



HAL
open science

The Impact of Data on Structure-Based Binding Affinity Predictions Using Deep Neural Networks

Pierre-Yves Libouban, Samia Aci-Sèche, Jose Carlos Gómez-Tamayo, Gary Tresadern, Pascal Bonnet

► **To cite this version:**

Pierre-Yves Libouban, Samia Aci-Sèche, Jose Carlos Gómez-Tamayo, Gary Tresadern, Pascal Bonnet. The Impact of Data on Structure-Based Binding Affinity Predictions Using Deep Neural Networks. International Journal of Molecular Sciences, 2023, 24 (22), pp.16120. 10.3390/ijms242216120 . hal-04662295

HAL Id: hal-04662295

<https://hal.science/hal-04662295>

Submitted on 25 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



1 Article

2 The impact of data on structure-based binding affinity predic- 3 tions using deep neural networks

4 Pierre-Yves Libouban ¹, Samia Aci-Sèche ¹, Jose C. Gómez-Tamayo ², Gary Tresadern ² and Pascal Bonnet ^{1,*}

5 ¹ Institute of Organic and Analytical Chemistry (ICOA); UMR7311, Université d'Orléans, CNRS; Pôle de
6 chimie rue de Chartres - 45067 Orléans Cedex 2, France; pierre-yves.libouban@univ-orleans.fr

7 ² Computational Chemistry, Janssen Research & Development; Janssen Pharmaceutica N. V.; B-2340 Beerse,
8 Belgium; gtresade@its.jnj.com

9 * Correspondence: pascal.bonnet@univ-orleans.fr

10 **Abstract:** Artificial intelligence (AI) has gained significant traction in the field of drug discovery,
11 with deep learning (DL) algorithms playing a crucial role in predicting protein-ligand binding af-
12 finities. Despite advancements in neural network architectures, system representation, and training
13 techniques, the performance of DL affinity prediction has reached a plateau, prompting the ques-
14 tion of whether it is truly solved or if the current performance is overly optimistic and reliant on
15 biased, easily predictable data. Like other DL related problems, this issue seems to stem from the
16 training and test sets used when building the models. In this work, we investigate the impact of
17 several parameters related to the input data on the performance of neural network affinity predic-
18 tion models. Notably, we identify the size of the binding pocket as a critical factor influencing the
19 performance of our statistical models; furthermore, it is more important to train a model with as
20 much data as possible, than to restrict the training on only the high quality datasets. Finally, we
21 also confirm the bias in the typically used current test sets. Therefore, several types of evaluation
22 and benchmarking are required to understand models decision-making process and accurately
23 compare the performance of models.

24 **Keywords:** protein-ligand; binding affinities; deep learning

25 1. Introduction

26 The importance of *in silico* work in the drug discovery pipeline has been growing for
27 several decades. Since the 1980's, numerous drugs have been successfully marketed after
28 being initially designed with the help of computers [1]. Approaches for computer-aided
29 drug design, aiming to identify lead compounds, have steadily improved over time. In
30 structure-based drug design (SBDD), docking is a method that predicts the mode of
31 binding of a molecule into a pocket protein and the affinity of such molecules for the
32 protein target using a scoring functions. This method helps in identifying molecular hits
33 in drug design projects.. A cornerstone step in this process is to evaluate accurately the
34 binding affinity of the protein-ligand complexes. To this end, various scoring functions,
35 such as knowledge-based, empirical, and force field-based methods, have been devel-
36 oped [2]. The development of scoring functions has advanced further with the integra-
37 tion of machine learning models for bioactivity assessment. Recently, neural networks
38 have gained attention for predicting binding affinity of protein-ligand complexes. With
39 the advent of big data and the access to increased computing power, DL algorithms have
40 emerged as promising tools for prediction purposes. These algorithms harness the
41 structural information of protein-ligand complexes to predict binding affinities, often
42 outperforming other scoring functions [3]. There are also alternative methods for calcu-
43 lating absolute binding free energies, including MMGB(PB)SA [4] and LIE [5]. Addition-
44 ally, TI and FEP [6] can provide highly accurate predictions, typically within one order of
45

26 **Citation:** Libouban, P.-Y.; Aci-Sèche,
27 S.; Gómez-Tamayo, J.C.; Tresadern,
28 G.; Bonnet, P. The impact of data on
29 binding affinity predictions using
30 deep neural networks. *Int. J. Mol. Sci.*
31 **2023**, *24*, x.
32 <https://doi.org/10.3390/xxxxx>

33 Academic Editor(s):

34 Received: date

35 Revised: date

36 Accepted: date

37 Published: date



38 **Copyright:** © 2023 by the authors.
39 Submitted for possible open access
40 publication under the terms and
41 conditions of the Creative Commons
42 Attribution (CC BY) license
43 (<https://creativecommons.org/licenses/by/4.0/>).
44
45

46 magnitude in affinity, although it is primarily used for relative binding free energy cal-
47 culations. However, these methods rely on computationally expensive molecular dy-
48 namics simulations. Therefore, in virtual screening scenarios, less computationally in-
49 tensive approaches like deep learning (DL) models are favored. Nevertheless, despite the
50 implementation of new deep neural networks, the performance of the statistical models
51 is stagnating [7].

52 The performance with DL algorithms relies heavily on the amount of data available
53 to train the statistical models. Unfortunately the amount of data available for the predic-
54 tion of binding affinity is relatively low in comparison to other application domains
55 where DL has been successfully applied, like computer vision [8]. Indeed, for binding
56 affinity predictions, models can be trained with the 3D structure of protein-ligand com-
57 plexes, which are determined by crystallography, NMR or cryogenic electron microscopy
58 (cryo-EM). On top of this, it is required to perform biophysical experiments, like surface
59 plasmon resonance (SPR) or isothermal titration calorimetry (ITC), or more common bi-
60 ochemical assays, in order to evaluate the binding affinity of the complexes. All these
61 experiments require extensive work therefore complicating the generation of new reli-
62 able data in this field.

63 We decided to evaluate the different variables related to the data to assess their
64 impact on the performance. First of all, a crucial question is to evaluate the minimum
65 amount of data necessary to achieve satisfactory performance. Would 10,000 complexes
66 be enough or at least 100,000 are required etc.? To add to these considerations, it is im-
67 portant to keep in mind that increase in the data complexity, leads to higher data size
68 requirements. This is especially true for 3D structural data, which are of higher com-
69 plexity in comparison to most usual deep learning applications. The current state of the
70 art structural-based affinity prediction models are typically trained on the PDBbind [9]
71 dataset. This dataset comprises 3D structures of protein-ligand complexes with known
72 binding affinity (K_d , K_i or IC_{50}). In the case that several forms of binding data were
73 available for a complex, K_d was selected over K_i , and K_i was selected over IC_{50} . This da-
74 taset contains 19,443 complexes in its current version (v.2020). Despite the size of the
75 PDBbind increasing every year, having more data is not translated into better perfor-
76 mance for the underlying models [7]. One of the main reasons is that the data lacks large
77 series of molecules targeting the same protein, as well as having the same molecule in
78 complex with several proteins. It is proposed that the sparsity of the protein-ligand ma-
79 trix makes it harder for DL to learn from interactions. On top of this, some teams decided
80 to focus on training on complexes of better quality instead of training on all the data
81 available. In order to validate this approach, we analyzed previously reported models
82 trained on the whole PDBbind, and solely PDBbind's high quality subset known as the
83 refined set. Furthermore, we have trained several models with Pafnucy [10], a
84 well-known CNN for the prediction of binding affinities, on both datasets.

85 Protein-ligand complexes are dynamic, and the binding free energy as ligand passes
86 from solvent to protein represents the energy difference between the ensemble of bound
87 and solvated states. To accurately predict the binding affinity of a complex, several fac-
88 tors have to be taken into account like the association/dissociation kinetic constants for
89 the prediction of K_d as well as the dynamic interactions between the ligands and the
90 proteins. Several studies were performed to predict K_d or k_{off} using molecular dynamics
91 simulations ([11,12]). Therefore, the models are only based on partial information, they
92 are single snapshots that although capture some experimentally favorable state, may still
93 be incomplete. Since models use only the interactions between the ligands and the pro-
94 teins, they are generally trained on proteins' pockets instead of using the whole protein.
95 Pockets have already been calculated for the complexes contained in the PDBbind and
96 are readily available when downloading the database. This removes the need for users to
97 detect new pockets by themselves. Nonetheless, binding affinity will be impacted by
98 conformational information from the ligand and protein local environment [7,13].
99 Therefore, pockets of different sizes can contain more or less information useful for get-

ting performant models. Here, we investigated the impact of the pocket's size on the binding affinity prediction.

Other considerations related to the data are also investigated in this study. Notably, the difficulty to predict the binding affinity of peptides and the impact on the DL models performance of using a training dataset including peptides or not. These difficulties stem from the higher degrees of freedom of peptides in comparison to small molecules. This leads to increased complexity of the entropic part when calculating free energies [14]. When training on the PDBbind, it appears that predicting the affinity of peptides becomes a challenging task. Therefore, some published models were developed by training only on nonpeptide ligands [15]. Nonetheless, some non-structural datasets are specifically designed for antibacterial peptides, and models trained on these datasets have shown good performance [16].

Another aspect pointed out in several recent publications [7,17] is related to DL models memorizing ligand and protein information instead of learning from the interactions. Here we have decomposed this, by training neural networks only on proteins or ligands and carrying out the prediction, to evaluate the bias in their predictions. We compared the performances of 3 well-known DL model predicting binding affinities, GraphBar, Pafnucy and OctSurf.

Overall, we find that it is important to train on as much data as possible, while even using complexes deemed of lower quality. Moreover, the size of the pocket does matter for the ability of the model to predict the binding affinity. The performance improves upon reaching a certain size (12 Å around the ligand); increasing pocket size further more will not improve the performance. On top of this it is difficult to predict peptides, even by training only on peptides. Finally, we point out that there is a big discrepancy on the ability of neural networks to learn from the interaction. Some models will heavily drop in performance by removing one of the 2 partners from the complex, while other rely on the memorization of bias in the data to carry out a prediction.

2. Materials and Methods

2.1. Datasets

The PDBbind dataset (<http://www.pdbbind.org.cn>) [9] was used to train the different models. It contains protein-ligand complexes with known binding activity. In its current version (v.2020), 19,443 complexes are available. In this publication, three versions of the PDBbind were used:

- The version 2016 that contains 13,308 protein-ligand complexes
- The version 2018 that contains 16,151 protein-ligand complexes
- The version 2019 that contains 17,679 protein-ligand complexes

The complexes present in the PDBbind are selected from the Protein Data Bank (<http://www.rcsb.org/>) [18]. Several modifications are added to these complexes, *e.g.*, the biological assembly of complexes are recreated, ligands' atoms and bonds are corrected; for the detail of all modifications, please refer to the "readme" provided with the PDBbind.

The PDBbind encompasses three sets of data: the general set, the refined set and the core set. The general set contains the totality of the dataset. The refined set is a subset made of 4,852 complexes (for the version 2019) selected on the basis of the following quality criteria [19]:

- Crystallographic structures, with a resolution of 2.5 Å maximum
- Complete ligands/pockets (without missing atoms) and without steric clash with the protein
- Noncovalently bound complexes, no nonstandard residues at a distance <5 Å from the ligand
- No other ligands are present in the binding site, *e.g.*, cofactors or substrates

- Binding affinity evaluated in K_i or K_d , and with a pK_i between 2 and 12
- Ligands with a molecular weight of less than 1000, less than 10 residues for peptides
- With ligands made only of the following atoms: C, N, O, P, S, F, Cl, Br, I, and H
- The buried surface area of the ligand is higher than 15% of the total surface area of the complex

The core set is broadly used as a test set to compare models' performance. Only two versions are available, the version 2013 which is composed of 195 complexes [20,21] and the version 2016 comprising 285 complexes [22]. Both core set have 107 complexes in common. The core set 2016 is made of 57 clusters of 5 complexes belonging to the same protein family. These groups are obtained by clustering complexes based on sequence similarity of 90% minimum.

In this study, peptides were flagged among the ligands coming from PDBbind's complexes. We detected the peptides by looking for ligands having in their mol2 files at least one atom named "CA", "CB", "CD", "CE", "CG", "CZ", "CA1", "CA2", "CB1", "CB2", "CD1", "CD2", "CE1", "CE2", "CG1", "CG2", "CZ1" or "CZ2". On top of this, we analyzed the PDBbind list of ligand names and flagged as peptides all the ligands containing "mer" in their name. Finally, ligands wrongly labeled as peptides were removed, by keeping only ligands matching with the following smart, which represent a peptide bond: [*]([NX3H2,NX4H3+]),[*]([NX3H](C)(C))][CX4H]([*])[CX3](=[OX1])[OX2H,OX1-,N]. By doing so, we were able to detect 2,915 peptides in the PDBbind (v.2019). The list of peptides curated from the PDBbind v.2019 was made available as a supplementary material (Table S1).

We used the pockets provided by the PDBbind to evaluate the impact on the performance of:

- The dataset sizes (general set or refined set)
- The types of ligands (peptide or nonpeptide)
- Using only ligand or only protein

We also created our own pockets using Pymol. Residues around the ligands were selected to create pockets. The pockets were constructed with different sizes: 6 Å, 8 Å, 10 Å, 12 Å and 14 Å. Two types of pockets were created, by selecting residues at a specific distance from:

- All the atoms of the ligands
- The center of geometry (CoG) of the ligands (Figure 1)

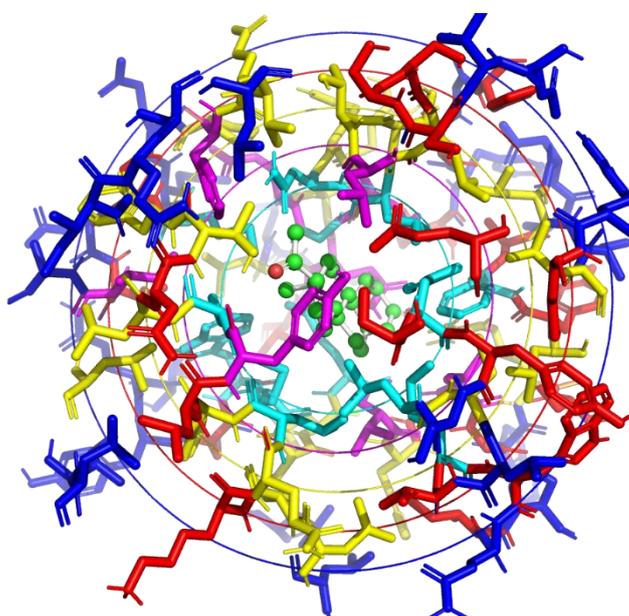


Figure 1. Pockets created and visualized with Pymol. The ligand is displayed in green. Residues are colored in cyan, purple, yellow, red and blue, according to their distance from the CoG of the ligand, respectively at 6 Å, 8 Å, 10 Å, 12 Å and 14 Å.

2.2. neural networks

Protein-ligand complexes can be used to train statistical models in many ways. The 3D representations of these complexes can be either 3D structures or 3D surfaces [23], which can be implemented in various ways, including 3D grids, point clouds, 3D graphs, or mesh [23,24]. Several types of neural networks were developed to handle these representations of the data, such as the convolutional neural networks (CNN) and the graph neural networks (GNN). The CNN are used on 3D grids which discretize the space in voxels of around 1 Å³. Then CNN perform convolutions over these voxels to extract the meaningful information for the prediction of binding affinities. The GNN are applied on graphs, where atoms serve as nodes and bonds as edges. In the case of graph convolutional network, the useful information stored in nodes and edges is extracted by performing graph convolutions.

