Full Paper

www.molinf.com

molecular informatics

DOI: 10.1002/minf.202000196

Prediction of Promiscuity Cliffs Using Machine Learning

Thomas Blaschke,[a] Christian Feldmann,[a] and Jürgen Bajorath*[a]

Abstract: Compounds with the ability to interact with multiple targets, also called promiscuous compounds, provide the basis for polypharmacological drug discovery. In recent years, a plethora of structural analogs with different promiscuity has been identified. Nevertheless, the molecular origins of promiscuity remain to be elucidated. In this study, we systematically extracted different structural analogs with varying promiscuity using the matched molecular pair (MMP) formalism from public biological screening and medicinal chemistry data. Care was taken to eliminate all compounds with potential false-positive activity annotations from the analysis. Promiscuity predictions were then attempted at the level of compound pairs representing promiscuity cliffs (PCs; formed by analogs with large promiscuity differences) and corresponding non-PC MMPs (analog pairs without significant promiscuity differences). To address this prediction task, different machine learning models were generated and the results were compared with single compound predictions. PCs encoding promiscuity differences were found to contain more structure-promiscuity relationship information than sets of individual promiscuous compounds. In addition, feature analysis was carried out revealing key contributions to the correct prediction of PCs and non-PC MMPs via machine learning.

Keywords: multitarget activity · promiscuity · polypharmacology · machine learning · deep learning · structure-promiscuity relationships

1 Introduction

The observation that many drugs bind to multiple biological targets has gained increased attention over the past two decades. These multitarget interactions, also known as compound promiscuity,[1] are the basis of polypharmacology, and major determinants of the efficacy of promiscuous drugs, but also responsible for undesired side effects.[2–6] An increasing number of polypharmacology studies has triggered a shift from the long dominant drug discovery paradigm of single target compound specificity to multitarget activity,[7,8] leading to coexistence of both principles. However, the exploration of promiscuity is challenging from an experimental and a computational perspective. Apparent promiscuity can also be caused by experimental artifacts leading to false positive activity annotations such as assay interference effects. Such unwanted effects include a tendency of liable compounds to aggregate, form non-specific interactions with target proteins, or react in various ways under assay conditions.[9–11] Additionally, when comparing the promiscuity of multiple compounds, apparent differences might be influenced by limited assay overlap or significantly different test frequencies. For example, if two close structural analogs are considered and one has been tested in 100 assays against 90 targets and the other in 10 assays against two targets, observed differences in promiscuity might be largely due to the much higher test frequency and target coverage of one of the analogs. Compound inactivity and test frequencies are typically not reported in the medicinal chemistry literature and are thus not available in ChEMBL,[12] the major public repository of compounds from medicinal chemistry.

Despite these challenges, rationalizing molecular origins of true multitarget activity continues to be of high interest for compound design. Molecular determinants of promiscuity remain to be fully understood. For example, it is largely unclear at present if specific structural characteristics might trigger defined multitarget activity of compounds. Different binding mechanisms of promiscuous compounds are just beginning to be elucidated on the basis of X-ray structures of relevant complexes.[13] Some studies have suggested that promiscuous compounds might often interact with similar binding sites in proteins.[14] These also include correlation analysis of binding site similarity and compound promiscuity.[15,16] However, promiscuous binding events are not restricted to similar binding sites. Other studies have identified a large number of promiscuous compounds interacting with proteins from different families and functional classes.[17] The multiclass compounds displayed a variety of binding modes with often only little shape similarity.[17] Hence, compound promiscuity cannot be

[a] *T. Blaschke, C. Feldmann, J. Bajorath*
Department of Life Science Informatics, B-IT, LIMES Program Unit Chemical Biology and Medicinal Chemistry, Rheinische Friedrich-Wilhelms-Universität, Endenicher Allee 19c, D-53115 Bonn, Germany
E-mail: bajorath@bit.uni-bonn.de

rationalized by assuming the presence of canonical binding modes.

For the systematic analysis of compounds with different levels of multitarget activity, the promiscuity cliff (PC) concept was introduced.[18] A PC is defined as a pair of structural analogs having a large difference in the number of proteins they are active against. Computationally, analog relationships can be established in different ways, for example, by applying the matched molecular pair (MMP) formalism,[19] which has become a preferred way of analog identification and representation. The number of targets a compound is active against is also referred to as the promiscuity degree (PD).[20] In different studies, large numbers of MMP-based PCs have been identified also including PCs formed by extensively tested compounds, thus providing experimental support for the PC concept.[20,21]

PCs can be organized in networks in which nodes represent compounds and edges indicate the formation of pairwise PC relationships.[22] These PC networks are rich in structure-promiscuity relationship information and provide additional target hypotheses for non-promiscuous analogs that can be experimentally assessed. A previous investigation made use of PC networks to assemble data sets of structural analogs that were either promiscuous or non-promiscuous.[23] The resulting data sets consisted of individual compounds with structural relationships across training and test sets. They were then used to train machine learning models to distinguish between promiscuous and non-promiscuous compounds. These models were found to be predictive and provided indirect evidence for the presence of structure-promiscuity relationships. However, the different models did not achieve classification accuracy of more than 75%, indicating that other factors such as data incompleteness or promiscuity pattern-insensitive molecular representations might also influence the calculations.[23]

In this work, we re-focus machine learning analysis and promiscuity prediction from single compounds to PCs capturing promiscuity differences, hence adding another layer of information, but also complexity to the prediction tasks. Correct prediction of MMP-based PCs requires distinguishing between PCs and other MMPs not encoding promiscuity differences. Thus, successful machine learning analysis at the level of compound pairs might provide further evidence for the existence of structure-promiscuity patterns. Therefore, PCs and corresponding non-PC MMPs were systematically assembled from different sources and multiple machine learning and control models were derived to systematically distinguish between them. Feature elimination analysis was carried out to rationalize these predictions. In the following, our analysis and findings are presented and discussed.

