Developing Novel Scoring Functions for Protein-Ligand Docking Using Machine Learning

Fergus Boyles

Supervised by Prof. Garrett M. Morris

Co-supervised by Prof. Charlotte M. Deane

Brasenose College

Department of Statistics

University of Oxford

Michaelmas 2019

A thesis submitted for the degree of Doctor of Philosophy.

WWW.OX.AC.UK

First edition, July 29, 2020

Statement of Originality

I, the undersigned, declare that this is my own work unless where otherwise acknowledged and referenced.

Candidate Fergus Boyles

Signed _____

Date July 29, 2020

Acknowledgements

Thanks to both of my supervisors; Garrett and Charlotte. Without your guidance, I would not have a DPhil thesis to be proud of. By encouraging me to push the limits of my ability over the course of my DPhil, you have helped me to discover that I am capable of far more than I previously thought. I am immensely grateful for your support over what turned out to be a busy and turbulent several years, and it has been an absolute pleasure to work with you.

Thanks to all the members of OPIG, past and present. Not only have you created a wonderful environment for learning and doing incredible science, but the memes and nerf wars have all been on point. Special thanks to former OPIGlet Jin, who nobly sacrificed his home directory in the name of testing my tutorial materials.

Thanks to everybody in the Department of Statistics for being such a warm and welcoming community. Special thanks to Jennifer Rogers for immensely helpful discussions about statistical testing, and to everyone in IT support for putting up with my esoteric software requests.

Thanks to Teviot gang for the years of wit and entertainment; for Pochinki drops and space trucking simulator; for trolley problems and the Pokémon drinking game. Though I graduated from The University of Edinburgh in 2015 and left that beautiful city behind, the friends I made have remained a universal constant in my life.

Thanks to everybody who has made the last four years in Oxford some of the best fun I've had in my life. Thanks to Brian, for persuading me to get a motorbike; and to Pete, for encouraging him. Thanks to the guys and girls of OUPLC for shouting at me while I picked up heavy things. Thanks to the Brasenose College HCR for four years of games night, coffee machine drama, and Game of Thrones disappointment.

Finally, and most of all, thanks to my father, Allan. Without your endless support and encouragement, I wouldn't be writing the last page of my DPhil thesis right now. You may not be around to see me finish, but you've been with me every step of the way. Thank you, for everything.

Abstract

Structure-based drug discovery uses information about the structure of a protein to identify novel ligands that bind to the protein. The fundamental problem in structure-based drug discovery is predicting if, how, and how strongly a possible ligand binds to a protein. This is often accomplished using scoring functions to rapidly estimate the strength with which a ligand binds to a protein – its binding affinity.

This thesis explores the use of machine learning techniques to improve scoring functions for protein-ligand binding affinity. We first analysed the features used by several published machine learning scoring functions, before showing that augmenting these features with ligand-based features can improve scoring function performance. We then compare the performance of different machine learning algorithms.

We next perform a series of experiments to investigate how the size and composition of the training set, and its similarity to the test set, influences the performance of Random Forest scoring functions. We find that regardless of training set composition, augmenting structure-based feature sets with additional ligand-based features leads to enhanced scoring function performance on a diverse test set. We further investigate the predictions of a Random Forest using only ligand-based features, and find that, when a ligand has different binding affinities for multiple binding partners, this ligand-only model is predictive of the mean binding affinity of a ligand for its binding partners.

Finally, we address the use of docked poses for the ligand instead of experimentally-determined binding modes. We find that pose prediction errors are common. We show that using docked poses in place of crystallographic binding modes reduces scoring function performance, and that augmenting a structure-based scoring function with ligand-based features can help to counteract this effect. We then construct a new data set and show that generalising to new data and novel targets remains challenging for machine learning scoring functions.

In this thesis we examine whether the use of a more detailed representation of the physicochemical properties of a ligand can improve machine learning scoring functions for protein-ligand binding affinity

Contents

List of Figures

List of Tables

List of Abbreviations

CADD Computer-aided drug design

CNN Convolutional neural network

CV Cross-validation

DUD Directory of Useful Decoys

DUD-E Directory of Useful Decoys – Enhanced

ECFP Extended connectivity fingerprint

FP Fingerprint

HTS High throughput screening

LBVS Ligand-based virtual screening

NN Neural network

PCA Principal component analysis

PCM Proteochemometric modelling

PDB Protein Data Bank

RBF Radial basis function

RF Random Forest

SBVS Structure-based virtual screening

SF Scoring function

SVM Support vector machine

VS Virtual screening

1

Introduction and Background

1.1 | Introduction

Drug discovery is the process of finding new medicines. This most commonly takes the form of identifying competitive inhibitors – small organic molecules, or 'ligands', that interact specifically with a protein target of therapeutic interest, usually with the effect of inhibiting the function of the protein. The drug discovery process is a slow, expensive, and financially risky undertaking. Recent reports estimate the cost of bringing a new medicine to market to be in excess of $1 billion USD (Avorn, 2015; DiMasi et al., 2016), and a development period of over 10 years from target identification to the conclusion of clinical trials (Taylor, 2015). This time and resource investment is no guarantee of success: much of the cost associated with drug development can be attributed to late-stage attrition where candidate drugs failing in clinical trials due to unanticipated problems relating to safety or toxicology (Waring et al., 2015). The high cost and low productivity in drug development is a long-standing and well-studied problem (Myers and Baker, 2001; Paul et al., 2010), for which a solution has yet to be found.

Motivated by the need to manage the rising costs of drug development, interest has grown in the use of computational methods to address these challenges. By reducing the need for slow, costly *in vitro* and *in vivo* studies and shifting toward rapid and less expensive *in silico* experiments, computer-aided drug design (CADD) offers the potential to expedite the drug discovery process and reduce development costs (Shekhar, 2008; Ou-Yang et al., 2012) while also providing a greater understanding of the molecular processes governing the behaviour of a candidate drug (Macalino et al., 2015).

A fundamental aim of CADD is to predict if a ligand binds to a protein, and if so, predicting the binding affinity of that ligand for the protein. Ideally, we want a potent molecule so as to reduce the required dosage. In addition to performing experimental assays to investigate binding, the binding affinity of a ligand for a protein can sometimes be predicted accurately using free energy perturbation methods with molecular dynamics or Monte Carlo simulations of the protein-ligand system (Deng et al., 2004; Chodera et al., 2011). However, these approaches are laborious and computationally expensive, and often require manual parameterization of the ligand, making them sub-optimal for screening large compound libraries in the early stages of drug discovery. Instead, the computationally-efficient approach of protein-ligand docking can be used to generate hypothetical binding poses of each ligand, and their affinity can be estimated using rapid scoring functions designed to evaluate protein-ligand interactions with little computational cost.

Much of this work rests on our knowledge of the phenomenon of molecular recognition: the ability of chemical and biological systems to distinguish between different molecules and interact selectively, thereby regulating the be-

haviour of the system. Molecular recognition relies on a wide range of non-covalent interactions between molecules, such as hydrogen bonds, van der Waals forces, and electrostatic interactions. An understanding of molecular recognition, and how best to represent it computationally, is thus a vital part of accurately predicting binding affinity. In identifying those features of proteins and small molecules that enable us to predict binding affinity accurately, we may be able to gain new insights into molecular recognition.

In this Chapter, we discuss the concept of molecular recognition, and how the kinematics of protein-ligand binding relates to the free energy of binding. We then briefly summarise the drug discovery process and discuss how computational methods are used in the early stages of drug discovery to quickly screen large virtual libraries of compounds. We introduce protein-ligand docking, and discuss the strengths and limitations of this computationally-efficient method of modelling protein-ligand binding. We review the scoring functions used in protein-ligand docking and describe how the introduction of machine learning techniques has led to a new class of scoring functions that have improved the accuracy of protein-ligand binding affinity prediction. We describe how scoring functions are evaluated, introducing structural and binding affinity validation sets and benchmarking exercises found in the literature. We finish by outlining the structure of and key contributions of this thesis.

1.2 | Molecular Recognition

Molecular recognition is the specific interaction between two or more molecules through noncovalent bonding driven by intermolecular forces such as van der

Waals, electrostatics, and hydrogen bonding interactions, and is the physical principle underlying protein-ligand interactions. The earliest model of molecular recognition was that of the "lock and key", hypothesised by Emil Fischer (1894). In this model, the active site of a host molecule has a specific fixed shape. Just as a key fits a lock, a guest molecule will only fit into the active site and bind if its shape is an exact complement to the shape and chemistry of the active site. This does not, however, take account of the flexibility of biological molecules. The induced fit model of molecular recognition proposed by Koshland (1958) instead suggests that changes in the conformation of a protein or ligand occur when each molecule is in the presence of the other, allowing initially incompatible molecules to adopt conformations more conducive to binding.

An alternative hypothesis to the induced fit model is that of conformation selection (Monod et al., 1965; Rubin and Changeux, 1966) in which different conformations of the protein and ligand exist independently, and binding occurs between complementary conformations. There has been much debate about whether the induced fit or conformation selection models more accurately describe molecular recognition (Changeux and Edelstein, 2011), and indeed how to determine which mechanism is at play in a given interaction (Hammes et al., 2009; Gianni et al., 2014). Both models have been implemented in docking protocols, however due to limitations in computational power, use of the induced fit model is often limited to allowing a small number of flexible side chains around the active site of the protein to move. Similarly, docking against an ensemble of conformations is limited by the cost of running a full docking simulation for each conformation of the protein and, where necessary, using tools such as molecular dynamics to generate the ensemble of protein conformations (Totrov and

Abagyan, 2008; Amaro et al., 2018).

1.2.1 | Free Energy of Binding

If we treat the bound complex and the unbound protein-ligand pair as two separate systems, then the strength of the binding is characterised by the difference in the Gibbs free energy of the two systems,

$$\Delta G = G_{\text{bound}} - G_{\text{unbound}}. \tag{1.1}$$

The difference in free energy is known as the free energy of binding, and results from an interplay of the potential energy associated with the various interactions involved in the binding, as well as entropic contributions due to the flexibility and motion of the molecules. In practice, solvation effects resulting from hydrophilic and hydrophobic contacts on the surface of the molecules being buried in the binding pocket rather than exposed to solvent also contribute to the difference in free energy. A complex with a highly negative free energy of binding is strongly bound, while a complex with a less negative free energy of binding is weakly bound and easily separated.

The Gibbs free energy of a thermodynamic system, G, is the internal energy of the system available to do work, and is defined as:

$$G = H - TS \tag{1.2}$$

where H is the enthalpy, S is the entropy, and T is the absolute temperature of the system. Enthalpy comprises the internal energy of a system plus the work done in making room for the system by displacing the surrounding environment. Entropy is a measure of the 'disorder' of the system and is directly related

to the number of microstates of the system, W, by the Boltzmann equation:

$$S = k_{\mathrm{B}} \ln W \tag{1.3}$$

where k_{B} is the Boltzmann constant.

At equilibrium, and under standard conditions, for an infinitesimal change at fixed temperature the change in free energy is given by:

$$\Delta G = \Delta H - T\Delta S, \tag{1.4}$$

so the binding interaction can be viewed as a valance between enthalpic and entropic changes upon binding. For many systems, changes in enthalpy are observed to occur alongside opposing changes in entropy, resulting in little change in the free energy. This phenomenon is known as the enthalpy-entropy compensation, and is often cited as an explanation for the behaviour of closely-related chemical systems (Dunitz, 1995; Qian, 1998). It has been suggested that overcoming enthalpy-entropy compensation is a significant limiting factor in binding affinity optimisation (Lafont et al., 2007; Freire, 2008; Reynolds and Holloway, 2011). However, there is much debate in the literature over whether enthalpy-entropy compensation is a physical phenomenon, or simply a mathematical artefact (Sharp, 2001; Starikov and Nordén, 2007), and consequently whether pursuing affinity optimisation directly is more appropriate than enthalpic and entropic optimisation (Geschwindner et al., 2015).

1.2.2 | Binding Kinematics

The free energy of binding can be related to the kinematics of protein-ligand binding as follows. For a protein, P, and ligand, L, we may write the following

reaction:

$$\mathrm{P} + \mathrm{L} \underset{k_{\mathrm{off}}}{\overset{k_{\mathrm{on}}}{\rightleftharpoons}} \mathrm{PL} \tag{1.5}$$

where k_{on} and k_{off} are the rate constants for the binding and unbinding reactions, respectively. At equilibrium, the binding and unbinding reactions are balanced, so we can write

$$k_{\mathrm{on}}[\mathrm{P}][\mathrm{L}] = k_{\mathrm{off}}[\mathrm{PL}] \tag{1.6}$$

where [L], [P], and [LP] are, respectively, the concentrations of unbound ligand, unbound protein, and bound protein-ligand complex, usually given in moles per litre, or mol L^{-1}. The dissociation constant K_{d} is then defined as the ratio of unbound to bound molecules, and has units of concentration:

$$K_{\mathrm{d}} = \frac{[\mathrm{L}][\mathrm{P}]}{[\mathrm{LP}]} = \frac{k_{\mathrm{off}}}{k_{\mathrm{on}}} \tag{1.7}$$

For a competitive inhibition process, where an inhibitor I binds to the protein, P, the inhibition constant K_{i} is defined equivalently to K_{d} as the ratio of the rate constants:

$$K_{\mathrm{i}} = \frac{[\mathrm{I}][\mathrm{P}]}{[\mathrm{IP}]} = \frac{k_{\mathrm{off}}}{k_{\mathrm{on}}}. \tag{1.8}$$

A small inhibition or dissociation constant thus corresponds to a tightly-bound ligand for which the rate of dissociation is much lower than the rate of binding.

A related quantity to the inhibition constant is the half-maximal inhibitory concentration (IC_{50}). This is the concentration of the inhibiting ligand at which the activity of the target protein is inhibited by half, and is related to the inhibition constant via the Cheng-Prusoff equation (Cheng and Prusoff, 1973; Lazareno and Birdsall, 1993),

$$K_{\mathrm{i}} = \frac{\mathrm{IC}_{50}}{1 + \frac{[\mathrm{A}]}{\mathrm{EC}_{50}}} \tag{1.9}$$

where [A] is the concentration of the agonist[1] used in the assay. EC_{50} is the equilibrium concentration of agonist at which 50% of the target protein is activated. Obtaining the inhibition constant from an IC_{50} measurement therefore requires knowledge of the experimental conditions under which the measurement was obtained. For this reason, measurements of IC_{50} are less useful than measurements of K_d or $K_textsubscripti$ for the development of protein-ligand scoring functions as it is not particularly meaningful to simply compare IC_{50} values for different ligands without knowledge of the assay conditions.

The free energy of binding is related to the dissociation constant by the Gibbs equation:

$$\Delta G = RT \ln \frac{K_\mathrm{d}}{\mathrm{c}^0} \tag{1.10}$$

where 1.987×10^{-3} kcal K^{-1} mol^{-1} is the ideal gas constant and T is temperature. $C^0 = 1$ mol L^{-1} is the standard concentration, and is required for the argument of the logarithm to be dimensionless. Since the value of K_i or K_d varies over many orders of magnitude, the negative base-10 logarithm of the binding constant is often reported. This is referred to as the pK of the molecule:

$$\mathrm{p}K = -\log_{10} K_\mathrm{d} \tag{1.11}$$

Table 1.1 shows the free energy of binding in kcal mol^{-1} for different values of the dissociation constant, ranging from weakly (millimolar affinity) to strongly (femtomolar affinity) binding. Ligands with affinity greater than picomolar are rare (Kuntz et al., 1999), but exceptional cases exist, such as the femtomolar

[1] A substance which binds to the target to activate it. In this context the inhibitor is referred to as the 'antagonist' as the inhibitory effect is opposite in function to that of the agonist. The agonist and antagonist bind competitively *i.e.* only one can be bound to the target at any one time.

ΔG (kcal mol^{-1})	K_d (mol L^{-1})	pK	Affinity	Units
-4.09	10^{-3}	3	millimolar	mM
-8.18	10^{-6}	6	micromolar	μM
-12.28	10^{-9}	9	nanomolar	nM
-16.37	10^{-12}	12	picomolar	pM
-20.46	10^{-15}	15	femtomolar	fM

Table 1.1: Free energy of binding corresponding to different values of the dissociation constant. Values were computed using Equation 1.10 assuming a temperature of 298.15 K

($K_d \approx 10^{-15}$mol L^{-1}) affinity of biotin for the protein avidin (DeChancie and Houk, 2007).

1.3 | The Drug Discovery Process

Drug discovery begins with the identification and verification of a therapeutic target of interest. This is usually a protein, DNA, or RNA molecule important to the development of the disease. Possible targets are identified through biochemical studies and, more recently, disease-related genomics - the study of the expression of genes and synthesis of the proteins they encode, and the role they play in the disease.

Once a target has been verified to play a key role in the disease, the next stage is assay development, which is followed by hit identification. Available compounds are tested for activity against the target in high-throughput screens, with compounds found to display activity known as 'hits'. These hits are then evaluated through further testing and assays to confirm their activity and test for necessary properties such as bioavailability and lack of toxicity. Promising hits will undergo medicinal chemistry optimisation to improve selectivity

and metabolism. The molecules selected at the end of this 'hit-to-lead' stage are known as 'lead molecules' or 'leads'.

Next, in the lead optimisation phase, the structure of each lead molecule is studied and modified to increase its affinity for the target. Sometimes, elimination of off-target effects is important, as well as optimisation of desirable properties such as bioavailability, metabolism, and stable storage will also be carried out, while retaining potency.

Once the lead molecules have been optimised, pre-clinical trials using both cell culture *in vitro* and animal *in vivo* models are used to better understand the behaviour of the compounds and ensure that they are safe to begin human trials. At this stage, compounds selected to proceed to clinical trials are known as 'candidate drugs'. Finally, the candidate drugs undergo several stages of clinical trials to determine whether they display the efficacy and safety required for human use.

1.4 | Virtual Screening

In drug discovery projects, large libraries of in-house compounds are routinely tested for activity against target proteins in a process known as high-throughput screening (HTS). HTS has traditionally been used to search for active compounds, however the low hit rate and high resource cost of this approach make it desirable to use computational techniques to identify the most promising drug candidates for experimental investigation (Bleicher et al., 2003). Virtual screening (VS) is a technique used to complement HTS by computationally searching through large ligand libraries for novel compounds with some affinity for the target pro-

tein (Walters et al., 1998; Lavecchia and Di Giovanni, 2013). The molecules identified during the virtual screening process can then be prioritised for further investigation, and if confirmed as hits, proceed to lead optimisation (Jorgensen, 2009).

VS techniques have historically been broken down into two categories: ligand-based and structure-based (Sliwoski et al., 2014). Ligand-based virtual screening (LBVS) relies on using knowledge of known active (and, for some methods, inactive) compounds to identify novel molecules likely to bind to the target of interest. These approaches tend to be computationally inexpensive and can be used when a reliable three-dimensional structure of the target is unavailable. In contrast, structure-based virtual screening (SBVS) uses three-dimensional structures of the target to examine how each ligand is expected to interact with the target. One of the most common approaches in SBVS involves using protein-ligand docking to predict the binding mode of the ligand in the binding pocket of the target. The three-dimensional model of the protein-ligand complex is then used to assess the strength of the interaction and rank the ligands. SBVS has the advantage of making direct use of information about the target, and does not require knowledge of any active molecules. A drawback of SBVS is the dramatically increased computational cost associated with docking each ligand into the target structure.

1.4.1 | Ligand-Based Virtual Screening

Modern LBVS techniques are an evolution in many ways of quantitative structure-activity relationship (QSAR) modelling, in which the physicochemical proper-

ties or theoretical descriptors of a molecule are used to construct a quantitative model of the activity of the molecule. QSAR modelling has itself seen numerous applications in drug design and drug discovery (Kubinyi, 1997a,b).

In the case where binding affinity data are available for known active compounds but not for inactives, there are generally two methods which are employed: pharmacophore modelling and similarity searching. Pharmacophore modelling consists of identifying 'pharmacophoric' (electronic and steric) features of the known actives. By selecting those features common to a representative set of actives it is possible to build up a model of those pharmacophoric features likely to be important for a ligand to bind to the target (Sun, 2008). By superimposing the pharmacophores of an unseen ligand on the model generated from the known actives and quantifying how similar the features are, it is possible to rank a set of ligands by their similarity to the pharmacophore model.

Another option is chemical similarity searching (Willett, 1998) where rather than comparing spatial features of compounds, a set of descriptors are computed for each molecule. By defining a meaningful distance metric for differences in the chosen descriptors it is then possible to rank ligands by their similarity to known actives. Perhaps the most commonly-used approach to similarity searching is the use of molecular fingerprints that encode the chemical structure of the model in a vector representation. Today, there are many different fingerprints used as ligand descriptors, and their application to virtual screening is well-studied (Cereto-Massagué et al., 2015). Even if the 3D structure of the target is known it still makes sense to take advantage of any available information. As such, a ligand-based virtual screen might be valuable as a precursor to a structure-based virtual screen, as ligand-based methods are much less compu-

tationally intensive and so can help to reduce the number of ligands used for structure-based screening.

Finally, if binding data are available for both active and inactive compounds, another option is to use supervised machine learning techniques to classify ligands as active or inactive based on a chosen set of descriptive features. Ligand-based virtual screening has been accomplished using a variety of techniques, including support vector machines (Jorissen and Gilson, 2005; Mahé et al., 2006; Hinselmann et al., 2011), Random Forest (Svetnik et al., 2003, 2004), and artificial neural networks (Betzi et al., 2006).

1.4.2 | Structure-Based Virtual Screening

In SBVS, a model of the three-dimensional structure of the protein target is used to propose a binding mode and/or assess the strength of the interaction between the small molecule and the protein (Lyne, 2002). The goal of SBVS is to perform two key tasks: predicting the position and orientation (and if it is flexible, conformation) of a small molecule with respect to the target (the 'binding pose' or 'binding mode');sc and estimating the binding affinity, or free energy of binding, of the bound complex. Unlike LBVS, which relies on knowledge of active molecules, SBVS can be used without knowing any active molecules for the target.

1.4.3 | Protein-Ligand Docking

The most commonly-used technique in SBVS is protein-ligand docking. A docking protocol consists of a search algorithm which samples a 'search space' de-

scribing the possible binding modes or 'poses' of the ligand with respect to the target; and a scoring function which ranks the proposed poses with the aim of both identifying the experimentally observed binding mode and estimating the affinity of the complex (Kitchen et al., 2004; Huang and Zou, 2010; Meng et al., 2011; Cheng et al., 2012). The success of docking is dependent both on the efficiency and reliability with which the search samples, positional, orientational, and conformational space, and the accuracy with which the scoring function estimates the binding affinity or discriminates between binders and non-binders.

While there has been great progress, one of the major remaining challenges in protein-ligand docking is incorporating receptor flexibility in a computationally-efficient manner. Early docking methods, such as DOCK (Kuntz et al., 1982), treated both the ligand and receptor as rigid bodies. AutoDock was the first reported docking software to model the ligand as flexible (Goodsell and Olson, 1990). In contrast, modern docking methods allow for flexibility in both the ligand and receptor, and have been used to demonstrate that accounting for induced fit effects results in improved docking performance (Rosenfeld et al., 2002; Meiler and Baker, 2006; Nabuurs et al., 2007; Barreca et al., 2009; Ding et al., 2010), yet despite advances in this direction, modelling receptor flexibility remains challenging (Lexa and Carlson, 2012; Yuriev et al., 2015)

Despite the challenges posed by modelling ligand and receptor flexibility, there has been great progress in the development of docking algorithms, with modern docking software often capable of reproducing experimentally-observed binding poses. In contrast, predicting the binding affinity of a protein-ligand complex, even when the binding pose is known, remains extremely challenging. In retrospective benchmarking studies (Cheng et al., 2009; Li et al., 2014c,b;

Su et al., 2018) and community exercises using unseen data (Smith et al., 2011; Damm-Ganamet et al., 2013; Smith et al., 2016; Carlson et al., 2016; Gathiaka et al., 2016; Gaieb et al., 2019), the ability of docking scoring functions to predict protein-ligand binding affinity is poor.

1.4.4 | Combining Virtual Screening Methods

While LBVS and SBVS have different sets of requirements, their use is not mutually exclusive. On the contrary, LBVS and SBVS methods are often viewed as complementary, and combinations of ligand-based and structure-based methods have been successfully applied in virtual screening projects (Drwal and Griffith, 2013).

Sequential approaches, in which computationally less-expensive ligand-based methods are used as a pre-filter for computationally more-expensive structure-based methods, have the advantage of minimising the computational cost associated with structure-based methods by reducing the number of molecules screened, and have been successfully applied to virtual screening projects (Houston et al., 2015; Floresta et al., 2018). One disadvantage of this approach is that mis-classification at any step can result in an active molecule being removed from the pipeline, even if subsequent steps would have correctly identified it as active.

Parallel approaches, in which different structure-based and ligand-based methods are applied to the full set of molecules, have also proven successful (Svensson et al., 2011; Swann et al., 2011; Joshi et al., 2017). In contrast with a sequential approach, a parallel approach allows for the strengths of different, complemen-

tary methods to be combined at the expense of an increase in the computational cost of screening each molecule.

Finally, hybrid approaches in which ligand-based and structure-based information are combined in a single method have also been developed (see for example Anighoro and Bajorath (2016)). The optimal approach for combining different methods and data remains an open question; however, several studies comparing different approaches have shown that parallel selection out-performs other data fusion approaches (Tan et al., 2008; Krüger and Evers, 2010; Svensson et al., 2011).

1.4.5 | Proteochemometric modelling

Proteochemometric modelling (PCM) is an approach to modelling protein-ligand interactions in which descriptors of the protein and the ligand are combined in a single predictive model (Lapinsh et al., 2001), which can be seen as an extension of (ligand-based) QSAR. By construction, QSAR models can only make predictions about ligands with respect to a single protein, and require data about ligands for that protein in order to build the model. Consequently, QSAR is unable to model ligand selectivity across multiple proteins or to predict the activity of ligands for a novel protein for which binding data are unavailable. Further, QSAR models often struggle to extrapolate to novel areas of chemical space not represented in the training data (Gedeck et al., 2006) and so are prone to failing to identify activity cliffs - small changes in chemical structure that result in a large change in the biological activity of the ligand (Maggiora, 2006).

In contrast to the ligand-based QSAR approach, PCM is able to model protein-

ligand interactions across both protein and ligand space simultaneously. PCM has been found to outperform QSAR when extrapolating to novel ligands (van Westen et al., 2011), and has been proven to be capable of predicting ligand selectivity (Ain et al., 2014) and capturing polypharmacology (Paricharak et al., 2015). For recent reviews of methodological advances and applications of PCM, the reader is referred to Cortés-Ciriano et al. (2015) and Qiu et al. (2016). The success of PCM in a range of applications demonstrates the value of integrating protein-based and ligand-based approaches when modelling protein-ligand interactions.

1.5 | Scoring Functions for Protein-Ligand Binding Affinity

Scoring functions are approximate, computationally-efficient methods for assessing molecular interactions. The scoring functions used in protein-ligand docking can generally be classified as one of four types according to the information and method used to score the ligand or predict its binding affinity for a given target. These four types are: force-field, knowledge-based, empirical, and machine learning.

Force-field scoring functions (Huang et al., 2006a,b; Genheden and Ryde, 2015) attempt to directly estimate the change in free energy upon binding by summing the terms of a molecular mechanics force field, such as van der Waals and electrostatic interactions between the two molecules. Other contributions to the interaction energy, such as the effects of a solvent on exposed hydrophilic

and hydrophobic atoms, are also often included. This approach has the advantage of being grounded in well-understood physical models without depending on experimental data to fit a model. However, it ignores statistical mechanics and fails to compute entropic contributions, with the exception of approximations of the desolvation of the protein and ligand upon binding. Although detailed molecular mechanics approaches have the potential to predict accurately the free energy of binding of a system, concerns about reproducibility and the computational cost of accurate calculations can limit the applicability of these techniques (Guvench and MacKerell Jr, 2009; Mobley, 2012).

Knowledge-based scoring functions, also known as statistical potentials, are derived from the distribution of pairwise distances between atoms in structures found in the Protein Data Bank, which are are then converted into an energy function using potentials of mean force (Miyazawa and Jernigan, 1985; Sippl, 1990). Although this technique is perhaps more commonly associated with protein structure prediction, it has seen numerous applications to the prediction of protein-ligand binding affinity (DeWitte and Shakhnovich, 1996; Morris et al., 1998; Muegge and Martin, 1999; Gohlke et al., 2000; Gohlke and Klebe, 2001; Ozrin and Subbotin, 2004; Mooij and Verdonk, 2005).

Empirical scoring functions consist of a weighted sum of physically-meaningful terms, examples of which might include force-field or force-field-like potential terms, the presence of hydrogen bonds, the molecular weight of the ligand, or entropic penalties due to the size and flexibility of the ligand. The weighting of these terms might be determined by regression using experimental binding data for known protein-ligand complexes or assigned manually. This approach was pioneered by Böhm with LUDI (Böhm, 1992, 1994, 1998) and adopted in

the scoring function of AutoDock 3 (Morris et al., 1998), and has since become a popular approach for rapidly estimating binding affinity (Eldridge et al., 1997; Gilson et al., 1997; Gilson and Zhou, 2007; Li et al., 2013). In particular, empirical scoring functions are commonly used in protein-ligand docking both for ranking binding poses and for affinity prediction (Morris et al., 2009; Trott and Olson, 2010; Friesner et al., 2004, 2006; Baek et al., 2017). Although most docking software and empirical scoring functions are intended for general-purpose use, variants of these scoring functions have been developed to represent a specific system by more accurately modelling specific interactions. An example of this approach is the work of Laederach and Reilly, who adapted AutoDock 3 for protein-carbohydrate complexes. This led to a reduction in a reduction in the residual standard error on the free energy of binding from 2.070 kcal/mol using the original AutoDock 3 scoring function to 1.101 kcal/mol using the adapted scoring function (Laederach and Reilly, 2003). Further examples of this include XBScore (Zimmermann et al., 2015) and AutoDock VinaXB (Koebel et al., 2016), both of which model halogen bonding, and AutoDock4$_{Zn}$ (Santos-Martins et al., 2014), which explicitly models zinc ion coordination, resulting in substantial improvements in the docking of ligands to zinc metalloproteins.

Common to these approaches is the *a priori* assumption of the functional form of the relationship between the predictive features and the binding affinity. In contrast to this, a range of machine learning methods have been developed to predict binding affinity and use a diverse range of representations of the protein-ligand complex. This deviates from more traditional approaches in that it is not necessary to assume the functional form of the relationship between descriptive features and binding affinity. Instead, machine learning methods are able to in-

fer the relationships between features and binding affinity directly from the data they are trained on. Indeed, deep learning algorithms can be used to extract features from the underlying data, albeit as a 'black box'. Because of this difference in methodology, force-field, statistical, and empirical scoring functions are commonly referred to collectively as 'classical' scoring functions to distinguish them from those built using machine learning algorithms.

1.5.1 | Machine Learning Scoring Functions

Beginning around 2004, there has been a growing interest in the use of machine learning techniques to develop more accurate scoring functions (Deng et al., 2004; Zhang et al., 2006). Numerous studies have been published reporting improved performance over classical scoring functions by either using the small number of features of existing scoring functions as input to a machine learning algorithm, or by the use of other sets of descriptors.

In 2010, Durrant and McCammon published NNScore (Durrant and McCammon, 2010), which uses a set of 194 features characterising a protein-ligand complex as input to a neural network with a single hidden layer. Although trained for virtual screening, the classification probabilities output by the resulting model were found to rank accurately the relative binding affinities of ligands, suggesting that the neural network model could also be used as a post-docking scoring function for affinity prediction. A neural network was also successfully used by Betzi et al. (2006) to combine several scoring functions in a consensus model. The authors reported enhanced virtual screening performance but did not address the affinity prediction problem. In 2011, Durrant and

McCammon published NNScore 2.0 (Durrant and McCammon, 2011b), an updated version of NNScore using a more detailed characterisation of the protein-ligand complex computed using the BINANA algorithm (Durrant and McCammon, 2011a) together with the features of the AutoDock Vina (Trott and Olson, 2010) scoring function. Unlike NNScore, NNScore 2.0 was specifically trained for the task of binding affinity prediction, and was found to outperform both the AutoDock Vina scoring function and a neural network using the AutoDock Vina features alone.

One particularly popular machine learning algorithm for scoring function development is Random Forest (RF) (Breiman, 2001), an ensemble learning algorithm based on averaging the predictions of a large number of uncorrelated decision tree estimators. In 2010, Ballester and Mitchell published RF-Score (Ballester and Mitchell, 2010), which used a set of thirty-six descriptors comprising counts of protein-ligand atom pairs in close proximity as input to a RF. The resulting scoring function was found to outperform classical scoring functions when evaluated on a held-out test set. In 2013, Zilian et al. (2013) used the descriptors of the classical scoring function SFScore (Sotriffer et al., 2008) as input features for a RF, resulting in more accurate affinity predictions. In 2014, Li et al. (2014a) showed that substituting RF for linear regression in the empirical scoring function Cyscore (Li et al., 2014a) led to improved affinity predictions, and that the RF was able to continue to learn as more training data were added while the performance of the linear regression hit a plateau. In 2014, Ballester et al. published RF-Score v2 as part of a study that showed that, perhaps counterintuitively, a 'more detailed' description of the protein-ligand complex (in this case, binning the atom-pair counts by separation and treating each binned count as a sepa-

rate descriptor) did not necessarily lead to more accurate predictions of binding affinity (Ballester et al., 2014). In 2015, Li et al. (2015b) showed that combining the atom-pair counts of RF-Score with the features of AutoDock Vina led to improved predictions of binding affinity, resulting in a new version of the scoring function, RF-Score v3. Even more recently, Wójcikowski et al. (2017) re-trained each version of RF-Score for virtual screening (*i.e.* binary classification of compounds as active or inactive). The resulting classifiers significantly outperformed classical docking scoring functions when examples of ligands for the same target were included in both the training and test data (a so-called 'horizontal split'); however, when the training and test sets contained different protein targets ('vertical split'), the performance of the RF classifiers was much closer to that of the classical scoring functions. This suggests that the performance of RF-Score in a virtual screening context is strongly dependent on the existence of previously-seen examples for the target of interest.

Deep Learning Scoring Functions

More recently, there has been a growing interest in the use of deep learning for pose prediction, affinity prediction, and virtual screening. In 2017, Ragoza et al. published two studies using convolutional neural networks to score protein-ligand complexes for both pose-prediction and virtual screening (Ragoza et al., 2017a,b). Inspired by the success of convolutional neural networks in image processing tasks, the authors treated scoring as an image recognition problem by representing each protein-ligand complex as a set of three-dimensional atom density maps. By computing the density of each atom species on a regularly-

spaced grid, a discretised representation of the protein-ligand complex's structure was generated. This representation was analogous to the red-green-blue colour channels of pixels in an image, but with the different atom types representing different colour channels.

This approach performed well at both pose prediction and virtual screening, though a convolutional neural network trained on a pose prediction data set tends to perform poorly at virtual screening and *vice versa*. The authors further showed that when trained on a mixture of pose prediction and virtual screening data, the convolutional neural network performs moderately well at both tasks, but not as well as when trained for either task alone. Their work builds on AutoDock Vina by re-scoring docked conformations generated by Vina. Interestingly, the authors note that while their convolutional neural network outperforms the Vina scoring function at inter-target pose prediction (prediction across multiple targets), it is outperformed by the Vina scoring function at intra-target pose prediction (prediction for a single target), which is the more likely scenario when docking. It is suggested that this is a result of their training protocol only considering the RMSD (root-mean-square deviation) of a docked pose and that intra-target pose prediction might be improved by weighting poses by the binding affinity of the ligand to prioritise correctly identifying the binding mode of active compounds. Although Ragoza et al. do not attempt to directly predict binding affinity, studies by Gomes et al. (2017), Jiménez et al. (2018), and Stepniewska-Dziubinska et al. (2018) found that a similar use of convolutional neural networks for affinity prediction achieved competitive results. More recently, Imrie et al. (2018) demonstrated that the methodology of Ragoza et al. can be enhanced by applying recent advances in the design of convolutional

neural network architecture, suggesting that advances in computer vision are directly transferable to the problem of virtual screening.

1.5.2 | Scoring Function Validation

Despite the rapid advances in scoring function development through the application of machine learning, concerns have been raised about the validation and interpretability of such models. Kramer and Gedeck (2010) showed that the performance of RF-Score drops substantially when applied to protein targets not present in the training data. In addition, Li and Yang (2017) and Li et al. (Li et al., 2018) found that the performance of RF-Score is strongly dependent on whether the training set includes examples of proteins with similar structure and sequence to those in the test set, suggesting that the scoring function can be expected to generalise poorly to a previously-unseen target. Gabel et al. (2014) showed that the performance of RF-Score was not significantly impacted by the choice of distance at which atom-pairs were counted. In particular, even when atom-pairs are only counted if their separation was between 10-12Å, the scoring function's performance was not significantly impacted, suggesting that the RF-Score descriptors do not, in fact, allow the RF to correctly identify physical interactions between the protein and ligand. These concerns are not limited to RF-Score, or even to affinity prediction in general. Indeed, Sieg et al. (2019) demonstrated that convolutional neural network virtual screening models are susceptible to bias and artificial enrichment in popular virtual screening benchmark sets such as DUD-E (Mysinger et al., 2012).

The ultimate test of the usefulness of a scoring function is its ability to en-

rich hit rates in a virtual screening campaign and prioritise strong binders over weaker ones. However, such prospective validation is often beyond the reach of academic research since it requires not only conducting a virtual screening campaign on a novel set of targets and/or compounds but also experimental validation of the predicted active compounds. This an expensive undertaking in terms of both research time and financial resources. Moreover it does not offer a readily-accessible benchmark against which researchers can test their methods to allow direct comparison with the present state of the field. In practice, scoring functions are typically assessed using publicly-available data such as crystal structures of bound protein-ligand complexes taken from the PDB (Berman et al., 2000) and bioactivity data taken from databases such as ChEMBL (Gaulton et al., 2017). However, the quantity and diversity of publicly-available data means that in the absence of an accepted standard for scoring function validation, it would be difficult to compare the results of two scoring function publications in any meaningful way. To address this issue, numerous benchmark sets and community challenges have been published with the aim of assessing the state of the field and offering robust standards by which different scoring functions can be compared.

Benchmarking Sets

Perhaps the most widely-adopted standard for scoring function comparison is the Comparative Assessment of Scoring Functions (CASF). Originally published in 2009 (Cheng et al., 2009) with subsequent updates published in 2014 (Li et al., 2014c,b) and 2018 (Su et al., 2018), CASF defines four different tests — docking

power, scoring power, ranking power, and screening power — designed to evaluate the performance of scoring functions in a range of tasks. The benchmark is based on a subset of the publicly-available PDBbind database (Liu et al., 2017), known as the 'core set', and all data associated with the exercise are made publicly available to allow researchers to directly compare methods against those tested in the exercise. A popular training and validation paradigm, particularly for machine learning scoring functions, is to hold out one or more versions of the 'core set' as an external test set, while using the remaining data from the PDBbind database for training purposes.

Several other benchmarking sets for the validation of virtual screening methods have been published in recent years. The Directory of Useful Decoys – Enhanced (DUD–E) (Mysinger et al., 2012) consists of 102 protein targets with a total of 22,886 active compounds and their affinities and was an extension of its predecessor, DUD (Huang et al., 2006c). For each active compound 50 mostly putative inactives, or 'decoys', are provided. The decoys are chosen to have similar physicochemical properties to the actives but dissimilar 2D topology, to prevent them being trivially differentiated on the basis of simple properties. Another such benchmarking set is DEKOIS 2.0 (Bauer et al., 2013): a set of 81 structurally-diverse protein targets with active compounds drawn from BindingDB (Gilson et al., 2015). Similar to DUD-E, 30 decoys are selected for each active by matching physicochemical properties while ensuring topological dissimilarity. Another popular benchmark is the set of 'Maximum Unbiased Validation' (MUV) datasets published by Rohrer and Baumann (2009). These are intended to avoid artificial enrichment of screening results by ensuring adequate embedding of active compounds among decoys in chemical space and features

experimentally verified non-binders.

Community Exercises

The Community Structure-Activity Resource ran four docking and scoring exercises in which the scientific community was invited to submit binding pose and activity predictions for a diverse set of protein targets (Smith et al., 2011; Dunbar et al., 2011; Damm-Ganamet et al., 2013; Dunbar et al., 2013; Smith et al., 2016; Carlson et al., 2016). Unlike CASF which is based entirely on publicly-available data, CSAR made use of previously-unseen in-house and industrial data; in particular, the 2014 exercise featured a large set of data donated by GlaxoSmithKline (Carlson et al., 2016). The goal of these exercises was not to compare methods and rank from 'best' to 'worst' but rather to assess the state of the field. Following the conclusion of the 2014 exercise, Carlson provided a summary of the lessons learned over the course of the four exercises (Carlson, 2016). Perhaps most illuminating are Carlson's comments on the difficulty of data set construction and the dire need for greater statistical rigour in the assessment of model performance.

More recently, the Drug Design Data Resource (D3R) was established for the purpose of sharing datasets and techniques for CADD, and regularly runs community exercises across a range of computational problems. To date, D3R has run four 'Grand Challenges': community-wide blinded exercises in binding pose and activity prediction, of which the overview and results of the first three have been published (Gathiaka et al., 2016; Gaieb et al., 2018, 2019). These challenges provide an overview of the state of the art in docking and affinity

prediction, and help to highlight challenges facing the field. The results of D3R Grand Challenge 3 suggest that although in many cases accurately predicting the binding pose of a ligand is possible, the results of docking can be inconsistent and depend on how a docking tool is used. Furthermore, when docked poses were used to predict binding affinity, the accuracy of the pose used was found to not be correlated with the accuracy of the affinity prediction, indicating that affinity prediction remains challenging even when docking accurately reproduces the binding pose (Gaieb et al., 2019).

Homology modelling in Drug Discovery

The data sets described previously all feature protein targets for which experimentally determined structures of the target are available. However, in a real-world drug discovery problem, this is not always the case. If the structure of the target has not been solved (through X-ray crystallography or NMR analysis), an alternative approach is to first use homology modelling to generate a structural model of the target. While this introduces an additional source of error, there are numerous success stories of the use of homology modelling in structure-based drug discovery (Muhammed and Aki-Yalcin, 2019). The use of modelled structures is particularly common for G-protein-coupled receptors (GPCRs), a large, diverse group of membrane proteins. While GPCRs make up a third of all FDA-approved drug targets (Hauser et al., 2017), the majority of human GPCRs do not have experimentally-solved structures available in the PDB, making them a good candidate for SBVS using homology models (Castleman et al., 2019).

Despite the prevalence of GPCRs as drug targets, and the successful use

of homology models of GPCRs in drug discovery campaigns, the application of SBVS methods to homology models remains challenging. Ferrara and Jacoby (2007) studied the use of homology models of the insulin-like growth factor 1 receptor and found that virtual screening performance varied greatly between different homology models, and that the best homology models performed worse than the best available crystal structures. Lim et al. (2018) performed a virtual screening benchmarking study across 19 human GPCRs with available crystal structures, and found that for 10 of the 19 GPCRs, virtual screening performance using a homology model matched the performance when using a crystal structure. Conversely, Loo et al. (2018) found that GPCR crystal structures consistently out-performed homology models in both binding pose prediction and virtual screening enrichment. Recent case studies in using homology models for docking have also met with mixed success (Chen et al., 2018; Pan et al., 2019; Costanzi et al., 2019), indicating that the use of homology models for SBVS remains challenging.

1.6 | Other Methods for Modelling Protein-Ligand Interactions

Although docking is the premier technique applied to structure-based drug discovery, owing to its computational efficiency, other, more computationally expensive methods are also used to explore protein-ligand interactions. Here we briefly review two such approaches: molecular dynamics and free energy perturbation.

1.6.1 | Molecular Dynamics

While protein-ligand docking and scoring functions enable rapid generation of the likely binding pose and estimates of binding affinity for large libraries of compounds, the models of molecular recognition used are simple approximations of the underlying physical processes, often relying on a single static snapshot of a protein to evaluate the binding interactions. A complementary approach to these computationally-cheap approximations is that of molecular dynamics, in which detailed simulations of the motions of the atoms comprising the molecules are used to study the behaviour of the molecules.

In molecular dynamics, a physics-based force field is used to define the interactions between the atoms in the simulation. By numerically integrating the resulting Newtonian equations of motion, a trajectory describing the time evolution of the system can be constructed. While molecular dynamics is more computationally expensive than docking, the resulting trajectories can be used to study the physical processes underlying binding pathways (Salmaso and Moro, 2018), and explore the dynamics of processes such as ligand unbinding (Huang and Caflisch, 2011b,a; Zhu et al., 2017).

Molecular dynamics is a well-established method in computational biology, with the simulation of protein dynamics beginning in the 1970s (McCammon et al., 1977) and the 2013 Nobel Prize in Chemistry awarded to Martin Karplus, Michael Levitt, and Arieh Warshel for laying the groundwork for the development of molecular dynamics (Smith and Roux, 2013). In recent years the increasing availability of high-quality protein structures has led to together with advances in computer hardware has led to a rapid growth in the application of

molecular dynamics to problems in structural and molecular biology (Karplus and McCammon, 2002; Hollingsworth and Dror, 2018). This increase in the accessibility of molecular dynamics is reflected in its growing application as part of the drug discovery process (Durrant and McCammon, 2011c; De Vivo et al., 2016).

Much as ligand-based and structure-based virtual screening methods are used in tandem, docking and molecular dynamics are regularly applied together as complementary techniques. One example of this is the relaxed complex scheme, in which the ligand is docked into different conformations of the receptor obtained by simulating the apo form using molecular dynamics (Lin et al., 2003; Amaro et al., 2008). This approach enables an ensemble of conformations of the protein-ligand complex to be generated, accounting for both ligand and receptor flexibility without the need for full molecular dynamics simulations of the complex. Molecular dynamics has also been applied as a pre-processing and post-processing step to protein-ligand docking for structure preparation, pose optimisation, affinity prediction, and ranking of docked ligands (Alonso et al., 2006).

1.6.2 | Free Energy Perturbation

The speed and computational efficiency of protein-ligand docking and scoring functions are made possible through approximations of the underlying physics, such as the use of simple models to describe the effects of solvent exposure and conformational entropy. While the computational efficiency of these methods makes it possible to rapidly screen large compound libraries in the early stages

of the drug discovery process, these approximations result in coarse-grained models that yield poor predictions of the free energy of binding (Schneider, 2010).

In contrast, free energy perturbation methods aim to directly calculate either the relative change in free energy between two states, or the absolute free energy of binding, by making a series of perturbations to the system and computing the change in energy at each step (Zwanzig, 1954). While these approaches are computationally expensive and require significant sampling of the system using either molecular dynamics (Jorgensen and Thomas, 2008) or Monte-Carlo (Jorgensen, 2009) simulations, the physical detail and potential for accurate binding energy predictions offered by free energy perturbation methods has led to growing interest in their application to drug discovery (Chodera et al., 2011; Wang et al., 2015; Lenselink et al., 2016; Williams-Noonan et al., 2017).

Finally, endpoint methods such as molecular mechanics Poisson–Boltzmann surface area (MM/PBSA) and molecular mechanics generalized Born surface area (MM/GBSA) offer a compromise between the speed of docking and the accuracy of FEP (Genheden and Ryde, 2015). These approaches are known as endpoint methods because they rely on simulating only the bound state and unbound state of the complex. For a recent review of the application of MM/PBSA methods to biomedical problems, including protein-ligand binding, see Wang et al. (2018)

1.7 | Thesis Structure and Contributions

In this Chapter we have given a brief overview of the drug discovery process and computer-aided drug design (CADD). We have discussed ligand-based virtual screening (LBVS) and structure-based virtual screening (SBVS), introducing protein-ligand docking and highlighted its success in predicting ligand binding poses and its weaknesses in predicting ligand binding affinity. We provided a summary of the classes of scoring functions used in protein-ligand docking to predict binding affinity, and discussed examples of how the application of machine learning techniques to the ever-growing body of publicly-available structural and affinity data has led to a new family of scoring functions displaying an enhanced ability to predict protein-ligand binding affinity.

In the thesis, we focus on new applications of supervised machine learning to further improve the prediction of protein-ligand binding affinity. In Chapter 2, we explore a range of structure-based and ligand-based features and how they may be used with different supervised machine learning algorithms for binding affinity prediction. We examine the dimensionality of the sets of features, and investigate how simple, parsimonious, and effective models can be constructed by using only a small set of informative features. We focus on the case where the correct binding mode is known, making use of x-ray crystal structures of the protein-ligand complex.

In Chapter 3, we investigate how the inclusion of a rich set of rapidly-computed ligand-based features improves the performance of RF-based scoring functions, and how scoring function performance is affected by the size and composition of the training set. We also examine how the size of the test set affects confidence

in the results of scoring function assessment. As in Chapter 2, we focus on using crystal structures of the protein-ligand complex. We find that the inclusion of ligand-based features consistently improves scoring function performance. We investigate the predictive power of models based on ligand-based features alone, and find that they are predictive of the mean observed affinity of a ligand for its binding partners. We conclude this chapter by testing whether scoring functions with and without ligand-based features are able to generalise to previously-unseen proteins and ligands, showing that despite strong predictive power on a diverse benchmark set, machine learning scoring functions generalise poorly to novel proteins and ligands.