Only previously published binding affinity neural networks approaches were used in this work. For the purpose of this study, we selected two CNN: Pafnucy [10] and OctSurf [25], both employing grids to discretize 3D structures and 3D surfaces, respectively. Additionally, we evaluated GraphBAR [26] that is a graph convolutional neural network. Here, we briefly describe each of them. The full description of the neural networks can be found in the original publications.

Pafnucy is a 3D convolutional neural network published in 2018. It uses the 3D coordinates of atoms, and performs convolutions on voxels of 1 Å³. In this paper, we generally used boxes of 21 Å, and modified the size of the box when different size of pockets were used. 19 features were used to describe an atom:

- 9 bits (one-hot or all null) encoding atom types: B, C, N, O, P, S, Se, halogen and metal
- 1 integer (1, 2, or 3) with atom hybridization: *hyb*
- 1 integer counting the numbers of bonds with other heavyatoms: *heavy_valence*
- 1 integer counting the numbers of bonds with other heteroatoms: *hetero_valence*
- 5 bits (1 if present) encoding properties defined with SMARTS patterns: hydrophobic, aromatic, acceptor, donor and ring
- 1 float with partial charge: *partial charge*
- 1 integer (1 for ligand, -1 for protein) to distinguish between the two molecules: *moltype*

This neural network uses data augmentation by learning from systematic rotations of complexes. The systematic rotations are obtained by performing the 24 rotations of the cube on each structure. The data augmentation with systematic rotations allows the models to be more robust since the models are independent of the orientations of the ligands and the proteins.

Here are the reported performance of Pafnucy trained on the pockets provided by the PDBbind 2016:

- Core set 2013: correlation coefficient of 0.70 taken from Stepniewska-Dziubinska *et al.* [10].
- Core set 2016: correlation coefficient of 0.78 take from Stepniewska-Dziubinska *et al.* [10].

We replicated the results of Pafnucy by using the code available here: <https://gitlab.com/cheminflBB/pafnucy>.

OctSurf is a 3D convolutional neural network published in 2021. It requires an elaborate data preparation before it can be used as input for the neural network. First, the 3D coordinates of atoms are turned into point clouds [27] representing their van der

236 Waals surfaces. Then the point clouds are rasterized into an octsurf which is a volumetric
237 representation based on octree data structure [28]. An octsurf is composed of octants, on
238 which are performed the convolutions. The octants can have variable sizes. This allows
239 for having octants of different sizes in the same octsurf, describing more or less precisely
240 different parts of the octsurf. Therefore, it is possible to have big octants in the solvent
241 and smaller ones (of 1 Å for example) in contact of the proteins and ligands. This way, we
242 can accelerate the convolution process, while keeping good performance.

243 The description of octants uses the 19 features described in Pafnucy. On top of that 5
244 more features were added to reach a total of 24 features:

- 245 • The hydrogen atom type
- 246 • Van der Waals atomic radius
- 247 • A normal vector with three coordinates direction, describing surface curvature and
248 shape complementarity

249 Data augmentation was performed by randomly rotating and translating the surface
250 points, reaching 40 octsurfs for each complex.

251 In the publication, OctSurf reached a correlation coefficient of 0.79 [25] on the core
252 set 2016 by training on the pockets provided by the PDBbind 2018.

253 The code of OctSurf is available here:
254 <https://github.uconn.edu/mldrugdiscovery/OctSurf>

255 GraphBAR is a graph convolutional neural network published in 2021. Graphs were
256 created with atoms as nodes, and bonds as edges. Node characterization reuses only 13
257 features established by Pafnucy, therefore not using the 5 properties encoded by
258 SMARTS patterns (hydrophobic, aromatic, acceptor, donor and ring).

259 Bonds are summarized in an adjacency matrix having a size of NxN, with N being
260 the number of nodes. In the adjacency matrix, the adjacent atoms are defined by a dis-
261 tance maximum of 4 Å for inter-molecular distances, and 2 Å for intra-molecular dis-
262 tances. It is possible to train the neural network with up to 8 adjacency matrices. If the
263 number of adjacency matrices is increased, the distance range covered by each is re-
264 duced. For example, in the case of using only one matrix, this one would cover interac-
265 tions up to 4 Å. While in the case of using two adjacency matrices, the first one would
266 account for the interactions up to 2 Å, and the other one deals with the interactions from 2
267 to 4 Å. The model established with two matrices achieved the best performance.

268 For data-augmentation purpose, docking was performed and best poses with less
269 than 3 Å of RMSD were selected, up to 3 poses.

270 GraphBAR was trained on the PDBbind 2016, while discarding the complexes
271 (pocket + ligand) containing too many atoms (>200 atoms). The models achieved coeffi-
272 cient correlations of 0.76 on the core set 2016 and 0.70 on the core set 2013. The da-
273 ta-augmentation provided little improvements on the core set 2016 with a coefficient
274 correlation of 0.78, and no improvement were measured for the core set 2013.

275 Performance was replicated using the code available here:
276 <https://github.com/jtson82/graphbar>

277 We carried out each experiment by replicating the training 10 times. All model rep-
278 licates were performed in the same conditions, *i.e.* with the same neural network, the
279 same hyper-parameters, the same input data, but different weights (randomized seeds) at
280 the initialisation of the neural network. The results were averaged and the standard de-
281 viation was calculated, in order to compare the performance of each experiments.

282 Models were trained with our laboratory cluster, on graphics processing unit (RTX
283 2080 and RTX 3090).

284 2.3. metrics

285 The model performance was evaluated by predicting the binding affinity of each
286 complexes of test sets and comparing the results with real values. Prediction error was
287 measured with the root mean square error (RMSE).

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^N (x_i - \hat{x}_i)^2}{N}} \quad (1)$$

The correlation between predicted binding affinity and the experimentally measured binding affinity were assessed with the Pearson correlation coefficient (R) and its standard deviation (SD).

$$R = \frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2 \sum (y_i - \bar{y})^2}} \quad (2)$$

Statistical plots were performed with the library statannot (<https://pypi.org/project/statannot/>). Assuming normal distribution, all comparison were performed with independent sample Student t-test with Bonferroni correction. Following p-values correspond to the annotations on the plots:

ns: $5.00e-02 < p \leq 1.00e+00$
 *: $1.00e-02 < p \leq 5.00e-02$
 **: $1.00e-03 < p \leq 1.00e-02$
 ***: $1.00e-04 < p \leq 1.00e-03$
 ****: $p \leq 1.00e-04$

3. Results

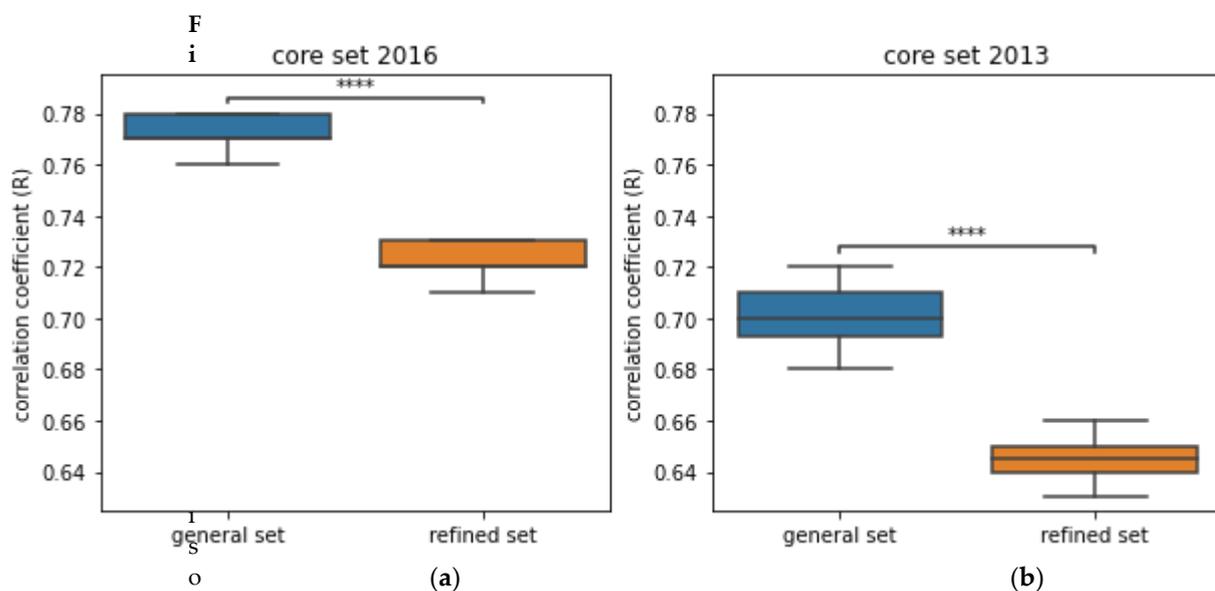
3.1. Impact of the amount of data on the performance

To reach good performance with DL algorithms, it is expected that more data is beneficial and that a high amount of data is a requirement to begin. In PDBbind (v.2019), the general set contains 17,679 protein-ligand structures. The refined set is a subset of 4,852 complexes selected from the general set based on quality criteria. A previously published study suggested that training on the general set of the PDBbind does not improve the performance in comparison to training only on the refined set [7]. While other studies [29-31] pointed out that they achieved better performance by training on the general set rather than only on the refined one.

In order to explore this further, we have trained Pafnucy [10] with the PDBbind general set and with only the refined set. Pafnucy was set up to perform convolutions over voxels of 1 \AA^3 and on a box of 21 \AA^3 centered on the ligand.

The models were applied to 2 test sets comprised of 285 and 195 complexes and referred to as core set 2016 and core set 2013. The complexes from the test sets were not used in training. Nonetheless as reported in GIGN [32], all the proteins and a third of the ligands from the test set are also used in the training set. In other words, none of the test set complexes are present in the training set, but the models have encountered at least one of the binding partners during training. As a result, we can anticipate biased results when making predictions on the test sets. The models might rely on specific data patterns to make predictions. For instance, certain ligands may consistently display either high or low affinity, regardless of the partner protein. This pattern could be exploited by the model, leading to artificially inflated performance. Analyzing further these sets, we found out that the distribution of the molecular weight of ligands is similar for the test set and the training set (Figure A1). The same can be said about the shape of the ligands, although there is a lack of spherical ligands in the test set (Figure A2). In addition, ligands with extreme affinity are over-represented in the test set in comparison to the training set (Figure A3). This can be a possible explanation for why current networks [7] predict over a small affinity range and therefore tend to fail predicting extreme affinities values of the test set.

When assessing performance, we compare the correlation between predicted and experimental activity using the Pearson correlation coefficient (R).



of the performance of Pafnucy [10], after being trained on the general or the refined set of the PDBbind 2019. 10 models were trained on each dataset. (a) Performance is evaluated on the core set 2016; (b) performance is evaluated on the core set 2013. For both core sets, performance by training on the general set significantly outperform the performance by training on the refined set. Following p-values correspond to the annotations on the plots:

ns: $5.00e-02 < p \leq 1.00e+00$

*: $1.00e-02 < p \leq 5.00e-02$

** : $1.00e-03 < p \leq 1.00e-02$

***: $1.00e-04 < p \leq 1.00e-03$

****: $p \leq 1.00e-04$

Models trained on the general set perform better than the one trained on the refined set when applied to the frequently used test sets: core set 2016 and core set 2013, Figure 2. These results are in accordance with a previously published comparison of performance of 11 neural networks [29]. For all, these neural networks the RMSE and MAE is lower when trained on the general set instead of the refined set. Likewise, the neural networks PointTransformer [30], DeepAtom [31] and the GNINA CNN v2018 [33] perform better by training the general set. These results differ a bit with 3D fusion [13], which is a model composed of a 3D-CNN and of a spatial graph CNN (SG-CNN). In this case, it seems that 3D-CNN perform better by training on the refined set only, unlike the SG-CNN. Overall, this confirms that having more data, albeit of lower quality, gives better performance. One might question whether the observed performance enhancement obtained by training on the general set can be attributed to proper learning. Did the model develop a more profound comprehension of the interactions, or simply enhance its ability to memorize patterns within ligands and proteins?

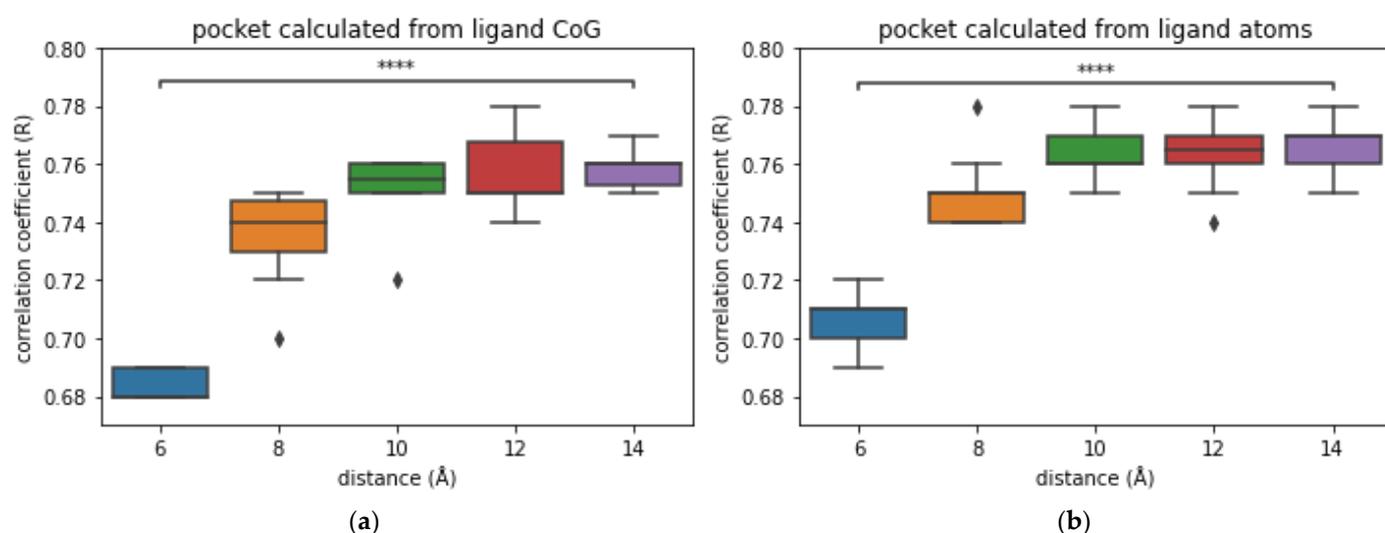
These results also showcase that there might be a misunderstanding in the field of cheminformatics about the quality of data. Indeed, having data of high quality is very important for carrying out good predictions. Therefore, several teams have decided to train their models only with the refined set, which is considered to be of higher quality. Contrary to that belief, we think that the data that is not in the refined set can be still considered as useful data. Indeed, we can compare this data to fuzzy images in image recognition. These images are essential for the robustness of the models in real-life condition, since in this case not all images presented to the model would be clear. For image recognition, the saying “garbage in, garbage out” indicate that the images have been

385 badly labeled; therefore impeding the training process and resulting into models with
386 worse performance. In the case of protein-ligand binding affinity predictions, the labeling
387 task of the data has been handled by the team that conceived and update the PDBbind.
388 They have been manually looking into publications to report the experimentally evalu-
389 ated binding affinities of complexes [34]. On top of this, the binding affinities obtained
390 were compared to those gathered from MOAD [35], which is another database compris-
391 ing protein-ligand complexes with binding affinities, in order to reduce the error rate.

