Selecting machine-learning scoring functions for structure-based virtual screening

Pedro J. Ballester[1*]

[1] Centre de Recherche en Cancérologie de Marseille (CRCM), Inserm, U1068, Marseille, F-13009, France. CNRS, UMR7258, Marseille, F-13009, France. Institut Paoli-Calmettes, Marseille, F-13009, France. Aix-Marseille University, UM 105, F-13284, Marseille, France.

Abstract

Interest in docking technologies has grown parallel to the ever increasing number and diversity of 3D models for macromolecular therapeutic targets. Structure-Based Virtual Screening (SBVS) aims at leveraging these experimental structures to discover the necessary starting points for the drug discovery process. It is now established that Machine Learning (ML) can strongly enhance the predictive accuracy of scoring functions for SBVS by exploiting large datasets from targets, molecules and their associations. However, with greater choice, the question of which ML-based scoring function is the most suitable for prospective use on a given target has gained importance. Here we analyse two approaches to select an existing scoring function for the target along with a third approach consisting in generating a scoring function tailored to the target. These analyses required discussing the limitations of popular SBVS benchmarks, the alternatives to benchmark scoring functions for SBVS and how to generate them or use them using freely-available software.

Contact: pedro.ballester@inserm.fr

Introduction

The primary goal of Virtual Screening (VS)[1,2] is to retrieve a small subset of molecules with the highest possible proportion of actives from the screened library. When a 3D structure of the protein target is available and the binding site is known, this problem is more specifically called Structure-Based VS (SBVS). Molecular docking is arguably the most common tool for SBVS. This type of technology has been employed to discover active molecules with novel chemical scaffolds in a fast and cost-effective manner[3–8].

Docking tools incorporate a regression model, its native Scoring Function (SF), which provides a fast estimate of how strongly the putative protein-ligand complex binds. In

SBVS, the SF is used to rank molecules docked to a therapeutic target, with top-ranked molecules being those predicted to bind more strongly. Top-ranked molecules are hence those predicted to have higher binding affinity (equivalently, lower dissociation constants K_d or lower free energies of binding $\Delta G_{binding}$), which are in turn assumed to have higher target activities (e.g. lower half-maximal inhibitory concentration IC_{50}). SFs can also be used to re-score the docked poses of putative ligand molecules. In this case, SFs can also be classification models returning the likelihood of the molecule to be active (class probability) or even directly the predicted label (active or inactive).

SFs built with Machine Learning (ML), arguably the most developed branch of Artificial Intelligence (AI), have demonstrated remarkable accuracy on various drug design applications[9–14]. In particular, when re-scoring crystal structures with ligand-bound proteins, or even their redocked poses, SFs are now able to predict the affinities of these binding molecules with high accuracy on many targets (we just wrote a review[13] focusing on this problem and discussing many examples from the literature). By contrast, the application of SFs to SBVS is a harder problem, as SFs may struggle to simultaneously predict the affinities of binding and non-binding molecules. With that said, ML-based SFs have been found to provide accurate SBVS performance on many targets[14]. This is particularly true for those targets for which some of their known binders can be employed to train target-specific ML-based SFs[15,16].There is however uncertainty as to which SF would be most appropriate for a given target. To shed light into this crucial question, this article presents evidence-based guidelines to select SFs for prospective SBVS against a target of interest.

Selecting a scoring function based on published evaluations

This is the fastest option, but the least reliable one. This option is available if the target of interest is in a published study and a given SF performs well on the corresponding test set. The question is how much trust we can put on such results. The test set for the target typically consists of a set of known actives and a larger set of assumed inactives (decoys). Property-Matched (PM) decoys, such as those in DUDE[17] or DEKOIS2.0[18], have been a popular choice for benchmarking SBVS methods. PM decoys are molecules selected on the basis of being hard to distinguish (decoys with similar physicochemical properties to those of the considered active) and being likely inactives (decoys with chemical structures dissimilar to that of their active). A historical perspective of PM decoys is available elsewhere[19,20].

As further discussed elsewhere[14], current SBVS benchmarks do not mimic real test sets and hence their ability to anticipate prospective performance must at best be suboptimal. A prime example of this is MUV[21], whose test sets were built from High-Throughput Screening (HTS) datasets. In MUV, actives that are not considered to be well embedded in the true inactives (called decoys in that study) are removed. However, by definition, an optimal embedding is the one that is provided by the real test set (the HTS library) and hence MUV is an unrealistic benchmark in this sense. On the other hand, the ability of benchmarks based on PM decoys to mimic real test sets has also been strongly criticised, as explained in the rest of this section. Property-unmatched decoys have been proposed as an alternative. Chen et al.[22] proposed an Actives as Decoys (AD) set of molecules per DUDE target to minimise the bias arising from the DUDE decoy selection criteria. Also, the lead authors of DUDE recently introduced a new benchmark using a single set of property-unmatched decoys for all targets (Goldilocks[23]) intended to represent the physical properties of the library to-be-docked in a more accurate manner.

