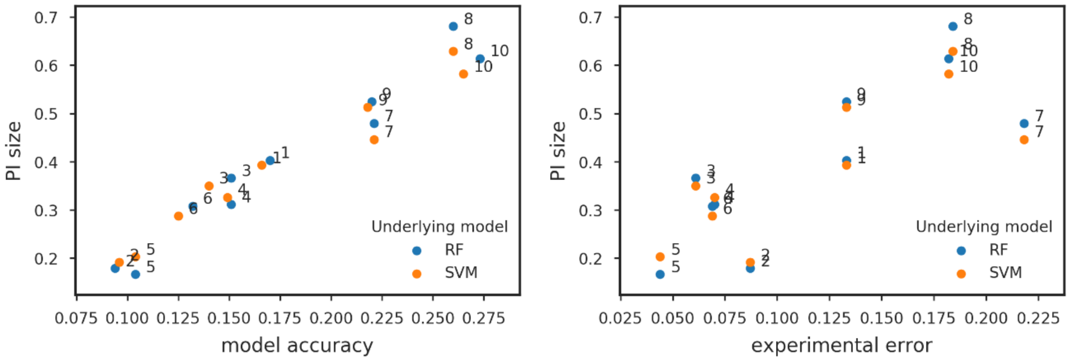


Figure 7-5. Size of non-normalised PIs estimated from standard ACPs of size 10 plotted against the accuracy of the underlying RF and SVM ADME models (left) and the experimental error of the data (right) measured by RMSE

Experimental error was used to assess the utility of the average PI size as it is often used as a benchmark of QSAR model accuracy. This involved calculating the ratio between the error margin of the PIs, i.e., half the PI size, obtained from standard ACP and the experimental error estimate of the data. The ratio between the PI error margins for the RF and SVM QSAR models and the experimental error estimate is provided for each dataset in Table 7-12.

A maximum threshold of three (Haas et al., 2013) was used to assess whether PIs were useful. Using this criterion, the PIs for all models were useful. However, the PIs for the RF and SVM models of datasets 1, 2, 7 had a ratio closer to one, which suggests that the experimental and prediction uncertainties are of similar size. A ratio close to two suggests that the prediction uncertainty of datasets 4, 5, 6, 8, 9 and 10 is approximately 2-fold the experimental uncertainty. Finally, a ratio of three suggests a 3-fold difference in the prediction and experimental uncertainties of dataset 3, which is on the borderline of the utility threshold.

Table 7-12. Utility assessment of standard ACP PI estimates based on the ratio between the PI error margin and the experimental error estimate

Dataset	**Ratio**	
	RF	**SVM**
1	1.5	1.5
2	1.0	1.1
3	3.0	2.9
4	2.2	2.3
5	1.9	2.3
6	2.2	2.1
7	1.1	1.0
8	1.9	1.7
9	2.0	1.9
10	1.7	1.6

More efficient PIs may be obtained, however, if normalisation is applied using an uncertainty estimation method that correlates well with the actual errors of the models' predictions. In the previous results, improvement in the efficiency of PIs was observed for the models that were less accurate than the RF and SVM models of datasets 2 and 5.

Pearson's correlation coefficient was calculated between the normalised PI estimates and the actual prediction errors of each dataset and the results are provided in Table 7-13 and Table 7-14, with correlations greater than 0.40 marked in bold and underlined. For the RF QSAR models, on average, stronger correlations were observed for the normalised PIs obtained from descriptor-based RF and SVM error models. For the SVM QSAR models, weaker correlations were obtained for normalised PIs, but these were stronger when normalisation with the AD-based RF error models was applied. In addition, the strongest correlations between the normalised PI estimates and prediction errors on the holdout data were observed for models that were more efficient without applying any normalisation. The calculation of Spearman's rank correlation coefficient produced similar results, which are provided as additional information in the Appendix (Table A 14 and Table A 15).

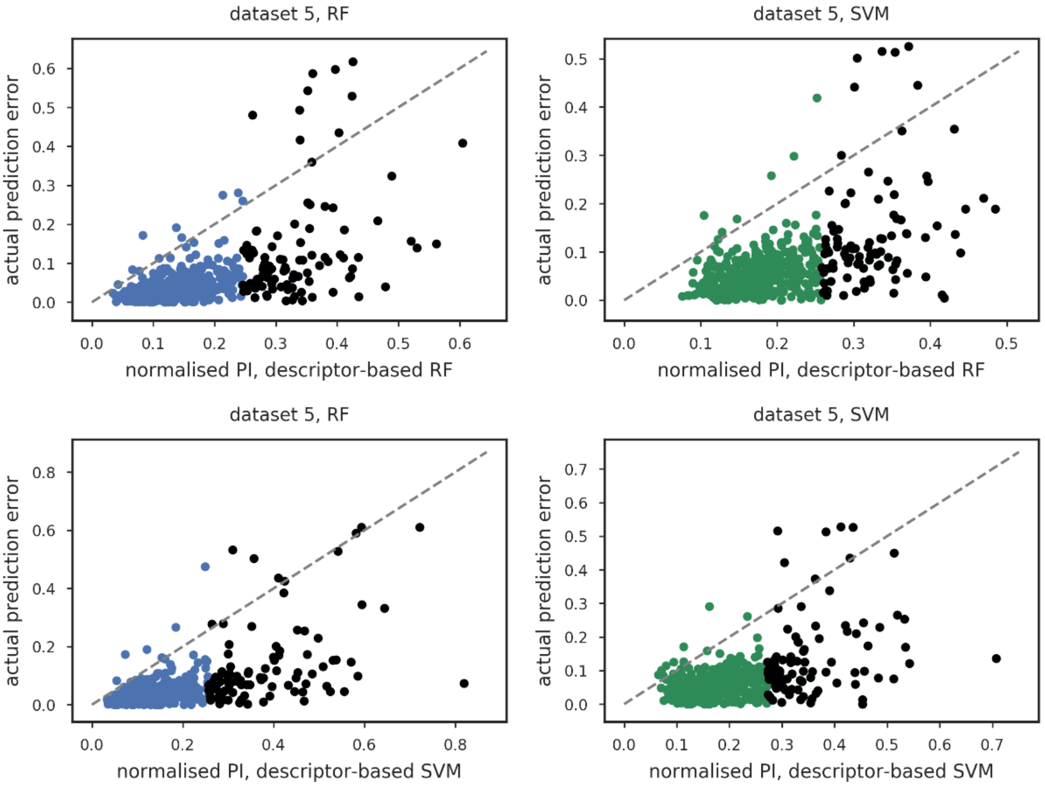
Table 7-13. Pearson's correlation coefficient between the variable PI estimates and the actual prediction errors of the underlying RF predictions

Dataset	**Normalisation method**			
	D2M	**AD-RF**	**RF**	**SVM**
1	0.07	0.06	0.27	0.15
2	0.10	0.06	0.21	0.25
3	0.26	**0.47**	**0.42**	0.40
4	0.22	0.35	0.38	0.36
5	0.21	**0.62**	**0.57**	**0.60**
6	0.20	0.27	0.31	0.35
7	0.18	0.39	0.39	0.34
8	-0.04	0.07	0.12	0.11
9	0.07	0.09	0.30	0.30
10	0.04	0.28	0.36	0.34
Mean	*0.13*	*0.27*	*0.33*	*0.32*

Table 7-14. Pearson's correlation coefficient between the variable PI estimates and the actual prediction errors of the underlying SVM predictions

Dataset	**Normalisation method**			
	D2M	**AD-RF**	**RF**	**SVM**
1	0.03	0.10	0.16	0.10
2	0.10	0.02	0.14	0.18
3	0.19	0.37	0.21	0.32
4	0.22	0.34	0.30	0.33
5	0.20	**0.54**	**0.51**	**0.51**
6	0.24	0.27	0.17	0.25
7	0.13	0.32	0.39	0.33
8	-0.04	0.06	0.11	0.08
9	0.06	0.33	0.30	0.23
10	0.03	0.23	0.30	0.26
Mean	*0.12*	*0.25*	*0.26*	*0.26*

For most datasets, normalisation has produced PIs with weak correlation to the actual prediction errors of the underlying models. Despite their poor predictive performance, estimates from error models seem to be a better option for normalisation than a simple D2M index, particularly when the underlying model is a RF. Figure 7-6 and Figure 7-7 show the normalised PIs plotted against the actual prediction errors of datasets 5 and 3, respectively. These plots are provided for other datasets with weaker correlation between the PI size and the actual prediction error in the Appendix (Figures A 16 - A 21). Highlighted in black are the predictions with a PI size greater than the 80th percentile of the PIs. Points to the left of the diagonal correspond to predictions with non-valid PIs, as their prediction errors are larger than the normalised PI estimates. For both datasets, many of these correspond to prediction error outliers. Here we define prediction error outliers using the 95th percentile as a threshold for the actual prediction errors, which corresponds to a value of 0.2 for dataset 5. It is seen that, applying the 80th percentile as a threshold is effective enough to identify most outliers of the RF and SVM predictions using the AD-based error model PIs, but is less effective for the PIs obtained from the descriptor based-error models. However, it is also seen that many accurate predictions that have been assigned large PIs are also excluded. The errors of dataset 3 are less variable and using a similar threshold is not as effective. It seems likely, therefore, that linear correlation between the PI size and the actual prediction errors stronger than 0.5 is required to identify prediction error outliers successfully.



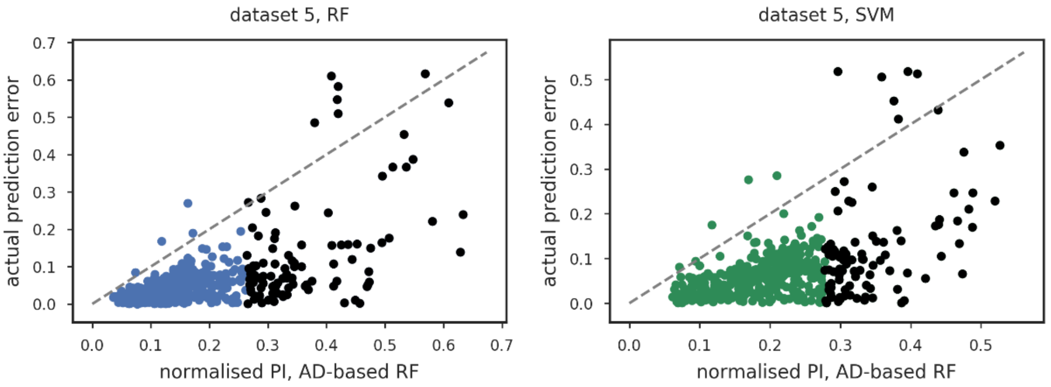


Figure 7-6. Actual prediction errors of the underlying RF (left) and SVM (right) models of dataset 5 plotted against the PIs normalised using descriptor-based RF and SVM error models and AD-based RF error models

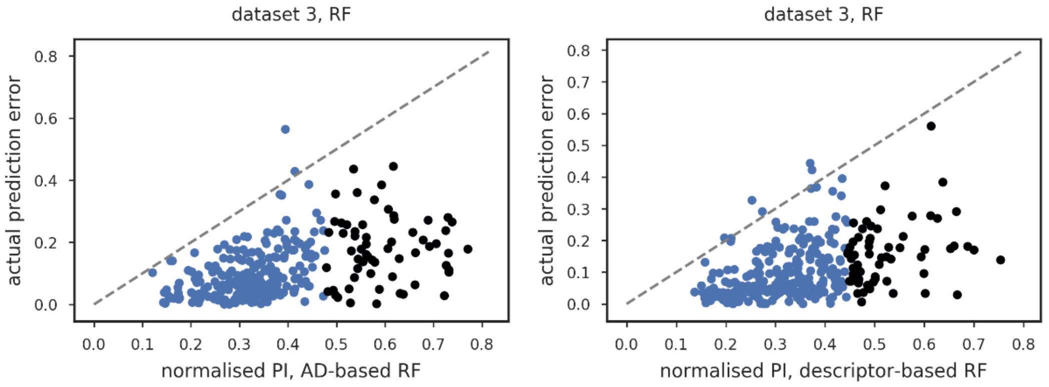


Figure 7-7. Actual prediction errors of the underlying RF model of dataset 3 plotted against PIs normalised using AD-based and descriptor-based RF error models

7.3.2.3 Average performance on multiple holdout data and temporal data

The average performance of the standard ACPs of size 10 was evaluated on 10 holdout samples obtained by random sampling and temporal test sets, as described in Chapter 3, for which the exchangeability of the data is not guaranteed. The performance of a normalised ACP using a RF error model was reassessed for comparison, but on different random holdout sets than in the previous section. The validity and the efficiency metrics of the standard and normalised ACPs are provided in Table 7-15 and Table 7-16, respectively. Normalisation was applied using a descriptor-based RF error model with a size of 200 trees for all datasets.

Table 7-15. Difference between the expected and actual error rate and size of PI estimated for the predictions of the RF QSAR models at 80% confidence

ACP		Dataset									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**Standard**	0.020	0.010	0.003	0.003	0.016	0.008	0.002	0.002	0.005	0.007
	Normalised with RF	0.041	0.038	0.015	0.017	0.031	0.030	0.029	0.036	0.026	0.027
PI size	**Standard**	0.365	0.171	0.348	0.296	0.162	0.287	0.394	0.721	0.506	0.599
	Normalised with RF	0.416	0.197	0.361	**<u>0.295</u>**	0.186	0.321	0.506	0.742	0.531	0.617

The mean PI estimates for RF QSAR models were more stable than for the SVM QSAR models with higher fluctuations observed only for dataset 1 (SD=0.028). The normalised PIs were not as efficient as the non-normalised PIs, except for dataset 4 where they were approximately equal. Greater fluctuations were generally observed in the mean PI estimates of the SVM QSAR models for datasets 2 (SD=0.025), 3 (SD=0.033), 4 (SD=0.021) and 6 (SD=0.020). The highest correlation of the PIs with the absolute residual errors was observed for the PI estimates of the SVM QSAR models for dataset 5 (r=0.408, ρ=0.476) and dataset 7 (r= 0.447, ρ=0.444). For the normalised PIs of RF QSAR models the highest correlation was observed in dataset 4 (r= 0.477 ρ=0.554), dataset 5 (r= 0.508, ρ=0.529) and dataset 7 (r= 0.362, ρ=0.412).

Table 7-16. Difference between the expected and actual error rate and size of PI estimated for SVM predictions at 80% confidence

ACP		Dataset									
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
ΔEr	**Standard**	0.021	*-0.005*	0.007	0.013	0.016	0.002	*-0.001*	*-0.002*	0.001	0.007
	Normalised with RF	0.027	0.029	0.036	0.021	0.036	0.022	0.020	0.018	0.016	0.025
PI size	**Standard**	0.378	0.187	0.316	0.308	0.199	0.268	0.471	0.776	0.499	0.570
	Normalised with RF	0.403	0.207	0.342	0.322	0.203	0.299	0.496	**<u>0.718</u>**	0.512	0.587

The validity and the efficiency of standard ACPs on the temporal holdout set is illustrated in Figure 7-8 and Figure 7-9. Non-valid ACPs that exceed the expected error rate appear on the upper side of the diagonal of Figure 7-8. The hashed, horizontal line indicates that for the 80% confidence threshold, valid PIs were obtained for the RF QSAR models of datasets 1, 2, 3, 8 and 9 and valid PIs for the SVM QSAR models datasets 1, 2, 8 and 9. However, PIs that were consistently valid for all confidence levels and both underlying models were obtained only for datasets 1 and 2. The training set and ACP parameters were the same as in the standard ACPs that were used to estimate the PI for the holdout data in Section 7.3.2.1.3, therefore, the distributions of the PI estimates for external data are the same. However, with the exception of datasets 1 and 2, the increased error rates for the ACPs of the RF and SVM QSAR models for the external data suggest that the external data are not representative of the calibration data. As a result, PI estimates underestimate the uncertainty of the models' predictions for the external test set. Furthermore, this implies that the external data represent a greater shift from the descriptor space of the underlying model for these datasets. Normalisation was not applied for the external data.

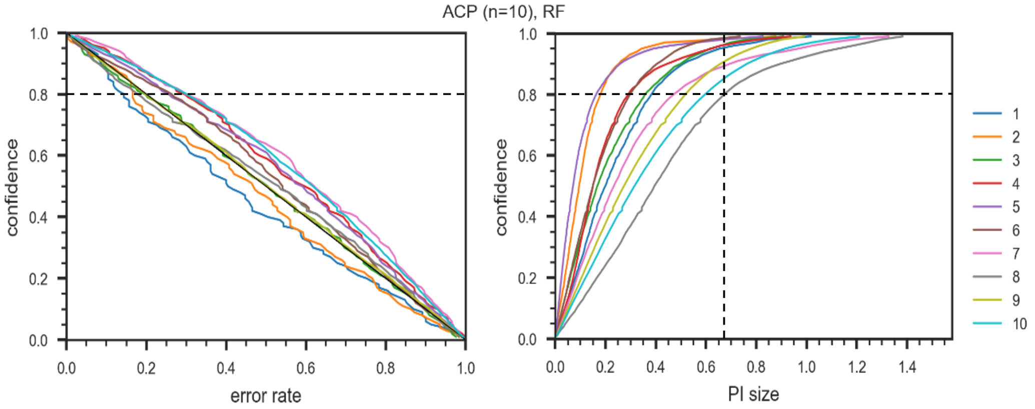


Figure 7-8. Error rate (left) and size of PIs (right) estimated by standard ACP (N=10) for the underlying RF models of the 10 ADME datasets on time-split test data.

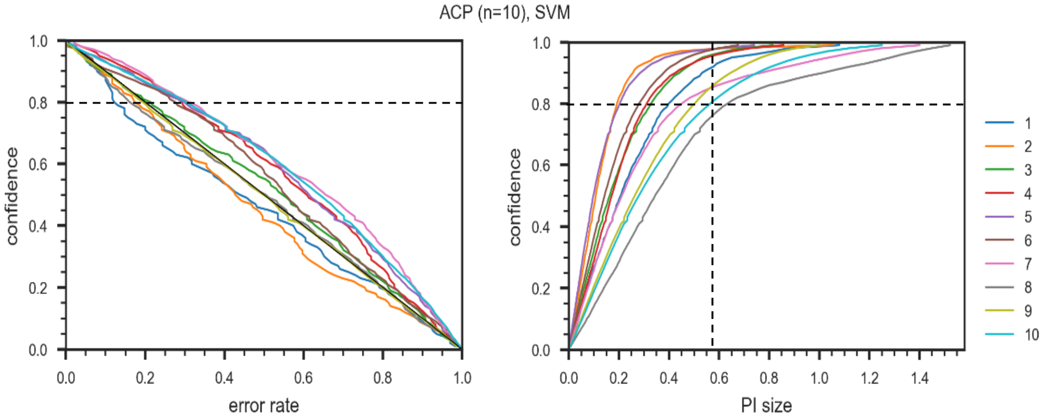


Figure 7-9. Error rate (left) and size of PIs (right) estimated by standard ACP (N=10) for the underlying SVM models of the 10 ADME datasets on time-split test data.

7.4 Discussion

The CP framework provides guarantees about the validity of PI estimates produced under the main assumption of exchangeability. In QSAR this assumption is satisfied by ensuring that the holdout data are representative of the model's applicability domain. Although it is not explicitly stated, it is implied that test data that are representative of the model's applicability domain are also representative of the model's error distribution. However, it has been seen in Chapter 5 that commonly applied applicability domain metrics in QSAR are not well correlated with the algorithm-specific errors. However, regardless of the quality of the reliability estimates, conformal prediction is still able to produce valid estimates by applying a high confidence threshold. Variation is introduced to the PIs of the individual predictions by normalisation, whereby the sizes of the PIs are scaled to represent the difficulty associated with making the prediction. The difficulty is assessed in relation to calibration data; and predictions that are less reliable than the calibration data at the chosen threshold are assigned larger PIs while predictions that are more reliable are assigned smaller PIs. As a result, it is possible to obtain valid PIs using any reliability scoring method even if it is random. However, using random reliability scores to assign PIs to predictions would not be meaningful to the user; even if the PIs were correlated to prediction errors by chance. This was seen in section 7.3.2.2 where the normalisation of ACP using D2M indices yielded PIs with very weak or no correlation to the actual prediction errors. Meaningful PIs that may be utilised to rank predictions require strong correlation of the PI estimates with the prediction errors, which is also challenging using error models. Moderate to strong rank correlation of error estimates with the prediction errors may be sufficient to highlight the presence of prediction error outliers even if the overall predictive performance of the error models is low. Normalised ACPs with rank correlations below 0.40 are not very useful for prioritisation; although, on average, they may yield more efficient PIs than standard PIs, which assign uniform PIs to all predictions. An interesting observation is that stronger correlations between normalised PIs and the underlying models' prediction errors occurred when the models' predictions were more strongly correlated to the prediction errors (see Table A 22 in the Appendix). This highlighted that the model's prediction could be an important variable in the estimation of errors in biased data, where prediction error increases or decreases monotonically with the prediction value, i.e., due to the lack of sufficient measurements on the upper or lower end of the endpoint value range.