2 Materials and Methods

2.1 Compound Selection

For our analysis, two data sources have been used including a comprehensive collection of publicly available human kinase inhibitors from different databases[22] and a large set of intensely assayed compounds extracted from PubChem screening data.[21] For the kinase inhibitor data sets, most activity information originated from ChEMBL. The data set contained 112,624 compounds that were active against a total of 426 human kinases. The composition of the kinase-inhibitor matrix and the kinase distribution over promiscuous inhibitors have been reported previously.[22] Only ~1% of these inhibitors were known to be active against at least 10 kinases. For the PubChem screening data set, unambiguous compound activity annotations were extracted from PubChem biological screening assays.[24] Only consistent qualitative compound assay results designated as 'active' or 'inactive' for human targets were considered. Because promiscuity analysis is particularly vulnerable to false positive assay outcomes,[20] compounds potentially causing assay artifacts were excluded. To detect potential pan-assay interference compounds (PAINS),[25] publicly available filters from ChEMBL,[12] RDKit,[26] and ZINC[27] were used. Additionally, the Aggregator Advisor[11] was used as a filter to exclude compounds that were likely to aggregate under assay conditions.

Furthermore, only PubChem compounds tested in at least 100 screening assays against at least 10 distinct target proteins were selected. To remove redundancies in PubChem target annotations, PubChem GenInfo Identifiers (GI) were mapped to UniProt identifiers (IDs).[28] If a single GI corresponded to multiple UniProt IDs, only a single 'reviewed' UniProt ID was selected. Applying these criteria, a total of 327,898 extensively assayed compounds were obtained that were tested in 1994 assays against 818 unique targets, yielding a total of ~94 million interactions.

For each compound, its PD was obtained by counting the number of targets it was active against. If multiple assays were available for a given target, a compound was required to have a consistent target annotation (e.g., active or inactive in all assays); otherwise, the compound-target annotations were discarded.

Kinase inhibitors and screening compounds were initially classified as promiscuous if they were active against at least 10 different targets. By contrast, compounds with one or no activity annotation were classified as non-promiscuous. All kinase inhibitors were – by definition – active against at least one kinase, while consistently inactive compounds (PD = 0) exclusively originated from screening data. From promiscuous and non-promiscuous compounds, PCs and non-PC MMPs were extracted, as further detailed below.

2.2 Training and Test Sets

From both compound data sources, MMPs were systematically extracted. Single-, dual-, and triple-cut fragmentation of exocyclic single-bonds was carried out using an in-house implementation of the Hussain and Rea algorithm.[19] In MMP fragmentation, only exocyclic bonds are cleaved to ensure integrity of ring structures. In addition, transformation size restrictions were applied to facilitate a meaningful distinction between core structures and substituents.[29] For the assembly of training and test sets, MMP-based PCs, as defined above, and non-PC MMPs were identified with the aid of network representations. Therefore, compounds selected for our analysis were initially organized in MMP networks (nodes: compounds; edges: pairwise MMP relationships). MMP networks were then converted into PC networks (edges: pairwise PC formation) by labeling compounds as promiscuous (PD $\geq$ 10) or non-promiscuous (PD = 1 or 0) and applying ΔPD criteria. In the case of the PubChem data set, the ΔPD for two analogs was calculated exclusively considering shared targets, thereby ensuring highest possible experimental confidence for ΔPD assignments on the basis of screening data. In addition, for the PubChem data set, multiple PC networks were generated on the basis of varying ΔPD criteria to further refine compound pair-based promiscuity analysis.

Specifically, for the PubChem data set, we constructed five PC networks applying different thresholds of ΔPD $\geq$ 10, [9–8], [7–6], [5–4], and [3–2]. Hence, ΔPD criteria for PC formation were gradually relaxed to increase the prediction challenge (distinguishing between PCs and non-PC MMPs). From the resulting PC networks, 900 MMPs were randomly selected. However, no selected compound was permitted to participate in multiple MMPs, hence ascertaining uniqueness of all pairs. Selected MMPs were randomly divided into equally sized training and test sets. In addition, from the MMP network, 900 non-PC MMPs formed by non-promiscuous compounds were selected and also divided into equally sized training and test sets. Thus, for the PubChem data set, a total of five training/test set combinations were obtained. Each combination consisted of 1800 MMPs comprising 3600 unique compounds. For control experiments using single compounds, in each case, 900 promiscuous compounds were randomly sampled from PCs and 900 non-promiscuous compounds from non-PC MMPs and divided into equally sized training and test sets. Hence, the number of pairs and individual compounds used as training and test instances was consistently the same. Single compound predictions served as a reference for pair-based promiscuity predictions.

For the kinase inhibitor data set, the same selection strategy was applied. However, due to the limited amount of available promiscuous inhibitors, only 500 PCs with ΔPD $\geq$ 10 were extracted and 500 non-PC, resulting in a total of 1000 pairs comprising 2000 unique compounds for training and testing.

For hyperparameter optimization of machine learning models, a random 80% vs. 20% training data split was carried out to obtain an additional validation set (20%).

2.3 Molecular Representations

Individual compounds were represented using the extended connectivity fingerprint with bond diameter 4 (ECFP4)[30] and MMPs were represented using an especially designed MMP fingerprint (MMPFP). ECFP4 is a feature set fingerprint that enumerates layered atom environments and encodes them as integers using a hashing function. It produces molecule-dependent feature sets of variable size. Each generated atom environment feature was mapped to a new position, yielding a fingerprint with a constant number of bits. An "unfolded" version of ECFP4 in which each unique feature was mapped to a specific position was obtained by calculating all features occurring in the PubChem data set. This version consisted of a total of 128,742 bits.