To my knowledge, there is no evidence showing that using PM decoys makes the resulting benchmark predictive of prospective performance. For example, larger ligands tend to enter more intermolecular interactions with the target, which in turn results in larger contributions from the corresponding SF components. In classical SFs, these components are simply weighted and added up, and hence their scores correlate, to some extent, with ligand size. Thus, employing molecular-weight-matched decoys prevents such SFs from easily discriminating between actives and decoys. However, a classical SF with poor performance on a test set using such PM decoys could still perform well prospectively (e.g. because not all its measured predictive power came from ligand size correlation and hence could rank high differently-sized actives in the screening library). Conversely, the same SF could obtain excellent performance on that retrospective benchmark, but perform poorly prospectively for a variety of reasons (e.g. the low chemical diversity of the PM decoys not representing well that of inactives in the screening library). Moreover, PM-decoy-like inactives are much rarer in the very large screening libraries used in prospective applications (the real test sets), as only a small proportion of library molecules would verify all the PM selection criteria.

The shortcomings of DUDE PM decoys have been investigated with both ML-based SFs[16,20,22] andclassical SFs[16,20,24,25]. Xiong et al.[16] showed that DUDE overestimates the performance of a range of SFs using protein-ligand features relative to its unbiased AD version (AD uses instead property-unmatched decoys[22]). When using ligand-only features instead, the classification accuracies on intra-target cross-validations are more strongly overestimated in DUDE[20] for both a classical SF based on logistic regression (mean AUC of 0.99) as well as ML-based SFs based on Random Forest (mean AUC of 1.0). Other cross-target experiments in this study[20] showed that a DUDE-trained SF using ligand-only features obtains essentially the same leave-target-cluster-out performance as a SF exploiting protein structure as well (protein-ligand features), both SFs built with Ragoza et al.'s Convolutional Neural Network (CNN) algorithm[26]. Based on this result, Sieg et al.[20] claimed that DUDE-trained SFs, regardless of whether they employ ligand-only or protein-ligand features, suffer from non-causal bias, because both employ ligand features and their performances on the DUDE benchmark are very similar (the latter was also found by others[22]). Models suffering from non-causal bias means here models that "do not work in general but only on data that fits the bias pattern, which makes them unusable for prospective predictions"[20]. These authors should have tested the claim to support it or reject it by evaluating both SFs on data without DUDE decoy bias, but they did not. The claim is indeed reasonable for the DUDE-trained SF using ligand-only features because, deprived from any protein structure information, the observed performance should all come from learning the decoy bias (i.e. the ligand-driven way in which decoys were selected with respect to each active). However, it is not plausible for the DUDE-trained SF using protein-ligand features because its performance can come from two sources: learning decoy bias and learning how its protein-ligand features relate to binding. While the decoy-bias component is responsible for overestimating its observed DUDE performance, the other component can make the SF work well on at least some other test sets without the same decoy bias.

In fact, these missing experiments were carried out by other authors. There are five independent studies showing that DUDE-trained SFs using protein-ligand features actually perform very well on test sets other than DUDE. First, DUDE-trained RF-Score-VS was tested on 76 DEKOIS2.0 targets[27], where it obtained on average 2.5 times the enrichment ($EF_{1\%}$) of Smina. Second, this was also shown independently by Chen et al.[28] on 55 DEKOIS2.0 targets, where DUDE-trained RF-Score-VS obtained 1.9, 3.1 and 1.9 times the $EF_{1\%}$ of Smina, DLIGAND and DLIGAND2, respectively (all three are classical SFs). The third study presented DUDE-trained RF-based SIEVE-Score[15], which achieved a 1.9

times higher $EF_{1\%}$ than Glide on average on DEKOIS2.0 targets, i.e. about twice its average hit rate on these targets. It must be noted that DEKOIS2.0 decoys come from similar selection criteria than DUDE decoys, and hence these test sets might still overestimate the VS performance of ML-based SFs using ligand-only features[20] (as explained before for DUDE decoys, any overestimation should be much smaller in ML-based SFs using protein-ligand features). The fourth and fifth studies proposed DUDE-trained CNN-based SFs using protein-ligand features have also been tested on datasets with property-unmatched decoys obtaining hit rates between 3 and 4 times higher than those from Vina[26,29] (i.e. not only the performance was not poor, but actually excellent compared to this classical SF). More concretely, each of these datasets generated by Riniker & Landrum[30] contained 100 diverse actives of a target along with 10,000 decoys (9,800 of these were effectively selected at random, while the remaining 200 were mildly atom-count-matched to an active). Incidentally, one of these SFs[26] was built with the same Ragoza et al.'s CNN algorithm that was later employed by Sieg et al. for their experiments[20], but surprisingly not even this directly contradictory result was mentioned by the authors.

