The Author(s) *BMC Bioinformatics* 2016, **17**(Suppl 11):308
DOI 10.1186/s12859-016-1169-4

BMC Bioinformatics

RESEARCH Open Access

Correcting the impact of docking pose generation error on binding affinity prediction

Hongjian Li[1], Kwong-Sak Leung[1], Man-Hon Wong[1] and Pedro J. Ballester[2,3,4,5*]

From 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB 2014) Cambridge, UK. 26-28 June 2014

Abstract

Background: Pose generation error is usually quantified as the difference between the geometry of the pose generated by the docking software and that of the same molecule co-crystallised with the considered protein. Surprisingly, the impact of this error on binding affinity prediction is yet to be systematically analysed across diverse protein-ligand complexes.

Results: Against commonly-held views, we have found that pose generation error has generally a small impact on the accuracy of binding affinity prediction. This is also true for large pose generation errors and it is not only observed with machine-learning scoring functions, but also with classical scoring functions such as AutoDock Vina. Furthermore, we propose a procedure to correct a substantial part of this error which consists of calibrating the scoring functions with re-docked, rather than co-crystallised, poses. In this way, the relationship between Vina-generated protein-ligand poses and their binding affinities is directly learned. As a result, test set performance after this error-correcting procedure is much closer to that of predicting the binding affinity in the absence of pose generation error (i.e. on crystal structures). We evaluated several strategies, obtaining better results for those using a single docked pose per ligand than those using multiple docked poses per ligand.

Conclusions: Binding affinity prediction is often carried out on the docked pose of a known binder rather than its co-crystallised pose. Our results suggest than pose generation error is in general far less damaging for binding affinity prediction than it is currently believed. Another contribution of our study is the proposal of a procedure that largely corrects for this error. The resulting machine-learning scoring function is freely available at http://istar.cse.cuhk.edu.hk/rf-score-4.tgz and http://ballester.marseille.inserm.fr/rf-score-4.tgz.

Keywords: Molecular docking, Binding affinity, Drug discovery, Machine learning

Background

Molecular docking tools are routinely utilised to predict the binding pose as well as the binding affinity of a ligand, usually a small organic molecule, bound to a target protein of interest. On one hand, the predicted pose suggests putative intermolecular interactions that can be helpful to understand the mechanism of protein-ligand binding. On the other hand, the predicted affinity prioritizes strong-binding ligands over weak-binding ones from a large library of compounds to evaluate.

A typical docking program implements a sampling algorithm to generate possible binding poses and a scoring function to estimate their binding affinity. The former operation is known as pose generation, and the latter is known as scoring. For example, modern docking tools such as AutoDock Vina [1] and idock [2] are currently capable of generating near-native poses with a redocking success rate of over 50 % on three diverse benchmarks [2].

Recent years have seen the emergence and prosperity of a new class of scoring functions that use machine learning techniques to increase the accuracy of binding

*Correspondence: pedro.ballester@inserm.fr
[2]Cancer Research Center of Marseille, INSERM U1068, F-13009 Marseille, France
[3]Institut Paoli-Calmettes, F-13009 Marseille, France
Full list of author information is available at the end of the article

affinity prediction (a first review on machine-learning scoring functions has now been published [3]). RF-Score [4] was the first machine-learning scoring function introducing a substantial improvement over classical scoring functions. Since then, several enhancements have been introduced, thereby resulting in RF-Score-v2 [5] and RF-Score-v3 [6], and other relevant studies [7]. RF-Score has been utilised [8] to successfully discover a large number of innovative binders of antibacterial DHQase2 targets, demonstrating its practical utility. To promote its use, RF-Score-v3 has been incorporated into a user-friendly webserver called istar [2], available at http://istar.cse.cuhk.edu.hk/idock, for large-scale docking-based prospective virtual screening. Furthermore, recent study [9] has investigated the benefit of training machine-learning scoring functions with low-quality structural and interaction data.

In prospective structure-based virtual screening [2], scoring of the docked poses of a molecule is required because the experimentally determined pose is not available in most cases. Therefore, accurate prediction of binding affinity of docked poses, rather than co-crystallised poses, is required for ranking compounds from screening libraries.

Pose generation error is typically measured by comparing the geometry of the pose generated by the docking software and that of the same molecule co-crystallised with the considered protein (Fig. 1). The impact of this error on binding affinity prediction is yet to be systematically analysed across diverse protein-ligand complexes. In this study we investigate the impact of pose generation error on the predictive performance of both classical and machine-learning scoring functions, and propose a novel approach to correct such error. Furthermore, we release free software implementing these improvements.

Methods

This section introduces and motivates the use of four scoring functions building upon AutoDock Vina, two benchmarks to evaluate and compare performance of these scoring functions, the performance metrics, and the experimental setup.

Model 1 - AutoDock Vina

AutoDock Vina [1] was chosen as a baseline scoring function because of its popularity among the research community. Vina's popularity roots in its substantial improvements on both the average accuracy of the binding pose prediction and the running speed. Its remarkable performance in pose generation as well as its open source nature are other appealing aspects of this widely-used tool.

Like all classical scoring functions [6], Vina assumes a predetermined functional form. Vina's score for the kth pose of a molecule is given by the predicted free energy of binding to the target protein and computed as:

$$e'_k = \frac{e_{k,inter} + e_{k,intra} - e_{1,intra}}{1 + w_6 N_{rot}} \tag{1}$$

where

$$\begin{aligned} e_{k,inter} &= w_1 \cdot Gauss1_k \\ &\quad + w_2 \cdot Gauss2_k \\ &\quad + w_3 \cdot Repulsion_k \\ &\quad + w_4 \cdot Hydrophobic_k \\ &\quad + w_5 \cdot HBonding_k \end{aligned} \tag{2}$$

$$\begin{aligned} w_1 &= -0.035579 \\ w_2 &= -0.005156 \\ w_3 &= 0.840245 \\ w_4 &= -0.035069 \\ w_5 &= -0.587439 \\ w_6 &= 0.05846 \end{aligned} \tag{3}$$

e'_k is the predicted free energy of binding reported by the Vina software when scoring the kth docked pose. $e_{k,inter}$ and $e_{k,intra}$ are the inter-molecular and intra-molecular contributions, respectively, which have both the same functional form described in Eq. 2 but are summed over different atom pairs. The values for the six weights were calculated by OLS (Ordinary Least Squares) using a non-linear optimisation algorithm as it has been the case in related force-field scoring functions [10], although this process was not fully disclosed in the original publication [1]. N_{rot} is the calculated number of rotatable bonds. The predicted free energy of binding in kcal/mol units was converted into pK_d units with $pK_d = -0.73349480509e$ so as to compare to binding affinities in pK_d or pK_i units. Mathematical expressions and further explanations can be found in [2].

Unlike our previous study [6] on scoring crystal poses, where $k = 1$ because only the crystal pose was considered, this study aims at training and testing on docked poses, so k will range from 1 to 9 depending on the specific pose to use for each molecule (Vina returns a maximum of 9 poses per docking run). Thus $e_{k,intra}$ and $e_{1,intra}$ do not necessarily cancel out. As a result, the five terms from $e_{k,intra}$ were considered as additional features in models 2, 3 and 4.