In Chapter 4 we study the effect of structural features derived from docked poses instead of experimentally-determined binding poses for affinity prediction. We re-dock the PDBbind database and assess the quality of the generated poses; we find that in many cases, pose prediction errors are common. We re-train and test RF scoring functions on docked poses and find that the accuracy of binding affinity predictions is worse than when trained on x-ray crystal structures, even when testing on docked poses close to the experimentally-determined binding pose. We show that when ligand-based features (that are independent of the binding pose) are included in the scoring function, the accuracy of affinity predictions is substantially improved compared to when using just structural features derived from docked poses. Finally, we construct a new data set of ligands for six protein targets from the DUD-E database with new affinity data taken from the ChEMBL database, and generate docked poses for these ligands. We test RF scoring functions on this new data set, and find that the scoring functions are unable to predict the binding affinity of the ligands for

each target, unless they are trained on ligands for that target. We conclude that while the use of ligand-based features can help to mitigate the deleterious effects of docking pose prediction errors, machine learning scoring functions generalise poorly to novel targets and data sets.

Finally, in Chapter 5, we summarise the results of this work. We discuss the conclusions we have drawn and describe future work that might follow from this thesis.

2

Exploration of Features and Algorithms for Binding Affinity Prediction

This Chapter contains the results of an exploratory analysis of ways to improve protein-ligand binding affinity predictions. We first investigate the use of Random Forest (RF) to improve the AutoDock Vina scoring function. We compute a diverse set of ligand-based features and show that the addition of these to the Vina scoring function improves binding affinity prediction on our test set. We also investigate the features used by several published machine learning scoring functions for protein-ligand binding affinity and show that RF models using these features are also improved by the inclusion of a diverse set of ligand-based features. We analyse the correlations present in the features used across our training set and find that there are strong correlations between features in each feature set. We show for each set of features that the dimensionality of the corresponding scoring function can be substantially reduced by removing less-useful features without significantly affecting performance. Finally, we investigate several other machine learning algorithms and compare their performance to that

of RF. We find that for each set of features, XGBoost achieves comparable performance to RF, while linear models, neural networks, and AdaBoost perform worse than XGBoost or RF.

2.1 | Introduction

The last decade has seen widespread adoption of the use of machine learning techniques to develop scoring functions for protein-ligand binding affinity. A wide array of methods have been employed, from using substituting machine learning models for linear models in classical scoring functions Li et al. (2014a) or using machine learning methods to correct classical scoring functions (Wang and Zhang, 2017), to designing novel feature sets specifically for use in machine learning models (Ballester and Mitchell, 2010; Durrant and McCammon, 2010; Wójcikowski et al., 2018), and even using deep learning to learn a latent representation of a protein-ligand complex directly from a structural model (Ragoza et al., 2017a; Jiménez et al., 2018; Stepniewska-Dziubinska et al., 2018). It has also seen the development and adoption by the community of standard benchmarks for scoring function performance (Cheng et al., 2009; Li et al., 2014c; Su et al., 2018; Liu et al., 2017) as well as numerous community exercies (Dunbar et al., 2011, 2013; Smith et al., 2016; Carlson et al., 2016) and grand challenges (Gathiaka et al., 2016; Gaieb et al., 2018, 2019) to assess the state of the art and guide the growth of the field.

With the abundance of tools and techniques available to aid in the task of scoring function development, it is important to explore these resources to understand how they might be used, and where improvements might be made.

In this Chapter we investigated the features used by several published machine learning scoring functions, namely: RF-Score (Ballester and Mitchell, 2010), RF-Score v3 (Li et al., 2015b), and NNScore 2.0 (Durrant and McCammon, 2011b), as well as the classical scoring function used by the protein-ligand docking tool, AutoDock Vina (Trott and Olson, 2010). We compute each of these sets of features for the protein-ligand complexes comprising the PDBbind 2016 refined set, and explore the structure of the resulting feature sets. As these features are predominantly structure-based, and as ligand-based methods have been employed successfully in virtual screening, we hypothesised that including a more detailed representation of the physicochemical properties of the ligand in a scoring function may be valuable for predicting protein-ligand binding affinity.

To test this hypothesis, we computed a detailed set of molecular descriptors – values describing physicochemical properties of the molecule – for the ligands in the PDBbind 2016 refined set using the open-source cheminformatics toolkit RDKit (Landrum, n.d.a). We investigated the performance of Random Forest (RF) regression models using each set of structure based features with and without the addition of these ligand-based features, validating our models using the PDBbind 2007 core set. We will show that, for each set of structure-based features, augmenting the structure-based features with additional ligand-based features results in a RF scoring function that outperforms a RF scoring function using the structure-based features alone.

Next, for each feature set, we investigated how the number of features used affects the performance of RF models. Using the feature importance computed by the RF, we identify the most useful features and re-train RF models using only those features. We find that the size of each feature set can be substantially

reduced without reducing performance on our test set. We also investigated the predictions of a RF using only ligand-based features, and observed a strong correlation between its predictions and the experimentally-determined binding affinity for the complexes in our test set.

Finally, we investigated the use of other different machine learning algorithms: regularised linear regression, artificial neural networks, AdaBoost, XGBoost. For each of our feature sets, and our ligand-based features, we use cross-validation to compare the performance of these machine learning algorithms to that of RF. We show that RF consistently outperforms linear, neural network, and AdaBoost models for each feature set, and achieves performance comparable to that of XGBoost.

2.2 | Materials and Methods

2.2.1 | Data

In this work we focused on the task of 'scoring': the prediction of protein-ligand binding affinity given the binding mode of the ligand. To accomplish this, we restricted our data to protein-ligand complexes for which a crystal structure of the bound complex and an experimentally-determined value of the binding affinity was available. For this, we use the PDBbind database (Liu et al., 2017): a curated set of bound macromolecule structures drawn from the Protein Data Bank (PDB) (Berman et al., 2000), each with an experimentally-measured binding affinity for its binding partner. Each release of PDBbind includes a 'general set', which contains all the protein-ligand structures in the database; and a 're-

fined set', a subset of protein-ligand complexes satisfying strict criteria concerning structure quality, affinity data reliability, and the nature of the complex. In this Chapter we use two versions of the PDBbind database: the 2016 release and the 2018 release. The 2016 release of PDBbind contains 13,308 protein-ligand complexes in the general set, with 4,057 complexes in the refined set, while the 2018 release of PDBbind contains 16,151 protein-ligand complexes in the general set, with 4,463 complexes in the refined set.

PDBbind provides for each complex an experimentally-determined value of the inhibition constant K_i, the dissociation constant K_d, or the half-maximal inhibitory concentration IC_{50}, in decreasing order of preference (*e.g.* if both K_i and K_d values are available, PDBbind reports the measurement of K_i). The refined set includes only measurements of K_i and K_d, while the general set also includes data for which only IC_{50} measurements were available. For our purposes, these values are used interchangeably and are collectively denoted by the binding constant, K. We used the negative base-10 logarithm of K, commonly denoted as pK:

$$pK = -\log_{10} K \tag{2.1}$$

Test Set

To validate our models, we used a subset of the PDBbind refined set referred to as the 'core set'. This is obtained by clustering the proteins in the refined set at 90% sequence identity and selecting representatives of each cluster for which the corresponding ligands have a broad range of binding affinity values, resulting in a diverse, non-redundant set of protein-ligand complexes. In this Chapter we

use the PDBbind 2007 core set, which contains 210 protein-ligand complexes, as our test set. This version of the core set was used as the 'scoring power' benchmark in the first iteration of the Comparative Assessment of Scoring Functions (CASF) exercise (Cheng et al., 2009), allowing us to directly compare the performance of our models to that of the classical scoring functions tested in CASF. Structures for which features could not be computed due to errors parsing the structural data were excluded, resulting in a test set of 196 protein-ligand complexes. The full list of PDB codes for the test set is included in Appendix Table A.1.

Training Set

Models were trained using data from the PDBbind refined set. We used the PDBbind 2016 refined set for all experiments except the comparison of different machine learning algorithms. This experiment was conducted later, enabling us to use the PDBbind 2018 refined set instead. In all cases, the training set consisted of all structures in the refined set excluding the structures in the test set. As with the test set, structures for which any features could not be computed due to errors parsing the structural data were excluded from the training set.

2.2.2 | Features

We used two distinct types of features to describe a protein-ligand complex: ligand-based features, derived from the 2D molecular graph of the ligand, and structure-based features, derived from a 3D structural model of the complex. As

part of the data preparation process, any feature with zero variance across the training set was not included in the models.

Ligand-Based Features

To represent the ligand in our models we used a diverse set of molecular descriptors computed using the cheminformatics toolkit RDKit. Using the *Descriptors* module of the Python RDKit package, we computed a set of 200 molecular descriptors for each ligand. These descriptors are conformation-independent and may be categorized as either (computed) experimental properties (*e.g.* molar refractivity, logP) or theoretical descriptors derived from a symbolic representation of the molecule. The theoretical descriptors may be further categorized according to the dimensionality of the representation of the molecule from which they are derived. The conformer-independent descriptors we consider are either 1-D compositional properties (*e.g.* heavy atom counts, bonds counts, and molecular weight) or 2-D topological properties (*e.g.* fragment counts, topological polar surface area, and connectivity index). Any features with zero variance across the data set, or that were null-valued (*i.e.* infinite or not computable) within the data set were excluded. We removed the Ipc index (an information theory-derived descriptor) as it produced extreme numerical values for larger molecules (too large to be represented as 32-bit floats). In total, 185 features were retained, and the full list of features is included in the Appendix. We refer to this set of features as 'RDKit features'.

Structure-Based Features

To investigate the effects of augmentation with ligand molecular descriptors, we considered the features of several publicly-available scoring functions, namely the AutoDock Vina scoring function (Trott and Olson, 2010), RF-Score (Ballester and Mitchell, 2010), RF-Score v3 (Li et al., 2015b), and NNScore 2.0 (Durrant and McCammon, 2011b). We computed the features of each of these scoring functions using the implementations provided by the Open Drug Discovery Toolkit (ODDT) version 0.6 (Wójcikowski et al., 2015).

The AutoDock Vina Scoring Function

We first combined the RDKit features with the features used by a classical scoring function. For this, we chose the scoring function used by the protein-ligand docking software AutoDock Vina (Trott and Olson, 2010). We chose the Vina scoring function over other classical scoring functions for two reasons. First, AutoDock Vina is open-source and widely used in the literature, with the derivative work Smina (Koes et al., 2013) providing a user-friendly interface. Second, the terms used by the AutoDock Vina scoring function have been used in the development of several machine-learning scoring functions (Durrant and McCammon, 2011b; Li et al., 2015b; Wang and Zhang, 2017), demonstrating that they are a useful set of features for binding affinity prediction.

The Vina scoring function uses a weighted sum of five inter-molecular energy terms, representing interactions between the protein and the ligand, with a non-linear penalty proportional to the flexibility of the ligand, representing the effect of conformational entropy on the free energy of binding. The contribu-

tions from the five inter-molecular energy terms takes the form:

$$c_{inter} = w_{gauss1}\Delta E_{gauss1} + w_{gauss2}\Delta E_{gauss2} + w_{repulsion}\Delta E_{repulsion} + w_{hydrophobic}\Delta E_{hydrophobic} + w_{hydrogen}\Delta E_{hydrogen} \quad (2.2)$$

where ΔE_{gauss1}, ΔE_{gauss2}, $\Delta E_{repulsion}$, $\Delta E_{hydrophobic}$, and $\Delta E_{hydrogen}$ are potentials obtained by summing the pairwise contributions of all protein-ligand atom pairs, defined in the original publication (Trott and Olson, 2010). The AutoDock Vina score obtained from this inter-molecular contribution is then given by:

$$S = \frac{c_{inter}}{1 + w_{rot}N_{rot}} \quad (2.3)$$

where N_{rot} is the number of rotatable bonds in the ligand. The weights w_i are shown in Table 2.1. Note that as $w_{rot} << 1$ the denominator in Equation 2.3 will be close to 1 for a small ligand that does not have many rotatable bonds, so the conformational entropy penalty will be low.

w_{gauss1}	-0.0356
w_{gauss2}	-0.00516
$w_{repulsion}$	0.840
$w_{hydrophobic}$	-0.0351
$w_{hydrogen}$	-0.587
w_{rot}	0.0585

Table 2.1: The weights used in the AutoDock Vina scoring function.

We refer to the set of five energy terms plus the number of rotatable bonds of the ligand as 'Vina features'.

RF-Score, RF-Score v3, and NNScore 2.0

In addition to combining the RDKit descriptors with the Vina scoring function, we also investigated whether more complex machine-learning scoring functions could be improved by including a more detailed description of the ligand. For this, we chose to use the features of three machine-learning scoring functions.

The RF-based scoring function RF-Score (Ballester and Mitchell, 2010) uses features that count the number of times atoms of each element belonging to the protein are found within 12 Å of atoms of each element belonging to the ligand; for example, the number of carbon atoms in the protein found within 12 Å of nitrogen atoms in the ligand. Four elements (C, N, O, S) are considered for the protein and nine elements (C, N, O, F, P, S, Cl, Br, I) for the ligand, giving a total of 36 pairwise interaction features. We refer to these as 'RF-Score features'. All RF-Score features had non-zero variance across the training set.

An updated version of RF-Score, RF-Sore v3 (Li et al., 2015b), combines the 36 features of RF-Score with the 6 features of the AutoDock Vina scoring function. This version displayed improved performance over either RF-Score or a RF using the Vina features, so we also considered this combination of features as an additional model. We refer to this combination of 42 features as 'RF-Score v3 features'.

The neural network-based scoring function NNScore 2.0 (Durrant and McCammon, 2011b) combines the force field-like terms used by the scoring function of AutoDock Vina with a large number of structure-based features describing interactions between the protein and ligand. It also uses a small number of features of the ligand. These interaction features are defined by the BINANA

(BINding ANAlyzer) algorithm (Durrant and McCammon, 2011a), and include counts of protein atoms found close to ligand atoms, electrostatic interactions, hydrophobic contacts, hydrogen bonds, salt bridges, π-interactions, as well as counts of each AutoDock atom type found in the ligand, and the number of rotatable bonds in the ligand. After excluding features with zero variance across the training set, a total of a total of 174 NNScore 2.0 features remained. The full list of features used is included in the Appendix. We refer to this set of features, including the AutoDock Vina terms, as 'NNScore2 features'.

2.2.3 | Machine Learning Algorithms

We investigated several machine learning algorithms for fitting regression models, which we outline here. With the exception of XGBoost, which has a dedicated Python API (Chen and Guestrin, 2016), we used the scikit-learn (Pedregosa et al., 2011) implementation of each algorithm

Elastic Net Linear Regression

As classical scoring functions such as the AutoDock Vina scoring function are typically linear combinations of features, it makes sense to include a linear model when comparing different machine learning algorithms. However, as we are introducing many additional features to the model that may contain correlations, ordinary-least-squares (OLS) regression may result in an unstable model with high variance. The solution to this problem is to reduce the variance of the model by introducing an acceptable level of bias. This process of lowering variance at the cost of introducing bias is known as regularization.

Two popular approaches to regularization are ridge regression and lasso regression. In ridge regression, the sum of squares of the coefficients of the linear model is penalised. This punishes large coefficients for features that do not significantly improve the fit, pushing the weights associated with less-useful features or those correlated with other features toward zero. Lasso regression is conceptually similar to ridge regression, except the sum of the absolute values of the coefficients is penalised instead. This pushes the coefficients of less-useful features toward zero and, unlike ridge regression, rewards the setting of coefficients to identically zero. Both approaches reduce the variance of the model, allowing it to better generalise to new data, at the cost of introducing some bias due to down-weighting features.

For our linear models, we used a form of regression called elastic net, which combines the penalties of ridge and lasso regression, with a weight assigned to each penalty. This allows an optimal combination of the two forms of penalty to be determined without imposing a hypothesis about whether ridge or lasso regression would be more appropriate. In the limiting cases where the optimal strategy would be to simply use one of the two, elastic net will assign a high weight to the optimal penalty and a low weight to the other penalty.

Random Forest

Random Forest (RF) is an ensemble learning method which functions by training a collection of decision trees. Predictions are made by collecting the predictions of each tree and taking either the mode (for classification) or mean (for regression) of the predictions. The general method of building ensembles of decision

tree learners was proposed by Ho (Ho, 1995, 1998), while the modern formulation of the RF was introduced by Breiman (Breiman, 2001) in 2001. The RF algorithm builds and trains a diverse set of decision trees as follows.

First, a number of subsets of the training data are randomly sampled *with replacement* in a process known as 'bootstrap aggregation', or 'bagging'. This ensures that the training data seen by each tree are slightly different but consistent. An estimate of the error of the RF when applied to unseen data, known as the out-of-bag (OOB) estimate can be obtained by computing the error in predictions for each training sample x, for those trees whose training data did not include x. This acts as a form of internal cross validation of the RF. In addition, the importance of each feature is estimated by permuting the value of that feature in the training data and comparing the OOB estimate for a RF trained on this permuted data to that obtained when training on the original data.

Second, at each node of a tree, the criterion for splitting is determined using a random subset of the features in the data. As the number of features used at each split increases, the predictive performance of each individual tree increases, but so does the correlation of the trees. This is the parameter to which the performance of the RF is most sensitive, and so optimal performance is usually obtained by identifying the optimal range for this parameter.

A major advantage of RF over other classification or regression methods is a resilience (though not an immunity) to over-fitting. This is generally explained by the sub-sampling of features at each node, also known as the random subspace method. This introduces a form of stochastic determination, which is considered to be an effective approach to reducing over-fitting (Kleinberg, 1996, 2000).

Boosting

Boosting is a family of ensemble learning algorithms that use a collection of weak predictors, typically shallow decision trees ('stumps'), to build a strong predictor. The learners in the ensemble are trained sequentially, with each learner attempting to correct the training errors of the previous.

Bagging and boosting can be understood with respect to the bias-variance tradeoff as follows. In a bagging algorithm, such as RF, each member of the ensemble is trained independently on a bootstrap sample of the training data. For a decision tree, which maximally overfits its training data, this results in a predictor with high variance (large errors on unseen data) and low bias (minimal training error). By training many decision trees on independent bootstrap samples of the data, the RF contains many different high-variance low-bias predictors. Bagging relies on the random errors of each independent predictor averaging out to zero when aggregated over a sufficiently large ensemble of predictors, resulting in a predictor with bias at the cost of some variance introduced by aggregation. Each learner captures unique insights about part of the data, and by pooling their knowledge, the ensemble of learners is able to generalise to previously-unseen data.

Boosting, in contrast with bagging, takes the approach of combining the predictions of an ensemble of high-variance low-bias predictors. In a boosting algorithm, each predictor in the ensemble is a weak learner (error only slightly better than random), such as a short decision tree with only a single split. Because of its low complexity, the learner is very stable (low variance) and can only learn a small part of the relationship in the training data (high bias). This high bias

means the learner will have large errors on some of the training data. The key insight behind boosting is that a weak learner can be converted into an arbitrarily strong (low error) learner (Schapire, 1990). By training an ensemble of such weak learners sequentially, with each learner attempting to correct the training errors of the previous, the resulting ensemble is predictor with low variance at the cost of some bias introduced by aggregation.

Adaptive Boosting

Adaptive Boosting, commonly known as 'AdaBoost' is a boosting algorithm in which each learner attempts to correct the errors of the previous learner by focusing on the samples that were poorly fitted. When fitting an AdaBoost predictor, the first learner is trained with equal weight assigned to all training samples. The training errors on the set of training examples are computed, and passed to the next learner as a set of weights on the training examples, with larger errors corresponding to greater weight. In this way, each subsequent learner focuses more on accurately predicting the examples poorly fitted by previous learners.

Gradient Boosting and XGBoost

Gradient boosting, just like adaptive boosting, builds a strong learner by training a sequence of weak learners, with the training errors of each weak learner informing the training of the next learner. The difference between the two approaches lies in how the training errors are propagated. Gradient boosting, as suggested by the title, is based on the concept of gradient descent. Rather than passing the training errors of a learner as weights to the next learner, each learner

is instead fitted to the residual of the predictions of the previous learner. By sequentially fitting to the residual and adding the resulting learner to the ensemble, each learner corrects the residual of the previous, resulting in an ensemble predictor with low error. More formally, for predictor variable x and target variable y, let $F_m(x)$ denote the ensemble predictor function at iteration m, and $h(x)$ the estimator added to the ensemble at this iteration. If $h(x)$ perfectly corrected the error on the predictions of F_m, the resulting ensemble predictor would be given by

$$F_{m+1}(x) = F_m(x) + h(x) = y. \tag{2.4}$$

Rearranging gives $h(x) = y - F_m(x)$, and so the target of the new estimator h is precisely the residual of the ensemble at iteration m.

XGBoost, which stands for 'eXtreme Gradient Boosting' is an open-source framework implementing a range of optimisations intended to improve the accuracy and computational speed of gradient boosting models (Chen and Guestrin, 2016). Since its introduction in 2016, XGBoost has grown rapidly as a popular choice for gradient boosting, with XGBoost solutions regularly winning machine learning competitions[1] such as those hosted by Kaggle[2]. Because of this success and the availability of a Python API, we used XGBoost to investigate gradient boosting models.

[1]https://github.com/dmlc/xgboost/tree/master/demo#machine-learning-challenge-winning-solutions, last accessed 22/10/2019

[2]https://www.kaggle.com/, last accessed 22/10/2019

Neural Networks

Neural networks are a family of models inspired by the networks of neurons found in the brain. A wide range of neural network architectures have been applied successfully to problems in cheminformatics and drug discovery, and these were discussed in Chapter 1. In this Chapter we focused on one class of neural network, the multilayer perceptron. This is a form of feed-forward neural network in which each layer consists of a number of nodes, with the output of each layer of nodes forming the input to the nodes of the next layer. By applying a non-linear transformation to a weighted sum of the inputs of each node (the 'activation function'), the network is able to learn non-linear relationships from the data. Historically, the activation function commonly took the form of a sigmoidal function such as the logistic function or hyperbolic tangent; more recently, advances in deep learning have led to the rectified linear unit (ReLU) gaining popularity as an activation function.

We used a simple neural network architecture consisting of a single hidden layer and one continuous-valued output node. The size of the hidden layer and the activation function were among the hyperparameters tuned using the training data.

2.2.4 | Scoring Function Assessment

Prediction of protein-ligand binding affinity using machine learning is a regression problem, where the objective is to achieve a linear correlation between the predicted and experimentally-determined binding affinity values. To assess the predictions of a scoring function, we used the Pearson correlation coefficient

between the predicted and experimentally-determined pK values. The Pearson correlation coefficient is a measure of the linear correlation between two variables X and Y and is calculated as follows: let x_i and y_i denote the i^{th} sample of X and Y respectively, then the Pearson correlation coefficient, ρ_{p}, between X and Y is:

$$\rho_{\text{p}} = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}} \tag{2.5}$$

where $\bar{x}$ and $\bar{y}$ denote the mean value of X and Y respectively. The square of the Pearson correlation coefficient, ${\rho_{\text{p}}}^2$, is a special case of the coefficient of determination R^2: it measures the proportion of the variance in Y that can be explained by X using a linear regression.

2.2.5 | Parameter Tuning for Machine Learning

For all algorithms except RF, parameters were tuned using random search on a five-fold cross-validation performed using our training set: the PDBbind 2018 refined set excluding structures found in the PDBbind 2007 core set. We used random search rather than an exhaustive grid search over all parameters as this approach has been shown to explore parameter space more efficiently, resulting in reduced search time and increased model performance (Bergstra and Bengio, 2012). For the cross-validation, the pK values of the training set were binned in 1 pK intervals, and the folds generated using stratified sampling across these bins, resulting in similar distributions of pK values across each of the five folds. Performance of the models was assessed by computing the Pearson correlation

coefficient between predicted and experimental pK values for each fold, and taking the mean of the Pearson correlation coefficients for the five folds.

RF is generally robust with respect to hyperparameter choice. However, we investigated two parameters to see whether parameter tuning had a significant impact on RF performance on the PDBbind data. The parameters tuned were the number of trees in the forest and the number of features randomly sampled when making a split in a tree. To determine the optimum value for these two parameters for each feature set, we trained RFs for different values of the parameters and computed the Pearson correlation coefficient on the out-of-bag predictions. These are predictions obtained for the training data by predicting each training sample using only the trees in the RF that were not trained on that sample. This can be viewed as a form of internal cross-validation that does not require splitting the data into folds.

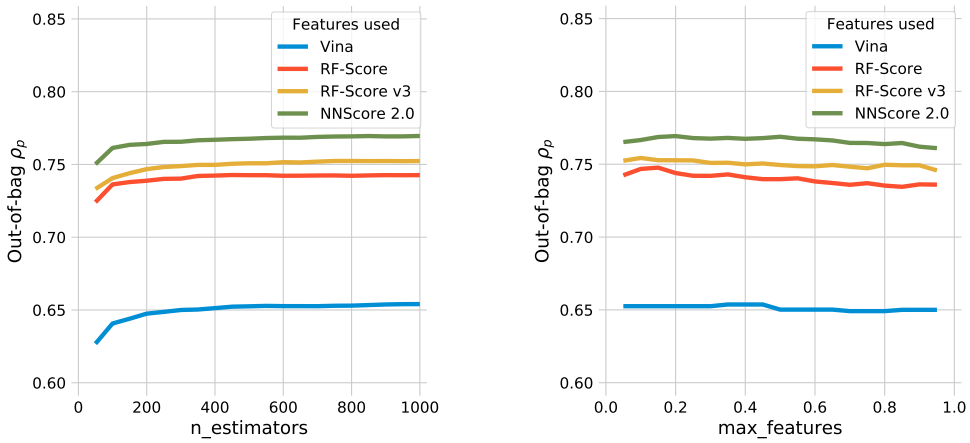
2.3 | Results and Discussion

2.3.1 | Exploratory Data Analysis

We first perform some exploratory analysis on the featurised data. For each set of features, a correlation matrix was constructed by computing the Pearson correlation coefficient between all pairs of features across the training set.

Principal Component Analysis

To quantify the dimensionality of each feature set, we use principal component analysis (PCA) to determine the number of dimensions necessary to capture the

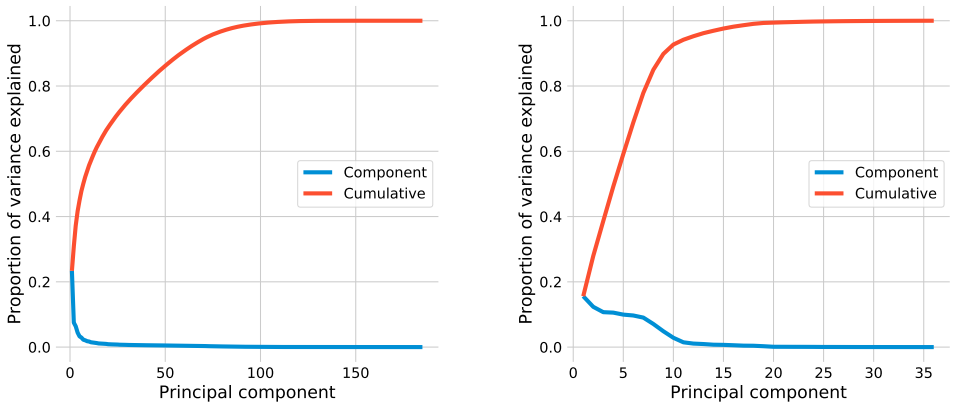


(a) Tuning number of trees used (b) Tuning number of features sampled

Figure 2.1: Out-of-bag performance when tuning RF parameters on the PDBbind 2016 refined set. (a) Performance increases with the number of trees used, reaching a plateau at 500 trees. (b) Performance fluctuates with the number of features sampled at each split; performance drops slightly for all feature sets for values of max_features greater than 0.4.

variance in the data. PCA performs a linear transformation of the data such that the first basis vector of the new coordinate system, the 'first principal component', captures the maximum possible amount of variance in the data. The second principal component captures the maximum amount of remaining unexplained variance, and so on. By determining the number of principal components necessary to capture a desired percentage of the variance in the data, it is possible to gain some insight into the number of dimensions necessary to capture the information content of the data.

For the RDKit features, RF-Score features, RF-Score v3 features, and NNScore 2.0 features, PCA was performed on the training set. In each case, the features were standardised by scaling to zero mean and unit variance before applying PCA. The variance explained by the principal components for these four feature



(a) RDKit features

(b) RF-Score features

(c) RF-Score v3 features

(d) NNScore 2.0 features

Figure 2.2: PCA explained variance for RDKit, RF-Score, RF-Score v3, and NNScore 2.0 features. The variance explained by each component is shown by the blue line. The cumulative variance explained by the first n features is shown by the red line.

sets is shown in Figure 2.2. For each feature set, the number of components necessary to explain most of the variance in the data is much smaller than the total number of features, suggesting that the dimensionality of the feature sets could be substantially reduced.

Correlation of Features

We next investigated the correlations within each feature set for the RDKit features, RF-Score v3 features, and NNScore 2.0 features. The results of the PCA in Figure 2.2 suggest that the number of dimensions necessary to capture the information in the features is much lower than the number of features. We computed the correlation matrix for the RDKit features, RF-Score features, and NNScore 2.0 features. This comprises the Pearson correlation coefficient between each pair of features in the feature set. We did not compute a separate correlation matrix for the RF-Score features or the Vina features because the RF-Score v3 features comprises both the RF-Score features and the Vina features, and the Vina features are also included in the NNScore 2.0 features.

Figures 2.3 to 2.5 show the correlation matrix for each feature set. Features are grouped along the axes: RDKit features are grouped according to the type of descriptor; RF-Score features are grouped by the ligand atom element; NNScore 2.0 features are grouped according to the type of interaction captured. For RF-Score v3 and NNScore 2.0 features, a group containing the Vina features was also defined.

Figure 2.3 shows that, with the exception of descriptors counting the presence of drug fragments, many of the RDKit features are highly correlated. For

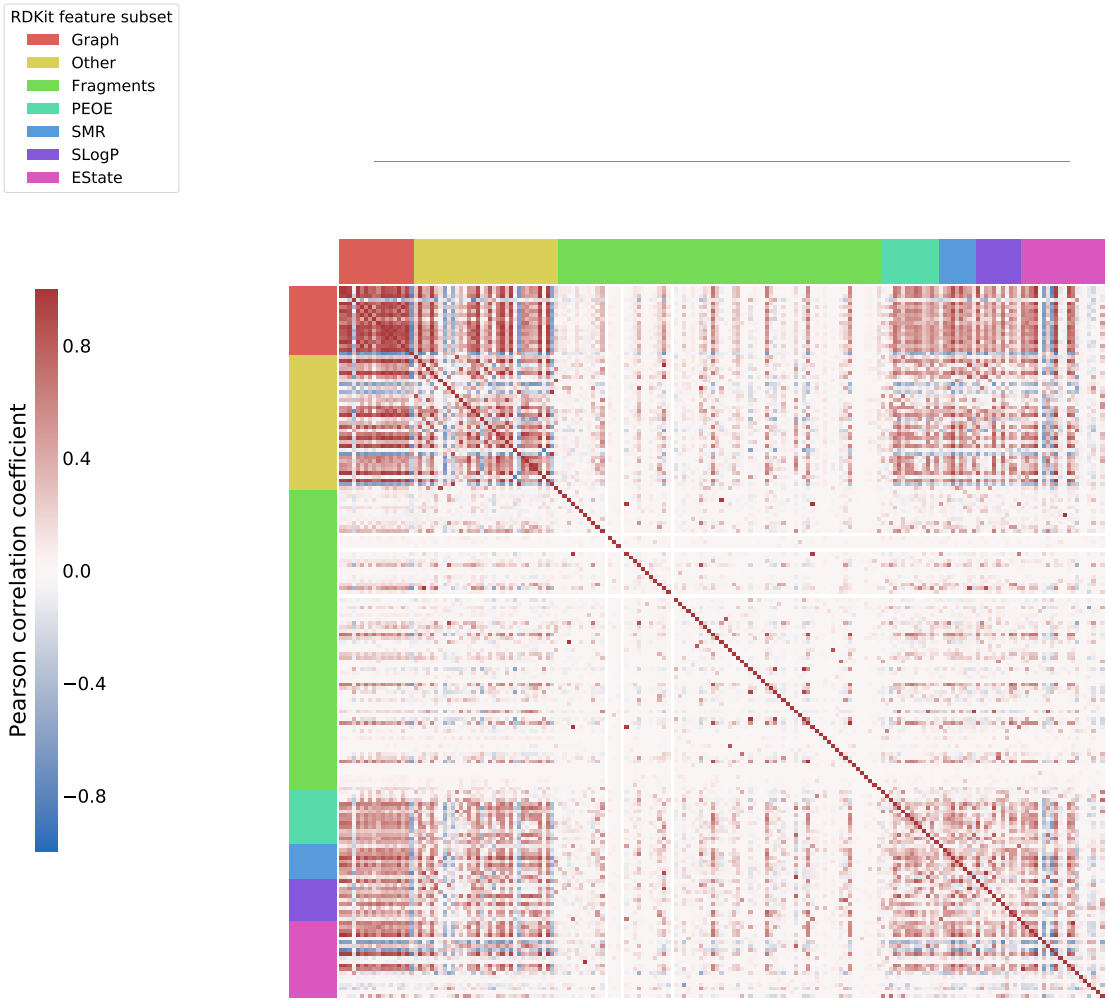


Figure 2.3: Correlation matrix for the RDKit features.

example, the SlogP descriptors (purple labels) are the computed Wildman-Crippen contributions to the molecule's logP (Wildman and Crippen, 1999), so we should expect some correlation between these features and the computed logP itself, which is included in the 'Other' features (yellow). Similarly, the graph-based descriptors ('Graph', red labels) capture the connectivity and complexity of the molecular graph (Hall and Kier, 1991), and so will be correlated with overall graph complexity.

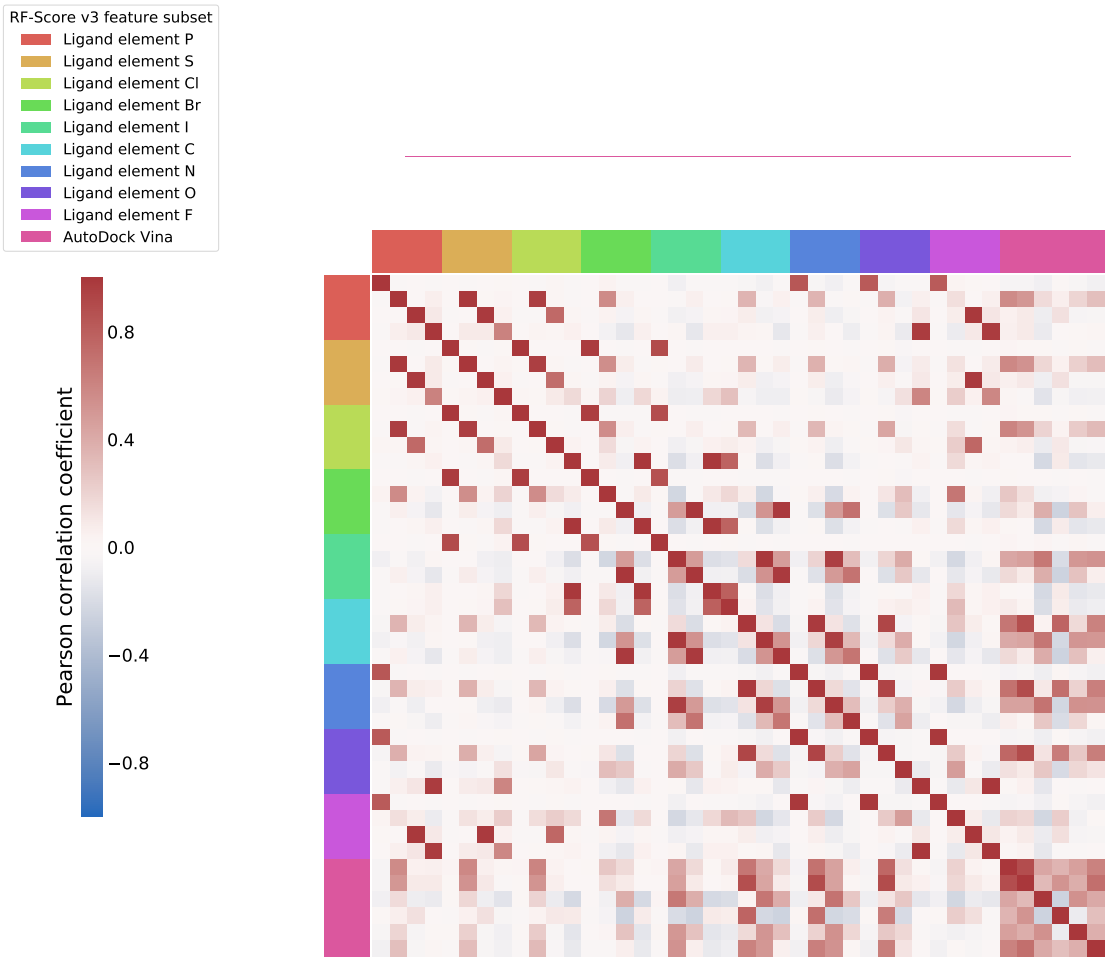


Figure 2.4: Correlation matrix for the RF-Score v3 features.

Figure 2.4 shows the correlations between RF-Score v3 features. The Vina features are indicated by magenta labels; RF-Score features are grouped according to the element of the ligand atom in the protein-ligand atom pair, each indicated by a different coloured label. The Vina features are more correlated with each other than they are with most of the RF-Score features, despite being computed using different potentials. In particular, the two Gaussian potentials are highly correlated, indicating that both capture very similar information. It is interesting to note that many of the RF-Score features appear to be very highly correlated (Pearson correlation coefficient greater than 0.9). We inspected the correlations between RF-Score features and found that for a given element in the ligand, the number of close contacts between that element and the four elements from the protein (C, N, O, and S) were all highly correlated. This suggests that the RF-Score descriptors are primarily a measure of the number of protein heavy atoms surrounding the ligand, and are too coarse-grained to capture detailed interactions with different elements in the protein.

Figure 2.5 shows the correlation between NNScore 2.0 features. The Vina features are indicated by red labels; other NNScore 2.0 features are grouped according to the type of interaction captured, each indicated by a different coloured label. The pairwise correlations between NNScore 2.0 features are overall lower than those observed for the RF-Score v3 features. This is not surprising, since the atom pair-style descriptors used by NNScore 2.0 are more fine-grained than those used by RF-Score v3, as they distinguish between different AutoDock atom types, interaction range, and electrostatic interactions, rather than simply counting all instances of a pair of atoms within 12Å.

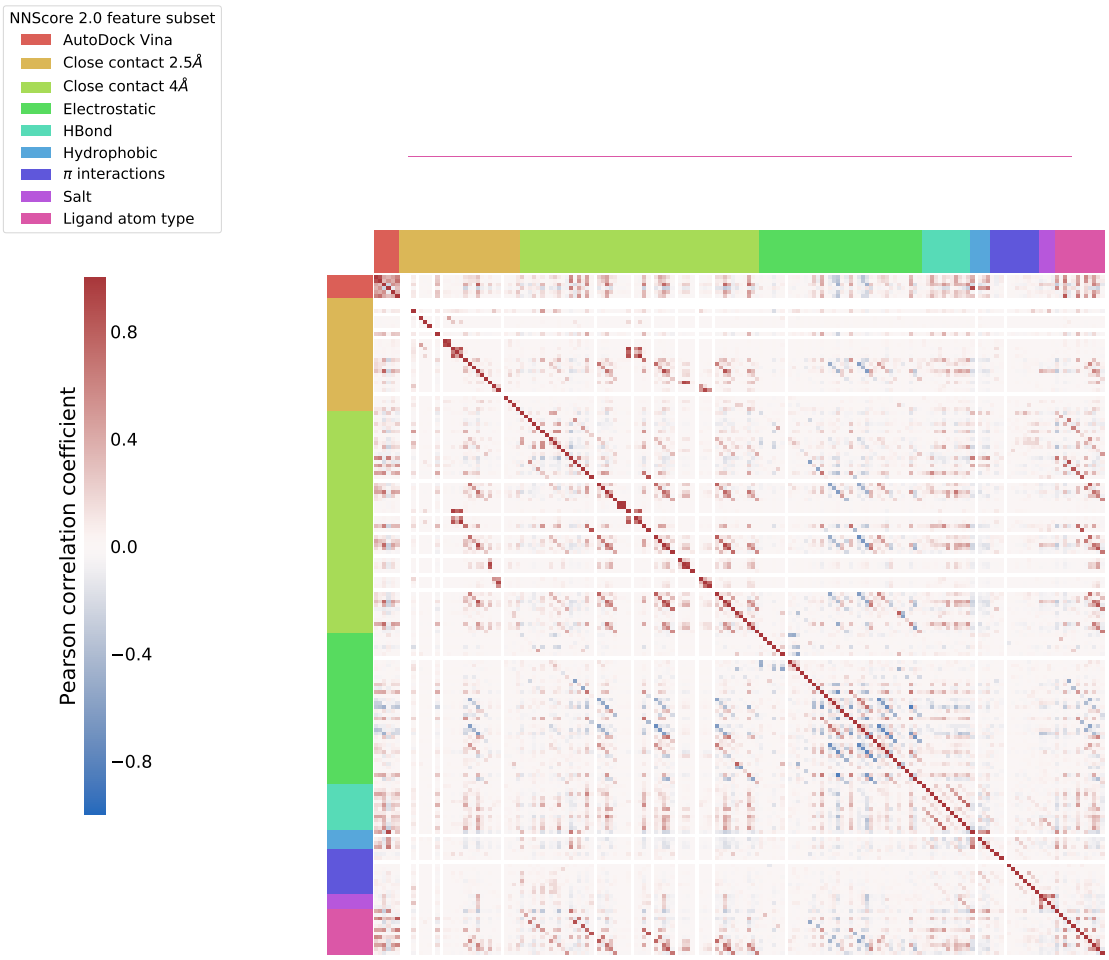


Figure 2.5: Correlation matrix for the NNScore 2.0 features.

2.3.2 | Benchmarking the AutoDock Vina Scoring Function

We first benchmarked the AutoDock Vina scoring function on the PDBbind 2007 core set. Figure 2.6 shows the predicted vs experimental pK values using the Vina scoring function for the structures in the PDBbind 2007 core set. The Vina scoring function predicts the change in free energy upon binding, so pK val-

ues were computed using Equation 1.10. The Pearson correlation coefficient between the predicted and measured pK values is 0.585, falling within the range of 0.545 to 0.644 for the scoring functions tested on the PDBbind 2007 core set in CASF2009 (Cheng et al., 2009).

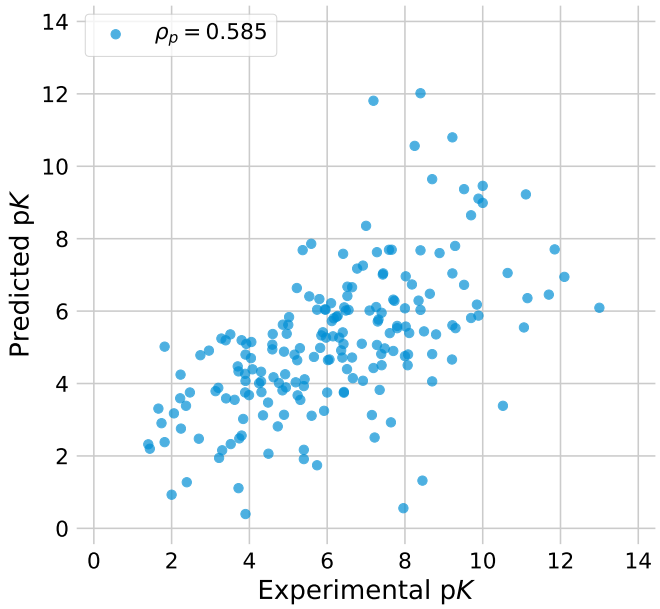


Figure 2.6: Predicted vs experimental pK values for the PDBbind 2007 core set using the AutoDock Vina scoring function.

One structure (PDB: 1TYR) was excluded as an outlier: the Vina scoring function assigned an unreasonably large, positive change in free energy (equivalent to a pK of -31.4). To determine the cause of this, we compared the computed interaction terms to the distribution of the values computed for the structures in the PDBbind 2016 refined set. We found that the 'repulsion' term is a clear outlier: this takes a value of 71.2 for the structure of 1TYR, compared to a mean of 6.2 with a standard deviation of 4.3 across the PDBbind 2016 refined set. This suggests some steric clash between the protein and the ligand in the 1TYR struc-

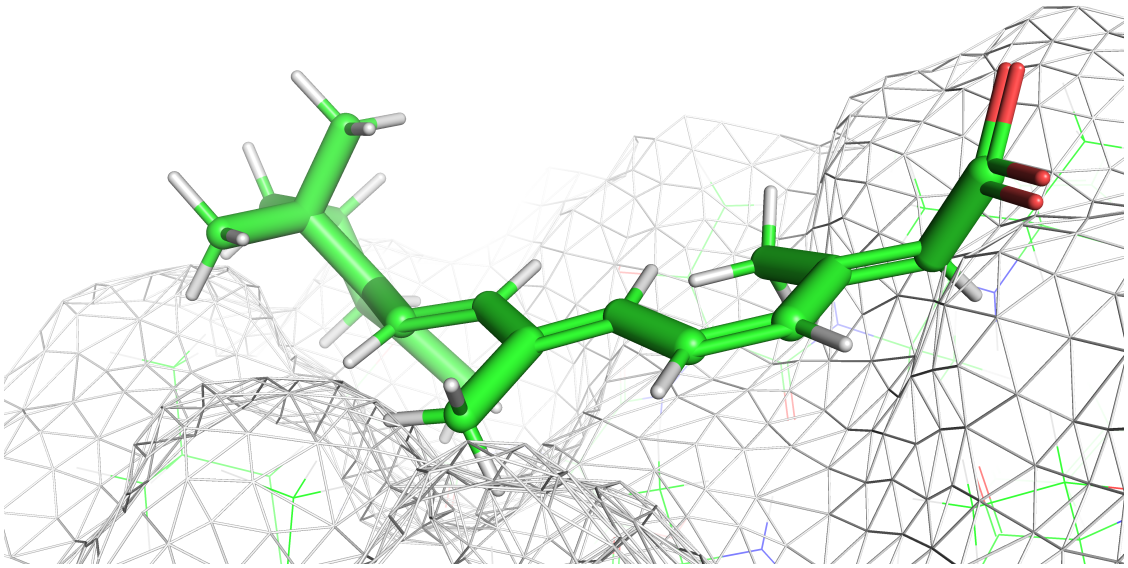


Figure 2.7: Binding surface in 1TYR. Close contacts and overlapping of the ligand (sticks) and surface of the protein (mesh) suggests steric clash resulting in a poor score using the AutoDock Vina scoring function

ture. The experimentally-determined binding pose of the ligand and the surface of the active site of the protein are shown in Figure 2.7. This image shows parts of the ligand in close proximity to the surface of the active site, which would explain the abnormally large repulsion term in the AutoDock Vina scoring function and the resulting large, positive predicted change in free energy upon binding.

2.3.3 | Augmenting Scoring Functions with Ligand Molecular Descriptors

We next investigated whether using RF in place of the AutoDock Vina scoring function improves binding affinity prediction on our test set. We also investi-

gated whether including additional ligand-based features in the scoring function improved binding affinity prediction. To do this, we trained two RFs on the PDBbind 2016 refined set: one using the Vina features, and one using the Vina features and the RDKit features. Figure 2.8A shows the predicted against measured pK values for the RF using the Vina features; Figure 2.8B shows the predicted against measured pK values for the RF using Vina features and RDKit features. The Pearson correlation coefficient for the RF using Vina features is 0.683; a considerable improvement over the 0.585 achieved by the Vina scoring function. The Pearson correlation coefficient for the RF using Vina features and RDKit features is 0.768, indicating that the inclusion of additional ligand-based features results in a substantial improvement in performance.

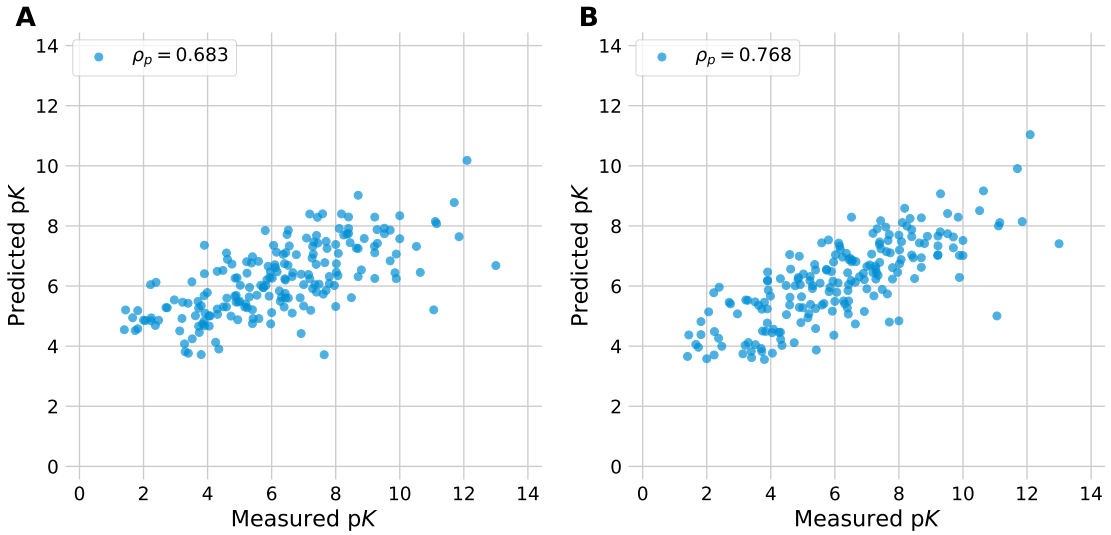


Figure 2.8: Predicted vs experimental pK values for the PDBbind 2007 core set. A: RF model using Vina features. B: RF model using Vina features and RDKit features.