To sum up, the results of SFs that are trained and tested on data using the same PM decoy selection criteria are very likely overestimated. However, if training set and test set employed decoys selected in different ways, the retrospective performance of the corresponding SFs should in principle be trustworthy.

Selecting a scoring function based on your own evaluation

In case the SFs have not been evaluated properly, or not at all, on the target of interest, making your own evaluation of selected SFs may lead to the identification of a predictive SF. This requires retrieving all known actives for the target in relevant databases[31,32] to include in a test set. The other part of the test set will be a much higher proportion of decoys. As discussed in the previous section, employing PM decoys have important shortcomings arising from not capturing the properties of the test sets of interest. In ML, supervised learning must be performed on a training set that resembles the intended test set, otherwise the resulting model will be unlikely to generalise to that test set. This is consequently an established requirement in QSAR too, where models with applicability domains covering test sets of interest are sought after. More specifically, Smusz et al. have argued[33] that, instead of being property-matched, training set decoys should be as representative as possible for the libraries that undergo screening. From this perspective, selecting actives and decoys in the expected proportion and with properties that are also observed in the screening library seems a much better option. Such property-unmatched decoys have already demonstrated their value to anticipate prospective performance. For instance, a retrospective analysis based on random decoys found that a ML-based SF (MIEC-SVM) was much more predictive than a classical SF (Autodock4.2) on the ALK target, which was exactly what was later observed prospectively[8]. This is not the only ML-based SF reporting excellent prospective SBVS results without any use of PM decoys[14,34]. It is important to note too that PM decoys are not required either to train or test QSAR models[35], despite predicting exactly the same *in vitro* potency/affinity endpoints as SFs (e.g. K_d is predicted by both SFs[36,37] and QSAR models[38,39]).

On the other hand, the number of HTS datasets has strongly increased in recent years[32,40]. If available for the target of interest, such dataset constitutes an interesting alternative as a test set (containing true instead of assumed inactives, better characterising the chemical diversity of molecules or representing by definition a realistic active-inactive proportion). It is however advisable to curate HTS datasets prior to modelling [41]. While HTS datasets have been used to test QSAR models[42,43], this is yet to be done with ML-based SFs. These studies are very encouraging. For example, Liu et al.[43] cross-validated a range of QSAR models on data from an HTS on a protein-protein interaction target (off-the-shelf classical SFs were also tested on the same validation folds, but not ML-based SFs). They found that the QSAR model using RF was the most predictive. All these models were tested on another dataset from a second HTS subsequently carried out by the authors on the same target. The RF model was also found to be the most predictive in this prospective test, outperforming QSAR models based on deep NN algorithms and classical SFs. This shows again that optimal models can be anticipated without any need for PM decoys.

Table 1 compiles several freely-available off-the-shelf generic ML-based SFs that could be used on the test set: NNScore[44,45], $\Delta_{vina}RF_{20}$[46], Δ_{vina}XGB[47], Convolutional NN[26], RF-Score-VS[27] or vScreenML[48]. For any employed SF, it is necessary to find out which protein-ligand complexes where used for training it and remove them from the test set if also found there. Using a freely-available classical SF, such as Vina[49], would be useful as a common performance baseline. In addition to evaluating the SFs on the entire test set, their performance on certain test subsets would be very informative. For instance, on the subset of complexes bound by molecules dissimilar to any training set molecule, as SFs performing well here should discover a higher proportion of novel compounds[5,8,48]. Another example is the subset of complexes with most potent actives/binders (e.g. lowest K_d), which would permit identifying the SFs prone to discover the most potent ligands of the target[25].

TABLE 1 Selected community resources relevant to SFs for SBVS. Abbreviations: B (Benchmark), D (Data), SF (Scoring Function), F (Features) and M (Modelling). NB: only SFs for SBVS are highlighted, SFs for other applications are reviewed elsewhere (e.g.[13]).