Model 2 - MLR::Vina

This model retains the 11 unweighted Vina terms (5 from $e_{k,inter}$, 5 from $e_{k,intra}$, and N_{rot}) as features, but changes the regression method to multiple linear regression (MLR), a regression model commonly adopted by

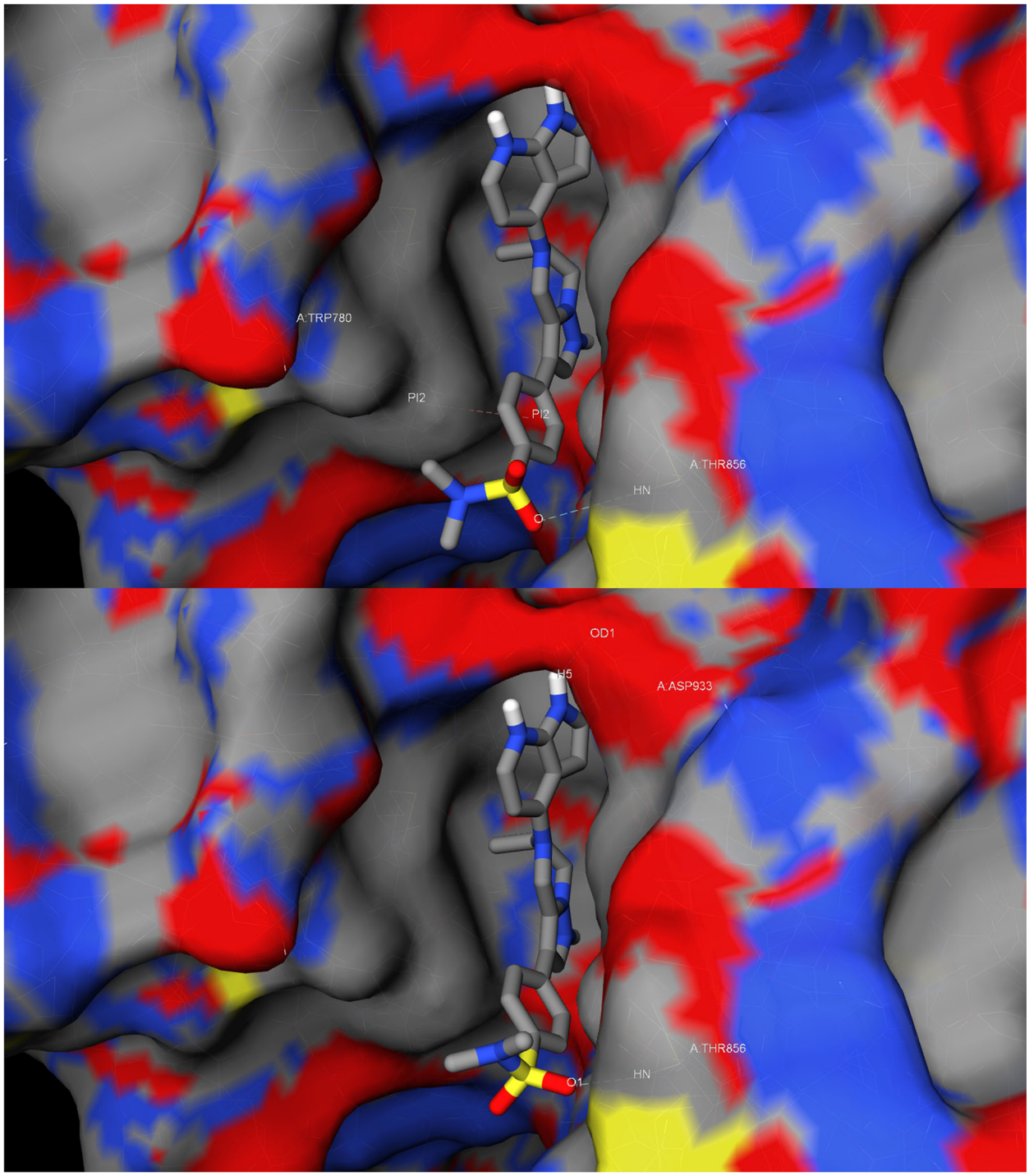

Fig. 1 Example of pose generation error. *Top*: crystal structure of PI3Kα in complex of a tetrahydropyrazolo[1,5-a]pyrazine codenamed 3K6 (PDB ID: 4WAF). *Bottom*: re-docked pose of 3K6, generated by idock [2]. Hydrogen bonds are rendered as dashed cyan lines, and π stackings are rendered as dashed pink lines. The RMSD (Root-Mean Square Deviation) between the co-crystallised pose and the re-docked pose of 3K6 is 1.15 Å, which is a quantitative measure of pose generation error. These two plots were created by iview [17], an interactive WebGL visualizer that circumvents the requirement of Java, yet supports the construction of macromolecular surface and the display of virtual reality effects and molecular interactions. iview is freely available at http://istar.cse.cuhk.edu.hk/iview/

classical scoring functions, such as empirical scoring functions. The use of MLR implies an additive functional form and therefore MLR::Vina is a classical scoring function [6].

Vina's scoring function is not exactly a sum of energetic terms because $w_6 \neq 0$ (although the denominator of Eq. 1 is close to 1 because of the low value of w_6. In order to make the problem amenable to MLR, we performed a grid search on w_6 and thereafter ran MLR on the remaining weights. More precisely, we sampled 101 values for w_6 from 0 to 1 with a step size of 0.01. Interestingly we found that the w_6 values of the best models were always between 0.000 and 0.030. Then we again sampled 31 values for w_6 in this range with a step size of 0.001, and used the w_6 value that resulted in the lowest RMSE (Root Mean Square Error) on the test set.

Model 3 - RF::Vina

This model also retains the 11 unweighted Vina terms as features, but changes the regression method to Random Forest (RF) [11], so as to implicitly learn the functional form from the data. Hence this model circumvents the modelling assumption of a predetermined functional form and thus allows to investigate the impact of such modelling assumption by comparing RF::Vina to MLR::Vina. Besides RF, other machine learning techniques such as SVR (Support Vector Regression) [12] can certainly be applied to this problem, although this is out of the scope of this study.

A RF is an ensemble of different decision trees randomly generated from the same training data via bootstraping [11]. RF trains its constituent trees using the CART algorithm [13], and selects the best data split at each node of the tree from a typically small number (mtry) of randomly chosen features. In regression applications, the RF prediction is given by arithmetic mean of all the individual tree predictions in the forest.

For each value of the mtry parameter from 1 to all 11 features, we built a RF model with 500 trees, as we and others [14] have not observed any substantial gain in performance by training RF with a higher number of trees on this class of problems. The selected model was the one that led to the lowest RMSE on a subset of training data of each tree collectively known as the OOB (Out of Bag) data. Because RF is stochastic, this process was repeated ten times with ten different random seeds. The predictive performance was reported for the RF with the best seed that resulted in the lowest RMSE on the test set. Further details on RF model building in this context can be found in [6].

Model 4 - RF::VinaElem

This model retains RF as the regression method, but expands the feature set to 47 features by adding the 36 RF-Score [4] features. Like in the training process of RF::Vina, the same ten seeds were used, and for a given random seed, a RF model for each mtry value from 1 to 47 was built and that with the lowest RMSE on OOB data was selected. The predictive performance was reported for the RF with the best seed that led to the lowest RMSE on the test set.