Furthermore, standard ACPs at 80% confidence produced PIs with sizes approximately double the error of the underlying model, on average. The intervals covered between 30 - 90% of the modelled endpoints value ranges, which are too wide to be useful in practice. In the work of (Svensson et al., 2018) PIs for smaller datasets were reported with approximately 30% coverage of the endpoint value range, following the removal of outliers. Normalised PIs resulted in similar coverage on average, with a number of individual PIs exceeding the endpoint measurement range for datasets 4, 7, 8, 9, 10. However, the large coverage is a weakness of the underlying models for datasets 7, 8, 9 and 10 which are less accurate and are associated with high experimental error. The presence of compounds in the holdout data that are less conforming to the calibration data, where conformity was determined by the error models, combined with the low accuracy of the underlying model is the reason for exceedingly large individual PIs. Overall, the normalised PIs obtained were not useful as quantitative estimates of prediction uncertainty for all datasets but they could be used in compound prioritisation to highlight the need that further sampling or testing is required for similar compounds or within a range of endpoint values so that the models' uncertainties will be reduced.

It is noted that datasets that contain prediction error outliers may produce large PI estimates at high confidence levels if these are sampled in the calibration data. Although these will improve the validity of the ACPs, as larger PIs are more likely to include future measurements, they will also reduce the efficiency of PIs. However, statistical outliers are often defined at higher percentiles than the 80th, therefore, selecting a lower confidence threshold avoids this problem. Also, as the prediction error distribution of the future data is unknown and may contain large prediction errors; it would not be beneficial to remove prediction error outliers from the calibration data. Retaining them is expected to produce more conservative PIs that are more likely to include the future measurement. Outlier removal under different outlier definitions, i.e., descriptor, response, error outliers, and thresholds are topics that could be investigated in future work.

7.5 Conclusions

The CP framework was implemented to evaluate the utility of error models in the estimation of compound-specific PI estimates for QSAR predictions. From the optimisation of the ACPs it was found that the size of the ACPs had less influence on the efficiency of the PIs than normalisation and that efficient enough PIs could be obtained from small ACPs with a size of 10. It also became clear that cross-subsampling was sensitive to the ACP size, which implicitly affected the size of the calibration samples; while bootstrap and random sampling resulted in equally efficient PI estimates. The investigations showed that using a single approach for the optimisation of conformal predictors did not yield useful PIs for all datasets; but this was, mainly, because of the varied accuracy of underlying models rather than the normalisation method. It is therefore expected that more informative PIs will be estimated for models with high accuracy, which are obtained from datasets with low experimental error. Underlying models with low accuracy produced PIs that were too broad to be useful but some improvement in the efficiency of their average PIs was observed with normalisation.

Correlation of the normalised PI estimates and the prediction errors was not always associated with improvements in the efficiency of non-normalised PIs. The correlation of normalised PIs and the actual prediction errors was weak for most datasets; with stronger correlations identified in holdout data where residual error outliers were present. The high correlation indicated that predictions with high uncertainty could be identified or at least extracted by ranking the PIs of the holdout data. Normalisation using AD-based RF error models and the descriptor-based SVM models was more effective for the prioritisation of RF predictions than descriptor-based

RF error models. However, both descriptor-based error models produced more efficient PIs, while the normalised PIs using AD-based RF error models were less efficient than non-normalised in most occasions. Further investigation is required to better understand how to improve the accuracy of the uncertainty estimates of error models so that the size of the PIs provide a more accurate representation of the actual prediction error.

Chapter 8 Conclusions

8.1 Summary of Findings

This thesis has investigated the application of machine learning algorithms for the estimation of prediction uncertainty in ADME regression models and has assessed their performance as methods for assigning confidence to individual predictions. The use of supervised learning methods to construct error models represents an important opportunity for the systematic definition of a model's applicability domain, as error models facilitate the investigation of non-linear relationships between the underlying data structure or other reliability indicators and the accuracy of the models' predictions. Nevertheless, error models that were predictive for ADME data were challenging to obtain in this work but analysis of the correlations of the error estimates with the prediction errors was enough to confirm that, in fact, error models with poor performance can contribute some useful information to the QSAR modelling process.

In Chapter 5, underlying regression models were built for LogD and ADME data sets using state-of-the-art machine learning algorithms. Validation of the LogD models' performance using range-based and distance-to-model definitions of the applicability domain showed that only few of the algorithms' prediction errors could be explained. Underlying models created using RF and SVM algorithms showed performance across all datasets with R^2 values in the range of 0.15 – 0.61 and normalised RMSE in the range of 0.056 - 0.268. For the ADME datasets, analysis of the residual error distributions revealed that ADME datasets may contain measurement bias, which results in models with skewed error distributions. This is an important observation as a common assumption for confidence estimation is that model errors are randomly and normally distributed.

The performance of regression error models for the estimation of the underlying models' prediction errors was then investigated in Chapter 6. RF error models were trained on the cross-validation errors of the RF and SVM models using both the underlying molecular descriptors and then applicability domain-based reliability estimates as features. Evaluation using standard measures of regression analysis suggested that the error models were not predictive on holdout data and that alternative variables or techniques should be explored for the purpose of prediction error estimation. The use of applicability-domain based descriptors yielded surprisingly better performance in the estimation of RF prediction errors measured by the R^2 on cross-validation data but larger RMSE values. However, this was attributed to the use of the standard deviation of RF predictions as a descriptor in the error model, which correlates well with the prediction errors of RF. The ability of the error models to rank predictions based on their true accuracy was then investigated. A higher correlation between the error estimates and the actual prediction errors indicated that the error distributions were skewed as a result of measurement bias in the data. Yet, this was only apparent for data with low experimental error that were easier to model. Higher correlations between the error estimates and the actual errors were observed in more diverse datasets; which suggested that the error models may be useful in identifying compounds that are different to the modelled data or underrepresented in the measured endpoint value range. When statistical thresholds were applied, it was confirmed that error estimates with higher correlation to the prediction errors were more effective in identifying poorly predicted compounds.

The magnitude of the error estimates was assessed by applying an information theoretic approach; whereby measurements and predictions were represented as Gaussian probability distributions. This approach facilitated the evaluation of the individual prediction error estimates while taking into account the actual error and the experimental error of the assay. The mean Kullback-Leibler divergence (KLD) scores calculated using the estimates of each error model were used to assess the average overlap of the paired distributions. The scores suggested that the estimates of the error models resulted in higher overlap of the prediction and measurement distributions than estimates based on the standard deviation of the ensemble predictions; yet, there was less overlap in the majority of the cases than when assigning a uniform estimate based on average model error. The KLD distributions calculated for each error model also suggested that for all ADME models there was overlap in more than 80% of the holdout data. These results suggested that the error estimates provided a large enough error margin for the calculation of prediction intervals. However, the lack of robustness in the error models and the poor correlation of the error estimates and the actual prediction errors did not justify this.

In Chapter 7, conformal prediction was applied to estimate the prediction intervals (PIs) for the underlying RF and SVM models from the error distribution of calibration data. Aggregate conformal predictors (ACPs) that applied RF and SVM error models for normalisation were used to estimate compound-specific prediction intervals. Several ACP parameters were optimised to produce PIs for different datasets, such as the ACP size, the sampling technique and the error normalisation method; yet, they were found to be less important than the experimental error of the data and the accuracy of the underlying model when assessing the overall utility of the PIs. The different sampling techniques and error models used for normalisation of the ACPs showed success across several datasets but no single approach consistently emerged as best across all datasets or modelling methods. It was concluded that, an ACP would have to be optimised across various alternative normalisation methods to generate compound-specific PIs for a new test. However, RF error models are a flexible alternative as they facilitate the investigation of different types of features using the same algorithm. It was seen that the utility of the prediction intervals is mainly determined by the underlying models' error, therefore, only models with high accuracy are likely to produce more efficient prediction intervals. Useful intervals were defined by applying a prediction interval size 3-fold the experimental error at the set confidence level. This meant that the PIs for models with 15% error or less qualified, and most importantly that useful PIs could not be obtained for models of endpoints with assay error that exceeded 10% of the response range.

A more realistic application of conformal prediction involved the use of ACP for the estimation of uncertainty in the predictions of temporal holdout data. Large error rates of the ACP were an indication that the holdout data were less conforming to the calibration data and, effectively, the modelled data. As a result, uniform PIs estimated by ACP were not valid across all confidence levels with the exception of the PIs for the models of two smaller datasets that were generally less diverse in descriptor space and more likely to conform to the modelled data.

Analysis of the correlation between the error estimates and the actual prediction errors suggested that the error models may be useful for the identification of compounds that are underrepresented in ADME datasets due to measurement bias. Assessment of the error models as methods for the estimation of prediction intervals was evaluated by applying an information theoretic approach and conformal prediction. The utility of prediction intervals estimates is limited, however, particularly for models of ADME data with lower accuracy.

Further to the limitation of the error models' poor performance; the validation of machine learning algorithms for error modelling outside the conformal prediction framework can be rather complicated, particularly if the propagation of the errors of both the underlying QSAR and the error model need to be taken into account. With regards to the methods applied for the evaluation of the error models the data are also subject to several requirements. First of all, an estimate of experimental error is required to evaluate the utility of error model estimates as prediction intervals, i.e., using KLD scores, including the PI estimates derived using conformal prediction. Information on the experimental error of the data or single-assay ADME datasets with repeated measurements are not easily available in public databases. Finally, the validity of the results by ACP is guaranteed for data that are exchangeable. This means that the estimated PIs for external data may not be valid if the test data are not representative of the modelled data.

8.2 Suggestions for Future Investigations

The availability of large, single-assay ADME datasets in public databases is fairly limited to allow a large scale study to confirm the performance of error models. However, this could be done using available datasets of small to medium size in the PHYSPROP and CHEMBL databases with available experimental error estimates. There are also many other descriptors that could be utilised in the underlying models, including calculated estimates of other physicochemical properties. It could then be assessed whether the error in the calculated descriptors may be used to estimate the prediction error of ADME models' using linear methods and error propagation.

Further investigation could be carried out on the use of statistical divergence measures as significance tests, whereby the evaluation of the models' performance may account for the uncertainties of measurements and predictions. The mathematical properties of alternative measures to the KLD metric may facilitate the definition of threshold values that are universal, i.e. they can be applied to all datasets.

Investigations could also be conducted in conformal prediction with respect to sampling of the calibration data. Unlike aggregate conformal prediction which repeatedly samples calibration data from the training set to estimate the model's average distribution, it would be interesting to assess the validity and the efficiency of aggregate conformal predictors when the calibration data are sampled from outside the training set, i.e., from a larger population. This could help understand when the exchangeability assumption of the conformal prediction framework is invalidated and how this may be assessed prior to the estimation of PIs.

Alternative methods for the estimation of PIs may also be compared to the PI estimates of conformal prediction. Several methods are available for the estimation of PIs of RF models and are based on the use of quantile regression RF, a variant of the original RF algorithm, which predicts the quantiles of individual predictions rather the mean. Two known methods have been discussed by (Meinshausen, 2006) and (Zhang et al., 2019).

References

Abdi, H. (2010). Partial least squares regression and projection on latent structure regression (PLS Regression) Wiley Interdisciplinary Reviews: Computational Statistics Volume 2, Issue 1. *Wiley Interdisciplinary Reviews: Computational Statistics*, *2*(1), 97–106. https://doi.org/10.1002/wics.051

Ahmed, L., Georgiev, V., Capuccini, M., Toor, S., Schaal, W., Laure, E., & Spjuth, O. (2018). Efficient iterative virtual screening with Apache Spark and conformal prediction. *Journal of Cheminformatics*, *10*(1), 8. https://doi.org/10.1186/s13321-018-0265-z

Akarachantachote, N., Chadcham, S., & Saithanu, K. (2014). Cutoff threshold of Variable Importance in Projection for variable selection. *International Journal of Pure and Apllied Mathematics*, *94*(3), 307–322. https://doi.org/10.12732/ijpam.v94i3.2

Alexander, D. L. J., Tropsha, A., & Winkler, D. A. (2015). Beware of R^2: Simple, Unambiguous Assessment of the Prediction Accuracy of QSAR and QSPR Models. *Journal of Chemical Information and Modeling*, *55*(7), 1316–1322. https://doi.org/10.1021/acs.jcim.5b00206

Altman, N. S. (1992). An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. *The American Statistician*, *46*(3), 175–185.

Anava, O., & Levy, K. Y. (2016). k*-Nearest Neighbors: From Global to Local. *NIPS'16 Proceedings of the 30th International Conference on Neural Information Processing Systems*, 4923–4931. Retrieved from http://arxiv.org/abs/1701.07266

Bajorath, J. (2018). Foundations of data-driven medicinal chemistry. *Future Science*, *4*(8), 57–60.

Balaban, A. T. (1995). Chemical Graphs: Looking Back and Glimpsing Ahead. *Journal of Chemical Information and Computer Sciences*, *35*(3), 339–350. https://doi.org/10.1021/ci00025a001

Baumann, D., & Baumann, K. (2014). Reliable estimation of prediction errors for QSAR models under model uncertainty using double cross-validation. *Journal of Cheminformatics*, *6*(1), 47. https://doi.org/10.1186/s13321-014-0047-1

Bender, A., & Glen, R. C. (2004). Molecular similarity: A key technique in molecular informatics. *Organic and*

Biomolecular Chemistry, *2*(22), 3204–3218. https://doi.org/10.1039/b409813g

Bland, J. M., & Altman, D. G. (2003). Applying the right statistics: analyses of measurement studies. *Ultrasound in Obstetrics and Gynecology*, *22*(1), 85–93.

Carhart, R. E., Smith, D. H., & Venkataraghavan, R. (1985). Atom pairs as molecular features in structure-activity studies: definition and applications. *Journal of Chemical Information and Computer Sciences*, *25*(2), 64–73. https://doi.org/10.1021/ci00046a002

Carrió, P., Pinto, M., Ecker, G., Sanz, F., & Pastor, M. (2014). Applicability domain analysis (ADAN): A robust method for assessing the reliability of drug property predictions. *Journal of Chemical Information and Modeling*, *54*(5), 1500–1511. https://doi.org/10.1021/ci500172z

Cereto-Massagué, A., Ojeda, M. J., Valls, C., Mulero, M., Garcia-Vallvé, S., & Pujadas, G. (2015). Molecular fingerprint similarity search in virtual screening. *Methods*, *71*, 58–63. https://doi.org/10.1016/j.ymeth.2014.08.005

Chirico, N., & Gramatica, P. (2011). Real external predictivity of QSAR models: How to evaluate It? Comparison of different validation criteria and proposal of using the concordance correlation coefficient. *Journal of Chemical Information and Modeling*, *51*(9), 2320–2335. https://doi.org/10.1021/ci200211n

Chomboon, K., Chujai, P., Teerarassammee, P., Kerdprasop, K., & Kerdprasop, N. (2015). An Empirical Study of Distance Metrics for k-Nearest Neighbor Algorithm. *Institute of Industrial Applications Engineers*, 280–285. https://doi.org/10.12792/iciae2015.051

Chrysostomou, K., Chen, S. Y., & Liu, X. (2008). Combining multiple classifiers for wrapper feature selection. *International Journal of Data Mining, Modelling and Management*, *1*(1), 91–102. https://doi.org/10.1504/IJDMMM.2008.022539

Consonni, V., Ballabio, D., & Todeschini, R. (2009). Comments on the Definition of the Q 2 Parameter for QSAR Validation. *Journal of Chemical Information and Modeling*, *49*(7), 1669–1678. https://doi.org/10.1021/ci900115y

Consonni, V., Ballabio, D., & Todeschini, R. (2010). Evaluation of model predictive ability by external validation techniques. *Journal of Chemometrics*, *24*(3–4), 194–201. https://doi.org/10.1002/cem.1290

Cortes-Ciriano, I. (2016). Benchmarking the Predictive Power of Ligand Efficiency Indices in QSAR. *Journal of Chemical Information and Modeling*, *56*(8), 1576–1587. https://doi.org/10.1021/acs.jcim.6b00136

Cortes-Ciriano, I., & Bender, A. (2019). Deep Confidence: A Computationally Efficient Framework for

Calculating Reliable Prediction Errors for Deep Neural Networks. *Journal of Chemical Information and Modeling*, *59*(3), 1269–1281. https://doi.org/10.1021/acs.jcim.8b00542

Cortes-Ciriano, I., Bender, A., & Malliavin, T. (2015). Prediction of PARP Inhibition with Proteochemometric Modelling and Conformal Prediction. *Molecular Informatics*, *34*(6–7), 357–366. https://doi.org/10.1002/minf.201400165

Cruz-Monteagudo, M., Medina-Franco, J. L., Pérez-Castillo, Y., Nicolotti, O., Cordeiro, M. N. D. S., & Borges, F. (2014). Activity cliffs in drug discovery: Dr Jekyll or Mr Hyde? *Drug Discovery Today*, *19*(8), 1069–1080. https://doi.org/10.1016/j.drudis.2014.02.003

Danishuddin, & Khan, A. U. (2016). Descriptors and their selection methods in QSAR analysis: paradigm for drug design. *Drug Discovery Today*, *21*(8), 1291–1302. https://doi.org/10.1016/j.drudis.2016.06.013

Dashevskiy, M., & Luo, Z. (2008). Network traffic demand prediction with confidence. *GLOBECOM - IEEE Global Telecommunications Conference*, (May), 1453–1457. https://doi.org/10.1109/GLOCOM.2008.ECP.284

Daylight Theory Manual. (2011). Retrieved December 4, 2015, from https://www.daylight.com/dayhtml/doc/theory/index.html

Dearden, J. C., Cronin, M. T. D., & Kaiser, K. L. E. (2009). How not to develop a quantitative structure-activity or structure-property relationship (QSAR/QSPR). *SAR and QSAR in Environmental Research*, *20*(3–4), 241–266. https://doi.org/10.1080/10629360902949567

DiMasi, J. A., Grabowski, H. G., & Hansen, R. W. (2016). Innovation in the pharmaceutical industry: New estimates of R&D costs. *Journal of Health Economics*, *47*, 20–33. https://doi.org/10.1016/j.jhealeco.2016.01.012

Dragos, H., Gilles, M., & Alexandre, V. (2009). Predicting the predictability: A unified approach to the applicability domain problem of qsar models. *Journal of Chemical Information and Modeling*, *49*(7), 1762–1776. https://doi.org/10.1021/ci9000579

Duffy, B. C., Zhu, L., Decornez, H., & Kitchen, D. B. (2012). Bioorganic & Medicinal Chemistry Early phase drug discovery : Cheminformatics and computational techniques in identifying lead series. *Bioorganic & Medicinal Chemistry*, *20*(18), 5324–5342. https://doi.org/10.1016/j.bmc.2012.04.062

Durant, J. L., Leland, B. A., Henry, D. R., Nourse, J. G., L., D. J., Leland, B. A., … Nourse, J. G. (2002). Reoptimization of MDL keys for use in Drug Discovery. *J Chem Inf Comput Sci*, *42*, 1273–1280.

Ekins, S., Puhl, A. C., Zorn, K. M., Lane, T. R., Russo, D. P., Klein, J. J., … Clark, A. M. (2019). Exploiting machine learning for end-to-end drug discovery and development. *Nature Materials*, *18*, 435–441. https://doi.org/10.1038/s41563-019-0338-z

Eklund, M., Norinder, U., Boyer, S., & Carlsson, L. (2012). Application of conformal prediction in QSAR (L. I. et al., Ed.). *IFIP Advances in Information and Communication Technology*, pp. 166–175. https://doi.org/10.1007/978-3-642-33412-2_17

Eklund, M., Norinder, U., Boyer, S., & Carlsson, L. (2015). The application of conformal prediction to the drug discovery process. *Annals of Mathematics and Artificial Intelligence*, *74*(1–2), 117–132. https://doi.org/10.1007/s10472-013-9378-2

Eliades, C., & Papadopoulos, H. (2017). Conformal Prediction for Automatic Face Recognition. *Proceedings of Machine Learning Research*, *60*, 62–81.