Resource	B & D	SF	F	M	Availability
DUDEZ Goldilocks[23]	✓	✗	✗	✗	http://dudez.docking.org/
AD[22]	✓	✗	✗	✗	https://doi.org/10.1371/journal.pone.0220113
CASF[50]	✓	✗	✗	✗	http://www.pdbbind-cn.org/casf.asp
DUDE[17]	✓	✗	✗	✗	http://dude.docking.org
DEKOIS 2.0[18]	✓	✗	✗	✗	http://www.dekois.com
Riniker & Landrum[30], MUV[21]	✓	✗	✗	✗	http://dx.doi.org/10.1186/1758-2946-5-26
D-COID[48]	✓	✗	✗	✗	https://data.mendeley.com/datasets/8czn4rxz68/1
Δ_{vina}RF$_{20}$[46], Δ_{vina}XGB[47]	✗	✓	✓	✗	https://www.nyu.edu/projects/yzhang/DeltaVina
Deep Convolutional NN[26]	✗	✓	✓	✗	https://github.com/gnina
RF-Score-VS[27]	✗	✓	✓	✗	https://github.com/oddt/rfscorevs_binary
SIEVE-Score[15]	✗	✓	✓	✗	https://github.com/sekijima-lab/SIEVE-Score
vScreenML[48]	✗	✓	✓	✗	https://github.com/karanicolaslab/vScreenML
SIEVE-Score[15]	✗	✓	✓	✗	https://github.com/sekijima-lab/SIEVE-Score
NNScore[44,45]	✗	✓	✓	✗	https://sourceforge.net/projects/nnscore/
BINANA[51]	✗	✗	✓	✗	https://git.durrantlab.pitt.edu/jdurrant/binana
Algebraic topology[52]	✗	✗	✓	✗	https://doi.org/10.1371/journal.pcbi.1005929.s002
RF-Score v3[53]	✗	✗	✓	✗	https://github.com/HongjianLi/RF-Score
MIEC-SVM[54]	✗	✓	✓	✓	http://wanglab.ucsd.edu/MIEC-SVM
ODDT[55]	✗	✓	✓	✓	https://github.com/oddt/oddt
Descriptor Data Bank[56]	✗	✓	✓	✓	http://www.descriptordb.com

Building and evaluating a tailored machine-learning scoring function

If not sufficiently predictive SF is identified on the test set, building a SF tailored to the target is on average more predictive than selecting the SF with the best average performance across targets[15,25]. With this purpose, there is a plethora of ML algorithms in several programming languages that can be tuned to this target, once data instances and features are in place. Table 1 compiles freely-available software to calculate protein-ligand features. Target-specific features can be calculated if the training set only includes ligands complexed with the target of interest, which gave excellent results with SIEVE-Score[15]. A target-specific SF can also be built using generic features (i.e. those that can be calculated for any target) if its training set only contains molecules complexed/docked to that target. The latter has been shown[16] to outperform classical SFs such as GlideScore-SP at SBVS by a large margin when using PM decoys (this was also true when using AD property-unmatched decoys). This is exciting given that there are many targets for which target-specific SFs could be generated (e.g. 917 single-protein targets in the ChEMBL database with at least 40 ligands per target using an activity threshold of 10 μM[57]). Also, selecting training complexes with similar targets and/or ligands as those in the test set have been found to improve performance in this and related docking problems[27,58]. Such transfer learning is particularly important for those targets with few or no known ligands yet, where training or even test sets might not be available. The SF with the best leave-target-out cross-validation[27] would be the most promising to tackle these challenging targets.

Furthermore, there are now tools to facilitate the generation and validation of ML-based SFs by providing integrated and well-documented software packages (Table 1). For instance, the Descriptor Data Bank (DDB)[56] implements a ML toolbox for automatic filtering and analysis of descriptors (features) as well as SF training and prediction. The descriptor filtering module can filter out irrelevant or noisy descriptors to produce a compact subset from the 2,700 descriptors that are initially considered. There is also a standalone version of DDB for download in the form of Python command line programs and a PyMOL plugin, which can be used for high-throughput processing and visualisation of docked complexes.

Another suitable software package is the Open Drug Discovery Toolkit (ODDT) [55]. This is a modular and comprehensive toolkit for use in chemoinformatics and molecular modelling in either Python programming language or its command line interface. ODDT handles ligand molecules in various file formats, generates features for protein-ligand complexes (e.g. BINANA[51], PLEC[59]) as well as ligands (e.g. USR[60], USRCAT[61]) and builds SFs using established ML algorithms such as RF or NN. Features and ML algorithms can build ML-based SFs such as RF-Score[37], NNScore[44,45] and PLECscore[59] on user-supplied data. Multiple metrics to quantify performance and cross-validations for model assessment are implemented. The trained SF can be saved to a file, shared between users, and subsequently reloaded and used in the command line directly.