RF-Score features are defined as the occurrence count of intermolecular contacts between elemental atom types i and j, as shown in Eqs. 4 and 5, where d_{kl} is the Euclidean distance between the kth protein atom of type j and the lth ligand atom of type i calculated from a structure; K_j is the total number of protein atoms of type j ($\#\{j\} = 4$, considered protein atom types are C, N, O, S) and L_i is the total number of ligand atoms of type i ($\#\{i\} = 9$, considered ligand atom types are C, N, O, F, P, S, Cl, Br, I); $\mathcal{H}$ is the Heaviside step function that counts contacts within a neighbourhood of d_{cutoff} Å. For instance, $x_{7,8}$ is the number of occurrences of ligand nitrogen atoms (i=7) hypothetically interacting with protein oxygen atoms (j=8) within a chosen neighbourhood. Full details on RF-Score features are available in [4, 12].

$$x_{ij} = \sum_{k=1}^{K_j} \sum_{l=1}^{L_i} \mathcal{H}(d_{cutoff} - d_{kl}) \tag{4}$$

$$\mathbf{x} = \{x_{ij}\} \in N^{36} \tag{5}$$

PDBbind v2007 benchmark

We adopted the PDBbind v2007 benchmark [15], arguably the most widely used [6, 7] for binding affinity prediction of diverse complexes. Its test set comprises 195 diverse complexes from the core set, whereas its training set comprises 1105 non-overlapping complexes from the refined set. Both the test and training sets come with measured binding affinities spanning more than 12 orders of magnitude. This benchmark has the advantage of permitting a direct comparison against the same four models that were trained and tested on crystal poses [6] of this benchmark.

PDBbind v2013 blind benchmark

We also adopted the PDBbind v2013 blind benchmark [6], a recently proposed new benchmark mimicking a blind test to provide a more realistic validation than the PDBbind v2007 benchmark. Its test set is composed of all the complexes in the PDBbind v2013 refined set that were not in the v2012 refined set, i.e. those 382 complexes that were newly added in the v2013 release. Its training set is simply the v2012 refined set, which contains 2897 complexes. By construction, this benchmark can be regarded as a blind test in that only data available until a certain year is used to build the scoring function that will be used to predict the binding affinity of future complexes as if these had not yet been measured. Consequently, the test set and training

set do not overlap. Again, this benchmark has the advantage of permitting a direct comparison against the same four models that were trained and tested on crystal poses [6] of this benchmark.

In addition to the above training set, three more training sets were added in order to study how the performance of the four models would vary given different number of training complexes. The refined sets of PDBbind v2002 (N=792), v2007 (N=1300), v2010 (N=2057) and v2012 (N=2897) were chosen so that there is approximately the same number of complexes between consecutive releases. Complexes containing metal ions not supported by Vina were discarded. More details about this benchmark can be found in [6].

Performance measures

As usual [15], predictive performance was quantified by the Root Mean Square Error (RMSE), Standard Deviation (SD), Pearson correlation (Rp) and Spearman rank-correlation (Rs) between predicted and measured binding affinities. Their mathematical expressions are shown in Eqs. 6, 7, 8, and 9. Given a scoring function f and the measured binding affinity $y^{(n)}$ and the features $\overrightarrow{x}^{(n)}$ characterising the nth complex out of N complexes in the test set, $p^{(n)} = f(\overrightarrow{x}^{(n)})$ is the predicted binding affinity, $\{\hat{p}^{(n)}\}$ are the fitted values from the linear model between $\{y^{(n)}\}$ and $\{p^{(n)}\}$ on the test set, whereas $\{y_r^{(n)}\}$ and $\{p_r^{(n)}\}$ are the rankings of $\{y^{(n)}\}$ and $\{p^{(n)}\}$, respectively. Note that SD was calculated in a linear correlation, but RMSE was not. Lower values in RMSE and SD and higher values in Rp and Rs indicate a better predictive performance.

$$RMSE = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(p^{(n)} - y^{(n)}\right)^2} \tag{6}$$

$$SD = \sqrt{\frac{1}{N-2}\sum_{n=1}^{N}\left(\hat{p}^{(n)} - y^{(n)}\right)^2} \tag{7}$$

$$R_p = \frac{N\sum_{n=1}^{N} p^{(n)}y^{(n)} - \sum_{n=1}^{N} p^{(n)}\sum_{n=1}^{N} y^{(n)}}{\sqrt{\left(N\sum_{n=1}^{N}(p^{(n)})^2 - \left(\sum_{n=1}^{N} p^{(n)}\right)^2\right)\left(N\sum_{n=1}^{N}(y^{(n)})^2 - \left(\sum_{n=1}^{N} y^{(n)}\right)^2\right)}} \tag{8}$$

$$R_s = \frac{N\sum_{n=1}^{N} p_r^{(n)}y_r^{(n)} - \sum_{n=1}^{N} p_r^{(n)}\sum_{n=1}^{N} y_r^{(n)}}{\sqrt{\left(N\sum_{n=1}^{N}(p_r^{(n)})^2 - \left(\sum_{n=1}^{N} p_r^{(n)}\right)^2\right)\left(N\sum_{n=1}^{N}(y_r^{(n)})^2 - \left(\sum_{n=1}^{N} y_r^{(n)}\right)^2\right)}} \tag{9}$$

The Root Mean Square Deviation (RMSD) measures how geometrically different the redocked pose is from the corresponding co-crystallized pose of the same ligand molecule, i.e. the pose generation error. Suppose N_a is the number of heavy atoms, $\left(x_c^{(n)}, y_c^{(n)}, z_c^{(n)}\right)$ and $\left(x_d^{(n)}, y_d^{(n)}, z_d^{(n)}\right)$ are the 3D coordinate of the nth heavy atom of the crystal and docked poses, respectively, the pose generation error is calculated as:

$$RMSD = \sqrt{\frac{1}{N_a}\sum_{n=1}^{N_a}\left[\left(x_c^{(n)} - x_d^{(n)}\right)^2 + \left(y_c^{(n)} - y_d^{(n)}\right)^2 + \left(z_c^{(n)} - z_d^{(n)}\right)^2\right]} \tag{10}$$

Experimental setup

To generate docked poses, each ligand in the two benchmarks was docked into the binding site of its target protein using Vina with its default settings. This process is known as redocking. The search space was defined by finding the smallest cubic box that covers the entire ligand and then by extending the box by 10Å in X, Y, Z dimensions. Water molecules were removed, while metal ions recognized by Vina were retained as part of the protein. This preprocessing procedure is commonly adopted in the development of both classical scoring functions [1] and machine-learning scoring functions [16].

Redocking a ligand into its cognate protein resulted in up to nine docked poses. Thus, the question arises of which pose best represents its molecule for calculating the values of the features. Here we evaluate different schemes referring to the specific pose from which the features are extracted. In scheme 1, the chosen pose is the crystal pose. In scheme 2, the chosen pose is the docked pose with the best Vina score, i.e. the one with the lowest Vina score in terms of estimated free energy of binding in kcal/mol units. We trained the four models on both crystal and docked poses (in both schemes), and tested them also on both crystal and docked poses (in both schemes).

To make our experiments comprehensive, we also evaluated additional schemes. In scheme 3, the chosen pose is the docked pose with the lowest RMSD. In scheme 4, the chosen pose is the docked pose with a Vina score closest to the measured binding affinity. In scheme 5, the chosen poses are all the 9 docked poses, which hence results in a 9 times larger feature set (the number of features is 91 for models 2 and 3, and 415 for model 4). For ligands with less than 9 docked poses returned, the features extracted from the pose with the lowest Vina score are repeated as many times as poses are missing. In scheme 6, the chosen poses are the 2 docked poses with the lowest and the second lowest Vina score, which hence results in a double-sized feature set (the number of features is 21 for models 2 and 3, and 93 for model 4). For ligands with less than 2 docked poses outputted, the features extracted from the pose with the lowest Vina score are repeated. The rationale of introducing these schemes is that, schemes 1 to 4 help to determine which particular pose would be useful

for improving predictive accuracy, while schemes 5 and 6 help to examine the effect of pose ensemble instead of a single pose.

Hereafter whenever we mention the docked pose, we implicitly refer to the one with the best Vina score (scheme 2), if not specified otherwise.