Engel, T. (2006). Basic overview of chemoinformatics. *Journal of Chemical Information and Modeling*, *46*(6), 2267–2277. https://doi.org/10.1021/ci600234z

Eriksson, L., Jaworska, J., Worth, A. P., Cronin, M. T. D. D., McDowell, R. M., & Gramatica, P. (2003). Methods for reliability and uncertainty assessment and for applicability evaluations of classification- and regression-based QSARs. *Environmental Health Perspectives*, *111*(10), 1361–1375. https://doi.org/10.1289/ehp.5758

Feng, D., Svetnik, V., Liaw, A., Pratola, M., & Sheridan, R. P. (2019). Building Quantitative Structure-Activity Relationship Models Using Bayesian Additive Regression Trees. *Journal of Chemical Information and Modeling*, *59*(6), 2642–2655. https://doi.org/10.1021/acs.jcim.9b00094

Filzmoser, P., Liebmann, B., & Varmuza, K. (2009). Repeated double cross validation. *Journal of Chemometrics*, *23*(4), 160–171. https://doi.org/10.1002/cem.1225

Firdaus Begam, B., & Satheesh Kumar, J. (2012). A study on cheminformatics and its applications on modern drug discovery. *Procedia Engineering*, *38*, 1264–1275. https://doi.org/10.1016/j.proeng.2012.06.156

Free, S. M., & Wilson, J. W. (1964). A Mathematical Contribution to Structure-Activity Studies. *Journal of Medicinal Chemistry*, *7*(4), 395–399. https://doi.org/10.1021/jm00334a001

Fujita, T., & Winkler, D. A. (2016). Understanding the Roles of the “two QSARs.” *Journal of Chemical Information and Modeling*, *56*(2), 269–274. https://doi.org/10.1021/acs.jcim.5b00229

Gadaleta, D., Mangiatordi, G. F., Catto, M., Carotti, A., & Nicolotti, O. (2016). Applicability Domain for QSAR Models. *International Journal of Quantitative Structure-Property Relationships*, *1*(1), 45–63.

https://doi.org/10.4018/ijqspr.2016010102

Gaspar, H. A., Baskin, I. I., & Varnek, A. (2016). *Visualization of a Multidimensional Descriptor Space*. *1222*, 243–267. https://doi.org/10.1021/bk-2016-1222.ch012

Glen, R. C., Bender, A., Arnby, C. H., Carlsson, L., Boyer, S., & Smith, J. (2006). Circular fingerprints: Flexible molecular descriptors with applications from physical chemistry to ADME. *IDrugs*, *9*(3), 199–204.

Golbraikh, A., & Tropsha, A. (2002a). Beware of q2! *Journal of Molecular Graphics and Modelling*, *20*(4), 269–276. https://doi.org/10.1016/S1093-3263(01)00123-1

Golbraikh, A., & Tropsha, A. (2002b). Predictive QSAR modeling based on diversity sampling of experimental datasets for the training and test set selection. *Molecular Diversity*, *5*(4), 231–243. https://doi.org/10.1023/A:1021372108686

Guha, R., & Jurs, P. C. (2004). Development of QSAR models to predict and interpret the biological activity of artemisinin analogues. *Journal of Chemical Information and Computer Sciences*, *44*(4), 1440–1449. https://doi.org/10.1021/ci0499469

Haas, J. V, Eastwood, B. J., Iversen, P. W., & al., et. (2013). Minimum Significant Ratio – A Statistic to Assess Assay Variability. In G. S. Sittampalam, N. P. Coussens, K. Brimacombe, & et al. (Eds.), *Assay Guidance Manual*. Retrieved from https://www.ncbi.nlm.nih.gov/books/NBK169432/

Hansch, C. (1969). A Quantitative Approach to Biochemical Structure-Activity Relationships. *Accounts of Chemical Research*, *2*(8), 232–239.

Hanser, T., Barber, C., Marchaland, J. F., & Werner, S. (2016). Applicability domain: towards a more formal definition. *SAR and QSAR in Environmental Research*, *27*(11), 865–881. https://doi.org/10.1080/1062936X.2016.1250229

Hawkins, D. M. (2004). The Problem of Overfitting. *Journal of Chemical Information and Computer Sciences*, *44*(1), 1–12. https://doi.org/10.1021/ci0342472

Heller, S. R., McNaught, A., Pletnev, I., Stein, S., & Tchekhovskoi, D. (2015). InChI, the IUPAC International Chemical Identifier. *Journal of Cheminformatics*, *7*(1), 1–34. https://doi.org/10.1186/s13321-015-0068-4

Hou, T. J., & Xu, X. J. (2003). ADME Evaluation in Drug Discovery. 2. Prediction of Partition Coefficient by Atom-Additive Approach Based on Atom-Weighted Solvent Accessible Surface Areas. *J. Chem. Inf. Comput. Sci*, (43), 1058–1067.

Ivanciuc, O. (2007). Applications of Support Vector Machines in Chemistry. In K. B. Lipkowitz & T. R. Cundari

(Eds.), *Reviews in Computational Chemistry* (pp. 291–400). Weinheim: Wiley-VCH.

Jaworska, J., Nikolova-Jeliazkova, N., & Aldenberg, T. (2005). QSAR applicability domain estimation by projection of the training set in descriptor space: A review. *ATLA Alternatives to Laboratory Animals*, *33*(5), 445–459. https://doi.org/10.1177/026119290503300508

Jindal, P., & Kumar, D. (2017). A Review on Dimensionality Reduction Techniques. *International Journal of Computer Applications*, *173*(2), 42–46. https://doi.org/10.5120/ijca2017915260

Johansson, U., Boström, H., Löfström, T., & Linusson, H. (2014). Regression conformal prediction with random forests. *Machine Learning*, *97*(1–2), 155–176. https://doi.org/10.1007/s10994-014-5453-0

Johnson, M., Basak, S., & Maggiora, G. (1988). A characterization of molecular similarity methods for property prediction. *Mathematical and Computer Modelling*, *11*(C), 630–634. https://doi.org/10.1016/0895-7177(88)90569-9

Kaneko, H. (2018). Discussion on Regression Methods Based on Ensemble Learning and Applicability Domains of Linear Submodels. *Journal of Chemical Information and Modeling*, *58*(2), 480–489. https://doi.org/10.1021/acs.jcim.7b00649

Kaneko, H., & Funatsu, K. (2014). Applicability domain based on ensemble learning in classification and regression analyses. *Journal of Chemical Information and Modeling*, *54*(9), 2469–2482. https://doi.org/10.1021/ci500364e

Katritzky, A. R., & Gordeeva, E. V. (1993). Traditional Topological Indices vs Electronic, Geometrical, and Combined Molecular Descriptors in QSAR/QSPR Research. *Journal of Chemical Information and Computer Sciences*, *33*(6), 835–857. https://doi.org/10.1021/ci00016a005

Keefer, C. E., Kauffman, G. W., & Gupta, R. R. (2013). Interpretable, probability-based confidence metric for continuous quantitative structure-activity relationship models. *Journal of Chemical Information and Modeling*, *53*(2), 368–383. https://doi.org/10.1021/ci300554t

Klingspohn, W., Mathea, M., Ter Laak, A., Heinrich, N., Baumann, K., Laak, A., … Baumann, K. (2017). Efficiency of different measures for defining the applicability domain of classification models. *Journal of Cheminformatics*, *9*(1), 1–17. https://doi.org/10.1186/s13321-017-0230-2

Krstajic, D., Buturovic, L. J., Leahy, D. E., & Thomas, S. (2014). Cross-validation pitfalls when selecting and assessing regression and classification models. *Journal of Cheminformatics*, *6*(1), 1–15. https://doi.org/10.1186/1758-2946-6-10

Kubinyi, H. (1993a). *QSAR: Hansch Analysis and Related Approaches* (R. Mannhold, P. Krogsgaard-Larsen, & H. Timmerman, Eds.). https://doi.org/10.1360/zd-2013-43-6-1064

Kubinyi, H. (1993b). The Additive Model (Free Wilson Analysis). In R. Mannhold, P. Krogsgaard-Larsen, & H. Timmerman (Eds.), *QSAR: Hansch analysis and related approaches* (Vol. 1, pp. 62–65). https://doi.org/10.1016/s0165-6147(00)89046-x

Kubinyi, H. (2002). From Narcosis to Hyperspace: The History of QSAR. *Quantitative Structure-Activity Relationships*, *21*(4), 348–356. https://doi.org/10.1002/1521-3838(200210)21:4<348::AID-QSAR348>3.0.CO;2-D

Kümmel, A., Bonate, P. L., Dingemanse, J., & Krause, A. (2018). Confidence and Prediction Intervals for Pharmacometric Models. *CPT: Pharmacometrics and Systems Pharmacology*, *7*(6), 360–373. https://doi.org/10.1002/psp4.12286

Kvalseth, T. O. (1985). *Cautionary Note about R2 Published by : Taylor & Francis , Ltd . on behalf of the American Statistical Association Stable URL : https://www.jstor.org/stable/2683704 Cautionary Note About R2*. *39*(4), 279–285.

Labute, P. (2000). A widely applicable set of descriptors. *Journal of Molecular Graphics and Modelling*, *18*(4–5), 464–477. https://doi.org/10.1016/S1093-3263(00)00068-1

Lambrou, A., Papadopoulos, H., Kyriacou, E., Pattichis, C. S., Pattichis, M. S., Gammerman, A., & Nicolaides, A. (2010). Assessment of stroke risk based on morphological ultrasound image analysis with conformal prediction. *IFIP Advances in Information and Communication Technology*, *339 AICT*, 146–153. https://doi.org/10.1007/978-3-642-16239-8_21

Lapins, M., Arvidsson, S., Lampa, S., Berg, A., Schaal, W., Alvarsson, J., & Spjuth, O. (2018). A confidence predictor for logD using conformal regression and a support-vector machine. *Journal of Cheminformatics*. https://doi.org/10.1186/s13321-018-0271-1

Lavecchia, A. (2015). Machine-learning approaches in drug discovery: Methods and applications. *Drug Discovery Today*, Vol. 20, pp. 318–331. https://doi.org/10.1016/j.drudis.2014.10.012

Lavecchia, A., & Di Giovanni, C. (2013). *Virtual Screening Strategies in Drug Discovery : A Critical Review*. 2839–2860.

Leach, A. R., & Gillet, V. J. (2007). An introduction to chemoinformatics. In *An Introduction To Chemoinformatics*. https://doi.org/10.1007/978-1-4020-6291-9

Linusson, H. (2017). *An Introduction To Conformal Prediction*. Retrieved from http://clrc.rhul.ac.uk/copa2017/presentations/CP_Tutorial_2017.pdf

Ma, J., Sheridan, R. P., Liaw, A., Dahl, G. E., & Svetnik, V. (2015). Deep neural nets as a method for quantitative structure-activity relationships. *Journal of Chemical Information and Modeling*, *55*(2), 263–274. https://doi.org/10.1021/ci500747n

Maggiora, G. (2006). On Outliers and Activity Cliffs Why QSAR Often Disappoints. *Journal of Chemical Information and Modeling*, *46*(4), 1535. https://doi.org/10.1021/ci060117s

Maggiora, G., Vogt, M., Stumpfe, D., & Bajorath, J. (2014). Molecular similarity in medicinal chemistry. *Journal of Medicinal Chemistry*, *57*(8), 3186–3204. https://doi.org/10.1021/jm401411z

Mahé, P. (2006). *Kernel functions for molecular structures and their application to virtual screening with Support Vector Machines* (Doctoral dissertation, École Nationale Supérieure des Mines de Paris, Paris, France). Retrieved from https://pastel.archives-ouvertes.fr/pastel-00002191

Mannhold, R., Poda, G. I., Ostermann, C., & Tetko, I. V. (2009). Calculation of molecular lipophilicity: State-of-the-art and comparison of log P methods on more than 96,000 compounds. *Journal of Pharmaceutical Sciences*, *98*(3), 861–893. https://doi.org/10.1002/jps.21494

Martin, T. M., Harten, P., Young, D. M., Muratov, E. N., Golbraikh, A., Zhu, H., & Tropsha, A. (2012). Does rational selection of training and test sets improve the outcome of QSAR modeling? *Journal of Chemical Information and Modeling*, *52*(10), 2570–2578. https://doi.org/10.1021/ci300338w

Mathea, M., Klingspohn, W., & Baumann, K. (2016). Chemoinformatic Classification Methods and their Applicability Domain. *Molecular Informatics*, *35*(5), 160–180. https://doi.org/10.1002/minf.201501019

Meinshausen, N. (2006). Quantile regression forests. *Journal of Machine Learning Research*, *7*, 983–999.

Menze, B. H., Kelm, B. M., Masuch, R., Himmelreich, U., Bachert, P., Petrich, W., & Hamprecht, F. A. (2009). A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC Bioinformatics*, *10*, 1–16. https://doi.org/10.1186/1471-2105-10-213

Mitchell, T. M. (1997). Instance-Based Learning. In E. M. Munson (Ed.), *Machine Learning* (pp. 233–234). McGraw-Hill Education.

Moriguchi, I., Hirono, S., Liu, Q., Nakagome, I., & Matsushita, Y. (1992). Simple Method of Calculating Octanol/Water Partition Coefficient. *CHEMICAL & PHARMACEUTICAL BULLETIN*, *40*(1), 127–130.

https://doi.org/10.1248/cpb.40.127

Netzeva, T. I., Worth, A., Aldenberg, T., Benigni, R., Cronin, M. T. D., Gramatica, P., … Yang, C. (2005). Current status of methods for defining the applicability domain of (quantitative) structure-activity relationships. The report and recommendations of ECVAM Workshop 52. *Alternatives to Laboratory Animals : ATLA*, *33*(2), 155–173. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/16180989

Nguyen, K. T., Blum, L. C., Van Deursen, R., & Reymond, J. L. (2009). Classification of organic molecules by molecular quantum numbers. *ChemMedChem*, *4*(11), 1803–1805. https://doi.org/10.1002/cmdc.200900317

Nigsch, F., Bender, A., Van Buuren, B., Tissen, J., Nigsch, E., & Mitchell, J. B. O. (2006). Melting point prediction employing k-nearest neighbor algorithms and genetic parameter optimization. *Journal of Chemical Information and Modeling*, *46*(6), 2412–2422. https://doi.org/10.1021/ci060149f

Nikolova, N., & Jaworska, J. (2003). Approaches to Measure Chemical Similarity - A Review. *QSAR and Combinatorial Science*, *22*(9–10), 1006–1026. https://doi.org/10.1002/qsar.200330831

Norinder, U., Carlsson, L., Boyer, S., & Eklund, M. (2015). Introducing conformal prediction in predictive modeling for regulatory purposes. A transparent and flexible alternative to applicability domain determination. *Regulatory Toxicology and Pharmacology*, *71*(2), 279–284. https://doi.org/10.1016/j.yrtph.2014.12.021

Norinder, U., Rybacka, A., & Andersson, P. L. (2016). Conformal prediction to define applicability domain – A case study on predicting ER and AR binding. *SAR and QSAR in Environmental Research*, *27*(4), 303–316. https://doi.org/10.1080/1062936X.2016.1172665

Nouretdinov, I., Gammerman, A., Qi, Y., & Klein-Seetharaman, J. (2012). Determining confidence of predicted interactions between HIV-1 and human proteins using conformal method. *Pacific Symposium on Biocomputing*, 311–322. https://doi.org/10.1142/9789814366496_0030

Papa, E., Kovarich, S., & Gramatica, P. (2009). Development, Validation and Inspection of the Applicability Domain of QSPR Models for Physicochemical Properties of Polybrominated Diphenyl Ethers. *QSAR & Combinatorial Science*, *28*(8), 790–796. https://doi.org/10.1002/qsar.200860183

Papadopoulos, H., Gammerman, A., & Vovk, V. (2009). Reliable diagnosis of acute abdominal pain with conformal prediction. *Engineering Intelligent Systems*, *17*(2–3), 127–137.

Papadopoulos, H., Vovk, V., & Gammerman, A. (2011). Regression conformal prediction with nearest neighbours. *Journal of Artificial Intelligence Research*, *40*, 815–840. https://doi.org/10.1613/jair.3198

Powell, E. C. (2000). A History of Chemical Abstracts Service , 1907-1998. *Science & Technology Libraries*, *18*(4), 93–110. https://doi.org/10.1300/J122v18n04

Raschka, S. (2018). *Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning*. Retrieved from http://arxiv.org/abs/1811.12808

Rasmussen, B. P., & Hines, J. W. (2003). Prediction Interval Estimation Techniques for Empirical Modeling Strategies and their Applications to Signal Validation Tasks. *Nuclear Engineering Department*, *Ph.D*, 366.

Robnik-Šikonja, M., & Kononenko, I. (2003). Theoretical and Empirical Analysis of ReliefF and RReliefF. *Machine Learning*. https://doi.org/10.1023/A:1025667309714

Rogers, D., & Hahn, M. (2010). Extended-connectivity fingerprints. *Journal of Chemical Information and Modeling*, *50*, 742–754. https://doi.org/10.1021/ci100050t

Roy, K., Ambure, P., & Aher, R. B. (2017). How important is to detect systematic error in predictions and understand statistical applicability domain of QSAR models? *Chemometrics and Intelligent Laboratory Systems*, *162*(May 2016), 44–54. https://doi.org/10.1016/j.chemolab.2017.01.010

Roy, K., Mitra, I., Ojha, P. K., Kar, S., Das, R. N., & Kabir, H. (2012). Introduction of rm2(rank) metric incorporating rank-order predictions as an additional tool for validation of QSAR/QSPR models. *Chemometrics and Intelligent Laboratory Systems*, *118*, 200–210. https://doi.org/10.1016/j.chemolab.2012.06.004

Sahigara, F., Ballabio, D., Todeschini, R., & Consonni, V. (2014). Assessing the Validity of QSARs for Ready Biodegredability of Chemicals: An Applicability Domain Perspective. *Current Computer-Aided Drug Design*, *10*(2), 137–147. https://doi.org/10.2174/1573409910666140410110241

Sahigara, F., Mansouri, K., Ballabio, D., Mauri, A., Consonni, V., & Todeschini, R. (2012). Comparison of different approaches to define the applicability domain of QSAR models. *Molecules*, *17*(5), 4791–4810. https://doi.org/10.3390/molecules17054791

Sahlin, U. (2015). Assessment of uncertainty in chemical models by Bayesian probabilities: Why, when, how? *Journal of Computer-Aided Molecular Design*, *29*(7), 583–594. https://doi.org/10.1007/s10822-014-9822-3

Sahlin, U., Jeliazkova, N., & Oberg, T. (2014). Applicability domain dependent predictive uncertainty in QSAR regressions. *Molecular Informatics*, *33*(1), 26–35. https://doi.org/10.1002/minf.201200131

Schüürmann, G., Ebert, R. U., Chen, J., Wang, B., & Kühne, R. (2008). External validation and prediction employing the predictive squared correlation coefficient - Test set activity mean vs training set activity mean.