Noting that the inclusion of additional ligand-based features improved the performance of a RF using the Vina features, we next investigated whether RFs

using larger, more complex feature sets of published machine learning scoring functions could also be improved by the inclusion of additional ligand-based features. To do this, we trained RFs using RF-Score, RF-Score v3, and NNScore 2.0 features and, for each set of features, we also trained a RF using that set of features together with the RDKit features. Each RF was then tested on our test set. Figure 2.9A shows the predicted against measured pK values for the RF using RF-Score features; Figure 2.9B shows the predicted against measured pK values for the RF using RF-Score features combined with the RDKit features. The RF using RF-Score and RDKit features out-performs the RF using RF-Score features alone (Pearson correlation coefficient 0.761 versus 0.706), suggesting that it is also beneficial to augment the RF-Score features with additional ligand-based features.

However, Ballester and Mitchell (2010) reported a Pearson correlation coefficient of 0.776 on the PDBbind 2007 core set using the published version of RF-Score, and we verified this result using the R code provided by Ballester and Mitchell. We hypothesised that the cause of this difference in performance could be the result of the substantial difference in training set: we trained on the PDBbind 2016 refined set (excluding structures found in our test set, and structures for which features could not be computed), resulting in a total of 3,689 training structures. In contrast, Ballester and Mitchell trained on the PDBbind 2007 refined set (excluding structures found in the PDBbind 2007 core set), for a total of 1,105 training structures. While the increased size of the training set could be expected to result in improved performance, the additional data may also be more dissimilar to the structures in the 2007 core set than those found in the 2007 refined set, and so may not contribute to improved performance on the

chosen benchmark. To test this hypothesis, we re-trained the RF using RF-Score features on the PDBbind 2007 refined set (excluding structures found in our test set and structures for which features could not be computed). The resulting model achieved a Pearson correlation coefficient of 0.789 on our test set, which is comparable to the result reported by Ballester and Mitchell, indicating that the choice of training set has a significant impact on scoring function performance when assessed using the PDBbind 2007 core set as a benchmark.

As test set performance should not be used to tune the model during training, we did not modify our training set in light of this result. Instead, we computed the out-of-bag Pearson correlation coefficient of the RF using RF-Score features when trained on the PDBbind 2007 refined set and compared it to the out-of-bag Pearson correlation coefficient when the same RF was trained on the PDBbind 2016 refined set. Training on the PDBbind 2007 refined set resulted in an out-of-bag Pearson correlation coefficient of 0.705, while training on the PDBbind 2016 refined set resulted in an out-of-bag Pearson correlation coefficient of 0.743, validating our choice of training on the more recent version of the PDBbind refined set. This result illustrates the limitations of relying on a single benchmark set to assess scoring function performance. In Chapter 3 we will explore in detail the effect of training RF scoring functions on different versions of the PDBbind database, and how this affects performance on different versions of the PDBbind core set.

Figure 2.10A shows the predicted against measured pK values for the RF using RF-Score v3 features; Figure 2.10B shows the predicted against measured pK values for the RF using RF-Score v3 features combined with the RDKit features. The RF using RF-Score v3 features (Pearson correlation coefficient 0.732)

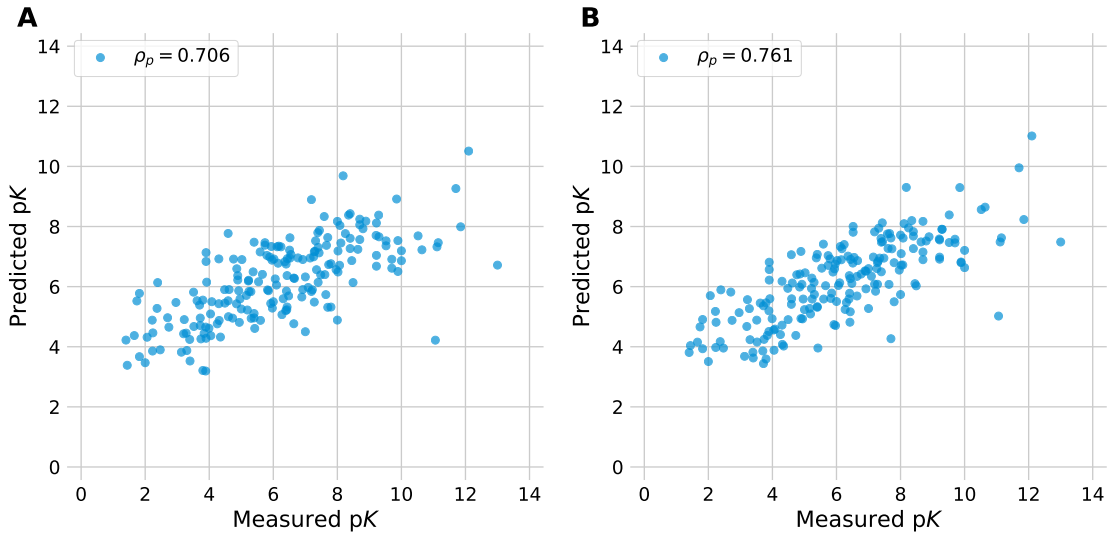


Figure 2.9: Predicted vs experimental pK values for the PDBbind 2007 core set. A: RF model using RF-Score features. B: RF model using RF-Score features and RDKit features.

out-performs the RF using either Vina features (Pearson correlation coefficient 0.683, Figure 2.8A) or RF-Score features (Pearson correlation coefficient 0.706, Figure 2.9A) separately. The RF using RF-Score v3 features and RDKit features out-performs the RF using RF-Score v3 features alone (Pearson correlation coefficient 0.772 versus 0.732), suggesting that each feature set adds different, useful information to the model.

Figure 2.11A shows the predicted against measured pK values for the RF using NNScore 2.0 features features; Figure 2.11B shows the predicted against measured pK values for the RF using NNScore 2.0 features combined with the RDKit features. The RF combining NNScore 2.0 features and RDKit features out-performs the RF using only NNScore 2.0 features (Pearson correlation coefficient 0.780 versus 0.750), and achieved the highest performance of all the models tested on the 2007 core set.

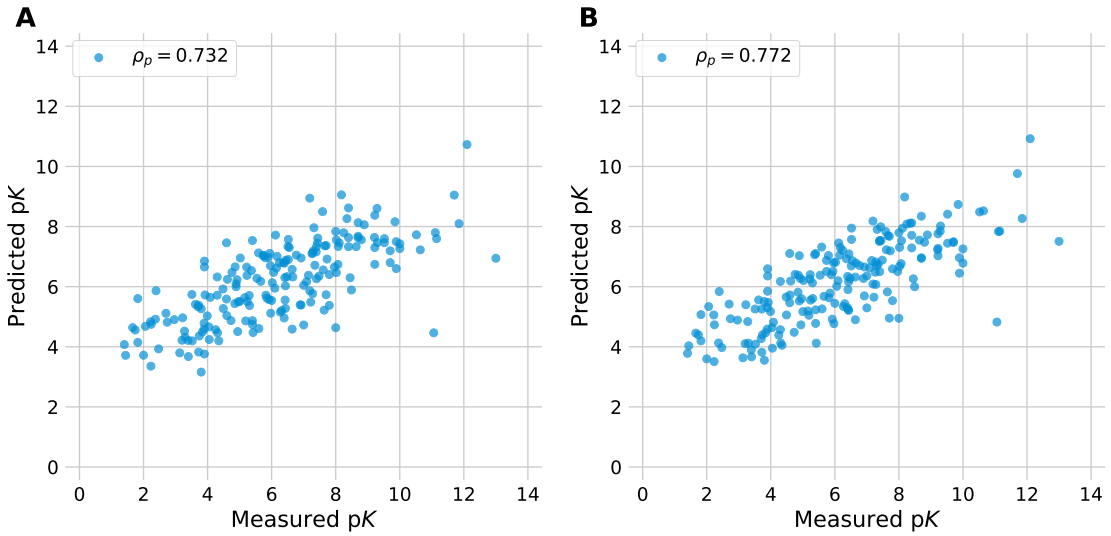


Figure 2.10: Predicted vs experimental pK values for the PDBbind 2007 core set. A: RF model using RF-Score v3 features. B: RF model using RF-Score v3 features and RDKit features.

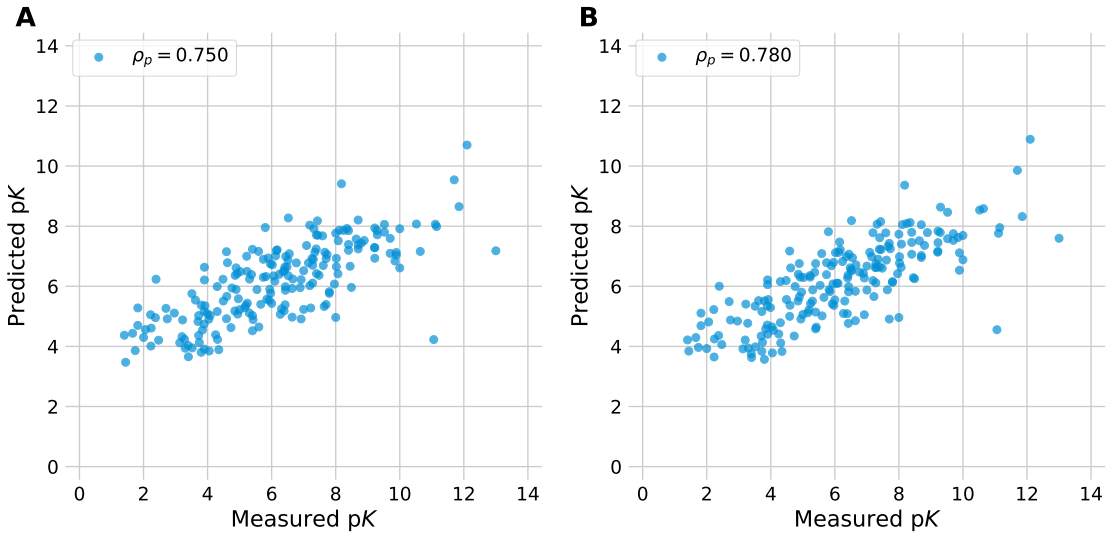


Figure 2.11: Predicted vs experimental pK values for the PDBbind 2007 core set. A: RF model using NNScore 2.0 features. B: RF model using NNScore 2.0 features and RDKit features.

2.3.4 | Using Fewer Features Does Not Diminish Performance

Next we investigated how the complexity of the model can be reduced by removing less-important features. The PCA and correlation matrices computed in Section 2.3.1 suggest that the dimensionality of each feature set can be reduced substantially without sacrificing information. To identify the features to be retained, we used the feature importance computed by the RF algorithm. For each feature set, a RF was trained on the PDBbind 2016 refined set, and the features ranked by the RF's feature importance. A series of new RFs were then trained; the first using only the highest-ranked feature, and each subsequent RF including the next highest-ranked feature. Figure 2.12 shows the Pearson correlation coefficient achieved on the PDBbind 2007 core set by RF models using up to 50 of the highest-ranked features from each feature set.

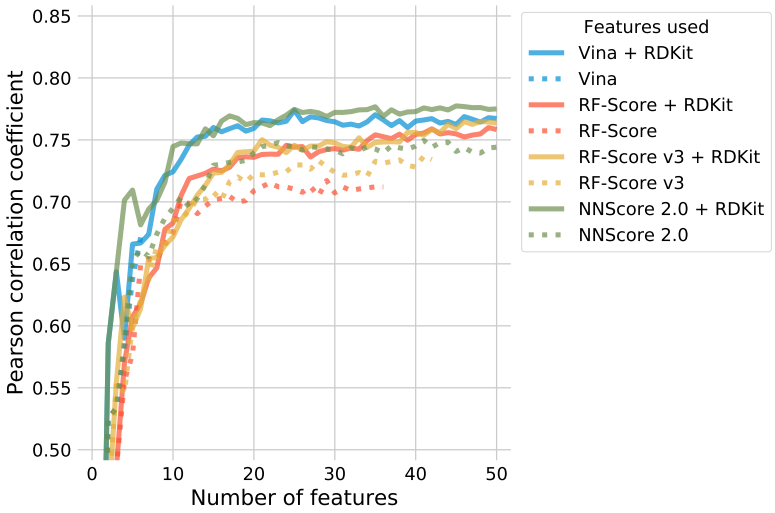


Figure 2.12: Performance of RF models using only the most important features.

For each feature set except Vina, which has only six features, using less than ten features results in diminished performance. With the exception of Vina, using only the top ten features from any feature set results in a Pearson correlation coefficient greater than 0.644, the highest reported for any of the classical scoring functions reported in CASF2009 (Cheng et al., 2009). Performance grows as more features are added, with steady improvement until the top twenty features are used. Beyond twenty features there is little improvement in performance, with the exception of models using RF-Score + RDKit or RF-Score v3 + RDKit features which show a small improvement as additional features are added. These results suggest that it is possible to substantially reduce the number of features used by these scoring functions without sacrificing predictive performance.

The PCA and feature correlation matrices discussed in Section 2.3.1 indicated that the variance in the data could be captured by a subset of the features in each feature set. The results shown in Figure 2.12 confirm that for each of these feature sets, the dimensionality of the model can be reduced substantially by removing correlated or low-information features without sacrificing model performance. When additional low-ranking features are included the performance does not degrade. This is because, if a feature is not useful, the decision trees in the RF will simply not use it if another, more useful, feature is available.

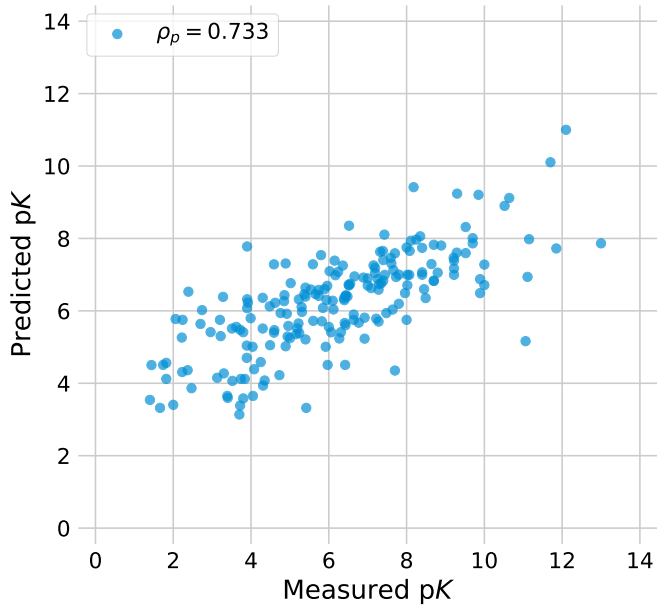


Figure 2.13: Predicted against experimental pK values on the PDBbind 2007 core set for a RF using RDKit molecular descriptors, trained on the PDBbind 2016 refined set.

2.3.5 | Ligand-Based Features Alone are Predictive of Binding Affinity

The RF using Vina terms and RDKit ligand molecular descriptors is strongly predictive of binding affinity, with a Pearson correlation coefficient of 0.768 on the PDBbind 2007 core set, comparable to that of the RF using the features of RF-Score, RF-Score v3, or NNScore 2.0. To determine the predictive power of the RDKit descriptors, we next trained a RF using *only* the RDKit descriptors on the PDBbind 2016 refined set. Figure 2.13 shows the predictions of this ligand-only model on the PDBbind 2007 core set. There is a strong correlation between the predicted and experimental pK, with a Pearson correlation coefficient of 0.733: higher than the 0.683 of the RF using the Vina features, but lower than the 0.768

of the RF using Vina + RDKit features.

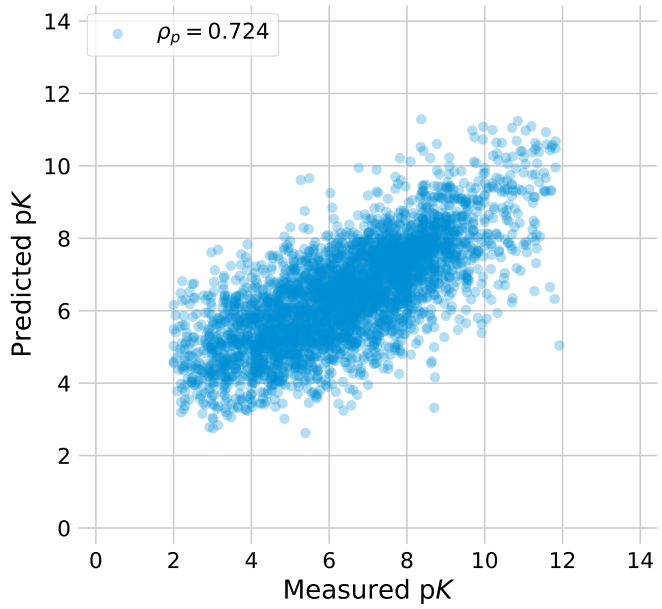


Figure 2.14: Out-of-bag predictions against experimental pK values for a RF using RDKit molecular descriptors, trained on the PDBbind 2016 refined set.

Figure 2.14 shows the out-of-bag predictions of the ligand-only model. The out-of-bag Pearson correlation coefficient is 0.724, very close to that achieved on the test set, suggesting that the model is generalising well to the unseen data. We verified that there was no strong correlation between any one of the RDKit descriptors and the pK values in the PDBbind 2016 refined set, so the predictive power of these features is not the result of a systematic bias in the chemical properties of the ligands represented in the data. Regardless of which structure-based features the RDKit descriptors were combined with, the resulting RF outperformed a RF using either the RDKit descriptors or the structure-based features alone, suggesting that some of the information captured by the ligand-based features is complementary to that captured by any of the structure-based features.

2.3.6 | RF Out-Performs Other Machine Learning Algorithms

We next explored the performance of different machine learning algorithms using each of the feature sets. Hyperparameters were tuned separately for each feature set-algorithm pairing using randomized search as described in Section 2.2.5. Figure 2.15 shows the mean Pearson correlation coefficient achieved by each feature set-algorithm pairing on a stratified five-fold cross-validation on the PDBbind 2018 refined set (excluding structure found in our test set). RF models achieved mean Pearson correlation coefficients from 0.65 (Vina features) to 0.77 (NNScore 2.0 + RDKit features), comparable with the performance of the RF models on the PDBbind 2007 core set in Section 2.3.3. XGBoost and RF achieve comparable performance using each feature set, with linear, neural network, and AdaBoost models achieving worse performance than either XGBoost or RF any given feature set. The poor performance of neural network models using NNScore 2.0 and NNScore 2.0 + RDKit features ($\rho_p = 0.6$ and $\rho_p = 0.58$ respectively) may be a result of not allowing a sufficiently large hidden layer during the parameter optimisation, as the NNScore 2.0 feature set is considerably larger than the Vina, RF-Score or RF-Score v3 feature sets. These results show that XGBoost and RF are both good algorithm choices when using each of our chosen feature sets.

It is also interesting to note that linear models achieved comparable performance to that of neural network and AdaBoost. Wójcikowski et al. (2018) also observed that a linear model achieved comparable results to neural networks and RF when using protein-ligand extended connectivity fingerprints

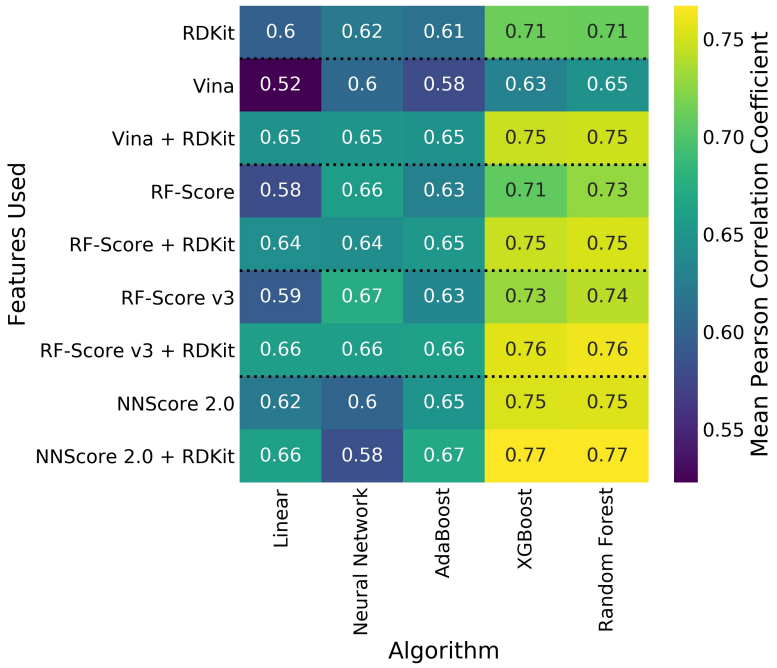


Figure 2.15: Performance of different machine learning algorithms and feature sets under stratified five-fold cross-validation.

as predictive features. These results indicate that using a more sophisticated machine learning algorithm does not necessarily lead to a more powerful predictive model. Instead, it is necessary to carefully tune and compare multiple algorithms in order to make the most appropriate choice for the task at hand.

Given the similar performance of XGBoost and RF, it is worth comparing other properties of these algorithms to determine which is most suitable for scoring function development. Both algorithms are based on ensembles of decision trees, and so both algorithms have similar definitions of feature importance to aid in interpreting the resulting models. XGBoost, like all gradient-boosting methods, fits each tree sequentially, making parallelism more difficult. In contrast, RF allows trees to be fitted in parallel, resulting in faster model training.

XGBoost performance is dependent on parameter tuning, while RF is generally robust with respect to hyperparameter choice (as illustrated in Section 2.2.5). Noise in the training data also affects XGBoost more strongly than RF, as the resulting training errors for noisy samples will result in those samples being continually prioritised at each stage of the boosting algorithm. As a result of these differences, RF is easier and faster to train than XGBoost, and since both algorithms achieved comparable cross-validation performance, we chose to focus on RF in subsequent experiments.

2.4 | Summary

In this chapter we explored the features used by four different scoring functions, and showed that combining these with a detailed set of computed molecular descriptors results in a scoring function with improved performance.

We found that RF regression models using four different sets of structure-based features were improved by the addition of a diverse set of computed molecular descriptors of the ligand. In particular, a RF using a combination of the interaction potential terms of the AutoDock Vina scoring function and the molecular descriptors of the ligand resulted in a scoring function with performance comparable to more complex machine-learning methods.

We investigated the use of several other machine learning algorithms, namely: elastic net linear regression, single-hidden-layer neural networks, AdaBoost, and XGBoost. We found that for all the feature sets considered, XGBoost models achieved performance comparable to those of RF models, while Linear, neural network, and AdaBoost models performed considerably worse than either XG-

Boost or RF. These results suggest that either XGBoost and RF are both good choices of regression algorithm for scoring function development.

We analysed the correlations between the features of each scoring function and found that, across the PDBbind 2016 refined set, many features were highly correlated. Performing PCA revealed that much of the variance in the data can be explained using a small number of features, and computing the correlation matrix for each feature set demonstrated that many of the features are correlated. We then used the feature importance computed by the RF algorithm to rank the features of each feature set, and trained new RFs using only the most informative features. For each feature set, model performance reached a plateau when only a subset of features were used, confirming that the dimensionality of each feature set can be reduced without sacrificing useful information.

Finally, we found that a RF using only the molecular descriptors of the ligand was strongly predictive of binding affinity, with both out-of-bag and test set Pearson correlation coefficients greater than 0.7, far exceeding that of the AutoDock Vina scoring function. These results suggest that there are properties of a ligand that are intrinsically useful for binding affinity prediction, and that machine learning scoring functions can be easily improved by the inclusion of a diverse set of molecular descriptors of the ligand.

3

Using Ligand-Based Features to Improve Binding Affinity Prediction

The majority of this chapter is based on work described in the following paper: F. Boyles, C. M. Deane, G. M. Morris. Learning from the Ligand: Using Ligand-Based Features to Improve Binding Affinity Prediction. *Bioinformatics*, btz665 (Boyles et al., 2019).

In Chapter 2 we explored the addition of ligand-based features to machine learning scoring functions, and the performance of different machine learning algorithms for binding affinity prediction. We found that including ligand-based features in a scoring function showed promising results for binding affinity prediction, with improved performance under cross-validation and the PDBbind 2007 core set benchmark, and showed that Random Forest (RF) and XGBoost consistently outperformed several other methods. In this Chapter we extend our analysis to the most recent version of the PDBbind database to examine how the composition of the training set affects scoring function performance. In particular, we explore the impact of protein and ligand similarity between the

training and test set. We also employ a leave-ligand-out and a leave-protein-out approach to model validation to determine how well the scoring functions generalise to unseen proteins or ligands, and how well they differentiate between similar structures. We investigate the apparent predictive power of models using only ligand-based features, and discuss the implications of these results for training and validating using PDBbind data.

3.1 | Introduction

In Chapter 2 we trained models on data from the PDBbind refined set and validated on the PDBbind 2007 core set. In this Chapter we perform a more detailed validation using multiple versions of the PDBbind core set, as well as leave-cluster-out validation, and examine how the similarity between the training and test data influences scoring function performance on each benchmark. The scoring functions used for protein-ligand binding affinity prediction and common approaches to scoring function validation were described in Chapter 1 Section 1.5. Here, we briefly review the research surrounding the use of PDBbind core set in the CASF benchmark of scoring function performance, and describe the limitations of this approach to scoring function validation.

3.1.1 | Evolution of the CASF Benchmark

Recognizing the need for a robust benchmark for scoring function performance against which advances in methodology can be critically assessed, the Wang group has published three scoring function benchmarking exercises: CASF 2009 (Cheng

et al., 2009), CASF 2013 (Li et al., 2014c,b) and CASF 2016 (Su et al., 2018), which use the core set derived from the PDBbind 2007, PDBbind 2013, and PDBbind 2016 releases respectively. These three core sets are a widely-adopted as a way of validating scoring function performance, and many publications report the performance of novel methods on one or more of the core sets. Together with the yearly updates of the PDBbind database, a common protocol for machine learning scoring function development has evolved: train on some subset of the PDBbind database, and test on the PDBbind core set(s). While this has resulted in a degree of standardisation in the training and validation of machine learning scoring functions, allowing for direct comparison of methods to establish the state-of-the-art, it is also important to consider the possible limitations of such a benchmark.

3.1.2 | Similarity Between Training and Test Data

The CASF exercises focus on 'classical' scoring functions (scoring functions not based on machine learning methods). As such, their suitability for benchmarking machine learning scoring functions, and how this relates to the data used to train such models, is largely ignored in the CASF publications. One machine-learning scoring function, $\Delta_{\text{vina}}\text{RF}_{20}$ (Wang and Zhang, 2017), was included in CASF 2016 and significantly outperformed all classical scoring functions at binding affinity prediction. The authors used the pre-trained $\Delta_{\text{vina}}\text{RF}_{20}$ out-of-the-box, noting that the training set used by Wang and Zhang, derived from the PDBbind database, included many of the structures used in the CASF 2016 exercise. Since RF can be expected to display near-perfect accuracy when tested

on its training set, it is somewhat unsurprising that this scoring function outperformed all others tested.

The PDBbind core sets are constructed by clustering the structures in the PDBbind refined set by sequence identity, and selecting a fixed number of representative structures for each cluster that span the range of experimentally-determined binding affinity values reported for those protein-ligand complexes. This means the core set is a diverse, representative set of the most common proteins in the PDBbind database without over-representation of any one protein. This selection mimics one of the properties desired of a scoring function: that they work out-of-the-box on a wide range of protein targets. However, by constructing the core set in this manner, it is necessarily representative of the most common proteins in the PDBbind database, so when a method is trained on PDBbind data and tested on a core set, there will be a large degree of protein structural and sequence similarity between the training and test data.

As there should be no overlap between training and test sets for any method, care must be taken when assessing a pre-trained scoring function on any community benchmark. It is also important to consider how similarity between complexes in the training and test sets can influence performance. Several authors have approached the issue of similarity between proteins and ligands in the training and test data, the primary result being that when proteins with high structural or sequence similarity to those in the core set are excluded from the training data, performance on the core set benchmark drops considerably (Li and Yang, 2017; Li et al., 2018). Kramer and Gedeck (2010) argued that rather than testing on the PDBbind core set, it is better to cluster the PDBbind database and test on each cluster individually in a leave-one-cluster-out manner. This ap-

proach has two advantages: first, it ensures that the proteins in the test set are not represented in the training set, so the scoring function is required to generalise to an unseen protein, and second, it makes it possible to identify proteins for which the scoring function performs particularly well (or poorly). The disadvantage of this approach is that, other than a few well-represented proteins (such as HIV-1 protease or carbonic anhydrase 2) the size of each cluster will be considerably smaller than any of the core sets, often less than 50 structures. Such a small sample size makes it difficult to quantify the performance of a scoring function, or to compare scoring functions, as there will be large confidence intervals for any performance metric.

3.1.3 | Sample Size and Uncertainty

The effect of sample size on the statistical significance of results in the context of scoring function development has received little attention in the literature. Carlson (2013) highlights the implications of a small test set on the ability to compare different scoring function, and bootstrapped confidence intervals were adopted for the CASF 2016 benchmark (Su et al., 2018). Yang et al. (2019) have addressed Carlson's comments in the context of the CASF benchmark and argue that the size of the PDBbind 2016 core set is adequate to make meaningful comparisons between scoring functions.

Another important consideration when training and validating models using experimental data is the experimental uncertainty associated with the data. The experimental error in publicly-available inhibition constant measurements and its consequences for the theoretical upper limit on the accuracy of binding affin-

ity predictions has been quantified (Kramer et al., 2012), as has the compatibility of public inhibition constant and IC_{50} data (Kalliokoski et al., 2013). This is particularly relevant to the training and validation of methods using PDBbind data as experimental errors are not readily available for the affinity data and, in the case of the PDBbind general set, K_i, K_d, and IC_{50} data are used interchangeably.

However, despite this growing body of work, there is still no community standard for benchmarking the performance of machine-learning scoring functions that accounts for similarity between training and test samples, or the experimental uncertainty in the data. We have not attempted to develop a new standard for benchmarking; but instead have explored how all of these factors influence the performance of our models and discuss the implications of our results for the development of machine-learning scoring functions.

3.2 | Materials and Methods

The full features, algorithms, and data used in this Chapter are described in Chapter 2 Section 2.2 'Materials and Methods'. Here we describe the approaches to training and validation used in this Chapter.

3.2.1 | Data

In this chapter we extended the analysis presented in Chapter 2 to the 2018 release of the PDBbind database. At the time of writing, this is the most recent version of the database, and contains 16,151 protein-ligand complexes in the general set, with 4,463 of these complexes in the refined set. In Chapter 2 we

used the refined set as our primary source of training data; however, it has been reported that including the lower-quality data comprising the remainder of the general set can still improve the performance of machine learning scoring functions (Li et al., 2015a), so we now extended our analysis to the general set. RDKit, Vina, RF-Score, RF-Score v3, and NNScore 2.0 features were computed for all complexes in the PDBbind 2018 general set, as described in Chapter 2 Section 2.2.2 'Features'.

In Chapter 2 we used the PDBbind 2007 core set (the validation set in CASF2009) as a validation set and trained on the non-overlapping data in the PDBbind 2016 refined set. In this Chapter we additionally test on the PDBbind 2013 and 2016 core sets, used as the validation sets in CASF2013 and CASF2016 respectively. Excluding proteins and ligands that could not be parsed by OpenBabel or RDKit, the 2007, 2013, and 2016 core sets contain 196, 180, and 276 structures respectively. The PDB codes of the structures used from the 2007, 2013, and 2016 core sets are listed in Appendix Tables A.1, A.2, and A.3 respectively.

These test sets are relatively small, making it difficult to identify statistically-significant differences in results generated by different models (see Figure 3.14). To address this shortcoming we constructed a new validation set by combining the structures from each core set, with duplicate structures removed. This 'combined core set' numbers 525 structures, almost twice the size of the PDBbind 2016 core set. The distribution of pK values for these four test sets are shown in Figure 3.1.

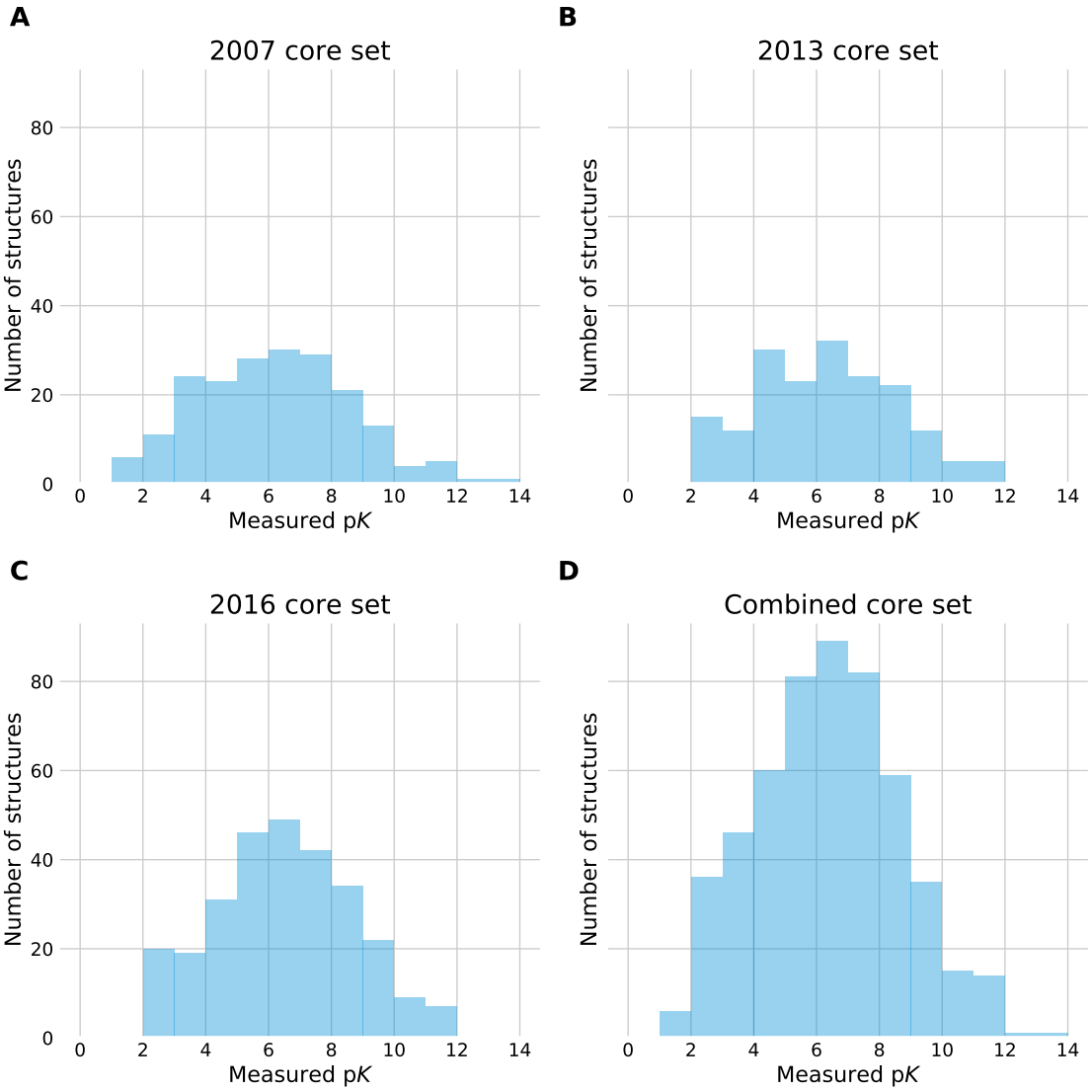


Figure 3.1: Distribution of experimentally-determined pK for the structures in the 2007, 2013, 2016, and combined core sets.

3.2.2 | Varying Training Set Size and Composition

We investigated the effect of four aspects of training set composition on the performance of our models: training set size; training data quality; similarity of ligands between training and test examples; and similarity of proteins between training examples.

Training Set Size and Quality

To examine the effect of training set size, we simulated the effect of adding more structural and affinity training data over time by restricting the training set to the six annual releases of the PDBbind database from 2013 through to 2018. Each release contains more data than the previous releases. We also trained separately on the general and refined sets of each year, to explore two different scenarios: a larger data set of varying quality, and a smaller data set with strict quality controls. The size of the general set and the refined sets from 2013 to 2018 inclusive, excluding the 525 core set structures, are shown in Figure 3.2. The number of general set structures available for training when using the 2018 version is twice that of the 2013 version (14803 versus 7364), while the number of refined set structures available for training is approximately 60% greater (3800 vs 2373).

Protein Similarity

To study the effect of including similar proteins in the training and test sets, for each version of PDBbind we constructed a series of training sets by removing from the PDBbind general and refined sets any structures with a protein sequence identity to any protein in the test set above a threshold of sequence

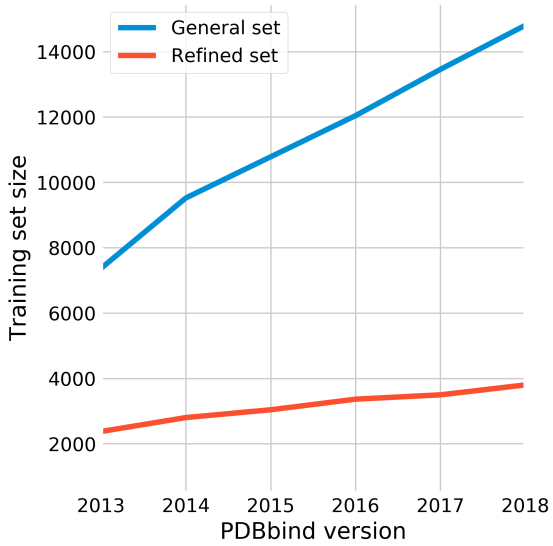


Figure 3.2: Training set size when using different versions of the PDBbind database.

identity. We used the BLASTclust clustering of the PDB provided on the PDB website[1]. The size of the resulting training set obtained from the PDBbind 2018 general and refined sets at each level of sequence identity is shown in Figure 3.3. For both the general set and the refined set, there is a large drop in training set size when structures with 100% sequence identity to any test set protein are removed: from 14804 to 10761 structures for the 2018 general set and from 3800 to 2371 structures for the 2018 refined set, or a reduction in size of 27.3% and 39.0% respectively. This shows that much of the similarity between training and test set structures is the result of the presence of many different structures of the same protein in the PDBbind database.

[1]http://www.rcsb.org/pdb/statistics/clusterStatistics.do, last accessed 13/05/2019

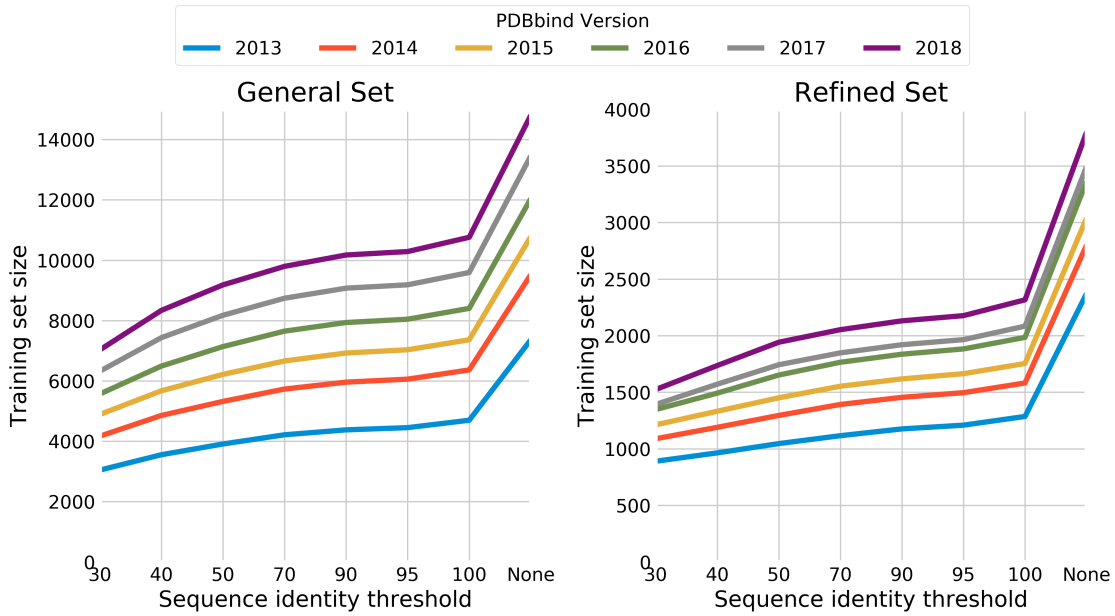


Figure 3.3: Training set size when using different versions of the PDBbind database and excluding structures with protein sequence identity above a threshold value to any core set structure.

Ligand Similarity

To investigate the effect of including similar ligands in both the training and test sets, we used RDKit to compute the Tanimoto similarity between the Morgan fingerprints (radius 2 and 2048 bits) of each pair of ligands. We then constructed a series of training sets by removing from the PDBbind 2018 general or refined sets any structure with a ligand Tanimoto similarity above a threshold value. This threshold of Tanimoto similarity ranges from 0.1 to 1.0 in increments of 0.1. The number of structures in the training set obtained from the PDBbind 2018 general and refined sets at each level of Tanimoto similarity is shown in Figure 3.4. The proportion of structures with ligand Tanimoto similarity of 1 to those in the core sets is much lower than that of proteins with 100% sequence identity:

from 14806 to 14011 structures for the 2018 general set and from 3800 to 3449 for the 2018 refined set, a reduction in size of 5.4% and 9.2% respectively (*cf.* 27.3% and 39.0% respectively).

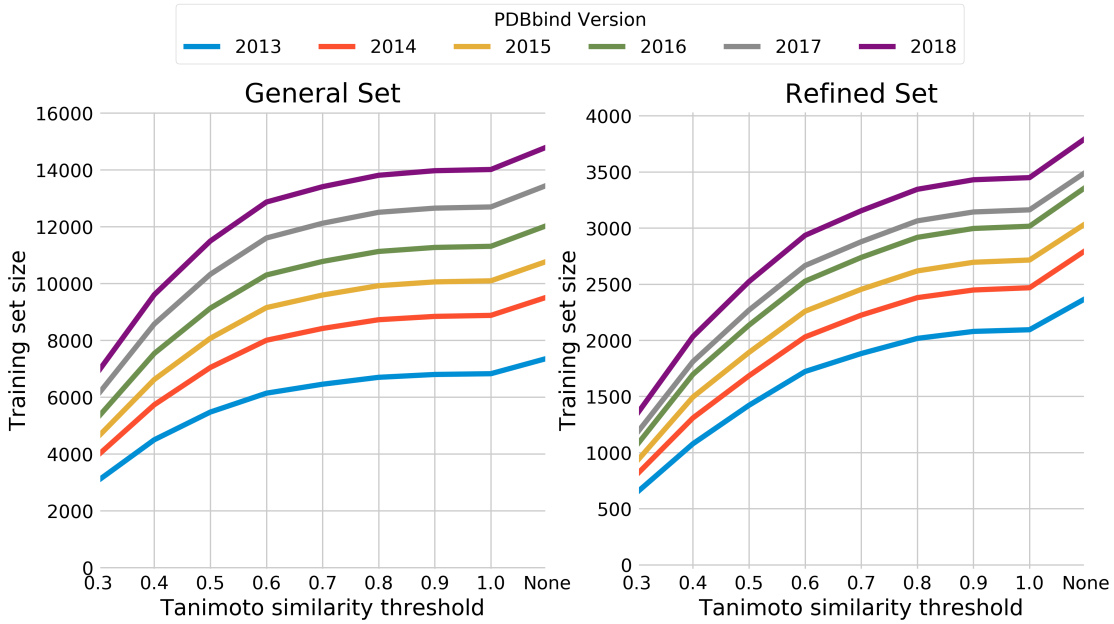


Figure 3.4: Training set size when using different versions of the PDBbind database and excluding structures with ligand Tanimoto similarity above a threshold value to any core set structure.

The number of distinct clusters of proteins and unique 'singleton' proteins in each version of the PDBbind refined set when clustered at different levels of sequence identity are shown in Figure 3.5. Similarly, the number of distinct clusters of ligands and unique singleton ligands in each version of the PDBbind refined set when clustered at different levels of ligand Tanimoto similarity are shown in Figure 3.6. With the exception of 2017 refined (in which there is an overall drop in the number of singleton proteins), the number of both clusters and unique proteins and ligands increases monotonically with successive PDB-

bind releases at each level of sequence identity or Tanimoto similarity. This indicates that as the size of the data set grows, so too does its diversity in proteins and ligands.

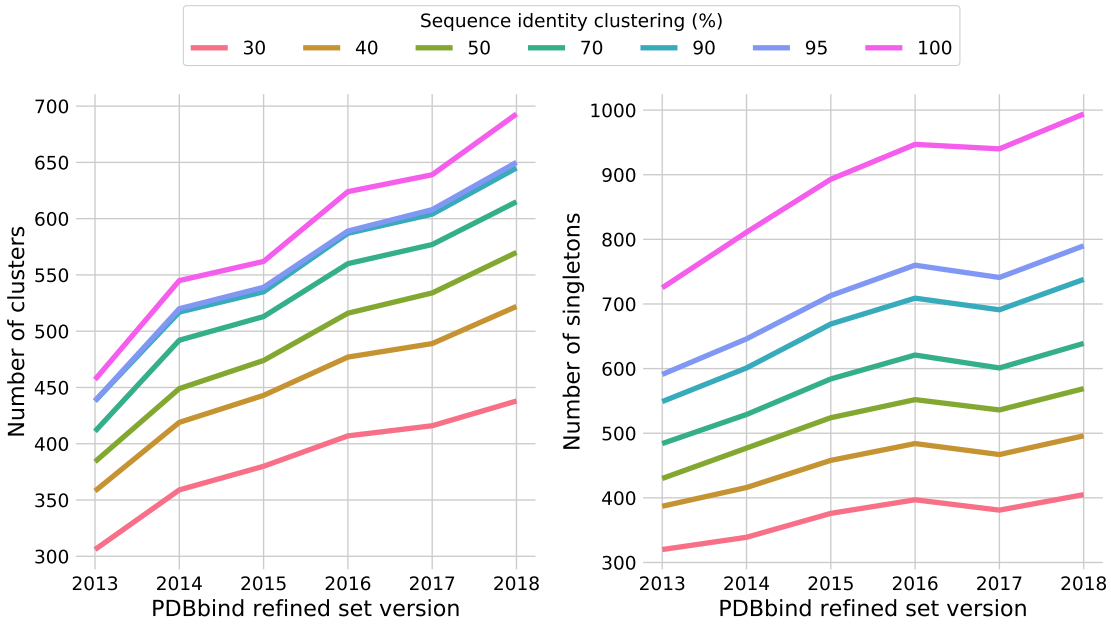


Figure 3.5: Number of distinct protein clusters (left) and unique protein singletons (right) when the PDBbind 2013 to 2018 refined sets are clustered at different levels of sequence identity.

When excluding test-set similar complexes from the training data, we treated each test set separately. For example, when testing on the PDBbind 2016 core set, only proteins similar to those found in the 2016 core set were excluded from the training set. Regardless of the choice of training and test set composition, all core set structures were always excluded from the training set.

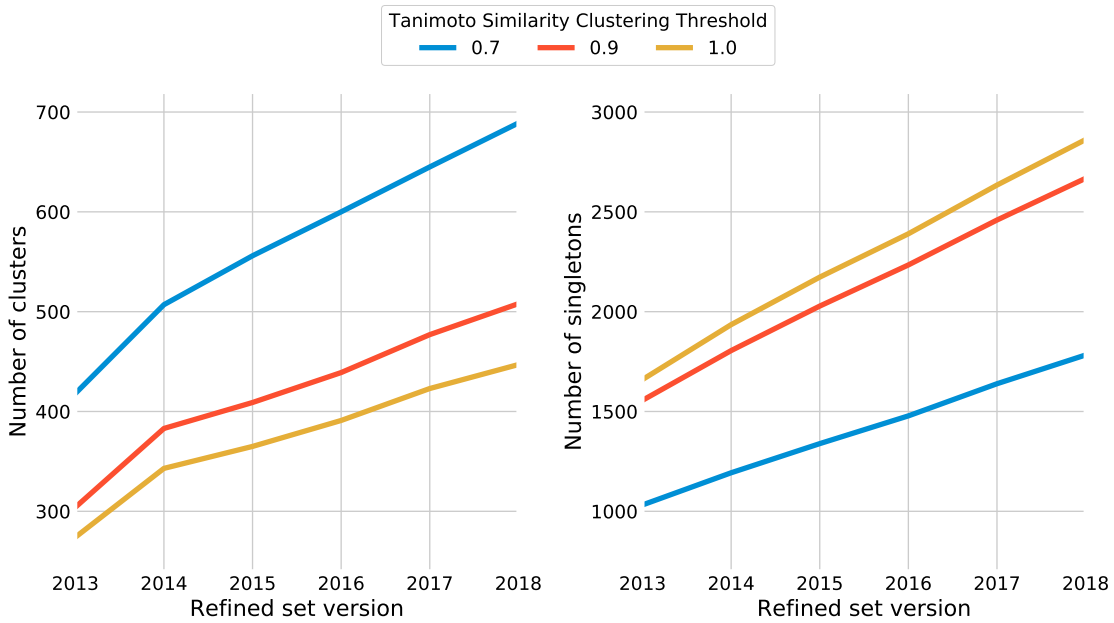


Figure 3.6: Number of distinct ligand clusters (left) and unique ligand singletons (right) when the PDBbind 2013 to 2018 refined sets are clustered at different levels of sequence identity.

3.2.3 | Leave-Cluster-Out Validation

To investigate how each scoring function differentiates between different complexes containing the same ligand, we identified groups of structures in the PDBbind 2018 general set that feature the same ligand. We clustered the structures of the PDBbind 2018 general set by the three-character chemical component ID of the ligand as specified in the PDB, and selected all ligands found in at least three structures. Holding out each ligand in turn as a test case, each model is trained on all structures not containing that ligand.