Conclusions

Retrospective SBVS evaluations are crucial to select a ML model that will perform well prospectively on the same target (e.g.[8,43]). We have discussed here three approaches to this problem. The first approach is to make the selection based on published results if SFs have already been evaluated on the target of interest. It is however important to discard results where the SF has been trained and tested using data with the same PM decoy selection criteria, as its performance will come, to some extent, from learning the way decoys were selected from actives rather than learning the physics of molecular recognition. However, if training set and test set employed decoys selected in different ways, the retrospective performance of the corresponding SFs should in principle be trustworthy.

The second approach consists in evaluating freely-available ML-based SFs oneself. This requires assembling a test set with molecules and their activities for the target along with a much higher proportion of assumed inactives (decoys). We have seen that employing PM decoys has important shortcomings coming from only representing one possible type of inactives found in screening libraries. Unlike test sets with PM decoys, property-unmatched decoys better representing the diversity and distribution of inactives in the intended test set have been shown to be predictive of prospective performance[8,43]. Striving to generate synthetic benchmarks exhibiting the latter capability is essential. Indeed, if benchmarks do not represent well the library to-be-docked, we will likely be applying prospectively SFs that only perform well on retrospective benchmarks and not applying actually predictive SFs that were not selected because of poor performance on unrealistic retrospective benchmarks. On the other hand, if a HTS exists for the target, using it unaltered as a test set represents the most realistic benchmark by definition. Regardless of whether the benchmark is synthetic or not, test subsets can be used to identify those SFs that excel at identifying the most potent and/or novel molecules for the target.

Building and evaluating a ML-based SF tailored to the target is the last discussed approach. This is the most promising, as exploiting target-specific data and/or features has been found to be more predictive than SFs using data from any target and generic features[15,16,25]. However, it was also the most complex to implement, even if codes for the individual components were available. This approach has now been enormously simplified by the release of integrated and well-documented software packages provided for this purpose.

Funding Information

This work has been carried out with the support of the ANR Tremplin-ERC grant (n° ANR-17-ERC2-0003-01) awarded to P.J.B.

References

[1] Schneider G. Virtual screening: an endless staircase? Nat Rev Drug Discov 2010;9:273–6. doi:10.1038/nrd3139.

[2] Cheng T, Li Q, Zhou Z, Wang Y, Bryant SH. Structure-Based Virtual Screening for Drug Discovery: a Problem-Centric Review. AAPS J 2012;14:133–141. doi:10.1208/s12248-012-9322-0.

[3] Li L, Khanna M, Jo I, Wang F, Ashpole NM, Hudmon A, et al. Target-Specific Support Vector Machine Scoring in Structure-Based Virtual Screening: Computational Validation, In Vitro Testing in Kinases, and Effects on Lung Cancer Cell Proliferation. J Chem Inf Model 2011;51:755–9. doi:10.1021/ci100490w.

[4] Ballester PJ, Mangold M, Howard NI, Robinson RLM, Abell C, Blumberger J, et al. Hierarchical virtual screening for the discovery of new molecular scaffolds in antibacterial hit identification. J R Soc Interface 2012;9:3196–207. doi:10.1098/rsif.2012.0569.

[5] Durrant JD, Carlson KE, Martin TA, Offutt TL, Mayne CG, Katzenellenbogen JA, et al. Neural-Network Scoring Functions Identify Structurally Novel Estrogen-Receptor Ligands. J Chem Inf Model 2015;55:1953–61. doi:10.1021/acs.jcim.5b00241.

[6] Zhao H, Caflisch A. Discovery of ZAP70 inhibitors by high-throughput docking into a conformation of its kinase domain generated by molecular dynamics. Bioorganic Med Chem Lett 2013;23:5721–6. doi:10.1016/j.bmcl.2013.08.009.

[7] Simmons KJ, Chopra I, Fishwick CWG. Structure-based discovery of antibacterial drugs. Nat Rev Micro 2010;8:501–10. doi:10.1038/nrmicro2349.

[8] Sun H, Pan P, Tian S, Xu L, Kong X, Li Y, et al. Constructing and Validating High-Performance MIEC-SVM Models in Virtual Screening for Kinases: A Better Way for Actives Discovery. Sci Rep 2016;6:24817. doi:10.1038/srep24817.

[9] Ain QU, Aleksandrova A, Roessler FD, Ballester PJ. Machine-learning scoring functions to improve structure-based binding affinity prediction and virtual screening. WIREs Comput Mol Sci 2015;5:405–24. doi:10.1002/wcms.1225.