3.2. Size of pockets

392
393 The GNN are ideally designed to handle the data representing protein-ligand com-
394 plexes. Indeed, this data is made of nodes (atoms or residues) and bonds (interactions
395 between molecules or intramolecular interactions). Thanks to this design, GNN focus on
396 the important information, being therefore efficient from a computational point of view.
397 This is not the case for CNN that are quite computationally intensive as convolutions are
398 performed on all the voxels of the 3D images. A lot of these voxels do not contain any
399 information about the protein or the ligand, as they are located in the solvent. This in-
400 creases the calculation time for no performance gain. Although some methodologies have
401 been developed to avoid these hindrances [25], the most common way to reduce the
402 computational requirements while maintaining good performance, is to only train mod-
403 els from the pockets instead of using the whole proteins.

404 The PDBbind provides pockets to the users for conveniences. They are constituted of
405 all residues within a distance of 10 Å from the ligand. As the amount of data available for
406 the training increases with the size of the ligand and therefore the size of the pockets, we
407 have investigated the influence of the pocket size on the performance of trained models.
408 For this purpose, we have created pockets of different sizes and trained 10 models per
409 size with Pafnucy. We calculated 2 types of pockets by selecting the residues located
410 within a specific distance measured from all the atoms, or from the center of geometry
411 (CoG), of the ligands. The size of pockets was defined by the residue detection distances,
412 ranging from 6 to 14 Å. The size of the box used in Pafnucy is equal to $2 \times \text{detection distance} + 1 \text{ Å}^3$.
413



414 **Figure 3.** Comparison of the performance of models trained with different sizes and types of
415 pockets. For each size, 10 models were trained and tested on the core set 2016. (a) Models trained
416 with pockets made of residues located within a specific distance from the center of geometry of the
417 ligands; (b) models trained with pockets created with residues located within a specific distance
418 from all atoms of the ligands.

419 For both type of pockets, there is a significant difference in the performance of
420 models trained on pockets of 6 Å and pockets of 14 Å (Figure 3). This is mostly due to the
421 fact that there is more information available in bigger pockets. Therefore, it is advised to
422 use pockets of 10 Å over pockets of 6 Å for training models, regardless of the type of
423 pockets used.

424 Nonetheless, there is very little improvement in term of performance between 10 to
425 14 Å. Thus, there is no interest in using bigger pockets than 10 Å. A compromise is re-
426 quired between using small pockets that do not contain enough information and big
427 pockets that are computationally more expensive, while not adding useful information.

428 Most of the interaction types fall within a range of 6 Å, thus it is difficult to under-
429 stand why a pocket size of 6 Å is not sufficient to predict accurately the binding affinity.
430 We think that this can be due to the bias in the data, in which case increasing pocket size,
431 and therefore adding more amino acids would help the model in memorizing and rec-
432 ognizing pattern in the protein. We could be tempted to think that if we keep increasing
433 the size of pockets, the performance would continuously improve. Although this does
434 not appear to be the case. Hence there might be a limit to how much the bias in the data
435 can artificially improve the performance.

436 Apart from the hypothesis of increased bias in the input data, there is an alternative
437 explanation related to the featurization of protein-ligand interactions [36]. Pafnucy de-
438 scribes ligand and protein atoms using 19 atomic features, and the interactions are not
439 explicitly encoded. In this case, the model could detect a hidden influence of amino acids
440 that are not in direct contact with the ligands. Therefore, the model would be able to in-
441 terpret some long-distance indirect interactions that are not easy for humans to decipher.
442 In this case, the limit in performance reached by using pockets of 10 Å would mean that
443 the amino acids added with bigger pockets are too distant from the ligand to influence it
444 in an indirect fashion. Further investigations are required to confirm or refute these hy-
445 pothesis.

446 Our limitation in interpreting such results is mostly due to the black box nature of
447 DL algorithm. We do not know the underlying reasons for a given prediction. By using
448 these algorithms on the FEP dataset [37], which contains chemical series of highly similar
449 molecules targeting the same protein with different affinities, it should help in inter-
450 preting model performance. Additionally, some methods were developed to alleviate the
451 black box issue, like the layer-wise relevance propagation [38,39], gradient based meth-
452 ods [40], or by masking atoms [41]. Such methods would be useful to better understand
453 the decisions taken by the model which leads to the prediction.

454 3.3. Peptide vs nonpeptide

455 Some neural networks were applied on protein-ligand complexes containing spe-
456 cific types of ligands. PointTransformer [30] was trained on the PDBbind 2016 of which
457 590 complexes, labeled as involving peptides, were removed.

458 Ahmed *et al.* developed a model by training only on proteins in complex with
459 nonpeptides [15]. They created their own dataset by looking into the PDB for pro-
460 tein-ligand complexes with:

- 461 • Crystallographic complexes with a resolution lower than 2.5 Å
- 462 • Known binding affinity (Kd/Ki)
- 463 • Ligand that does not have protein chain, and are not DNA/RNA

464 This selection resulted in a dataset of 4,041 complexes. By using their neural network
465 called DEELIG, they obtained a model that achieved a correlation coefficient of 0.889 on
466 the PDBbind 2016 core set. These results are encouraging, and it seems worth looking
467 into training models with only peptides and without them.

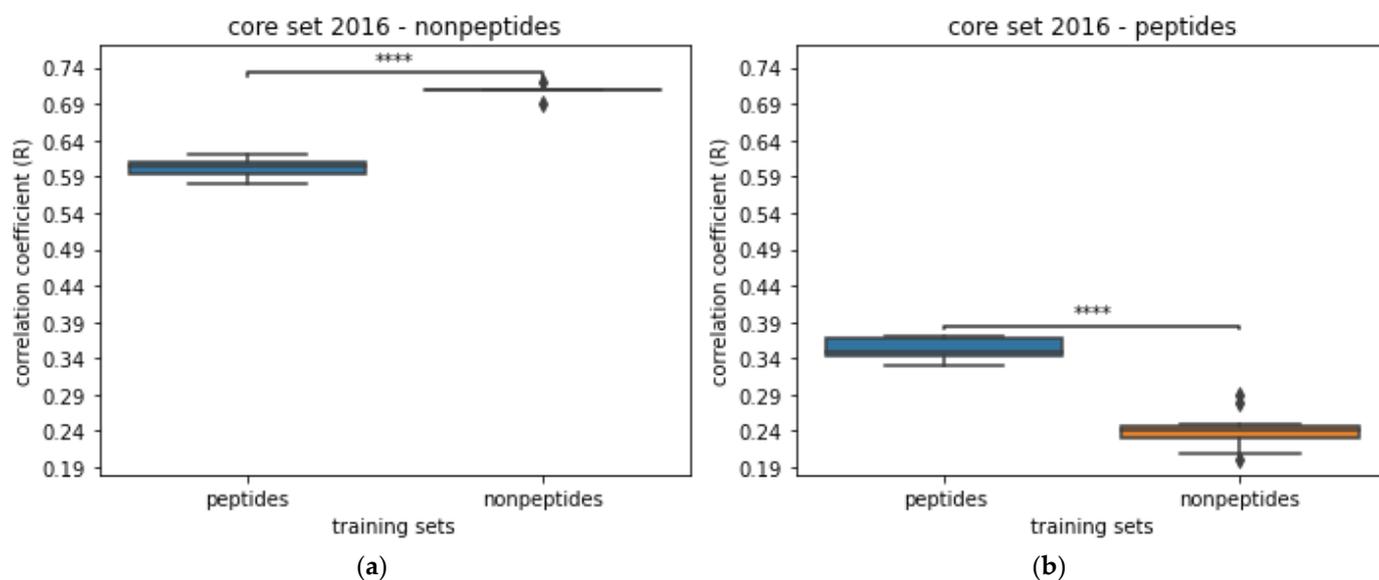
468 To evaluate the impact of training only with or without peptides, we flagged the
469 complexes with peptide from the PDBbind. Indeed, among the numerous rules that the
470 PDBbind established in order to select protein-ligand complexes, it has been decided that

471 peptides having 20 residues or less would be considered as ligands [42]. Therefore, we
472 have detected 2,915 complexes interacting with peptides among the 17,679 complexes of
473 the PDBbind (v.2019).

474 By using Pafnucy, models were trained with complexes interacting with peptides or
475 with complexes interacting with nonpeptides. As the dataset of protein-nonpeptide (PN)
476 complexes is larger than the dataset of protein-peptide (PP) complexes, we randomly
477 subsampled the dataset of PN complexes in order to have datasets of same size. We
478 trained models by training on each of the even size datasets. We obtained a model trained
479 on the PN dataset, and a model trained on the PP dataset. The performance of models
480 was evaluated on the core set 2013 and 2016 (Figure A4). Performance was significantly
481 better by training on PN complexes. Subsequently, we compared the performance of
482 models by evaluating them on each type of molecules from the core set 2016. Therefore,
483 we tested them only on the PN complexes, and only on the PP complexes (Figure 4).

484 Unsurprisingly, in comparison to the prediction on the whole core set 2016, we see
485 that the prediction gap increases a bit when predicting only on PN complexes. This can
486 be also explained by the fact that all proteins from the PN test set are present in the PN
487 training set, while 40% of them are not in the PP complexes training set. On top of it, 30%
488 of ligands from the PN test set are in the PN training set, and there are none in the PP
489 training set.

490 As for the prediction carried only on the PP complexes, although the performance of
491 models trained with PP complexes lowers a bit, the drop in performance is more drastic
492 for the model trained on PN complexes. Therefore, it seems that there is information
493 contained in the dataset of PP complexes useful to predict the PP complexes from the
494 core set 2016, albeit the predictions were carried out only on 19 complexes. We can point
495 out that 50% of the ligands are in the PP training set, while none are in the PN training
496 set.
497



498 **Figure 4.** Comparison of the performance of models trained with peptide-protein complexes and
499 with nonpeptide-protein complexes. Models were trained with Pafnucy on 2,383 complexes and
500 validated on 492 complexes. (a) Performance evaluated on 266 complexes with nonpeptides from
501 the core set 2016; (b) performance evaluated on 19 complexes with peptides from the core set 2016.

502 We explored the chemical space of the PDBbind to better understand the difference
503 in performance between models trained on PN and PP complexes, by performing a
504 principal component analysis (PCA) on the ligand of the complexes from the PDBbind
505 dataset. This allows us to compare the distribution of peptides and nonpeptide ligands

506
507
508
509
510
511

(Figure 5). The descriptors used to characterize the ligands were selected based on the literature [43], then the correlated descriptors were removed. The following 5 descriptors were used to carry out the PCA: hydrophobicity (LogP), Topological Polar Surface Area (TPSA), Fraction of SP³ hybridized Carbon (FCSP³), Number of Aromatic Rings (NAR), and Molecular Weight (MW).

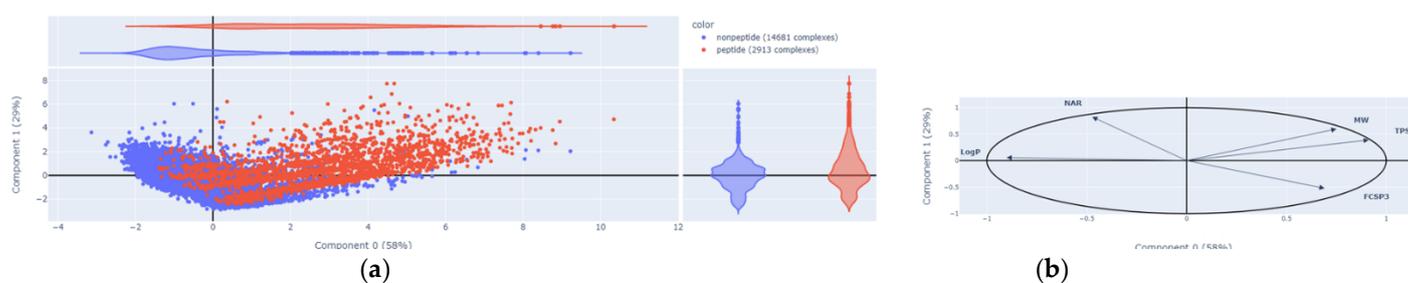
512
513
514

Figure 5. Principal component analysis applied on the ligands of the PDBbind dataset. Peptides were colored in red, while the rest of the ligands are displayed in blue. (a) Plot of the individuals; (b) correlation circle.

515
516
517
518
519
520
521
522
523

The PCA displays 87% of the variance of the data. It appears that the 2 populations of ligands are well separated. These results showcase the difference between peptides and small molecules, which help explain the lower performance from training with complexes involving only one type of ligand and predicting on the other type. Furthermore, the peptides are known to have high degrees of freedom especially due to the peptide bonds [14]. This increased flexibility results into high level of entropic energy, which needs to be taken into account when carrying out free energy prediction. Consequently, the evaluation of such values is very challenging. This can be an explanation for the poor performance of models in predicting the binding affinity for PP complexes.

524
525
526
527
528
529
530

We also evaluated the performance of models trained only on PN in comparison to training with both ligands mixed. Contrary to what we expected, it seems that training only on PN complexes does not improve the performance of the models (Figure A5). This comes as a surprise as we anticipated to obtain better performance in a similar fashion to DEELIG [15]. An explanation for the very high performance ($R=0.889$) obtained by DEELIG is that 68% of the test set complexes were used for the training, therefore skewing the evaluation of performance.

531
532
533
534
535
536
537

Nonetheless, even if it is better to train on the maximum amount of data as possible, there are promises to develop some local models focused on specific type of ligands. This is a practice less common than creating local models based on the type of proteins involved, but that can lead to interesting results. Moreover, it would be worth investigating transfer learning on such cases. For example, general models would be developed by learning general rules on the maximum amount of data, and then be specialized on predicting the binding affinities of peptides for example.

538
539
540
541
542

Once again, these results should be interpreted with caution, as there are strong indications of bias in the test set. For example, as we pointed out previously, all the protein families from the test are also present in the train set. The same issue applies to the ligands from the test set, with at least 30% of them being also in the train sets but bound to different proteins.

543

3.4. Replication of results

544
545
546
547

Most neural networks are non-deterministic. This behavior leads to variation in the performance of models trained with the same neural network and the same data. Indeed, several factors influence the variability, one of them being that initial weights are assigned randomly across the neural network at the beginning of the training. Due to the

randomized assignment of weights, the model is more likely to fall into certain local minima, creating uncertainty for the estimation. One way to overcome this issue is to modify the learning rate during the training, by using learning rate scheduler, and therefore getting out of local minima. The other solution is to train several model replicates, to increase the chances of having a model that did not fall into a local minimum. In any case, it is still necessary to carry out ensemble approaches [44] in order to accurately evaluate model performance and replicability. This implies training several models, averaging their performance and evaluating the standard deviation. This was done in the publication of OctSurf, where each value was averaged from 5 models. For this study, we replicated the results of 3 neural networks (Pafnucy [10], GraphBAR [26] and OctSurf [25]) and evaluated their averaged performance by training 10 models each time (Table 1).

Table 1. Replication of results from 3 neural networks (Pafnucy, GraphBAR and OctSurf) compared to the results presented in their respective publications. Models are evaluated based on their correlation coefficients and RMSE on the PDBbind core set 2016 (test set of 285 complexes).