MMPFP was generated from the unfolded ECFP4 fingerprint of compounds forming an MMP. Bits shared by the two fingerprints constituted the fingerprint of the common core fragment, termed core fingerprint (CFP). Bits present only in the fingerprint of the individual MMP partners represented the chemical transformation (substitution) and formed substituent 1 fingerprint (S1FP) and substituent 2 fingerprint (S2FP), respectively. For each MMP, the CFP, S1FP, and S2FP components were concatenated, thereby providing a fingerprint representation of an analog pair. The similarity of two MMPFPs was determined as follows:

$$\mathrm{Tc}(\mathrm{MMPFP1}, \mathrm{MMPFP2}) = \mathrm{Tc}([\mathrm{CFP1}, \mathrm{S1FP1}, \mathrm{S2FP1}], [\mathrm{CFP2}, \mathrm{S1FP2}, \mathrm{S2FP2}]) \quad (1)$$

$$\mathrm{Tc}(\mathrm{MMPFP1}, \mathrm{MMPFP2}) = \mathrm{Tc}(\mathrm{CFP1}, \mathrm{CFP2}) \times \mathrm{Tc}(\mathrm{S1FP1}, \mathrm{S1FP2}) \times \mathrm{Tc}(\mathrm{S2FP1}, \mathrm{S2FP2}) \quad (2)$$

Here, Tc represents the Tanimoto or Jaccard coefficient.[31] The RDKit toolkit[26] and in-house Python scripts were used to generate all fingerprints.

2.4 Machine Learning Models

Nearest neighbor (k-NN) classification, support vector machine (SVM), random forest (RF), and a deep neural network (DNN) were used as classification methods. For pair-based models, training pairs were represented as a feature vector $x \in X$ and associated with a class label $y \in \{0,1\}$ distinguishing PCs and non-PC MMPs. For single compound models, the class label $y \in \{0,1\}$ was used to distinguish promiscuous and non-promiscuous compounds.

2.4.1 Nearest Neighbor Classification

The *k*-NN classifier was used as a control for machine learning models. It stores the feature vectors and class labels of the training set. A test pair is classified by calculating similarity values for all training instances and returning the majority class label of the *k* nearest neighbors with the highest similarity. As a hyperparameter, *k* was optimized using values of 1, 3, and 5. In addition, if not selected, a 1-NN classifier was also used as a reference.

2.4.2 Random Forest

RF represents an ensemble of decision trees, each of which is built from distinct subsets of the training data with replacement (bootstrapping).[32,33] Each tree is constructed from a random subset of features during node splitting.[34] Class label prediction is facilitated on the basis of the majority of ensemble votes. The number of randomly selected features available at each split was set to the square root of the number of ECFP4 features, and the minimum number of samples required to reach a leaf node was set to 1. During hyperparameter optimization, the maximum depth of individual trees was optimized using values of 10, 100, and 'unlimited'. In addition, the number of trees per ensemble was optimized by 5-fold cross-fold validation using values of 1, 10, 100, 500, and 1000. RF models were built using scikit-learn.[35]

2.4.3 Support Vector Machine

SVM is a machine learning algorithm aiming to construct a hyperplane *H* to separate two classes of training data by maximizing the distance between the classes in feature space.[36] The training data is projected into feature space *X* to determine a hyperplane *H* by a weight vector *w* and a bias *b* such that $H = \{x|\langle w, x\rangle + b = 0\}$. To generalize the model, slack variables are added permitting a limited number of classification errors of training instances falling within the margin or on the incorrect side of *H*. The training error and margin size result from hyperparameter *C*, which was optimized by 5-fold cross-validation using values of 2^i, with *i* representing all natural numbers between −10 and 10. Usually, linear separation of the training instances in a given feature space *X* is not possible. Therefore, as a central part of SVM modeling, training data are projected into a higher-dimensional space *H*. The projection is facilitated through the use of kernel functions replacing the standard scalar product, the so-called 'kernel trick',[36] which circumvents explicit mapping of *X* into *H*. To compute the similarity between compounds, the Tanimoto kernel[37] was used. For the comparison of compound pairs, the MMP kernel, which is an extension of the Tanimoto kernel, was used. The similarity between two MMPs using the MMP kernel equals the MMP fingerprint similarity as defined in section 2.3 by equations (1) and (2). SVM models were implemented using scikit-learn.

2.4.4 Feedforward Deep Neural Network

A feedforward DNN derives a function $y = f(x;w)$ that maps an input value *x* to a class *y* and learns the value of parameters *w* to achieve the best approximation. The DNN architecture is composed of different layers of neurons including an input layer, multiple hidden layers, and an output layer.[38] Each neuron in a DNN accepts an n-dimensional input *x* and produces an m-dimensional output vector *y* using a linear transformation $y = W^T \times x$, where *W* is a parameter of dimensions (n, m). Usually, neurons are associated with an additional m-dimensional parameter *b*, which is added after the linear transformation. In the next step, the output of the neuron is passed through a non-linear activation function. During training, parameters *W* and *b* are modified to yield the correct output *y* on the basis of a gradient descent cost function using backpropagation.[38] For training, data subsets (batches) are used, and the parameters *W* and *b* are updated accordingly. Implementations were based on PyTorch version 1.3.1.[39]

DNN hyperparameters were optimized as follows. For the drop-out rate, values of 0%, 25%, and 50% were evaluated, the learning rate was set to 0.0005, and the number of epochs was set to 5. A set of different network architectures (values of output features in hidden layers) was investigated including [250,250], [250,500], [500,250], [500,250,100], [100,250,500], and [250,100,250]. To these ends, pyramidal, rectangular, and autoencoder architectures were considered during hyper-parameter optimization. The Adam optimization algorithm[40] was chosen as the optimization function, the rectified linear unit[41] was selected as the activation function, and the batch size was set to 10.

2.5 Performance Measures

To assess model performance, four different measures were applied including balanced accuracy (ACC), Matthew's correlation coefficient (MCC), F1 values, and Area Under the Receiver Operating Characteristic Curve (ROC AUC). ACC, MCC, F1, and ROC AUC are defined as:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (3)$$

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

$$F1 = 2 \times \frac{TP}{2\,TP + FP + FN} \quad (5)$$

$$ROC\ AUC = \frac{1}{2} - \frac{1}{2}\frac{FP}{FP+TN} + \frac{1}{2}\frac{TP}{TP+FN} \quad (6)$$

Abbreviations: TP, true positives; TN, true negatives; FP, false positives; FN, false negatives.