Journal of Chemical Information and Modeling, *48*(11), 2140–2145. https://doi.org/10.1021/ci800253u

Sheridan, R. P. (2012). Three useful dimensions for domain applicability in QSAR models using random forest. *Journal of Chemical Information and Modeling*, *52*(3), 814–823. https://doi.org/10.1021/ci300004n

Sheridan, R. P. (2013a). Time-split cross-validation as a method for estimating the goodness of prospective prediction. *Journal of Chemical Information and Modeling*, *53*(4), 783–790. https://doi.org/10.1021/ci400084k

Sheridan, R. P. (2013b). Using random forest to model the domain applicability of another random forest model. *Journal of Chemical Information and Modeling*, *53*(11), 2837–2850. https://doi.org/10.1021/ci400482e

Sheridan, R. P. (2015). The Relative Importance of Domain Applicability Metrics for Estimating Prediction Errors in QSAR Varies with Training Set Diversity. *Journal of Chemical Information and Modeling*, *55*(6), 1098–1107. https://doi.org/10.1021/acs.jcim.5b00110

Sheridan, R. P., Feuston, B. P., Maiorov, V. N., & Kearsley, S. K. (2004). Similarity to molecules in the training set is a good discriminator for prediction accuracy in QSAR. *Journal of Chemical Information and Computer Sciences*, *44*(6), 1912–1928. https://doi.org/10.1021/ci049782w

Shi, L. M., Fang, H., Tong, W., Wu, J., Perkins, R., Blair, R. M., … Sheehan, D. M. (2001). QSAR Models Using a Large Diverse Set of Estrogens. *Journal of Chemical Information and Computer Sciences*, *41*(1), 186–195. https://doi.org/10.1021/ci000066d

Stanforth, R. W., Kolossov, E., & Mirkin, B. (2007). A measure of domain of applicability for QSAR modelling based on intelligent K-means clustering. *QSAR and Combinatorial Science*, *26*(7), 837–844. https://doi.org/10.1002/qsar.200630086

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., & Zeileis, A. (2008). Conditional variable importance for random forests. *BMC Bioinformatics*, *9*(1), 307. https://doi.org/10.1186/1471-2105-9-307

Stumpfe, D., Hu, Y., Dimova, D., & Bajorath, J. (2014). Recent progress in understanding activity cliffs and their utility in medicinal chemistry. *Journal of Medicinal Chemistry*, *57*(1), 18–28. https://doi.org/10.1021/jm401120g

Sun, J., Carlsson, L., Ahlberg, E., Norinder, U., Engkvist, O., & Chen, H. (2017). Applying Mondrian Cross-Conformal Prediction to Estimate Prediction Confidence on Large Imbalanced Bioactivity Data Sets. *Journal of Chemical Information and Modeling*, *57*(7), 1591–1598. https://doi.org/10.1021/acs.jcim.7b00159

Sushko, I., Novotarskyi, S., Körner, R., Pandey, A. K., Kovalishyn, V. V., Prokopenko, V. V., & Tetko, I. V. (2010). Applicability domain for in silico models to achieve accuracy of experimental measurements. *Journal of Chemometrics*, *24*(3–4), 202–208. https://doi.org/10.1002/cem.1296

Svensson, F., Aniceto, N., Norinder, U., Cortes-Ciriano, I., Spjuth, O., Carlsson, L., & Bender, A. (2018). Conformal Regression for Quantitative Structure-Activity Relationship Modeling - Quantifying Prediction Uncertainty. *Journal of Chemical Information and Modeling*, *58*(5), 1132–1140. https://doi.org/10.1021/acs.jcim.8b00054

Svensson, F., Norinder, U., & Bender, A. (2017). Improving Screening Efficiency through Iterative Screening Using Docking and Conformal Prediction. *Journal of Chemical Information and Modeling*, *57*(3), 439–444. https://doi.org/10.1021/acs.jcim.6b00532

Tetko, I. V., Bruneau, P., Mewes, H. W., Rohrer, D. C., & Poda, G. I. (2006). Can we estimate the accuracy of ADME-Tox predictions? *Drug Discovery Today*, *11*(15–16), 700–707. https://doi.org/10.1016/j.drudis.2006.06.013

Tetko, I. V, Sushko, I., Pandey, A. K., Zhu, H., Tropsha, A., Papa, E., … Varnek, A. (2008). Critical assessment of QSAR models of environmental toxicity against tetrahymena pyriformis: Focusing on applicability domain and overfitting by variable selection. *Journal of Chemical Information and Modeling*, *48*(9), 1733–1746. https://doi.org/10.1021/ci800151m

Thas, O. (2010a). Graphical Tools. In *Comparing Distributions* (pp. 49–75). https://doi.org/10.1007/978-0-387-92710-7_3

Thas, O. (2010b). Methods Based on the Empirical Distribution Function. In *Comparing Distributions* (pp. 123–160). https://doi.org/10.1007/978-0-387-92710-7_5

Todeschini, R., Ballabio, D., & Grisoni, F. (2016). Beware of Unreliable Q2! A Comparative Study of Regression Metrics for Predictivity Assessment of QSAR Models. *Journal of Chemical Information and Modeling*, *56*(10), 1905–1913. https://doi.org/10.1021/acs.jcim.6b00277

Todeschini, R., & Consonni, V. (2009). Molecular Descriptors for Chemoinformatics. In *Molecular Descriptors for Chemoinformatics* (2nd editio). https://doi.org/10.1002/9783527628766

Toplak, M., Močnik, R., Polajnar, M., Bosnić, Z., Carlsson, L., Hasselgren, C., … Staišlring, J. (2014). Assessment of machine learning reliability methods for quantifying the applicability domain of QSAR regression models. *Journal of Chemical Information and Modeling*, *54*(2), 431–441. https://doi.org/10.1021/ci4006595

Topliss, J. G., & Edwards, R. P. (1979). Chance Factors in Studies of Quantitative Structure-Activity Relationships. *Journal of Medicinal Chemistry*, *22*(10), 1238–1244. https://doi.org/10.1021/jm00196a017

Tran, T. N., Afanador, N. L., Buydens, L. M. C., & Blanchet, L. (2014). Interpretation of variable importance in Partial Least Squares with Significance Multivariate Correlation (sMC). *Chemometrics and Intelligent Laboratory Systems*, *138*(November), 153–160. https://doi.org/10.1016/j.chemolab.2014.08.005

Tropsha, A., Gramatica, P., & Gombar, V. K. (2003). The importance of being earnest: Validation is the absolute essential for successful application and interpretation of QSPR models. *QSAR and Combinatorial Science*, *22*(1), 69–77. https://doi.org/10.1002/qsar.200390007

van de Waterbeemd, H., & Gifford, E. (2003). ADMET in silico modelling: Towards prediction paradise? *Nature Reviews Drug Discovery*, Vol. 2, pp. 192–204. https://doi.org/10.1038/nrd1032

Varnek, A., & Baskin, I. (2012). Machine learning methods for property prediction in chemoinformatics: Quo Vadis? *Journal of Chemical Information and Modeling*, *52*(6), 1413–1437. https://doi.org/10.1021/ci200409x

Wang, J. B., Cao, D. S., Zhu, M. F., Yun, Y. H., Xiao, N., & Liang, Y. Z. (2015). In silico evaluation of logD7.4 and comparison with other prediction methods. *Journal of Chemometrics*, *29*(7), 389–398. https://doi.org/10.1002/cem.2718

Wechsler, H. (2015). Cyberspace Security Using Adversarial Learning and Conformal Prediction. *Intelligent Information Management*, *07*(04), 195–222. https://doi.org/10.4236/iim.2015.74016

Weininger, D. (1988). SMILES, a Chemical Language and Information System: 1: Introduction to Methodology and Encoding Rules. *Journal of Chemical Information and Computer Sciences*, *28*(1), 31–36. https://doi.org/10.1021/ci00057a005

Wenlock, M. C., & Carlsson, L. A. (2015). How experimental errors influence drug metabolism and pharmacokinetic QSAR/QSPR models. *Journal of Chemical Information and Modeling*, *55*(1), 125–134. https://doi.org/10.1021/ci500535s

Wild, D. J., & Blankley, C. J. (2000). Comparison of 2D Fingerprint Types and Hierarchy Level Selection Methods for Structural Grouping Using Ward's Clustering. *Journal of Chemical Information and Computer Sciences*, *40*(1), 155–162. https://doi.org/10.1021/ci990086j

Wildman, S. A., & Crippen, G. M. (1999). Prediction of physicochemical parameters by atomic contributions. *Journal of Chemical Information and Computer Sciences*, *39*(5), 868–873. https://doi.org/10.1021/ci990307l

Willett, P. (2008). From chemical documentation to chemoinformatics: 50 years of chemical information science. *Journal of Information Science*, *34*(4), 477–499. https://doi.org/10.1177/0165551507084631

Willink, R. (2012). Confidence intervals and other statistical intervals in metrology. *International Journal of Metrology and Quality Engineering*, *3*(3), 169–178. https://doi.org/10.1051/ijmqe/2012029

Winkler, D. A. (2002). The role of quantitative structure--activity relationships (QSAR) in biomolecular discovery. *Briefings in Bioinformatics*, *3*(1), 73–86. https://doi.org/10.1093/bib/3.1.73

Wishart, D. S., Feunang, Y. D., Guo, A. C., Lo, E. J., Marcu, A., Grant, J. R., … Wilson, M. (2018). DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Research*, *46*(D1), D1074–D1082. https://doi.org/10.1093/nar/gkx1037

Wood, D. J., Buttar, D., Cumming, J. G., Davis, A. M., Norinder, U., & Rodgers, S. L. (2011). Automated QSAR with a hierarchy of global and local models. *Molecular Informatics*, *30*(11–12), 960–972. https://doi.org/10.1002/minf.201100107

Wood, D. J., Carlsson, L., Eklund, M., Norinder, U., & Stålring, J. (2013). QSAR with experimental and predictive distributions: An information theoretic approach for assessing model quality. *Journal of Computer-Aided Molecular Design*, *27*(3), 203–219. https://doi.org/10.1007/s10822-013-9639-5

Worth, A. P., Bassan, A., Gallegos, A., Netzeva, T. ., Patlewicz, G., Pavan, M., … Vracko, M. (2005). The characterisation of (Quantitative) Structure-Activity Relationships: Preliminary guidance. In *ECB Report EUR 21866: European Commision, Joint Research Center*. Ispra.

Xu, J., & Hagler, A. (2002). Chemoinformatics and drug discovery. *Molecules*, *7*(8), 566–600. https://doi.org/10.3390/70800566

Yun, Y.-H. H., Wu, D.-M. M., Li, G.-Y. Y., Zhang, Q.-Y. Y., Yang, X., Li, Q.-F. F., … Xu, Q.-S. S. (2017). A strategy on the definition of applicability domain of model based on population analysis. *Chemometrics and Intelligent Laboratory Systems*, *170*(April), 77–83. https://doi.org/10.1016/j.chemolab.2017.09.007

Zhang, H., Zimmerman, J., Nettleton, D., & Nordman, D. J. (2019). Random Forest Prediction Intervals. *The American Statistician*, *0*(0), 1–15. https://doi.org/10.1080/00031305.2019.1585288

Marvin was used for drawing, displaying and characterizing chemical structures, substructures and reactions, Marvin 16.11.21.0, 2016, ChemAxon (http://www.chemaxon.com)

Appendix A

A 1. Results from one sample KS test for normality applied to the residual errors of the LogD models

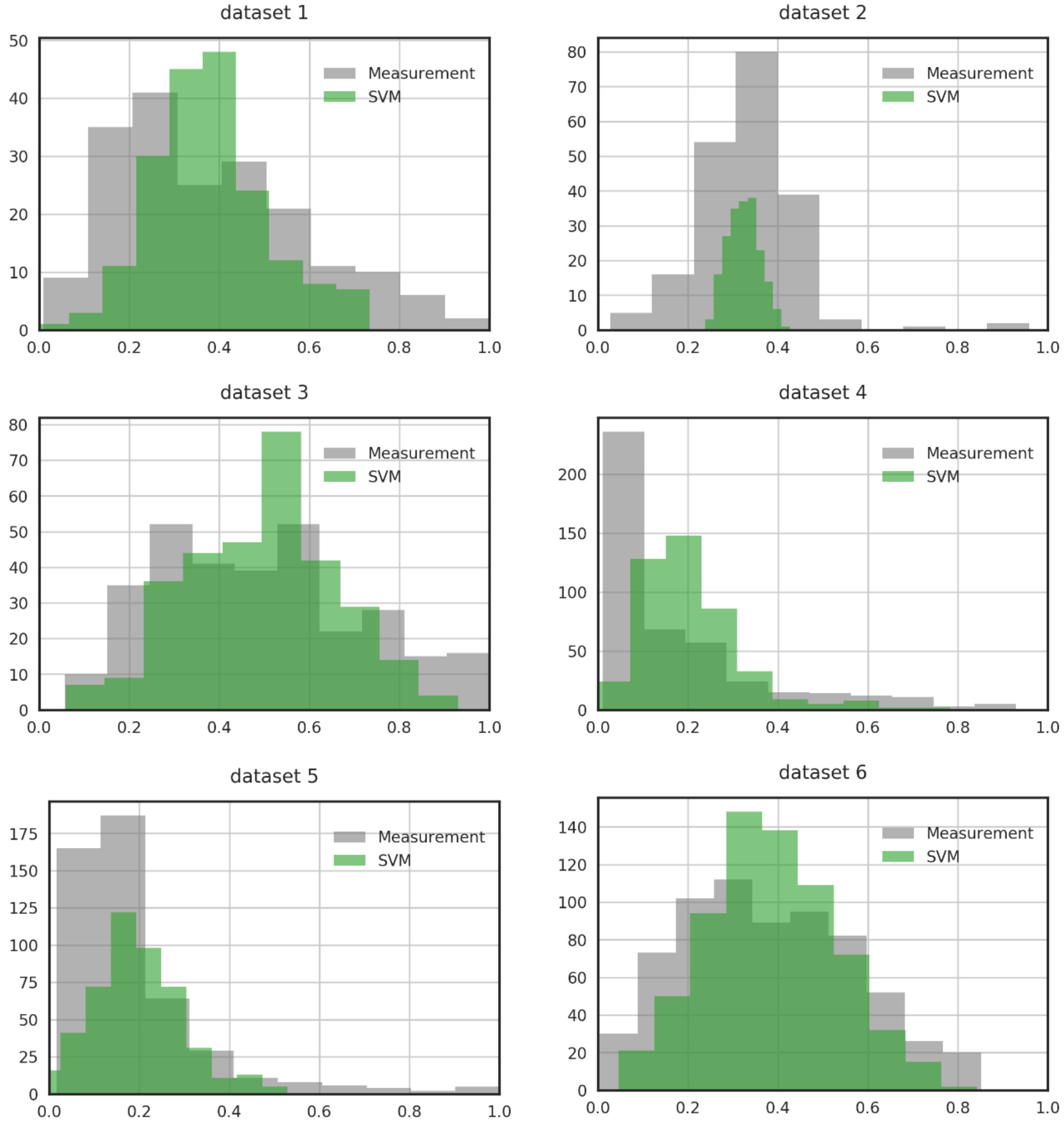
Model	**KS test**	**p**
KNN	0.50	0.00
RF	0.501	0.00
PLS	0.502	0.00
SVM	0.501	0.00

A 2. Optimal parameters for the RF and SVM ADME models identified by grid search optimisation

Dataset	**SVM**		**RF**		
	gamma	**C**	**Number of trees**	**Min. leaf size**	**Max. number of features**
1	0.001	2	100	5	default
2	0.01	0.1	100	35	default
3	0.01	2	100	5	default
4	0.01	1	100	5	sqrt
5	0.001	2	100	5	sqrt
6	0.01	1	100	5	default
7	0.01	0.1	100	5	default
8	0.01	0.1	100	5	sqrt
9	0.001	10	100	5	default
10	0.01	1	100	5	default

default: total number of descriptors
sqrt: square root of the total number of descriptors

