A machine learning approach to predicting protein-ligand binding affinity with applications to molecular docking

Bentley Wingert

September 28, 2015

Pedro J. Ballester[1,*,†] and John B. O. Mitchell[2,*]

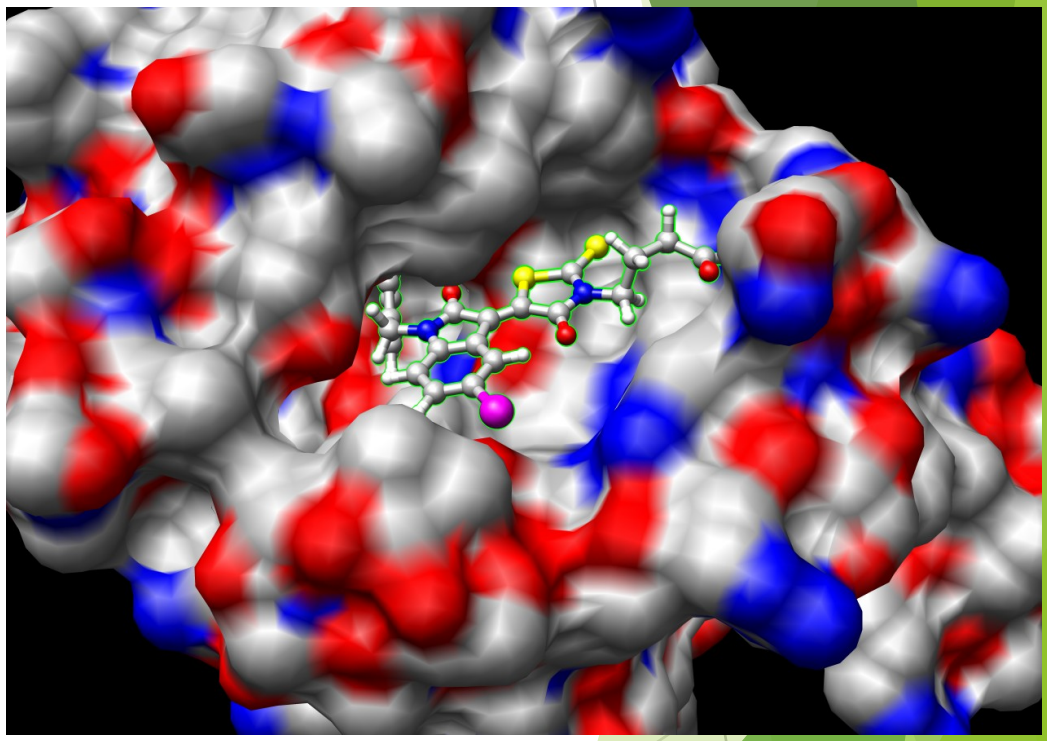
[1]Unilever Centre for Molecular Science Informatics, Department of Chemistry, University of Cambridge, Lensfield Road, Cambridge CB2 1EW and [2]Centre for Biomolecular Sciences, University of St Andrews, North Haugh, St Andrews KY16 9ST, UK

Associate Editor: Burkhard Rost

Contents

Background

- Molecular Docking
  - Model binding between small molecules and proteins at atomic scale
  - Important for drug discovery projects
  - Computationally screen libraries of small molecules against known protein structure
  - Two stages:
    - Pose identification
    - Scoring



"Docking" by Chaos - Own work. Licensed under CC BY-SA 3.0 via Commons - https://commons.wikimedia.org/wiki/File:Docking.jpg#/media/File:Docking.jpg

Scoring Functions

- Force field
  - Calculate potential energy of system as sum of energy terms from interactions between bonded and non-bonded atoms
  - Empirical parameter estimations
- Knowledge-based
  - 3D coordinates of complex being tested is compared against library of known coordinates
  - Scored based on how similar it is to known complexes
- Empirical
  - Counts number of certain types of interactions between protein and ligand
  - Uses scaled factors such as H-bonding, hydrophobic interactions, etc.

Random Forests

- Uses decision trees grown from bootstrap samples for classification or regression
  - Picks set of N complexes with replacement
  - At each node picks best split from random amount of features
- Grows many trees and votes or averages among them
- Measure importance of features by evaluating change in error of randomly permuted out-of-box samples

Goals

- Use machine learning to develop a scoring function that isn't based on predetermined forms
  - Allows for more accurate scoring of complexes that don't fit modeling assumptions
  - Use resampling techniques to prevent over fitting to training dataset
  - Estimate feature importance

Methods

- Pre-processing
  - Feature defined as total number of times a pair of atoms from protein and ligand from following sets occurs

$$\{P(j)\}_{j=1}^{9} = \{\mathrm{C,N,O,F,P,S,Cl,Br,I}\} \quad \{L(i)\}_{i=1}^{9} = \{\mathrm{C,N,O,F,P,S,Cl,Br,I}\}$$

  - Pair is counted if atoms are within 12 Å
  - Leads to vector of 36 features
  - Dataset generated as $D = \left\{\left(y^{(n)}, \vec{x}^{(n)}\right)\right\}_{n=1}^{N} \quad y \equiv -\log_{10} K$
  - Trained using 1105 complexes and tested against 195 randomly picked from set.