3 Results and Discussion

3.1 Compound Pair-Based Promiscuity Predictions

To further investigate compound promiscuity with the aid of machine learning, we devised an analysis involving five different approaches and two distinct data sources including a large set of extensively assayed screening compounds with experimentally confirmed (non-)promiscuity and another manually curated set of human kinase inhibitors from the medicinal chemistry literature. The two intrinsically different data sets were intentionally selected to enable independent assessments of promiscuity via machine learning in the presence or absence of experimental test frequency information. This also assigned different levels of confidence to PCs and non-PC MMPs, directly addressing the data incompleteness issue. The primary task of our analysis has been promiscuity prediction at the level of compound pairs. Compound pair-based predictions were previously only attempted for activity cliffs.[42]

Therefore, MMPs were systematically calculated for both data sets and PCs as well as non-PC MMPs were assembled as detailed above and reported in Table 1. In addition, training and test sets for single compound promiscuity predictions were also extracted from both data sources.

Table 1. Reported are the numbers of PCs for different ΔPD thresholds.

Data source	ΔPD	PCs	Cpds[a]	Non-prom.[b]	Prom.[c]
PubChem screening compounds	2–3	273,862	152,971	111,399	41,572
	4–5	57,608	46,302	35,751	10,551
	6–7	18,759	17,268	13,640	3628
	8–9	6403	6762	5364	1398
	≥10	5750	5694	4706	998
Kinase inhibitors	≥10	5615	4187	3588	599

[a] Number of compounds
[b] Number of non-promiscuous compounds
[c] Number of promiscuous compounds

On the basis of PubChem data, a total of 170,606 compounds formed 362,362 PCs with a ΔPD of 2 or larger. Among these, there were 5750 PC with ΔPD ≥ 10.

For the formation of kinase inhibitor PCs, which exclusively consisted of active compounds, the ΔPD ≥ 10 criterion was consistently applied. This also ensured that kinase inhibitors PCs exclusively consisted of compounds with large differences in the number of kinase targets they were active against.

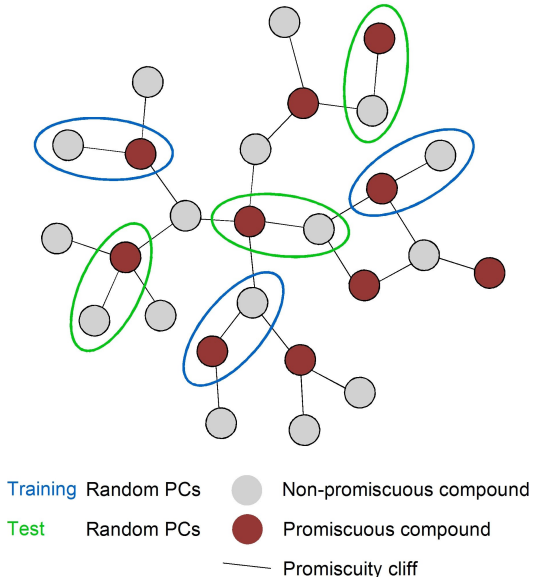

Figure 1. The network-based selection strategy is schematically illustrated. Red nodes represent promiscuous compounds in PCs with ΔPD ≥ 10, gray nodes non-promiscuous PC partners (PD < = 1), and edges the formation of pairwise PCs.

3.2 Selection of Training and Test Set

Machine learning models were trained on the basis of carefully assembled balanced training and test sets to distinguish between PCs and non-PC MMPs. To aid in the selection of unique pairs, each of the six sets of PCs described above was organized in a PC network. Figure 1 shows a schematic representation of the PC network and PC selection. In addition, non-PC MMPs were obtained by randomly sampling an MMP network built from all non-promiscuous compounds.

In addition, we compared pair-based promiscuity difference predictions with promiscuity predictions of individual compounds taken from PCs, which served as a reference for PC predictions. Therefore, promiscuous and non-promiscuous compounds were taken from PCs and non-PC MMPs, respectively, and the same number of the training and test instances was selected in each case.

3.3 Molecular Representation for Machine Learning

Each compound was encoded using the ECFP4 format. For each training and test PCs, the ΔPD was assigned in a direction-dependent manner such that the promiscuous compound was presented preceding the non-promiscuous

analog. This ordering provided the basis for the generation of the MMPFP, as shown in Figure 2a. Here, separate fingerprint components representing the common core and distinguishing substituents were combined. MMPFP similarity was calculated as the product of pairwise Tanimoto similarity comparisons of the three components, as schematically illustrated in Figure 2b. This similarity assessment was inspired by the design of MMP-based kernel functions that were successfully used for the prediction of activity cliffs.[42] For prospective applications predicting PCs among new MMPs, ordering is not essential for test instances.

3.4 Prediction of Promiscuity Cliffs

For systematically distinguishing between PCs and non-PC MMPs, 1-NN, k-NN, SVM, DF, and DNN classification models were generated. The predictive performance of each model was assessed using ACC, MCC, F1, and ROC AUC values (as defined in Materials and Methods).

First, the models were used to predict PCs with $\Delta PD \geq 10$ from the screening compound and kinase inhibitor data sets. These PCs encoded largest differences in analog promiscuity compared to non-PC MMPs. The results are summarized in Table 2.

Table 2. Reported are the ACC, MCC, F1, and ROC AUC values using 1-NN, k-NN, SVM, RF, and DNN models predicting PC formation.

Data source	Metric	1-NN	k-NN	SVM	RF	DNN
PubChem screening compounds	ACC	0.71	0.70	0.78	0.78	0.76
	MCC	0.42	0.40	0.56	0.56	0.53
	F1	0.72	0.71	0.77	0.77	0.75
	ROC AUC	0.71	0.77	0.85	0.86	0.84
Kinase inhibitors	ACC	0.64	0.64	0.64	0.71	0.70
	MCC	0.28	0.28	0.31	0.42	0.40
	F1	0.66	0.66	0.56	0.71	0.66
	ROC AUC	0.64	0.64	0.72	0.78	0.77

For pairs of screening compounds, all models were found to be predictive with an accuracy of at least 70%. The accuracy of SVM and RF approached 80%, indicating a clear tendency to distinguish between substitutions in the context of specific core structures leading to a large change in the promiscuity of structural analogs. Given the consistently high MCC (max. 0.56), F1 (0.77), and ROC AUC (0.86) values, there was no significant difference in prediction accuracy between the SVM, RF, and DNN machine learning models.