A 3. Overlay of measurements and SVM predictions for the ADME datasets



dataset 1

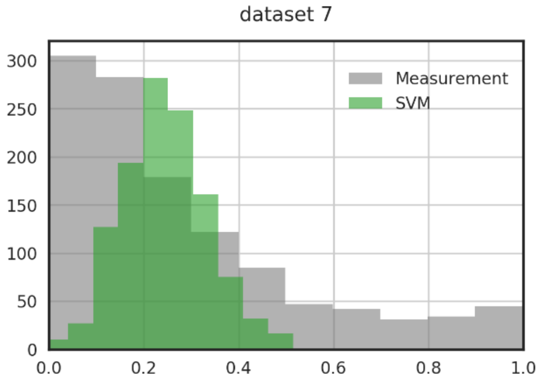
dataset 2

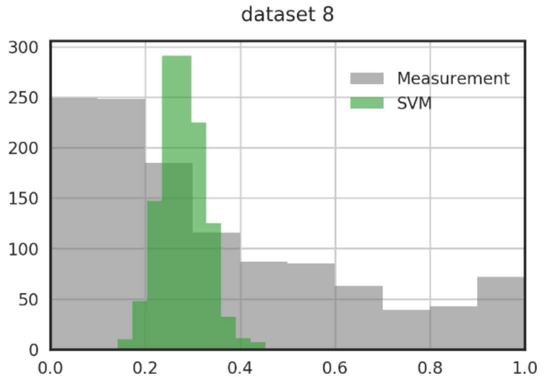
dataset 3

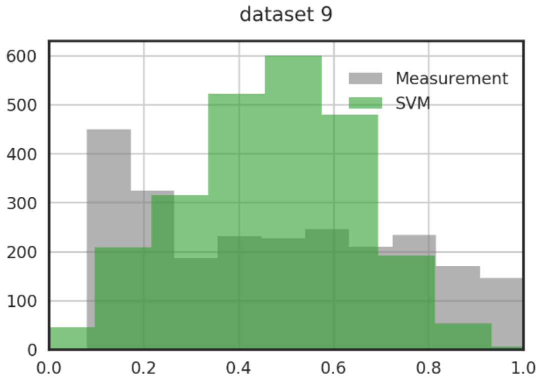
dataset 4

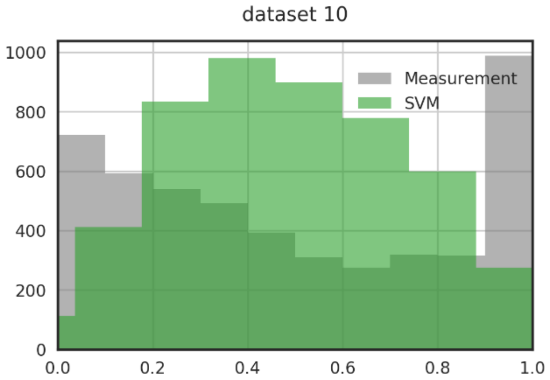
dataset 5

dataset 6

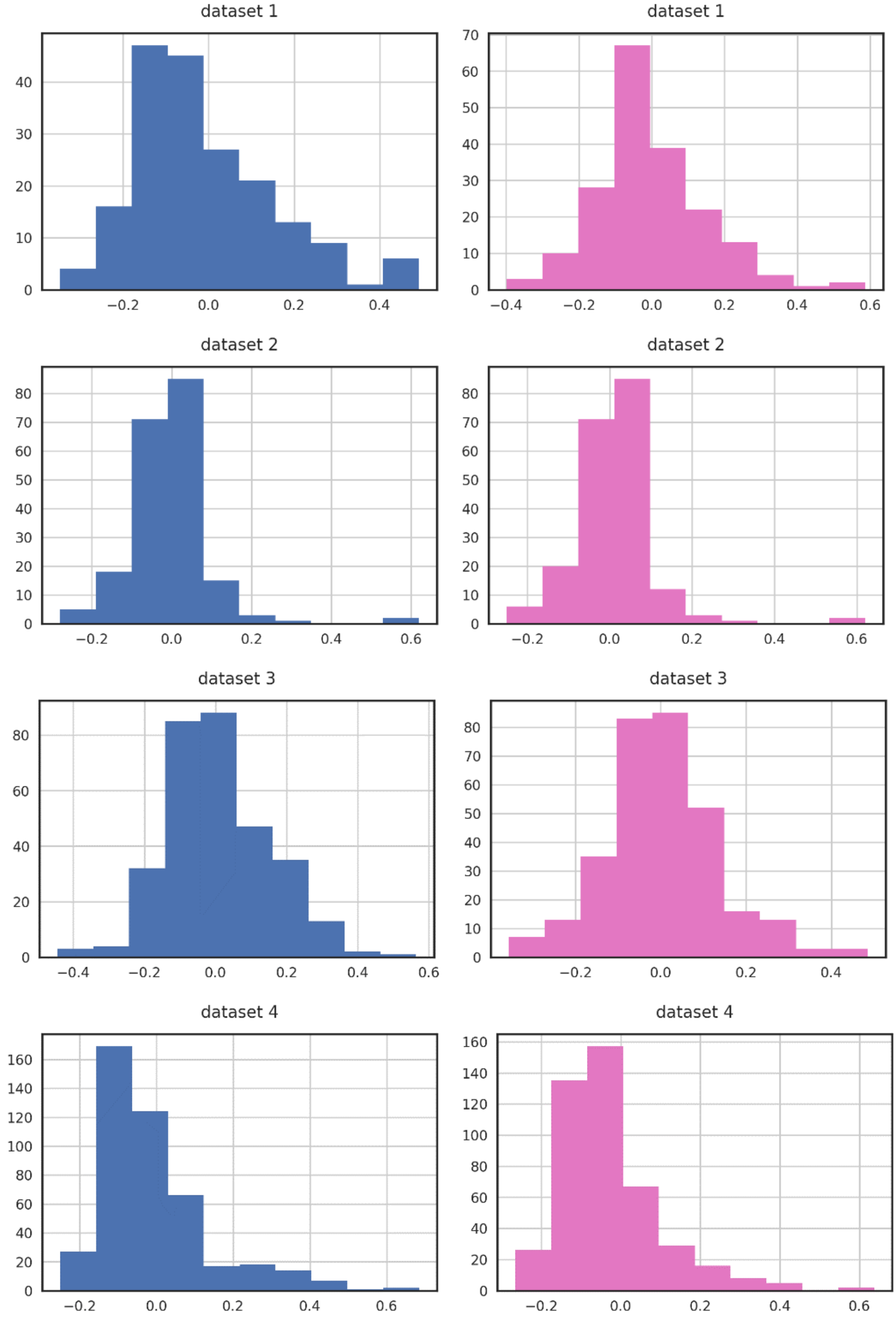

dataset 7
Measurement
SVM
300
250
200
150
100
50
0
0.0
0.2
0.4
0.6
0.8
1.0

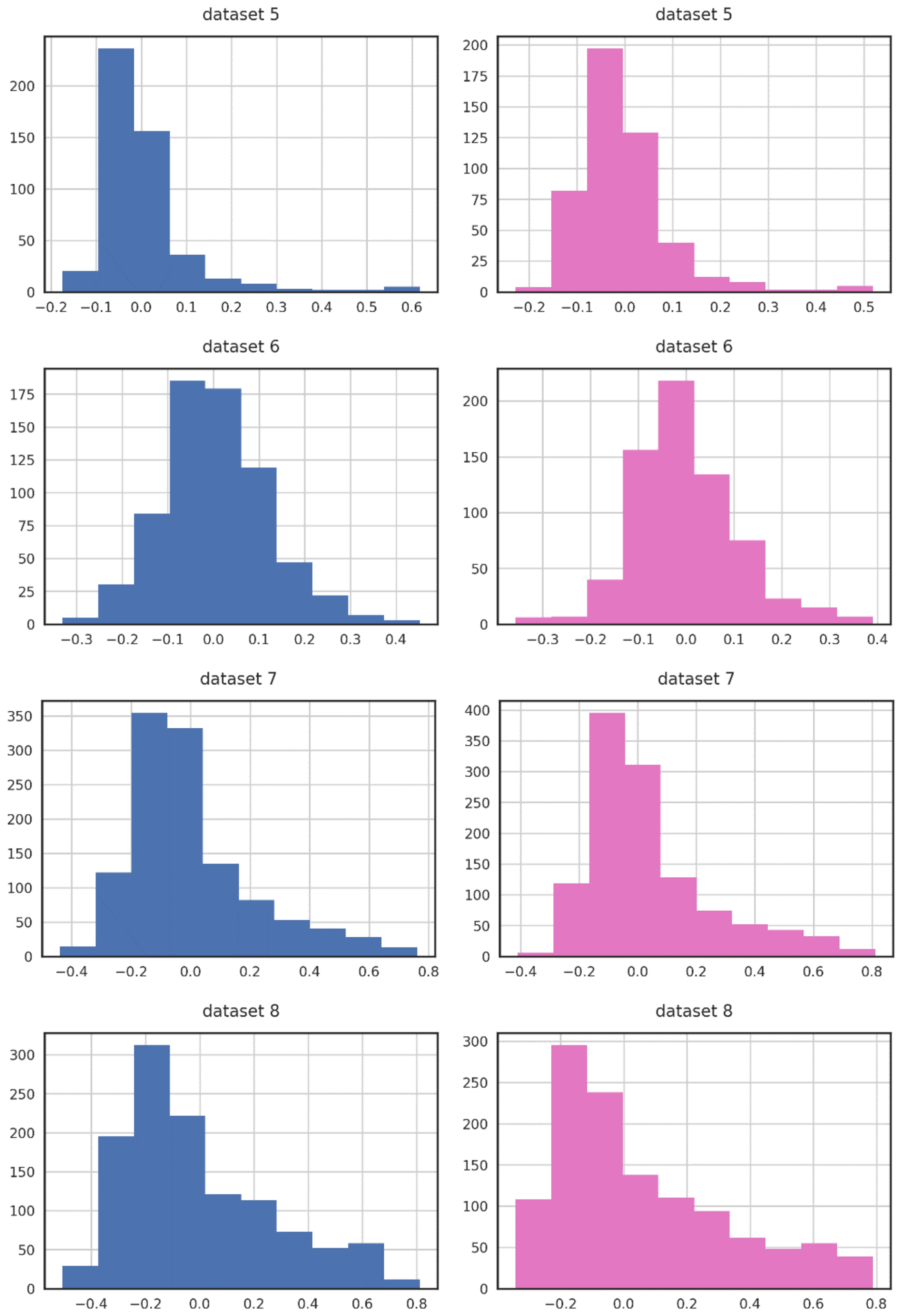

dataset 8
Measurement
SVM
300
250
200
150
100
50
0
0.0
0.2
0.4
0.6
0.8
1.0

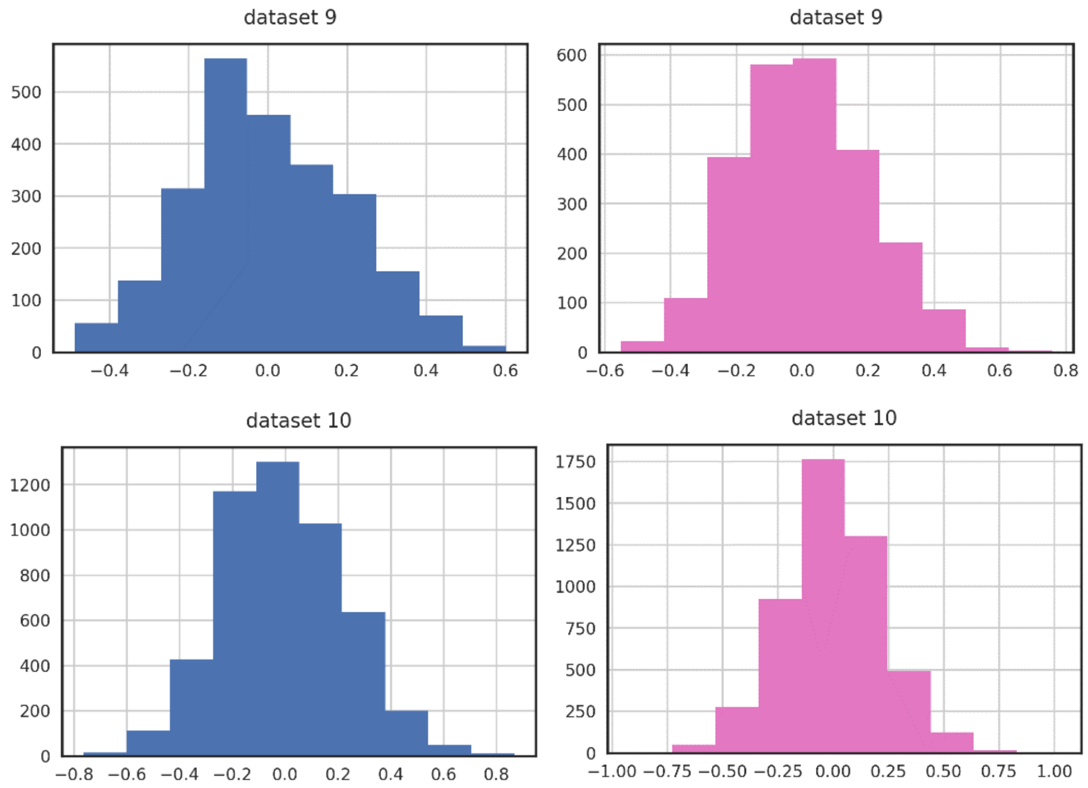

dataset 9
Measurement
SVM
600
500
400
300
200
100
0
0.0
0.2
0.4
0.6
0.8
1.0


dataset 10
Measurement
SVM
1000
800
600
400
200
0
0.0
0.2
0.4
0.6
0.8
1.0

A 4. Histograms of RF (left) and SVM (right) residual errors




dataset 5
0
50
100
150
200
−0.2
−0.1
0.0
0.1
0.2
0.3
0.4
0.5
0.6
dataset 5
0
25
50
75
100
125
150
175
200
−0.2
−0.1
0.0
0.1
0.2
0.3
0.4
0.5
dataset 6
0
25
50
75
100
125
150
175
−0.3
−0.2
−0.1
0.0
0.1
0.2
0.3
0.4
dataset 6
0
50
100
150
200
−0.3
−0.2
−0.1
0.0
0.1
0.2
0.3
0.4
dataset 7
0
50
100
150
200
250
300
350
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
dataset 7
0
50
100
150
200
250
300
350
400
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
dataset 8
0
50
100
150
200
250
300
−0.4
−0.2
0.0
0.2
0.4
0.6
0.8
dataset 8
0
50
100
150
200
250
300
−0.2
0.0
0.2
0.4
0.6
0.8



A 5. Accuracy of underlying QSAR models and the descriptor-based RF error model expressed in units of nRMSE

Dataset			**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
RF	**QSAR**	**CV**	0.170	0.094	0.151	0.151	0.104	0.132	0.221	0.260	0.220	0.273
		Holdout	0.170	0.070	0.142	0.129	0.091	0.117	0.183	0.180	0.201	0.247
	Error model	**CV**	0.161	0.077	0.139	0.148	0.078	0.153	0.183	0.200	0.163	0.161
		Holdout	0.153	0.120	0.125	0.189	0.115	0.147	0.181	0.214	0.165	0.159
SVM	**QSAR**	**CV**	0.166	0.096	0.140	0.149	0.104	0.125	0.221	0.260	0.218	0.265
		Holdout	0.169	0.071	0.137	0.126	0.090	0.110	0.214	0.230	0.212	0.237
	Error model	**CV**	0.159	0.074	0.147	0.119	0.076	0.152	0.202	0.209	0.083	0.145
		Holdout	0.163	0.119	0.116	0.162	0.121	0.133	0.193	0.216	0.145	0.162

A 6. Accuracy of underlying QSAR models and the AD-based RF error model expressed in units of nRMSE

Dataset			1	2	3	4	5	6	7	8	9	10
RF	QSAR	CV	0.17	0.094	0.151	0.151	0.104	0.132	0.221	0.26	0.22	0.273
		Holdout	0.17	0.07	0.142	0.129	0.091	0.117	0.183	0.18	0.201	0.247
	Error model	CV	0.213	0.086	0.184	0.166	0.129	0.264	0.243	0.251	0.217	0.225
		Holdout	0.202	0.134	0.165	0.211	0.191	0.253	0.241	0.268	0.220	0.222
SVM	QSAR	CV	0.166	0.096	0.14	0.149	0.104	0.125	0.221	0.26	0.218	0.265
		Holdout	0.169	0.071	0.137	0.126	0.09	0.11	0.214	0.23	0.212	0.237
	Error model	CV	0.155	0.081	0.150	0.119	0.083	0.170	0.213	0.224	0.087	0.129
		Holdout	0.160	0.131	0.119	0.162	0.131	0.148	0.203	0.232	0.151	0.144

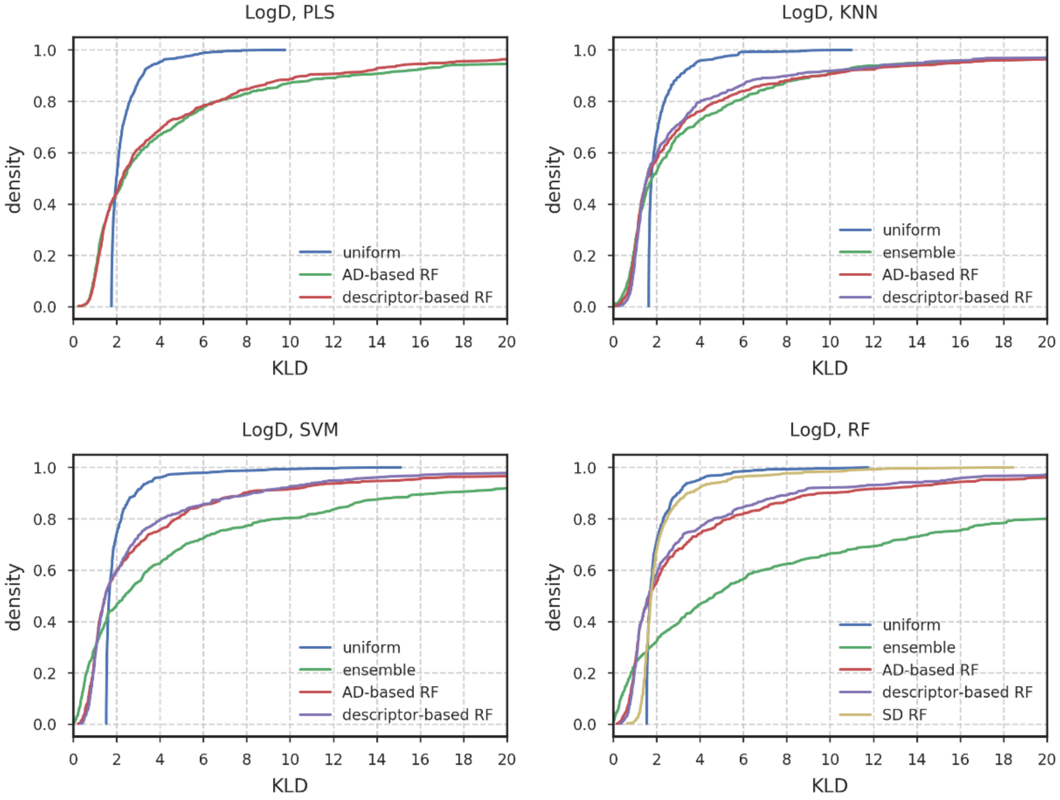
A 7. Fold-difference between the average errors of the descriptor-based RF error model and the average errors of RF and SVM QSAR models on cross-validation and holdout data

Dataset		1	2	3	4	5	6	7	8	9	10
RF	CV	0.9	0.8	0.9	1.0	0.7	1.2	0.8	0.8	0.7	0.6
	Holdout	0.9	1.7	0.9	1.5	1.3	1.3	1.0	1.2	0.8	0.6
SVM	CV	1.0	0.8	1.0	0.8	0.7	1.2	0.9	0.8	0.4	0.5
	Holdout	1.0	1.7	0.8	1.3	1.3	1.2	0.9	0.9	0.7	0.7

A 8. Fold-difference between the average errors of the AD-based RF error model and the average errors of RF and SVM QSAR models on cross-validation and holdout data

Dataset		1	2	3	4	5	6	7	8	9	10
RF	CV	1.3	0.9	1.2	1.1	1.2	2.0	1.1	1.0	1.0	0.8
	Holdout	1.2	1.9	1.2	1.6	2.1	2.2	1.3	1.5	1.1	0.9
SVM	CV	0.9	0.8	1.1	0.8	0.8	1.4	1.0	0.9	0.4	0.5
	Holdout	0.9	1.8	0.9	1.3	1.5	1.3	0.9	1.0	0.7	0.6

A 9. Overlay of KLD distributions for the predictions of the LogD models with a KLD cut-off at 20