Results

Pose generation error slightly worsens binding affinity prediction

This question was analysed by using schemes 1 and 2. After redocking by Vina, we used RMSD to quantify the pose generation error, i.e. how different the 3D geometry of the redocked pose is from the corresponding crystal pose of the same ligand molecule. A RMSD value of 2Å was used as a commonly accepted threshold for a correctly reproduced crystal pose. 101 out of the 195 ligands (52 %) in the PDBbind v2007 benchmark and 219 out of the 382 ligands (57 %) in the PDBbind v2013 blind benchmark had their best-scoring docked pose with RMSD < 2Å. When all the docked poses of the molecule were considered, the redocking success rate of the two benchmarks increased to 76 % (149 out of 195) and 81 % (311 out of 382), respectively. These results are consistent with the previous results obtained in [2], where Vina managed to predict a pose sufficiently close to that of the co-crystallized ligand as the best-scoring pose in over half of the cases.

Tables 1 and 2 show the predictive performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses on the PDBbind v2007 benchmark and the PDBbind v2013 blind benchmark, respectively. Figures 2 and 3 visualize the same results using boxplots, as RF models are stochastic. Note that Vina (model 1) was trained on crystal poses and used out-of-the-box without re-training, so its results for training scheme 2 are simply a duplicate of its results for training scheme 1.

From these results, several interesting observations can be made. First, for model 1, its performance tested on docked poses was always better than its performance tested on crystal poses (except for just a small degradation in the Rs performance on the PDBbind v2007 benchmark). Particularly, the RMSE error was greatly dropped from 2.41 to 2.02 on the PDBbind v2007 benchmark and from 2.30 to 1.87 on the PDBbind v2013 blind benchmark. The result that Vina made better prediction of binding affinity from docked poses than from crystal poses is possibly due to the fact that docked poses are by construction optima of the objective function spanned by the Vina score, which may favor prediction of docked poses over unoptimized crystal poses.

Second, for models 2, 3 and 4 trained on crystal poses, their performance tested on docked poses was always worse than their performance tested on crystal poses (e.g. by comparing second and first columns in the Rs plot of Fig. 3). This is well anticipated because of the presence of pose generation error. For instance, on the PDBbind v2013 blind benchmark, model 4 trained on crystal poses obtained Rs=0.662 when tested on crystal poses

Table 1 Performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses (schemes 1 and 2) on the PDBbind v2007 benchmark. Comparing the same models from the two first blocks (crystal:crystal and crystal:docked) shows that the pose generation error also introduces a small degradation in the test set performance. Making the same comparisons between the second and fourth blocks shows that a substantial part of this error has been corrected

Model	Training	Test	RMSE	SD	Rp	Rs
1 (Vina)	Crystal	Crystal	2.41	1.99	0.554	0.608
2 (MLR::Vina)	Crystal	Crystal	1.88	1.85	0.630	0.680
3 (RF::Vina)	Crystal	Crystal	1.66	1.59	0.744	0.752
4 (RF::VinaElem)	Crystal	Crystal	1.52	1.42	0.803	0.799
1 (Vina)	Crystal	Docked	2.02	1.98	0.557	0.597
2 (MLR::Vina)	Crystal	Docked	1.90	1.87	0.622	0.670
3 (RF::Vina)	Crystal	Docked	1.76	1.72	0.693	0.710
4 (RF::VinaElem)	Crystal	Docked	1.60	1.52	0.772	0.771
2 (MLR::Vina)	Docked	Crystal	1.91	1.88	0.618	0.648
3 (RF::Vina)	Docked	Crystal	1.74	1.69	0.705	0.716
4 (RF::VinaElem)	Docked	Crystal	1.58	1.45	0.794	0.790
2 (MLR::Vina)	Docked	Docked	1.86	1.83	0.640	0.667
3 (RF::Vina)	Docked	Docked	1.69	1.63	0.730	0.730
4 (RF::VinaElem)	Docked	Docked	1.55	1.45	0.795	0.789

Table 2 Performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses (schemes 1 and 2) on the PDBbind v2013 blind benchmark

Model	Training	Test	RMSE	SD	Rp	Rs
1 (Vina)	Crystal	Crystal	2.30	1.81	0.406	0.414
2 (MLR::Vina)	Crystal	Crystal	1.67	1.67	0.535	0.521
3 (RF::Vina)	Crystal	Crystal	1.54	1.54	0.629	0.593
4 (RF::VinaElem)	Crystal	Crystal	1.43	1.43	0.689	0.662
1 (Vina)	Crystal	Docked	1.87	1.78	0.437	0.432
2 (MLR::Vina)	Crystal	Docked	1.70	1.69	0.520	0.505
3 (RF::Vina)	Crystal	Docked	1.61	1.60	0.585	0.549
4 (RF::VinaElem)	Crystal	Docked	1.49	1.49	0.656	0.633
2 (MLR::Vina)	Docked	Crystal	1.69	1.69	0.521	0.509
3 (RF::Vina)	Docked	Crystal	1.62	1.61	0.580	0.560
4 (RF::VinaElem)	Docked	Crystal	1.48	1.47	0.669	0.650
2 (MLR::Vina)	Docked	Docked	1.68	1.68	0.524	0.509
3 (RF::Vina)	Docked	Docked	1.59	1.59	0.594	0.553
4 (RF::VinaElem)	Docked	Docked	1.47	1.48	0.665	0.643

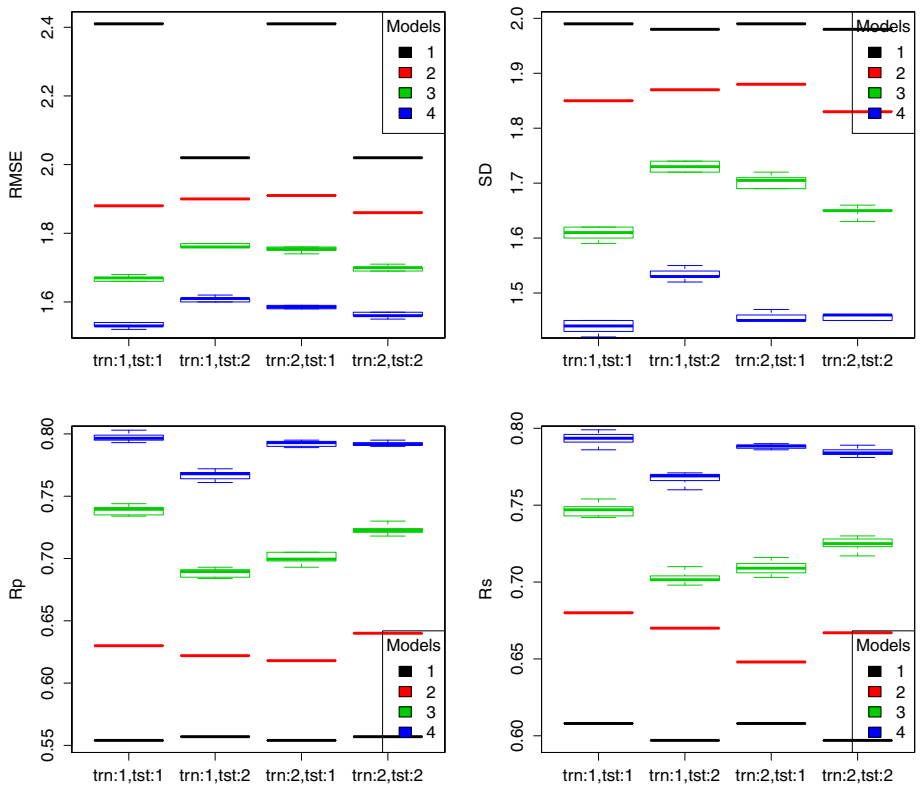

Fig. 2 Box plots of performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses of the PDBbind v2007 benchmark. Model 1 is AutoDock Vina, model 2 is MLR::Vina, model 3 is RF::Vina, and model 4 is RF::VinaElem. In the x axis, trn:1 means the four models were trained in scheme 1, i.e. on crystal poses, and trn:2 means the four models were trained in scheme 2, i.e. on docked poses. Likewise, tst-1 and tst-2 mean the four models were tested on crystal and docked poses, respectively. Model 1 was executed out-of-the-box, so its results for training scheme 1 were repeated for training scheme 2

(Additional file 1). Its performance degraded when tested on docked poses of the same molecules with Rs=0.633 (Additional file 2). The impact of pose generation error on binding affinity prediction is thus quantified by ΔRs=-0.029.