For kinase inhibitors, the models displayed similar trends. Although global model performance decreased relative to screening compounds, the machine learning models were comparably predictive, with max. ACC and F1 values of 0.71 and max. MCC value of 0.42. For all machine learning models, MCC values were ~0.10 lower than for screening compounds. The overall lower predictive performance on the kinase inhibitor set might be due to two factors including the general structural similarity of many (ATP site-directed) kinase inhibitors and the uncertainties associated with unknown test frequencies, potentially resulting in underestimated PDs for a number of inhibitors, due to data incompleteness.

For comparison, the results of promiscuity predictions based upon single compounds are reported in Table 3.

Table 3. Reported are the ACC, MCC, F1, and ROC AUC values using 1-NN, k-NN, SVM, RF, and DNN models predicting promiscuity at the level of single compounds.

Data source	Metric	1-NN	k-NN	SVM	RF	DNN
PubChem screening compounds	ACC	0.63	0.65	0.68	0.71	0.69
	MCC	0.27	0.31	0.37	0.42	0.39
	F1	0.65	0.66	0.69	0.72	0.67
	ROC AUC	0.63	0.71	0.76	0.79	0.77
Kinase inhibitors	ACC	0.59	0.60	0.62	0.62	0.58
	MCC	0.19	0.22	0.24	0.25	0.16
	F1	0.63	0.66	0.65	0.65	0.60
	ROC AUC	0.59	0.63	0.64	0.64	0.60

While these models were also predictive, their accuracy was generally lower than for pair-based predictions. For example, for single compounds, MCC values only reached 0.42 and 0.25 for the screening compounds and kinase inhibitors, respectively. These findings indicated that PCs and non-PC MMPs captured more structure-promiscuity relationship information than individual compounds. For single-compound and PC predictions, results for PubChem compounds were slightly or moderately superior to kinase inhibitors. This was likely due to the fact that most kinase inhibitors target the ATP binding site and are often structurally similar. PubChem targets are diverse and the screening hits cover a variety of chemical classes.

Interestingly, the difference in predictive performance between simple nearest neighbor classifiers and machine learning models was only small across all prediction tasks, consistent with observations made when first attempting to predict individual promiscuous compounds.[23] Notably, for individual kinase inhibitors, prediction accuracy of the 1-NN and *k*-NN classifiers exceeded the accuracy achieved with the complex DNN. Hence, predictions of individual promiscuous compounds were mostly determined by nearest neighbor effects. On the other hand, for pair-based predictions, the accuracy of machine learning models was overall higher than for the *k*-NN classifiers, indicating that nearest neighbor effects alone could not fully explain

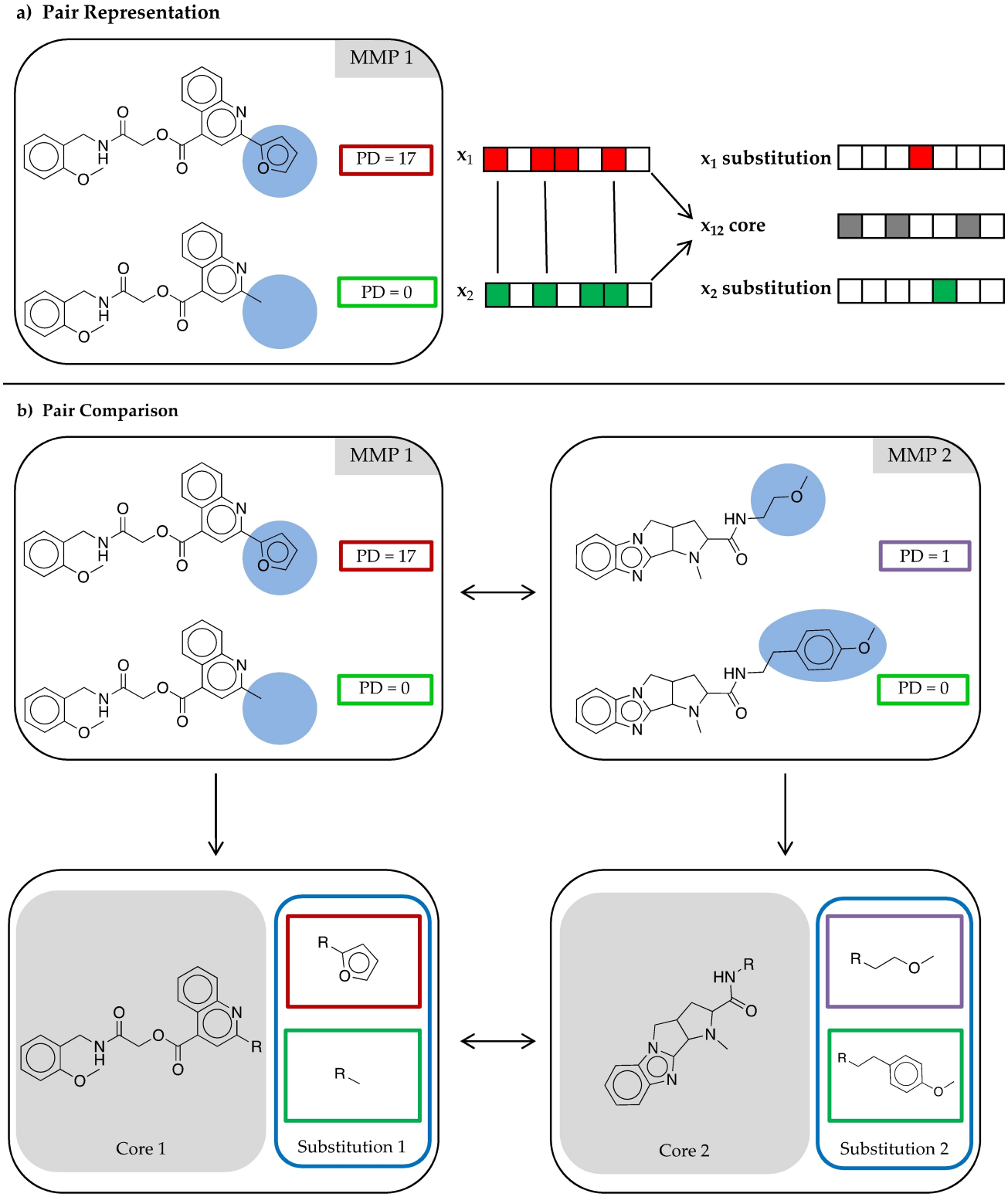

Figure 2. In (a), the fingerprint representation of a PC is shown. The MMPFP consists of the core fingerprint (common bits) and two substituent fingerprints (unique bits for compound 1 and 2, respectively). (b) illustrates pair-based similarity assessment combining contributions from individual fingerprint components.