[10] Shen C, Ding J, Wang Z, Cao D, Ding X, Hou T. From machine learning to deep learning: Advances in scoring functions for protein-ligand docking. Wiley Interdiscip Rev Comput Mol Sci 2019:e1429. doi:10.1002/wcms.1429.

[11] Yang X, Wang Y, Byrne R, Schneider G, Yang S. Concepts of Artificial Intelligence for Computer-Assisted Drug Discovery. Chem Rev 2019;119:10520–94. doi:10.1021/acs.chemrev.8b00728.

[12] Jensen KF, Coley CW, Eyke NS. Autonomous discovery in the chemical sciences part I: Progress. Angew Chemie Int Ed 2019. doi:10.1002/anie.201909987.

[13] Li H, Sze K-H, Lu G, Ballester PJ. Machine-learning scoring functions for structure-based drug lead optimization. WIREs Comput Mol Sci 2020:e1465. doi:10.1002/wcms.1465.

[14] Li H, Sze K-H, Lu G, Ballester PJ. Machine-learning scoring functions for structure-based virtual screening. WIREs Comput Mol Sci 2020:e1478.

(** This article reviews the last five years in this research topic. It pays particular attention to the limitations of established benchmarks, how these limitations are being overcome and the results of comparing different scoring functions on these new benchmarks. Prospective applications of machine-learning scoring functions are reviewed too.)

[15] Yasuo N, Sekijima M. An Improved Method of Structure-based Virtual Screening via Interaction-energy-based Learning. J Chem Inf Model 2019;59:1050–61. doi:10.1021/acs.jcim.8b00673.

(* This paper presents SIEVE-Score, a random forest-based scoring function employing target-specific protein-ligand features. Trained on DUDE data and tested on DEKOIS2.0 data, SIEVE-Score achieves on average across targets twice the hit rate of Glide and its code is freely available.

[16] Xiong G-L, Ye W-L, Shen C, Lu A-P, Hou T-J, Cao D-S. Improving structure-based virtual screening performance via learning from scoring function components. Brief Bioinform 2020:bbaa094. doi:10.1093/bib/bbaa094.

(** This study presents target-specific machine-learning scoring functions to classify docked molecules. Instead of combining the scoring function components classically, i.e. as a weighted sum of its componets, this is done nonlinearly using the XGBoost algorithm. The improvements over the corresponding classical SF are large on both DUDE test sets and test sets employing property-unmatched decoys).

[17] Mysinger M, M. M, Carchia M, Irwin JJ, Shoichet BK. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking. J Med Chem 2012;55:6582–94. doi:10.1021/jm300687e.

[18] Bauer MR, Ibrahim TM, Vogel SM, Boeckler FM. Evaluation and Optimization of Virtual Screening Workflows with DEKOIS 2.0 – A Public Library of Challenging Docking Benchmark Sets. J Chem Inf Model 2013;53:1447–62. doi:10.1021/ci400115b.

[19] Réau M, Langenfeld F, Zagury J-F, Lagarde N, Montes M. Decoys Selection in Benchmarking Datasets: Overview and Perspectives. Front Pharmacol 2018;9:11. doi:10.3389/fphar.2018.00011.

[20] Sieg J, Flachsenberg F, Rarey M. In Need of Bias Control: Evaluating Chemical Data for Machine Learning in Structure-Based Virtual Screening. J Chem Inf Model 2019;59:947–61. doi:10.1021/acs.jcim.8b00712.

[21] Rohrer SG, Baumann K. Maximum Unbiased Validation (MUV) Data Sets for Virtual Screening Based on PubChem Bioactivity Data. J Chem Inf Model 2009;49:169–84. doi:10.1021/ci8002649.

[22] Chen L, Cruz A, Ramsey S, Dickson CJ, Duca JS, Hornak V, et al. Hidden bias in the DUD-E dataset leads to misleading performance of deep learning in structure-based virtual screening. PLoS One 2019;14:e0220113. doi:10.1371/journal.pone.0220113.

[23] Stein R. Property-unmatched decoys in docking benchmarks. UCSF, 2020.

[24] Chaput L, Martinez-Sanz J, Saettel N, Mouawad L. Benchmark of four popular virtual screening programs: construction of the active/decoy dataset remains a major determinant of measured performance. J Cheminform 2016;8:56. doi:10.1186/s13321-016-0167-x.

[25] Fresnais L, Ballester PJ. The impact of compound library size on the performance of scoring functions for structure-based virtual screening. Brief Bioinform 2020:(In Press).

[26] Ragoza M, Hochuli J, Idrobo E, Sunseri J, Koes DR. Protein–Ligand Scoring with Convolutional Neural Networks. J Chem Inf Model 2017;57:942–57. doi:10.1021/acs.jcim.6b00740.