Neural networks	Results from publication		Results from replication	
Pafnucy	R = 0.78 ¹	RMSE = 1.42 ¹	R = 0.77 SD = 0.01 ¹	RMSE = 1.41 SD = 0.01 ¹
GraphBAR	R = 0.76 ¹	RMSE = 1.44 ¹	R = 0.76 SD = 0.02 ³	RMSE = 1.43 SD = 0.03 ³
OctSurf	R = 0.79 ± 0.01 ²	RMSE = 1.45 ± 0.02 ²	R = 0.79 SD = 0.01 ²	RMSE = 1.46 SD = 0.03 ²

Training on: ¹ PDBbind v2016 (13,308 complexes); ² PDBbind v2018 (16,151 complexes); ³ PDBbind v2019 (17,679 complexes)

We were able to reproduce the performance displayed in the publication of each neural networks.

All the standard deviations (SD) have low values like 0.01 or 0.02. Nonetheless, a SD of 0.02 means that, with GraphBAR, it is as likely to get models with a correlation coefficient of 0.74 as of 0.78 on a similar test set. As this is relatively a big difference in term of performance, we think that deep ensemble averaging [45] should always be applied when publishing the results of training models with a neural network. Although this is computationally intensive, it gives more reliable expectations for people re-using the same neural network, as well as preventing bias like selecting the best model and publishing its results as representative of the neural network performance.

Another use of model replicates is to build ensemble models. Instead of measuring the coefficient correlation for each model and calculating the mean and the standard deviation, it is possible to calculate the mean prediction for each sample and then to calculate the correlation coefficient. This methodology has already been applied for several deep learning models like PIGNet [46] and in Francoeur *et al.* [33]. It leads to some small gain of performance, for example by using this methodology, Pafnucy and GraphBAR get an R = 0.79. As for their RMSE, Pafnucy improve from 1.41 to 1.38 and GraphBAR from 1.43 to 1.37. Such consensus methods are therefore a good way of improving performance while being less subject to variations.

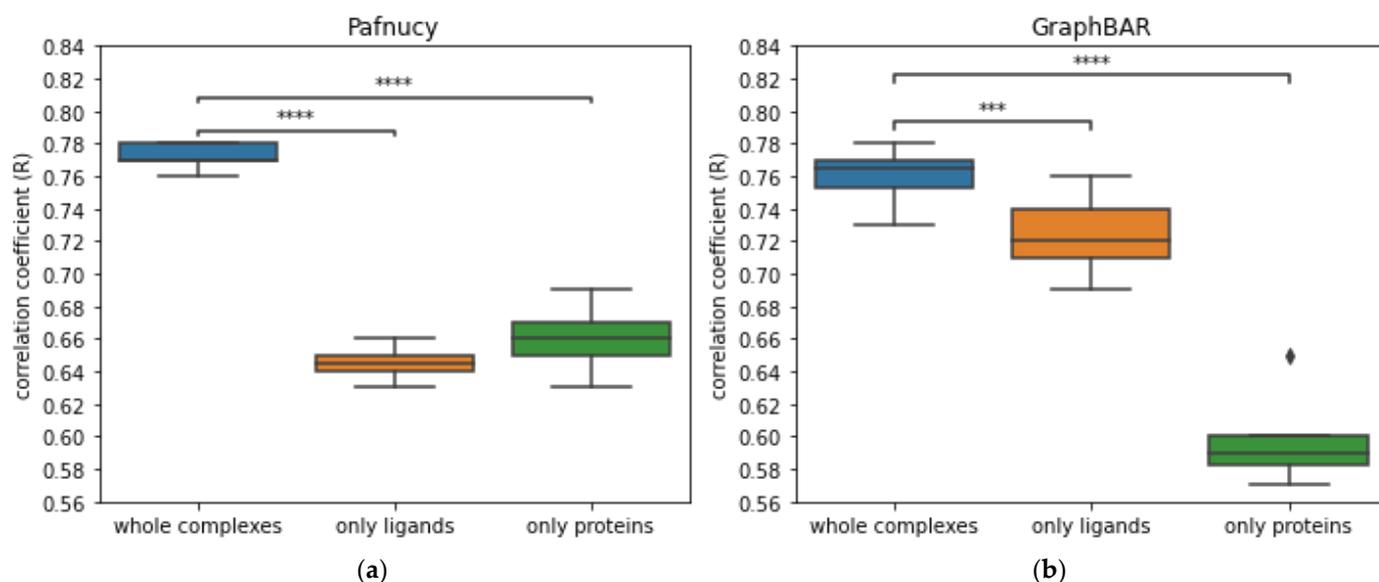
3.5. Learning from ligand only, protein only or interactions

Achieving good performance on a test set is the primary goal in model development, but it is also necessary to verify if such high performance is not due to learnt biases from the data. As mentioned previously, the PDBbind core set is heavily biased, with both proteins (all) and ligands (~30%) represented in the training set. Therefore, models will tend to shortcut learning by using easily learnable biases which might be not present in other datasets. This is what is called a noncausal bias, where there is correlation but no causation. As mentioned in Sieg *et al.* [47] models can artificially achieve good predictions

592 by learning patterns that are not related to meaningful physico-chemical mechanisms for
593 binding. For example, it appears that most of the reported binding affinity prediction
594 models only memorize ligand and protein information instead of learning from their in-
595 teractions [7]. This appears to be a major issue in the field, as it leads to poor generaliza-
596 tion power.

597 A number of strategies have been suggested to compel neural networks to learn
598 from interactions for virtual screening purpose [48,49]. For instance, decoy poses have
599 been generated by modifying the position of ligands. These decoys poses were obtained
600 by redocking active compounds and selecting a low energy poses with a high RMSD
601 from the initial position. Even simpler methods like rotating and translating the ligands
602 have been applied. In a similar way, we propose that this could be applied on the
603 PDBbind dataset, by either redocking, rotating or translating high affinity ligands. The
604 resulting decoy poses would be labeled with low affinity. Consequently, when trained on
605 such datasets, models will encounter several occurrences of the same complexes, with
606 different ligand positions and different binding affinities. Therefore, we anticipate that
607 these models could adapt from primarily performing QSAR to potentially gaining a
608 deeper comprehension of protein-ligand interactions. Previous works were published on
609 the topic of data augmentation with docking for scoring functions [26,33,50,51]. To the
610 best of our knowledge, all of them focused on selecting poses similar to the crystallo-
611 graphic one, and assigning similar binding affinities. Another idea would be to dock
612 ligands with low affinity from the ChEMBL, especially the ones that are structurally
613 similar to high affinity ligands from the PDBbind. In the case that these ligands interact
614 with the same proteins, we would add the notion of activity cliff to the models. These
615 data augmentation methods would help the models generalize by making it focus on the
616 interactions rather than memorizing the bias inside the dataset. However, it is essential to
617 exercise caution when combining experimental and synthetic data. We have not used the
618 aforementioned methods in this study and we will discuss this in more detail in future
619 work.

620 As mentioned previously, there are several visualization tools that reveal which
621 parts of a structure are important when carrying out a prediction. In Hochuli *et al.* [41]
622 those methods were applied on GNINA CNN v2017 [49] in order to understand its un-
623 derlying reasoning for the classification of active and inactive molecules. Another way to
624 uncover if a model truly learnt from the protein-ligand interactions, is to train other
625 models by removing either the protein or the ligand. Subsequently, the models trained on
626 partial data are evaluated on the test set with the same partners removed. This evaluation
627 helps us understand the performance difference between learning and predicting with
628 the entire complex compared to learning and predicting with only the ligand or protein.
629 To facilitate this comparison, we calculate the prediction gap between learning on the full
630 complexes and learning on one of the 2 partners. The bigger the gap in prediction is, the
631 better the model's understanding of the interactions. However, these considerations are
632 relatively recent. Only a few neural networks have been evaluated for their ability to
633 learn from interactions, and not only memorize structural patterns in proteins or ligands.
634 For this purpose, at the Figure 6, we have evaluated the ability of learning on interactions
635 for two already published neural networks: a convolutional neural network (Pafnucy)
636 and graph convolutional neural network (GraphBAR).



637
638
639
640
641

Figure 6. Comparison of the performance of Pafnucy and GraphBAR without either the protein or the ligand. The performance of models was evaluated on the core set 2016. Learning on the whole complexes lead to significantly better performance. (a) The mean prediction gap between training on whole complexes or training on ligands alone is at 0.12 of coefficient correlation for Pafnucy; (b) while it is only at 0.03 for GraphBAR by training only on the ligands.

642
643
644
645
646
647
648
649

With both neural networks, training on the whole complexes give significantly better performance than training on the ligand or protein structures alone. Nonetheless, we can see disparities between the two neural networks as the difference in correlation coefficient by training only on the ligands compared to the whole complexes is 0.12 for Pafnucy, while it is only at 0.03 for GraphBAR. This means that Pafnucy does a better job at analyzing the interactions made between the proteins and the ligands, while GraphBAR seems to more heavily rely on learning patterns from ligands and then correlate them to binding affinities.

650
651
652
653
654

In the publication of OctSurf the performance was also evaluated by training only on ligands and only on proteins. A correlation coefficient of 0.79 was reported for the full complex, while reaching 0.73 with ligands, and 0.65 with proteins. Thus, the prediction gap is at 0.06, which is between Pafnucy and GraphBAR.

655
656
657
658
659

Other binding affinity models have been tested for their ability to learn from the interactions, by training only on proteins or only on ligands. All these results have been summarized in Table 2. The results of the Modular MPNN [7] are in accordance with previously evaluated neural networks. Nonetheless Deep Fusion [13] and PointTransformer [30] achieve a bigger prediction gap by removing either the ligand or the protein. It goes up to 0.41 for PointTransformer when learning only on ligands.

660
661

Table 2. Comparison of performance of several neural networks, on the PDBbind core set 2016 (test set of 285 complexes) with/without protein/ligand.

Neural networks	Whole complex (R, RMSE)	Only ligand (R, RMSE)	Only protein (R, RMSE)
Pafnucy ¹	0.77, 1.41	0.65, 1.67	0.66, 1.64
GraphBAR ²	0.76, 1.43	0.73, 1.51	0.59, 1.77
OctSurf ¹	0.79, 1.45	0.73 (n.a.)	0.64 (n.a.)
Modular MPNN ² [7]	0.81, 1.51	0.75, 1.57	0.73, 1.57
Deep Fusion ³ [13]	0.81, 1.31	0.49, 3.01	0.5, 4.00
PointTransformer ⁴ [30]	0.86, 1.19	0.45 (n.a.)	0.2 (n.a.)

Type of neural networks: ¹ 3D convolutional neural network (3D-CNN); ² Graph neural network (GNN); ³ 3D-CNN + spatial graph CNN (SG-CNN); ⁴ Transformer, n.a. not available

From these results, it seems that the ability of neural networks to learn from the interactions can vary importantly. The PDBbind 2019 was used as training data for both Pafnucy and GraphBAR, and both used similar descriptions of atoms. Therefore, the main factor differencing the two is the underlying structure of the networks, and the ensuing way of handling the data and carrying out prediction.

Accordingly, Deep Fusion reuse the same preparation protocol as Pafnucy, in terms of atomic description for example. Furthermore, it combines a 3D-CNN and a spatial graph CNN; this unique approach might be the reason for the model ability to better understand the protein-ligand interactions.

PointTransformer is a point cloud based neural network, like OctSurf. Therefore, we expected this tool to have similar prediction gap to OctSurf. On the contrary, the prediction gap was much more important with PointTransformer.

3.6. Other test sets

As shown throughout this paper, there are numerous biases contained in the core sets from the PDBbind. Due to this, we think it is important to use other types of benchmark datasets to accurately validate the new models developed. Indeed, the evaluation of models across several test sets grants a higher confidence when comparing performance. Across time, several other tests set have been developed to evaluate the scoring and ranking power of models. The scoring and ranking power are, respectively, the model ability to predict accurately the binding affinity and its ability to correctly rank ligands by using the predicted binding affinity.

There are test datasets that have already been used in numerous of publications [2]. For example, the Astex diverse set [52] was used to validate Pafnucy [10], DeepAtom [31] and RosENet [53]. It includes 85 protein-ligand complexes, 74 of which have known binding affinity. There are, as well, other test sets called the CSAR-NCS HiQ set 1 and set 2 [54] which are composed of 176 and 167 complexes from the Binding MOAD [35] and the PDBbind. After removing the complexes overlapping with usual training set, around 50 and 40 complexes remain for both test sets (Table A6). They have been used to evaluate Kdeep [55], RosENet [53], OnionNet-2 [56], graphDelta [57], GraphBAR [26], PIGNet [46], BAPA [58], CAPLA [59] and GIGN [32].

The FEP dataset [37] originally used in free energy perturbation studies has also been applied to evaluate the binding affinity predictions of several models [53,55,57]. It is used to test the ability of a model to discriminate between several similar ligands with different binding affinities for the same protein. It is composed of 8 proteins: BACE, CDK2, JNK1, MCL1, p38, PTP1B, Thrombin and Tyk2. Each protein family is represented by one structure. There are 200 ligands obtained from a small number of scaffolds. Their 3D positions in the binding site are provided. Their affinities have been obtained experimentally. This information is summarized in the Table A7.

Hold-out test sets have also been developed to evaluate performance of models on recent data. These test sets are obtained by performing a temporal split over a dataset, *i.e.* training models on complexes released before a specific date, and testing them on complexes released afterward. The hold-out test sets are generally big sized, with complexes that were not cherry-picked and thus are less likely to be biased.

- An example of such dataset can be found in Volkov *et al.* [7], where a modular MPNN and Pafnucy were trained on PDBbind 2016 and were evaluated by predicting on a 2019 hold-out set. To create this test set, they selected 3,386 complexes from the PDBbind 2019 that are not in the PDBbind 2016. Instead of using the files provided by the PDBbind, they downloaded the structures from the Protein Data Bank [18]. The complexes were curated and processed with Protoss v.4.0 [60] and IChem [61], *e.g.* protonation was optimized. Subsequently Isert *et al.* [62] reused

714 these data to train models with electron density-based geometric neural networks,
715 and they validated their binding affinity predictions on the same 2019 hold-out set.

- 716 • Another 2019 hold-out set of 4,366 complexes was used to evaluate GIGN [32]. They
717 compared their results against a dozen of neural networks, including OnionNet [63],
718 Pafnucy and GNN-DTI [64]. It is worth mentioning that the protein overlap rate
719 between test and training sets is of 69% instead of 100% for the core set 2016. As for
720 the ligand overlap rate, it goes down to 25% when it was at 38% for the core set 2016.
- 721 • Due to similar consideration, Deep Fusion [13] was evaluated on a test set of 222
722 complexes that was developed from the 2019 hold-out set, by removing complexes
723 with ligand or protein already present in the PDBbind 2016. Deep Fusion, Kdeep
724 and Pafnucy were trained on the PDBbind 2016, and evaluated on this test set.
- 725 • AK-score [65] was trained on the refined set of the PDBbind 2016 and it was evalu-
726 ated by predicting the binding affinity of 534 complexes newly released in the re-
727 fined set of the PDBbind 2018. For comparison purposes, they also evaluated the
728 performance of other scoring functions, namely X-score [66] and ChemPLP [67].
- 729 • The atomic convolutional neural network (ACNN) [68] was trained and tested on
730 several different splits of the PDBbind dataset. On top of a temporal split, they used
731 a stratified split based on the pKi value of complexes and a ligand scaffold split. The
732 stratified split allowed to select complexes covering all binding affinities in the train
733 and test sets. In the case of the scaffold split, ligands with unusual scaffold were
734 placed in the test set, therefore preventing the effect of QSAR in the prediction.
- 735 • In a similar way, MoleculeNet [69] has been trained and tested on PDBbind dataset
736 with a temporal split. As for PotentialNet [70], they performed cross-validation by
737 performing a pairwise structural homology split and a sequence similarity split.
738 Both splits are explained in detail in Li & Yang [71]. They were carried out via an
739 agglomerative hierarchical clustering, on the PDBbind 2007 refined set, resulting in a
740 test set of 118 and 101 samples, respectively.