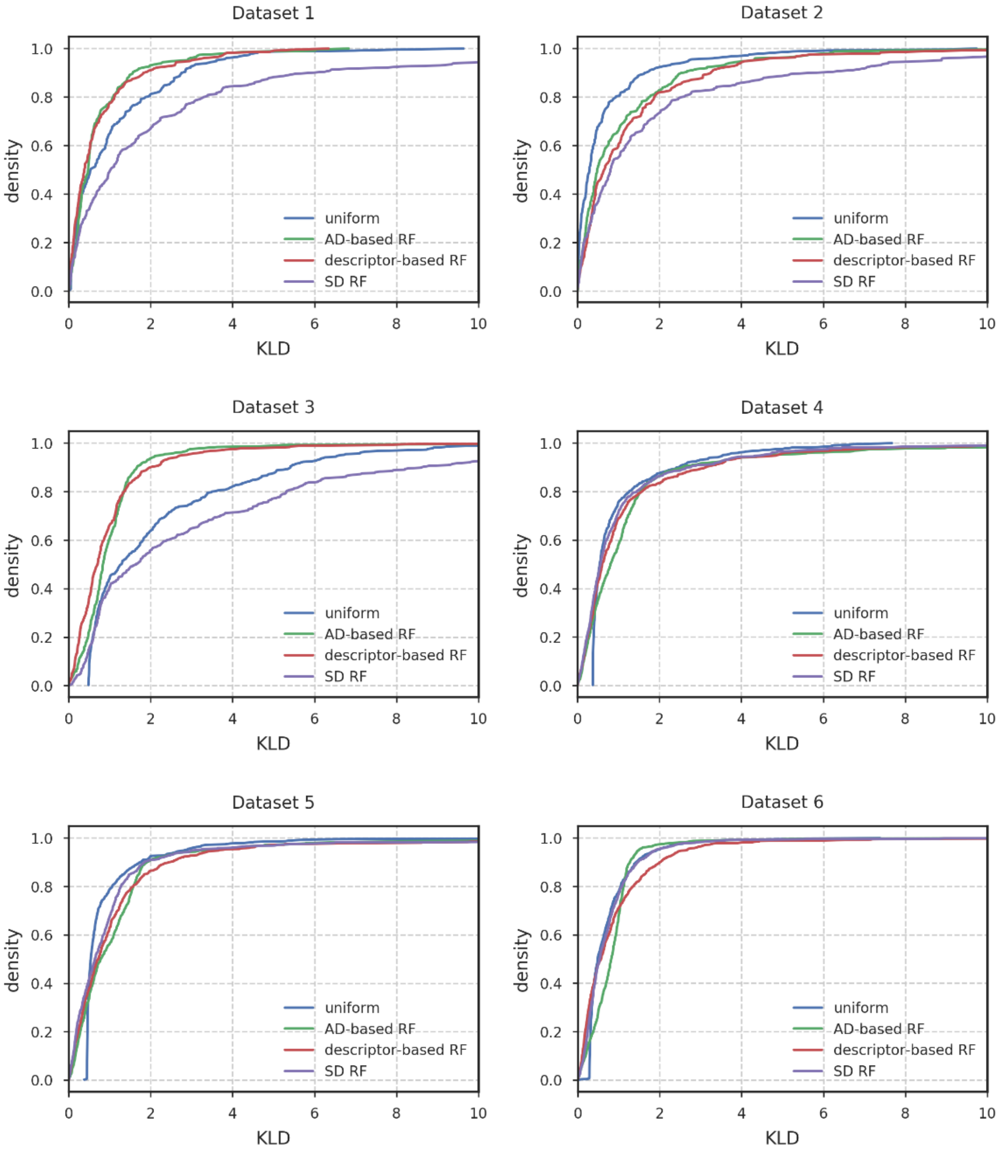

LogD, PLS

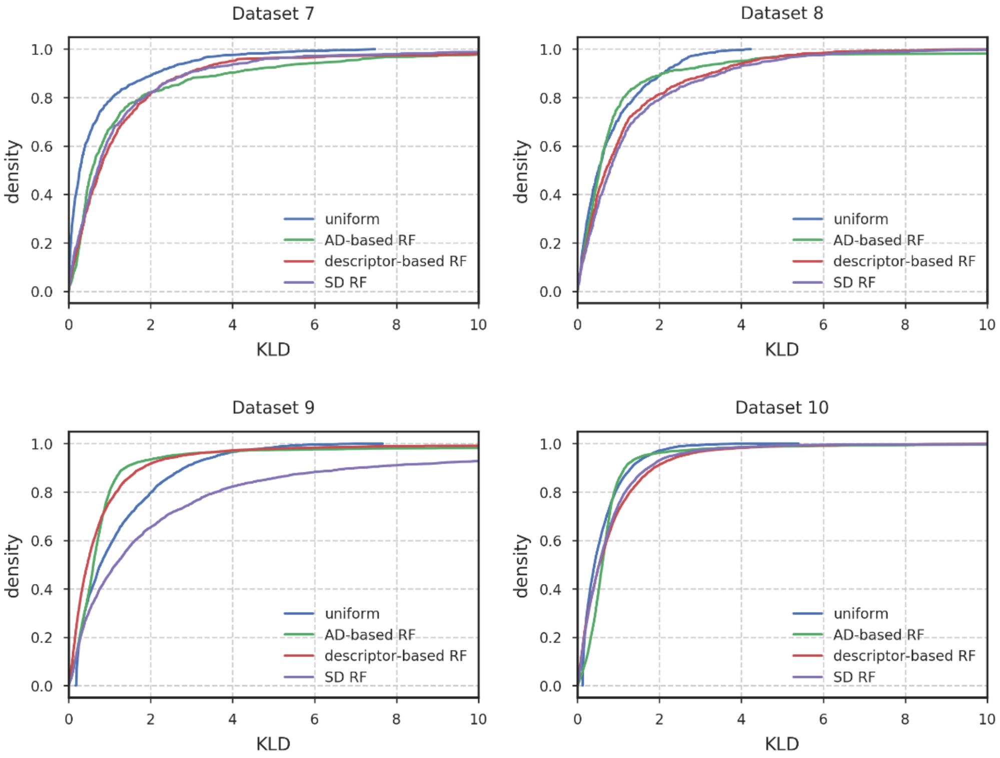
LogD, KNN

LogD, SVM

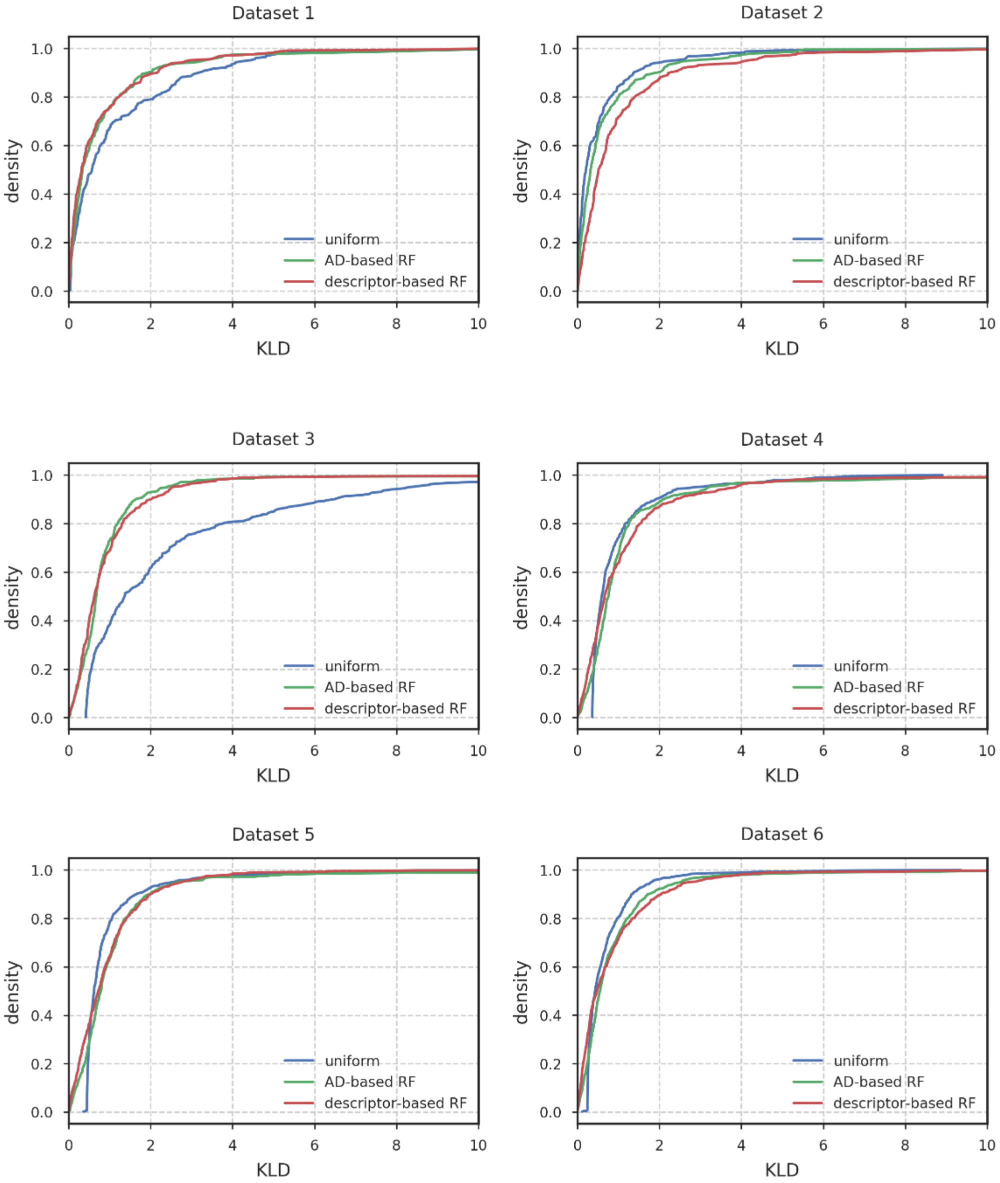
LogD, RF

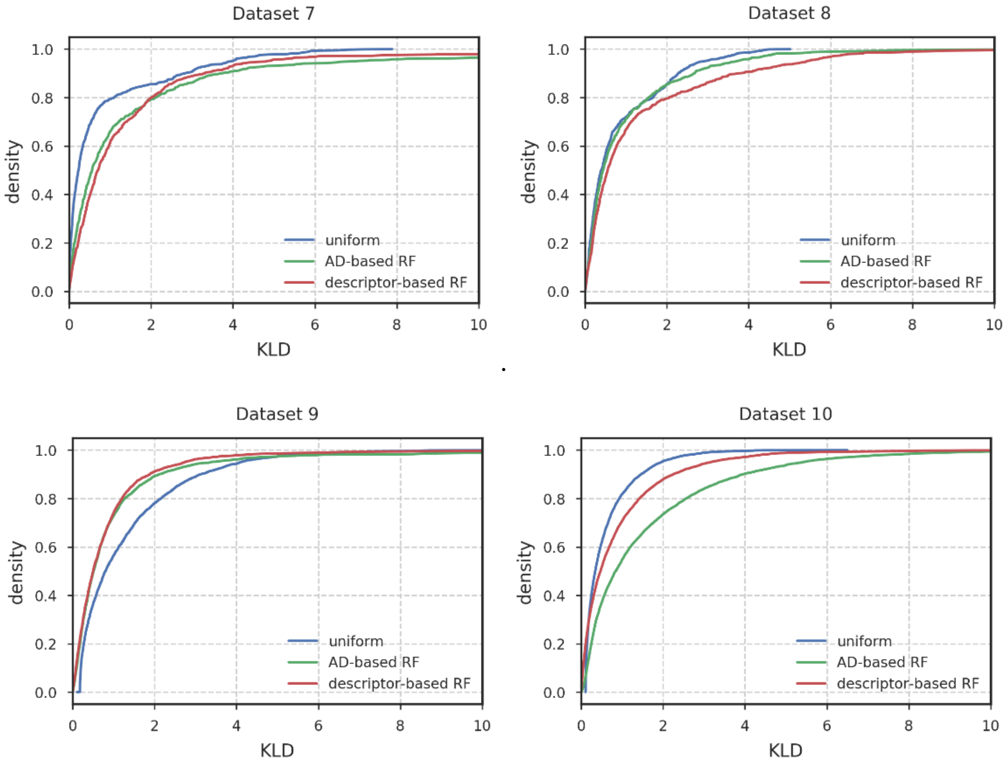
A 10.Overlay of KLD distributions for the RF predictions of the ADME models with a KLD cut-off at 10.




Dataset 7
Dataset 8
Dataset 9
Dataset 10
density
KLD
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10
uniform
AD-based RF
descriptor-based RF
SD RF

A 11. Overlay of KLD distributions for the SVM predictions of the ADME models with a KLD cut-off at 10.




Dataset 7
Dataset 8
Dataset 9
Dataset 10
density
KLD
1.0
0.8
0.6
0.4
0.2
0.0
0
2
4
6
8
10
uniform
AD-based RF
descriptor-based RF

A 12. Difference between the expected and actual error rate and PI size of ACPs for the underlying RF ADME models

		N	1	2	3	4	5	6	7	8	9	10
ΔEr	**B**	**10**	0.036	-0.020	-0.010	0.022	0.021	0.033	0.017	0.018	0.015	0.023
		100	0.036	-0.005	0.003	0.027	0.023	0.034	0.001	0.021	0.015	0.023
		1000	0.036	0.000	0.013	0.020	0.017	0.036	0.002	0.025	0.016	0.024
	C	**10**	0.015	0.015	-0.006	0.022	0.027	0.025	-0.002	0.014	0.005	0.017
		100	0.137	0.135	0.077	0.067	0.050	0.043	0.076	0.074	0.009	0.022
		1000	-0.170	-0.155	-0.174	-0.027	0.042	0.024	-0.028	-0.059	0.043	0.064
	R	**10**	0.020	-0.010	0.000	0.029	0.015	0.024	0.001	0.020	0.007	0.018
		100	0.025	-0.010	0.006	0.025	0.015	0.033	-0.002	0.021	0.009	0.019
		1000	0.025	-0.010	-0.003	0.025	0.021	0.031	-0.001	0.018	0.009	0.018
PI size	**B**	**10**	0.405	0.183	0.355	0.311	0.163	0.309	0.491	0.679	0.527	0.618
		100	0.413	0.181	0.366	0.305	0.167	0.308	0.483	0.683	0.531	0.616
		1000	0.410	0.184	0.364	0.307	0.166	0.309	0.483	0.685	0.531	0.616
	C	**10**	0.391	0.192	0.355	0.303	0.168	0.299	0.477	0.670	0.518	0.599
		100	0.580	0.338	0.423	0.361	0.177	0.312	0.601	0.801	0.519	0.601
		1000	0.263	0.126	0.226	0.263	0.172	0.291	0.436	0.585	0.558	0.669
	R	**10**	0.403	0.179	0.366	0.311	0.166	0.308	0.479	0.681	0.525	0.614
		100	0.400	0.184	0.364	0.309	0.166	0.306	0.480	0.676	0.529	0.614
		1000	0.403	0.182	0.362	0.308	0.168	0.307	0.480	0.673	0.528	0.613

A 13. Difference between the expected and actual error rate and PI size of ACPs for the underlying SVM ADME models

		N	1	2	3	4	5	6	7	8	9	10
ΔEr	**B**	**10**	0.031	0.045	0.026	0.018	0.011	0.049	-0.009	0.010	0.013	0.017
		100	0.036	0.030	0.023	0.034	0.011	0.056	-0.007	0.020	0.021	0.019
		1000	0.041	0.030	0.016	0.031	0.013	0.056	-0.002	0.021	0.019	0.020
	C	**10**	0.036	0.035	0.023	0.020	0.009	0.047	0.001	0.010	0.011	0.011
		100	0.142	0.145	0.094	0.061	0.019	0.055	0.078	0.067	0.014	0.011
		1000	-0.133	-0.145	-0.119	-0.045	-0.012	0.030	-0.021	-0.027	0.045	0.053
	R	**10**	0.031	0.010	0.016	0.031	0.011	0.038	-0.011	0.010	0.013	0.018
		100	0.020	0.020	0.016	0.029	0.005	0.052	-0.006	0.010	0.013	0.015
		1000	0.031	0.020	0.013	0.027	0.002	0.053	-0.006	0.012	0.011	0.017
PI size	**B**	**10**	0.408	0.203	0.347	0.326	0.205	0.294	0.447	0.644	0.511	0.585
		100	0.404	0.197	0.347	0.331	0.204	0.297	0.457	0.651	0.512	0.587
		1000	0.404	0.170	0.184	0.330	0.205	0.296	0.460	0.651	0.514	0.588
	C	**10**	0.409	0.198	0.336	0.306	0.199	0.286	0.460	0.634	0.506	0.568
		100	0.566	0.331	0.395	0.351	0.202	0.291	0.636	0.824	0.511	0.568
		1000	0.264	0.135	0.319	0.260	0.189	0.273	0.428	0.583	0.542	0.644
	R	**10**	0.393	0.191	0.350	0.326	0.203	0.287	0.446	0.629	0.513	0.582
		100	0.393	0.196	0.345	0.318	0.202	0.290	0.449	0.632	0.510	0.581
		1000	0.395	0.195	0.343	0.321	0.202	0.291	0.450	0.636	0.511	0.581

A 14. Spearman's rank correlation coefficient between the variable PI estimates and the actual prediction errors of the underlying RF predictions

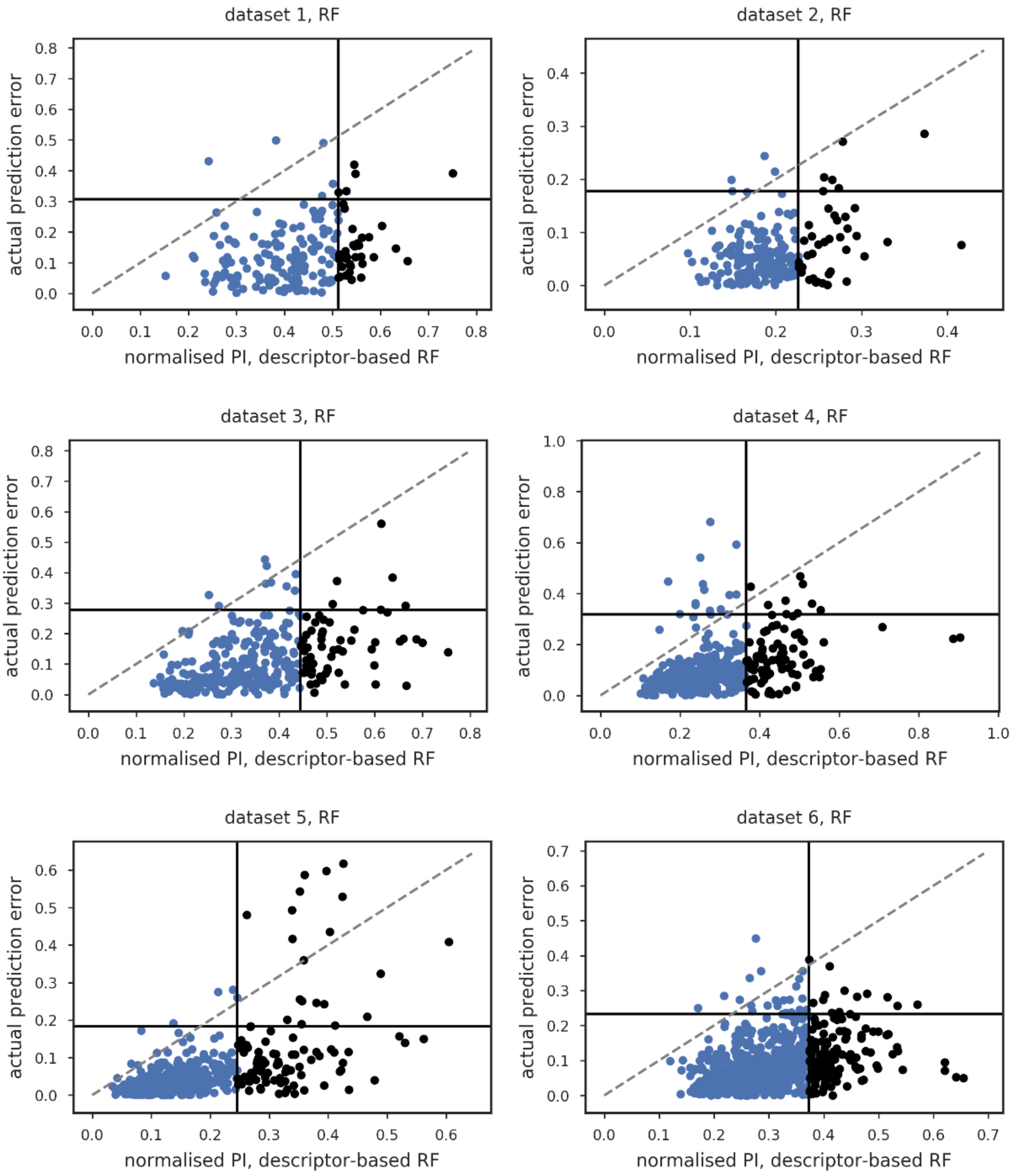
Dataset	**Normalisation method**			
	D2M	**AD-RF**	**RF**	**SVM**
1	0.1	0.08	0.30	0.12
2	0.12	-0.04	0.21	0.29
3	0.31	0.5	0.45	0.45
4	0.33	0.42	0.46	0.46
5	0.11	0.57	0.60	0.57
6	0.17	0.26	0.33	0.34
7	0.24	0.38	0.44	0.41
8	-0.02	0.05	0.12	0.14
9	0.08	0.1	0.31	0.28
10	0.08	0.3	0.40	0.37
Mean	*0.15*	*0.26*	*0.36*	*0.34*

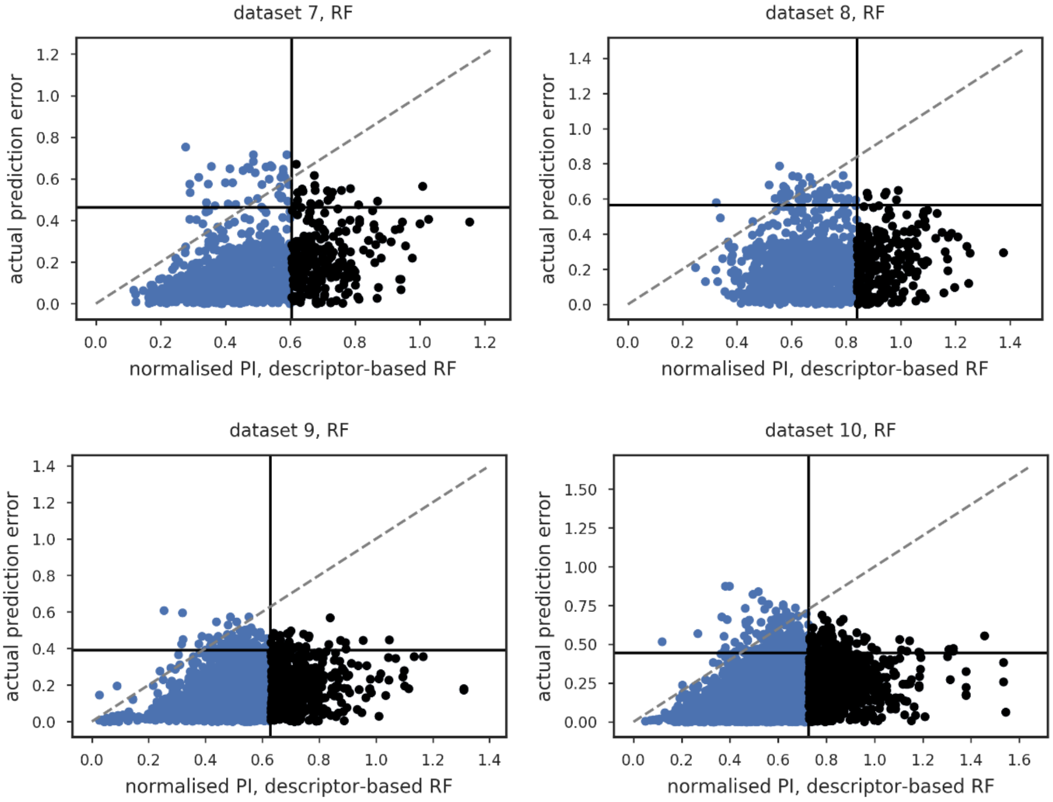
A 15. Spearman's rank correlation coefficient between the variable PI estimates and the actual prediction errors of the underlying SVM predictions

Dataset	**Normalisation method**			
	D2M	**AD-RF**	**RF**	**SVM**
1	0.05	0.17	0.22	0.12
2	0.11	0.08	0.13	0.23
3	0.32	0.37	0.28	0.35
4	0.17	0.4	0.33	0.37
5	0.15	0.51	0.48	0.42
6	0.23	0.19	0.19	0.24
7	0.19	0.34	0.4	0.38
8	-0.02	0.1	0.12	0.09
9	0.02	0.33	0.31	0.24
10	0.01	0.27	0.32	0.28
Mean	*0.12*	*0.27*	*0.28*	*0.27*

A 16. Normalised PIs using descriptor-based RF error models plotted against prediction errors of the RF ADME models for holdout data.