Third, for models 2, 3 and 4 tested on docked poses, their performance was better when they were trained on docked poses than their counterparts trained on crystal poses (e.g. by comparing fourth and second columns in the Rs plot of Fig. 3). In other words, a substantial part of the pose generation was corrected. For instance, on the PDBbind v2013 blind benchmark, model 4 trained on docked poses obtained Rs=0.643 when tested on docked poses (Additional file 3). Hence the impact of pose generation error on binding affinity prediction is reduced in a 33 % (from ΔRs=-0.029 to ΔRs=-0.019). This means that a way to improve performance on docked poses is to train the model on docked poses instead of on crystal poses. Indeed, test set performance after this error-correcting procedure is much closer to that of predicting the binding affinity in the absence of pose generation error, i.e. on crystal structures. In practice, different scoring functions can be built depending on whether one wants to score crystal poses or docked poses.

Fourth, for models 2, 3 and 4 tested on crystal poses, the models trained on docked poses (Additional file 4) did not outperform their counterparts trained on crystal poses. This is also well anticipated due to the impact of pose generation error, and suggests that it is not feasible to improve the predictive performance on crystal poses by using docked poses for training. To sum up, if the desired application is to score a crystal pose, it would be better to train the scoring function on crystal poses; and if the desired application is to score a docked pose, it would be better to train the scoring function on docked poses.

Lastly, regardless of the training or test schemes, model 4 consistently outperformed model 3, which in turn outperformed model 2, which in turn outperformed model 1. It is remarkable that the best scoring function, model 4 (RF::VinaElem), when trained on docked poses, achieved the highest performance in the literature on the PDBbind v2007 benchmark in the more common application of re-scoring docked poses, as it is required when carrying out docking-based prospective virtual screening [2]. Here we denote this version of RF::VinaElem as RF-Score-v4

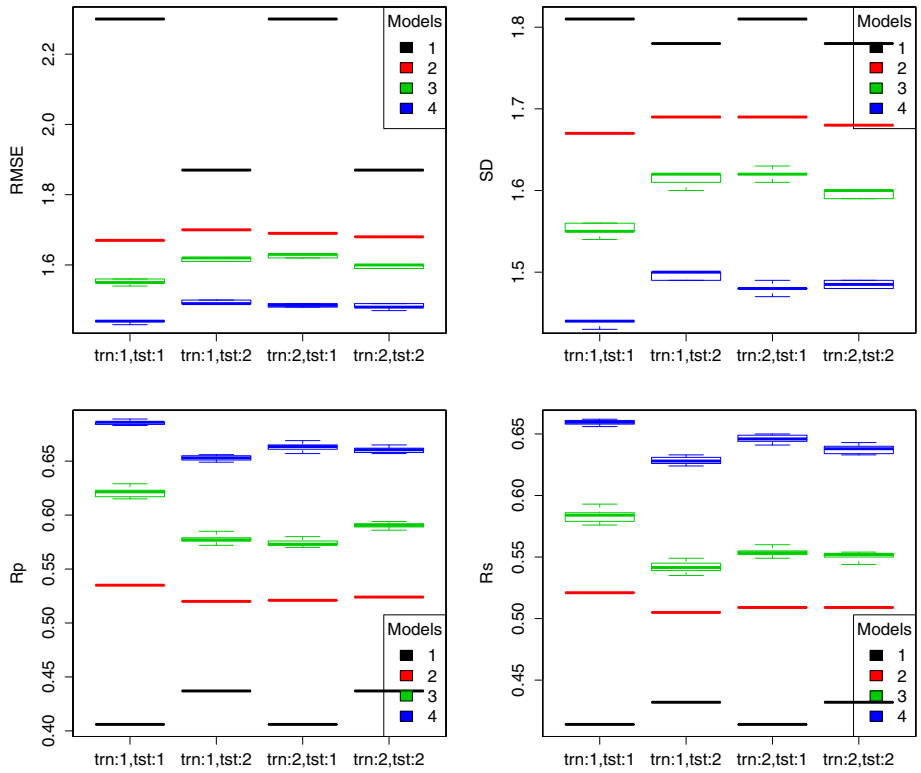

Fig. 3 Box plots of performance of the four models trained on crystal and docked poses and tested also on crystal and docked poses of the PDBbind v2013 blind benchmark. The same notations are applied here as in Fig. 2

specifically for the purpose of binding affinity prediction given a docked pose from Vina. Importantly, since Vina and RF::Vina used the same features and were trained on the same data, RF::Vina performed much better in predicting binding affinity than the widely-used Vina software while having the same applicability domain.

Training with more complexes on docked poses still improves predictive performance

In a previous study [6], with the help of the PDBbind v2013 blind benchmark, we showed that training RF models with larger datasets greatly improved their predictive performance on scoring crystal poses, while the performance of MLR::Vina nearly stayed flat. Here we observe similar results when the models were trained on docked poses and tested also on docked poses. As shown in Fig. 4, when more complexes were used for model training, RF::VinaElem consistently increased its predictive accuracy in terms of RMSE, SD, Rp and Rs. In contrast, for MLR::Vina, its accuracy improvement obtained from larger training sets was just marginal, if not negligible. The performance gap between MLR::Vina and RF::VinaElem is not only substantial, but grows as more data is available for training, thus increasing the importance of employing RF in scoring function development. More importantly, the availability of crystal poses is limited by the number of experimentally resolved structures, whereas docked poses can be generated by docking tools if their binding site is known. This means that, by using docked poses for training, the training data size can be remarkably larger than limiting the training data to crystal poses only, and therefore even higher performance could in principle be achieved by incorporating more training complexes produced by docking.

Correlation between pose generation error and binding affinity prediction error is low

We analyse how different RMSD values affect binding affinity prediction by comparing the RMSD of the docked pose with the individual absolute error in its binding affinity prediction by the four models (note that the square root of the summation of the square of these errors is the RMSE measure). It is widely believed that the higher the pose generation error, the larger the error on predicting the binding affinity of that pose will be. Nevertheless, this is actually not the case.