741 The PDE10A dataset [72] have been recently released, with 1,162 docked or
742 co-crystallized PDE10A inhibitors. These data are sourced from a former project of Roche,
743 thus the binding affinity (IC₅₀) were obtained in a consistent way. There are 77 PDE10A
744 complex structures obtained by crystallography, and the rest of the complexes were
745 generated through multi-template docking. The test sets were obtained by using tem-
746 poral and binding mode splits. There are three temporal split test sets, the 2011, 2012 and
747 2013 test sets, with 250, 141 and 73 complexes respectively. Similarly there are three
748 binding mode split test sets, the aminohetaryl_c1_amide, c1_hetaryl_alkyl_c2_hetaryl
749 and the aryl_c1_amide_c2_hetaryl test sets, composed of 452, 291 and 419 complexes re-
750 spectively. They compared their 2D3D ML methods against PotentialNet [70] and ACNN
751 [68]. Isert *et al.* [62] also benchmarked their neural networks on these test sets.

752 Apart from the scoring and the ranking power, there are other criteria that can be
753 used to evaluate drug-target interactions models, like the virtual screening (VS) power.
754 This criterion defines the ability of a model to discriminate between decoys and active
755 molecules. As brought up in PIGNet [46], in order to accurately assess the performance of
756 a model, it is advised to evaluate not only its scoring power but also its virtual screening
757 power. For evaluating such ability, datasets incorporating decoys have also been used as
758 test set. Nonetheless, warnings must be raised about using these datasets. Indeed most of
759 them are also biased [47], especially when splitting one of them in training and test sets,
760 which usually leads the underlying models achieving artificially high performance. On
761 the contrary when training a scoring function on the PDBbind and predicting on VS da-
762 taset the results are usually lower. The performance of models evaluated on VS datasets
763 are measured by calculating the area under the ROC curve (AUC), which increases when
764 active molecules are predicted with higher binding affinities than decoys. Furthermore, it
765 is possible to evaluate scoring functions by calculating the enrichment factor (EF) from

766 the ROC curve. The EF is obtained by measuring the true positive rate (TPR) for a given
767 false positive rate (FPR). Therefore, it is possible to evaluate the model ability to find ac-
768 tive molecules over decoys for its best scored docking poses. Hence, The EF is more rep-
769 resentative of the use of VS tools in real condition, as users are mostly interested by the
770 ligands with the highest score.

771 Examples of such datasets are the DUD [73] (directory of useful decoys) and DUD-E
772 (enhanced DUD) [74]. They are used for benchmarking molecular docking by providing
773 active molecules and decoys (assumed inactive) for given targets. They have been de-
774 veloped to deal with usual dataset problems, like “artificial enrichment” which corre-
775 spond to having decoys that are very different from active molecules, and “false negative
776 bias” referring to decoy turning out to be active after being tested experimentally. The
777 DUD-E is an enhanced version of the DUD with increased amount of data. It is designed
778 to address the “analogue bias” of having highly similar active molecules. The DUD and
779 DUD-E are composed respectively of 2,950 and 22,886 active molecules, as well as 95,326
780 and 1,411,214 decoys (up to 50 decoys per active molecule charge states), for 40 and 102
781 targets. Unfortunately, there are still biases present in the DUD-E [75]. Especially, an an-
782 analogue bias intra and inter target was detected. These biases add up with the decoy bias,
783 which is the similarity of decoy from the same target. When trained on a part of the
784 DUD-E and evaluated on the other part, models obtain the same high performance (AUC
785 > 0.9) if we keep the whole complexes or only use the structure of the ligand. Therefore, it
786 leads to similar issues as the ones related to the PDBbind core set.

- 787 • The DUD-E was used to train AtomNet [76] and to evaluate its virtual screening
788 power. AtomNet is the first CNN applied on 3D grids to predict protein-ligand
789 binding affinities. 30 targets from DUD-E were used as test set, while the remaining
790 72 targets were used as the training set. On top of using DUD-E dataset, a derived
791 dataset, called “ChEMBL-20 PMD”, has been compiled to further benchmark
792 AtomNet. It was created based on several quality criteria and it is composed of
793 78,904 actives, 2,367,120 property-matched decoys (PMD), and 290 targets. That da-
794 taset is composed of decoy structurally different from the active molecules to pre-
795 vent the false negative bias issue which on the other hand results in an artificial en-
796 richment issue. Therefore, another dataset, called “ChEMBL-20 inactives”, was de-
797 veloped in order to evaluate AtomNet’s ability to classify experimentally-verified
798 active and inactive molecules. ChEMBL-20 inactives was obtained by replacing the
799 PMD by 363,187 molecules known to be inactive.
- 800 • In Lim *et al.* [64], they used the DUD-E and the PDBbind in order to constitute a
801 training set and a test set. Molecules were docked with Smina [77], resulting in a
802 dataset of docked poses for DUD-E’s 21,705 active molecules and 1,337,409 decoys.
803 As for PDBbind, the molecules were re-docked with Smina. If the pose had a RMSD
804 < 2 Å from the crystallographic pose, then it was classified it as a positive sample
805 and if the pose was at > 4 Å from crystallographic pose, then it was classified it as a
806 negative sample. Therefore, 2,094 positives and 12,246 negatives samples were ob-
807 tained. The training set was subsequently created with the docked poses of 72 pro-
808 teins from the DUDE and 70% of PDBbind redocked dataset. The test set consisted of
809 the docked poses from the remaining 25 proteins from the DUDE and 30% of
810 PDBbind redocked dataset. PDBbind split of data was based on a split of the targets,
811 so no proteins would be in the training and test sets. Thereafter another test set was
812 developed by selecting, from the ChEMBL, molecules with known binding affinity
813 for the 25 proteins from the DUDE test set. The affinity threshold was put to an IC50
814 of 1.0 µM, splitting the test set in 27,389 active and 26,939 inactive molecules.

815 Similarly to the DUD/DUD-E, the DEKOIS 2.0 [78] dataset was developed to evalu-
816 ate scoring functions for their virtual screening power. It is composed of 81 benchmark
817 sets for 80 protein targets (one target having 2 different binding sites and benchmark
818 sets). There are 40 active molecules per benchmark set. For each active molecule, 30

819 structurally diverse decoys were selected, resulting into 1,200 decoys per benchmark set.
820 The DEKOIS dataset is constituted of decoys that have not been tested experimentally,
821 therefore decoys were selected by matching the properties of the active molecules in order
822 to avoid artificial enrichment. Furthermore, the selection of the decoy has been tailored
823 to prevent the occurrence of latent actives in the decoy set (LADS). LADS are molecules
824 supposed to be decoy, which actually have an activity for the target. This issue was
825 previously referred to in the study as false negative bias. Only 4 targets out of the 81 of
826 the DEKOIS dataset are in common with the DUD-E [79], but 26 targets have at least 95%
827 of sequence identity with DUD-E targets [80]. As pointed out in Ballester paper [81],
828 several machine learning scoring functions [79,80,82] were trained on DUD-E and evaluated
829 on DEKOIS.

830 The Maximum unbiased validation (MUV) is another dataset developed to benchmark
831 virtual screening tools. It is composed of active and inactive molecules experimentally
832 tested for 17 target proteins. For each target proteins, there are 30 actives and
833 15,000 decoys with known binding affinities. In a similar fashion, Riniker and Landrum
834 [83] created a dataset from ChEMBL comprising 50 targets, with 100 diverse active
835 molecules per target and 2 decoys per active molecule leading to 10,000 decoys. The
836 GNN-DTI from Lim *et al.* [64] was evaluated on the MUV dataset. GNINA CNN v2017
837 [49] and the DenseNet CNN from Imrie *et al.* [84] were evaluated on a part of both the
838 MUV dataset and the ChEMBL dataset from Riniker and Landrum. The active molecules
839 and decoys were docked with smina [77] or AutoDock [85]. For the MUV dataset, out of
840 the 17 target proteins, 9 were used in the test set. Therefore, leading to 1,913 poses associated
841 with the 270 actives molecules and 1,177,989 poses associated with the 135,000
842 decoys. As for the ChEMBL dataset, 13 targets among the 50 targets were used, leading
843 to 11,406 poses associated with 1,300 active compounds and 663,671 poses associated
844 with 10,000 decoys.

845 In the CASF update [22], the scoring power, the ranking power, the docking power
846 and the screening power of several scoring functions were evaluated on the core set 2016.
847 The docking power correspond to the ability of a scoring functions, to identify the native
848 ligand binding pose among several decoy poses of the same ligand. More than 30 scoring
849 functions were evaluated for these criteria.

- 850 • To assess the docking power, decoy poses were generated by redocking PDBbind's
851 ligands in their binding site. For each complex, up to 100 decoy poses were selected,
852 by setting up 10 bins of 1 Å based on their RMSD values (0-10 Å) to the initial pose.
853 For each bin, ligand poses were clustered based on their conformation, and up to 10
854 poses were selected. This leads to a dataset composed of 22,492 decoy poses.
- 855 • In order to evaluate virtual screening power, the ligands were cross-docked. The
856 dataset is composed of 16,245 protein–ligand interaction pairs, by docking 285 lig-
857 ands into 57 proteins. The docking was performed on the protein structure with
858 highest affinity for each cluster. 100 poses were selected for each protein–ligand interaction
859 pair. Overall, 1,624,500 decoy poses make up this dataset.

860 In Francoeur *et al.* [33] several docking datasets have been compiled in order to train
861 and test their neural networks. They obtained a test set of 4,618 poses by redocking 280
862 complexes from the PDBbind core set 2016 and selecting up to 20 poses per complex. In a
863 similar fashion they redocked 3,805 complexes from the refined set and 11,324 from the
864 general set, leading to respectively 66,953 and 201,839 poses. Thereafter, they created the
865 CrossDocked2020 dataset, by crossdocking complexes from the Protein Data Bank [18]
866 that were selected based on the similarity of the binding pockets. They trained their
867 neural networks on a first version of this dataset, then selected wrongly predicted poses
868 as data augmentation for retraining the model. This iterative reinforcement learning
869 method led to a dataset of 22,584,102 poses (786,960 redocked poses and 21,797,142
870 cross-docked poses) from 18,450 complexes. 42% of these complexes have known binding
871 affinities from the PDBbind. From there, the BigBind dataset [86] was created, by map-

ping ChEMBL activities to the 3D structures of protein pockets in CrossDocked. By doing so, the number of pockets was reduced from 2,922 (in CrossDocked2020) to 1,067. The resulting dataset contains 11,430 3D structures, with 851,359 activities spanning 531,560 unique compounds.

In the GNINA CNN v2017 publication [49], the docking power was evaluated by redocking the 2013 PDBbind core (195 complexes). They obtained 98 low RMSD poses (<2 Å from the crystallographic pose) among a total of 897 poses. The training was carried on redocked complexes from the CSAR-NRC HiQ data set [54] and the CSAR HiQ Update. From the initial 466 complexes, they redocked 377 complexes having a binding affinity > 5 pK. Poses at less than 2 Å from crystallographic poses, were labeled as positive, while the one at more than 4 Å were labelled as negative. The one between 2 and 4 Å were discarded. This leads to a dataset composed of 745 positive poses (from 327 complexes) and 3,251 negative poses (from 300 complexes).

Famous datasets like the PDBbind/CASF, the DUD-E or the MUV have been applied to train and evaluate many models. Unfortunately, it appears that most of the famous datasets are biased. Although they may still be relevant to some extent for comparison purpose, we have seen the development of a myriad of new benchmark datasets. Many papers presenting new neural networks, demonstrated their performance on custom test sets. For example, six papers developed their own training and test sets by performing a temporal split. For a better comparison of models, it would be preferable to evaluate their performance on a common benchmark dataset obtained through temporal split.

Overall, we think that it is important to evaluate the scoring power of models on several benchmark datasets, to get an accurate evaluation of their performance. On top of that, we advise for the evaluation of their ranking, docking and screening powers. By doing so, we can get a better idea of their usefulness in real case scenario.

4. Conclusion

For some years now, deep learning models have been developed to predict protein-ligand binding affinity using structural data. The scientific community has been trying to establish guidance on how to use these tools. Data plays a central role in training DL models. Therefore, we have been investigating how the data can impact the performance of models, as well as the intrinsic biases from the PDBbind. Among all the problems related to the data, the question of quality and the quantity of the data used to train DL algorithms seems crucial. For instance, another study has delved into the influence of the quantity and quality of non-structural data on predicting binding affinities using deep learning [87]. Additionally, in structure-based affinity prediction, a lot of neural networks have been trained only on the PDBbind's refined set, instead of the totality of the data available. The refined set is made of complexes selected based on quality criteria. The reasoning for training on only the refined set is to avoid the "garbage in, garbage out" issue. We have evaluated this factor by training Pafnucy, a well-known CNN for the prediction of protein-ligand binding affinity, on the refined set only and on the entire dataset. We found out that the performance was lower by training on the refined set. Therefore, we think that it is important to train on most of the data available, as long as the data has been accurately labelled.

The PDBbind database groups several types of ligands together, with peptides and small molecules being the main populations involved in protein-ligand complexes. As only a few neural networks [15] have focused on training on complexes involving a specific type of ligand, we trained Pafnucy on the protein-peptide and protein-nonpeptide complexes of the PDBbind. We compared the performance by training on similar sized datasets and found out that models trained with peptides were able to better predict the binding affinity of protein-peptide complexes. Therefore, it would be interesting to investigate transfer learning on such type of data, to reach good performance for the prediction of binding affinity of protein-peptide complexes.

924 Due to the computational expensive nature of CNN and their high requirement on
925 RAM, it is not possible to train models on the whole protein structure. Indeed, before-
926 hand it is required to create pockets around the ligands. We have evaluated performance
927 of models trained on pockets made of the amino acids detected at 6, 8, 10, 12 and 14 Å
928 from the ligand. By increasing the size of pockets, we see performance increase until 10
929 Å, thereafter performance stagnate. This performance trend, increasing with pocket size
930 until reaching a certain value, aligns with OnioNet 2 [56], which showed performance
931 improvement up to 15 Å. As most protein-ligand interactions should be already consid-
932 ered at a distance of 6 Å from the ligand, we propose that the increase in performance is
933 due to the bias in the data. In other word, adding more information about the proteins,
934 would not add any useful physical information but just help the models to overfit. An-
935 other possible explanation would be related to the existence of some long distance in-
936 fluences of these amino acids on the ligand, which would impact the affinity of the com-
937 plexes. Therefore, the AI would detect these indirect interactions that would be hard to
938 notice for a human.

939 Following on the topic of biases in the PDBbind core set, we evaluated different
940 types of neural networks for their ability to learn from the interactions instead of memo-
941 rizing the biases in the data. From these results, it seems that GraphBar does mostly
942 QSAR since it has nearly the same performance with and without the proteins, or in other
943 words Pafnucy seems to better understand the interaction between the protein and the
944 ligand. On that topic, published work [13,30] reported even bigger performance gaps.