Similarly, to investigate how each scoring function generalises to a previously-unseen protein, we group the structures in the PDBbind 2018 general set by protein and employ a leave-one-out validation strategy. We perform this grouping

in two ways. First, we group the structures of the PDBbind 2018 general set by the UniProt ID of the protein, with the largest groups (at least 50 structures) chosen as validation sets. The UniProt IDs and names of these 50 proteins are shown in Table 3.1. Second, we cluster the structures in the PDBbind 2018 general set at 30% protein sequence identity and choose the largest clusters (at least 20 structures) as validation sets. The PDB ID and name of a protein representative for each cluster are shown in Table 3.2. This results in a smaller number of validation sets, and ensures the model cannot learn from similar structures in the rest of the database. In both cases, for each validation set, the training set comprises all remaining structures in the PDBbind 2018 general set.

3.2.4 | Performance Measurement

As in Chapter 2, models are assessed by computing the Pearson correlation coefficients between the predicted and experimental pK values for the samples in the test set. All Pearson correlation coefficients are computed using the Python package SciPy (Jones et al., 2001).

To estimate confidence intervals on the Pearson correlation coefficient, we take 10,000 bootstrap samples of equal size to the test set and computed the ρ_{p} for the bootstrap sample. The two-tailed 95% confidence interval was then taken to be the 2.5% and 97.5% percentiles of the bootstrapped values of ρ_{p}.

For each set of predictions on a test set, we took $m = 10,000$ random permutations of the predicted pK values and computed ρ_{p} between the permuted predictions and the unpermuted experimental pK values. Under the null hypothesis that any correlations are due to random chance, the p-value for the

UniProt ID	Protein Name
O14757	Serine/threonine-protein kinase Chk1
O14965	Aurora kinase A
O60674	Tyrosine-protein kinase JAK2
O60885	Bromodomain-containing protein 4
P00489	Glycogen phosphorylase, muscle form
P00517	cAMP-dependent protein kinase catalytic subunit alpha
P00533	Epidermal growth factor receptor
P00734	Prothrombin
P00742	Coagulation factor X
P00749	Urokinase-type plasminogen activator
P00760	Cationic trypsin
P00797	Renin
P00811	Beta-lactamase
P00918	Carbonic anhydrase 2
P02766	Transthyretin
P03366	Gag-pol polyprotein
P03367	Gag-pol polyprotein
P03372	Estrogen receptor
P04585	Gag-pol polyprotein
P07900	Heat shock protein HSP 90-alpha
P08581	Hepatocyte growth factor receptor
P11309	Serine/threonine-protein kinase pim-1
P18031	Tyrosine-protein phosphatase non-receptor type 1
P19491	Glutamate receptor 2
P23639	Proteasome subunit alpha type-2
P24941	Cyclin-dependent kinase 2
P26663	Genome polyprotein
P27487	Dipeptidyl peptidase 4
P34913	Bifunctional epoxide hydrolase 2
P48736	Phospshoinositide 3-Kinase 3-kinase gamma
P56817	Beta-secretase 1
Q00987	E3 ubiquitin-protein ligase Mdm2
Q16539	Mitogen-activated protein kinase 14
Q9H2K2	Poly [ADP-ribose] polymerase tankyrase-2
Q9Y233	cAMP and cAMP-inhibited cGMP 3',5'-cyclic phosphodiesterase 10A

Table 3.1: UniProt IDs of PDBbind targets. The three entries for Gag-pol polyprotein, P03366, P06672, and P04585, correspond to three different isolates of human immunodeficiency virus type 1 group M subtype B, namely: BH10, BRU/LAI, and HXBU, respectively.

Cluster	Representative Protein Name	PDB ID
1	Antibody Fab fragment	1A4K
2	Cyclin-dependent kinase 2	1B38
3	Prothrombin	1A4W
4	Progesterone receptor	1A28
5	HIV-1 protease	1A30
6	Penicillopepsin-1	1APV
7	Bromodomain-containing protein 4	2YEL
8	Adipocyte lipid-binding protein	1ADL
9	Carbonic anhydrase 2	1AVN
10	Purine nucleodise phosphorylase	1A69
11	Acetylcholine-binding protein	1UV6
12	Transthyretin	1BM7
13	Glutamate receptor 2	1FTM
14	cAMP-specific 3′,5′-cyclic phosphodiesterase 4B	1RO6
15	Myrosinase	1E6Q
16	Heat shock protein 90	1AMW
17	Protein tyrosine phosphatase 1B	1BZC
18	Stromelysin-1	1B8Y
19	Alpha thrombin	1BCU
20	HIV-1 integrase	3AO2
21	Histidine-binding protein	1HSL
22	Factor VIIa	5U6J
23	tRNA-guanine transglycosylase	1ENU
24	Pantothenate synthetase	3COW
25	FimH adhesin	1UWF
26	Factor Xa	1EZQ
27	Periplasmic oligopeptide-binding protein	1B05

Table 3.2: Representative proteins and PDB structures of clusters formed when the PDBbind 2018 general set was clustered at 30% sequence identity.

permutation test is given by $p = (b+1)/(m+1)$ where b is the number of permutations for which the value of ρ_p was at least as large as that obtained for the unpermuted predictions.

Where the confidence intervals overlapped for two models, we also apply the Mann-Whitney U test to the distributions of the bootstrapped values of the correlation coefficient to test for a statistically-significant difference in the performance of the models. We compute the Mann-Whitney U test statistic using SciPy.

3.3 | Results and Discussion

In Chapter 2 we showed that including a detailed set of physicochemical descriptors of the ligand in a scoring function improves binding affinity prediction across the protein-ligand complexes found in the PDBbind database. We also found that a model using only ligand-based features is able to predict binding affinity more accurately than the AutoDock Vina scoring function. In this chapter we explore in detail how the composition of the training set affects scoring function performance on the CASF benchmark sets, and under what circumstances the inclusion of ligand-based features in the scoring function is beneficial. We examine whether homology bias between the training and test sets leads to overly-optimistic assessment of performance when training on PDBbind data and testing on the CASF sets. Finally, we address the question of why the ligand-only model is predictive of binding affinity and whether this is indicative of limitations to current methods of benchmarking.

In these experiments we exclusively use RF, as it achieved the best perfor-

mance among the machine learning algorithms investigated in the cross-validation and benchmarking experiments discussed in Chapter 2.

3.3.1 | Ligand-Based Features Improve Scoring Function Performance

Figures 3.7 and 3.8 show the Pearson correlation coefficient between predicted and experimental pK achieved by each RF scoring function on five-fold cross-validation, the PDBbind 2007, 2013, and 2016 core sets, and the combined core set, when trained on the PDBbind 2018 general set and the PDBbind 2018 refined set, respectively. In all cases, a scoring function combining structure-based protein-ligand features with the RDKit features outperforms the corresponding scoring function using protein-ligand features alone. The difference between the Pearson correlation coefficient in each such case was found to be significant at 95% confidence (Mann-Whitney U test $p < 0.05$), however difference in performance is marginal for scoring functions using the RF-Score v3 and NNScore 2.0 features trained on the 2018 general set and tested on the 2016 core set (0.814 vs 0.821 and 0.819 vs 0.826), and for the scoring function using NNScore 2.0 features when trained on the 2018 refined set and tested on the 2013 core set (0.739 vs 0.742). Our best-performing models are competitive with state-of-the-art ML scoring functions reported in the literature, such as PLECScore (up to $\rho_{\mathrm{p}} = 0.83$ on the 2016 core set (Wójcikowski et al., 2018)), $\mathrm{K}_{\mathrm{DEEP}}$ ($\rho_{\mathrm{p}} = 0.82$ on the 2016 core set (Jiménez et al., 2018)) and RF-Score v3 ($\rho_{\mathrm{p}} = 0.803$ on the 2007 core set (Li et al., 2015b)). In particular, the RF using both Vina features and RDKit features

is competitive with RFs using RF-Score, RF-Score v3, and NNScore 2.0 features, achieving $\rho_{\mathrm{p}} = 0.840$, $\rho_{\mathrm{p}} = 0.749$, and $\rho_{\mathrm{p}} = 0.792$ respectively on the 2007, 2013, and 2016 core sets. The RF models trained on the PDBbind 2018 general set outperform those trained on the PDBbind 2018 refined set when tested on the core sets. There is no single model that performs best on the 2007, 2013, and 2016 core sets. Cross-validation performance of all models is lower than observed on the core sets. There are two factors that might contribute to this. First, the effective training set size during five-fold cross-validation is 80% of the size of the full training set. Second, the PDBbind general and refined sets are far more diverse than the core sets, including many examples of unique protein structures. Thus, under cross-validation, the test set contains a greater variety of structures many of which are not represented in the training data, and so we should expect performance to be lower.

3.3.2 | Effect of Training Set Composition on Scoring Function Performance

Next we consider how the composition of the training set, and its similarity to data found in the benchmark, affects our assessment of scoring function performance. We investigate three factors: the size of the training set, the similarity between proteins in the training and test set, and the similarity between ligands in the training and test set. Here, we present results obtained by testing on the combined core set. The Pearson correlation coefficient for each model on the combined core set when trained on different versions of the PDBbind general

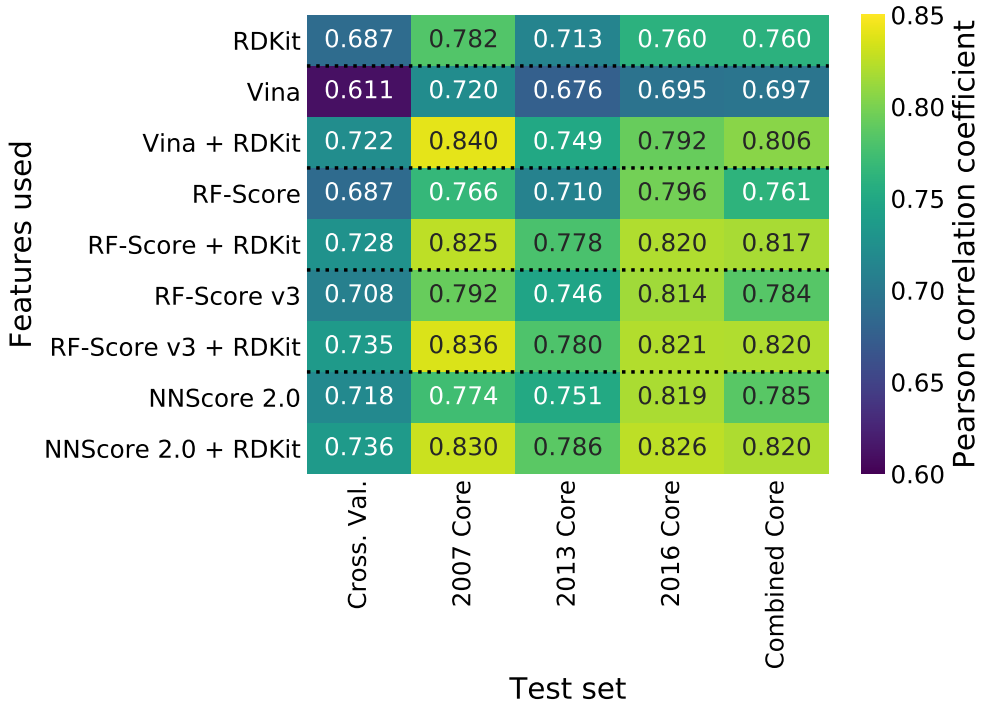


Figure 3.7: Pearson correlation coefficient of predicted against experimental affinity achieved by RF models trained on the PDBbind 2018 general set.

set is shown in Figure 3.11. Analogous results obtained by testing separately on the 2007, 2013, and 2016 core sets are available in Appendix Figures A.1 and A.2.

Figure 3.9 shows how the performance of each model on the combined core set varies with the level of sequence identity permitted between proteins in the training and test set. There is a significant drop in the performance of all scoring functions even when a sequence identity cutoff of 100% is imposed, *i.e.* when only proteins with identical sequence to those in the test set are excluded from the training set. Reducing the sequence identity cutoff from 90% to 50% has a smaller impact on performance than the initial imposition of a 100% cutoff. Further reducing the cutoff from 50% to 30% has a more apparent effect. This sug-

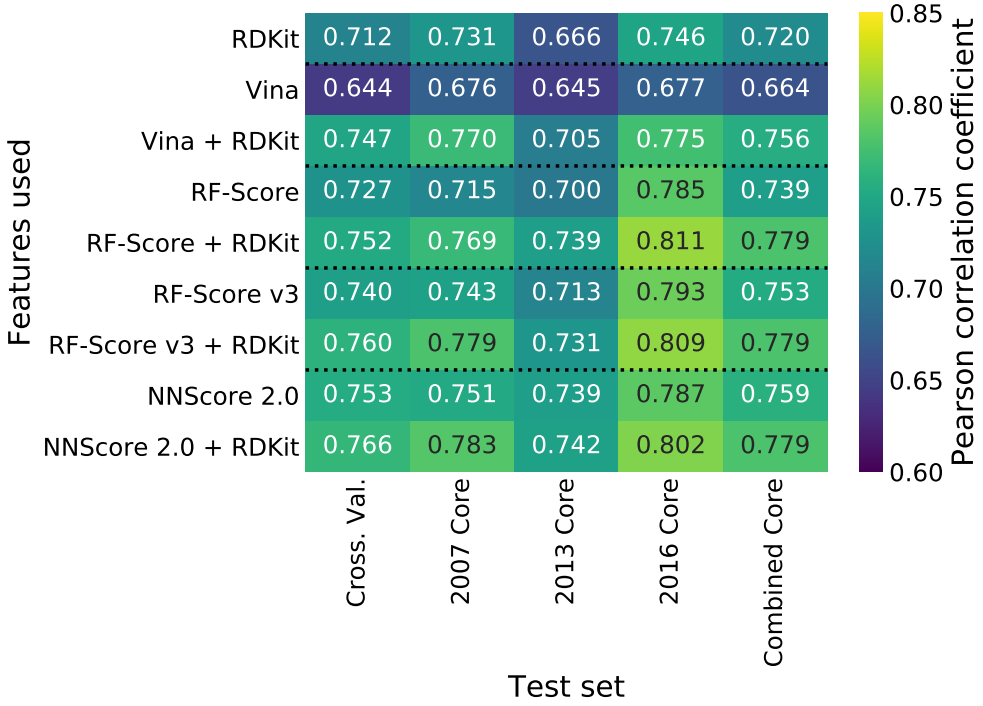


Figure 3.8: Pearson correlation coefficient of predicted against experimental affinity achieved by RF models trained on the PDBbind 2018 refined set.

gests that the performance of all models considered is dependent on the availability of relevant training data, with larger errors present when attempting to generalise to a novel protein target. Regardless of training set construction the addition of ligand-based features to a structure-based RF scoring function improves performance. The trend toward ligand-based features improving scoring function performance is exemplified by the RF using Vina features and RDKit features. This combination results in predictive performance comparable to that of the RFs using RF-Score, RF-Score v3, or NNScore 2.0 features, suggesting that a more complex and detailed set of features describing protein-ligand interactions is not necessarily more predictive than a comparatively simple set of force-

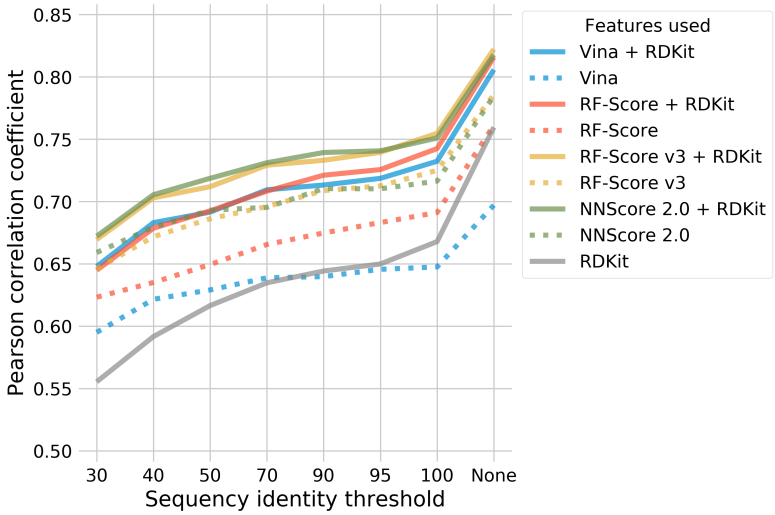


Figure 3.9: Pearson correlation coefficient achieved on the combined core set by RF models using different sets of features when trained on data from the PDBbind 2018 general set.

field-like terms (Vina features) and molecular descriptors of the ligand (RDKit features).

Figure 3.3 showed that the largest drop in training set size occurs when proteins with 100% sequence identity to those in the test set are excluded from the training set, with only a small decrease in training set size occurring when proteins with sequence identity of 90% or more are also excluded. This suggests that 'homology bias' due to the presence of similar proteins in the training and test sets is predominantly caused by the presence of many complexes of the same protein (100% sequence identity), rather than the presence of large numbers of similar or nearly-identical proteins. This explains the sharp drop in performance when excluding proteins from the training set at 100% sequence identity, since this eliminates almost all of the most-similar structures to those in the test set, with subsequent, stricter sequence identity thresholds removing fewer less-

similar structures.

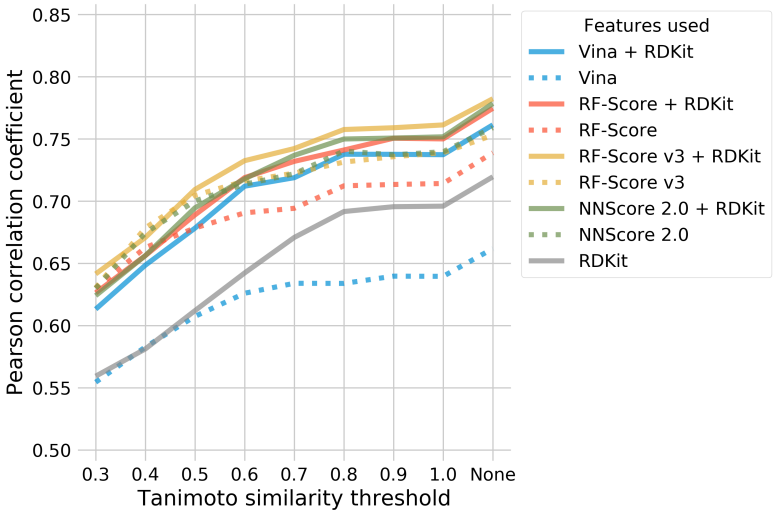


Figure 3.10: Pearson correlation coefficient achieved on the combined core set by RF models using different sets of features when trained on data from the PDBbind 2018 general set. The ligand ECFP4 fingerprint Tanimoto similarity above which structures similar to those in the test set were excluded from the training set is indicated on the horizontal axis.

The performance of each model on the combined core set when trained on the PDBbind 2018 general set, when ligands with an ECFP4 Tanimoto similarity above a set threshold to those in the test set are excluded from the training set, is shown in Figure 3.10. There is a small drop in the performance of every model when ligands with Tanimoto similarity of 1 to those in the test set are excluded from the training set; lowering the Tanimoto similarity threshold to 0.8 does not further decrease model performance. Figure 3.4 showed that most of the ligands with Tanimoto similarity of 0.8 or greater have Tanimoto similarity of 1, similarly to the case of protein similarity. Importantly, this overlap in protein and ligand similarity is not due to the presence of identical protein-ligand complexes in both the combined core set and the remainder of the general set, so the drop in

performance is not the result of memorisation of different structures of the same complex.

Performance of all models starts to drop at a Tanimoto similarity threshold of 0.7, with the models using ligand-based features dropping off more rapidly. Further decreasing the Tanimoto similarity threshold results in decreased performance for every model. Models using the ligand-based features suffer a more rapid decline in performance, with the the combination of structure-based and ligand-based features ceasing to outperform the structure-based features alone once ligands with Tanimoto similarity of 0.5 or greater are excluded from the training data, and the performance of the ligand-only model becomes comparable to that of the RF using Vina features.

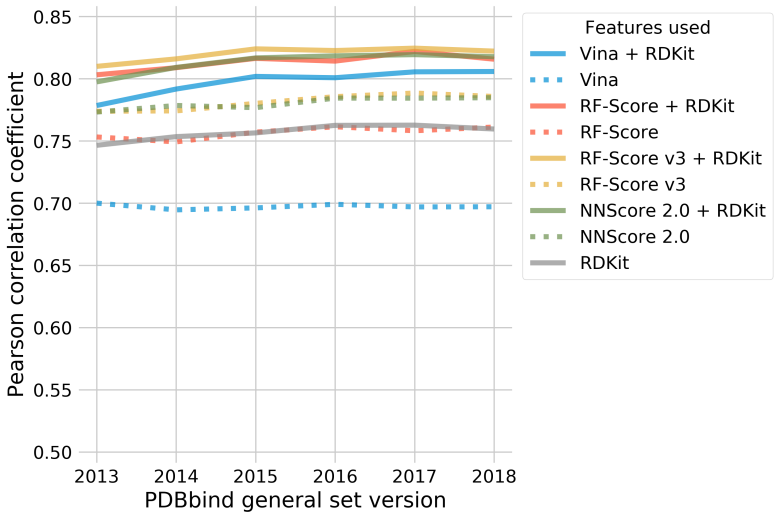


Figure 3.11: Pearson correlation coefficient achieved on the combined core set by RF models using different sets of features when trained on different versions of the PDBbind general set.

Overall, there is little improvement in performance when larger, more recent versions of the general set are used, suggesting that more training data does not

necessarily translate to improved performance. This is contrary to the results of Li et al. (Li et al., 2015a) who found that a larger training set resulted in improved performance. The smallest training set used here is larger than the smallest training set used by Li et al., suggesting an element of learning saturation, with diminishing returns once the training set reaches a certain size. Similar results are obtained when using different versions of the PDBbind refined set in place of the general set (see Appendix Figures A.3 and A.4); however, there is a systematic increase in performance when the general set is used in place of the refined set for a given version of PDBbind, suggesting a difference in composition of these sets. This difference in performance between models trained on the general and refined sets vanishes when complexes with ligands with Tanimoto similarity of 0.9 or greater are excluded from the training set (see Appendix Figures A.5 and A.6), and is greatly reduced when structures with 90% protein sequence identity to those in the test set are excluded. This suggests that the increase in performance when training on the general set can be attributed to increased representation of the core set proteins and ligands in the general set.

To better understand these results, we also investigated the performance of the scoring functions on bootstrapped samples of each version of the refined set when trained on the rest of the data from the same version of the refined set. We find that the mean correlation coefficient achieved by each scoring function did not vary with the version of the refined set used, as shown in Figure 3.12. The variance of the performance does not increase with the larger, more diverse versions of PDBbind; this is illustrated by the box plots of the Pearson correlation coefficients achieved by the RF using RF-Score v3 and RDKit features on the bootstrapped test sets shown in Figure 3.13.

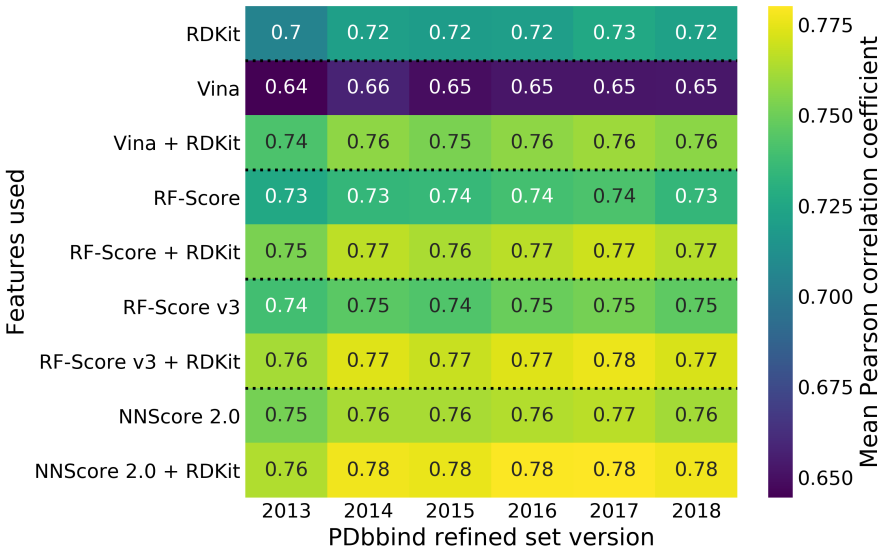


Figure 3.12: Mean Pearson correlation coefficient achieved by RF models using each feature set trained on different versions of the PDBbind refined set and tested on bootstrapped samples from the same version of the refined set.

As shown in Figure 3.5 the number of distinct clusters of structures at different sequence identity thresholds increases with each release of PDBbind. The number of distinct clusters of ligands also increases with each release of PDBbind, as shown in Figure 3.6. This suggests that by using a larger, more diverse training set, the domain of applicability of the scoring function grows, allowing it to generalise to a more diverse test set without affecting performance. This is consistent with the fact that performance on a constant benchmark does not change when training on more recent versions of PDBbind.

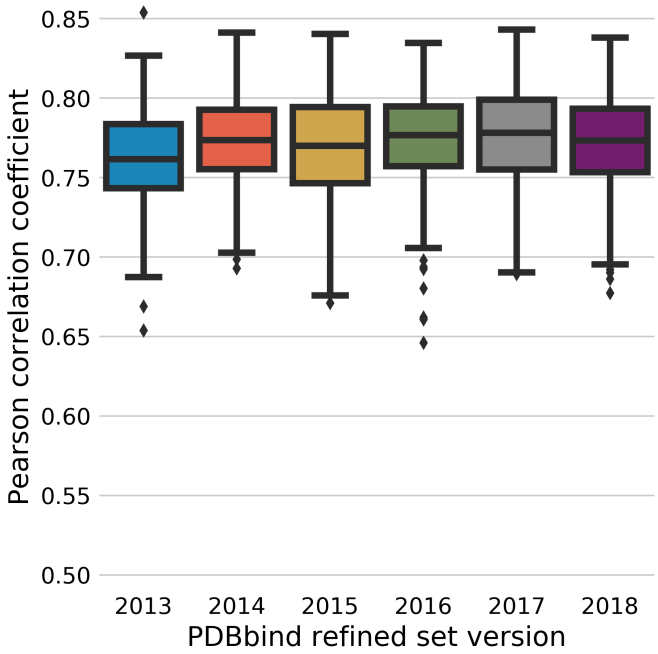


Figure 3.13: Box plot of the Pearson correlation coefficients achieved by RF models using RF-Score v3 and RDKit features trained on different versions of the PDBbind refined set and tested on bootstrapped samples from the same version of the refined set. The box shows the mean and quartiles of the distribution of the Pearson correlation coefficient across the bootstrapped samples, with outliers shown by markers.

3.3.3 | Sample Size Strongly Impacts Confidence in Scoring Function Performance

Figure 3.14 shows the 95% confidence intervals estimated using bootstrap sampling for the Pearson correlation coefficient RF models trained on the PDBbind 2018 general set and tested on the 2007, 2013, 2018, and combined core sets. The confidence intervals obtained when testing on the 2007 and 2013 core sets ($n < 200$), and are most narrow on the combined core set ($n = 525$). There is near-total overlap of the confidence intervals for the models using RF-Score + RDKit,

RF-Score v3, RF-Score v3 + RDKit, NNScore 2.0, and NNScore 2.0 + RDKit features on the 2016 core set, corresponding to the lack of discernible difference in performance of these models on this core set when trained on the PDBbind 2018 general set (Figure **??**).

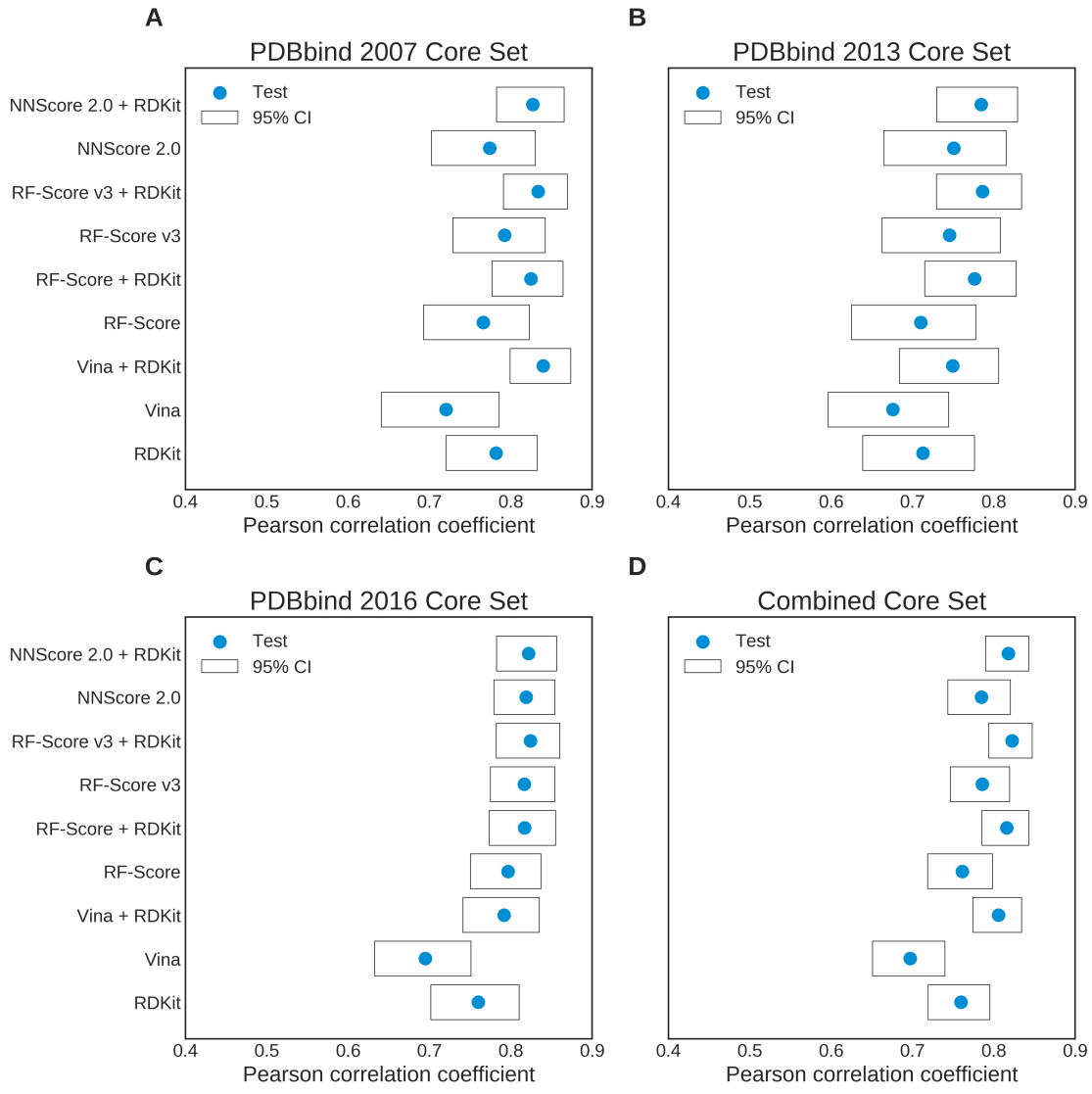


Figure 3.14: Bootstrapped 95% confidence interval on the Pearson correlation coefficient achieved by each RF scoring function on each test set when trained on the PDBbind 2018 general set.

3.3.4 | Ligand-Based Features are Predictive of Mean Binding Affinity

The results presented up to this point show that a purely ligand-based model is predictive of protein-ligand binding affinity for complexes found in the PDBbind database. This should not work - we should not expect to be able to predict the affinity of a ligand for its protein binding partner without knowing anything about that protein. To determine the reason for the success of the ligand-based model, we investigate cases where there are multiple structures featuring the same ligand bound to different proteins with different binding affinities. A structure-based method should be able to distinguish between these, whereas a ligand-based method cannot. We first identify the most common ligands in the PDBbind 2018 general set and, for each, train the RF model using RDKit features on all complexes that do not contain that ligand. The single value predicted for each ligand, together with the set of experimentally-reported affinities of that ligand in different complexes, are shown in Figure 3.15. With the clear exception of biotin (BTN), the marker is often close to the centre of the swarm plot, indicating that the model appears to be estimating the mean affinity of the ligand even when the experimental data have a range of several pK units. Many of the structures featuring biotin are biotin-streptavidin or biotin-avidin complexes, explaining the incredibly high experimental affinity values. Since ensemble-based methods such as RF cannot extrapolate beyond the range of values seen during training, it is unsurprising that the model cannot predict such a high average affinity when no such examples are included in the training set.

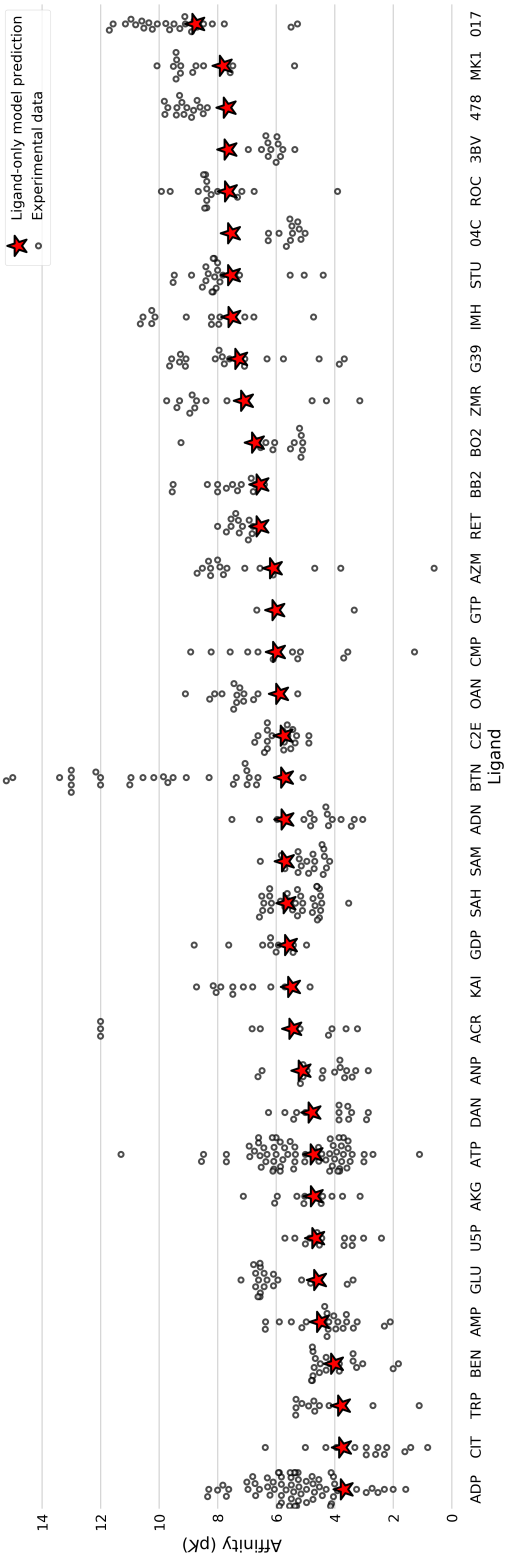


Figure 3.15: Affinity predicted by ligand-only model (red marker) and experimental affinity data (black swarm plot) for the most common ligands in the PDBbind 2018 general set. The three-character chemical ID of each ligand is indicated on the horizontal axis. For each ligand, the ligand-only model was trained on all structures in the PBDbind 2018 general set that did not contain that ligand.

When the RF model using RDKit features is tested on a previously-unseen ligand, having been trained on all other data in the PDBbind 2018 general set, the score is strongly correlated with the mean experimental pK of that ligand for its targets across the PDBbind 2018 general set ($\rho_{\mathrm{p}} = 0.71$, as shown in Figure 3.16). For the most common ligands in the PDBbind 2018 general set, the reported affinity values can span several orders of magnitude, so the RF model using RDKit features is not simply predicting a single 'correct' value for a ligand that happens to have many similar affinity measurements.

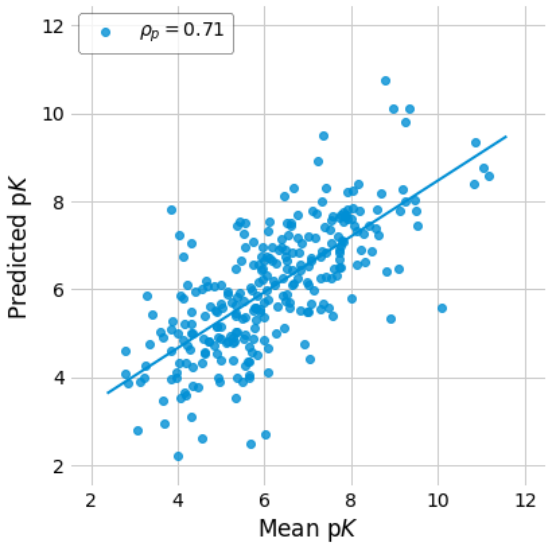


Figure 3.16: Ligand-only model predicted pK (vertical axis) against mean experimental pK (horizontal axis) for ligands found in multiple structures in the PDBbind 2018 general set. Each marker represents a single ligand; the straight line denotes a linear regression fit through the points. For each ligand, the ligand-only model was trained on the PDBbind 2018 general set excluding all complexes containing that ligand, and tested only on that ligand. The line denotes a linear regression through the points.

When nearly-identical ligands (Tanimoto similarity > 0.9) are excluded from the training data, this correlation is actually stronger (ρ_{p} = 0.74, Figure 3.17 A).

The correlation remains strong ($\rho_p > 0.7$) when ligands with Tanimoto similarity greater than 0.7 to the test ligand are excluded from the training data (Figure 3.17 C), while a moderate correlation ($\rho_p \approx 0.6$) remains when ligands with Tanimoto similarity greater than 0.4 to the test ligand are excluded from the training data (Figure 3.17 F). When ligands with Tanimoto similarity greater than 0.1 to the test ligand are excluded from the training data, there is little correlation between the predicted and mean experimental pK (Figure 3.17 I).This gradually weakening correlation suggesting that the ligand-only model is able to make inferences about similar molecules, rather than simply memorizing affinity measurements for near-identical molecules.

Structure-based scoring functions should be able to differentiate between different complexes featuring the same ligand. To test this, we repeated the above experiment with each set of structure-based features, and computed the correlation coefficient between the predicted and experimental pK values for the structures featuring each ligand. These are shown in Figure 3.18. In most cases, there is no correlation between the predicted and experimental pK, and there is no consistent trend toward a certain model performing well on certain sets of ligands. As an example of this behaviour Figure 3.19 shows the predicted against experimental pK values for all structures featuring the ligand ADP from models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features. The models using RF-Score v3 and RF-Score v3 + RDKit features are both unable to accurately predict the pK values associated with the different structures, achieving no meaningful correlation between predicted values and experimental values. The single value predicted by the model using only ligand-based RDKit features is lower than any predicted by the two structure-based models. The predictions of the

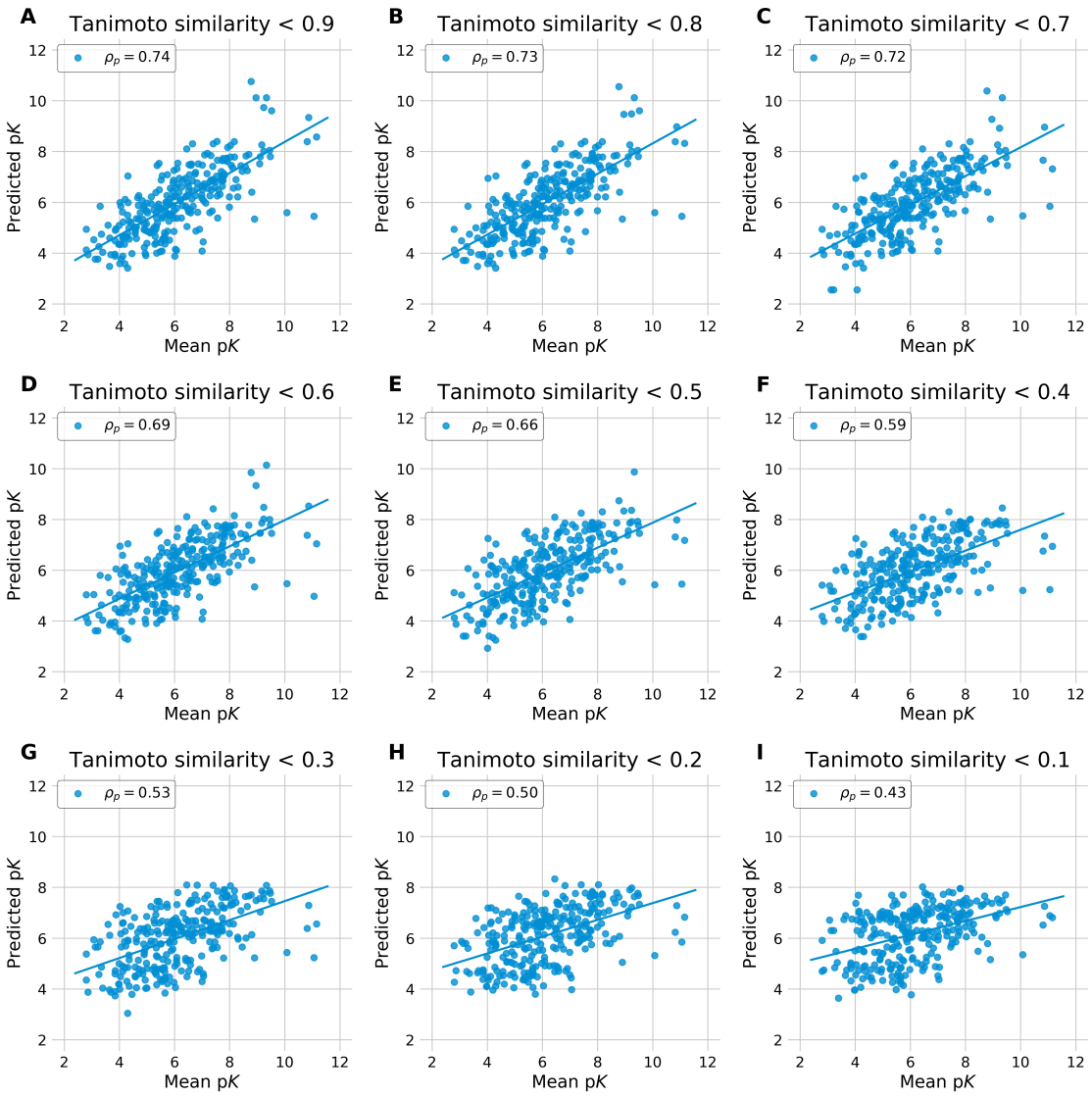


Figure 3.17: Ligand-only model predicted pK against mean experimental pK for ligands found in multiple structures in the PDBbind 2018 general set, for varying values of the Tanimoto similarity threshold above which similar ligands were excluded from the training set. For each ligand, the model was trained on the PDBbind 2018 general set excluding all complexes containing similar ligands, and tested only on that ligand. Each marker represents a single ligand. The line denotes a linear regression through the plotted points.

RF-Score v3 + RDKit model are systematically lower than those of the RF-Score v3 model, showing that the ligand-based features bias the predictions toward that of the ligand-only model.

Figure 3.20 shows the predictions of the ligand-only model (red markers) against the predictions of RF-Score v3 (black swarm plot) when tested on those ligands with the largest number of structures. Comparing the predictions of RF-Score v3 with the experimental data for these ligands shown in Figure 3.15, it is clear that RF-Score v3 is unable to predict the distribution of the experimental affinity data for each ligand. Instead, the values predicted by RF-Score v3 for a given ligand are very similar regardless of the protein target or experimental affinity, and in most cases are close to the value predicted by the ligand-only model. Similar results were observed for models using the Vina features, RF-Score features, and NNScore 2.0 features, and are included in Appendix Figure A.7.

3.3.5 | Both Structure-Based and Ligand-Based Features are Important

The relative importance of the twenty highest-ranked ligand-based features for the RF model using RDKit features trained on the PDBbind 2018 general set is shown in Fig. 3.21. The bulk properties molar refractivity (MolMR) and the logarithm of the octanol-water partition coefficient (MolLogP) are ranked highest. Molar refractivity captures the total polarizability of the molecule and log P captures its solubility; we might therefore expect these features to capture useful

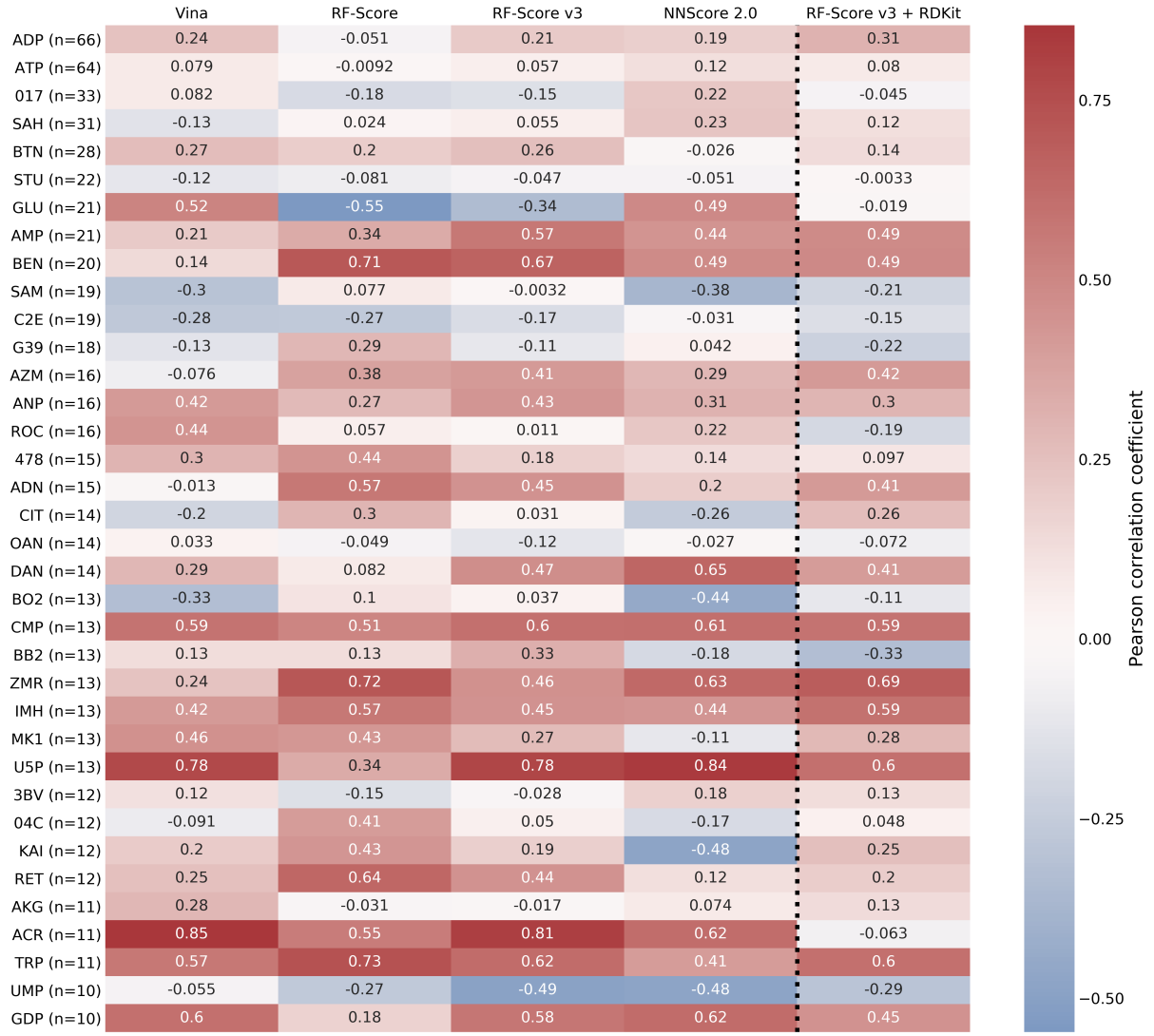


Figure 3.18: Pearson correlation coefficient for predicted against measured pK for groups of structures containing the same ligand in the PDBbind 2018 general set. Each row corresponds to a set of structures featuring a single ligand, indicated on the vertical axis. Each column corresponds to a different set of features used by the scoring function, indicated on the horizontal axis. Red cells correspond to positive correlations; blue cells to negative correlations.

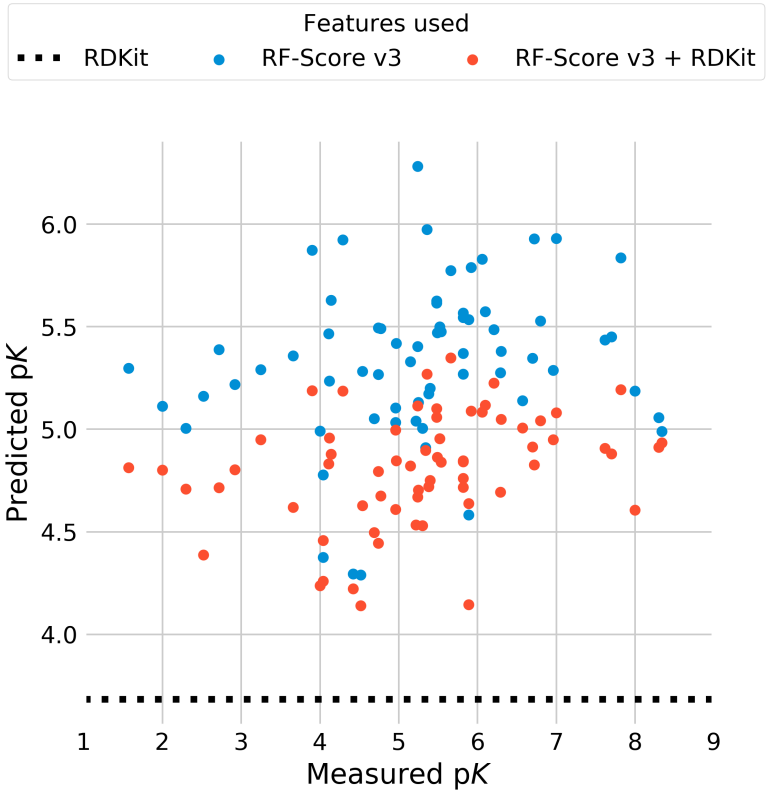


Figure 3.19: Predicted against measured binding affinity for the complexes featuring ADP in the PDBbind 2018 general set. Each marker represents one complex: blue markers correspond to RF-Score v3 predictions; red markers correspond to RF-Score v3 + RDKit predictions. Both models fail to predict the range of experimental data, and there is no correlation between the predicted and experimental values. The single value predicted by the ligand-only RDKit model is indicated by the black dotted line.