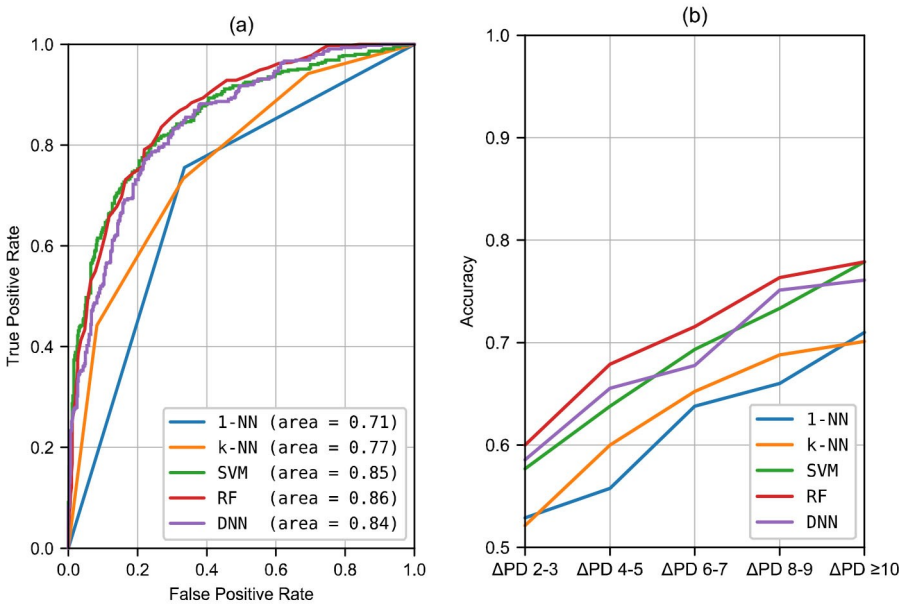

Figure 3. In (a), ROC curves are shown for predictions of PCs with $\Delta PD \geq 10$. In (b), prediction accuracy is compared for distinguishing between PCs with varying ΔPD values and non-PC MMPs.

predictive performance and reinforcing the notion of higher promiscuity-relevant information content of analog pairs compared to single compounds.

3.5 Predicting Promiscuity Cliffs with Varying ΔPDs

For screening compounds, we also investigated the influence of different ΔPD threshold values according to Table 1 on the prediction of PCs. In these calculations, PC training and test sets were varied but the same set of non-PC MMPs was used. Representative ROC curves in Figure 3a show that the performance of machine learning models was generally very similar for PC predictions and higher than of NN classifiers. Equivalent observations were made for PCs at all ΔPD thresholds. However, as shown in Figure 3b, the accuracy of PC predictions generally decreased with decreasing ΔPD threshold values. For example, for the RF model, accuracy decreased from 78% for PCs $\Delta PD \geq 10$ to 60% for PCs with ΔPD of 2–3. The same trend was observed for all models. For the NN classifiers, prediction accuracy for PCs with ΔPD of 2–3 was very close to random classification. Thus, PCs with decreasing ΔPD values between structural analogs presented increasingly difficult prediction tasks. These findings indicated that compounds having little differences in promiscuity were difficult to distinguish on the basis of structural patterns represented by transformations, providing further evidence for the presence of defined structure-promiscuity relationships. It also followed that the largest-magnitude PCs with $\Delta PD \geq 10$ represented a primary source of structural patterns determining prediction accuracy.

3.6 Structure-Promiscuity Relationships

The findings discussed above raised the question whether one might be able to identify structural features determining accurate PC predictions. To address this question, an in-depth analysis of feature relevance was carried out.

Therefore, the influence of individual fingerprint features on the predictions of screening compound PCs with $\Delta PD \geq 10$ was assessed using a surrogate model approach based on linear SVM and feature elimination. The use of linear SVM models permitted direct comparison of the importance of different features at the cost of lower predictive performance compared to nonlinear SVM models (using the MMP-kernel). To these ends, linear SVM models were iteratively built using reduced feature sets relative to the previous step. On the basis of SVM feature weights, we defined a positive and negative feature as a feature contributing to the correct classification of PCs and non-PC MMPs, respectively. In each iteration, the top 100 features with largest positive or negative weights in SVM predictions were identified and removed. This process was repeated for 40 iterations. Training data for predicting PCs with $\Delta PD \geq 10$ yielded a total of 51,146 distinct fingerprint features. Of these, 5,178 were positive and 5,432 negative. The relatively small size of subsets of positive or negative features compared to all features was attributable to inherent feature redundancy of fingerprints capturing layered atom