945 Finally, we pointed out some flaws inside PDBbind 2016 core set. For example, 30%
946 of the ligands from the test set are also in PDBbind general set. As for the proteins, this
947 value goes up to 100%. In the GNINA CNN v2017 publication [49], this was mitigated by
948 removing test targets with more than 80% sequence similarity with a target from the
949 training set. In a similar fashion, PIGNet [46] exclude, from the CSAR NRC-HiQ, the
950 complexes that have at least 60% of sequence similarity with a target from the training
951 set. Following these examples, Yang *et al.* [17] advocate for the removal, from test sets, of
952 complexes with structurally similar proteins and ligands in comparison to training sets.
953 Although doing as such prevents the evaluation of models in the situation of drug re-
954 purposing and hit to lead optimization [7]. Therefore, we recommend evaluating models
955 on several test sets to better assess their ability to generalize and to accurately predict the
956 binding affinity. On top of the CASF and the CSAR NRC-HIQ, we can list the Astex di-
957 verse set, the FEP dataset and the holdout test sets. Several neural networks have already
958 evaluated their performance on such datasets, allowing for easier comparison with the
959 newly developed methodologies.

960 For a thorough evaluation of the models, we also advise evaluating their screening
961 power. To measure that criterion, it is required to dock active molecules and decoys, be-
962 fore evaluating their binding affinities and ranking the molecules. Some datasets propose
963 list of decoys and active molecules, like the DUD-E [74], DEKOIS [78], MUV [88] or the
964 “Riniker and Landrum ChEMBL” [83]. The difference between these datasets depends
965 mostly on the way they define decoys, and how they tried to prevent the appearance of
966 biases. Unfortunately, biases can still be found in these datasets. In the end, models
967 trained on the PDBbind did not outperform docking software in term of VS power when
968 applied on the DUD-E [75]. Nonetheless, if it is possible to obtain better VS power, even
969 at the expense of lowering scoring power performance on PDBbind core set, this would
970 mean we are likely going in the right direction. This should be achievable by training
971 models on a decoy poses augmented PDBbind dataset, which should force models to
972 learn from the interactions instead of memorizing ligand and protein structures. How-
973 ever, by using decoy poses, we might not represent accurately the physico-chemical re-
974 ality of the interactions of a protein and a ligand. Indeed, the interactions between them
975 are dynamic, thus the ligand might take several positions inside the binding site across
976 time. As mentioned previously in the literature [89], it would be more suitable to perform
977 data augmentation with molecular dynamics simulations. For example, snapshots could

978 be extracted from the simulations and fed to neural networks. This way, we can expect to
979 improve models understanding of protein-ligand interactions.

980 **Supplementary Materials: Table S1:** The dataset of peptides curated from the PDBbind v.2019 is
981 provided as a CSV file.

982 **Author Contributions:** Investigation and writing—original draft preparation, P.Y.L.; supervision,
983 S.A.S., P.B., and G.T.; writing—review and editing, S.A.S., P.B., J.C.G.T. and G.T.; All authors have
984 read and agreed to the published version of the manuscript.

985 **Funding:** This research was funded by JANSSEN.

986 **Institutional Review Board Statement:** Not applicable.

987 **Informed Consent Statement:** Not applicable.

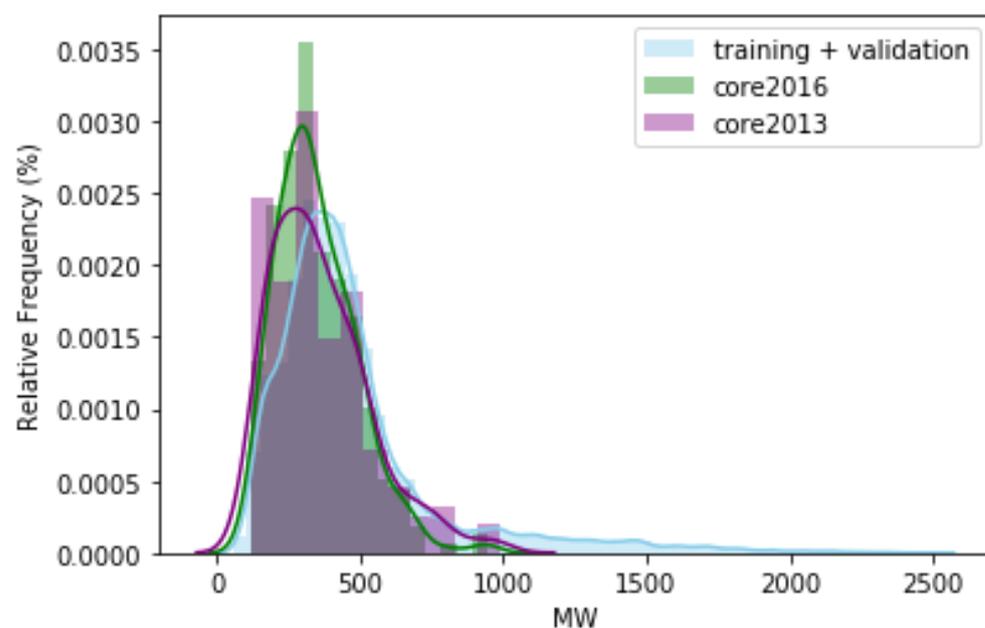
988 **Data Availability Statement:** The data presented in this study are openly available in PDBbind at
989 10.1021/jm048957q

990 **Acknowledgments:** Authors gratefully acknowledge major financial support from Janssen which
991 made this study possible. P.Y.L, S.A.S and P.B. are thankful to the projects CHemBio (FEDER-FSE
992 2014-2020-EX003677), Techsab (FEDER-FSE 2014-2020-EX011313), the RTR Motivhealth
993 (2019-00131403) and the Labex programs SYNORG (ANR-11-LABX-0029) and IRON
994 (ANR-11-LABX-0018-01) for their financial support of ICOA, UMR 7311, University of Orléans,
995 CNRS

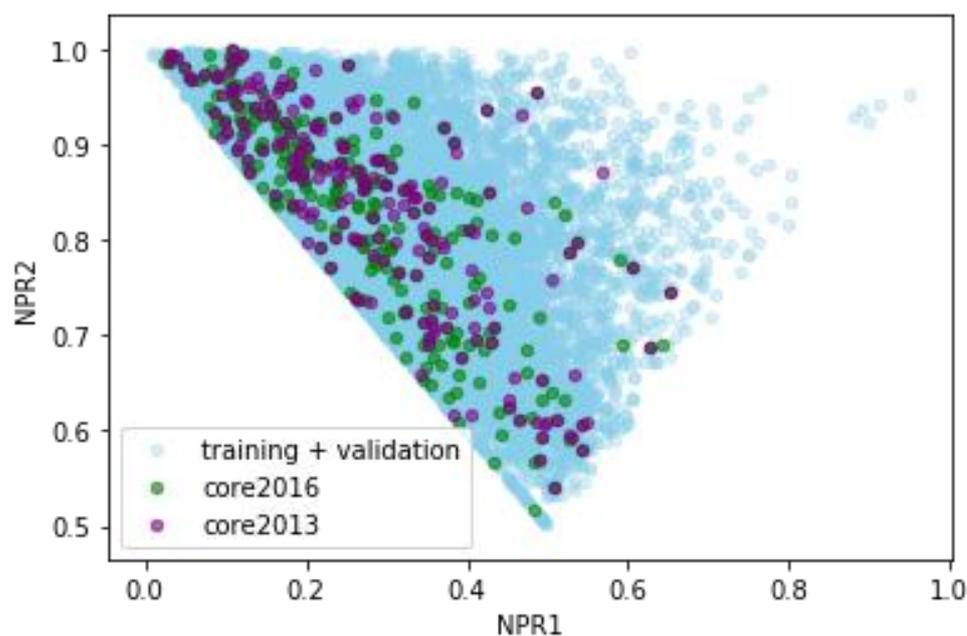
996 **Conflicts of Interest:** The authors declare no conflict of interest.

997 **Sample Availability:** Not applicable.

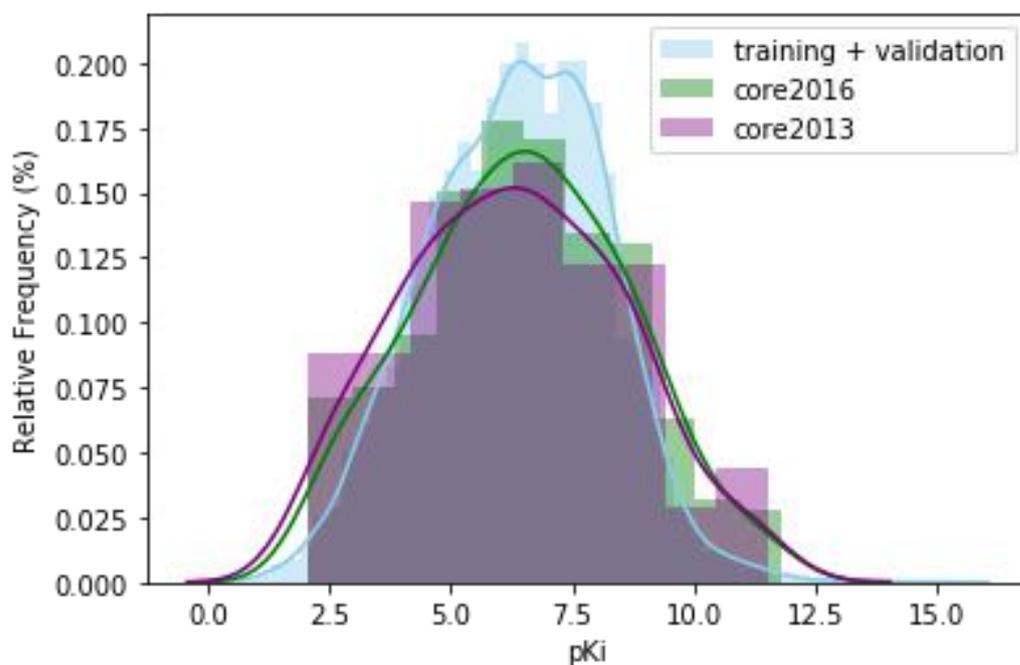
998 Appendix A



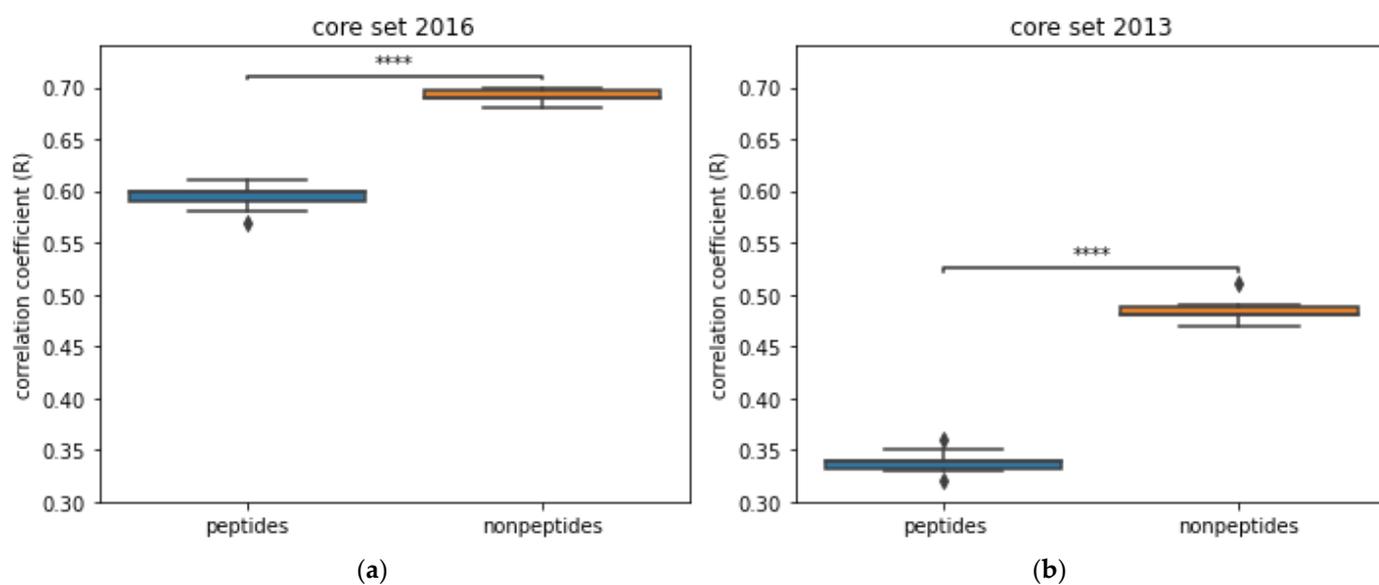
1000 **Figure A1.** Distribution of PDBbind's ligands in function of their molecular weight. The training
1001 and validation set are plotted together in blue, the test sets are colored in pink and green, corre-
1002 sponding to the core set 2013 and core set 2016 respectively.
1003
1004
1005



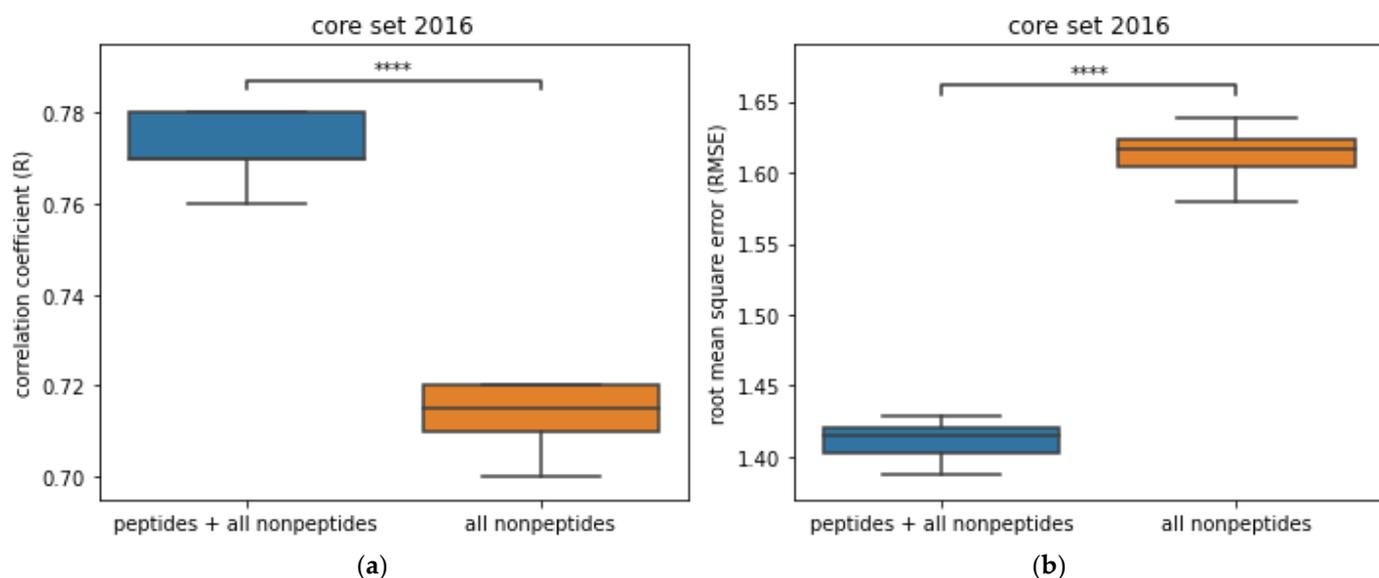
1006
1007
1008
1009
1010
1011
1012
1013
Figure A2. Distribution of PDBbind's ligands in function of their shape. The normalized PMI ratio (NPR) are calculated from the principal moment of inertia (PMI) of the ligands. The ligands located at the top right of the plot are spherical, while the one at the top left are rod-like. Lastly, the ligands in the bottom of the plot are with the shape of a disc. The training and validation set are plotted together in blue. While the test sets are in pink and green, corresponding to the core set 2013 and core set 2016 respectively.