Figure 3.20: Affinity predicted by ligand-only model (red marker) and RF-Score v3 (black swarm plot) for the most common ligands in the PDBbind 2018 general set. The three-character chemical ID of each ligand is indicated on the horizontal axis. For each ligand, both models were trained on all structures in the PBDbind 2018 general set that did not contain that ligand.

information about the ability of a small molecule to bind to a charged, buried active site. Both these properties are also used to characterize drugs (Ghose et al., 1999) and log P is also used to predict bioavailability (Lipinski, 2004). It is possible that the predictive power of these features can in part be attributed to systematic bias in favour of crystallising complexes featuring high-affinity engineered compounds. However, we found that there was no trivial correlation between either of these features and the pK of a compound ($\rho_{\mathrm{p}} = 0.26$ and $\rho_{\mathrm{p}} = 0.16$, respectively, across the PDBbind 2018 general set), suggesting that their contribution to the scoring function is only in concert with other features.

Perhaps easier to explain are features capturing molecular weight and charge (ExactMolWt, MaxAbsPartialCharge, MolWt, MinAbsPartialCharge, MaxPartialCharge) as the size and charge of the molecule will impose constraints on both its ability to fit within a binding pocket, its electrostatic complementarity, and the number of interactions it has the ability to form. Similarly, topological polar surface area (TPSA) is an approximation of a molecule's polar surface area computed using its 2D chemical graph, and may provide information about its hydrophobicity and ability to fit within a binding pocket. Van der Waals surface area contributions, captured by the PEOE_VSA descriptors, likewise characterise the molecular surface and hence potential interactions. More complicated are the 2D descriptors capturing molecular connectivity (Chi) and graph complexity (BertzCT), whose contribution to the model might also be through capturing the shape and surface area of the molecule or some aspect of conformational entropy.

We also found that when ligand-based features were combined with structure-based features in our other models, both ligand-based and structure-based fea-

tures were ranked highly, and that the same ligand-based features were consistently found to be important regardless of which structure-based features were used. This is illustrated in Figure 3.22. This suggests that the ligand-based features are consistently capturing useful information that is not present in the structure-based features, beyond the number of rotatable bonds in the ligand. These results suggest that when using ML to predict protein-ligand binding affinity, it is better to use a richer description of the ligand in the model.

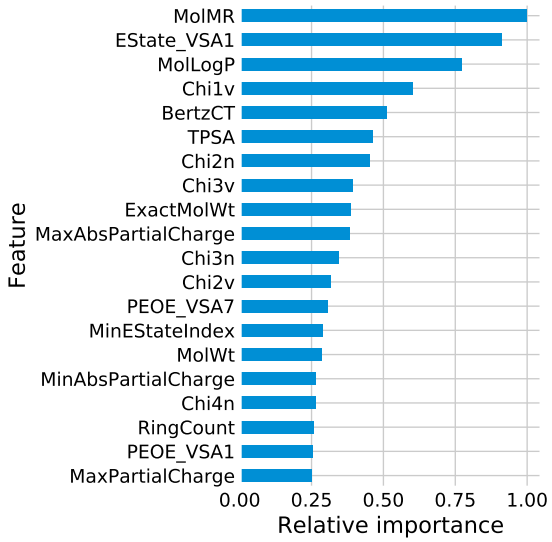


Figure 3.21: Relative importance of features in the RF model using RDKit features trained on the PDBbind 2018 general set. A description of each feature is provided in the RDKit documentation (https://www.rdkit.org/docs/GettingStartedInPython.html, last accessed 17/05/2019).

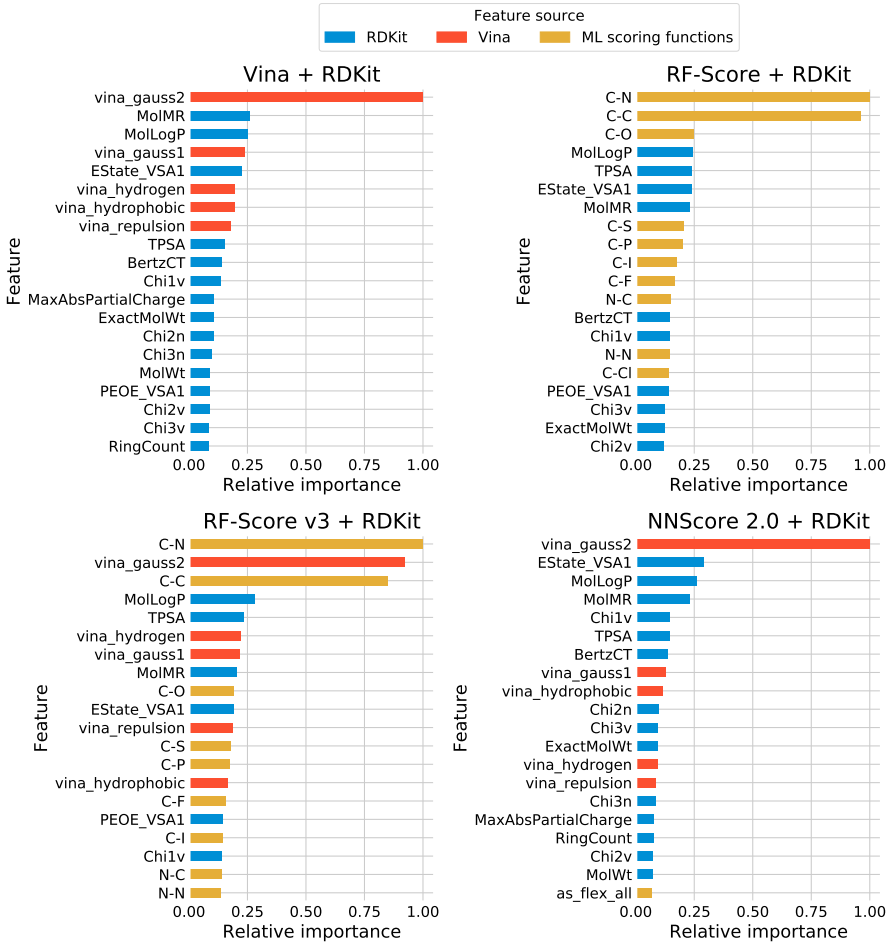


Figure 3.22: Relative importance of ligand-based and structure-based features in hybrid RF scoring functions trained on the PDBbind 2018 general set. RDKit features are shown in blue. Vina features are shown in red. RF-Score and NNScore 2.0 features excluding the Vina features are shown in yellow.

3.3.6 | RF Scoring Functions Tend to Fail on Unseen Protein Targets

The previous results indicate that the structure-based scoring functions fail to differentiate between different structures featuring the same ligand. To complete the picture, we also investigate whether these methods can distinguish between, and accurately score, different structures of the same, or similar, protein(s) binding different ligands. We first grouped the structures in the PDBbind 2018 general set by protein UniProt ID. Each group was held out in turn as a test set, with the models trained on all remaining structures in the PDBbind 2018 general set. As we did not exclude sequence-similar structures from the training set, this simulates the scenario of a novel protein target for which some data on similar proteins are available.

Figure 3.23 shows the Pearson correlation coefficient between predicted and experimental pK achieved by RF models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features, for groups of structures of the same protein in the PDBbind 2018 general set. Results are shown for all proteins with at least 50 structures available. There is considerable variation in performance across the set of proteins; all three models achieve a Pearson correlation coefficient greater than 0.9 when tested on the set of structures for Q9Y233 (cAMP and cAMP-inhibited cGMP 3',5'-cyclic phosphodiesterase 10A), while for many proteins none of the models achieve a meaningful correlation. There is a general trend toward all three models having comparable performance on a protein-by-protein basis: for a given protein, either all three of the models achieved a meaningful correlation, or no model achieved a meaningful correlation.

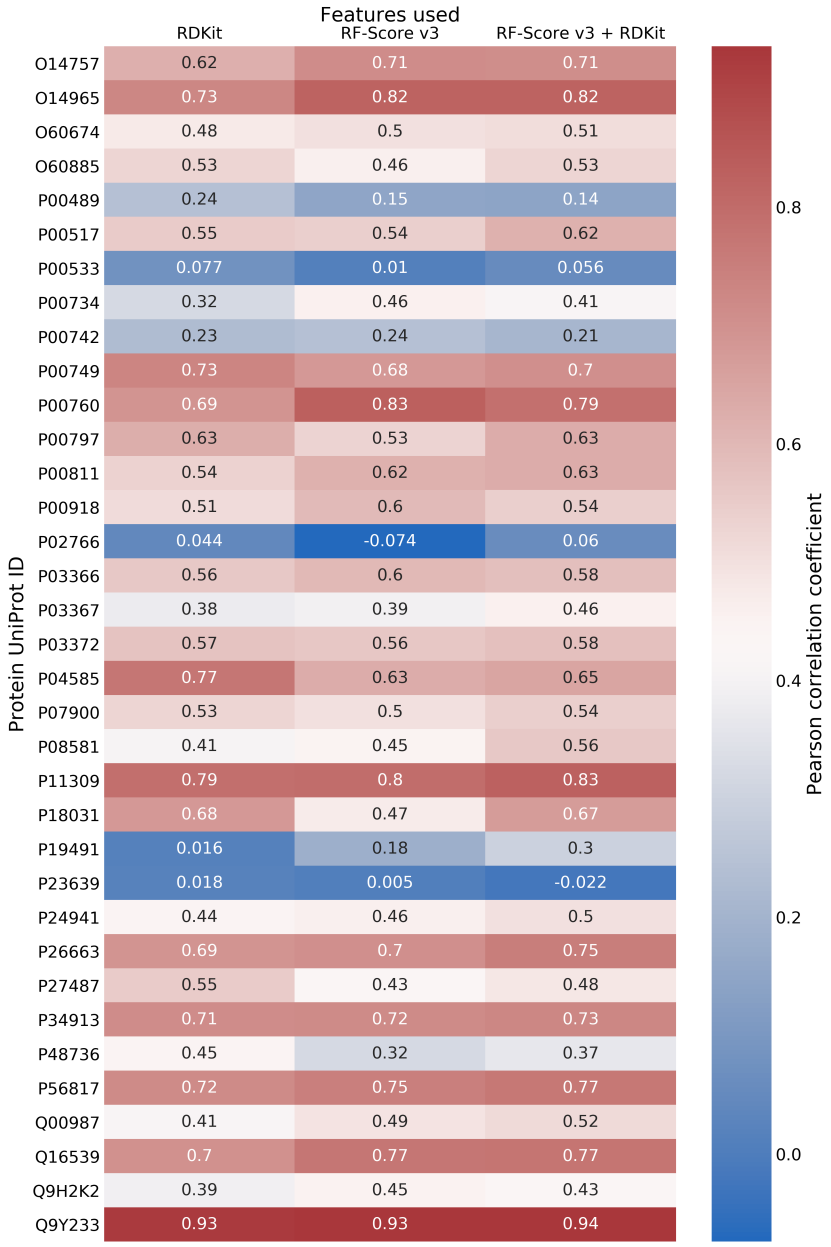


Figure 3.23: Pearson correlation coefficient for predicted against measured pK for groups of structures of the same protein in the PDBbind 2018 general set. Each row corresponds to a set of structures on a single protein, the UniProt ID of which is indicated on the vertical axis. The full name of the protein corresponding to each UniProt ID is shown in Table 3.1. Each column corresponds to a different set of features used by the scoring function, indicated on the horizontal axis. Red cells correspond to positive correlations; blue cells to negative correlations.

We next clustered the structures at 30% protein sequence identity and repeated the experiment, holding each cluster out in turn as a test set, and training the models on the remaining structures. This results in a smaller number of clusters, each of which contains a more diverse set of related structures, and the largest of which contained over 200 structures. This simulates the scenario of a completely novel protein target for which there are no similar structures available for training.

Figure 3.24 shows the Pearson correlation coefficient between predicted and experimental pK achieved by RF models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features, on clusters of structures in the PDBbind 2018 general set clustered at 30% sequence identity. Results are shown for all clusters containing at least 30 structures. In each case, the model was trained on the PDBbind 2018 general set excluding the structures in the test cluster. As in the case of grouping the structures by identical UniProt ID, performance of the three models varies greatly across the different cluster and, in general, with the performance of all three models being comparable on a given cluster. This is in contrast with the variation in performance of different scoring functions on clusters of structures featuring the same ligand (Figure 3.18) suggesting that the ligand and its representation in the model play an important role in scoring function performance. These results echo those of Kramer and Gedeck (2010), suggesting that despite the availability of much larger sets of training data and the use of more complex models exhibiting greater benchmark performance, generalising to a novel protein target remains challenging.

The results shown in Figures 3.18 and 3.20 indicate that these RF-based scoring functions are unable to reliably distinguish between different structures fea-

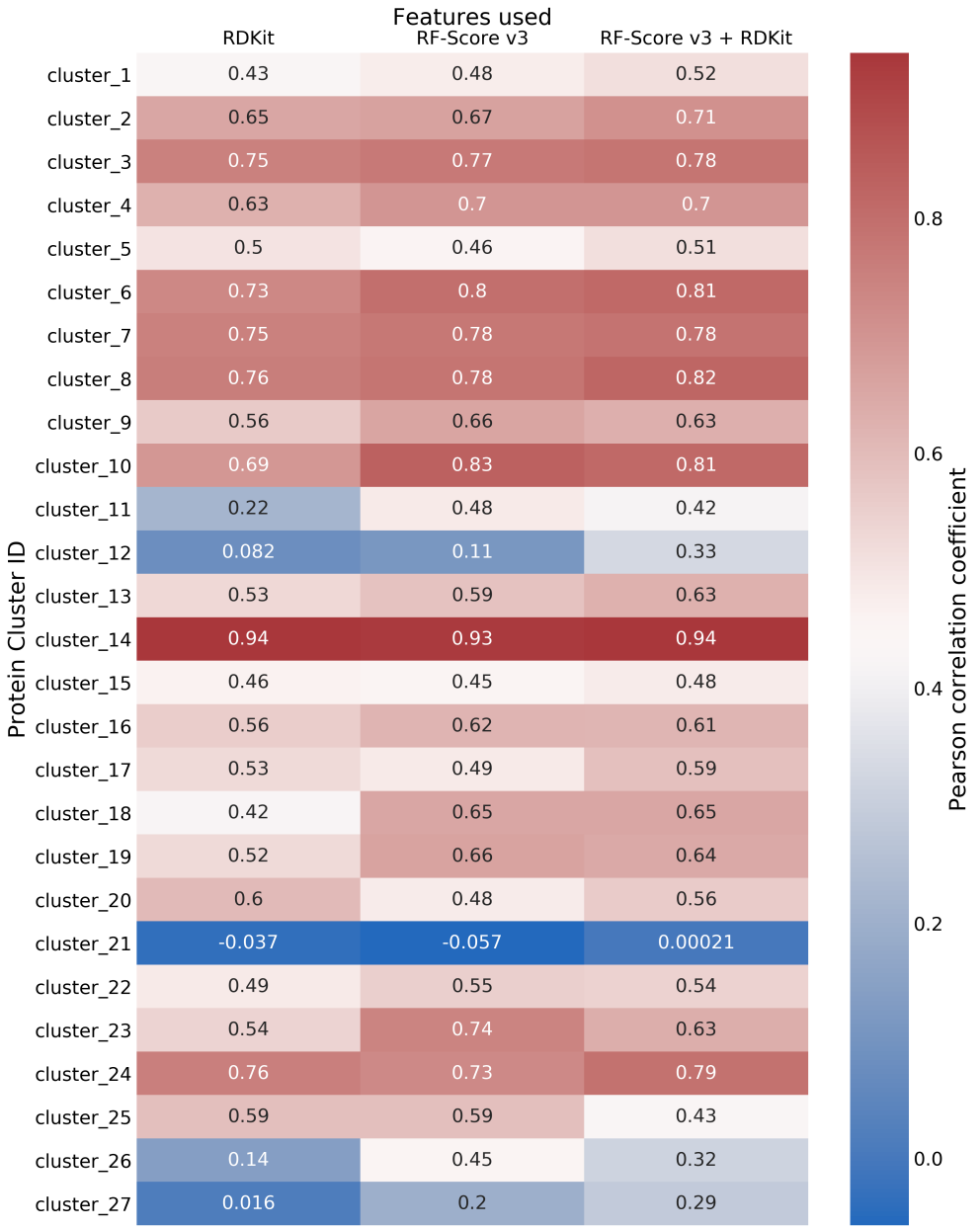


Figure 3.24: Pearson correlation coefficient for predicted against measured pK for clusters of structures at 30% sequence identity in the PDBbind 2018 general set. Each row corresponds to a single cluster. The cluster numberings correspond to the representative proteins listed in Table 3.2. Each column corresponds to a different set of features used by the scoring function, indicated on the horizontal axis. Red cells correspond to positive correlations; blue cells to negative correlations.

turing the same ligand, regardless of the differences in experimentally-determined binding affinity for those complexes. Figure 3.16 shows that the predictions of the ligand-only model are strongly correlated with the mean of the experimentally-determined pK values associated with a ligand. The binding data for the core sets spans the range of pK values present in the PDBbind refined set, from a pK of 2 to a pK of 12; approximately ten orders of magnitude. The range of values associated with any single ligand is typically considerably smaller (Figure 3.15, with the full range of pK values only covered when data for many ligands are aggregated. A model that is able to accurately estimate the mean pK associated with a single ligand, would therefore be expected to achieve a moderate or even strong correlation across the full range of experimental data simply by estimating the mean of many different subsets of that data. This could explain why all of these methods exhibit strong performance on a diverse benchmark such as the PDBbind core sets despite poor performance when tested on sets of different structures featuring the same ligand.

3.4 | Conclusions

In this Chapter we found that the inclusion of readily-computed ligand-based features in machine learning scoring functions consistently improves predictive power across a large, diverse benchmark and a range of training scenarios.

Varying the composition of the training set chronologically, by restricting to data available only up until a particular year, had little effect on affinity predictions. This suggests an element of learning saturation for the targets tested with the data currently available. We showed that, in contrast, excluding proteins from the training set that are sequence-similar to those in the test set has a deleterious effect on affinity predictions and that even excluding only those proteins with identical sequence to those in the test set leads to significantly reduced scoring function performance. We also showed that even when ligands with high Tanimoto similarity to those in the test set were excluded from the training set, the predictive power of the scoring functions was still increased by including ligand-based features.

Given the power of the ligand-based features, we investigated their predictive ability for ligands that bind to multiple targets and found that the predicted binding affinity of a model using only ligand-based features was strongly correlated with the mean of the experimental protein-ligand binding affinity of a ligand for its binding partners. This correlation remained strong when ligands with a Tanimoto similarity of greater than 0.9 to the test ligand were excluded from the training data, and gradually weakened when progressively less similar ligands were also excluded, suggesting that while the model's predictions are not reliant on overfitting to previously-seen highly-similar ligands, it does

not extrapolate well to completely novel ligands.

We also investigated how both structure-based and ligand-based methods perform when tested on held-out clusters of previously-unseen proteins, and found that while performance varied greatly from protein to protein, structure-based and ligand-based methods achieved comparable performance on any given protein. These results suggest that the representation of the ligand in a scoring function can play a more important role in its predictive performance and that, despite the ever-growing quantity and diversity of available training data, generalising to a previously-unseen protein target remains challenging.

Finally, we analysed the relative importance of the features of each scoring function. We found that when structure-based and ligand-based features are combined, both structure-based and ligand-based features were ranked highly, and that the same ligand-based features are ranked highly regardless of which structure-based features they were combined with.

Our results suggest that even under stringent validation, the addition of a diverse, quickly-computed set of ligand-based features to a scoring function yields improved predictions of binding affinity.

4

Using Docked Poses to Predict Binding Affinity

In Chapter 2 we explored the use of different features and machine learning algorithms for predicting protein-ligand binding affinity and found that Random Forest (RF) models combining structure-based and ligand-based features achieved the strongest performance under cross-validation and on a held-out test set.

In Chapter 3 we investigated how training set composition affected the performance of our models, and found that protein and ligand similarity between the training and test sets had a strong influence on model performance. We also examined the predictions of a purely ligand-based model and found that it was predictive of the mean affinity of a ligand for its protein targets for a diverse set of protein-ligand complexes.

Common to both Chapter 2 and Chapter 3 was the use of crystal structures of the protein-ligand complex in both the training and test sets. In this Chapter we explored how our models performed when trained and validated using docked poses instead of the experimentally-determined binding mode of the lig-

and. I will describe how I redocked ligands to their cognate proteins from the PDBbind database and assessed the quality of the generated binding poses, and I will show that pose prediction errors are common. Using the combined core set, I will describe how we examined the impact of pose generation error on scoring function performance, and show that using a combination of structure-based and ligand-based features results in a smaller drop in performance than when using structure-based features alone. I will also describe experiments to investigate how the accuracy of the predicted binding mode affects scoring function performance, and will show that using less-accurate poses results in worse performance. Finally, I describe how we constructed a new data set for binding affinity prediction using ligand binding affinity data from the ChEMBL database. I will describe how we trained and validated our models on this new data set and show that models trained on PDBbind data generalised poorly to external data. I will also show that when training data are available for some ligands of a target, a ligand-based scoring function out-performs a structure-based one, but when no data are available for a target, both ligand-based and structure-based methods fail to generalise to a novel target.

4.1 | Introduction

Within the field of scoring function development, it is a standard choice to use crystal structures of bound protein-ligand complexes when developing and evaluating methods for binding affinity, as this allows the isolation of the task of affinity prediction from noise that might be introduced through errors in ligand pose prediction, or as a result of the rigid receptor hypothesis often used

in docking (Cheng et al., 2009; Li et al., 2014b; Su et al., 2018). However, in a real-world drug discovery scenario, it is much less likely that an experimentally-determined binding mode is are available for the candidate ligands being screened. Instead, protein-ligand docking is commonly used to generate putative binding modes for each ligand. Thus, for a scoring function to be useful in a practical setting, it must be able to accurately predict the binding affinity of a protein-ligand complex using only a docked pose of the ligand.

While the use of crystal structures of bound complexes for scoring function development and evaluation is common in the literature, the use of docked poses and systematic evaluation of the effect this has on scoring function performance is less common, and the effect of using docked poses in place of crystal poses is rarely evaluated. For example, Durrant and McCammon (2011b) used both crystal and docked poses in the development of evaluation of NNScore 2.0, noting that the optimal choice of docking protocol is highly system-dependent, but did not examine the difference in performance between using crystal or docked poses. Similarly, Zilian and Sotriffer (2013) trained $\text{SFCScore}_{\text{RF}}$ using crystal poses from the PDBbind database (Wang et al., 2004; Liu et al., 2017), and validated on both crystal poses from the PDBbind database and docked poses of the CSAR–NRC HiQ benchmark (Dunbar et al., 2011) and CSAR 2012 benchmark (Damm-Ganamet et al., 2013). More recently, Jimenez et al. (2018) validated K_{DEEP} using both crystal poses and docked poses but, as in earlier studies, used docked poses when crystal structures were unavailable, so the impact of the use of docked poses in place of crystal structures is unknown.

To address this question, Li et al. (2016) investigated how the use of docked poses in place of crystal poses affected the performance of the AutoDock Vina

scoring function, RF-Score, and RF-Score v3. The authors reported that, contrary to what might be expected, that using docked poses in place of crystal poses only had a small effect on the accuracy of binding affinity predictions; for example, the Spearman rank correlation coefficient attained by RF-Score v3 on the PDBbind 2013 core set dropped from 0.662 to 0.633. The authors further reported that by this drop in performance was reduced by training RF-Score v3 on docked poses instead of crystal poses, resulting in a Spearman rank correlation coefficient of 0.643 on the PDBbind 2013 core set, suggesting that the errors introduced by using docked poses can be partially compensated for by also using docked poses when training the scoring function. However, it remains unclear how the use of docked poses might affect other scoring functions, or how well this result generalises to data sets outside of the PDBbind database.

This Chapter explores how the use of docked poses in place of crystal poses affects the performance of the models studied in Chapters 2 and 3. This is especially relevant when working with ligand-based features that are independent of the pose of the ligand, as these will be unaffected by errors in the binding pose introduced by docking. It is therefore interesting to investigate how the addition of ligand-based features to a structure-based scoring function affects performance when docked poses are used, and whether the addition of pose-independent ligand-based features to a structure-based scoring function can help to reduce errors in affinity prediction introduced by the use of docked poses.

4.2 | Materials and Methods

The features and models used for binding affinity prediction in this Chapter were discussed in Chapter 2. Here we describe the methods used for generating and assessing docked poses, and the construction of a new data set for binding affinity prediction.

4.2.1 | Data Preparation

All docking calculations were performed using Smina (Koes et al., 2013), a fork of AutoDock Vina (Trott and Olson, 2010) featuring an enhanced command-line interface and performance optimisations for minimizing large sets of ligands. Smina also features support for custom scoring functions, but for this work we used Smina with the default AutoDock Vina scoring function.

Protein and ligand structures were prepared for docking using the following protocol. For each ligand, a conformer was generated using the ETKDG method (Riniker and Landrum, 2015) implemented in RDKit (Landrum, n.d.a). Protein PDB files and ligand SDF files were prepared for docking by generating PDBQT files using OpenBabel (O'Boyle et al., 2011). This step assigns Gasteiger-Marsili partial charges (Gasteiger and Marsili, 1980) and AutoDock atom types to the atoms in the structure, both of which are required to perform docking calculations. Smina is capable of converting PDB and SDF files to PDBQT format internally using OpenBabel; however, in the interest of reproducibility, we chose to generate the PDBQT files manually to ensure the files remained available for future use.

4.2.2 | Redocking PDBbind

We first generated docked poses for the PDBbind 2018 refined set. We restricted this experiment to the refined set as the structures are of overall higher quality (2.5Å or better and no missing side chains) than those in the general set (which lack quality criteria). This also allowed us to invest computational time in performing a more exhaustive search of conformational space for the highest-quality structures, rather than reducing exhaustiveness to expedite the docking of the much larger general set.

The structure files for the protein and ligand of each complex in the PDBbind 2018 refined set were prepared as described in Section 4.2.1. We intentionally generated a new conformer for the ligand to ensure the docking results were not biased toward the experimentally-determined conformer provided by PDBbind. The prepared protein and ligand were then docked using Smina, treating the receptor as fully rigid. The SDF file of the ligand provide by PDBbind was used to define the centre of the search space via the *autobox_ligand* argument of Smina. The search parameters used with Smina are shown in Table 4.1.

Parameter	**Value**
autobox_add	8
exhaustiveness	20
num_modes	20
random_seed	42

Table 4.1: Smina docking parameters used to dock the ligands to their cognate protein partners.

Of the 4,463 complexes in the PDBbind 2018 refined set, RDKit failed to parse the ligand for 210 complexes. These complexes were not re-docked using Smina,

as a new conformer could not be generated by RDKit. In total, 4,253 complexes from the PDBbind 2018 refined set were re-docked using Smina. It is important to note that in practical applications of docking, a ligand are unlikely to be re-docked into the crystal structure of the conformation of the protein when that ligand is bound. Instead, a ligand will most likely be docked into either a crystal structure of of the protein when a different ligand is bound, or if no such bound structures are available, an unbound (apo) structure of the protein. This poses an additional challenge for docking as the conformation of the protein may change depending on which ligand, if any, is bound. If the flexibility of the protein is not adequately modelled during docking, reduced shape complementarity between the ligand and the active site may make it difficult to identify the native binding mode. Further, if no experimentally-determined structures are available, the ligand must instead be docked into a homology model of the protein structure. In addition to the challenge of docking into an apo structure, the homology model itself may be inaccurate, resulting in additional difficulty in identifying the native binding mode. For these reasons, the poses generated by re-docking the PDBbind database are likely an optimistic estimate of docking performance, even when the initial conformation of the ligand is randomly generated.

4.2.3 | ChEMBL Data Set

To further validate models trained on docked poses, we constructed an additional data set of ligands with binding affinity data selected from the ChEMBL database (Gaulton et al., 2017). We chose to use eight of the 102 protein targets

from the DUD-E virtual screening database (Mysinger et al., 2012): serine/threonine-protein kinase AKT; β-lactamase; cytochrome P450 3A4; C-X-C chemokine receptor type 4; glucocorticoid receptor; HIV-1 protease; HIV-1 reverse transcriptase; and kinesin-like protein 1. These eight proteins comprise a subset of the DUD-E database known as the 'diverse' set, which contains representatives of the different protein families in the DUD-E database. We used targets from DUD-E as this database is commonly used in benchmarking docking and virtual screening methods, allowing us to build a data set complementary to that commonly used in the literature.

For each target, we queried ChEMBL version 25 for ligands for that target for which a measurement of K_i was available. We restricted the data set to measurements of K_i to minimise noise due to heterogeneous data. Two targets, β-lactamase and C-X-C chemokine receptor type 4, did not have any ligands with recorded values of K_i, so were removed from the target set. For the remaining six targets, we downloaded the SMILES and affinity data for the available ligands. We used the pChEMBL value, which is the negative base-10 logarithm of the binding constant reported by ChEMBL, equivalent to the pK value[1]. Duplicate pChEMBL were removed; while for ligands with multiple different pChEMBL values, we used the mean of the pChEMBL value for that ligand. The names of the six targets for which K_i data were available in ChEMBL, together with the PDB code of the structure provided by DUD-E and the number of ligands in our data set, are shown in Table 4.2.

The distributions of the pChEMBL values for the six targets are shown in

[1]https://chembl.gitbook.io/chembl-interface-documentation/frequently-asked-questions/chembl-data-questions#what-is-pchembl; last accessed 04/01/2020

Protein name	Abbreviation	PDB code	# ligands
Serine/threonine-protein kinase AKT	AKT1	3CGW	216
Cytochrome P450 3A4	CP3A4	3NXU	137
Glucocorticoid receptor	GCR	3BQD	863
HIV-1 protease	HIVPR	1XL2	2023
HIV-1 reverse transcriptase	HIVRT	3LAN	69
Kinesin-like protein 1	KIF11	3CJO	143

Table 4.2: Summary of the ChEMBL data set. The abbreviation and PDB structure for each target correspond to those found in DUD-E.

Figure 4.1. The data for HIVRT and CP3A4 span only four orders of magnitude, from a pChEMBL of 4.0 to to a pChEMBL of 8.0, with the exception of two ligands of CP3A4 with a pChEMBL greater than 10.0. The data for GCR span a slightly larger range, with pChEMBL values ranging from 4.7 to 10.0, while the data for AKT1, HIVPT, and KIF11 span at least six orders of magnitude. These data all lie within the range of values represented in the PDBbind 2018 refined set, so a RF model trained on PDBbind data could be expected to interpolate well within this range of values.

The PDB file provided by DUD-E and the ligand SMILES strings downloaded from ChEMBL were prepared for docking by generating PDBQT files as described in Section 4.2.1. For each protein, we used the crystallographic ligand from the corresponding PDB structure provided by DUD-E was used to define the centre of the search space. Finally, for each prepared protein, its prepared ligands were docked using Smina with the parameters shown in Table 4.1.

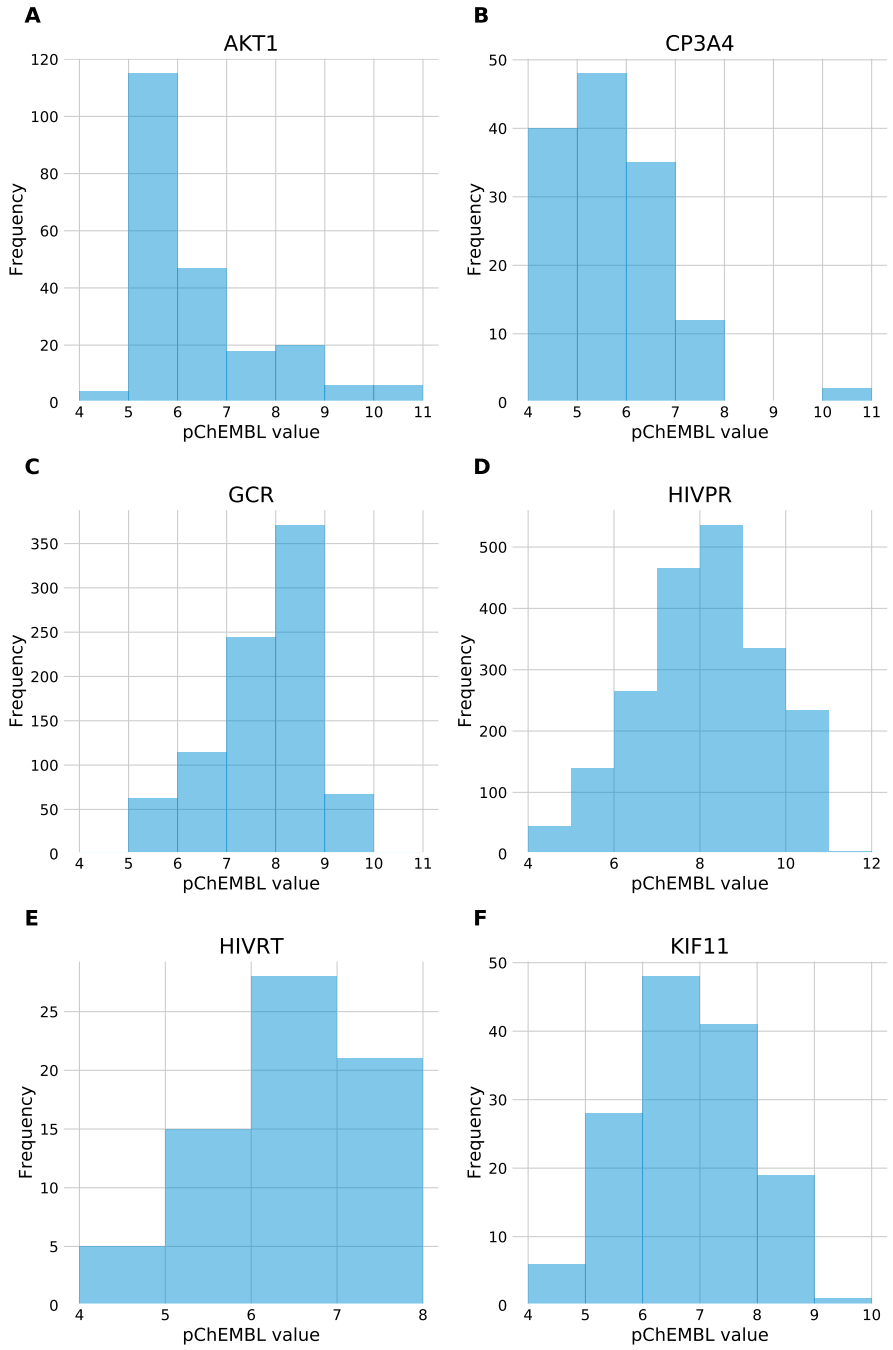


Figure 4.1: Distribution of pChEMBL values for the ligands of six protein targets. (A) serine-threonine-protein kinase AKT, AKT1; (B) Cytochrome P450 3A4, CP3A4; (C) glucocorticoid receptor, GCR; (D) HIV-1 protease, HIVPR; (E) HIV-1 reverse transcriptase, HIV1RT; and (F) kinesin-like protein 1, KIF11. More details on each target are included in Table 4.2.

4.2.4 | Docking Evaluation

The quality of a docked pose was assessed by computing the root-mean-squared deviation (RMSD) of the positions of the atoms of the ligand's docked pose to that of the crystal pose. For a molecule with N heavy atoms, let d_i denote the distance between the position of atom i in two conformers of the molecule, then the RMSD of the atomic positions is defined as:

$$\text{RMSD} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} d_i}. \tag{4.1}$$

Because the definition of RMSD in Equation 4.1 uses the distance between indexed atoms, the computed RMSD of a conformer of a symmetric molecule will be dependent on the indices of the symmetric atoms. This could result in two or more physically identical conformers of a molecule being assigned artificially high RMSD values if the indices of the symmetric atoms are not identical between the two conformers. To ensure we correctly account for symmetry when computing the RMSD between two conformers, we identified symmetrically equivalent permutations of the atomic indices of a molecule by performing a substructure match of the molecule against itself using RDKit. We then applied these permutations to the indices of the atoms and re-computed the RMSD of each conformer using Equation 4.1. The lowest computed RMSD value was then taken as the RMSD of that conformer.

4.2.5 | Training and Validation

In Chapter 3 we found that RF models using either RF-Score v3 features or NNScore 2.0 features achieved comparable performance, and out-performed RF models using AutoDock Vina features or RF-Score features. We therefore chose to focus on using RF-Score v3 features as our structure-based feature set as it is the smaller and simpler of the two top-performing feature sets.

We built RF models using three feature sets described in Chapter 2: (i) RDKit features, (ii) RF-Score v3 features, and (iii) RF-Score v3 + RDKit features. When training models using features computed from docked poses, for each complex we used the pose ranked highest by the native AutoDock Vina scoring function as implemented in Smina. We did not restrict the training set to ligands with poses with RMSD below 2Åas this would have substantially reduced the size of the training set. For the first validation, using the combined core set described in Section 3.2.1, models were tested on the pose ranked highest by Smina, simulating the real-world scenario where a crystallographic pose is not available.

4.3 | Results and Discussion

4.3.1 | Pose-Prediction Errors are Common

We first assessed the quality of the docked poses generated by Smina. Figure 4.2 shows the distribution of the RMSD of the best docked pose for the PDBbind 2018 refined set. Of the 4,253 complexes re-docked by Smina, only 1,357 had at least one docked pose with RMSD below 2Å, while for 1,003 complexes every

docked pose had RMSD greater than 4Å. Further, the AutoDock Vina scoring function used by Smina to rank the poses often failed to identify a good pose even when one was sampled: of the 1,357 complexes with at least one pose with RMSD below 2Å, the top-ranked pose returned for Smina has RMSD greater than 2Å for 691 complexes, with 371 of these having an RMSD of greater than 4Å. This is not surprising, as it has been reported that docking scoring functions are not always a reliable way of selecting the best pose from a set of putative binding poses (Ramírez and Caballero, 2018).

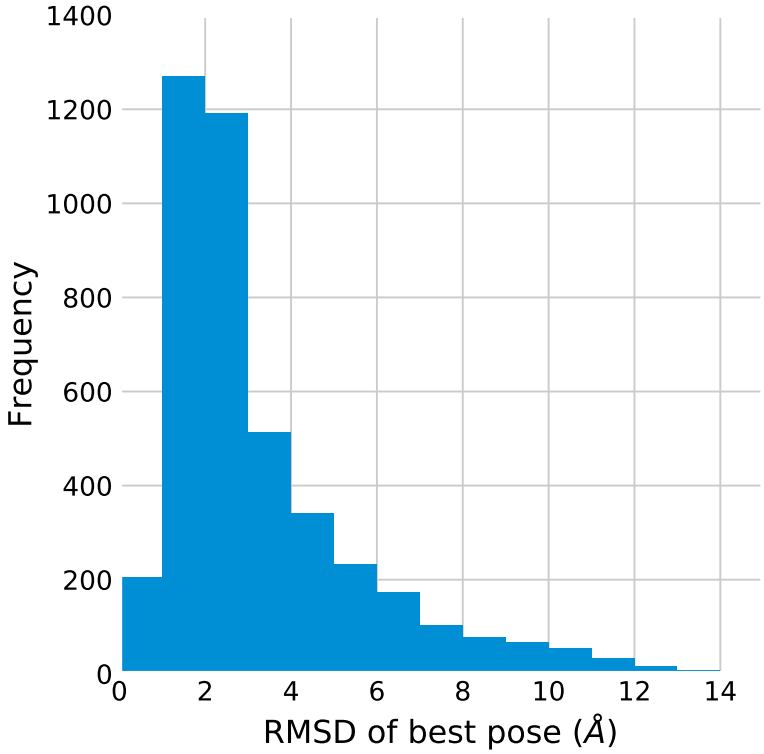


Figure 4.2: Distribution of the RMSD of the best docked pose generated by Smina for 4,253 structures from the PDBbind 2018 refined set.

One possible source of docking error is the use of large, flexible molecules, whose conformational space is large and difficult to sample effectively. How-

ever, this does not appear to be the case when docking the PDBbind 2018 refined set using Smina. Figure 4.3 shows the RMSD of the best pose generated by Smina against the number of heavy atoms (Figure 4.3 A) and the number of rotatable bonds in the ligand (Figure 4.3 B). There is little correlation between the RMSD of the best pose and the size or flexibility of the ligand, indicating that while these factors may contribute to difficulties in docking, they do not fully explain the cases where poor poses were generated.

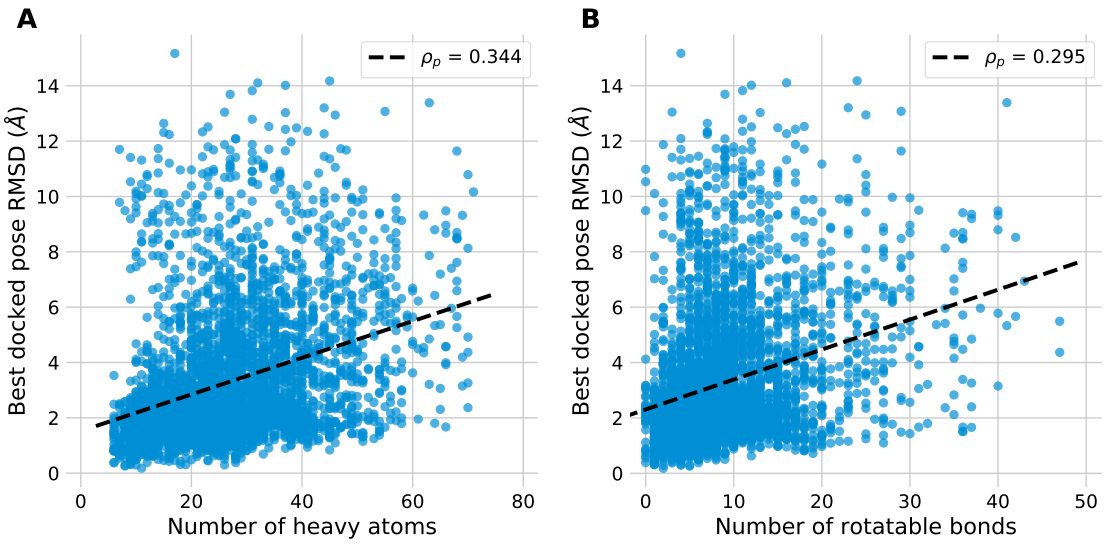


Figure 4.3: RMSD of the docked pose ranked highest by the AutoDock Vina scoring function. A: RMSD against the number of heavy atoms in the ligand. B: RMSD against the number of rotatable bonds in the ligand. The broken black line indicates a linear regression fit to the points.

These results indicate that, while in many cases a simple docking protocol can yield binding poses close to the native pose found in a crystal structure, success in reproducing the experimentally-determined binding pose is far from guaranteed. Thus, in any study attempting to use docked poses to predict binding affinity, errors in the binding pose should be anticipated. Further, even when

the native binding pose of the ligand is sampled by the docking method, it may not be selected by the docking scoring function as the top-scoring pose. However, as docking is used when the native binding pose is not known, in any practical application we need a means of choosing the pose or poses used for binding affinity prediction. Though far from perfect (Ramírez and Caballero, 2018), docking scoring functions are much better at selecting a good pose than they are at ranking binding affinity (Cheng et al., 2009; Li et al., 2014b; Su et al., 2018). Thus, while not always ideal, using the top-scoring pose returned by Smina is a straightforward way of choosing a single pose.

4.3.2 | Including Ligand-Based Features Reduces the Performance Gap Between Docked and Crystal Poses

Figure 4.4 shows the Pearson correlation coefficient achieved by RF models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features on the combined core set when trained on the PDBbind 2018 refined set (excluding the combined core set), for varying levels of protein sequence similarity (Figure 4.4A) and ligand Tanimoto similarity 4.4B) between the training and test sets respectively.

The performance of the model using RF-Score v3 features is worse when docked poses are used: when no data are excluded from the training set, the Pearson correlation coefficient of 0.767 when using crystal poses, and 0.713 when using docked poses. There is a much smaller drop in performance when using RF-Score v3 + RDKit features: when no data are excluded from the training set,

the Pearson correlation coefficient falls from 0.790 when using crystal poses to 0.774 when using docked poses. The performance when using RF-Score v3 + RDKit features and docked poses is comparable to that of using RF-Score v3 features and crystal poses. These trends remains present when sequence-similar proteins are excluded from the training set, suggesting that including the RDKit features in the model helps to mitigate the loss in performance resulting from the use of docked poses.

The gap in performance between models using RF-Score v3 features and models using RF-Score v3 + RDKit features is greater when using docked poses than when using crystal poses when the Tanimoto similarity threshold between the training and test sets is 0.6 or greater. When this threshold is reduced to 0.5 or lower, the gap in performance is greatly reduced. This corresponds to the drop in performance of the models augmented with RDKit features observed in Chapter 3 Figure 3.10 when ligands with Tanimoto similarity of 0.5 or greater to those in the test set were excluded from the training set. These results suggest that, provided sufficient training data are available, the inclusion of ligand-based features in the model helps to reduce the effect of docking pose error on scoring function performance.

4.3.3 | Docking Quality Affects Binding Affinity Prediction Accuracy

We demonstrated in Section 4.3.1 that errors in pose prediction are common when docking using Smina, and commented on the fact that the pose ranked

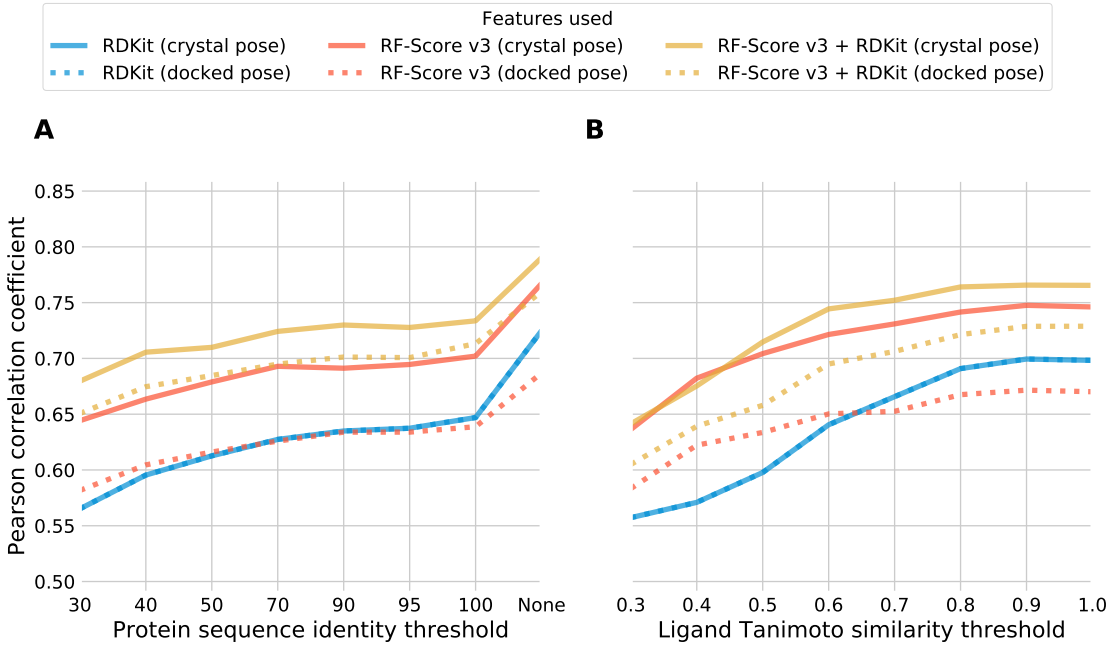


Figure 4.4: Pearson correlation coefficient between predicted and experimental pK values on the combined core set. Solid lines show performance when trained and tested using crystallographic binding poses; dotted lined show performance when trained and tested using docked poses.

highest by the native AutoDock Vina scoring function as implemented in Smina may not be the most accurate of the poses that were generated. We next investigated how best to score a complex when multiple docked poses are available, and how the quality of the docked poses affects scoring function performance.

We first investigated three strategies for assigning a score. These were: (i) scoring the pose ranked highest by Smina; (ii) scoring all poses and taking the highest score; and (iii) scoring all poses and taking the mean score. Using the PDBbind 2018 refined set, excluding the combined core set, we randomly split the data into a training set and a validation set in an 80:20 ratio. This resulted in a new training set of 3,133 complexes and a validation set of 783 complexes.

We then trained two RF models, one using RF-Score v3 features, and one using RF-Score v3 + RDKit features, on this new training set, using the pose ranked highest by Smina for each complex. Then, for both models, we predicted the pK for the complexes in the validation set using all 20 docked poses for each complex. We then took for each complex: (i) the predicted pK for the pose ranked highest by Smina; (ii) the highest predicted pK; and (iii) the mean of the predicted pK values. For each of these strategies, and for both models, we then computed the Pearson correlation coefficient between the predicted and experimental pK values. These are shown in Table 4.3.