Figures 5 and 6 plot this information for each of the four scoring functions trained and tested on docked poses of the two benchmarks, respectively. Strikingly, the four scoring functions are particularly robust to pose

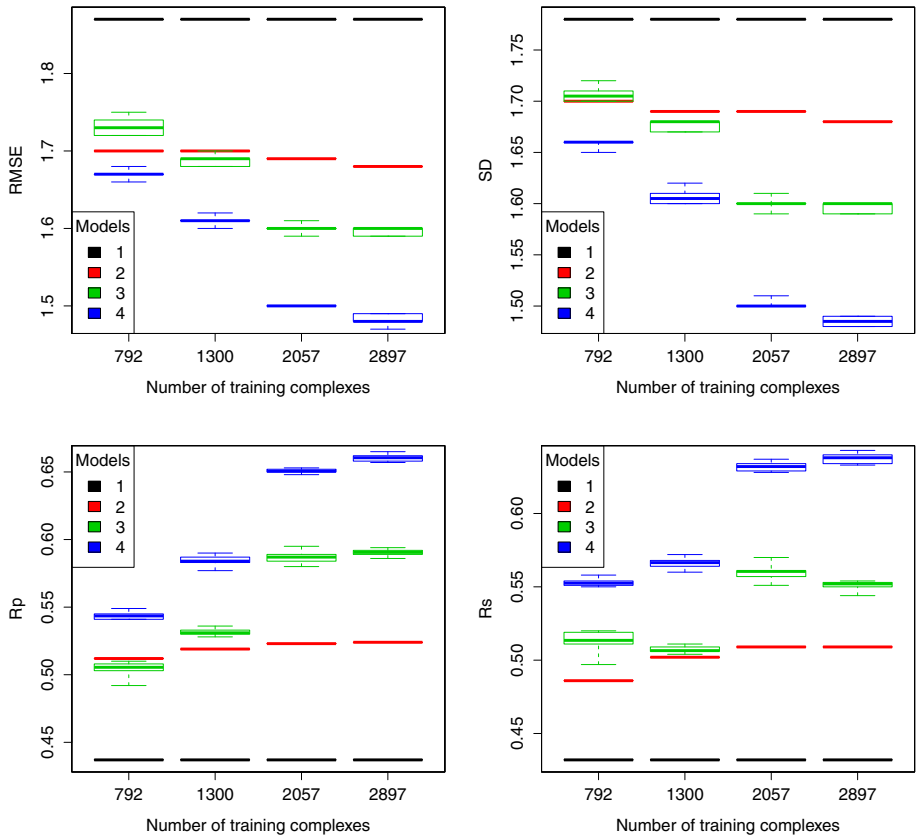

Fig. 4 Box plots of performance of the four models trained on docked poses and tested also on docked poses of the PDBbind v2013 blind benchmark, with four incrementally-sized training sets. Model 1 is AutoDock Vina, model 2 is MLR::Vina, model 3 is RF::Vina, and model 4 is RF::VinaElem

generation error, with reasonably accurate prediction still being obtained in poses with RMSD of almost 15Å. The Rp and Rs values stated at the top of these plots quantify how little the pose generation error generally correlates with the binding affinity prediction error, regardless of whether a classical or machine-learning scoring function is being considered. This is likely to be connected to uncertainty associated to relating a static crystal structure of the complex with its measured binding affinity which is the outcome of the dynamic process of binding, as discussed in [5]. To the best of our knowledge, these behaviour has not been communicated yet for classical scoring functions, which is highly surprising given the intense research that has been carried out over the years in this area. On the other hand, it is noteworthy that, while the binding affinities of some complexes are very well predicted (pK_d error close to 0), some others have errors of more than 7 pK_d units (see the topleft plots for Vina).

Using multiple docked poses for training does not improve predictive performance

In addition to using crystal and docked poses (schemes 1 and 2) for training and testing, we further evaluated several strategies (schemes 3, 4, 5 and 6), aiming to see if using another docked pose of a molecule, or even multiple docked poses, could possibly increase the predictive performance of the resultant models. Remember that scheme 3 uses the docked pose with the lowest RMSD, scheme 4 uses the docked pose with a Vina score closest to the measured binding affinity, scheme 5 uses all the 9 docked poses, and scheme 6 uses the two top-scoring docked poses. In practice, schemes 3 and 4 cannot be used for testing purpose because neither the RMSD nor the measured binding affinity of the test set complexes are known. Hence, models trained in schemes 3 and 4 had to be tested in schemes 1 and 2 instead. On the other hand, models trained in schemes 5 and 6 can only be tested in schemes 5 and 6, respectively, because the same set of features must be used in both training and testing.

The results of schemes 3 to 6 on the two benchmarks are shown in Tables 3 and 4. Interestingly, when tested on crystal poses (in scheme 1), none of the models trained in schemes 3 to 6 outperformed their counterparts trained in scheme 1. Similarly, when tested on docked poses (in scheme 2), none of the models trained in schemes 3 to 6 outperformed their counterparts trained in scheme 2,

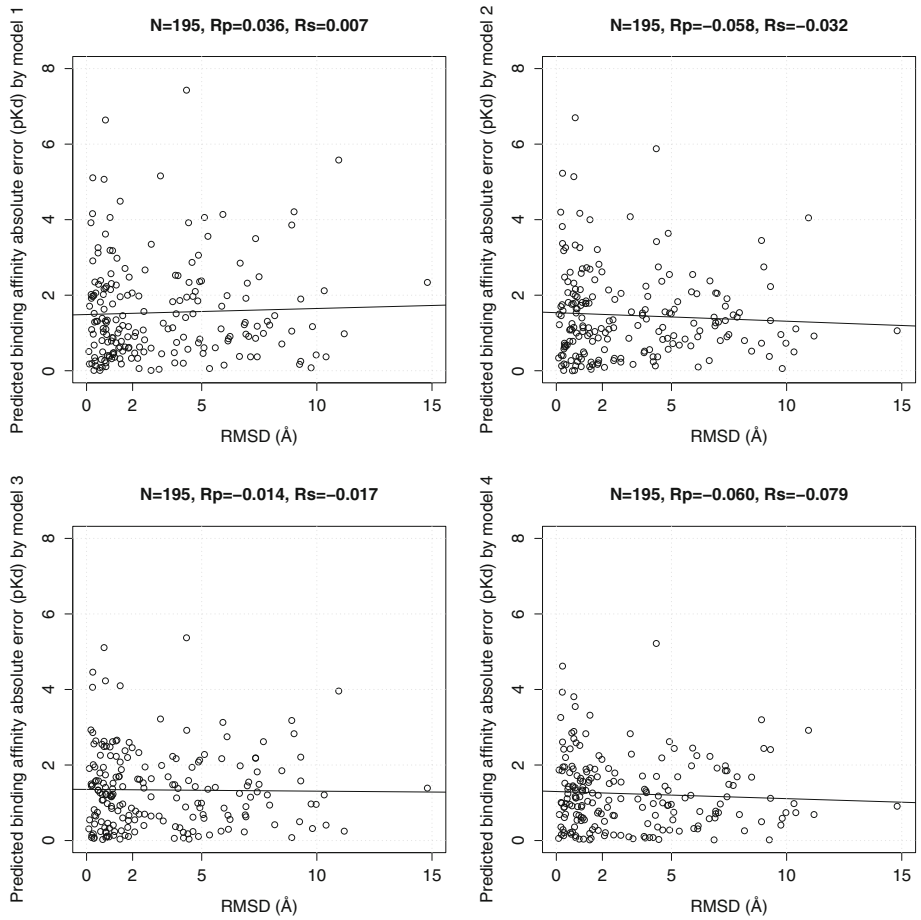

Fig. 5 Correlation plots of predicted binding affinity absolute errors achieved by the four models trained on docked poses and tested on docked poses of the PDBbind v2007 benchmark against the RMSD values from redocking the 195 test set complexes by Vina. Model 1 is AutoDock Vina, model 2 is MLR::Vina, model 3 is RF::Vina, and model 4 is RF::VinaElem

either. The interpretation of such results is two-fold. First, training with the docked pose with the lowest RMSD (scheme 3) or the docked pose with a Vina score closest the measured binding affinity (scheme 4) did not help to improve predictive performance on the test set. Second, training with multiple docked poses of a molecule, instead of a specific single docked pose, did not help to improve predictive performance either. Taken together, these results suggest that a novel way to improve predictive performance on docked poses is to train the scoring functions on docked poses, i.e. those with the best Vina score.