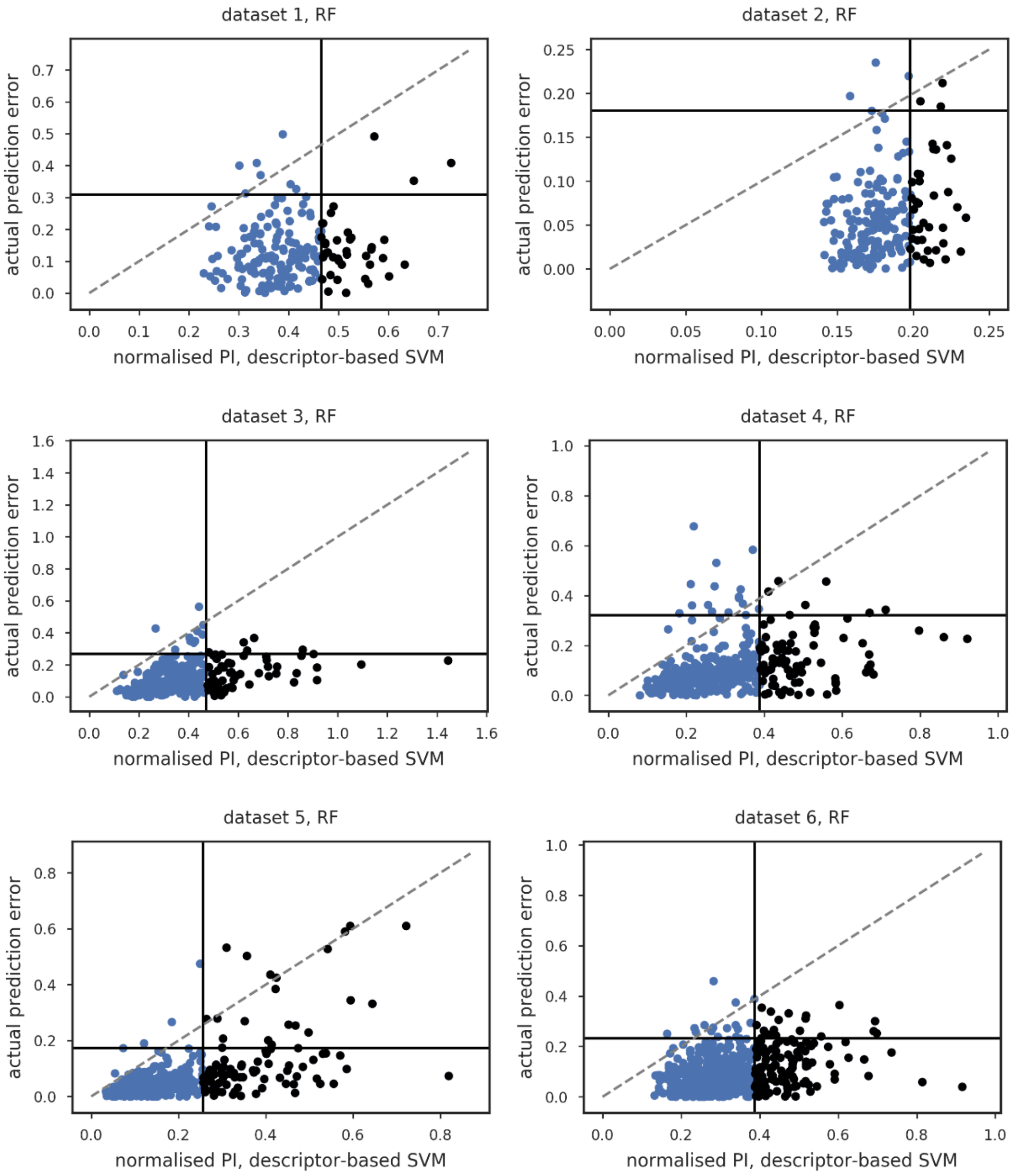
The diagonal separates non-valid (left) from valid (right) PIs. In black are the predictions with PIs greater than the 80th percentile of the PIs, which is indicated by the vertical line. The horizontal line indicates the 95th percentile of the actual prediction errors which is used to define prediction error outliers

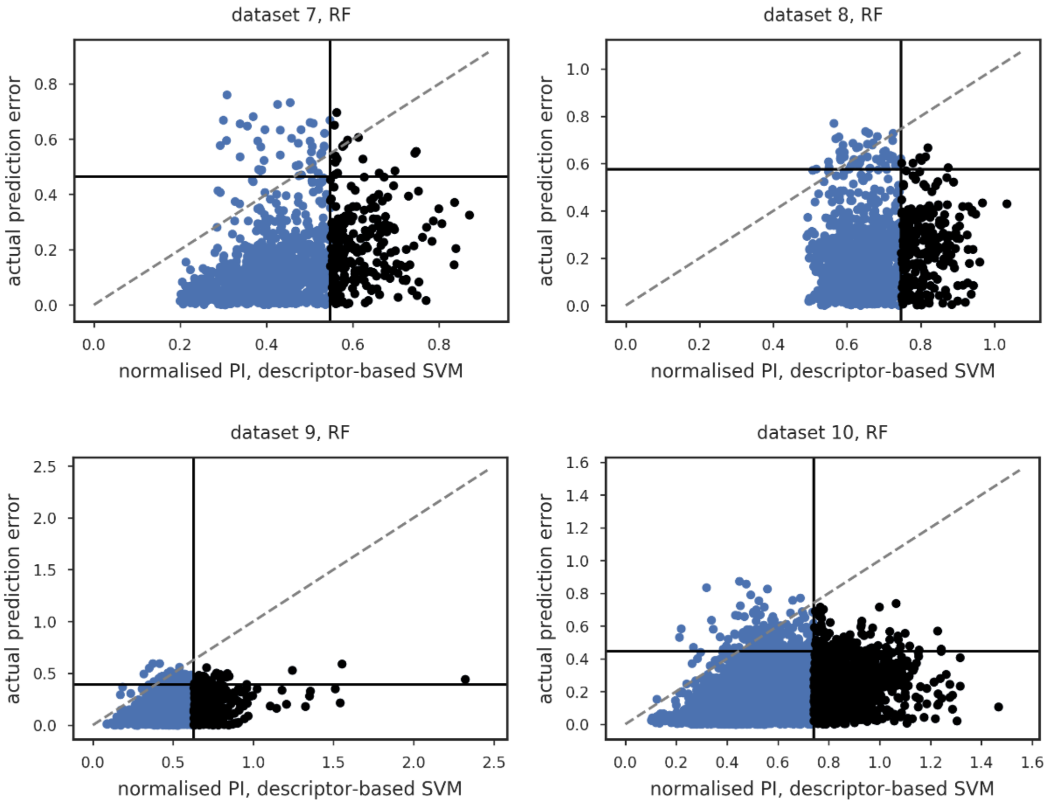



dataset 7, RF
dataset 8, RF
dataset 9, RF
dataset 10, RF
actual prediction error
normalised PI, descriptor-based RF
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
0.00
0.25
0.50
0.75
1.00
1.25
1.50

A 17. Normalised PIs using descriptor-based SVM error models plotted against prediction errors of the RF ADME models for holdout data.

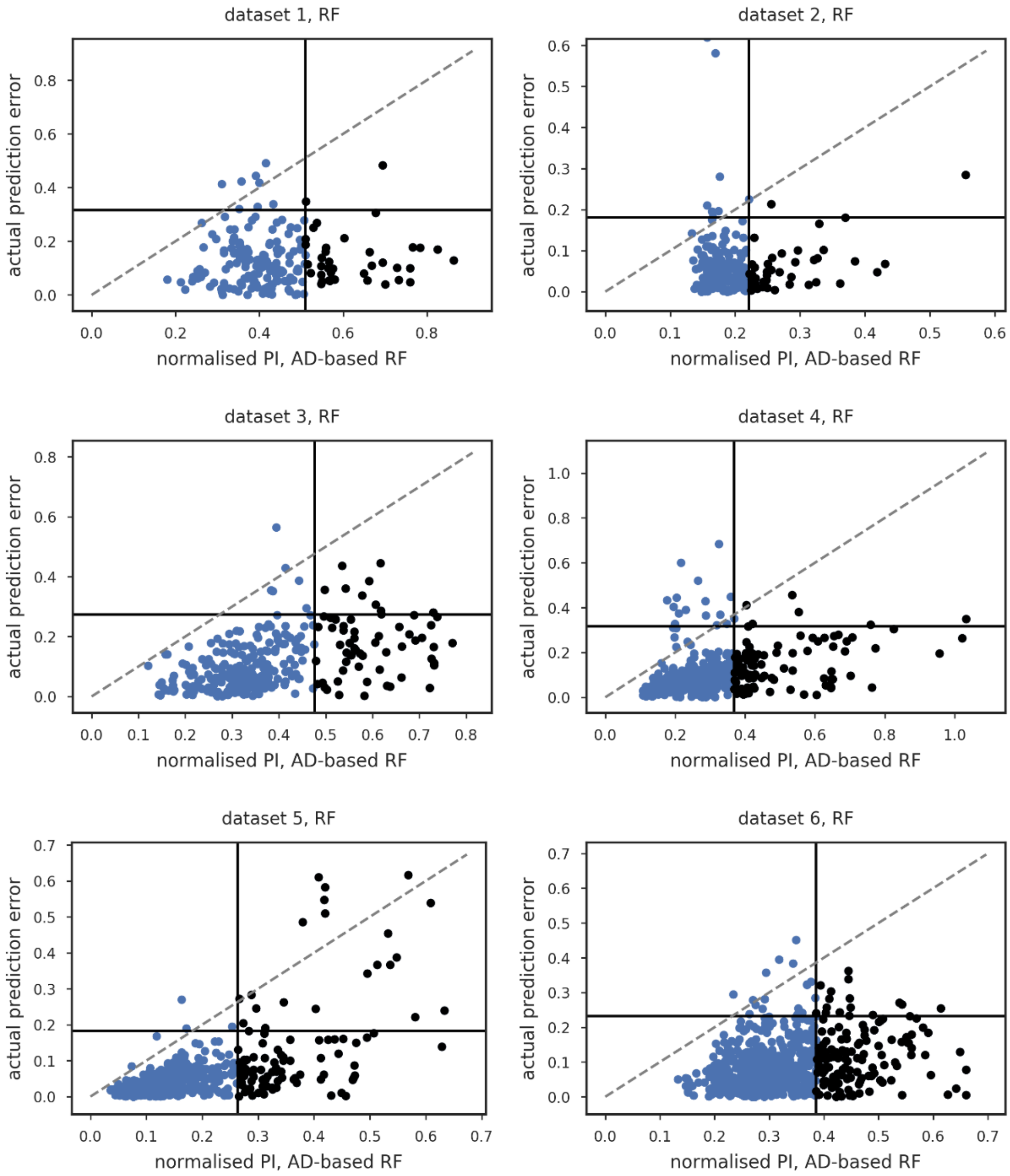
The diagonal separates non-valid (left) from valid (right) PIs. In black are the predictions with PIs greater than the 80th percentile of the PIs, which is indicated by the vertical line. The horizontal line indicates the 95th percentile of the actual prediction errors which is used to define prediction error outliers.

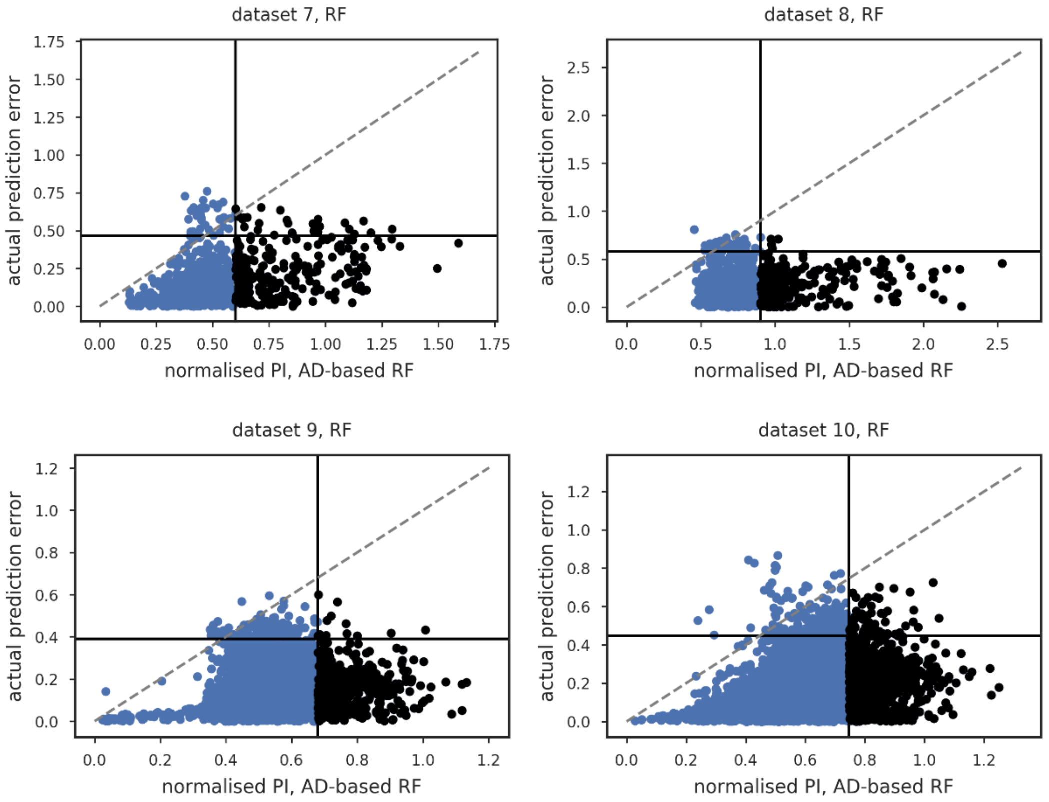



dataset 7, RF
actual prediction error
0.0
0.2
0.4
0.6
0.8
0.0
0.2
0.4
0.6
0.8
normalised PI, descriptor-based SVM
dataset 8, RF
actual prediction error
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
normalised PI, descriptor-based SVM
dataset 9, RF
actual prediction error
0.0
0.5
1.0
1.5
2.0
2.5
0.0
0.5
1.0
1.5
2.0
2.5
normalised PI, descriptor-based SVM
dataset 10, RF
actual prediction error
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
normalised PI, descriptor-based SVM

A 18. Normalised PIs using AD-based RF error models plotted against prediction errors of the RF ADME models for holdout data.

The diagonal separates non-valid (left) from valid (right) PIs. In black are the predictions with PIs greater than the 80th percentile of the PIs, which is indicated by the vertical line. The horizontal line indicates the 95th percentile of the actual prediction errors which is used to define prediction error outliers.





dataset 7, RF

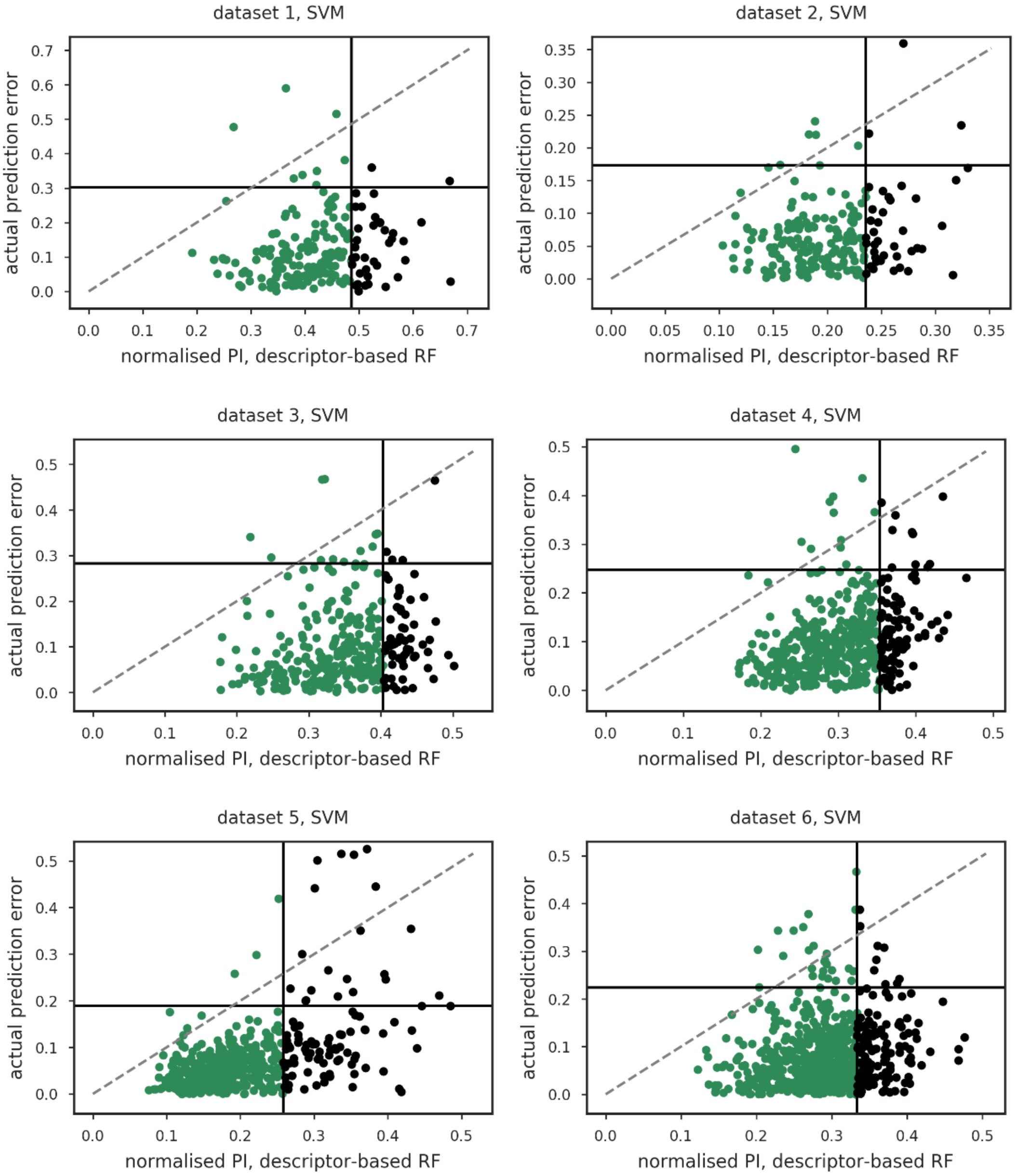
dataset 8, RF

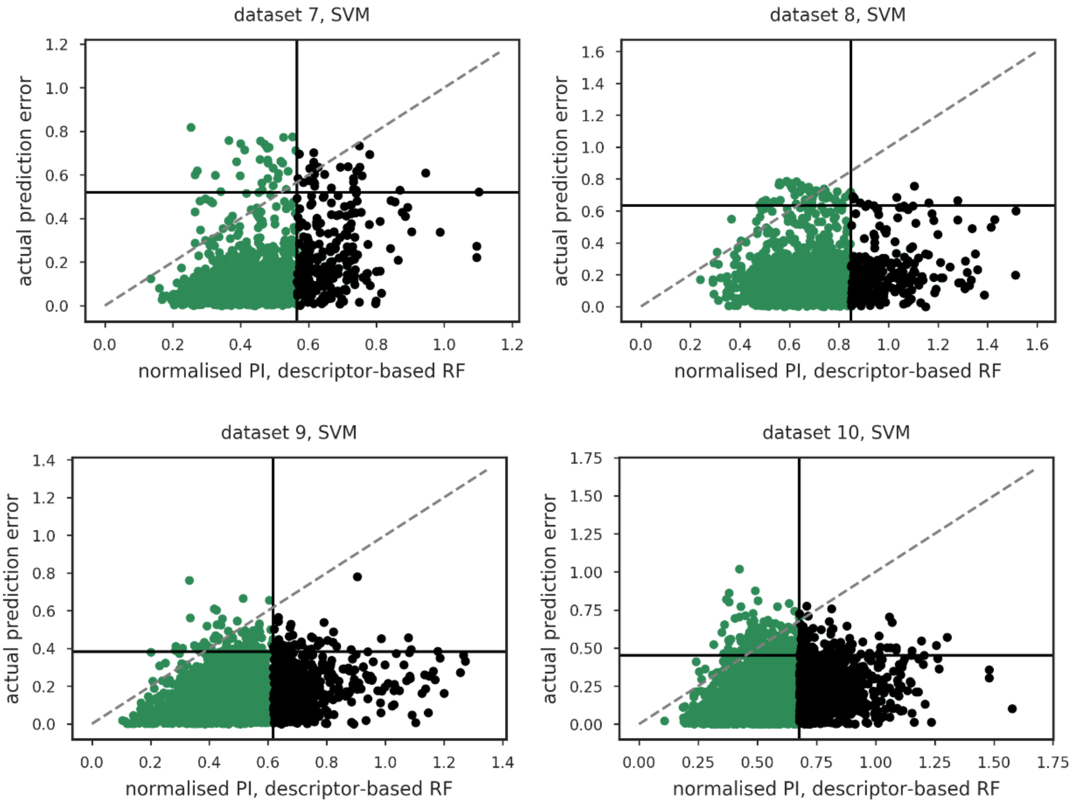
dataset 9, RF

dataset 10, RF

A 19. Normalised PIs using descriptor-based RF error models plotted against prediction errors of the SVM ADME models for holdout data.

