RSC Advances

PAPER

View Article Online
View Journal | View Issue

Check for updates

Cite this: *RSC Adv.*, 2018, **8**, 12127

A machine learning approach towards the prediction of protein–ligand binding affinity based on fundamental molecular properties†

Indra Kundu,[a] Goutam Paul *[b] and Raja Banerjee *[c]

There is an exigency of transformation of the enormous amount of biological data available in various forms into some significant knowledge. We have tried to implement Machine Learning (ML) algorithm models on the protein–ligand binding affinity data already available to predict the binding affinity of the unknown. ML methods are appreciably faster and cheaper as compared to traditional experimental methods or computational scoring approaches. The prerequisites of this prediction are sufficient and unbiased features of training data and a prediction model which can fit the data well. In our study, we have applied Random forest and Gaussian process regression algorithms from the Weka package on protein–ligand binding affinity, which encompasses protein and ligand binding information from PdbBind database. The models are trained on the basis of selective fundamental information of both proteins and ligand, which can be effortlessly fetched from online databases or can be calculated with the availability of structure. The assessment of the models was made on the basis of correlation coefficient (R^2) and root mean square error (RMSE). The Random forest model gave R^2 and RMSE of 0.76 and 1.31 respectively. We have also used our features and prediction models on the dataset used by others and found that our model with our features outperformed the existing ones.

Received 1st January 2018
Accepted 13th March 2018

DOI: 10.1039/c8ra00003d

rsc.li/rsc-advances

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

Introduction

The cardinal goal of drug discovery is to design and deliver selective compounds against individual biological targets. In general, it takes about 15 years and up to 800 million dollars to convert a promising new compound into a drug.[1] The approaches and methodologies used in drug design have been changed over time. In an early stage of the drug discovery process, the focus is on reducing the number of drug candidates and this problem has been deciphered using computational approaches.[2] A drug is a small molecule which activates or inhibits the function of protein, as proteins are one of the popular targets for the drug designing process.[3] The interaction between a protein and ligand is specific. These specific molecular interactions between proteins and its ligand plays crucial role to a broad spectrum of biological functions.[4–6] Predicting interactions between ligand and proteins is an indispensable element in the drug discovery process.[3,7]

In order to perform a rapid search for molecules that may bind to targets of biological interest computational techniques such as structure based drug designing (SBDD) is carried out, which includes structure based virtual screening (SBVS) or molecular docking followed by Molecular Dynamics.[3,8,9] Molecular docking is one of the most frequently used methods because of its ability to predict the conformation and affinity of ligand binding to the target site, with a substantial degree of accuracy.[10,11] Docking methods effectively search high-dimensional spaces for plausible interaction and use a scoring function that correctly ranks the candidate.[12] Although the results of docking are specific and reliable; however, screening of umpteen molecules maneuvering every step of docking can be wearisome. As docking is a time consuming process, it could be worth it if some faster methods can be employed to predict whether a molecule can bind the biologically active target molecule to initiate the biological function. Towards the end, Machine Learning (ML)[13] techniques can be an alternative choice.

Machine learning algorithms build a model from training inputs in order to make data-driven predictions or decisions, expressed as outputs.[13,14] These methods will statistically analyze the correlation between chemical structures and interaction status of known protein and ligand pairs to derive

[a]Department of Bioinformatics, Maulana Abul Kalam Azad University of Technology (formerly known as West Bengal University of Technology), Kolkata, India. E-mail: indraknd@gmail.com

[b]Indian Statistical Institute, Kolkata, India. E-mail: goutam.k.paul@gmail.com

[c]Maulana Abul Kalam Azad University of Technology (formerly known as West Bengal University of Technology), Kolkata, India. E-mail: banraja10@gmail.com

† Electronic supplementary information (ESI) available: S1 contains the list of proteins used in our experiments. S2 contains the lists of features of proteins and ligands used for regression. S3 contains figures representing error count for prediction on datasets. GitHub access: https://github.com/kundu-i/protein-ligand-binding-affinity-prediction. See DOI: 10.1039/c8ra00003d