Features	**Top-ranked pose**	**Highest score**	**Mean score**
RF-Score v3	0.681	0.699	0.665
RF-Score v3 + RDKit	0.757	0.764	0.745

Table 4.3: Effect of scoring strategy on affinity prediction accuracy. The Pearson correlation coefficient between predicted and experimental pK on a validation set of 783 complexes is shown for two RF models (one using RF-Score v3 features and one using RF-Score v3 + RDKit features) and three scoring strategies ('top-ranked pose', 'highest score', and 'mean score').

The results in Table 4.3 suggest that there is only a small difference between using the pose ranked highest by Smina and scoring all poses then taking the highest score. The RF using RF-Score v3 features achieved Pearson correlation coefficients of 0.681 using the 'top pose' strategy, 0.699 using the 'highest score' strategy, and 0.665 using the 'mean score' strategy, respectively, suggesting that there is a minor advantage to using the 'highest score' strategy, and that simply using the pose ranked highest by Smina is better than averaging the scores assigned to each pose. The difference in performance between the three strategies is smaller for the RF using RF-Score v3 + RDKit features, which is expected as

the ligand-based RDKit features are pose-independent.

We next investigated the effect of pose quality on model performance. To do this, we identified all complexes in the PDBbind 2018 refined set, excluding the combined core set, for which at least one 'good' pose (i.e. less than 2Å RMSD) and at least one 'poor' (i.e. greater than 4Å RMSD) pose were generated. This resulted in a set of 1,284 protein-ligand complexes. Because this set is considerably smaller than the set used to investigate scoring strategies, we chose to use a cross-validation approach to ensure the effective size of the test set was not too small. The 1,248 complexes were split randomly into five folds. Each fold was held out in turn as a test set, with the remaining four folds combined to form the training set. We then trained and tested two RF models: one using RF-Score v3 features, and one using RF-Score v3 + RDKit features. For training, because a single pose was needed for each complex, we simply used the pose ranked highest by Smina. For each test fold, we scored all 20 poses for each complex and separated the poses into 'good' (<2Å RMSD) and 'poor' (>4Å RMSD). For each complex, we then took the highest score across the 'good' poses as the score for that complex using 'good' poses, and the highest score across the 'poor' poses as the score for that complex using 'poor' poses. We adopted the 'highest score' strategy as this performed best when we investigated scoring strategy. We computed the Pearson correlation coefficient between predicted and experimental pK values across each fold for the 'good' poses, and separately for the 'poor' poses. The cross-validation Pearson correlation coefficient when using 'good' poses was then taken to be the mean of the Pearson correlation coefficient computed across the five folds when using 'good' poses, and similarly for the 'poor' poses. The results are shown in Table 4.4.

Features	Good (RMSD <2Å) poses	Bad (RMSD >4Å) poses
RF-Score v3	0.724	0.652
RF-Score v3 + RDKit	0.738	0.707

Table 4.4: Effect of docking quality on affinity prediction accuracy. The mean Pearson correlation coefficient between predicted and experimental pK across five-fold cross-validation on a set of 1,284 complexes is shown for two RF models (one using RF-Score v3 features and one using RF-Score v3 + RDKit features) tested on either only 'good' (RMSD < 2Å) or only 'poor' (RMSD > 4Å) poses.

When only 'good' docked poses are used, there is a modest improvement in performance when using RF-Score v3 + RDKit features over RF-Score v3 features alone ($\rho_p = 0.738$ versus $\rho_p = 0.724$), consistent with the results of Chapter 2 and Chapter 3. However, when only 'poor' docked poses are available, there is a much larger improvement in performance when using RF-Score v3 + RDKit features over RF-Score v3 features alone ($\rho_p = 0.707$ versus $\rho_p = 0.652$. In addition to this, performance using RF-Score v3 + RDKit features with 'poor' docks is lower than performance using RF-Score v3 features alone with 'good' docks ($\rho_p = 0.707$ versus $\rho_p = 0.724$). This shows that the use of 'poor' (RMSD greater than 4Å) docked poses has a more deleterious effect on the performance of the structure-based scoring function than the use of 'good' (RMSD less than 4Å) docked poses. This further shows that the inclusion of ligand-based features in the scoring function has a greater beneficial effect when relying on poorly-docked poses.

These results suggest that where 'good' low-RMSD docked poses are available, structure-based methods can be expected to perform well, and the performance gained by augmenting with ligand-based features is comparable to that observed when training and testing on crystallographic binding poses. How-

ever, when there are errors in the docked poses, a purely structure-based approach performs substantially worse, and augmentation with ligand-based features helps to recover performance to near that achieved on low-RMSD docked poses.

4.3.4 | Scoring Function Performance Does Not Generalise to New Data Sets

We next tested our models on the ligands selected from ChEMBL for six DUD-E targets: serine/threonine-protein kinase AKT (AKT1), cytochrome P450 3A4 (CP3A4), glucocorticoid receptor (GCR), HIV-1 protease (HIVPR), HIV-1 reverse transcriptase (HIVRT), and kinesin-like protein 1 (KIF11). For each target, we prepared the PDB structure listed in Table 4.2 and the ligands downloaded from ChEMBL for docking according to the protocol described in Section 4.2.1. For each ligand, 20 poses were generated using Smina by docking into the rigid, crystallographic strucutre of the protein. As shown in Section 4.3.3, the choice of docked pose used to score a complex has an effect on the accuracy of the predicted affinity, with low-RMSD poses resulting in more accurate predictions than high-RMSD poses. Because crystal structures of the protein-ligand complex were not available for these proteins and their ligands, the accuracy of the docked poses was unknown. To address this, when testing scoring functions using this data set, we scored every docked pose for each ligand and took the highest score as the predicted pK of that ligand for its protein target. We chose to use the max score because there was no guarantee that the pose ranked highest

by Smina would be accurate, and as shown in Section 4.3.3, using the max score across multiple poses resulted in more accurate pK predictions than taking the mean score across multiple poses.

First, we tested models trained on docked poses of the PDBbind 2018 refined set. Figure 4.5 shows the predicted versus experimental pK values for the ligands of AKT1, CP3A4, and GCR, while Figure 4.6 shows the predicted versus experimental pK values for the ligands of HIVPR, HIVRT, and KIF11. Overall performance is poor, with no model achieving a Pearson correlation coefficient greater than 0.5 on any set of ligands. RF-Score v3 features and RF-Score v3 + RDKit features achieve a weak correlation ($\rho_p > 0.4$) for the ligands of AKT1, RDKit features achieves a weak correlation for the ligands of GCR, and all three models achieve a weak correlation for the ligands of HIVPR. The ligands of HIVRT and KIF11 were predicted especially poorly, with no model achieving a meaningful correlation between predicted and experimental pK for either target. The poor performance of all three models on the HIVPR dataset is particularly surprising: this target is well-represented in the PDBbind database, and the results of Chapter 3 suggest that these models should be expected to perform well on data for a protein target that is well represented in the training set. These results are in stark contrast with the strong performance achieved by these models on PDBbind data under cross-validation and on the combined core set, indicating that even under strict training and validation conditions, the models generalise poorly to data sourced from outside the PDBbind database.

Next, we re-trained the models using the ChEMBL data to determine whether training and testing on data from the same source would improve performance. We used two validation strategies: intra-target and inter-target. For the intra-

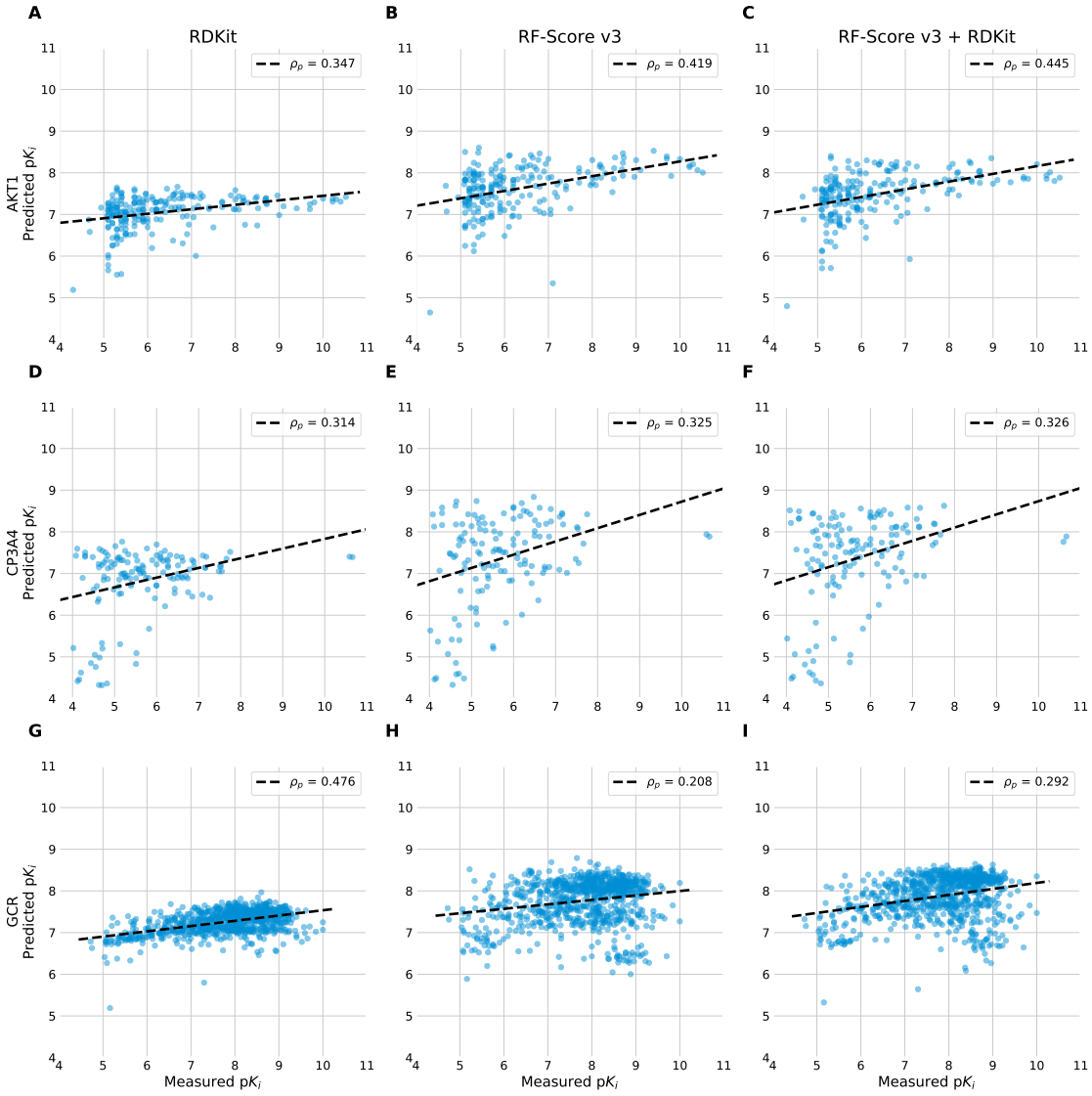


Figure 4.5: Predicted versus experimental pK values for ligands of the targets AKT1, CP3A4, and GCR by RFs using RDKit, RF-Score v3, and RF-Score v3 + RDKit features. Models were trained on docked poses of the PDBbind 2018 refined set. The broken black line indicates a linear regression fit to the points.

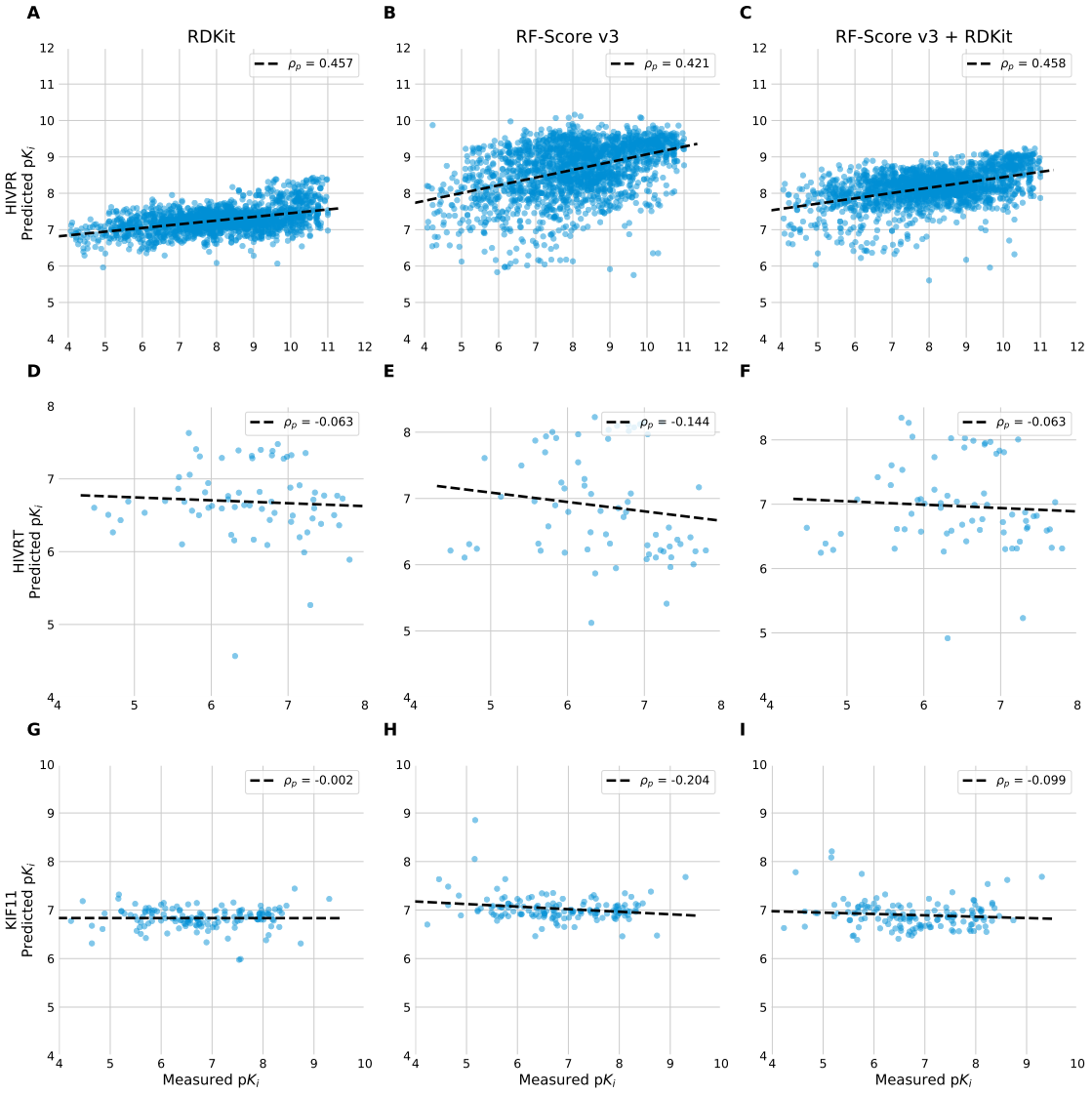


Figure 4.6: Predicted versus experimental pK values for ligands of the targets HIVPR, HIVRT, and KIF11 by RFs using RDKit, RF-Score v3, and RF-Score v3 + RDKit features. Models were trained on docked poses of the PDBbind 2018 refined set. The broken black line indicates a linear regression fit to the points.

target validation, the data for each target were randomly split into training and test sets in an 80:20 ratio, so that each model was trained and tested on data for a single target. For the inter-target validation, the data for each target were held out in turn as a test set, with the models trained on the data for the remaining five targets. In both cases, as the accuracy of the docked poses was unknown and a single pose was required for training, we chose to use the pose ranked highest by Smina for each training complex. As before, for each test complex we scored all 20 of the docked poses and took the max score as the predicted pK value for that complex.

Figure 4.7 shows the predicted versus experimental pK values for the ligands of AKT1, CP3A4, and GCR under intra-target validation, while Figure 4.8 shows the predicted versus experimental pK values for the ligands of HIVPR, HIVRT, and KIF11 under intra-target validation. Performance varies greatly between targets, but overall performance is much better than the case where models were trained on PDBbind data. In all but two cases, each model achieved a Pearson correlation coefficient in the range of $\rho_{\mathrm{p}} \approx 0.5$ to $\rho_{\mathrm{p}} \approx 0.8$, in contrast with the weak correlations obtained when training on PDBbind data ($\rho_{\mathrm{p}} < 0.5$. The two exceptions were CP3A4 and HIVRT, for which the model using RF-Score v3 features achieved correlation coefficients of 0.342 and 0.490 respectively. Models using RDKit features outperform models using RF-Score v3 features on every target and, with the exception of HIVRT (for which the sample size is small) the combination of RF-Score v3 + RDKit features is comparable to that of the RDKit features alone. This is not surprising in this intra-target validation scenario: ligands were docked into a single crystal structure of the target, so the structural information about the target is a fixed constant across each set of data.

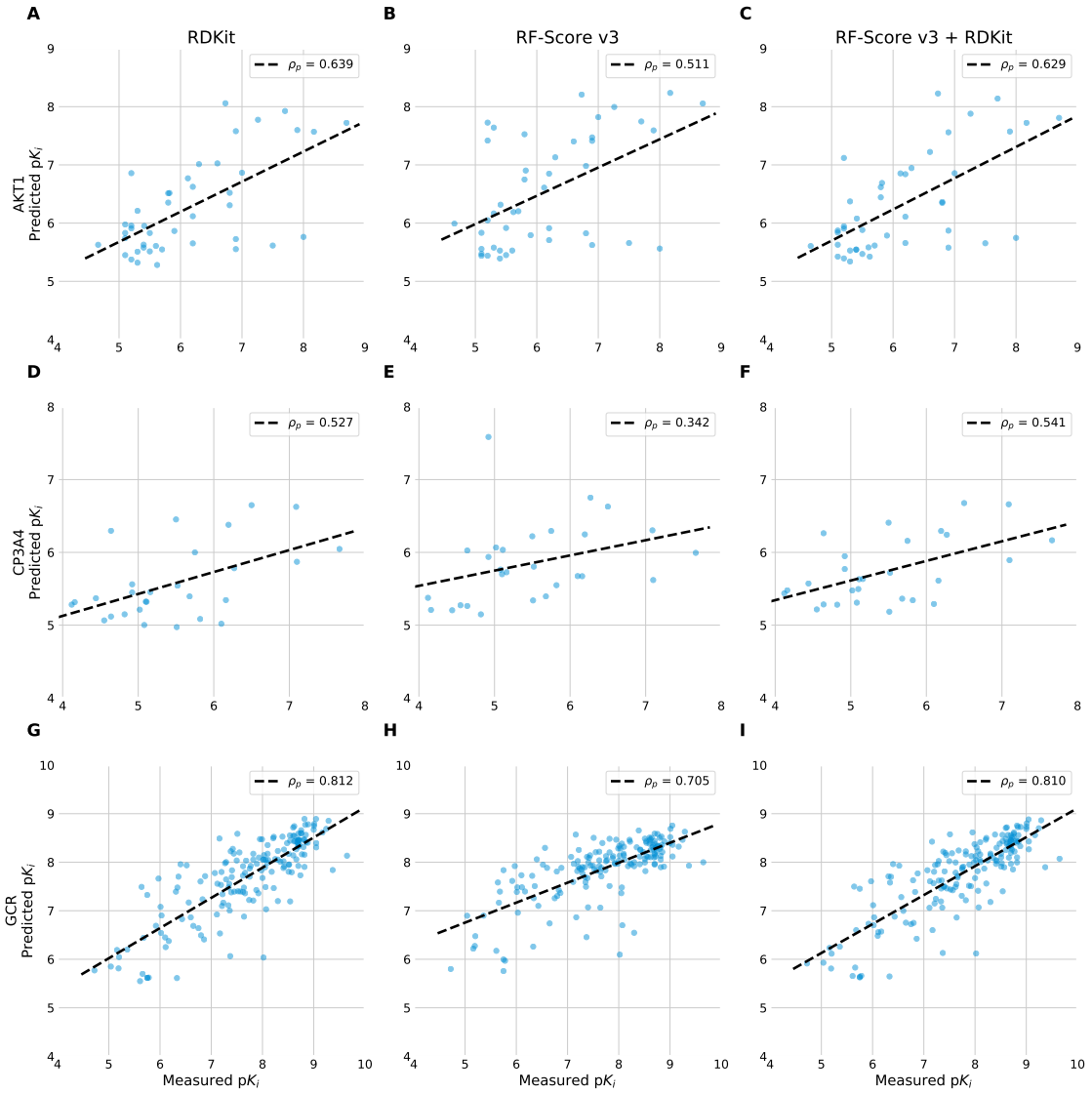


Figure 4.7: Intra-target predicted versus experimental pK values for ligands of the targets AKT1, CP3A4, and GCR by RFs using RDKit, RF-Score v3, and RF-Score v3 + RDKit features. For each target, 20% of the ligands for that target were held out as a test set, with the remaining 80% used as the training set. The broken black line indicates a linear regression fit to the points.

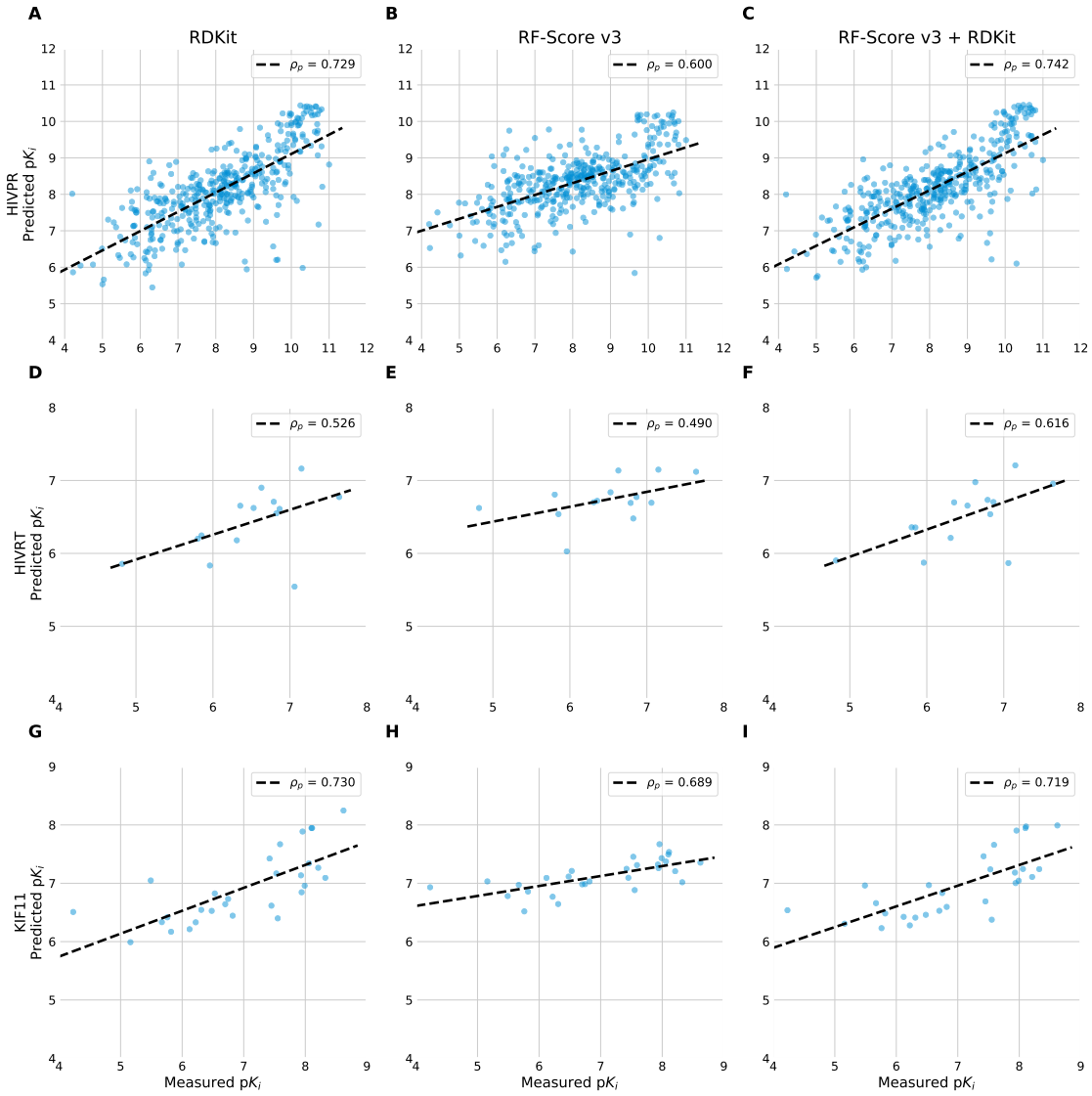


Figure 4.8: Intra-target predicted versus experimental pK values for ligands of the targets HIVPR, HIVRT, and KIF11 by RFs using RDKit, RF-Score v3, and RF-Score v3 + RDKit features. For each target, 20% of the ligands for that target were held out as a test set, with the remaining 80% used as the training set. The broken black line indicates a linear regression fit to the points.

Figure 4.9 shows the predicted versus experimental pK values for the ligands of AKT1, CP3A4, and GCR under inter-target validation, while Figure 4.10 shows the predicted versus experimental pK values for the ligands of HIVPR, HIVRT, and KIF11 under inter-target validation. In contrast with the intra-target validation, none of the models achieve a meaningful correlation between the predicted and experimental pK values for any of the targets ($\rho_{\mathrm{p}} < 0.4$), indicating that the models are unable to generalise to a previously-unseen target. With the exception of CP3A4, the predicted pK values also span a very narrow range of values compared to the range of experimentally-determined values. This is particularly striking for the HIVPR data (Figure 4.10 A-C), for which the predictions of the RF using RDKit features spans 2 pK units, and the predictions of the RF using RF-Score v3 features spans 1 pK unit. This echoes the results of Chapter 3 where the performance of models on individual protein targets or clusters of protein targets from the PDBbind database was inconsistent and often poor, and models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features all achieved similar performance on any given protein target or cluster. These results demonstrate that performance on a single benchmark, in this case the PDBbind core sets, is not always indicative of how well a scoring function can be expected to perform on a new data set.

4.4 | Summary

In this Chapter we investigated how the use of docked ligand poses instead of ligand crystal structures affects machine learning scoring function performance.

We re-docked the PDBbind refined set using Smina and found that in many

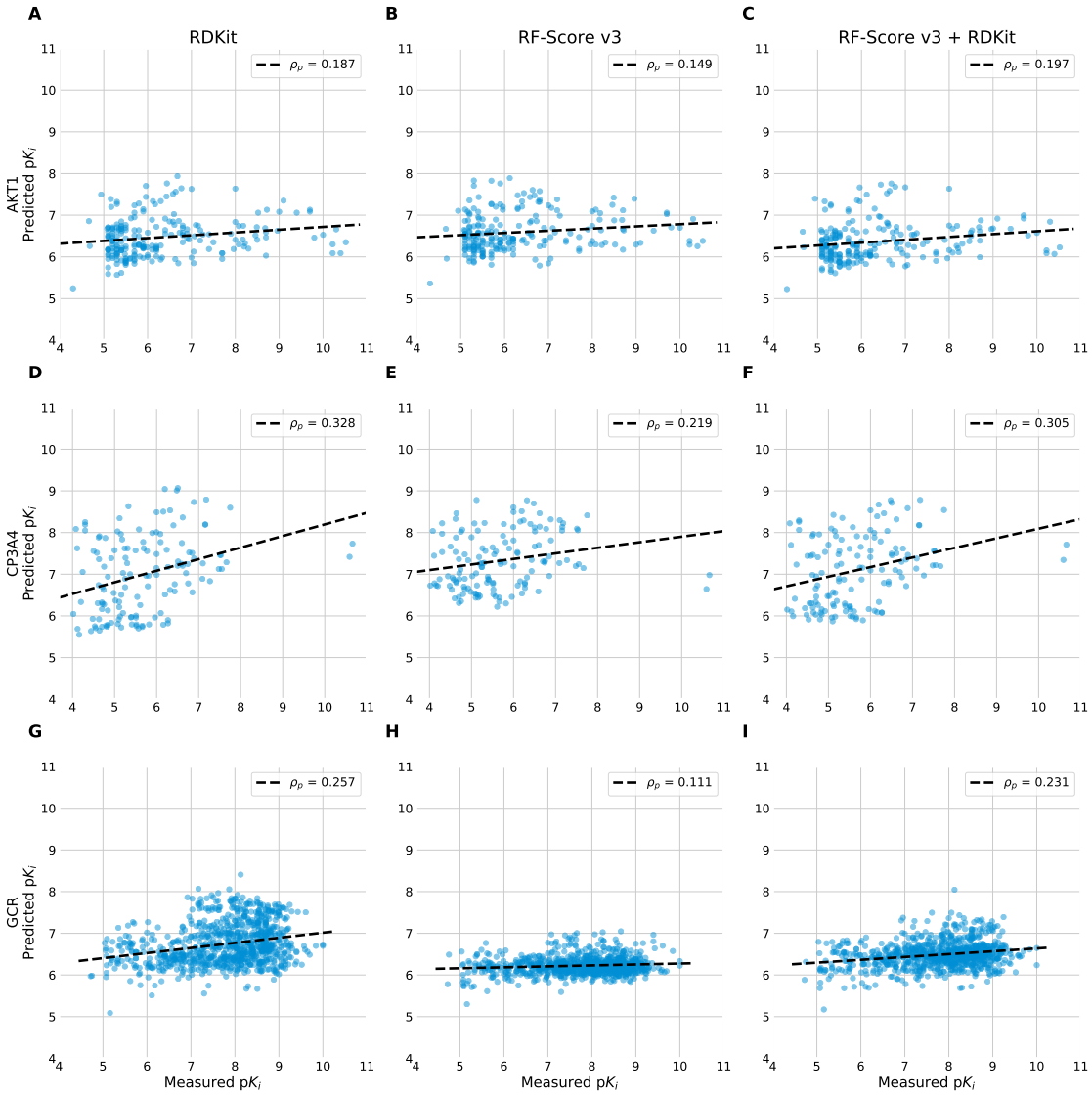


Figure 4.9: Inter-target predicted versus experimental pK values for ligands of the targets AKT1, CP3A4, and GCR by RFs using RDKit, RF-Score v3, and RF-Score v3 + RDKit features. For each target, all ligands for that target were held out as a test set, with the ligands of the five other targets used as the training set. The broken black line indicates a linear regression fit to the points.

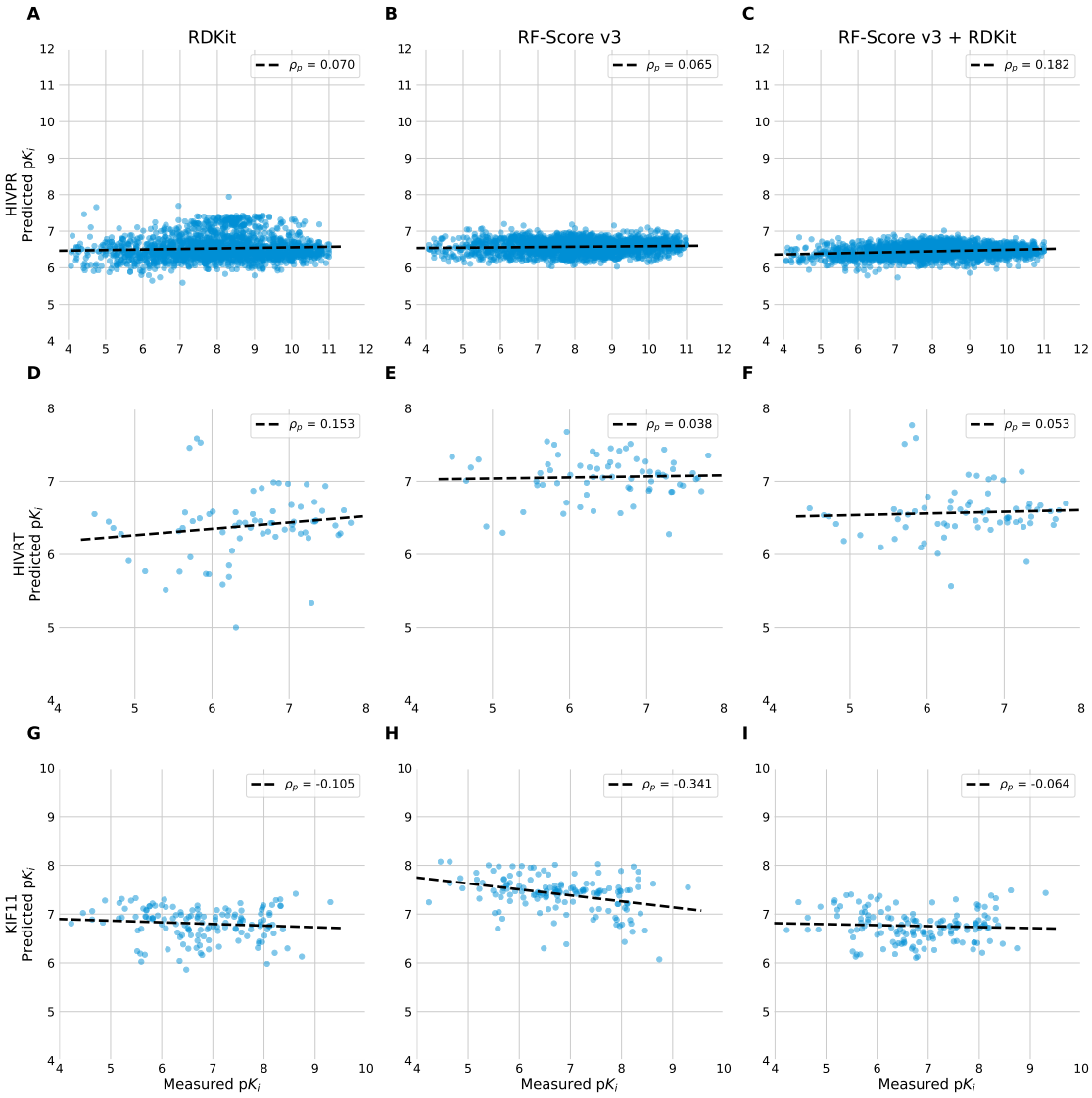


Figure 4.10: Inter-target predicted versus experimental pK values for ligands of the targets HIVPR, HIVRT, and KIF11 by RFs using RDKit, RF-Score v3, and RF-Score v3 + RDKit features. For each target, all ligands for that target were held out as a test set, with the ligands of the five other targets used as the training set. The broken black line indicates a linear regression fit to the points.

cases the native AutoDock Vina docking scoring function was unable to correctly rank the generated binding poses by their RMSD to the crystallographic binding mode, consistent with findings reported in the literature (Ramírez and Caballero, 2018). Further, binding poses with RMSD below 2Å to the crystallographic binding pose were generated for only one in three complexes. The quality of the best docked pose generated by Smina for each complex was not strongly correlated with the size or flexibility of the ligand, suggesting that the complexity of the ligand was not the cause of the failure to reproduce the crystallographic binding pose.

We found as expected that the use of Smina docked poses in place of X-ray crystallographic binding modes for training and validation substantially reduces the performance of a structure-based model. Further, we showed that a hybrid model combining structure-based and ligand-based features is less negatively affected by the use of docked poses than a purely structure-based model, and can achieve performance comparable to that of a structure-based model trained and tested on crystallographic binding poses. This suggests that the inclusion of ligand-based features into a scoring function can help to reduce the error in affinity prediction introduced by the use of docked poses.

We showed under cross-validation that the quality of the docked pose affects the performance of the model, with the use of 'good' poses with below 2Å RMSD to the crystal pose giving better performance than the use of 'poor' poses with greater than 4Å RMSD to the crystal pose for the same complexes. Further, we found that when multiple docked poses are available, scoring every pose and taking the highest score rather than scoring the pose ranked highest by the native AutoDock Vina scoring function results in slightly better correlations

between the predicted and experimentally-determined binding affinity.

We constructed a new data set of ligands for six targets found in DUD-E database using experimental measurements of the inhibition constant K_i from ChEMBL version 25 (Gaulton et al., 2017). Using docked poses generated by Smina, we applied models trained on docked poses of the PDBbind refined set to this new data set and found that the predictions correlated poorly with the ChEMBL affinity data for all six targets, suggesting that the models trained on PDBbind data generalise poorly to external data.

Under an intra-target validation, where 80% of the data for a target was used to train the model and 20% of the data was held out as a test set, models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features all achieved positive Pearson correlation coefficients between the predicted and experimental data, ranging from $\rho_p = 0.342$ to $\rho_p = 0.812$, indicating that these target-specific models are capable of predicting this data. The ligand-based model using RDKit features out-performed the structure-based model using RF-Score v3 features in this training scenario, suggesting that in the absence of a crystal structure of the protein-ligand complex, the use of well-studied ligand-based approaches such as those used in QSAR in place of structure-based methods may be sufficient if binding affinity data for other ligands are available.

Under an inter-target validation scenario, where data for five of the targets were used to train the model, with all of the data for the remaining target held out as a test set, models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features all failed to achieve any meaningful correlation between predicted and experimental binding affinity, echoing the results of Chapter 3 which showed that both structure-based and ligand-based scoring functions often failed to gen-

eralise to an unseen protein target. This is consistent with many studies in the literature, confirming that generalising to an unseen protein target remains a challenge for the field of scoring function development.

The results of this Chapter indicate that, on a benchmark such as the PDBbind core sets, the use of Smina docked poses for training and validation negatively affects the performance of a structure-based scoring function, and that the inclusion of ligand-based features in the scoring function helps to counteract this effect. However, we have also shown that a model trained on PDBbind data generalises poorly to an external data set, suggesting that additional validation and benchmarking sets are needed for scoring function development. Finally, when binding affinity data are available for some ligands of a protein target, a ligand-based method out-performs a structure-based method using Smina docked poses, suggesting that a ligand-based approach remains a competitive option in this scenario.

5

Conclusions and Future Directions

5.1 | Summary

5.1.1 | Introduction and Background

In Chapter 1 we gave an overview of the drug discovery process, a key challenge of which is predicting the strength of protein-ligand interactions. We introduced the principle of molecular recognition, and described how the change in free energy upon binding relates to the kinematics of protein-ligand interactions. We then discuss the use of computational methods for predicting protein-ligand interactions using virtual screening during early-stage drug discovery. The growing availability of structural and bioactivity data and computational power has led to an increasing use of computational methods in drug discovery, including docking, proteochemometric modeling, molecular dynamics, and free energy calculation.

We discussed the development of protein-ligand docking, in particular the challenge posed by modelling receptor and ligand flexibility. We described the

classical scoring functions used in protein-ligand docking to rapidly assess protein-ligand interactions, and discuss the challenge of rapidly, yet accurately, predicting binding affinity using a single, static snapshot of the system. We then reviewed recent developments in the use of machine learning methods to develop scoring functions that demonstrate significantly improved performance over classical scoring functions at the task of binding affinity prediction. Finally, we discuss molecular dynamics and free energy perturbation. These more fine-grained approaches to predicting binding modes and binding affinity represent an alternative paradigm to that of protein-ligand docking and scoring functions, offering a greater understanding of the dynamics of protein-ligand binding and rigorous calculation of the free energy of binding in exchange for greatly increased expenses in both time and computational resources.

In Chapter 2 we explored the use of different features and machine learning algorithms for predicting protein-ligand binding affinity. We examined the features used by four different scoring functions: the AutoDock Vina scoring function; RF-Score; RF-Score v3, and NNScore 2.0, as well as a diverse set of molecular descriptors for the ligand, computed using the open-source cheminformatics package RDKit.

Using the PDBbind 2016 refined set as our data set, we first performed an exploratory analysis of each of these sets of features. We analysed the correlations between features in each of these feature sets and found that, across our data set, many of the features within each feature set were highly correlated. We used principal component analysis to explore the dimensionality of each feature set and found that, for each feature set, much of the variance in the data could be explained by a small number of features.

We investigated the predictive power of each set of the Vina, RF-Score, RF-Score v3, and NNScore 2.0 features, and the effect of augment each of these sets of structure-based features with the ligand-based RDKit features. By training Random Forest regression models on the PDBbind 2016 refined set and testing on the PDBbind 2007 core set, we showed that a Random Forest using the combination of the Vina features and the RDKit features achieved performance comparable to Random Forest using the more complex RF-Score, RF-Score v3, or NNScore 2.0 features. We further found that the performance of a Random Forest model using RF-Score, RF-Score v3, or NNScore 2.0 features was also enhanced by the addition of the ligand-based RDKit features, suggesting that the inclusion of a more detailed physicochemical representation of the ligand can improve scoring function performance.

Having found that there were strong correlations within each feature set, and that much of the variance in the data could be captured using a small number of features, we explored the effect of using fewer features on scoring function performance. We found that for each set of features, using a subset of the most informative features resulted in performance close to that obtained when using the full set of features, suggesting that by eliminating less-informative features it is possible to develop simple, parsimonious scoring functions without sacrificing performance.

We next investigated the use of different machine learning algorithms for binding affinity prediction. We used regularised linear regression, artificial neural networks, AdaBoost, XGBoost, and Random forest, and found that under cross-validation using the PDBbind 2016 refined set, Random Forest and XGBoost models outperformed linear models, neural networks, and support vec-

tor machines using each of the previously-studied sets of features. We also observed that, for each machine learning algorithm, using only the ligand-based RDKit features resulted in higher cross-validation performance than using the Vina features. We then trained a Random Forest using only the RDKit features on the PDBbind 2016 refined set and tested this model on the PDBbind 2007 core set. We found that this ligand-only model achieved performance greater than both the AutoDock Vina scoring function and a Random Forest using the Vina features on this benchmark.

The results of Chapter 2 suggest that there are properties of a ligand that are intrinsically useful for binding affinity prediction, and that machine learning scoring functions can be easily improved by the inclusion of a diverse set of molecular descriptors of the ligand.

In Chapter 3 we investigated in more detail the effect of including ligand molecular descriptors in Random Forest scoring functions under different training and validation scenarios.

First, we showed that, when training Random Forest models on the PDBbind refined set or general set, using larger, more recent versions of the PDBbind database did not lead to improved performance when tested on the PDBbind 2007, 2013, or 2016 core sets. In contrast with this, we showed that excluding from the training set proteins and ligands that are similar to those in the test set had a deleterious effect on scoring function performance, regardless of the features used in the model. We also found that, regardless of the size of the training set and its similarity to the test sets, a hybrid scoring function combining structure-based features with the ligand-based RDKit features out-performed a scoring function using the structure-based features alone.

Next, we tested each model on bootstrapped samples of the PDBbind refined set, in each case training the models on the refined set excluding the bootstrapped sample. By varying chronologically the version of the refined set used, we demonstrated that the performance of Random Forest models using each set of features remained approximately constant across different versions of the refined set. Because the diversity of the structures found in the refined set increased with each successive release, this constant performance across different releases suggested that the domain of applicability of the models increased when trained on larger, more diverse sets of data, without sacrificing predictive power.

To investigate the predictive power of the ligand-based features, we examined the predictions of the ligand-only Random Forest model for ligands that bind to multiple targets. We found that the predicted binding affinity of a model using only ligand-based features was strongly correlated with the mean of the experimental protein-ligand binding affinity of a ligand for its binding partners. We showed that this correlation remained strong when ligands with a Tanimoto similarity of greater than 0.9 to the test ligand were excluded from the training data, and gradually weakened when progressively less similar ligands were also excluded. This suggested that while the model's predictions are not reliant on overfitting to previously-seen highly-similar ligands, it does not extrapolate well to completely novel ligands. We then examined the predictions of structure-based models for sets of structures featuring the same ligand and found that, regardless of the features used, the structure-based models failed to reproduce the range of experimentally-observed binding affinities, and often failed to achieve a correlation between the predicted and observed affinity data, suggesting that

generalising to a novel ligand is challenging for the structure-based models.

We also investigated the performance of our models on novel protein targets by clustering the PDBbind database by protein sequence similarity, and using each cluster in turn as a held-out validation set. We found that although performance varied greatly from protein to protein, both structure-based and ligand-based methods achieved comparable performance on any given protein. This suggests that both structural and ligand-based information may play an important role in the performance of a scoring function on a novel target. Further, these results demonstrate that, despite the ever-growing quantity and diversity of available training data, generalising to a previously-unseen protein target remains challenging.

Finally, we examined the relative importance of the structure-based and ligand-based features in the Random Forest models and found that, regardless of which structure-based features were used, the Random Forest algorithm ranked both structure-based and ligand-based features as important.

The results of Chapter 3 show that while generalising to novel targets and ligands remains challenging, the inclusion of rapidly-computed ligand-based features can consistently improve the performance of Random Forest-based scoring functions in a broad range of training and testing scenarios.

In Chapter 4 we assessed the impact of using docked poses in place of the experimentally-determined binding pose of the ligand for training and testing Random Forest scoring functions.

We first re-docked the protein-ligand complexes comprising the PDBbind 2018 refined set. We found that, unsurprisingly, errors in the predicted binding mode were common, with an RMSD of less than 2Å between the crystallo-

graphic binding mode and the best docked pose achieved for only a third of the complexes. We further observed that, even when accurate binding poses were generated, the native AutoDock Vina scoring function was not always able accurately rank the binding poses by their RMSD to the crystallographic binding mode. We showed that the quality of the lowest-RMSD pose was not strongly correlated with either the size or the flexibility of the ligand, suggesting that failure to reproduce the crystallographic binding mode could not be solely attributed to the complexity of the ligand.

We found as expected that the performance of structure-based Random Forest scoring functions was reduced when the model was trained and tested on docked poses instead of crystallographic binding modes. We also found that the performance of hybrid scoring functions combining structure-based and ligand-based features was less negatively impacted by the use of docked poses than that of scoring functions using structure-based features alone. This result suggests that including ligand-based features in a scoring function can help to reduce the deleterious effect that the use of docked poses has on the accuracy of binding affinity predictions.

Next, we performed a cross-validation experiment using complexes for with both 'good' (RMSD <2Å) and 'poor' (RMSD >4Å) docked poses were generated. We found that using low-RMSD poses resulted in more accurate affinity predictions than using high-RMSD poses. We further found that, when multiple docked poses are available, scoring each pose and taking the highest score resulted in slightly better correlations between the predicted and experimentally-observed binding affinity.

To explore how our results on PDBbind data generalised to unseen data from

outside the PDBbind database, we constructed a new data set of ligands for six targets found in DUD-E database using experimental measurements of the inhibition constant K_i from ChEMBL version 25. For each target and set of ligands, we generated docked poses using Smina. We tested models trained on the PDBbind database on this new data set and found that the predictions correlated poorly with the ChEMBL affinity data for all six targets, suggesting that the models trained on PDBbind data generalise poorly to external data.

We next re-trained each model using the ChEMBL data. Under an intra-target validation, where 80% of the data for a target was used to train the model and 20% of the data was held out as a test set, models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features all achieved positive Pearson correlation coefficients between the predicted and experimental data, demonstrating that these target-specific models are capable of predicting this data. We found that the ligand-based model using only RDKit features out-performed the structure-based model using RF-Score v3 features in this training scenario, suggesting that ligand-based methods are competitive with structure-based methods when known ligands for a target are available. In contrast, under an inter-target validation where data for five of the targets were used to train the model, with all of the data for the remaining target held out as a test set, models using RDKit, RF-Score v3, and RF-Score v3 + RDKit features all failed to achieve any meaningful correlation between predicted and experimental binding affinity. These results are consistent with both the results of Chapter 3 and many studies in the literature showing that prediction of protein-ligand binding affinity for novel protein targets remains challenging for scoring functions.

Although generalising to novel targets remains challenging, the results of

Chapter 4 show that the negative effect of using docked poses in place of crystallographic binding modes on scoring function performance can be lessened by augmenting the scoring function using a detailed set of ligand-based features.

5.2 | Future Directions

Here we describe future directions that might build upon and enhance the work presented in this thesis.

5.2.1 | Integration With Docking Tools

Throughout this thesis, we isolate the problem of binding affinity prediction from other tasks in structure-based drug discovery, firstly through the use of crystal structures of protein-ligand complexes in Chapters 2 and 3, and secondly through using the native AutoDock Vina scoring function when generating docked poses in Chapter 4. While this approach is beneficial when investigating if, how, and why a chosen technique leads to accurate predictions of binding affinity, the resulting models require additional work to be deployed as part of drug discovery pipeline involving protein-ligand docking.

We have shown that several machine learning scoring functions can be enhanced by the addition of readily-computed ligand molecular descriptors. However, the resulting machine learning models require both code (which has package dependencies) and training data to be used effectively. This presents both an additional barrier to their use, and an additional possible point of failure when integrated into a virtual screening pipeline. One way to reduce the bar-

rier to adoption of these models would be to integrate them into existing tools for protein-ligand docking, providing the end user with a single piece of software to perform both docking and final affinity predictions. Because our best-performing models make use of the terms used in the AutoDock Vina scoring function, a natural choice would be to integrate these models into either AutoDock Vina itself, or the wider AutoDock ecosystem. The latter is perhaps the most straightforward approach, as the AutoDockTools (Morris et al., 2009) suite is implemented in Python, the same language used for both feature computation and machine learning modelling. Another option is developing a Python wrapper for AutoDock Vina (Trott and Olson, 2010) Smina (Koes et al., 2013) that implements new Random Forest-based scoring functions as an optional post-docking re-scoring stage, and functionality to re-train the Random Forest scoring functions using new data and user-selected features. This would offer both easy access to pre-trained models for integration with existing pipelines as well as enabling users to make use of new or proprietary data sets without needing to work directly with the underlying code base.