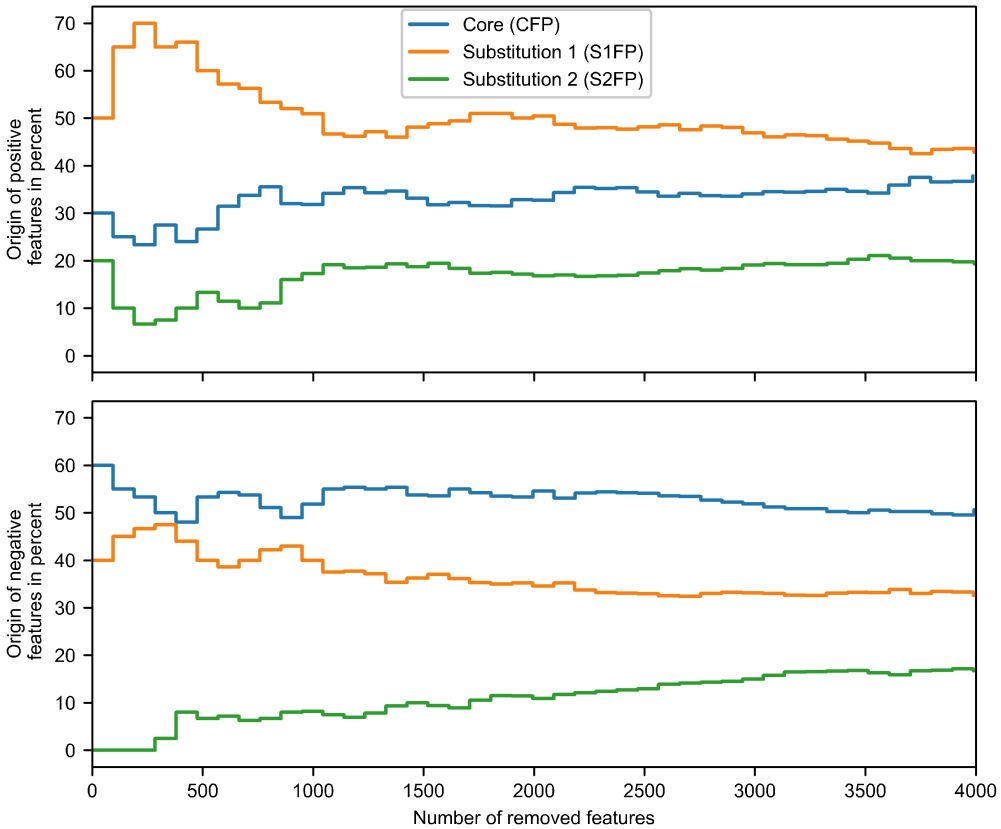

Figure 4. The graphs report the cumulative fraction of corresponding subsets of eliminated MMPFP features (top: positive, bottom: negative features). Eliminated features were mapped to the different CFP, S1FP, and S2FP components of MMPFPs.

environments. On the other hand, feature redundancy enabled the iterative generation of models with reduced feature sets retaining predictive power (i.e., eliminated features could be replaced with others).

Figure 4 reports the origin of positive (top) and negative features (bottom). Considering the first 100 positive features, 50% originated from S1FP representing the substituent fragment of the promiscuous PC analog 1. In addition, 30% of these features originated from the shared core structure, thus emphasizing contributions from the structural context in which S1FP features were presented. The remaining 20% originated from S2FP representing the substituent fragment of the non-promiscuous analog. Among negative features, only features produced by the core and substituent fragment 1 were highly ranked. During removal of 200–1000 features, the number of positive features from the core increased while the number of positive features from substituent 1 decreased. At the same time, the fraction of features from substituent 2 increased. After removal of 1000 features, the proportions of positive features from different fingerprint components remained largely constant and were essentially the same as prior to feature removal. For negative features, removal resulted in a gradual, albeit only minor increase in features from substituent 2. After removal of 4000 negative features, about 20% of negative features originated from substituent 2, 30% from substituent 1, and 50% from the common core.

Hence, compared to positive features, proportions of negative features from substituent 1 and the core were essentially inverted.

Taken together, the results indicated that features from both compounds in a pair contributed to the prediction of promiscuity differences. In all predictions, features from the three different components of MMPFP made significant contributions including features originating from the common core. These observations indicated that core-substituent combinations often played an important role for predicting pair-based promiscuity differences. Interestingly, features from substituent 1, the substituent of the promiscuous analog in PCs, often strongly contributed to correct predictions of promiscuity differences. However, predictions of non-PC MMPs were predominantly driven by core features. Hence, different distributions of positive and negative features differentiated between these predictions.

4 Conclusions

In this study, we have attempted to systematically predict PCs. The MMP formalism was applied to generate pairs of structural analogs encoding differences in compound promiscuity. From a conceptual point of view, the prediction of promiscuity differences between analogs on the basis of chemical structure is a non-trivial task. The underlying assumption is that structural differences in MMPs can be related to promiscuity differences and then be compared across analog pair populations. No previous analysis of compound promiscuity was carried out at the level of compound pairs. We approached this task using different machine learning models and compound data sources. First, we investigated extensively assayed screening compounds for which experimental test frequencies were available and exclusively assembled experimentally confirmed PCs and non-PC MMPs on the basis of shared targets. Second, we assembled PCs and non-PC MMPs from kinase inhibitors originating from the medicinal chemistry literature. In both cases, machine learning differentiated with reasonable to high accuracy between PCs and non-PC MMPs; an encouraging finding. Control calculations targeting individual promiscuous and non-promiscuous compounds were largely dominated by structural nearest neighbor effects. For compound pair-based predictions, accuracy was higher for experimentally confirmed PCs from screening compounds than for kinase inhibitors. Furthermore, prediction accuracy was found to increase with increasing ΔPD values captured by PCs; another encouraging observation. Taken together, the results of our predictions provided evidence for the presence of structural patterns that were associated with differences in promiscuity between structural analogs. Moreover, the component-based design of MMPFP made it possible to further explore structural features of different origins and their contributions to the predictions. SVM-based feature weighting and elimination revealed preferential feature contributions to accurate PC and non-PC MMP predictions from substituents of promiscuous analogs and the MMP core, in the context of which substitutions were presented. These insights strictly depended on exploring compound pair-based prediction of promiscuity and on successfully predicting promiscuity differences encoded by PCs. In light of our findings, we anticipate that PCs will be of considerable interest in further exploring structure-promiscuity relationships using computational approaches. Therefore, our data sets and the machine learning models reported herein are made freely available as an open access deposition on the Zenodo platform (https://zenodo.org/record/4013954).

Conflict of Interest

None declared.