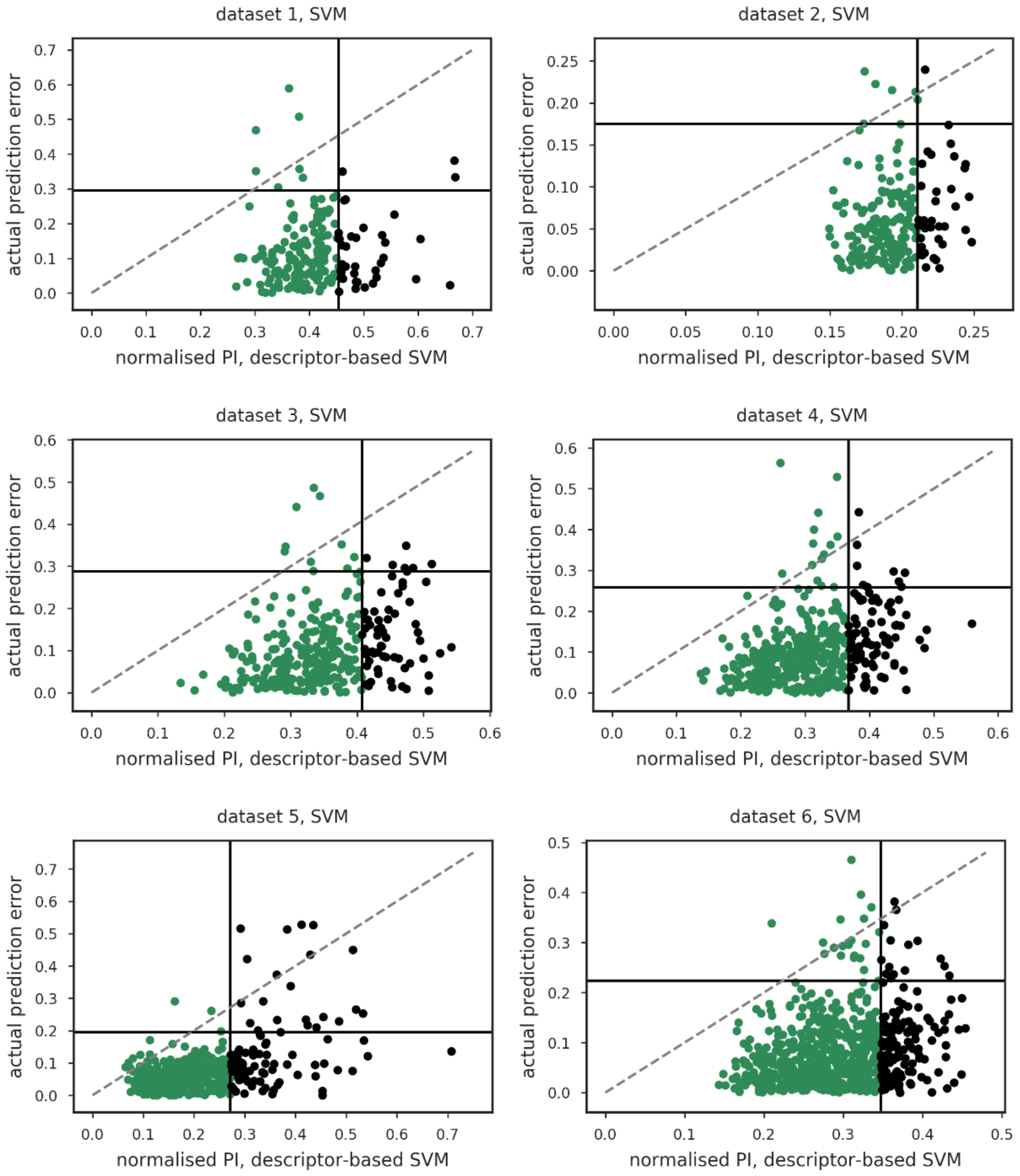
The diagonal separates non-valid (left) from valid (right) PIs. In black are the predictions with PIs greater than the 80th percentile of the PIs, which is indicated by the vertical line. The horizontal line indicates the 95th percentile of the actual prediction errors which is used to define prediction error outliers.

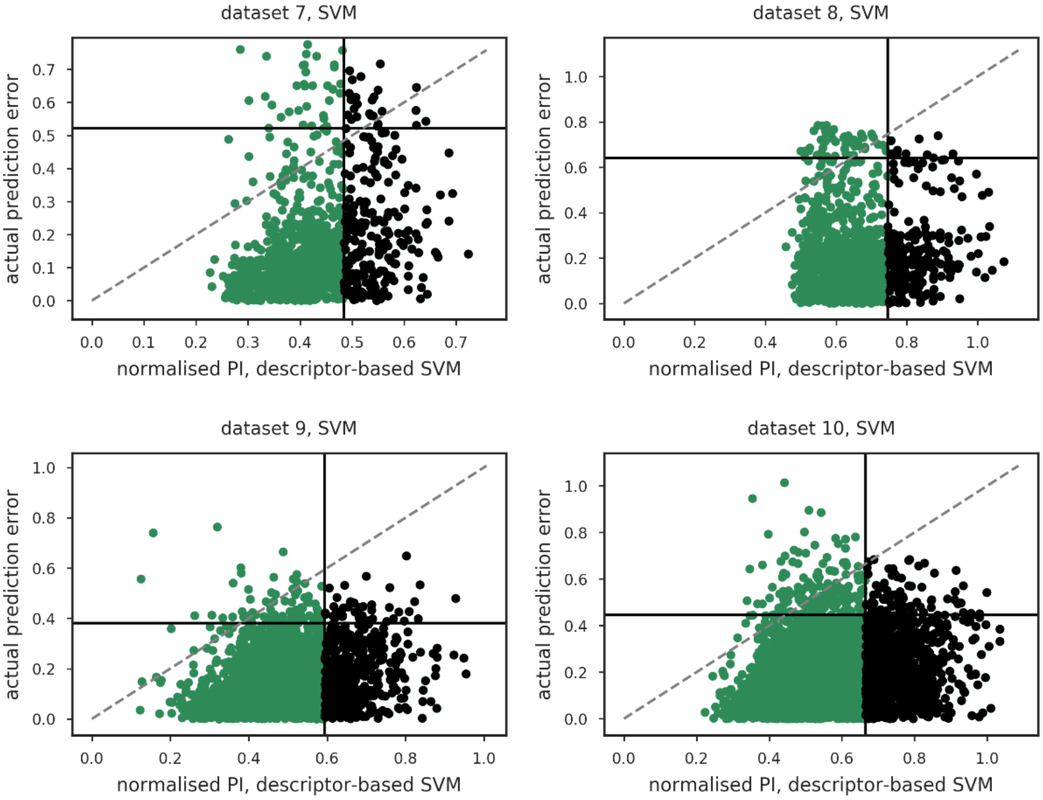



dataset 7, SVM
actual prediction error
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
normalised PI, descriptor-based RF
dataset 8, SVM
actual prediction error
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
normalised PI, descriptor-based RF
dataset 9, SVM
actual prediction error
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
normalised PI, descriptor-based RF
dataset 10, SVM
actual prediction error
1.75
1.50
1.25
1.00
0.75
0.50
0.25
0.00
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
normalised PI, descriptor-based RF

A 20. Normalised PIs using descriptor-based SVM error models plotted against prediction errors of the SVM ADME models for holdout data.

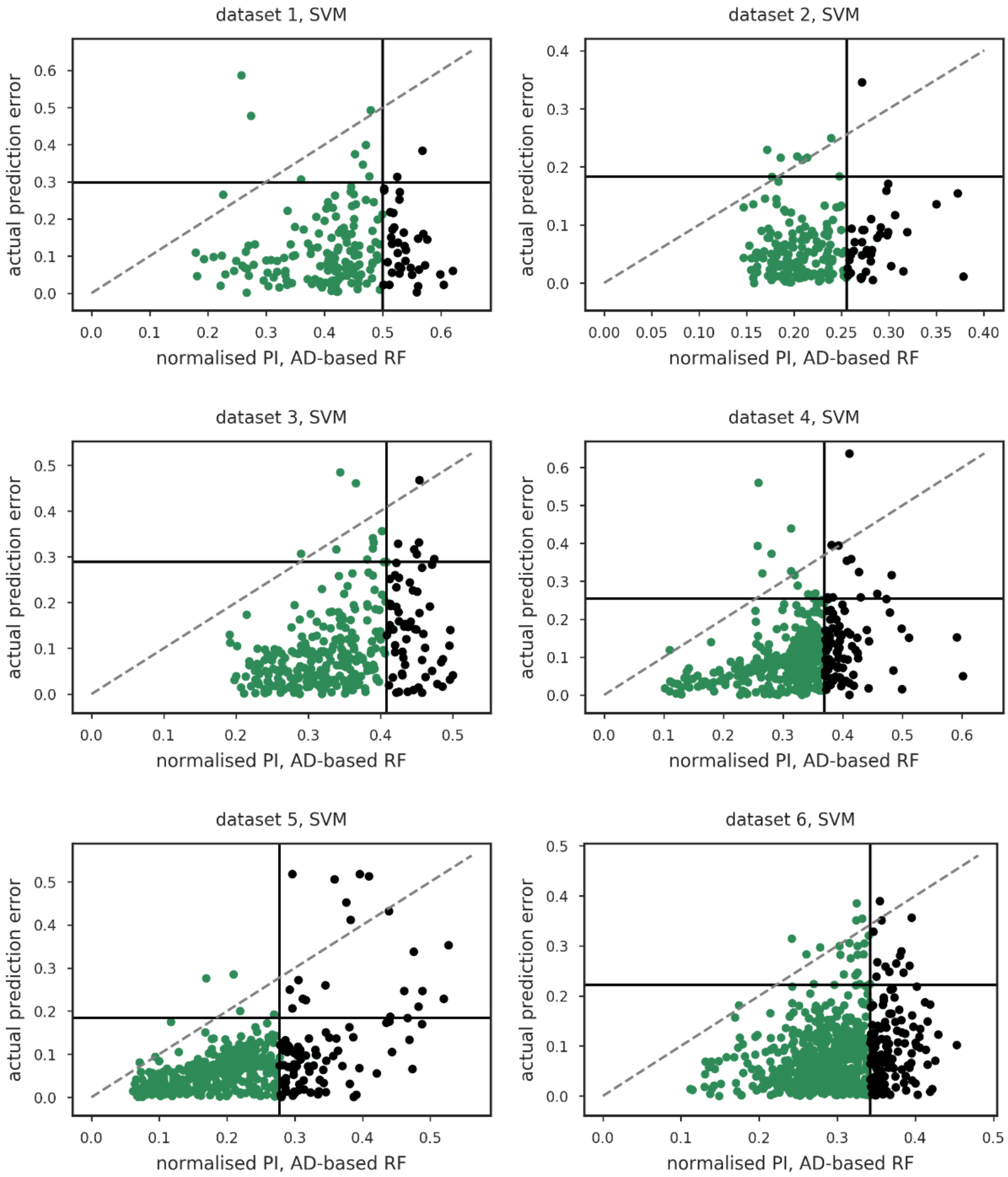
The diagonal separates non-valid (left) from valid (right) PIs. In black are the predictions with PIs greater than the 80th percentile of the PIs, which is indicated by the vertical line. The horizontal line indicates the 95th percentile of the actual prediction errors which is used to define prediction error outliers.

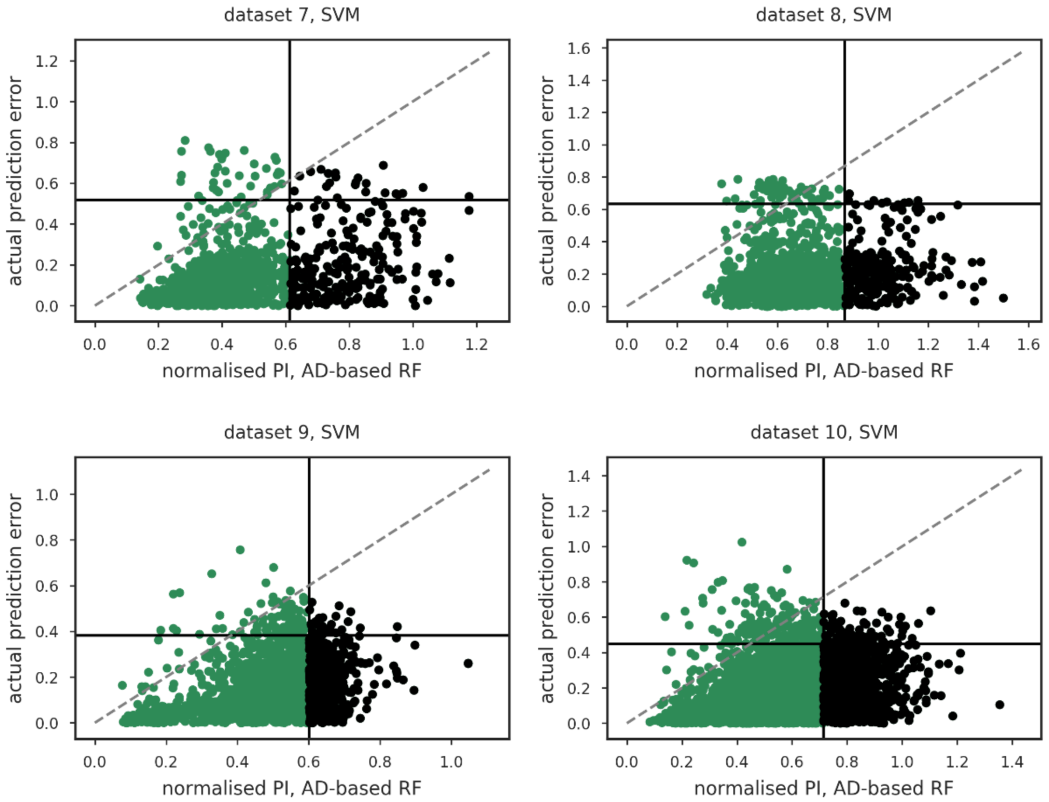



dataset 7, SVM
actual prediction error
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
normalised PI, descriptor-based SVM
dataset 8, SVM
actual prediction error
0.0
0.2
0.4
0.6
0.8
1.0
normalised PI, descriptor-based SVM
dataset 9, SVM
actual prediction error
0.0
0.2
0.4
0.6
0.8
1.0
normalised PI, descriptor-based SVM
dataset 10, SVM
actual prediction error
0.0
0.2
0.4
0.6
0.8
1.0
normalised PI, descriptor-based SVM

A 21. Normalised PIs using AD-based RF error models plotted against prediction errors of the SVM ADME models for holdout data.

The diagonal separates non-valid (left) from valid (right) PIs. In black are the predictions with PIs greater than the 80th percentile of the PIs, which is indicated by the vertical line. The horizontal line indicates the 95th percentile of the actual prediction errors which is used to define prediction error outliers.




dataset 7, SVM
actual prediction error
normalised PI, AD-based RF
0.0
0.2
0.4
0.6
0.8
1.0
1.2
dataset 8, SVM
actual prediction error
normalised PI, AD-based RF
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4
1.6
dataset 9, SVM
actual prediction error
normalised PI, AD-based RF
0.0
0.2
0.4
0.6
0.8
1.0
dataset 10, SVM
actual prediction error
normalised PI, AD-based RF
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1.4

A 22. Linear correlation between underlying predictions (p) and normalised PIs (PI) to the actual prediction errors of the underlying models.

Highlighted in bold are the results that suggest that the correlation between the prediction and the actual errors could be a good indicator of the correlation between the PIs and the actual errors in biased datasets.

			Dataset									
Underlying model	**Normalising method**		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
RF	SVM	**p**	0.11	-0.1	0.21	**0.36**	**0.57**	0.23	**0.36**	0.15	0.18	0.08
		PI	0.15	0.25	0.40	**0.36**	**0.60**	0.35	**0.34**	0.11	0.30	0.34
SVM	RF	**p**	0.11	-0.1	0.19	**0.33**	**0.49**	0.12	**0.30**	0.06	0.13	0.07
		PI	0.16	0.14	0.21	**0.30**	**0.51**	0.17	**0.39**	0.11	0.30	0.30
SVM	ADRF	**p**	0.11	-0.09	0.20	**0.30**	**0.50**	0.13	**0.31**	0.05	0.13	0.08
		PI	0.10	0.02	0.37	**0.34**	**0.54**	0.23	**0.32**	0.06	0.33	0.33
SVM	SVM	**p**	0.12	-0.11	0.21	**0.31**	**0.49**	0.12	**0.31**	0.05	0.13	0.07
		PI	0.10	0.18	0.32	**0.33**	**0.51**	0.25	**0.33**	0.08	0.23	0.26
RF	ADRF	**p**	0.13	-0.09	0.21	**0.33**	**0.56**	**0.21**	**0.36**	0.13	0.18	0.08
		PI	0.06	0.06	0.47	**0.35**	**0.62**	**0.27**	**0.39**	0.07	0.09	0.28
RF	RF	**p**	0.12	-0.07	0.21	**0.35**	**0.56**	0.23	**0.37**	0.16	0.18	0.08
		PI	0.27	0.21	0.42	**0.38**	**0.57**	0.31	**0.39**	0.12	0.30	0.36

Appendix B

B 1. Definition of KLD metric and demonstration of its behaviour

Given a measurement distribution, p(x), and a prediction distribution, q(x); then KLD may be used to measure the statistical divergence of the two distributions. In information theory, this quantity is also referred to as relative entropy and represents the information loss incurred by replacing the distribution p(x) with q(x). The definition of KLD is provided in Equation B- 1:

$$\begin{aligned} KL(p||q) &= -\int p(x)\ln q(x)\,dx - \left(-\int p(x)\ln p(x)\,dx\right) \\ &= -\int p(x)\ln\frac{q(x)}{p(x)}dx \end{aligned} \qquad \text{B- 1}$$

If the measurement distribution and the prediction distribution are Gaussian; then the KLD may be parametrically calculated using Equation B- 2:

$$D_{KL}(S_{p_i}, S_{q_i}) = \left[\frac{(\mu_{pi}-\mu_{qi})^2}{2\sigma_{qi}{}^2} + \frac{\sigma_{pi}{}^2}{2\sigma_{qi}{}^2} + \ln\frac{\sigma_{qi}}{\sigma_{pi}}\right] - \frac{1}{2} \qquad \text{B- 2}$$

where μ_{pi} is the measurement value with a standard error of σ_{pi} i.e., the measurement error and $\mu_{qi}$is the predicted value obtained from the QSAR model with a prediction error estimate of σ_{qi} . The value of D_{KL} is unbounded, non-negative and equal to zero only when the two distribution fully overlap. Therefore, the smaller the value of the KLD score the higher is the overlap of the two distributions. The order of the value increases when the distance of the means becomes increasingly larger than the prediction error estimate and when the magnitude of the prediction error estimate and the measurement error estimate varies. An example of how this may vary is provided below.

For a QSAR predictions with equal uncertainty estimates as the assay, Equation B- 2 is simplified to

$$D_{KL}(S_{p_i}, S_{q_i}) = \frac{(\mu_{pi}-\mu_{qi})^2}{2\sigma_{qi}{}^2} \qquad \text{B- 3}$$

And $D_{KL}(S_{p_i}, S_{q_i})$ is equal to 0 for $(\mu_{pi}-\mu_{qi})^2 = \sigma_{qi}{}^2$.

If we assume that the uncertainty of a model's predictions will always be greater than the uncertainty of the measurement in Equation B- 3, then the logarithmic component will always be positive. For example, if the uncertainty of the measurement is ${\sigma_{pi}}^2{=}\left(2\sigma_{qi}\right)^2$ using Equation B- 3 we have

$$D_{KL}\left(S_{p_i}, S_{q_i}\right) = \left[\frac{(\mu_{pi} - \mu_{qi})^2}{2{\sigma_{qi}}^2} + \frac{4{\sigma_{qi}}^2}{2{\sigma_{qi}}^2} + \ln\frac{\sigma_{qi}}{2\sigma_{qi}}\right] - \frac{1}{2}$$

$$= \left[\frac{(\mu_{pi} - \mu_{qi})^2}{2{\sigma_{qi}}^2} + 2 + \ln\frac{1}{2}\right] - \frac{1}{2}$$

$$= \left[\frac{(\mu_{pi} - \mu_{qi})^2}{2{\sigma_{qi}}^2} + 2 + \ln\frac{1}{2}\right] - \frac{1}{2} \qquad \text{B- 4}$$

which has a minimum value of 0.81 when the means overlap, but cannot be zero. The effect in the size of the uncertainty estimates on the minimum KLD value is demonstrated in Table B 2. The calculations show that KLD penalises more harshly the distributions 1) with a difference of the mean that is greater than measurement uncertainty estimate and 2) the prediction uncertainty estimates that exceed the measurement uncertainty estimates more than 2-fold and 3) the prediction uncertainty estimates that are smaller than the uncertainty estimates of the measurement.

B 2. Effect in the relationship between the means and standard deviation of the measured and predicted Gaussian probability distributions, N(μ_{pi}, σ_{pi}) and N(μ_{qi}, σ_{qi}) respectively, on the minimum KLD value

	$\boldsymbol{min\ D_{KL}}$			
	$(\mu_{pi} - \mu_{qi})^2 = 0$	$(\mu_{pi} - \mu_{qi})^2 = {\sigma_{qi}}^2$	$(\mu_{pi} - \mu_{qi})^2 = 4{\sigma_{qi}}^2$	$(\mu_{pi} - \mu_{qi})^2 = 9{\sigma_{qi}}^2$
$\sigma_{pi} = \frac{1}{3}\sigma_{qi}$	0.65	1.15	2.65	5.15
$\sigma_{pi} = \frac{1}{2}\sigma_{qi}$	0.32	0.82	2.32	4.82
$\sigma_{\mathrm{pi}} = \sigma_{\mathrm{qi}}$	0.00	0.50	2.00	4.50
$\sigma_{pi} = 2\sigma_{qi}$	0.81	1.31	2.81	5.31
$\sigma_{pi} = 3\sigma_{qi}$	2.90	3.40	4.90	7.40