5.2.2 | Enhanced Benchmarking for Scoring Functions

In Chapter 3 we found that the level of similarity between both the proteins and the ligands found in the training and test set strongly influenced the performance of all of the models we tested. We further showed that, when complexes whose ligands had a high Tanimoto similarity to those in the test set, or whose proteins had a high sequence identity to those in the test set, were excluded from the training set, there was little difference in performance between

models trained on the PDBbind general set and models trained on the PDBbind refined set. We also found that, under a leave-protein-cluster-out validation, model performance was highly protein-dependent. We also showed that the structure-based models tested were unable to reliably differentiate between different complexes featuring the same ligand bound with different affinities.

These results highlight the limitations of assessing scoring function performance on a standard benchmark without consideration for how the data used to train the scoring function influences its domain of applicability. As one of the main objectives in virtual screening is to identify novel hits for a target protein, it is important to understand not only how a scoring function performs on a diverse set of protein-ligand complexes, but also how is can be expected to perform when applied to novel targets and compounds. Community exercises such as CSAR (Dunbar et al., 2011, 2013; Smith et al., 2016; Carlson et al., 2016) and grand challenges such as D3R (Gathiaka et al., 2016; Gaieb et al., 2018, 2019) using previously-unseen data serve as a valuable means to assess the state of the field, but their use as resources to actively guide scoring function development and validation is limited by their lower quantitiy and diversity of data when compared to large publicly-available curated databases such as PDBbind (Liu et al., 2017), BindingDB (Gilson et al., 2015), and ChEMBL (Gaulton et al., 2017).

Rather than attempting to design and curate yet another database, one possible approach to address the need for a performance benchmark that captures and quantifies not only the performance of a scoring function, but also its domain of applicability, is to develop a protocol for partitioning an existing database into training and validation sets that control for similarity between data points. This is not a novel approach; on the contrary, it is common for benchmarking

sets used by the machine learning community to be partitioned in a manner that eliminates where possible obvious sources of bias that might influence performance. For example, the MNIST[1] database for image classification consists of images of hand-drawn digits, split into a training set and a test set. The images are grouped by author so that digits written by the same author are not found in both the training and test sets, ensuring that the the performance of models on the test set is not influenced by their having previously seen examples of handwriting of the same authors. Similar ideas have been applied to developing data sets for virtual screening: Rohrer and Baumann (2009) developed the MUV data sets to combat artificial enrichment in virtual screening data sets such as DUD-E (Mysinger et al., 2012) and DEKOIS2.0 (Bauer et al., 2013) caused by the similarity of active and decoy molecules in chemical space.

This approach to benchmark construction could be applied to binding affinity prediction by developing a protocol to automatically partition the PDBbind database into training and validation sets according to factors such as protein structural and sequence similarity, protein family and fold, and ligand chemical properties and scaffold, such as that presented in Chapter 3. This protocol could be applied to each release of the PDBbind database, or to other data sets such as proprietary in-house datasets in an industrial setting, to automatically generate a benchmark that assesses how a scoring function can be expected to generalise to novel data. This would offer a complementary benchmark to that of CASF (Cheng et al., 2009; Li et al., 2014c; Su et al., 2018) without the need to compile and curate an additional data set.

[1] http://yann.lecun.com/exdb/mnist/; last accessed 09/01/2020

5.2.3 | Binding Affinity Prediction Using Docked Poses

In Chapter 3 we found that, although augmenting a Random Forest scoring function with ligand-based features helped to reduce the deleterious effect of using docked poses for affinity prediction, these results did not generalise to new ligands and binding affinity data sourced from ChEMBL 25 (Gaulton et al., 2017). For the sake of expediency, we used only six protein targets, and docked the ligands for each into a single crystal structure of the receptor. We also treated the receptor as fully rigid. This rigid receptor assumption may have resulted in poor docking results as, unlike the case of re-docking the complexes in the PDBbind database, each ligand was not docked into a holo structure corresponding to that particular ligand. Repeating this experiment using flexible docking, which would allow flexible residues in the active site to adopt conformations more complementary to each ligand, could result in more accurate putative binding poses that may be more amenable to training and scoring.

This work might also be built upon by collecting ligands and binding data for additional targets from the DUD-E database. In addition to increasing the size of the training and validation sets, this would allow grouping of the targets by protein family in order to develop family-specific models. By comparing general-purpose scoring functions trained on a diverse set of proteins so family-specific scoring functions trained on a single protein family, this would provide insight into the question of whether 'one size fits all' (Ross et al., 2013).

5.3 | Final Words

We explored the use of different features and machine learning algorithms to develop scoring functions to predict protein-ligand binding affinity. We showed by augmenting structure-based scoring functions with rapidly-computed ligand-based features that the inclusion of a more detailed representation of the physicochemical properties of the ligand results in more accurate predictions of binding affinity. We carefully analysed how the size and diversity of the training set, and the similarity between the proteins in the training and test set, affects benchmarking of scoring function performance. We showed that even when proteins and ligands similar to those in the test set are excluded from the training set, the addition of ligand-based features still improves scoring function performance. We investigated the effect of using docked poses in place of crystallographic binding modes, and found that the inclusion of ligand-based features in a scoring function helps to reduce the deleterious effect that using docked poses has on scoring function performance. Although there are still challenges to be addressed, the last decade has seen great advances in scoring function development. The next decade will undoubtedly see the state of the art progress even further and bring powerful new tools to bear on the problem of drug discovery.

A

Appendix

A.1 | List of Features

RDKit Features

The following RDKit molecular descriptors comprise the 'RDKit features' feature set described in Chapter 2:
BalabanJ, BertzCT, Chi0, Chi0n, Chi0v, Chi1, Chi1n, Chi1v, Chi2n, Chi2v, Chi3n, Chi3v, Chi4n, Chi4v, EState_VSA1, EState_VSA10, EState_VSA11, EState_VSA2, EState_VSA3, EState_VSA4, EState_VSA5, EState_VSA6, EState_VSA7, EState_VSA8, EState_VSA9, ExactMolWt, FpDensityMorgan1, FpDensityMorgan2, FpDensityMorgan3, FractionCSP3, HallKierAlpha, HeavyAtomCount, HeavyAtomMolWt, Kappa1, Kappa2, Kappa3, LabuteASA, MaxAbsEStateIndex, MaxAbsPartialCharge, MaxEStateIndex, MaxPartialCharge, MinAbsEStateIndex, MinAbsPartialCharge, MinEStateIndex, MinPartialCharge, MolLogP, MolMR, MolWt, NHOHCount, NOCount, NumAliphaticCarbocycles, NumAliphaticHeterocycles, NumAliphaticRings, NumAromaticCarbocycles, NumAromaticHeterocycles, NumAromati-

cRings, NumHAcceptors, NumHDonors, NumHeteroatoms, NumRadicalElectrons, NumRotatableBonds, NumSaturatedCarbocycles, NumSaturatedHeterocycles, NumSaturatedRings, NumValenceElectrons, PEOE_VSA1, PEOE_VSA10, PEOE_VSA11, PEOE_VSA12, PEOE_VSA13, PEOE_VSA14, PEOE_VSA2, PEOE_VSA3, PEOE_VSA4, PEOE_VSA5, PEOE_VSA6, PEOE_VSA7, PEOE_VSA8, PEOE_VSA9, RingCount, SMR_VSA1, SMR_VSA10, SMR_VSA2, SMR_VSA3, SMR_VSA4, SMR_VSA5, SMR_VSA6, SMR_VSA7, SMR_VSA9, SlogP_VSA1, SlogP_VSA10, SlogP_VSA11, SlogP_VSA12, SlogP_VSA2, SlogP_VSA3, SlogP_VSA4, SlogP_VSA5, SlogP_VSA6, SlogP_VSA7, SlogP_VSA8, TPSA, VSA_EState10, VSA_EState4, VSA_EState5, VSA_EState8, VSA_EState9, fr_Al_COO, fr_Al_OH, fr_Al_OH_noTert, fr_ArN, fr_Ar_COO, fr_Ar_N, fr_Ar_NH, fr_Ar_OH, fr_COO, fr_COO2, fr_C_O, fr_C_O_noCOO, fr_C_S, fr_HOCCN, fr_Imine, fr_NH0, fr_NH1, fr_NH2, fr_N_O, fr_Ndealkylation1, fr_Ndealkylation2, fr_Nhpyrrole, fr_SH, fr_aldehyde, fr_alkyl_carbamate, fr_alkyl_halide, fr_allylic_oxid, fr_amide, fr_amidine, fr_aniline, fr_aryl_methyl, fr_azide, fr_azo, fr_barbitur, fr_benzene, fr_benzodiazepine, fr_bicyclic, fr_dihydropyridine, fr_epoxide, fr_ester, fr_ether, fr_furan, fr_guanido, fr_halogen, fr_hdrzine, fr_hdrzone, fr_imidazole, fr_imide, fr_isothiocyan, fr_ketone, fr_ketone_Topliss, fr_lactam, fr_lactone, fr_methoxy, fr_morpholine, fr_nitrile, fr_nitro, fr_nitro_arom, fr_nitroso, fr_oxazole, fr_oxime, fr_para_hydroxylation, fr_phenol, fr_phenol_noOrthoHbond, fr_piperdine, fr_piperzine, fr_priamide, fr_pyridine, fr_quatN, fr_sulfide, fr_sulfonamd, fr_sulfone, fr_term_acetylene, fr_tetrazole, fr_thiazole, fr_thiocyan, fr_thiophene, fr_urea, qed

For implementation details and further information about these descriptors, we refer the reader to the official RDKit documentation (Landrum, n.d.b).

NNScore 2.0 Features

The following features from NNScore 2.0 (Durrant and McCammon, 2011b) comprise the 'NNScore 2.0 features' feature set described in Chapter 2:
as_flex_all, as_flex_backbone_alpha, as_flex_backbone_beta, as_flex_backbone_other, as_flex_sidechain_alpha, as_flex_sidechain_beta, as_flex_sidechain_other, cc_A.A_2.5, cc_A.A_4, cc_A.BR_4, cc_A.CL_4, cc_A.C_2.5, cc_A.C_4, cc_A.F_2.5, cc_A.F_4, cc_A.HD_2.5, cc_A.HD_4, cc_A.I_4, cc_A.NA_4, cc_A.N_2.5, cc_A.N_4, cc_A.OA_2.5, cc_A.OA_4, cc_A.P_4, cc_A.S_4, cc_C.CD_4, cc_C.CL_2.5, cc_C.CL_4, cc_C.C_2.5, cc_C.C_4, cc_C.F_2.5, cc_C.F_4, cc_C.HD_2.5, cc_C.HD_4, cc_C.I_4, cc_C.MG_4, cc_C.NA_4, cc_C.N_2.5, cc_C.N_4, cc_C.OA_2.5, cc_C.OA_4, cc_C.P_4, cc_C.S_4, cc_CD.OA_2.5, cc_CU.HD_4, cc_CU.N_4, cc_FE.HD_2.5, cc_FE.HD_4, cc_FE.NA_4, cc_FE.N_2.5, cc_FE.N_4, cc_FE.OA_2.5, cc_FE.OA_4, cc_HD.HD_2.5, cc_HD.HD_4, cc_HD.I_4, cc_HD.MG_4, cc_HD.NA_2.5, cc_HD.NA_4, cc_HD.N_2.5, cc_HD.N_4, cc_HD.OA_2.5, cc_HD.OA_4, cc_HD.P_2.5, cc_HD.P_4, cc_HD.S_2.5, cc_HD.S_4, cc_MG.NA_4, cc_MG.N_4, cc_MG.OA_2.5, cc_MG.OA_4, cc_MG.P_4, cc_MG.S_4, cc_MN.NA_4, cc_MN.N_2.5, cc_MN.N_4, cc_MN.OA_2.5, cc_MN.OA_4, cc_MN.P_4, cc_MN.S_4, cc_N.NA_2.5, cc_N.NA_4, cc_N.N_2.5, cc_N.N_4, cc_N.OA_2.5, cc_N.OA_4, cc_N.P_4, cc_N.S_4, cc_NA.OA_2.5, cc_NA.OA_4, cc_NA.P_4, cc_NA.S_4, cc_OA.OA_2.5, cc_OA.OA_4, cc_OA.P_4, cc_OA.S_4, ele_A.A_4, ele_A.BR_4, ele_A.CL_4, ele_A.C_4, ele_A.F_4, ele_A.HD_4, ele_A.I_4, ele_A.NA_4, ele_A.N_4, ele_A.OA_4, ele_A.P_4, ele_A.S_4, ele_C.CL_4, ele_C.C_4, ele_C.F_4, ele_C.HD_4, ele_C.I_4, ele_C.NA_4, ele_C.N_4, ele_C.OA_4, ele_C.P_4, ele_C.S_4, ele_HD.HD_4, ele_HD.I_4, ele_HD.NA_4, ele_HD.N_4, ele_HD.OA_4, ele_HD.P_4, ele_HD.S_4, ele_N.NA_4, ele_N.N_4, ele_N.OA_4, ele_N.P_4, ele_N.S_4, ele_NA.OA_4, ele_NA.P_4, ele_NA.S_4, ele_OA.OA_4,

ele_OA.P_4, ele_OA.S_4, hb_4_mol_backbone_alpha, hb_4_mol_backbone_beta, hb_4_mol_backbone_other, hb_4_mol_sidechain_alpha, hb_4_mol_sidechain_beta, hb_4_mol_sidechain_other, hb_4_rec_backbone_alpha, hb_4_rec_backbone_beta, hb_4_rec_backbone_other, hb_4_rec_sidechain_alpha, hb_4_rec_sidechain_beta, hb_4_rec_sidechain_other, hyd_4_all, hyd_4_backbone_other, hyd_4_sidechain_alpha, hyd_4_sidechain_beta, hyd_4_sidechain_other, lig_A, lig_BR, lig_C, lig_CL, lig_F, lig_HD, lig_I, lig_N, lig_NA, lig_OA, lig_P, lig_S, num_rotors, pi_cat_mol_6_alpha, pi_cat_mol_6_beta, pi_cat_mol_6_other, pi_cat_rec_6_alpha, pi_cat_rec_6_beta, pi_cat_rec_6_other, pi_stack_7.5_alpha, pi_stack_7.5_beta, pi_stack_7.5_other, pi_t_7.5_alpha, pi_t_7.5_beta, pi_t_7.5_other, salt_bridge_5.5_all, salt_bridge_5.5_alpha, salt_bridge_5.5_beta, salt_bridge_5.5_other, vina_gauss1, vina_gauss2, vina_hydrogen, vina_hydrophobic, vina_repulsion

For more details on these features, and their implementation in the BINANA algorithm, see Durrant and McCammon (2011a). We used the implementation of BINANA provided in the open-source Python package `oddt`. For details see either Wójcikowski et al. (2015) or the official `oddt` documentation[1].

[1] https://oddt.readthedocs.io/en/latest/; last accessed 09/01/2020.

10GS	1A30	1A69	1ABF	1AI5	1AJP	1AJQ	1APW	1AVN
1AX0	1AXZ	1B11	1B39	1B7H	1B8O	1B9J	1BCU	1BMA
1BRA	1BXO	1C84	1D7J	1DET	1DF8	1DHI	1E1V	1E5A
1E66	1ELA	1ELB	1F4E	1F4F	1F4G	1F5K	1FCX	1FCZ
1FD0	1FH7	1FH8	1FH9	1FKB	1FKI	1FKN	1FLR	1FO0
1FTM	1FZK	1G7Q	1GNI	1GPK	1GZ9	1H23	1HA2	1HFS
1HI4	1HK4	1IF7	1J16	1J17	1JAQ	1JQ9	1JQD	1JQE
1JYS	1K4G	1K9S	1KV1	1L2S	1L83	1LI3	1LI6	1LOL
1LOQ	1M0N	1M0Q	1M2Q	1MQ6	1N2V	1NC1	1NDW	1NDY
1NDZ	1NFY	1NHU	1NJA	1NJE	1NNY	1NVQ	1NWL	1O0H
1O3F	1O3P	1OM1	1P1Q	1PB9	1PBQ	1PPM	1PR5	1PXO
1PZ5	1Q7A	1Q8T	1RDI	1RDJ	1RDL	1RE8	1RNT	1S39
1SL3	1SLG	1SQA	1SV3	1SYH	1TMN	1TOI	1TOJ	1TOK
1TRD	1TSY	1TTM	1TYR	1U1B	1U2Y	1U33	1UTP	1UWT
1V2O	1V48	1VFN	1VZQ	1X1Z	1XGJ	1Y1M	1Y6Q	1YDT
1ZC9	1ZOE	1ZS0	1ZVX	2AOU	2AYR	2B1V	2B7D	2BAJ
2BAK	2BOK	2BRB	2BRM	2BZ6	2BZZ	2C02	2CEQ	2CER
2CET	2CGR	2CTC	2D0K	2D1O	2D3U	2D3Z	2DRC	2ER9
2F01	2F80	2FAI	2FDP	2FLB	2FZC	2G5U	2G8R	2G94
2GSS	2HDQ	2I0D	2J77	2J78	2QWB	2QWD	2QWE	2RKM
2STD	2USN	3GSS	3PCE	3PCH	3PCJ	3STD	4ER2	4FIV
4TLN	5ABP	5ER1	6FIV	6RNT	6STD	8ABP		

Table A.1: PDB IDs of the structures used from the PDBbind 2007 core set. Structures from the PDBbind 2007 core set for which features could not be computed were excluded.

10GS	1A30	1BCU	1E66	1F8B	1F8C	1F8D	1GPK	1H23
1HFS	1HNN	1IGJ	1LBK	1LOL	1LOQ	1LOR	1MQ6	1N1M
1N2V	1NVQ	1O3F	1O5B	1OYT	1P1Q	1PS3	1Q8T	1Q8U
1QI0	1R5Y	1SLN	1SQA	1U1B	1U33	1UTO	1W3K	1W3L
1W4O	1XD0	1YC1	1Z95	1ZEA	2BRB	2CBJ	2CET	2D1O
2D3U	2FVD	2G70	2GSS	2HB1	2IWX	2J62	2J78	2JDM
2JDU	2JDY	2OBF	2OLE	2P4Y	2PCP	2QBP	2QBR	2QMJ
2R23	2V00	2V7A	2VL4	2VO5	2VOT	2VVN	2VW5	2W66
2WBG	2WCA	2WEG	2WTV	2X00	2X0Y	2X8Z	2XB8	2XBV
2XDL	2XHM	2XNB	2XYS	2Y5H	2YFE	2YGE	2YKI	2YMD
2ZJW	2ZWZ	2ZX6	2ZXD	3ACW	3AG9	3AO4	3B3S	3B3W
3B68	3BFU	3BKK	3BPC	3CFT	3CJ2	3COY	3CYX	3D4Z
3DD0	3DXG	3E93	3EBP	3EHY	3EJR	3F17	3F3A	3F3C
3F3E	3F80	3FCQ	3FV1	3G0W	3G2N	3G2Z	3GBB	3GCS
3GE7	3GNW	3GY4	3HUC	3IMC	3IVG	3JVS	3K5V	3KGP
3KV2	3KWA	3L3N	3L4U	3L4W	3L7B	3LKA	3MFV	3MSS
3MYG	3N7A	3N86	3NOX	3NQ3	3NW9	3OE5	3OV1	3OWJ
3OZT	3PE2	3PWW	3PXF	3S8O	3SU2	3SU3	3SU5	3U9Q
3UDH	3UEU	3UEX	3UO4	3URI	3UTU	3VH9	3ZSO	3ZSX
4DE1	4DE2	4DES	4DEW	4DJR	4DJV	4G8M	4GID	4GQQ

Table A.2: PDB IDs of the structures used from the PDBbind 2013 core set. Structures from the PDBbind 2013 core set for which features could not be computed were excluded.

1A30	1BCU	1C5Z	1E66	1EBY	1G2K	1GPK	1GPN	1H22
1H23	1K1I	1LPG	1MQ6	1NC1	1NC3	1NVQ	1O0H	1O3F
1O5B	1OWH	1OYT	1P1N	1P1Q	1PS3	1PXN	1Q8T	1Q8U
1QF1	1QKT	1R5Y	1S38	1SQA	1SYI	1U1B	1UTO	1W4O
1Y6R	1YC1	1YDR	1YDT	1Z6E	1Z95	1Z9G	2AL5	2BR1
2BRB	2C3I	2CBV	2CET	2FVD	2FXS	2HB1	2IWX	2J78
2J7H	2P15	2P4Y	2POG	2QBP	2QBQ	2QBR	2QE4	2QNQ
2R9W	2V00	2V7A	2VKM	2VVN	2VW5	2W4X	2W66	2WBG
2WCA	2WEG	2WER	2WN9	2WNC	2WTV	2WVT	2X00	2XB8
2XBV	2XDL	2XII	2XJ7	2XNB	2XYS	2Y5H	2YFE	2YGE
2YKI	2YMD	2ZB1	2ZDA	2ZY1	3ACW	3AG9	3AO4	3ARP
3ARQ	3ARU	3ARV	3ARY	3B1M	3B27	3B5R	3B65	3B68
3BGZ	3BV9	3CJ4	3COY	3COZ	3D4Z	3D6Q	3DD0	3DX1
3DX2	3DXG	3E5A	3E92	3E93	3EBP	3EHY	3EJR	3F3A
3F3C	3F3D	3F3E	3FCQ	3FUR	3FV1	3FV2	3G0W	3G2N
3G2Z	3G31	3GBB	3GC5	3GE7	3GNW	3GR2	3GV9	3GY4
3IVG	3JVR	3JVS	3JYA	3K5V	3KGP	3KR8	3KWA	3L7B
3LKA	3MSS	3MYG	3N76	3N7A	3N86	3NQ9	3NW9	3NX7
3O9I	3OE4	3OE5	3OZS	3OZT	3P5O	3PRS	3PWW	3PXF
3PYY	3QGY	3QQS	3R88	3RLR	3RR4	3RSX	3RYJ	3SYR
3TSK	3TWP	3U5J	3U8K	3U8N	3U9Q	3UDH	3UEU	3UEV
3UEW	3UEX	3UI7	3UO4	3UP2	3URI	3UTU	3UUO	3WTJ
3WZ8	3ZDG	3ZSO	3ZSX	3ZT2	4ABG	4AGN	4AGP	4AGQ
4BKT	4CIG	4CIW	4CR9	4CRA	4CRC	4DDH	4DDK	4DE1
4DE2	4DE3	4DJV	4DLD	4DLI	4E5W	4E6Q	4EA2	4EKY
4EO8	4EOR	4F09	4F2W	4F3C	4F9W	4GFM	4GID	4GKM
4GR0	4HGE	4IH5	4IH7	4IVB	4IVC	4IVD	4J21	4J28
4J3L	4JFS	4JIA	4JSZ	4JXS	4K18	4K77	4KZ6	4KZQ
4KZU	4LLX	4LZS	4M0Y	4M0Z	4MGD	4MME	4OGJ	4OWM
4PCS	4QAC	4QD6	4RFM	4TWP	4TY7	4U4S	4W9C	4W9H
4W9I	4W9L	4WIV	4X6P	5A7B	5ABA	5C28	5C2H	5DWR

Table A.3: PDB IDs of the structures used from the PDBbind 2016 core set. Structures from the PDBbind 2016 core set for which features could not be computed were excluded.

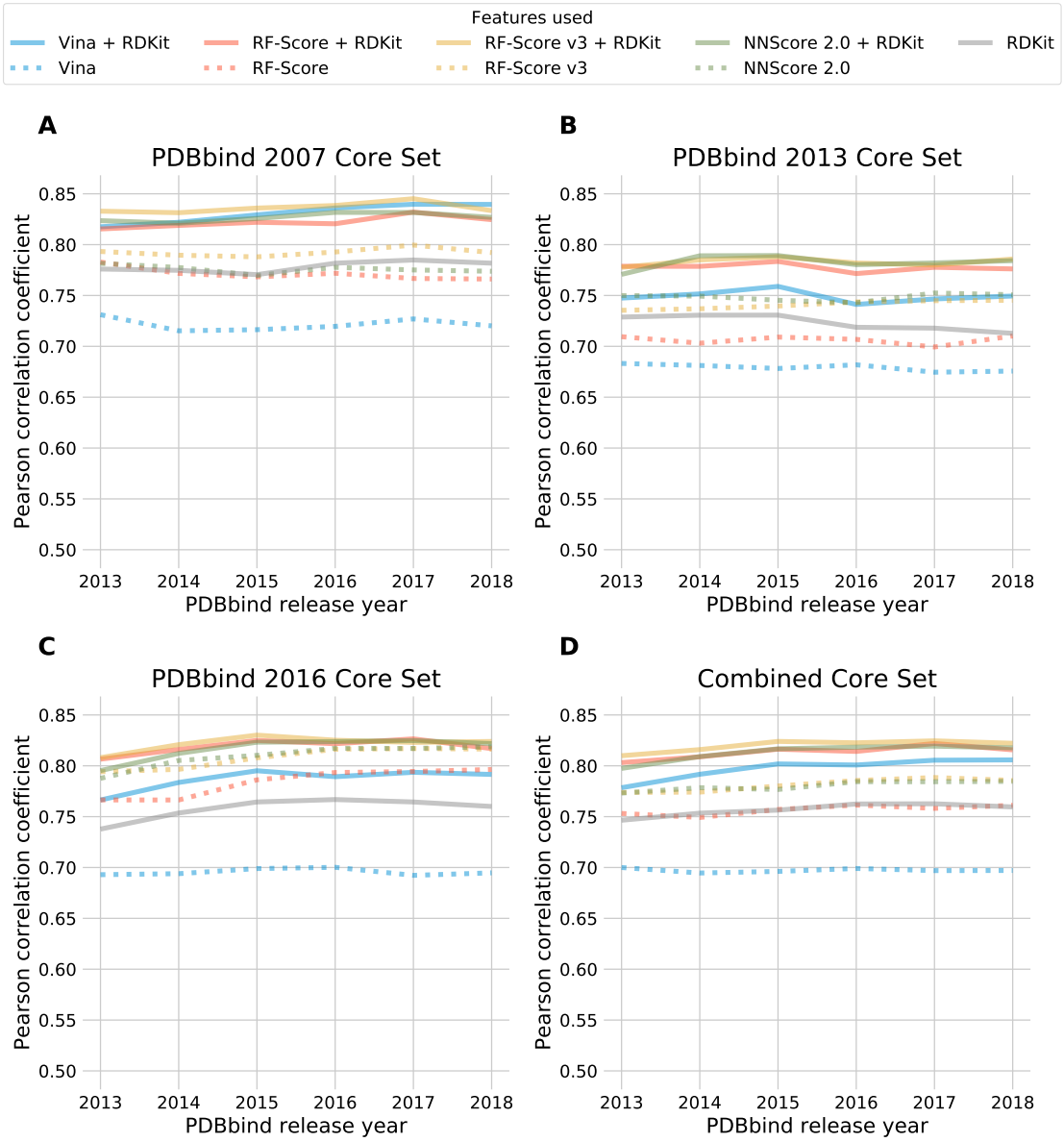


Figure A.1: Performance of RF scoring functions on the 2007, 2013, 2016, and combined core sets when trained on different versions of the PDBbind 2018 general set. The six versions of the PDBbind general set were used, from the 2013 release to the 2018 release inclusive, are indicated on the horizontal axis.

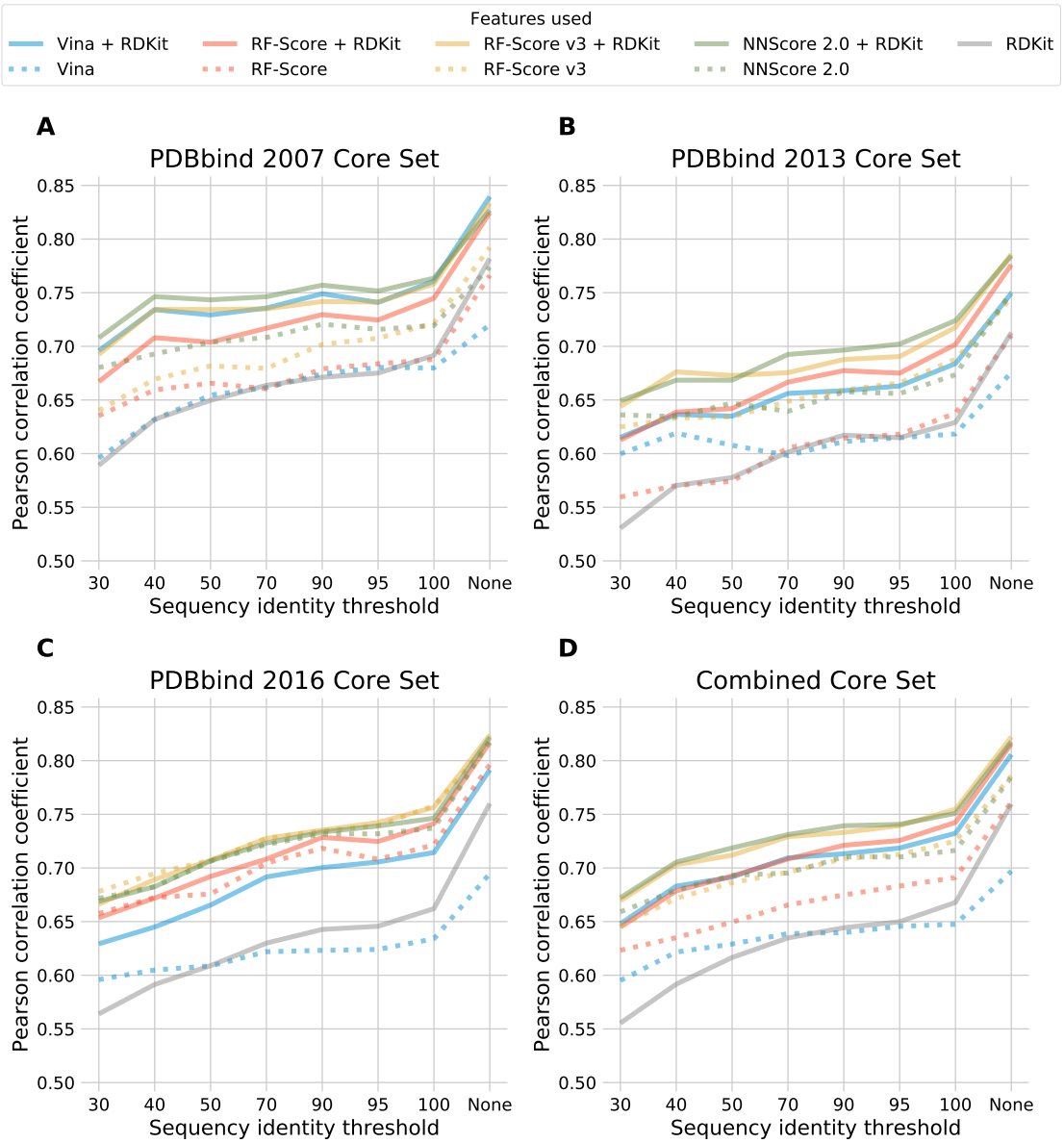


Figure A.2: Performance of RF scoring functions on the 2007, 2013, 2016, and combined core sets when trained on the PDBbind 2018 general set. The protein sequence identity percentage above which structures similar to those in the test set were excluded from the training set is indicated on the horizontal axis.

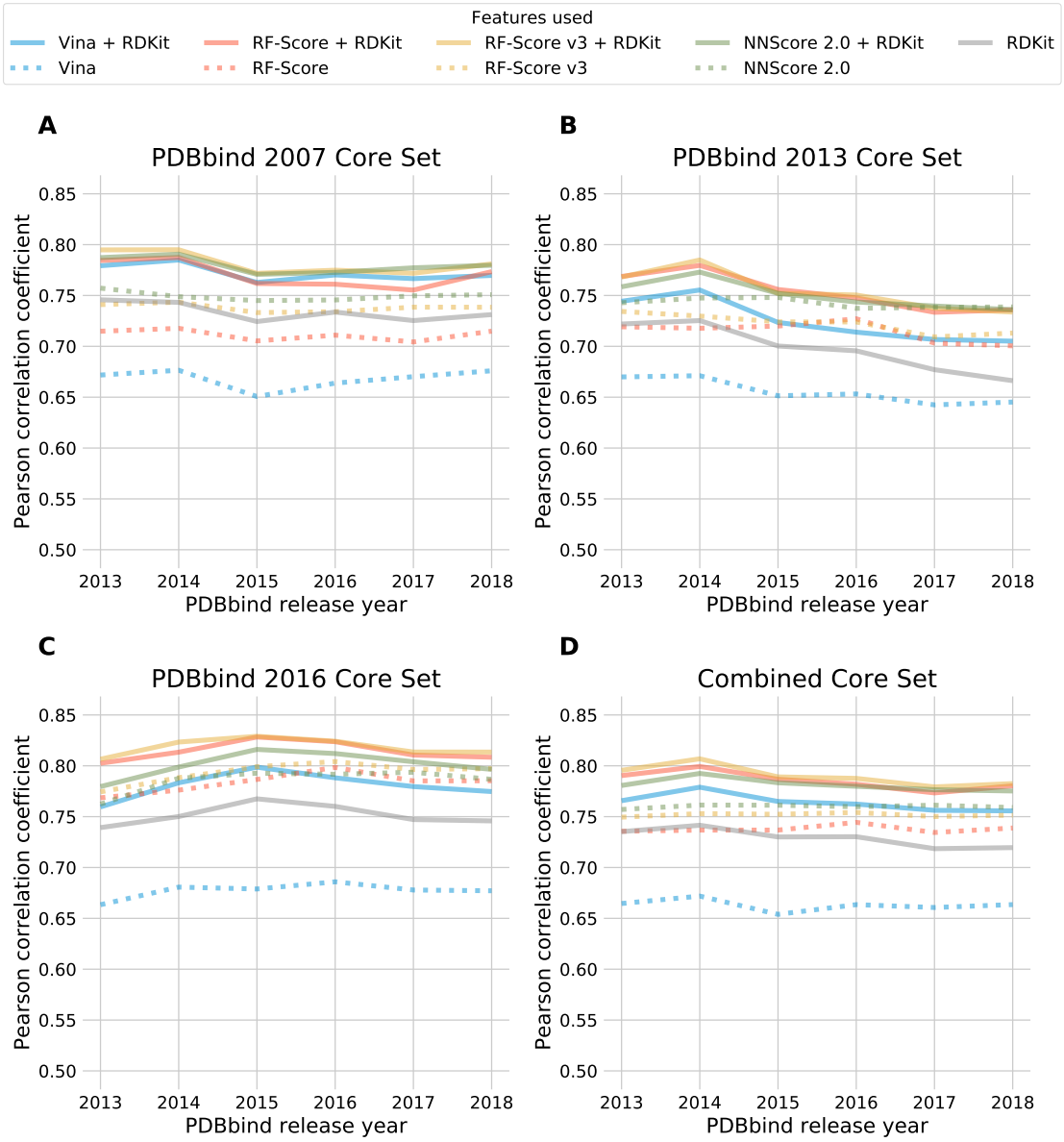


Figure A.3: Performance of RF scoring functions on the 2007, 2013, 2016, and combined core sets when trained on different versions of the PDBbind 2018 refined set. The six versions of the PDBbind refined set were used, from the 2013 release to the 2018 release inclusive, are indicated on the horizontal axis.

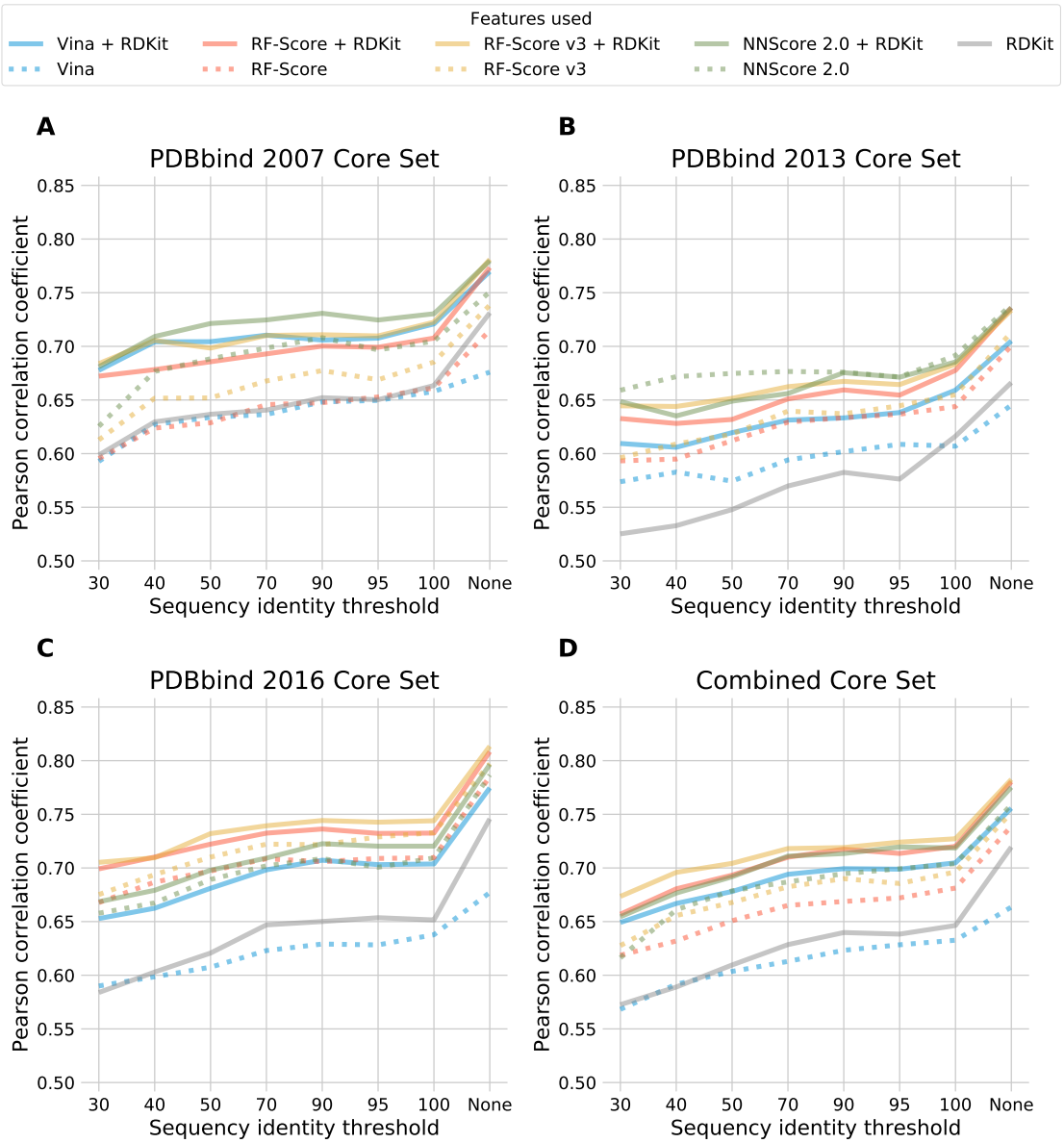


Figure A.4: Performance of RF scoring functions on the 2007, 2013, 2016, and combined core sets when trained on the PDBbind 2018 refined set. The protein sequence identity percentage above which structures similar to those in the test set were excluded from the training set is indicated on the horizontal axis.

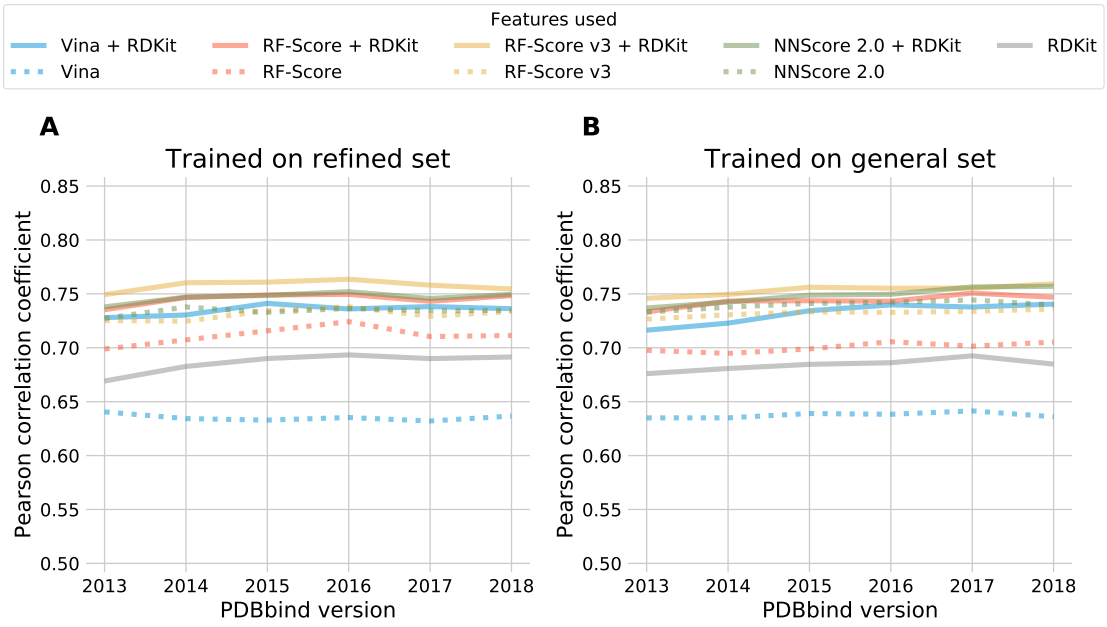


Figure A.5: Performance of RF scoring functions when trained on different versions of the PDBbind refined set (A) or general set (B), when ligands with Tanimoto similarity greater than 0.9 to those in the test set were excluded from the training set. The version of the PDBbind refined or general set is indicated on the horizontal axis.

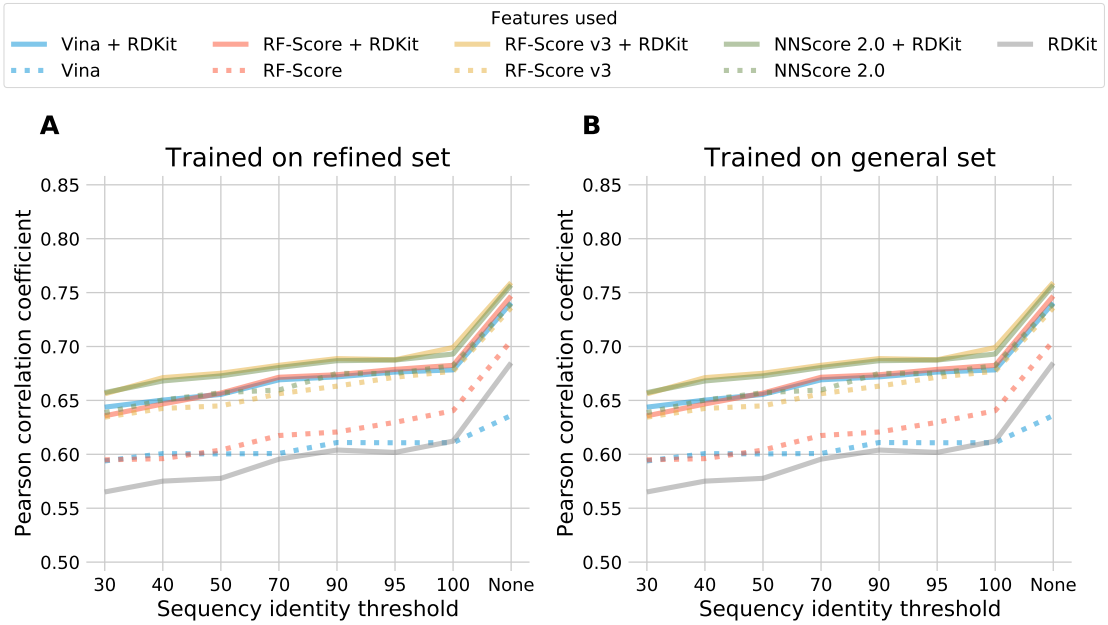


Figure A.6: Performance of RF scoring functions when trained on the PDBbind 2018 refined set (A) or general set (B), when ligands with Tanimoto similarity greater than 0.9 to those in the test set were excluded from the training set. The protein sequence identity percentage above which structures similar to those in the test set were excluded from the training set is indicated on the horizontal axis.

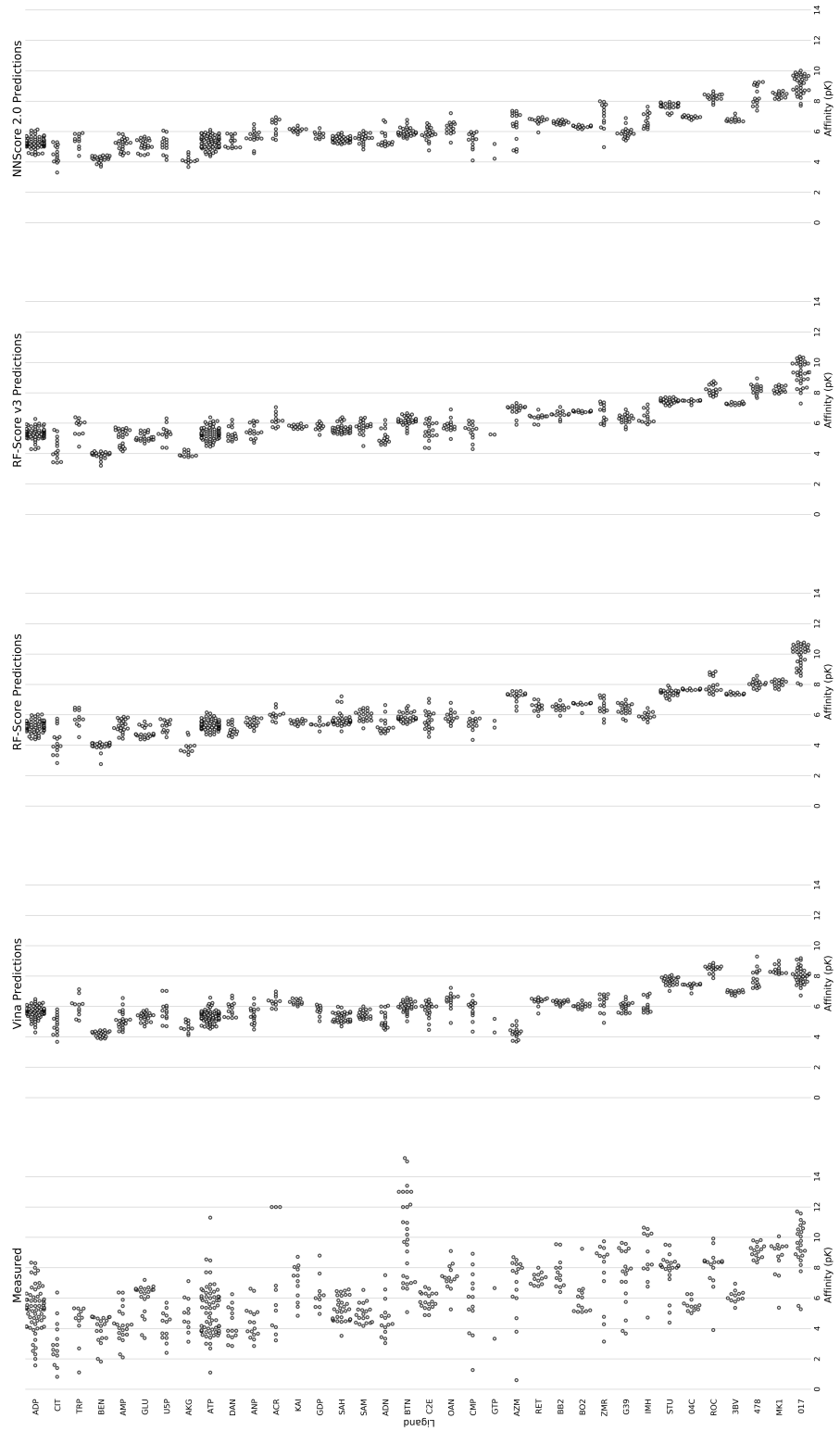


Figure A.7: Experimentally observed pK values and the values predicted by RF models using Vina, RF-Score, RF-Score v3, and NNScore 2.0 features, for sets of multiple structures featuring the same ligand. For each ligand and each set of features, the RF model fails to reproduce the range of experimentally-observed values.

1A08	1A1B	1D09	1F5K	1GPK	1IS0	1KV5	1OLS	1OLU
1V16	1XD1	2H3E	4TIM	4TMN	7CPA	8CPA		

Table A.4: PDB IDs of structures from the PDBbind 2007 core set that were not used due to either inability to generate a conformer for the ligand or Smina docking failure.

1GPK	1JYQ	1KEL	1LOR	1OS0	1VSO	2PQ9	2QFT	2X97
2XY9	2ZCQ	2ZCR	2ZJW	3FK1	3I3B	3MUZ	3VD4	4TMN

Table A.5: PDB IDs of structures from the PDBbind 2013 core set that were not used due to either inability to generate a conformer for the ligand or Smina docking failure.

1BZC	1GPK	1VSO	2ZCQ	2ZCR	3GR2	4K18	4TMN	5TMN

Table A.6: PDB IDs of structures from the PDBbind 2016 core set that were not used due to either inability to generate a conformer for the ligand or Smina docking failure.

References

Ain, Q. U., Mendez-Lucio, O., Ciriano, I. C., Malliavin, T., van Westen, G. J., and Bender, A. Modelling Ligand Selectivity of Serine Proteases Using Integrative Proteochemometric Approaches Improves Model Performance and Allows the Multi-Target Dependent Interpretation of Features. *Integrative Biology*, 6(11):1023–33, 2014.

Alonso, H., Bliznyuk, A. A., and Gready, J. E. Combining Docking and Molecular Dynamic Simulations in Drug Design. *Medicinal Research Reviews*, 26(5):531–568, 2006.

Amaro, R. E., Baron, R., and McCammon, J. A. An Improved Relaxed Complex Scheme for Receptor Flexibility in Computer-Aided Drug Design. *Journal of Computer-Aided Molecular Design*, 22(9):693–705, 2008.