Discussions and conclusions

This is the first study that systematically investigates the impact of pose generation error on binding affinity prediction for both classical and machine-learning scoring functions. Our comprehensive results show that pose generation error only introduces a small degradation in the accuracy of scoring functions. To minimize this negative impact, we found that re-training the scoring functions on docked poses, instead of crystal poses, corrects a substantial part of this degradation. Machine-learning scoring functions are almost always trained on crystal poses and tested on crystal or docked poses without changing composition of training or test sets. Here we still tested the scoring functions on docked poses, but now trained them on docked poses, which has been shown to improve test set performance with respect to the scoring functions trained on crystal poses. In short, one straightforward approach to enhancing predictive accuracy on docked poses is to re-train the scoring function on docked poses.

We have also found that training RF::VinaElem on docked poses with more complexes substantially increased its predictive accuracy, whereas it was not the case for MLR::Vina. Their performance gap will become larger given that more and more structural and

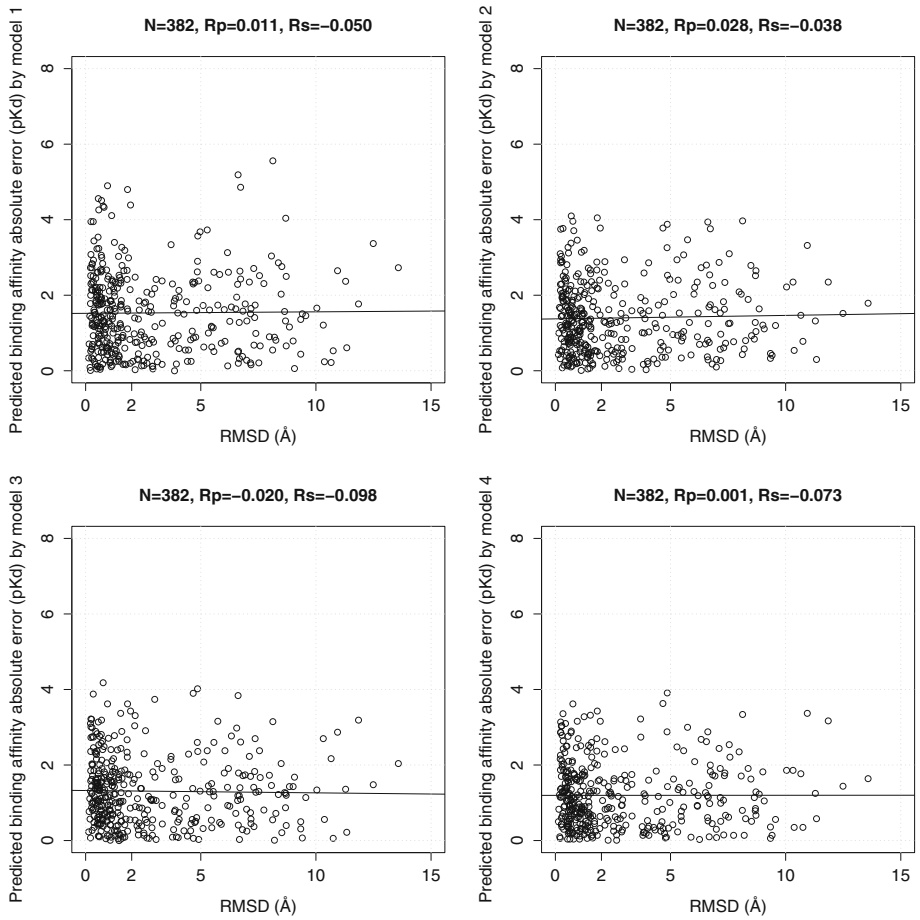

Fig. 6 Correlation plots of predicted binding affinity absolute errors achieved by the four models trained on docked poses and tested on docked poses of the PDBbind v2013 blind benchmark against the RMSD values from redocking the 382 test set complexes by Vina. Model 1 is AutoDock Vina, model 2 is MLR::Vina, model 3 is RF::Vina, and model 4 is RF::VinaElem

interaction data will be available for training in the future. Importantly, whereas the availability of crystal poses is limited by the number of experimentally resolved structures, docked poses of the many known ligands of these targets can be generated by docking tools if their binding site is known. This means that, by using docked poses for training, the training data size can be remarkably larger than limiting the training data to crystal poses only, and therefore even higher performance could in principle be achieved by incorporating more training complexes produced by docking.

Furthermore, we investigated the dependency of RMSD of test set complexes with binding affinity prediction. In contrast to the commonly-held view that the higher the pose generation error, the larger the prediction error of the binding affinity of that pose, we actually observed that the correlation between pose generation error and binding affinity prediction error is low. This indicates that predicting the binding affinity of a docked pose having a large pose generation error is not necessarily more difficult than predicting the binding affinity of a docked pose having a small pose generation error.

Meanwhile, we studied the effect of using docked pose ensemble of a molecule, in addition to merely a single pose, for training scoring functions. This is worth doing because until now existing scoring functions all use just one pose per molecule for training. Although our presented schemes 3 to 6 did not succeed in increasing predictive performance, further analysis of the influence of the number of poses on the error in binding affinity prediction might lead to better performance.

Another contribution of this study is the release of free software implementing RF::VinaElem trained on docked poses, named RF-Score-v4, so that it can be directly used by the large number of Vina users (poses generated from other docking programs can also be re-scored by our software once these are converted to AutoDock's pdbqt format). With this purpose, we have trained the best

Table 3 Performance of models 2, 3, 4 trained in schemes 3, 4, 5, 6 and tested in schemes 1, 2, 5, 6 on the PDBbind v2007 benchmark

Model	Training	Test	RMSE	SD	Rp	Rs
2 (MLR::Vina)	3	1	1.89	1.85	0.629	0.675
3 (RF::Vina)	3	1	1.76	1.73	0.691	0.694
4 (RF::VinaElem)	3	1	1.58	1.45	0.795	0.792
2 (MLR::Vina)	3	2	1.88	1.85	0.630	0.661
3 (RF::Vina)	3	2	1.72	1.68	0.711	0.714
4 (RF::VinaElem)	3	2	1.57	1.45	0.793	0.780
2 (MLR::Vina)	4	1	1.93	1.93	0.589	0.648
3 (RF::Vina)	4	1	1.81	1.80	0.656	0.669
4 (RF::VinaElem)	4	1	1.63	1.53	0.769	0.769
2 (MLR::Vina)	4	2	1.94	1.93	0.589	0.636
3 (RF::Vina)	4	2	1.79	1.75	0.682	0.686
4 (RF::VinaElem)	4	2	1.63	1.53	0.769	0.762
2 (MLR::Vina)	5	5	1.90	1.89	0.609	0.641
3 (RF::Vina)	5	5	1.74	1.70	0.700	0.699
4 (RF::VinaElem)	5	5	1.65	1.55	0.760	0.754
2 (MLR::Vina)	6	6	1.86	1.83	0.640	0.670
3 (RF::Vina)	6	6	1.73	1.69	0.707	0.707
4 (RF::VinaElem)	6	6	1.60	1.49	0.780	0.769

Table 4 Performance of models 2, 3, 4 trained in schemes 3, 4, 5, 6 and tested in schemes 1, 2, 5, 6 on the PDBbind v2013 blind benchmark