1014
1015
1016
1017
1018
1019
1020
Figure A3. Distribution of PDBbind's complexes in function of their affinity. The training and validation set are plotted together in blue. While the test sets are in pink and green, corresponding to the core set 2013 and core set 2016 respectively.



1021 **Figure A4.** Comparison of the performance of models trained with peptide-protein complexes and
 1022 with nonpeptide-protein complexes. Models were trained with Pafnucy on 2,383 complexes and
 1023 validated on 492 complexes. (a) Performance is evaluated on the core set 2016; (b) performance are
 1024 evaluated on the core set 2013.



1026 **Figure A5.** Comparison of the performance of models trained on the whole PDBbind and with only
 1027 nonpeptide-protein complexes (trained on 13,403 complexes and validated on 1000 complexes).
 1028 Models were trained with Pafnucy. Performance is evaluated on the core set 2016 (285 complexes).
 1029 (a) The performance is evaluated with the coefficient correlation; (b) The performance are evalu-
 1030 ated with the root mean square error (RMSE).

1031 **Table A6.** Number of complexes of the CSAR NRC-HiQ set 1 & 2, used in each publication. In
 1032 GIGN, the sets were merged together.

Neural networks	CSAR NRC-HiQ set1	CSAR NRC-HiQ set2
Kdeep [55]	55	49

RosENet [53]	33	10
OnionNet-2 [56]	55	49
graphDelta [57]	53	49
GraphBAR [26]	51	36
PIGNet [46]	48 & 37	37 & 22
BAPA [58]	50	44
CAPLA [59]	51	36
GIGN [32]	47	

Table A7. Summary of the FEP dataset from Kdeep [55] and Wang *et al.* [37]. This table displays the target (protein family), the reference PDB id used, the number of ligands positioned in 3D in each structure and the experimental affinity range of complexes belonging to the same protein family.

Target	PDB ID	Number of ligands	Affinity range (kcal/mol)
MCL1	4HW3	42	4.2
BACE	4DJW	36	3.5
p38	3FLY	34	3.8
PTP1B	2QBS	23	5.1
JNK1	2GMX	21	3.4
CDK2	1H1Q	16	4.2
Tyk2	4GIH	16	4.3
Thrombin	2ZFF	11	1.7

References

- Baig, M.H.; Ahmad, K.; Roy, S.; Ashraf, J.M.; Adil, M.; Siddiqui, M.H.; Khan, S.; Kamal, M.A.; Provazník, I.; Choi, I. Computer Aided Drug Design: Success and Limitations. *Current pharmaceutical design* **2016**, *22*, 572-581, doi:10.2174/1381612822666151125000550.
- Meli, R.; Morris, G.; Biggin, P. Scoring functions for protein-ligand binding affinity prediction using structure-based deep learning: a review. *Frontiers in Bioinformatics* **2022**, *2*, doi:10.3389/fbinf.2022.885983.
- Shen, C.; Zhang, X.; Hsieh, C.-Y.; Deng, Y.; Wang, D.; Xu, L.; Wu, J.; Li, D.; Kang, Y.; Hou, T.; et al. A generalized protein–ligand scoring framework with balanced scoring, docking, ranking and screening powers. *Chemical Science* **2023**, *14*, 8129-8146, doi:10.1039/D3SC02044D.
- Hou, T.; Wang, J.; Li, Y.; Wang, W. Assessing the performance of the MM/PBSA and MM/GBSA methods. 1. The accuracy of binding free energy calculations based on molecular dynamics simulations. *Journal of chemical information and modeling* **2011**, *51*, 69-82, doi:10.1021/ci100275a.
- Jukič, M.; Janežič, D.; Bren, U. Potential Novel Thioether-Amide or Guanidine-Linker Class of SARS-CoV-2 Virus RNA-Dependent RNA Polymerase Inhibitors Identified by High-Throughput Virtual Screening Coupled to Free-Energy Calculations. *International Journal of Molecular Sciences* **2021**, *22*, 11143.
- Gapsys, V.; Pérez-Benito, L.; Aldeghi, M.; Seeliger, D.; van Vlijmen, H.; Tresadern, G.; de Groot, B.L. Large scale relative protein ligand binding affinities using non-equilibrium alchemy. *Chemical Science* **2020**, *11*, 1140-1152, doi:10.1039/C9SC03754C.
- Volkov, M.; Turk, J.-A.; Drizard, N.; Martin, N.; Hoffmann, B.; Gaston-Mathé, Y.; Rognan, D. On the Frustration to Predict Binding Affinities from Protein–Ligand Structures with Deep Neural Networks. *Journal of medicinal chemistry* **2022**, doi:10.1021/acs.jmedchem.2c00487.

- 1059 8. Deng, J.; Dong, W.; Socher, R.; Li, L.J.; Kai, L.; Li, F.-F. ImageNet: A large-scale hierarchical image database. In Proceedings
1060 of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, 20-25 June, 2009; pp. 248-255.
- 1061 9. Wang, R.; Fang, X.; Lu, Y.; Wang, S. The PDBbind database: collection of binding affinities for protein-ligand complexes
1062 with known three-dimensional structures. *Journal of medicinal chemistry* **2004**, *47*, 2977-2980, doi:10.1021/jm030580l.
- 1063 10. Stepniewska-Dziubinska, M.M.; Zielenkiewicz, P.; Siedlecki, P. Development and evaluation of a deep learning model for
1064 protein-ligand binding affinity prediction. *Bioinformatics (Oxford, England)* **2018**, *34*, 3666-3674,
1065 doi:10.1093/bioinformatics/bty374.
- 1066 11. Braka, A.; Garnier, N.; Bonnet, P.; Aci-Sèche, S. Residence Time Prediction of Type 1 and 2 Kinase Inhibitors from
1067 Unbinding Simulations. *Journal of chemical information and modeling* **2020**, *60*, 342-348, doi:10.1021/acs.jcim.9b00497.
- 1068 12. Ziada, S.; Diharce, J.; Raimbaud, E.; Aci-Sèche, S.; Ducrot, P.; Bonnet, P. Estimation of Drug-Target Residence Time by
1069 Targeted Molecular Dynamics Simulations. *Journal of chemical information and modeling* **2022**, *62*, 5536-5549,
1070 doi:10.1021/acs.jcim.2c00852.
- 1071 13. Jones, D.; Kim, H.; Zhang, X.; Zemla, A.; Stevenson, G.; Bennett, W.F.D.; Kirshner, D.; Wong, S.E.; Lightstone, F.C.; Allen,
1072 J.E. Improved Protein-Ligand Binding Affinity Prediction with Structure-Based Deep Fusion Inference. *Journal of chemical
1073 information and modeling* **2021**, *61*, 1583-1592, doi:10.1021/acs.jcim.0c01306.
- 1074 14. Unarta, I.C.; Xu, J.; Shang, Y.; Cheung, C.H.P.; Zhu, R.; Chen, X.; Cao, S.; Cheung, P.P.; Bierer, D.; Zhang, M.; et al. Entropy
1075 of stapled peptide inhibitors in free state is the major contributor to the improvement of binding affinity with the GK
1076 domain. *RSC chemical biology* **2021**, *2*, 1274-1284, doi:10.1039/d1cb00087j.
- 1077 15. Ahmed, A.; Mam, B.; Sowdhamini, R. DEELIG: A Deep Learning Approach to Predict Protein-Ligand Binding Affinity.
1078 *Bioinformatics and biology insights* **2021**, *15*, doi:10.1177/11779322211030364.
- 1079 16. Jukič, M.; Bren, U. Machine Learning in Antibacterial Drug Design. *Frontiers in Pharmacology* **2022**, *13*,
1080 doi:10.3389/fphar.2022.864412.
- 1081 17. Yang, J.; Shen, C.; Huang, N. Predicting or Pretending: Artificial Intelligence for Protein-Ligand Interactions Lack of
1082 Sufficiently Large and Unbiased Datasets. *Frontiers in Pharmacology* **2020**, *11*, doi:10.3389/fphar.2020.00069.
- 1083 18. Berman, H.M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T.N.; Weissig, H.; Shindyalov, I.N.; Bourne, P.E. The Protein
1084 Data Bank. *Nucleic acids research* **2000**, *28*, 235-242, doi:10.1093/nar/28.1.235.
- 1085 19. Li, Y.; Liu, Z.; Li, J.; Han, L.; Liu, J.; Zhao, Z.; Wang, R. Comparative Assessment of Scoring Functions on an Updated
1086 Benchmark: 1. Compilation of the Test Set. *Journal of chemical information and modeling* **2014**, *54*, 1700-1716,
1087 doi:10.1021/ci500080q.
- 1088 20. Li, Y.; Han, L.; Liu, Z.; Wang, R. Comparative Assessment of Scoring Functions on an Updated Benchmark: 2. Evaluation
1089 Methods and General Results. *Journal of chemical information and modeling* **2014**, *54*, 1717-1736, doi:10.1021/ci500081m.
- 1090 21. Li, Y.; Su, M.; Liu, Z.; Li, J.; Liu, J.; Han, L.; Wang, R. Assessing protein-ligand interaction scoring functions with the
1091 CASF-2013 benchmark. *Nature protocols* **2018**, *13*, 666-680, doi:10.1038/nprot.2017.114.
- 1092 22. Su, M.; Yang, Q.; Du, Y.; Feng, G.; Liu, Z.; Li, Y.; Wang, R. Comparative Assessment of Scoring Functions: The CASF-2016
1093 Update. *Journal of chemical information and modeling* **2019**, *59*, 895-913, doi:10.1021/acs.jcim.8b00545.
- 1094 23. Özçelik, R.; van Tilborg, D.; Jiménez-Luna, J.; Grisoni, F. Structure-Based Drug Discovery with Deep Learning.
1095 *ChemBioChem* **2023**, *24*, e202200776, doi:10.1002/cbic.202200776.
- 1096 24. Isert, C.; Atz, K.; Schneider, G. Structure-based drug design with geometric deep learning. 2022.
- 1097 25. Liu, Q.; Wang, P.-S.; Zhu, C.; Gaines, B.B.; Zhu, T.; Bi, J.; Song, M. OctSurf: Efficient hierarchical voxel-based molecular
1098 surface representation for protein-ligand affinity prediction. *Journal of Molecular Graphics and Modelling* **2021**, *105*, 107865,
1099 doi:10.1016/j.jmgm.2021.107865.

- 1100 26. Son, J.; Kim, D. Development of a graph convolutional neural network model for efficient prediction of protein-ligand
1101 binding affinities. *PLoS One* **2021**, *16*, e0249404, doi:10.1371/journal.pone.0249404.
- 1102 27. Guo, Y.; Wang, H.; Hu, Q.; Liu, H.; Liu, L.; Bennamoun, M. Deep learning for 3d point clouds: A survey. *IEEE transactions*
1103 *on pattern analysis and machine intelligence* **2020**, *43*, 4338-4364.
- 1104 28. Meagher, D. *Octree Encoding: A New Technique for the Representation, Manipulation and Display of Arbitrary 3-D Objects by*
1105 *Computer*; 1980.
- 1106 29. Li, S.; Zhou, J.; Xu, T.; Huang, L.; Wang, F.; Xiong, H.; Huang, W.; Dou, D.; Xiong, H. Structure-Aware Interactive Graph
1107 Neural Networks for the Prediction of Protein-Ligand Binding Affinity. In Proceedings of the 27th ACM SIGKDD
1108 Conference on Knowledge Discovery & Data Mining, August 14 - 18, 2021; pp. 975–985.
- 1109 30. Wang, Y.; Wu, S.; Duan, Y.; Huang, Y. A point cloud-based deep learning strategy for protein-ligand binding affinity
1110 prediction. *Briefings in bioinformatics* **2022**, *23*, doi:10.1093/bib/bbab474.
- 1111 31. Li, Y.; Rezaei, M.A.; Li, C.; Li, X. DeepAtom: A Framework for Protein-Ligand Binding Affinity Prediction. *IEEE*
1112 *International Conference on Bioinformatics and Biomedicine (BIBM)* **2019**, 303-310, doi:10.1109/BIBM47256.2019.8982964.
- 1113 32. Yang, Z.; Zhong, W.; Lv, Q.; Dong, T.; Yu-Chian Chen, C. Geometric Interaction Graph Neural Network for Predicting
1114 Protein-Ligand Binding Affinities from 3D Structures (GIGN). *The Journal of Physical Chemistry Letters* **2023**, *14*, 2020-2033,
1115 doi:10.1021/acs.jpcclett.2c03906.
- 1116 33. Francoeur, P.G.; Masuda, T.; Sunseri, J.; Jia, A.; Iovanisci, R.B.; Snyder, I.; Koes, D.R. Three-Dimensional Convolutional
1117 Neural Networks and a Cross-Docked Data Set for Structure-Based Drug Design. *Journal of chemical information and*
1118 *modeling* **2020**, *60*, 4200-4215, doi:10.1021/acs.jcim.0c00411.
- 1119 34. Wang, R.; Fang, X.; Lu, Y.; Yang, C.-Y.; Wang, S. The PDBbind Database: Methodologies and Updates. *Journal of medicinal*
1120 *chemistry* **2005**, *48*, 4111-4119, doi:10.1021/jm048957q.
- 1121 35. Hu, L.; Benson, M.L.; Smith, R.D.; Lerner, M.G.; Carlson, H.A. Binding MOAD (Mother Of All Databases). *Proteins:*
1122 *Structure, Function, and Bioinformatics* **2005**, *60*, 333-340, doi:10.1002/prot.20512.
- 1123 36. Xiong, G.; Shen, C.; Yang, Z.; Jiang, D.; Liu, S.; Lu, A.; Chen, X.; Hou, T.; Cao, D. Featurization strategies for protein-ligand
1124 interactions and their applications in scoring function development. *WIREs Computational Molecular Science* **2022**, *12*, e1567,
1125 doi:<https://doi.org/10.1002/wcms.1567>.
- 1126 37. Wang, L.; Wu, Y.; Deng, Y.; Kim, B.; Pierce, L.; Krilov, G.; Lupyan, D.; Robinson, S.; Dahlgren, M.K.; Greenwood, J.; et al.
1127 Accurate and Reliable Prediction of Relative Ligand Binding Potency in Prospective Drug Discovery by Way of a Modern
1128 Free-Energy Calculation Protocol and Force Field. *Journal of the American Chemical Society* **2015**, *137*, 2695-2703,
1129 doi:10.1021/ja512751q.
- 1130 38. Montavon, G.; Binder, A.; Lapuschkin, S.; Samek, W.; Müller, K.-R. Layer-Wise Relevance Propagation: An Overview. In
1131 Proceedings of the Explainable AI, 2019.
- 1132 39. Karpov, P.; Godin, G.; Tetko, I.V. Transformer-CNN: Swiss knife for QSAR modeling and interpretation. *Journal of*
1133 *Cheminformatics* **2020**, *12*, 17, doi:10.1186/s13321-020-00423-w.
- 1134 40. Nielsen, I.E.; Dera, D.; Rasool, G.; Ramachandran, R.P.; Bouaynaya, N.C. Robust Explainability: A tutorial on
1135 gradient-based attribution methods for deep neural networks. *IEEE Signal Processing Magazine* **2022**, *39*, 73-84,
1136 doi:10.1109/MSP.2022.3142719.
- 1137 41. Hochuli, J.; Helbling, A.; Skaist, T.; Ragoza, M.; Koes, D.R. Visualizing convolutional neural network protein-ligand
1138 scoring. *Journal of Molecular Graphics and Modelling* **2018**, *84*, 96-108, doi:10.1016/j.jmgn.2018.06.005.
- 1139 42. Liu, Z.; Li, Y.; Han, L.; Li, J.; Liu, J.; Zhao, Z.; Nie, W.; Liu, Y.; Wang, R. PDB-wide collection of binding data: current status
1140 of the PDBbind database. *Bioinformatics (Oxford, England)* **2014**, *31*, 405-412, doi:10.1093/bioinformatics/btu626.