Amaro, R. E., Baudry, J., Chodera, J., Demir, Ö., McCammon, J. A., Miao, Y., and Smith, J. C. Ensemble Docking in Drug Discovery. *Biophysical Journal*, 114(10):2271–2278, 2018.

Anighoro, A. and Bajorath, J. Three-Dimensional Similarity in Molecular Docking: Prioritizing Ligand Poses on the Basis of Experimental Binding Modes. *Journal of Chemical Information and Modeling*, 56(3):580–587, 2016.

Avorn, J. The $2.6 Billion Pill – Methodologic and Policy Considerations. *New England Journal of Medicine*, 372(20):1877–1879, 2015.

Baek, M., Shin, W.-H., Chung, H. W., and Seok, C. GalaxyDock BP2 Score: A Hybrid Scoring Function for Accurate Protein–Ligand Docking. *Journal of Computer-Aided Molecular Design*, 31(7):653–666, 2017.

Ballester, P. J., Schreyer, A., and Blundell, T. L. Does a More Precise Chemical Description of Protein-Ligand Complexes Lead to More Accurate Prediction of Binding Affinity? *Journal of Chemical Information and Modelling*, 54(3):944–955, 2014.

Ballester, P. J. and Mitchell, J. B. O. A Machine Learning Approach to Predicting Protein–Ligand Binding Affinity With Applications to Molecular Docking. *Bioinformatics*, 26(9):1169–1175, 2010.

Barreca, M. L., Iraci, N., De Luca, L., and Chimirri, A. Induced-Fit Docking Approach Provides Insight into the Binding Mode and Mechanism of Action of HIV-1 Integrase Inhibitors. *ChemMedChem: Chemistry Enabling Drug Discovery*, 4(9):1446–1456, 2009.

Bauer, M. R., Ibrahim, T. M., Vogel, S. M., and Boeckler, F. M. Evaluation and Optimization of Virtual Screening Workflows With DEKOIS 2.0 – A Public Library of Challenging Docking Benchmark Sets. *Journal of Chemical Information and Modeling*, 53(6):1447–1462, 2013.

Bergstra, J. and Bengio, Y. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, 2000.

Betzi, S., Suhre, K., Chétrit, B., Guerlesquin, F., and Morelli, X. GFScore: A General Nonlinear Consensus Scoring Function for High-Throughput Docking. *Journal of Chemical Information and Modelling*, 46(4):1704–1712, 2006.

Bleicher, K. H., Böhm, H.-J., Müller, K., and Alanine, A. I. Hit and Lead Generation: Beyond High-Throughput Screening. *Nature Reviews Drug discovery*, 2(5):369, 2003.

Böhm, H.-J. The Computer Program LUDI: A New Method for the de novo Design of Enzyme Inhibitors. *Journal of Computer-Aided Molecular Design*, 6(1):61–78, 1992.

Böhm, H.-J. The Development of a Simple Empirical Scoring Function to Estimate the Binding Constant for a Protein-Ligand Complex of Known Three-Dimensional Structure. *Journal of Computer-Aided Molecular Design*, 8(3):243–256, 1994.

Böhm, H.-J. Prediction of Binding Constants of Protein Ligands: A Fast Method for the Prioritization of Hits Obtained from de novo Design or 3D Database Search Programs. *Journal of Computer-Aided Molecular Design*, 12(4):309–309, 1998.

Boyles, F., Deane, C. M., and Morris, G. M. Learning from the Ligand: Using Ligand-Based Features to Improve Binding Affinity Prediction. *Bioinformatics*, 08 2019. ISSN 1367-4803. btz665.

Breiman, L. Random Forests. *Machine Learning*, 45(1):5–32, 2001.

Carlson, H. A. Check Your Confidence: Size Really *Does* Matter. *Journal of Chemical Information and Modeling*, 53(8):1837–1841, 2013.

Carlson, H. A. Lessons Learned over Four Benchmark Exercises from the Community Structure–Activity Resource. *Journal of Chemical Information and Modeling*, 56(6):951–954, 2016.

Carlson, H. A., Smith, R. D., Damm-Ganamet, K. L., Stuckey, J. A., Ahmed, A., Convery, M. A., Somers, D. O., Kranz, M., Elkins, P. A., Cui, G., Peishoff, C. E., Lambert, M. H., and Dunbar, J. B. CSAR 2014: A Benchmark Exercise Using Unpublished Data from Pharma. *Journal of Chemical Information and Modeling*, 56(6):1063–1077, 2016.

Castleman, P. N., Sears, C. K., Cole, J. A., Baker, D. L., and Parrill, A. L. GPCR Homology Model Template Selection Benchmarking: Global Versus Local Similarity Measures. *Journal of Molecular Graphics and Modelling*, 86:235–246, 2019.

Cereto-Massagué, A., Ojedà, M. J., Valls, C., Mulero, M., Garcia-Vallvé, S., and Pujadas, G. Molecular Fingerprint Similarity Search in Virtual Screening. *Methods*, 71:58–63, 2015.

Changeux, J. P. and Edelstein, S. Conformational Selection or Induced Fit? 50 Years of Debate Resolved. *F1000 Biology Reports*, 3(19), 2011.

Chen, H., Fu, W., Wang, Z., Wang, X., Lei, T., Zhu, F., Li, D., Chang, S., Xu, L., and Hou, T. Reliability of Docking-Based Virtual Screening for GPCR Ligands with Homology Modeled Structures: A Case Study of the Angiotensin II Type I Receptor. *ACS Chemical Neuroscience*, 10 (1):677–689, 2018.

Chen, T. and Guestrin, C. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794. ACM, 2016.

Cheng, T., Li, X., Li, Y., Liu, Z., and Wang, R. Comparative Assessment of Scoring Functions on a Diverse Test Set. *Journal of Chemical Information and Modeling*, 49(4):1079–1093, 2009.

Cheng, T., Li, Q., Zhou, Z., Wang, Y., and Bryant, S. H. Structure-Based Virtual Screening for Drug Discovery: a Problem-Centric Review. *The AAPS Journal*, 14(1):133–141, 2012.

Cheng, Y. C. and Prusoff, W. H. Relationship Between the Inhibition Constant (k_i) and the Concentration of Inhibitor Which Causes 50 per cent Inhibition (I_{50}) of an Enzymatic Reaction. *Biochemical Pharmacology*, 22(23):3099–3108, 1973.

Chodera, J. D., Mobley, D. L., Shirts, M. R., Dixon, R. W., Branson, K., and Pande, V. S. Alchemical Free Energy Methods for Drug Discovery: Progress and Challenges. *Current Opinion in Structural Biology*, 21(2):150–160, 2011.

Cortés-Ciriano, I., Ain, Q. U., Subramanian, V., Lenselink, E. B., Méndez-Lucio, O., IJzerman, A. P., Wohlfahrt, G., Prusis, P., Malliavin, T. E., van Westen, G. J. P., and Bender, A. Polypharmacology Modelling Using Proteochemometrics (PCM): Recent Methodological Developments, Applications to Target Families, and Future Prospects. *MedChemComm*, 6(1):24–50, 2015.

Costanzi, S., Cohen, A., Danfora, A., and Dolatmoradi, M. Influence of the Structural Accuracy of Homology Models on Their Applicability to Docking-Based Virtual Screening: The β_2 Adrenergic Receptor as a Case Study. *Journal of Chemical Information and Modeling*, 2019.

Damm-Ganamet, K. L., Smith, R. D., Dunbar, J. B., Stuckey, J. A., and Carlson, H. A. CSAR Benchmark Exercise 2011–2012: Evaluation of Results from Docking and Relative Ranking of Blinded Congeneric Series. *Journal of Chemical Information and Modeling*, 53(8):1853–1870, 2013.

De Vivo, M., Masetti, M., Bottegoni, G., and Cavalli, A. Role of Molecular Dynamics and Related Methods in Drug Discovery. *Journal of Medicinal Chemistry*, 59(9):4035–4061, 2016.

DeChancie, J. and Houk, K. N. The Origins of Femtomolar ProteinLigand Binding: Hydrogen-Bond Cooperativity and Desolvation Energetics in the Biotin(Strept)Avidin Binding Site. *Journal of the American Chemical Society*, 129(17):5419–5429, 2007.

Deng, W., Breneman, C., and Embrechts, M. J. Predicting ProteinLigand Binding Affinities Using Novel Geometrical Descriptors and Machine-Learning Methods. *Journal of Chemical Information and Computer Sciences*, 44(2):699–703, 2004.

DeWitte, R. S. and Shakhnovich, E. I. SMoG: de Novo Design Method Based on Simple, Fast, and Accurate Free Energy Estimates. 1. Methodology and Supporting Evidence. *Journal of the American Chemical Society*, 118(47):11733–11744, 1996.

DiMasi, J. A., Grabowski, H. G., and Hansen, R. W. Innovation in the Pharmaceutical Industry: New Estimates of R&D Costs. *Journal of Health Economics*, 47:20–33, 2016.

Ding, F., Yin, S., and Dokholyan, N. V. Rapid Flexible Docking Using a Stochastic Rotamer library of ligands. *Journal of Chemical Information and Modeling*, 50(9):1623–1632, 2010.

Drwal, M. N. and Griffith, R. Combination of Ligand-and Structure-Based Methods in Virtual Screening. *Drug Discovery Today: Technologies*, 10(3):e395–e401, 2013.

Dunbar, J. B., Smith, R. D., Yang, C.-Y., Ung, P. M.-U., Lexa, K. W., Khazanov, N. A., Stuckey, J. A., Wang, S., and Carlson, H. A. CSAR Benchmark Exercise of 2010: Selection of the Protein–Ligand Complexes. *Journal of Chemical Information and Modeling*, 51(9):2036–2046, 2011.

Dunbar, J. B., Smith, R. D., Damm-Ganamet, K. L., Ahmed, A., Esposito, E. X., Delproposto, J., Chinnaswamy, K., Kang, Y.-N., Kubish, G., Gestwicki, J. E., Stuckey, J. A., and Carlson, H. A.

CSAR Data Set Release 2012: Ligands, Affinities, Complexes, and Docking Decoys. *Journal of Chemical Information and Modeling*, 53(8):1842–1852, 2013.

Dunitz, J. D. Win Some, Lose Some: Enthalpy-Entropy Compensation in Weak Intermolecular Interactions. *Chemistry & Biology*, 2(11):709–712, 1995.

Durrant, J. D. and McCammon, J. A. NNScore: A Neural-Network-Based Scoring Function for the Characterization of ProteinLigand Complexes. *Journal of Chemical Information and Modeling*, 50(10):1865–1871, 2010.

Durrant, J. D. and McCammon, J. A. BINANA: A Novel Algorithm for Ligand-Binding Characterization. *Journal of Molecular Graphics and Modelling*, 29(6):888–893, 2011a.

Durrant, J. D. and McCammon, J. A. NNScore 2.0: A Neural-Network Receptor–Ligand Scoring Function. *Journal of Chemical Information and Modeling*, 51(11):2897–2903, 2011b.

Durrant, J. D. and McCammon, J. A. Molecular Dynamics Simulations and Drug Discovery. *BMC biology*, 9(1):71, 2011c.

Eldridge, M. D., Murray, C. W., Auton, T. R., Paolini, G. V., and Mee, R. P. Empirical Scoring Functions: I. The Development of a Fast Empirical Scoring Function to Estimate the Binding Affinity of Ligands in Receptor Complexes. *Journal of Computer-Aided Molecular Design*, 11(5): 425–445, 1997.

Ferrara, P. and Jacoby, E. Evaluation of the Utility of Homology Models in High Throughput Docking. *Journal of Molecular Modeling*, 13(8):897–905, Aug 2007.

Fischer, E. Einfluss der Configuration auf die Wirkung der Enzyme. *European Journal of Inorganic Chemistry*, 27(3):2985–2993, 1894.

Floresta, G., Amata, E., Barbaraci, C., Gentile, D., Turnaturi, R., Marrazzo, A., and Rescifina, A. A Structure-And Ligand-Based Virtual Screening of a Database of "Small" Marine Natural Products for the Identification of "Blue" Sigma-2 Receptor Ligands. *Marine Drugs*, 16(10):384, 2018.

Freire, E. Do Enthalpy and Entropy Distinguish First in Class from Best in Class? *Drug Discovery Today*, 13(19-20):869–874, 2008.

Friesner, R. A., Banks, J. L., Murphy, R. B., Halgren, T. A., Klicic, J. J., Mainz, D. T., Repasky, M. P., Knoll, E. H., Shelley, M., Perry, J. K., Shaw, D. E., Francis, P., and Shenkin, P. S. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 1. Method and Assessment of Docking Accuracy. *Journal of Medicinal Chemistry*, 47(7):1739–1749, 2004.

Friesner, R. A., Murphy, R. B., Repasky, M. P., Frye, L. L., Greenwood, J. R., Halgren, T. A., Sanschagrin, P. C., and Mainz, D. T. Extra Precision Glide: Docking and Scoring Incorporating a Model of Hydrophobic Enclosure for Protein- Ligand Complexes. *Journal of Medicinal Chemistry*, 49(21):6177–6196, 2006.

Gabel, J., Desaphy, J., and Rognan, D. Beware of Machine Learning-Based Scoring Functions—On the Danger of Developing Black Boxes. *Journal of Chemical Information and Modeling*, 54(10):2807–2815, 2014.

Gaieb, Z., Liu, S., Gathiaka, S., Chiu, M., Yang, H., Shao, C., Feher, V. A., Walters, W. P., Kuhn, B., Rudolph, M. G., Burley, S. K., Gilson, M. K., and Amaro, R. E. D3R Grand Challenge 2: Blind Prediction of Protein–Ligand Poses, Affinity Rankings, and Relative Binding Free Energies. *Journal of Computer-Aided Molecular Design*, 32(1):1–20, 2018.

Gaieb, Z., Parks, C. D., Chiu, M., Yang, H., Shao, C., Walters, W. P., Lambert, M. H., Nevins, N., Bembenek, S. D., Ameriks, M. K., Mirzadegan, T., Burley, S. K., Amaro, R. E., and Gilson, M. K. D3R Grand Challenge 3: Blind Prediction of Protein–Ligand Poses and Affinity Rankings. *Journal of Computer-Aided Molecular Design*, 33(1):1–18, 2019.

Gasteiger, J. and Marsili, M. Iterative Partial Equalization of Orbital Electronegativity—a Rapid Access to Atomic Charges. *Tetrahedron*, 36(22):3219–3228, 1980.

Gathiaka, S., Liu, S., Chiu, M., Yang, H., Stuckey, J. A., Kang, Y. N., Delproposto, J., Kubish, G., Dunbar, J. B., Carlson, H. A., Burley, S. K., Walters, W. P., Amaro, R. E., Feher, V. A., and Gilson, M. K. D3R Grand Challenge 2015: Evaluation of Protein–Ligand Pose and Affinity Predictions. *Journal of Computer-Aided Molecular Design*, 30(9):651–668, 2016.

Gaulton, A., Hersey, A., Nowotka, M., Bento, A. P., Chambers, J., Mendez, D., Mutowo, P., Atkinson, F., Bellis, L. J., Cibrián-Uhalte, E., Davies, M., Dedman, N., Karlsson, A., Magariños, M. P., Overington, J. P., Papadatos, G., Smit, I., and Leach, A. R. The ChEMBL Database in 2017. *Nucleic Acids Research*, 45(D1):D945–D954, 2017.

Gedeck, P., Rohde, B., and Bartels, C. QSAR- How Good Is It in Practice? Comparison of Descriptor Sets on an Unbiased Cross Section of Corporate Data Sets. *Journal of Chemical Information and Modeling*, 46(5):1924–1936, 2006.

Genheden, S. and Ryde, U. The MM/PBSA and MM/GBSA Methods to Estimate Ligand-Binding Affinities. *Expert Opinion on Drug Discovery*, 10(5):449–461, 2015.

Geschwindner, S., Ulander, J., and Johansson, P. Ligand Binding Thermodynamics in Drug Discovery: Still a Hot Tip? *Journal of Medicinal Chemistry*, 58(16):6321–6335, 2015.

Ghose, A. K., Viswanadhan, V. N., and Wendoloski, J. J. A Knowledge-Based Approach in Designing Combinatorial or Medicinal Chemistry Libraries for Drug Discovery. 1. A Qualitative and Quantitative Characterization of Known Drug Databases. *Journal of Combinatorial Chemistry*, 1(1):55–68, 1999.

Gianni, S., Dogan, J., and Jemth, P. Distinguishing Induced Fit From Conformational Selection. *Biophysical Chehmistry*, 189:33–39, 2014.

Gilson, M. K. and Zhou, H. X. Calculation of Protein-Ligand Binding Affinities. *Annual Review of Biophysics and Biomolecular Structure*, 36:21–42, 2007.

Gilson, M. K., Given, J. A., and Head, M. S. A New Class of Models for Computing Receptor-Ligand Binding Affinities. *Chemistry & Biology*, 4(2):87–92, 1997.

Gilson, M. K., Liu, T., Baitaluk, M., Nicola, G., Hwang, L., and Chong, J. BindingDB in 2015: A Public Database for Medicinal Chemistry, Computational Chemistry and Systems Pharmacology. *Nucleic Acids Research*, 44(D1):D1045–D1053, 2015.

Gohlke, H. and Klebe, G. Statistical Potentials and Scoring Functions Applied to Protein–Ligand Binding. *Current Opinion in Structural Biology*, 11(2):231–235, 2001.

Gohlke, H., Hendlich, M., and Klebe, G. Knowledge-Based Scoring Function to Predict Protein-Ligand Interactions. *Journal of Molecular Biology*, 295(2):337–356, 2000.

Gomes, J., Ramsundar, B., Feinberg, E. N., and Pande, V. S. Atomic Convolutional Networks for Predicting Protein-Ligand Binding Affinity, 2017.

Goodsell, D. S. and Olson, A. J. Automated Docking of Substrates to Proteins by Simulated Annealing. *Proteins: Structure, Function, and Bioinformatics*, 8(3):195–202, 1990.

Guvench, O. and MacKerell Jr, A. D. Computational Evaluation of Protein–Small Molecule Binding. *Current Opinion in Structural Biology*, 19(1):56–61, 2009.

Hall, L. H. and Kier, L. B. The Molecular Connectivity Chi Indexes and Kappa Shape Indexes in Structure-Property Modeling. *Reviews in Computational Chemistry*, pages 367–422, 1991.

Hammes, G. G., Chang, Y.-C., and Oas, T. G. Conformational Selection or Induced Fit: A Flux Description of Reaction Mechanism. *Proceedings of the National Academy of Sciences*, 106(33): 13737–13741, 2009.

Hauser, A. S., Attwood, M. M., Rask-Andersen, M., Schiöth, H. B., and Gloriam, D. E. Trends in GPCR Drug Discovery: New Agents, Targets and Indications. *Nature Reviews Drug Discovery*, 16(12):829, 2017.

Hinselmann, G., Rosenbaum, L., Jahn, A., Fechner, N., Ostermann, C., and Zell, A. Large-Scale Learning of Structure-Activity Relationships Using a Linear Support Vector Machine and Problem-Specific Metrics. *Journal of Chemical Information and Modelling*, 51(2):203–213, 2011.

Ho, T. K. Random Decision Forest. In *Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, 14-16 August 1995*, pages 278–282, 1995.

Ho, T. K. The Random Subspace Method for Constructing Decision Forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.

Hollingsworth, S. A. and Dror, R. O. Molecular Dynamics Simulation for All. *Neuron*, 99(6): 1129–1143, 2018.

Houston, D. R., Yen, L.-H., Pettit, S., and Walkinshaw, M. D. Structure-and Ligand-Based Virtual Screening Identifies New Scaffolds for Inhibitors of the Oncoprotein MDM2. *PLoS One*, 10(4): e0121424, 2015.

Huang, D. and Caflisch, A. The Free Energy Landscape of Small Molecule Unbinding. *PLoS Computational Biology*, 7(2):e1002002, 2011a.

Huang, D. and Caflisch, A. Small Molecule Binding to Proteins: Affinity and Binding/Unbinding Dynamics From Atomistic Simulations. *ChemMedChem*, 6(9):1578–1580, 2011b.

Huang, N., Kalyanaraman, C., Bernacki, K., and Jacobson, M. P. Molecular Mechanics Methods for Predicting Protein–Ligand Binding. *Physical Chemistry Chemical Physics*, 8(44):5166–5177, 2006a.

Huang, N., Kalyanaraman, C., Irwin, J. J., and Jacobson, M. P. Physics-Based Scoring of Protein-Ligand Complexes: Enrichment of Known Inhibitors in Large-Scale Virtual Screening. *Journal of Chemical Information and Modeling*, 46(1):243–253, 2006b.

Huang, N., Shoichet, B. K., and Irwin, J. J. Benchmarking Sets for Molecular Docking. *Journal of Medicinal Chemistry*, 49(23):6789–6801, 2006c.

Huang, S.-Y. and Zou, X. Advances and Challenges in Protein-Ligand Docking. *International Journal of Molecular Sciences*, 11(8):3016–3034, 2010.

Imrie, F., Bradley, A. R., van der Schaar, M., and Deane, C. M. Protein Family-Specific Models Using Deep Neural Networks and Transfer Learning Improve Virtual Screening and Highlight the Need for More Data. *Journal of Chemical Information and Modeling*, 58(11):2319–2330, 2018.

Jiménez, J., Skalic, M., Martinez-Rosell, G., and De Fabritiis, G. K_{DEEP}: Protein–Ligand Absolute Binding Affinity Prediction via 3D-Convolutional Neural Networks. *Journal of Chemical Information and Modeling*, 58(2):287–296, 2018.

Jones, E., Oliphant, T., and and others, P. P. SciPy: Open Source Scientific Tools for Python, 2001. URL `http://www.scipy.org/`. [Online; last accessed 15/10/2019].

Jorgensen, W. L. Efficient Drug Lead Discovery and Optimization. *Accounts of Chemical Research*, 42(6):724–733, 2009.

Jorgensen, W. L. and Thomas, L. L. Perspective on Free-Energy Perturbation Calculations for Chemical Equilibria. *Journal of Chemical Theory and Computation*, 4(6):869–876, 2008.

Jorissen, R. N. and Gilson, M. K. Virtual Screening of Molecular Databases Using a Support Vector Machine. *Journal of Chemical Information and Modelling*, 45(3):549–561, 2005.

Joshi, P., McCann, G. J., Sonawane, V. R., Vishwakarma, R. A., Chaudhuri, B., and Bharate, S. B. Identification of Potent and Selective CYP1A1 Inhibitors via Combined Ligand and Structure-Based Virtual Screening and Their in Vitro Validation in Sacchrosomes and Live Human Cells. *Journal of Chemical Information and Modeling*, 57(6):1309–1320, 2017.

Kalliokoski, T., Kramer, C., Vulpetti, A., and Gedeck, P. Comparability of Mixed IC50 Data–A Statistical Analysis. *PloS one*, 8(4):e61007, 2013.

Karplus, M. and McCammon, J. A. Molecular Dynamics Simulations of Biomolecules. *Nature Structural & Molecular Biology*, 9(9):646, 2002.

Kitchen, D. B., Decornez, H., Furr, J. R., and Bajorath, J. Docking and Scoring in Virtual Screening for Drug Discovery: Methods and Applications. *Nature Reviews Drug discovery*, 3(11):935, 2004.

Kleinberg, E. M. An Overtraining-Resistant Stochastic Modeling Method for Pattern Recognition. *The Annals of Statistics*, 24(6):2319–2349, 1996.

Kleinberg, E. M. On the Algorithmic Implementation of Stochastic Discrimination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(5):473–490, 2000.

Koebel, M. R., Schmadeke, G., Posner, R. G., and Sirimulla, S. AutoDock VinaXB: Implementation of XBSF, New Empirical Halogen Bond Scoring Function, Into AutoDock Vina. *Journal of Cheminformatics*, 8(1):27, 2016.

Koes, D. R., Baumgartner, M. P., and Camacho, C. J. Lessons Learned in Empirical Scoring with Smina from the CSAR 2011 Benchmarking Exercise. *Journal of Chemical Information and Modeling*, 53(8):1893–1904, 2013.

Koshland, D. E. Application of a Theory of Enzyme Specificity to Protein Synthesis. *Proceedings of the National Academy of Sciences*, 44(2):98–104, 1958.

Kramer, C. and Gedeck, P. Leave-Cluster-Out Cross-Validation Is Appropriate for Scoring Functions Derived from Diverse Protein Data Sets. *Journal of Chemical Information and Modeling*, 50 (11):1961–1969, 2010.

Kramer, C., Kalliokoski, T., Gedeck, P., and Vulpetti, A. The Experimental Uncertainty of Heterogeneous Public Ki Data. *Journal of Medicinal Chemistry*, 55(11):5165–5173, 2012.

Krüger, D. M. and Evers, A. Comparison of Structure-and Ligand-Based Virtual Screening Protocols Considering Hit List Complementarity and Enrichment Factors. *ChemMedChem: Chemistry Enabling Drug Discovery*, 5(1):148–158, 2010.

Kubinyi, H. QSAR and 3D QSAR in Drug Design Part 1: Methodology. *Drug discovery today*, 2 (11):457–467, 1997a.

Kubinyi, H. QSAR and 3D QSAR in Drug Design Part 2: Applications and Problems. *Drug Discovery Today*, 2(12):538–546, 1997b.

Kuntz, I. D., Chen, K., Sharp, K. A., and Kollman, P. A. The Maximal Affinity of Ligands. *Proceedings of the National Academy of Sciences*, 96(18):9997–10002, 1999.

Kuntz, I. D., Blaney, J. M., Oatley, S. J., Langridge, R., and Ferrin, T. E. A Geometric Approach to Macromolecule-Ligand Interactions. *Journal of Molecular Biology*, 161(2):269–288, 1982.

Laederach, A. and Reilly, P. J. Specific Empirical Free Energy Function for Automated Docking of Carbohydrates to Proteins. *Journal of Computational Chemistry*, 24(14):1748–1757, 2003.

Lafont, V., Armstrong, A. A., Ohtaka, H., Kiso, Y., Mario Amzel, L., and Freire, E. Compensating Enthalpic and Entropic Changes Hinder Binding Affinity Optimization. *Chemical Biology & Drug Design*, 69(6):413–422, 2007.

Landrum, G. RDKit: Open-Source Cheminformatics, n.d.a. URL `http://www.rdkit.org`. Accessed 20/07/2018.

Landrum, G. Getting Started with the RDKit in Python, n.d.b. URL `http://www.rdkit.org/docs/GettingStartedInPython.html`. Accessed 20/07/2018.

Lapinsh, M., Prusis, P., Gutcaits, A., Lundstedt, T., and Wikberg, J. E. Development of Proteo-Chemometrics: A Novel Technology for the Analysis of Drug-Receptor Interactions. *Biochimica et Biophysica Acta (BBA)-General Subjects*, 1525(1-2):180–190, 2001.

Lavecchia, A. and Di Giovanni, C. Virtual Screening Strategies in Drug Discovery: A Critical Review. *Current Medicinal Chemistry*, 20(23):2839–2860, 2013.

Lazareno, S. and Birdsall, N. J. M. Estimation of Competitive Antagonist Affinity from Functional Inhibition Curves Using the Gaddum, Schild and Cheng-Prusoff Equations. *British Journal of Pharmacology*, 109(4):1110–1119, 1993.

Lenselink, E. B., Louvel, J., Forti, A. F., van Veldhoven, J. P., de Vries, H., Mulder-Krieger, T., McRobb, F. M., Negri, A., Goose, J., Abel, R., et al. Predicting Binding Affinities for GPCR Ligands Using Free-Energy Perturbation. *ACS Omega*, 1(2):293–304, 2016.

Lexa, K. W. and Carlson, H. A. Protein Flexibility in Docking and Surface Mapping. *Quarterly Reviews of Biophysics*, 45(3):301–343, 2012.

Li, G.-B., Yang, L.-L., Wang, W.-J., Li, L.-L., and Yang, S.-Y. ID-Score: A New Empirical Scoring Function Based on a Comprehensive Set of Descriptors Related to Protein–Ligand Interactions. *Journal of Chemical Information and Modeling*, 53(3):592–600, 2013.

Li, H., Leung, K.-S., Wong, M.-H., and Ballester, P. J. Substituting Random Forest for Multiple Linear Regression Improves Binding Affinity Prediction of Scoring Functions: Cyscore as a Case Study. *BMC Bioinformatics*, 15(1):291, 2014a.

Li, H., Leung, K.-S., Wong, M.-H., and Ballester, P. Low-Quality Structural and Interaction Data Improves Binding Affinity Prediction via Random Forest. *Molecules*, 20(6):10947–10962, 2015a.

Li, H., Leung, K.-S., Wong, M.-H., and Ballester, P. J. Improving AutoDock Vina Using Random Forest: The Growing Accuracy of Binding Affinity Prediction by the Effective Exploitation of Larger Data Sets. *Molecular Informatics*, 34(2-3):115–126, 2015b.

Li, H., Leung, K.-S., Wong, M.-H., and Ballester, P. J. Correcting the Impact of Docking Pose Generation Error on Binding Affinity Prediction. *BMC Bioinformatics*, 17(11):308, 2016.

Li, H., Peng, J., Leung, Y., Leung, K.-S., Wong, M.-H., Lu, G., and Ballester, P. J. The Impact of Protein Structure and Sequence Similarity on the Accuracy of Machine-Learning Scoring Functions for Binding Affinity Prediction. *Biomolecules*, 8(1):12, 2018.

Li, Y., Han, L., Liu, Z., and Wang, R. Comparative Assessment of Scoring Functions on an Updated Benchmark: 2. Evaluation Methods and General Results. *Journal of Chemical Information and Modeling*, 54(6):1717–36, 2014b.

Li, Y., Liu, Z., Li, J., Han, L., Liu, J., Zhao, Z., and Wang, R. Comparative Assessment of Scoring Functions on an Updated Benchmark: 1. Compilation of the Test Set. *Journal of Chemical Information and Modeling*, 54(6):1700–16, 2014c.

Li, Y. and Yang, J. Structural and Sequence Similarity Makes a Significant Impact on Machine-Learning-Based Scoring Functions for Protein–Ligand Interactions. *Journal of Chemical Information and Modeling*, 57(4):1007–1012, 2017.

Lim, V. J. Y., Du, W., Chen, Y. Z., and Fan, H. A Benchmarking Study on Virtual Ligand Screening Against Homology Models of Human GPCRs. *Proteins: Structure, Function, and Bioinformatics*, 86(9):978–989, 2018.

Lin, J.-H., Perryman, A. L., Schames, J. R., and McCammon, J. A. The Relaxed Complex Method: Accommodating Receptor Flexibility for Drug Design with an Improved Scoring Scheme. *Biopolymers: Original Research on Biomolecules*, 68(1):47–62, 2003.

Lipinski, C. A. Lead-and Drug-Like Compounds: The Rule-of-Five Revolution. *Drug Discovery Today: Technologies*, 1(4):337–341, 2004.

Liu, Z., Su, M., Han, L., Liu, J., Yang, Q., Li, Y., and Wang, R. Forging the Basis for Developing Protein–Ligand Interaction Scoring Functions. *Accounts of Chemical Research*, 50(2):302–309, 2017.

Loo, J. S. E., Emtage, A. L., Ng, K. W., Yong, A. S. J., and Doughty, S. W. Assessing GPCR Homology Models Constructed from Templates of Various Transmembrane Sequence Identities: Binding Mode Prediction and Docking Enrichment. *Journal of Molecular Graphics and Modelling*, 80:38–47, 2018.

Lyne, P. Structure-Based Virtual Screening: An Overview. *Drug Discovery Today*, 7(20):1047–1055, 2002.

Macalino, S. J. Y., Gosu, V., Hong, S., and Choi, S. Role of Computer-Aided Drug Design in Modern Drug Discovery. *Archives of Pharmacal Research*, 38(9):1686–1701, 2015.

Maggiora, G. M. On Outliers and Activity Cliffs Why QSAR Often Disappoints, 2006.

Mahé, P., Ralaivola, L., Stoven, V., and Vert, J. P. The Pharmacophore Kernel for Virtual Screening with Support Vector Machines. *Journal of Chemical Information and Modelling*, 46(5):2003–2014, 2006.

McCammon, J. A., Gelin, B. R., and Karplus, M. Dynamics of Folded Proteins. *Nature*, 267(5612): 585, 1977.

Meiler, J. and Baker, D. ROSETTALIGAND: Protein–Small Molecule Docking with Full Side-Chain Flexibility. *Proteins: Structure, Function, and Bioinformatics*, 65(3):538–548, 2006.

Meng, X.-Y., Zhang, H.-X., Mezei, M., and Cui, M. Molecular Docking: A Powerful Approach for Structure-Based Drug Discovery. *Current Computer-Aided Drug Design*, 7(2):146–157, 2011.

Miyazawa, S. and Jernigan, R. L. Estimation of Effective Interresidue Contact Energies from Protein Crystal Structures: Quasi-Chemical Approximation. *Macromolecules*, 18(3):534–552, 1985.

Mobley, D. L. Let's Get Honest About Sampling. *Journal of Computer-Aided Molecular Design*, 26 (1):93–95, 2012.

Monod, J., Wyman, J., and Changeux, J. P. On the Nature of Allosteric Transitions: A Plausible Model. *Journal Molecular Biology*, 12(1):88–118, 1965.

Mooij, W. T. M. and Verdonk, M. L. General and Targeted Statistical Potentials for Protein–Ligand Interactions. *Proteins: Structure, Function, and Bioinformatics*, 61(2):272–287, 2005.

Morris, G. M., Goodsell, D. S., Halliday, R. S., Huey, R., Hart, W. E., Belew, R. K., and Olson, A. J. Automated Docking Using a Lamarckian Genetic Algorithm and an Empirical Binding Free Energy Function. *Journal of Computational Chemistry*, 19(14):1639–1662, 1998.

Morris, G. M., Huey, R., Lindstrom, W., Sanner, M. F., Belew, R. K., Goodsell, D. S., and Olson, A. J. AutoDock4 and AutoDockTools4: Automated Docking With Selective Receptor Flexibility. *Journal of Computational Chemistry*, 30(16):2785–2791, 2009.

Muegge, I. and Martin, Y. C. A General and Fast Scoring Function for ProteinLigand Interactions: A Simplified Potential Approach. *Journal of Medicinal Chemistry*, 42(5):791–804, 1999.

Muhammed, M. T. and Aki-Yalcin, E. Homology Modeling in Drug Discovery: Overview, Current Applications, and Future Perspectives. *Chemical Biology & Drug Design*, 93(1):12–20, 2019.

Myers, S. and Baker, A. Drug Discovery—an Operating Model for a New Era. *Nature Biotechnology*, 19(8):727, 2001.

Mysinger, M. M., Carchia, M., Irwin, J. J., and Shoichet, B. K. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. *Journal of Medicinal Chemistry*, 55(14):6582–6594, 2012.

Nabuurs, S. B., Wagener, M., and De Vlieg, J. A Flexible Approach to Induced Fit Docking. *Journal of Medicinal Chemistry*, 50(26):6507–6518, 2007.

O'Boyle, N. M., Banck, M., James, C. A., Morley, C., Vandermeersch, T., and Hutchison, G. R. Open Babel: An Open Chemical Toolbox. *Journal of Cheminformatics*, 3(1):33, 2011.

Ou-Yang, S.-s., Lu, J.-y., Kong, X.-q., Liang, Z.-j., Luo, C., and Jiang, H. Computational Drug Discovery. *Acta Pharmacologica Sinica*, 33(9):1131, 2012.

Ozrin, V. and Subbotin, S., M.V.and Nikitin. PLASS: Protein-Ligand Affinity Statistical Score – A Knowledge-Based Force-Field Model of Interaction Derived from the PDB. *Journal of Computer-Aided Molecular Design*, 18(4):261–270, 2004.

Pan, Z., Wang, Y., Gu, X., Wang, J., and Cheng, M. Refined Homology Model of Cytochrome bcc Complex B Subunit for Virtual Screening of Potential Anti-Tuberculosis Agents. *Journal of Biomolecular Structure and Dynamics*, (just-accepted):1–17, 2019.

Paricharak, S., Cortes-Ciriano, I., AP, I. J., Malliavin, T. E., and Bender, A. Proteochemometric Modelling Coupled to in Silico Target Prediction: An Integrated Approach for the Simultaneous Prediction of Polypharmacology and Binding Affinity/Potency of Small Molecules. *Journal of Chemoinformatics*, 7:15, 2015.

Paul, S. M., Mytelka, D. S., Dunwiddie, C. T., Persinger, C. C., Munos, B. H., Lindborg, S. R., and Schacht, A. L. How to Improve R&D Productivity: the Pharmaceutical Industry's Grand Challenge. *Nature reviews Drug discovery*, 9(3):203, 2010.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Qian, H. Entropy-Enthalpy Compensation: Conformational Fluctuation and Induced-Fit. *The Journal of Chemical Physics*, 109(22):10015–10017, 1998.

Qiu, T., Qiu, J., Feng, J., Wu, D., Yang, Y., Tang, K., Cao, Z., and Zhu, R. The Recent Progress in Proteochemometric Modelling: Focusing on Target Descriptors, Cross-Term Descriptors and Application Scope. *Briefings in Bioinformatics*, 18(1):125–136, 2016.

Ragoza, M., Hochuli, J., Idrobo, E., Sunseri, J., and Koes, D. R. Protein-Ligand Scoring with Convolutional Neural Networks. *Journal of Chemical Information and Modelling*, 57(4):942–957, 2017a.

Ragoza, M., Turner, L., and Koes, D. R. Ligand Pose Optimization with Atomic Grid-Based Convolutional Neural Networks. *arXiv preprint arXiv:1710.07400*, 2017b.

Ramírez, D. and Caballero, J. Is It Reliable to Take the Molecular Docking Top Scoring Position as the Best Solution without Considering Available Structural Data? *Molecules*, 23(5):1038, 2018.

Reynolds, C. H. and Holloway, M. K. Thermodynamics of Ligand Binding and Efficiency. *ACS Medicinal Chemistry Letters*, 2(6):433–437, 2011.

Riniker, S. and Landrum, G. A. Better Informed Distance Geometry: Using What We Know To Improve Conformation Generation. *Journal of Chemical Information and Modeling*, 55(12): 2562–2574, 2015.

Rohrer, S. G. and Baumann, K. Maximum Unbiased Validation (MUV) Data Sets for Virtual Screening Based on PubChem Bioactivity Data. *Journal of Chemical Information and Modelling*, 49(2):169–184, 2009.

Rosenfeld, R. J., Garcin, E. D., Panda, K., Andersson, G., Åberg, A., Wallace, A. V., Morris, G. M., Olson, A. J., Stuehr, D. J., Tainer, J. A., and Getzoff, E. D. Conformational Changes in Nitric Oxide Synthases Induced by Chlorzoxazone and Nitroindazoles: Crystallographic and Computational Analyses of Inhibitor Potency. *Biochemistry*, 41(47):13915–13925, 2002.

Ross, G. A., Morris, G. M., and Biggin, P. C. One Size Does Not Fit All: The Limits of Structure-Based Models in Drug Discovery. *Journal of Chemical Theory and Computation*, 9(9):4266–4274, 2013.

Rubin, M. M. and Changeux, J. P. On the Nature of Allosteric Transitions: Implications of Non-Exclusive Ligand Binding. *Journal of Molecular Biology*, 21(2):265–274, 1966.

Salmaso, V. and Moro, S. Bridging Molecular Docking to Molecular Dynamics in Exploring Ligand-Protein Recognition Process: An Overview. *Frontiers in Pharmacology*, 9, 2018.

Santos-Martins, D., Forli, S., Ramos, M. J., and Olson, A. J. AutoDock4ZN: An Improved AutoDock Force Field for Small-Molecule Docking to Zinc Metalloproteins. *Journal of Chemical Information and Modelling*, 54(8):2371–2379, 2014.

Schapire, R. E. The Strength of Weak Learnability. *Machine Learning*, 5(2):197–227, 1990.

Schneider, G. Virtual Screening: An Endless Staircase? *Nature Reviews Drug Discovery*, 9(4):273, 2010.

Sharp, K. Entropy—Enthalpy Compensation: Fact or Artifact? *Protein Science*, 10(3):661–667, 2001.

Shekhar, C. In Silico Pharmacology: Computer-Aided Methods Could Transform Drug Development. *Chemistry & Biology*, 15(5):413–414, 2008.

Sieg, J., Flachsenberg, F., and Rarey, M. In Need of Bias Control: Evaluating Chemical Data for Machine Learning in Structure-Based Virtual Screening. *Journal of Chemical Information and Modeling*, 59(3):947–961, 2019.

Sippl, M. J. Calculation of Conformational Ensembles from Potentials of Mean Force: An Approach to the Knowledge-Based Prediction of Local Structures in Globular Proteins. *Journal of Molecular Biology*, 213(4):859–883, 1990.

Sliwoski, G., Kothiwale, S., Meiler, J., and Lowe, E. W. Computational Methods in Drug Discovery. *Pharmacological Reviews*, 66(1):334–395, 2014.

Smith, J. C. and Roux, B. Eppur Si Muove! The 2013 Nobel Prize in Chemistry. *Structure*, 21(12): 2102–2105, 2013.

Smith, R. D., Dunbar, J. B., Ung, P. M.-U., Esposito, E. X., Yang, C.-Y., Wang, S., and Carlson, H. A. CSAR Benchmark Exercise of 2010: Combined Evaluation Across All Submitted Scoring Functions. *Journal of Chemical Information and Modeling*, 51(9):2115–2131, 2011.

Smith, R. D., Damm-Ganamet, K. L., Dunbar, J. B., Ahmed, A., Chinnaswamy, K., Delproposto, J. E., Kubish, G. M., Tinberg, C. E., Khare, S. D., Dou, J., Doyle, L., Stuckey, J. A., Baker, D., and Carlson, H. A. CSAR Benchmark Exercise 2013: Evaluation of Results from a Combined Computational Protein Design, Docking, and Scoring/Ranking Challenge. *Journal of Chemical Information and Modeling*, 56(6):1022–1031, 2016.

Sotriffer, C. A., Sanschagrin, P., Matter, H., and Klebe, G. SFCscore: Scoring Functions for Affinity Prediction of Protein–Ligand Complexes. *Proteins: Structure, Function, and Bioinformatics*, 73(2):395–419, 2008.

Starikov, E. B. and Nordén, B. Enthalpy-Entropy Compensation: A Phantom or Something Useful? *The Journal of Physical Chemistry B*, 111(51):14431–14435, 2007.

Stepniewska-Dziubinska, M. M., Zielenkiewicz, P., and Siedlecki, P. Development and Evaluation of a Deep Learning Model for Protein–Ligand Binding Affinity Prediction. *Bioinformatics*, 34(21):3666–3674, 2018.

Su, M., Yang, Q., Du, Y., Feng, G., Liu, Z., Li, Y., and Wang, R. Comparative Assessment of Scoring Functions: The CASF-2016 Update. *Journal of Chemical Information and Modeling*, 2018.

Sun, H. Pharmacophore-Based Virtual Screening. *Current medicinal chemistry*, 15(10):1018–1024, 2008.

Svensson, F., Karlén, A., and Sköld, C. Virtual Screening Data Fusion Using Both Structure-And Ligand-Based Methods. *Journal of Chemical Information and Modeling*, 52(1):225–232, 2011.

Svetnik, V., Liaw, A., Tong, C., and Wang, T. In *International Workshop on Multiple Classifier Systems*. Springer, 2004.

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., and Feuston, B. P. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. *Journal of Chemical Information and Computer Sciences*, 43(6):1947–1958, 2003.

Swann, S. L., Brown, S. P., Muchmore, S. W., Patel, H., Merta, P., Locklear, J., and Hajduk, P. J. A Unified, Probabilistic Framework for Structure-And Ligand-Based Virtual Screening. *Journal of Medicinal Chemistry*, 54(5):1223–1232, 2011.

Tan, L., Geppert, H., Sisay, M. T., Gütschow, M., and Bajorath, J. Integrating Structure-And Ligand-Based Virtual Screening: Comparison of Individual, Parallel, and Fused Molecular Docking and Similarity Search Calculations on Multiple Targets. *ChemMedChem: Chemistry Enabling Drug Discovery*, 3(10):1566–1571, 2008.

Taylor, D. The Pharmaceutical Industry and the Future of Drug Development. 2015.

Totrov, M. and Abagyan, R. Flexible Ligand Docking to Multiple Receptor Conformations: A Practical Alternative. *Current Opinion in Structural Biology*, 18(2):178–184, 2008.

Trott, O. and Olson, A. J. AutoDock Vina: Improving the Speed and Accuracy of Docking with a New Scoring Function, Efficient Optimization, and Multithreading. *Journal of Computational Chemistry*, 31(2):455–461, 2010.

van Westen, G. J., Wegner, J. K., IJzerman, A. P., van Vlijmen, H. W. T., and Bender, A. Proteochemometric Modeling as a Tool to Design Selective Compounds and for Extrapolating to Novel Targets. *MedChemComm*, 2(1):16–30, 2011.

Walters, W. P., Stahl, M. T., and Murcko, M. A. Virtual Ccreening – An Overview. *Drug Discovery Today*, 3(4):160–178, 1998.

Wang, C., Greene, D., Xiao, L., Qi, R., and Luo, R. Recent Developments and Applications of the MMPBSA Method. *Frontiers in Molecular Biosciences*, 4:87, 2018.

Wang, C. and Zhang, Y. Improving Scoring-Docking-Screening Powers of Protein–Ligand Scoring Functions Using Random Forest. *Journal of Computational Chemistry*, 38(3):169–177, 2017.

Wang, L., Wu, Y., Deng, Y., Kim, B., Pierce, L., Krilov, G., Lupyan, D., Robinson, S., Dahlgren, M. K., Greenwood, J., Romero, D. L., Masse, C., Knight, J. L., Steinbrecher, T., Beuming, T., Damm, W., Harder, E., Sherman, W., Brewer, M., Wester, R., Murcko, M., Frye, L., Farid, R., Lin, T., Mobley, D. L., Jorgensen, W. L., Berne, B. J., Friesner, R. A., and Abel, R. Accurate and Reliable Prediction of Relative Ligand Binding Potency in Prospective Drug Discovery by Way of a Modern Free-Energy Calculation Protocol and Force Field. *Journal of the American Chemical Society*, 137(7):2695–2703, 2015.

Wang, R., Fang, X., Lu, Y., and Wang, S. The PDBbind Database: Collections of Binding Affinities for Protein-Ligand Complexes with Known Three-Dimensional Structures. *Journal of Medicinal Chemistry*, 47(12):2977–2980, 2004.

Waring, M. J., Arrowsmith, J., Leach, A. R., Leeson, P. D., Mandrell, S., Owen, R. M., Pairaudeau, G., Pennie, W. D., Pickett, S. D., Wang, J., et al. An Analysis of the Attrition of Drug Candidates from Four Major Pharmaceutical Companies. *Nature Reviews Drug Discovery*, 14(7):475, 2015.

Wildman, S. A. and Crippen, G. M. Prediction of Physicochemical Parameters by Atomic Contributions. *Journal of Chemical Information and Computer Sciences*, 39(5):868–873, 1999.

Willett, P. Chemical Similarity Searching. *Journal of Chemical Information and Modelling*, 36(6): 983–996, 1998.

Williams-Noonan, B. J., Yuriev, E., and Chalmers, D. K. Free Energy Methods in Drug Design: Prospects of "Alchemical Perturbation" in Medicinal Chemistry: Miniperspective. *Journal of Medicinal Chemistry*, 61(3):638–649, 2017.

Wójcikowski, M., Zielenkiewicz, P., and Siedlecki, P. Open Drug Discovery Toolkit (ODDT): A New Open-Source Player in the Drug Discovery Field. *Journal of Cheminformatics*, 7(1):26, 2015.

Wójcikowski, M., Ballester, P. J., and Siedlecki, P. Performance of Machine-Learning Scoring Functions in Structure-Based Virtual Screening. *Scientific Reports*, 7:46710, 2017.

Wójcikowski, M., Kukiełka, M., Stepniewska-Dziubinska, M. M., and Siedlecki, P. Development of a Protein–Ligand Extended Connectivity (PLEC) Fingerprint and Its Application for Binding Affinity Predictions. *Bioinformatics*, 2018.

Yang, Q., Su, M., Li, Y., and Wang, R. Revisiting the Relationship Between Correlation Coefficient, Confidence Level, and Sample Size. *Journal of Chemical Information and Modeling*, 2019.

Yuriev, E., Holien, J., and Ramsland, P. A. Improvements, Trends, and New Ideas in Molecular Docking: 2012–2013 in Review. *Journal of Molecular Recognition*, 28(10):581–604, 2015.

Zhang, S., Golbraikh, A., and Tropsha, A. Development of Quantitative StructureBinding Affinity Relationship Models Based on Novel Geometrical Chemical Descriptors of the Protein-Ligand Interfaces. *Journal of Medicinal Chemistry*, 49(9):2713–2724, 2006.

Zhu, J., Lv, Y., Han, X., Xu, D., and Han, W. Understanding the Differences of the Ligand Binding/Unbinding Pathways between Phosphorylated and Non-Phosphorylated ARH1 Using Molecular Dynamics Simulations. *Scientific Reports*, 7(1):12439, 2017.

Zilian, D. and Sotriffer, C. A. SFCscore RF: A Random Forest-Based Scoring Function for Improved Affinity Prediction of Protein–Ligand Complexes. *Journal of Chemical Information and Modeling*, 53(8):1923–1933, 2013.

Zimmermann, M. O., Lange, A., and Boeckler, F. M. Evaluating the Potential of Halogen Bonding in Molecular Design: Automated Scaffold Decoration Using the New Scoring Function XBScore. *Journal of Chemical Information and Modeling*, 55(3):687–699, 2015.

Zwanzig, R. W. High-Temperature Equation of State by a Perturbation Method. I. Nonpolar Gases. *The Journal of Chemical Physics*, 22(8):1420–1426, 1954.