[27] Wójcikowski M, Ballester PJ, Siedlecki P. Performance of machine-learning scoring functions in structure-based virtual screening. Sci Rep 2017;7:46710. doi:10.1038/srep46710.

[28] Chen P, Ke Y, Lu Y, Du Y, Li J, Yan H, et al. DLIGAND2: an improved knowledge-based energy function for protein–ligand interactions using the distance-scaled, finite, ideal-gas reference state. J Cheminform 2019;11:52. doi:10.1186/s13321-019-0373-4.

[29] Imrie F, Bradley AR, van der Schaar M, Deane CM. Protein Family-Specific Models Using Deep Neural Networks and Transfer Learning Improve Virtual Screening and Highlight the Need for More Data. J Chem Inf Model 2018;58:2319–30. doi:10.1021/acs.jcim.8b00350.

[30] Riniker S, Landrum GA. Open-source platform to benchmark fingerprints for ligand-based virtual screening. J Cheminform 2013;5:26. doi:10.1186/1758-2946-5-26.

[31] Gaulton A, Hersey A, Nowotka ML, Patricia Bento A, Chambers J, Mendez D, et al. The ChEMBL database in 2017. Nucleic Acids Res 2017;45:D945–54. doi:10.1093/nar/gkw1074.

[32] Wang Y, Cheng T, Bryant SH. PubChem BioAssay: A Decade's Development toward Open High-Throughput Screening Data Sharing. SLAS Discov Adv Life Sci R D 2017;22:655–66. doi:10.1177/2472555216685069.

[33] Smusz S, Kurczab R, Bojarski AJ. The influence of the inactives subset generation on the performance of machine learning methods. J Cheminform 2013;5:17. doi:10.1186/1758-2946-5-17.

[34] Wijewardhane PR, Jethava KP, Fine JA, Chopra G. Combined Molecular Graph Neural Network and Structural Docking Selects Potent Programmable Cell Death Protein 1/Programmable Death-Ligand 1 (PD-1/PD-L1) Small Molecule Inhibitors. ChemRxiv Prepr 2020. doi:10.26434/CHEMRXIV.12083907.V1.

[35] Neves BJ, Braga RC, Melo-Filho CC, Moreira-Filho JT, Muratov EN, Andrade CH. QSAR-Based Virtual Screening: Advances and Applications in Drug Discovery. Front Pharmacol 2018;9:1275. doi:10.3389/fphar.2018.01275.

[36] Cheng T, Li X, Li Y, Liu Z, Wang R. Comparative Assessment of Scoring Functions on a Diverse Test Set. J Chem Inf Model 2009;49:1079–93. doi:10.1021/ci9000053.

[37] Li H, Peng J, Leung Y, Leung K-SK-S, Wong M-HM-H, Lu G, et al. The Impact of

Protein Structure and Sequence Similarity on the Accuracy of Machine-Learning Scoring Functions for Binding Affinity Prediction. Biomolecules 2018;8:12. doi:10.3390/biom8010012.

[38] Olier I, Sadawi N, Bickerton GR, Vanschoren J, Grosan C, Soldatova L, et al. Meta-QSAR: a large-scale application of meta-learning to drug design and discovery. Mach Learn 2018;107:285–311. doi:10.1007/s10994-017-5685-x.

[39] Bosc N, Atkinson F, Felix E, Gaulton A, Hersey A, Leach AR. Large scale comparison of QSAR and conformal prediction methods and their applications in drug discovery. J Cheminform 2019;11. doi:10.1186/s13321-018-0325-4.

[40] Ekins S, Clark AM, Dole K, Gregory K, Mcnutt AM, Spektor AC, et al. Data mining and computational modeling of high-throughput screening datasets. Methods Mol. Biol., vol. 1755, Humana Press Inc.; 2018, p. 197–221. doi:10.1007/978-1-4939-7724-6_14.

[41] Kim MT, Wang W, Sedykh A, Zhu H. Curating and Preparing High-Throughput Screening Data for Quantitative Structure-Activity Relationship Modeling. Methods Mol Biol 2016;1473:161–72. doi:10.1007/978-1-4939-6346-1_17.

[42] Soufan O, Ba-Alawi W, Magana-Mora A, Essack M, Bajic VB. DPubChem: A web tool for QSAR modeling and high-throughput virtual screening. Sci Rep 2018;8. doi:10.1038/s41598-018-27495-x.

[43] Liu S, Alnammi M, Ericksen SS, Voter AF, Ananiev GE, Keck JL, et al. Practical Model Selection for Prospective Virtual Screening. J Chem Inf Model 2019;59:282–93. doi:10.1021/acs.jcim.8b00363.