- 1141 43. Bournez, C.; Carles, F.; Peyrat, G.; Aci-Sèche, S.; Bourg, S.; Meyer, C.; Bonnet, P. Comparative Assessment of Protein
1142 Kinase Inhibitors in Public Databases and in PKIDB. *Molecules* **2020**, *25*, 3226, doi:10.3390/molecules25143226.
- 1143 44. Jiménez-Luna, J.; Grisoni, F.; Schneider, G. Drug discovery with explainable artificial intelligence. *Nature Machine
1144 Intelligence* **2020**, *2*, 573-584, doi:10.1038/s42256-020-00236-4.
- 1145 45. Lakshminarayanan, B.; Pritzel, A.; Blundell, C. Simple and scalable predictive uncertainty estimation using deep
1146 ensembles. In Proceedings of the 31st International Conference on Neural Information Processing Systems, Long Beach,
1147 California, USA, 2017; pp. 6405–6416.
- 1148 46. Moon, S.; Zhung, W.; Yang, S.; Lim, J.; Kim, W.Y. PIGNet: a physics-informed deep learning model toward generalized
1149 drug–target interaction predictions. *Chemical Science* **2022**, *13*, 3661-3673, doi:10.1039/D1SC06946B.
- 1150 47. Sieg, J.; Flachsenberg, F.; Rarey, M. In Need of Bias Control: Evaluating Chemical Data for Machine Learning in
1151 Structure-Based Virtual Screening. *Journal of chemical information and modeling* **2019**, *59*, 947-961,
1152 doi:10.1021/acs.jcim.8b00712.
- 1153 48. Scantlebury, J.; Brown, N.; Von Delft, F.; Deane, C.M. Data Set Augmentation Allows Deep Learning-Based Virtual
1154 Screening to Better Generalize to Unseen Target Classes and Highlight Important Binding Interactions. *Journal of chemical
1155 information and modeling* **2020**, *60*, 3722-3730, doi:10.1021/acs.jcim.0c00263.
- 1156 49. Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D.R. Protein-Ligand Scoring with Convolutional Neural Networks.
1157 *Journal of chemical information and modeling* **2017**, *57*, 942-957, doi:10.1021/acs.jcim.6b00740.
- 1158 50. Li, H.; Leung, K.-S.; Wong, M.-H.; Ballester, P.J. Correcting the impact of docking pose generation error on binding affinity
1159 prediction. *BMC Bioinformatics* **2016**, *17*, 308, doi:10.1186/s12859-016-1169-4.
- 1160 51. Boyles, F.; Deane, C.M.; Morris, G.M. Learning from Docked Ligands: Ligand-Based Features Rescue Structure-Based
1161 Scoring Functions When Trained on Docked Poses. *Journal of chemical information and modeling* **2022**, *62*, 5329-5341,
1162 doi:10.1021/acs.jcim.1c00096.
- 1163 52. Hartshorn, M.J.; Verdonk, M.L.; Chessari, G.; Brewerton, S.C.; Mooij, W.T.M.; Mortenson, P.N.; Murray, C.W. Diverse,
1164 High-Quality Test Set for the Validation of Protein–Ligand Docking Performance. *Journal of medicinal chemistry* **2007**, *50*,
1165 726-741, doi:10.1021/jm061277y.
- 1166 53. Hassan-Harrirou, H.; Zhang, C.; Lemmin, T. RosENet: Improving Binding Affinity Prediction by Leveraging Molecular
1167 Mechanics Energies with an Ensemble of 3D Convolutional Neural Networks. *Journal of chemical information and modeling*
1168 **2020**, *60*, 2791-2802, doi:10.1021/acs.jcim.0c00075.
- 1169 54. Dunbar, J.B., Jr.; Smith, R.D.; Damm-Ganamet, K.L.; Ahmed, A.; Esposito, E.X.; Delproposto, J.; Chinnaswamy, K.; Kang,
1170 Y.-N.; Kubish, G.; Gestwicki, J.E.; et al. CSAR Data Set Release 2012: Ligands, Affinities, Complexes, and Docking Decoys.
1171 *Journal of chemical information and modeling* **2013**, *53*, 1842-1852, doi:10.1021/ci4000486.
- 1172 55. Jiménez, J.; Škalič, M.; Martínez-Rosell, G.; De Fabritiis, G. KDEEP: Protein–Ligand Absolute Binding Affinity Prediction
1173 via 3D-Convolutional Neural Networks. *Journal of chemical information and modeling* **2018**, *58*, 287-296,
1174 doi:10.1021/acs.jcim.7b00650.
- 1175 56. Wang, Z.; Zheng, L.; Liu, Y.; Qu, Y.; Li, Y.-Q.; Zhao, M.; Mu, Y.; Li, W. OnionNet-2: a convolutional neural network model
1176 for predicting protein-ligand binding affinity based on residue-atom contacting shells. *Frontiers in chemistry* **2021**, *9*, 913,
1177 doi:10.3389/fchem.2021.753002.
- 1178 57. Karlov, D.S.; Sosnin, S.; Fedorov, M.V.; Popov, P. graphDelta: MPNN Scoring Function for the Affinity Prediction of
1179 Protein–Ligand Complexes. *ACS Omega* **2020**, *5*, 5150-5159, doi:10.1021/acsomega.9b04162.
- 1180 58. Seo, S.; Choi, J.; Park, S.; Ahn, J. Binding affinity prediction for protein–ligand complex using deep attention mechanism
1181 based on intermolecular interactions. *BMC Bioinformatics* **2021**, *22*, 542, doi:10.1186/s12859-021-04466-0.

- 1182 59. Jin, Z.; Wu, T.; Chen, T.; Pan, D.; Wang, X.; Xie, J.; Quan, L.; Lyu, Q. CAPLA: improved prediction of protein–ligand
1183 binding affinity by a deep learning approach based on a cross-attention mechanism. *Bioinformatics (Oxford, England)* **2023**,
1184 *39*, doi:10.1093/bioinformatics/btad049.
- 1185 60. Bietz, S.; Urbaczek, S.; Schulz, B.; Rarey, M. Protoss: a holistic approach to predict tautomers and protonation states in
1186 protein-ligand complexes. *Journal of Cheminformatics* **2014**, *6*, 12, doi:10.1186/1758-2946-6-12.
- 1187 61. Da Silva, F.; Desaphy, J.; Rognan, D. IChem: A Versatile Toolkit for Detecting, Comparing, and Predicting Protein–Ligand
1188 Interactions. *ChemMedChem* **2018**, *13*, 507–510, doi:10.1002/cmdc.201700505.
- 1189 62. Isert, C.; Atz, K.; Riniker, S.; Schneider, G. Exploring protein-ligand binding affinity prediction with electron
1190 density-based geometric deep learning. *ChemRxiv preprint* **2023**, doi:10.26434/chemrxiv-2023-585vf.
- 1191 63. Zheng, L.; Fan, J.; Mu, Y. OnionNet: a Multiple-Layer Intermolecular-Contact-Based Convolutional Neural Network for
1192 Protein–Ligand Binding Affinity Prediction. *ACS Omega* **2019**, *4*, 15956–15965, doi:10.1021/acsomega.9b01997.
- 1193 64. Lim, J.; Ryu, S.; Park, K.; Choe, Y.J.; Ham, J.; Kim, W.Y. Predicting Drug–Target Interaction Using a Novel Graph Neural
1194 Network with 3D Structure-Embedded Graph Representation. *Journal of chemical information and modeling* **2019**, *59*,
1195 3981–3988, doi:10.1021/acs.jcim.9b00387.
- 1196 65. Kwon, Y.; Shin, W.-H.; Ko, J.; Lee, J. AK-Score: Accurate Protein-Ligand Binding Affinity Prediction Using an Ensemble of
1197 3D-Convolutional Neural Networks. *International Journal of Molecular Sciences* **2020**, *21*, 8424, doi:10.3390/ijms21228424.
- 1198 66. Wang, R.; Lai, L.; Wang, S. Further development and validation of empirical scoring functions for structure-based binding
1199 affinity prediction. *J Comput Aided Mol Des* **2002**, *16*, 11–26, doi:10.1023/a:1016357811882.
- 1200 67. Korb, O.; Stütze, T.; Exner, T.E. Empirical scoring functions for advanced protein-ligand docking with PLANTS. *Journal of*
1201 *chemical information and modeling* **2009**, *49*, 84–96, doi:10.1021/ci800298z.
- 1202 68. Gomes, J.; Ramsundar, B.; Feinberg, E.N.; Pande, V.S. Atomic convolutional networks for predicting protein-ligand
1203 binding affinity. *arXiv preprint* **2017**, doi:arXiv:1703.10603.
- 1204 69. Wu, Z.; Ramsundar, B.; Feinberg, E.N.; Gomes, J.; Geniesse, C.; Pappu, A.S.; Leswing, K.; Pande, V. MoleculeNet: a
1205 benchmark for molecular machine learning. *Chemical Science* **2018**, *9*, 513–530, doi:10.1039/C7SC02664A.
- 1206 70. Feinberg, E.N.; Sur, D.; Wu, Z.; Husic, B.E.; Mai, H.; Li, Y.; Sun, S.; Yang, J.; Ramsundar, B.; Pande, V.S. PotentialNet for
1207 Molecular Property Prediction. *ACS Central Science* **2018**, *4*, 1520–1530, doi:10.1021/acscentsci.8b00507.
- 1208 71. Li, Y.; Yang, J. Structural and Sequence Similarity Makes a Significant Impact on Machine-Learning-Based Scoring
1209 Functions for Protein–Ligand Interactions. *Journal of chemical information and modeling* **2017**, *57*, 1007–1012,
1210 doi:10.1021/acs.jcim.7b00049.
- 1211 72. Tosstorff, A.; Rudolph, M.G.; Cole, J.C.; Reutlinger, M.; Kramer, C.; Schaffhauser, H.; Nilly, A.; Flohr, A.; Kuhn, B. A high
1212 quality, industrial data set for binding affinity prediction: performance comparison in different early drug discovery
1213 scenarios. *Journal of Computer-Aided Molecular Design* **2022**, *36*, 753–765, doi:10.1007/s10822-022-00478-x.
- 1214 73. Huang, N.; Shoichet, B.K.; Irwin, J.J. Benchmarking Sets for Molecular Docking. *Journal of medicinal chemistry* **2006**, *49*,
1215 6789–6801, doi:10.1021/jm0608356.
- 1216 74. Mysinger, M.M.; Carchia, M.; Irwin, J.J.; Shoichet, B.K. Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and
1217 Decoys for Better Benchmarking. *Journal of medicinal chemistry* **2012**, *55*, 6582–6594, doi:10.1021/jm300687e.
- 1218 75. Chen, L.; Cruz, A.; Ramsey, S.; Dickson, C.J.; Duca, J.S.; Hornak, V.; Koes, D.R.; Kurtzman, T. Hidden bias in the DUD-E
1219 dataset leads to misleading performance of deep learning in structure-based virtual screening. *PLOS ONE* **2019**, *14*,
1220 e0220113, doi:10.1371/journal.pone.0220113.
- 1221 76. Wallach, I.; Dzamba, M.; Heifets, A. AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in
1222 Structure-based Drug Discovery. *arXiv e-prints* **2015**, arXiv:1510.02855.

- 1223 77. Koes, D.R.; Baumgartner, M.P.; Camacho, C.J. Lessons learned in empirical scoring with smina from the CSAR 2011
1224 benchmarking exercise. *Journal of chemical information and modeling* **2013**, *53*, 1893-1904, doi:10.1021/ci300604z.
- 1225 78. Bauer, M.R.; Ibrahim, T.M.; Vogel, S.M.; Boeckler, F.M. Evaluation and Optimization of Virtual Screening Workflows with
1226 DEKOIS 2.0 – A Public Library of Challenging Docking Benchmark Sets. *Journal of chemical information and modeling* **2013**,
1227 *53*, 1447-1462, doi:10.1021/ci400115b.
- 1228 79. Wójcikowski, M.; Ballester, P.J.; Siedlecki, P. Performance of machine-learning scoring functions in structure-based virtual
1229 screening. *Sci Rep* **2017**, *7*, 46710, doi:10.1038/srep46710.
- 1230 80. Chen, P.; Ke, Y.; Lu, Y.; Du, Y.; Li, J.; Yan, H.; Zhao, H.; Zhou, Y.; Yang, Y. DLIGAND2: an improved knowledge-based
1231 energy function for protein–ligand interactions using the distance-scaled, finite, ideal-gas reference state. *Journal of*
1232 *Cheminformatics* **2019**, *11*, 52, doi:10.1186/s13321-019-0373-4.
- 1233 81. Ballester, P.J. Selecting machine-learning scoring functions for structure-based virtual screening. *Drug Discovery Today:*
1234 *Technologies* **2019**, *32-33*, 81-87, doi:10.1016/j.ddtec.2020.09.001.
- 1235 82. Yasuo, N.; Sekijima, M. Improved Method of Structure-Based Virtual Screening via Interaction-Energy-Based Learning.
1236 *Journal of chemical information and modeling* **2019**, *59*, 1050-1061, doi:10.1021/acs.jcim.8b00673.
- 1237 83. Riniker, S.; Landrum, G.A. Open-source platform to benchmark fingerprints for ligand-based virtual screening. *Journal of*
1238 *Cheminformatics* **2013**, *5*, 26, doi:10.1186/1758-2946-5-26.
- 1239 84. Imrie, F.; Bradley, A.R.; van der Schaar, M.; Deane, C.M. Protein Family-Specific Models Using Deep Neural Networks
1240 and Transfer Learning Improve Virtual Screening and Highlight the Need for More Data. *Journal of chemical information*
1241 *and modeling* **2018**, *58*, 2319-2330, doi:10.1021/acs.jcim.8b00350.
- 1242 85. Trott, O.; Olson, A.J. AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient
1243 optimization, and multithreading. *J Comput Chem* **2010**, *31*, 455-461, doi:10.1002/jcc.21334.
- 1244 86. Brocidiacono, M.; Francoeur, P.; Aggarwal, R.; Popov, K.; Koes, D.; Tropsha, A. BigBind: Learning from Nonstructural
1245 Data for Structure-Based Virtual Screening. *ChemRxiv preprint* **2022**, doi:10.26434/chemrxiv-2022-2t0dq-v3.
- 1246 87. Fan, F.J.; Shi, Y. Effects of data quality and quantity on deep learning for protein-ligand binding affinity prediction.
1247 *Bioorganic & Medicinal Chemistry* **2022**, *72*, 117003, doi:<https://doi.org/10.1016/j.bmc.2022.117003>.
- 1248 88. Rohrer, S.G.; Baumann, K. Maximum Unbiased Validation (MUV) Data Sets for Virtual Screening Based on PubChem
1249 Bioactivity Data. *Journal of chemical information and modeling* **2009**, *49*, 169-184, doi:10.1021/ci8002649.
- 1250 89. Pérez, A.; Martínez-Rosell, G.; De Fabritiis, G. Simulations meet machine learning in structural biology. *Current opinion in*
1251 *structural biology* **2018**, *49*, 139-144, doi:10.1016/j.sbi.2018.02.004.
- 1252

1253 **Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual
1254 author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury
1255 to people or property resulting from any ideas, methods, instructions or products referred to in the content.