Methods

- Scoring function
  - Use CART algorithm (Breiman *et. al.*, 1984) to grow each tree
  - Score defined as average of all trees for given complex
- Resampling
  - Used to quickly validate data by testing on out-of-box samples
  - Mean square error (MSE)
  - m_{try} with min MSE value used for RF score.

$$\mathrm{RF}(\vec{x}^{(n)}; m_{\mathrm{try}}) \equiv \frac{1}{P}\sum_{p=1}^{P} T_p(\vec{x}^{(n)}; m_{\mathrm{try}}) \quad T_p : \aleph^{36} \rightarrow \Re^{+} \forall p$$

$$\mathrm{MSE}^{\mathrm{OOB}}(m_{\mathrm{try}}) = \frac{1}{\sum_{p=1}^{P} \left| I_p^{\mathrm{OOB}} \right|} \sum_{p=1}^{P} \sum_{n \in I_p^{\mathrm{OOB}}} \left(y^{(n)} - T_p(\vec{x}^{(n)}; m_{\mathrm{try}}) \right)^2$$

Results

- Scoring functions compared by their Pearson's correlation coefficient (R), Spearman's correlation coefficient (R_s), standard deviation (SD), and root mean square error (RMSE).
- Reproduces training set well
  - R=0.953, RMSE=0.74
- Prediction on out-of-box samples also performs well
  - R=0.699, RMSE=1.52
- Similar performance on 195 complex test data
  - R=0.776, RMSE=1.58

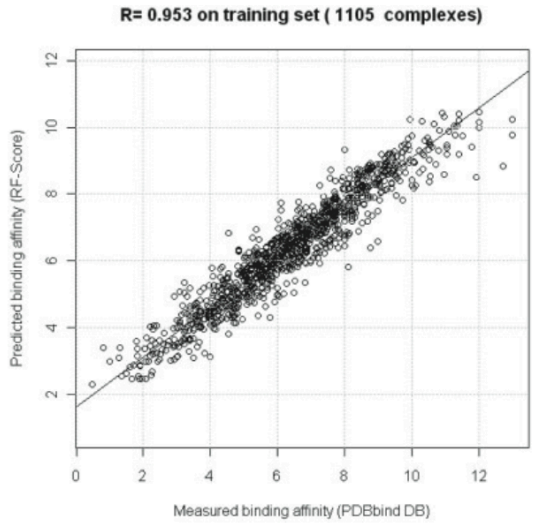


Fig. 1. RF-Score reproduces its training data with very high accuracy (Pearson's correlation coefficient $R = 0.953$ and RMSE = 0.74).

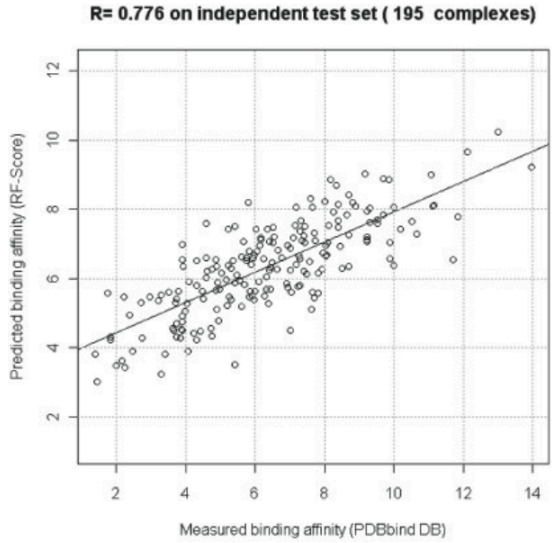


Fig. 3. RF-Score predicts the test data with high accuracy (Pearson's correlation coefficient $R = 0.776$ and RMSE = 1.58).