Model	Training	Test	RMSE	SD	Rp	Rs
2 (MLR::Vina)	3	1	1.70	1.69	0.521	0.511
3 (RF::Vina)	3	1	1.60	1.58	0.602	0.575
4 (RF::VinaElem)	3	1	1.48	1.48	0.666	0.643
2 (MLR::Vina)	3	2	1.69	1.69	0.523	0.509
3 (RF::Vina)	3	2	1.59	1.58	0.601	0.562
4 (RF::VinaElem)	3	2	1.49	1.49	0.655	0.635
2 (MLR::Vina)	4	1	1.88	1.80	0.413	0.415
3 (RF::Vina)	4	1	1.72	1.71	0.499	0.477
4 (RF::VinaElem)	4	1	1.57	1.57	0.610	0.589
2 (MLR::Vina)	4	2	1.77	1.75	0.468	0.447
3 (RF::Vina)	4	2	1.70	1.66	0.544	0.508
4 (RF::VinaElem)	4	2	1.58	1.57	0.611	0.582
2 (MLR::Vina)	5	5	1.65	1.65	0.550	0.526
3 (RF::Vina)	5	5	1.58	1.58	0.603	0.578
4 (RF::VinaElem)	5	5	1.49	1.50	0.653	0.633
2 (MLR::Vina)	6	6	1.68	1.68	0.526	0.514
3 (RF::Vina)	6	6	1.57	1.57	0.608	0.581
4 (RF::VinaElem)	6	6	1.47	1.48	0.665	0.643

of RF-Score-v4 on the most comprehensive set of high-quality complexes (the 3441 complexes from the PDBbind v2014 refined set) and implemented it as easy-to-use software that directly re-scores Vina-generated poses. See the abstract for availability and the README file therein for operating instructions.

Last but not the least, although we only used RF in this study as a proof of concept, we believe our conclusions can be applicable to other machine-learning scoring functions, which could possibly achieve even better results on this problem.

Additional files

Additional file 1: Correlation plots of measured and predicted binding affinities by the four models trained on crystal poses and tested on crystal poses of the PDBbind v2013 blind benchmark. (PDF 15 kb)

Additional file 2: Correlation plots of measured and predicted binding affinities by the four models trained on crystal poses and tested on docked poses of the PDBbind v2013 blind benchmark. (PDF 15 kb)

Additional file 3: Correlation plots of measured and predicted binding affinities by the four models trained on docked poses and tested on docked poses of the PDBbind v2013 blind benchmark. (PDf 15 kb)

Additional file 4: Correlation plots of measured and predicted binding affinities by the four models trained on docked poses and tested on crystal poses of the PDBbind v2013 blind benchmark. (PDF 15 kb)

Declaration

This work has been carried out thanks to the support of the A*MIDEX grant (n° ANR-11-IDEX-0001-02) funded by the French Government ≪Investissements d'Avenir'≫ program, the Direct Grant from the Chinese University of Hong Kong and the GRF Grant (Project Reference 414413) from the Research Grants Council of Hong Kong SAR. Publication of this article was funded by the Direct Grant from the Chinese University of Hong Kong and the GRF Grant (Project Reference 414413) from the Research Grants Council of Hong Kong SAR.
This article has been published as part of BMC Bioinformatics Volume 17 Supplement 11, 2016. Selected articles from the 11th International Meeting on Computational Intelligence Methods for Bioinformatics and Biostatistics (CIBB 2014). The full contents of the supplement are available online https://bmcbioinformatics.biomedcentral.com/articles/supplements/volume-17-supplement-11.

Availability of data and materials

This study did not generate data. Materials to reproduce the results and apply the presented machine-learning scoring functions to the structures of other protein-ligand complexes are available at http://istar.cse.cuhk.edu.hk/rf-score-4.tgz and http://ballester.marseille.inserm.fr/rf-score-4.tgz.

Authors' contributions

PJB designed the study. HL wrote the manuscript with PJB HL implemented the software and ran all the numerical experiments. All authors discussed results and commented on the manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Consent for publication

Not applicable.

Ethics approval and consent to participate

Not applicable.

Author details
[1]Department of Computer Science and Engineering, Chinese University of Hong Kong, Hong Kong, China. [2]Cancer Research Center of Marseille, INSERM U1068, F-13009 Marseille, France. [3]Institut Paoli-Calmettes, F-13009 Marseille, France. [4]Aix-Marseille Université, F-13284 Marseille, France. [5]CNRS UMR7258, F-13009 Marseille, France.

Published: 22 September 2016

References

1. Trott O, Olson AJ. AutoDock Vina: Improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. J Comput Chem. 2010;31(2):455–61.
2. Li H, Leung KS, Ballester PJ, Wong MH. istar: A Web Platform for Large-Scale Protein-Ligand Docking. PLoS ONE. 2014;9(1):85678.
3. Ain QU, Aleksandrova A, Roessler FD, Ballester PJ. Machine-learning scoring functions to improve structure-based binding affinity prediction and virtual screening. WIREs Comput Mol Sci. 2015;5(6):405–24.
4. Ballester PJ, Mitchell JBO. A machine learning approach to predicting protein-ligand binding affinity with applications to molecular docking. Bioinformatics. 2010;26(9):1169–75.
5. Ballester PJ, Schreyer A, Blundell TL. Does a More Precise Chemical Description of Protein–Ligand Complexes Lead to More Accurate Prediction of Binding Affinity? J Chem Inf Model. 2014;54(3):944–55.
6. Li H, Leung KS, Wong MH, Ballester PJ. Improving AutoDock Vina Using Random Forest: The Growing Accuracy of Binding Affinity Prediction by the Effective Exploitation of Larger Data Sets. Mol Inf. 2015;34(2-3):115–26.
7. Li H, Leung KS, Wong MH, Ballester P. Substituting random forest for multiple linear regression improves binding affinity prediction of scoring functions: Cyscore as a case study. BMC Bioinforma. 2014;15(1):291.
8. Ballester PJ, Mangold M, Howard NI, Robinson RLM, Abell C, Blumberger J, Mitchell JBO. Hierarchical virtual screening for the discovery of new molecular scaffolds in antibacterial hit identification. J R Soc Interface. 2012;9(77):3196–207.
9. Li H, Leung KS, Wong MH, Ballester PJ. Low-Quality Structural and Interaction Data Improves Binding Affinity Prediction via Random Forest. Molecules. 2015;20(6):10947–62.
10. Wang JC, Lin JH, Chen CM, Perryman AL, Olson AJ. Robust Scoring Functions for Protein–Ligand Interactions with Quantum Chemical Charge Models. Journal of Chemical Information and Modeling. 2011;51(10):2528–37.
11. Breiman L. Random Forests. Mach Learn. 2001;45(1):5–32.
12. Ballester PJ. Machine Learning Scoring Functions Based on Random Forest and Support Vector Regression. In: Pattern Recognition in Bioinformatics. Lecture Notes in Computer Science, vol. 7632. Berlin: Springer; 2012. p. 14–25.
13. Breiman L, Friedman J, Stone CJ, Olshen RA. Classification and Regression Trees. London: Chapman & Hall; 1984.
14. Svetnik V, Liaw A, Tong C, Culberson JC, Sheridan RP, Feuston BP. Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling. J Chem Inf Comput Sci. 2003;43(6):1947–58.
15. Cheng T, Li X, Li Y, Liu Z, Wang R. Comparative Assessment of Scoring Functions on a Diverse Test Set. J Chem Inf Model. 2009;49(4):1079–93.
16. Zilian D, Sotriffer CA. SFCscoreRF: A Random Forest-Based Scoring Function for Improved Affinity Prediction of Protein–Ligand Complexes. J Chem Inf Model. 2013;53(8):1923–33.
17. Li H, Leung KS, Nakane T, Wong MH. iview: an interactive WebGL visualizer for protein-ligand complex. BMC Bioinformatics. 2014;15(1):56.