Acknowledgments

We thank the OpenEye Scientific Software, Inc., for providing a free academic license of the OpenEye toolkit. Open Access funding enabled and organized by Projekt DEAL.

References

[1] Y. Hu, J. Bajorath, *Drug Discovery Today* **2013**, *18*, 644–650.
[2] G. R. Zimmermann, J. Lehár, C. T. Keith, *Drug Discovery Today* **2007**, *12*, 34–42.
[3] A. Anighoro, J. Bajorath, G. Rastelli, *J. Med. Chem.* **2014**, *57*, 7874–7887.
[4] M. L. Bolognesi, *Curr. Med. Chem.* **2013**, *20*, 1639–1645.
[5] M. Rosini, *Future Med. Chem.* **2014**, *6*, 485–487.
[6] M. L. Bolognesi, A. Cavalli, *ChemMedChem* **2016**, *11*, 1190–1192.
[7] D. H. Roukos, *Pharmacogenomics* **2011**, *12*, 695–698.
[8] H. Liu, J. Wang, W. Zhou, Y. Wang, L. Yang, *J. Ethnopharmacol.* **2013**, *146*, 773–793.
[9] J. Baell, M. A. Walters, *Nature* **2014**, *513*, 481–483.
[10] C. Aldrich, C. Bertozzi, G. I. Georg, L. Kiessling, C. Lindsley, D. Liotta, K. M. Merz, A. Schepartz, S. Wang, *J. Med. Chem.* **2017**, *60*, 2165–2168.
[11] J. J. Irwin, D. Duan, H. Torosyan, A. K. Doak, K. T. Ziebart, T. Sterling, G. Tumanian, B. K. Shoichet, *J. Med. Chem.* **2015**, *58*, 7076–7087.
[12] A. Gaulton, A. Hersey, M. Nowotka, A. P. Bento, J. Chambers, D. Mendez, P. Mutowo, F. Atkinson, L. J. Bellis, E. Cibrián-Uhalte, M. Davies, N. Dedman, A. Karlsson, M. P. Magariños, J. P. Overington, G. Papadatos, I. Smit, A. R. Leach, *Nucleic Acids Res.* **2017**, *45*, D945–D954.
[13] E. Gilberg, M. Gütschow, J. Bajorath, *ACS Omega* **2019**, *4*, 1729–1737.
[14] N. Sturm, J. Desaphy, R. J. Quinn, D. Rognan, E. Kellenberger, *J. Chem. Inf. Model.* **2012**, *52*, 2410–2421.
[15] V. J. Haupt, S. Daminelli, M. Schroeder, *PLoS One* **2013**, *8*, e65894.
[16] L. Pinzi, F. Caporuscio, G. Rastelli, *Drug Discovery Today* **2018**, *23*, 1889–1896.
[17] C. Feldmann, J. Bajorath, *Int. J. Mol. Sci.* **2020**, *21*, DOI 10.3390/ijms21113782.
[18] D. Dimova, Y. Hu, J. Bajorath, *J. Med. Chem.* **2012**, *55*, 10220–10228.
[19] J. Hussain, C. Rea, *J. Chem. Inf. Model.* **2010**, *50*, 339–348.
[20] Y. Hu, J. Bajorath, *Futur. Sci. OA* **2017**, *3*, FSO179.
[21] Y. Hu, S. Jasial, E. Gilberg, J. Bajorath, *AAPS J.* **2017**, *19*, 856–864.
[22] F. Miljković, J. Bajorath, *ACS Omega* **2018**, *3*, 17295–17308.
[23] T. Blaschke, F. Miljković, J. Bajorath, *ACS Omega* **2019**, *4*, 6883–6890.
[24] Y. Wang, S. H. Bryant, T. Cheng, J. Wang, A. Gindulyte, B. A. Shoemaker, P. A. Thiessen, S. He, J. Zhang, *Nucleic Acids Res.* **2017**, *45*, D955–D963.
[25] J. B. Baell, G. A. Holloway, *J. Med. Chem.* **2010**, *53*, 2719–2740.
[26] "RDKit: Open-Source Cheminformatics and Machine Learning Software," can be found under https://www.rdkit.org/, **2017**.
[27] T. Sterling, J. J. Irwin, *J. Chem. Inf. Model.* **2015**, *55*, 2324–2337.
[28] T. UniProt Consortium, *Nucleic Acids Res.* **2018**, *46*, 2699–2699.
[29] X. Hu, Y. Hu, M. Vogt, D. Stumpfe, J. Bajorath, *J. Chem. Inf. Model.* **2012**, *52*, 1138–1145.

[30] D. Rogers, M. Hahn, *J. Chem. Inf. Model.* **2010**, *50*, 742–754.
[31] P. Jaccard, E. Zurich, *Bull. la Société Vaudoise des Sci. Nat.* **1901**, *37*, 547–579.
[32] L. Breiman, *Mach. Learn.* **2001**, *45*, 5–32.
[33] B. Efron, *Ann. Stat.* **1979**, *7*, 1–26.
[34] E. Alpaydin, *Introduction to Machine Learning*, MIT Press, **2009**.
[35] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, A. Müller, J. Nothman, G. Louppe, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, É. Duchesnay, *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
[36] T. Joachims, *Advances in Kernel Methods: Support Vector Learning*, MIT Press, Cambridge, MA, **1999**.
[37] L. Ralaivola, S. J. Swamidass, H. Saigo, P. Baldi, *Neural Networks* **2005**, *18*, 1093–1110.
[38] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, **2016**.
[39] N. Ketkar, in *Deep Learn. with Python*, Apress, Berkeley, CA, **2017**, pp. 195–208.
[40] D. P. Kingma, J. Ba, *arXiv:1412.6980* **2014**, 1–15.
[41] V. Nair, G. E. Hinton, in *Proc. 27th Int. Conf. Int. Conf. Mach. Learn.*, Omnipress, USA, **2010**, pp. 807–814.
[42] K. Heikamp, X. Hu, A. Yan, J. Bajorath, *J. Chem. Inf. Model.* **2012**, *52*, 2354–2365.

Received: August 6, 2020
Accepted: September 3, 2020
Published online on September 29, 2020