Results

- RF-Score performance increases with size of training data

Table 1. Dependence of RF-Score on size of training set (N_{train})

N_{train}	R	R_s	RMSE	m_{best}	$RMSE^{OOB}$	ΔRMSE
1105	0.776	0.762	1.58	5	1.52	0.06
900	0.750	0.740	1.63	9	1.51	0.12
700	0.734	0.735	1.69	4	1.52	0.17
500	0.685	0.684	1.77	6	1.44	0.33
300	0.609	0.628	1.90	10	1.46	0.44
100	0.562	0.572	2.01	7	1.56	0.45

Results

- Feature importance
  - Hydrophobic interactions
  - Polar/Non-polar contacts
  - Hydrogen bonds

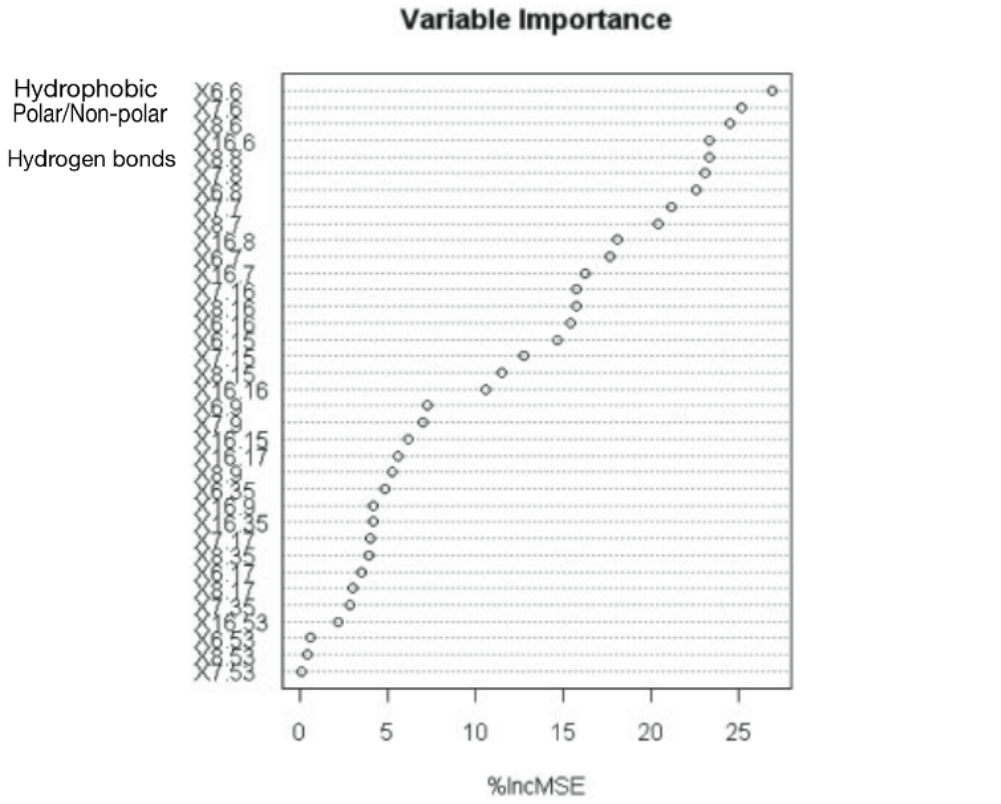


Fig. 2. Estimation of feature importance based on internal validation data. Overall, it shows the importance of each type of protein–ligand contact across training complexes, which are by construction representative of the entire PDB.

Results

- Results of testing against PDBbind benchmark compared to those of other commercial and academic scoring functions
- Significantly improved correlation coefficients and standard deviation over state of the art

Table 2. Performance of scoring functions on the PDBbind benchmark

Scoring function	R	R_s	SD
RF-Score	0.776	0.762	1.58
X-Score::HMScore	0.644	0.705	1.83
DrugScore^{CSD}	0.569	0.627	1.96
SYBYL::ChemScore	0.555	0.585	1.98
DS::PLP1	0.545	0.588	2.00
GOLD::ASP	0.534	0.577	2.02
SYBYL::G-Score	0.492	0.536	2.08
DS::LUDI3	0.487	0.478	2.09
DS::LigScore2	0.464	0.507	2.12
GlideScore-XP	0.457	0.435	2.14
DS::PMF	0.445	0.448	2.14
GOLD::ChemScore	0.441	0.452	2.15
SYBYL::D-Score	0.392	0.447	2.19
DS::Jain	0.316	0.346	2.24
GOLD::GoldScore	0.295	0.322	2.29
SYBYL::PMF-Score	0.268	0.273	2.29
SYBYL::F-Score	0.216	0.243	2.35

Conclusions

- Authors were able to successfully generate scoring function through non-parametric machine learning that improves on state-of-the-art.
- Performance shown to increase with amount of training data, so method should continue to improve with increasing in docking data available.