(** This study discusses the use of High-Throughput Screens to train and select machine-learning models to predict the activities of compounds on a protein-protein interaction target. Further, it shows that retrospective validations on this benchmark are highly predictive of what was subsequently observed on the prospective applications of the same models).

[44] Durrant JD, McCammon JA. NNScore: A Neural-Network-Based Scoring Function for the Characterization of Protein−Ligand Complexes. J Chem Inf Model 2010;50:1865–71. doi:10.1021/ci100244v.

[45] Durrant JD, McCammon JA. NNScore 2.0: A Neural-Network Receptor–Ligand

Scoring Function. J Chem Inf Model 2011;51:2897–903. doi:10.1021/ci2003889.

[46] Wang C, Zhang Y. Improving scoring-docking-screening powers of protein–ligand scoring functions using random forest. J Comput Chem 2017;38:169–77. doi:10.1002/jcc.24667.

[47] Lu J, Hou X, Wang C, Zhang Y. Incorporating Explicit Water Molecules and Ligand Conformation Stability in Machine-Learning Scoring Functions. J Chem Inf Model 2019;59:4540–9. doi:10.1021/acs.jcim.9b00645.

[48] Adeshina YO, Deeds EJ, Karanicolas J. Machine learning classification can reduce false positives in structure-based virtual screening. Proc Natl Acad Sci 2020:202000585. doi:10.1073/pnas.2000585117.

[49] Trott O, Olson AJ. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. J Comput Chem 2010;31:455–461. doi:10.1002/jcc.21334.

[50] Su M, Du Y, Yang Q, Wang R, Liu Z, Feng G, et al. Comparative Assessment of Scoring Functions: The CASF-2016 Update. J Chem Inf Model 2018;59:895–913. doi:10.1021/acs.jcim.8b00545.

[51] Durrant JD, McCammon JA. BINANA: A novel algorithm for ligand-binding characterization. J Mol Graph Model 2011;29:888–93. doi:10.1016/j.jmgm.2011.01.004.

[52] Cang Z, Mu L, Wei G-W. Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening. PLOS Comput Biol 2018;14:e1005929. doi:10.1371/journal.pcbi.1005929.

(* This paper introduces novel algebraic topology approaches for the description of small molecules bound to macromolecular targets, which led to highly predictive deep learning models for structure-based virtual screening).

[53] Li H, Leung K-S, Wong M-H, Ballester PJ. Improving AutoDock Vina Using Random Forest: The Growing Accuracy of Binding Affinity Prediction by the Effective Exploitation of Larger Data Sets. Mol Inform 2015;34:115–126. doi:10.1002/minf.201400132.

[54] Li N, Ainsworth RI, Wu M, Ding B, Wang W. MIEC-SVM: automated pipeline for protein peptide/ligand interaction prediction. Bioinformatics 2016;32:940–2.

doi:10.1093/bioinformatics/btv666.

[55] Wójcikowski M, Zielenkiewicz P, Siedlecki P. Open Drug Discovery Toolkit (ODDT): a new open-source player in the drug discovery field. J Cheminform 2015;7:26. doi:10.1186/s13321-015-0078-2.

[56] Ashtawy HM, Mahapatra NR. Descriptor Data Bank (DDB): A Cloud Platform for Multiperspective Modeling of Protein–Ligand Interactions. J Chem Inf Model 2018;58:134–47. doi:10.1021/acs.jcim.7b00310.

(*A comprehensive software package to build and validate machine-learning scoring functions. A stand-alone version can be freely downloaded).

[57] Peón A, Dang CC, Ballester PJ. How Reliable Are Ligand-Centric Methods for Target Fishing? Front Chem 2016;4:15. doi:10.3389/fchem.2016.00015.

[58] Li H, Peng J, Sidorov P, Leung Y, Leung KS, Wong MH, et al. Classical scoring functions for docking are unable to exploit large volumes of structural and interaction data. Bioinformatics 2019;35:3989–95. doi:10.1093/bioinformatics/btz183.

[59] Wójcikowski M, Kukiełka M, Stepniewska-Dziubinska MM, Siedlecki P. Development of a protein–ligand extended connectivity (PLEC) fingerprint and its application for binding affinity predictions. Bioinformatics 2019;35:1334–1341.

[60] Ballester PJ. Ultrafast shape recognition: method and applications. Future Med Chem 2011;3:65–78.

[61] Schreyer A, Blundell T. USRCAT: real-time ultrafast shape recognition with pharmacophoric constraints. J Cheminform 2012;4:27. doi:10.1186/1758-2946-4-27.