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

statistical models for predicting the status of other unknown compounds.[15] It does not demand the explicit program for the learning procedure of the machine. Supervised prediction can be based on either classification or regression.[16] Classification is used when the discrete value is to be predicted, whereas regression is used where the values are diverse and cannot be predicted exactly hence, accuracy is measured on the basis of closeness of predicted value to true value. Prediction of binding energy value requires regression algorithm and predicting the feasibility of interaction can be fulfilled by classification. Other than this, statistical learning method has recently been used for classification of G-protein coupled receptors and DNA-binding proteins. It has also been employed in a number of other protein structure, interaction prediction studies including fold recognition,[17] protein–protein interaction prediction,[18,19] solvent accessibility[20] and structure prediction.[21]

However, studies combining the spheres of protein–ligand interactions and machine learning conducted till date were mostly focused on a particular protein or a particular class of proteins. Laurent Jacob *et al.* carried out a study involving targets with no or few known ligands and succeeded in predicting enzymes and GPCR with an accuracy of 86.2% and 77.6% respectively.[22] Masayuki Yarimizu *et al.* accomplished a study using Support Vector Machine (SVM) on tyrosine receptor and predicted whether a molecule is a ligand of the tyrosine receptor or not with a very high accuracy, AUC was 0.996.[23] The study is focused on tyrosine kinases. Laurent Jacob *et al.*[22] and Masayuki Yarimizu *et al.*[23] used ML for classification, where there is binary class *i.e.* yes or no. For prediction of discrete value ML regression is employed.[13,14,16] Xue *et al.*[24] utilised regression for prediction of binding energy in terms of log K using SVM models, their study was focused on drugs against a single protein, human serum albumin.

In order to utilize the application of ML in much broader aspect beyond a particular protein or particular class of proteins, in this paper, we have addressed this issue over a heterogeneous class of proteins with variety of ligands and trained the machine so as to predict the preferable interaction through calculation of binding energy. To the best of our knowledge, such an effort would be reported for the first time. We have used Weka 3.6.8,[16,25,26] which is a popular data mining tool that provides various machine learning algorithms.

Receptors which were diverse in their molecular function, biological process, and cellular component were considered. Deng *et al.*[27] used a diverse dataset of 105 protein–ligand complexes, Kramer and Gedeck[28] used pdbbind version 2009, whereas Wang *et al.*[29] did a wide range study on pdbbind benchmark version 2007 and 2012. We have used pdbbind dataset version 2015 for our study in which we obtained correlation coefficient (R^2) and root mean square error (RMSE) 0.76 and 1.31 respectively. We have also tested our model and features on the protein–ligand list used by Wang *et al.*,[29] Deng *et al.*,[27] Xue *et al.*[24] and Kramer and Gadeck[28] and have successfully recorded a better correlation coefficient of 0.75, 0.75, 0.86, and 0.72 than the reported 0.67, 0.64, 0.63 and 0.69 respectively.

Materials and methods

Dataset

All the protein–ligand binding affinity data were acquired from the PdbBind Database.[30,31] The database (v2015) comprises of binding energy for all types of biomolecular complexes available in RCSB.[32] Hence, it bridges structural information with the binding affinity of complexes. We have focused only on the protein–ligand complexes, excluding protein–nucleic acid and protein–protein complexes. There were 11 987 instances of protein and ligand. Binding affinity data was available in terms of K_i (inhibition constant), K_d (dissociation constant), IC_{50}, EC_{50}. We have taken into account the instances having activity in terms of either K_i or K_d, rejecting those which have the activity in terms of assay dependent IC_{50} or EC_{50}. The dataset was further refined by excluding the ligands which had incomplete structure. Proteins which bind to isomers, peptide like compounds, and more than one compound or have metal ions as cofactors were also eliminated. Our final dataset consists of 2864 instances, which comprises proteins of diverse class listed in ESI S1.† There, we have given the classification of the proteins used by us on the basis of their functions. They are of diverse types and the data is non-redundant in the sense that no two rows are exactly the same – either the protein or the ligand is different. Training and testing set are assigned randomly but we have also reported blind set validation.

Our primary goal was to develop a tool which can be used in the very initial stage of drug discovery process for predicting potential candidates. As diverse drugs act on a diverse set of enzymes, therefore we focused on training machine on heterogenous class of protein which are both functionally and structurally diverse. We have used pdbbind version 2015 for our study. Wang *et al.*[29] did a similar study using 2012 version but they did not eliminate isomers or incomplete ligand structure, whereas we have included even the low-resolution structure. Despite the dataset being created from the same source our dataset varies a lot listed in ESI S1.† We have used their exact training and test dataset for one generic and 3 family specific dataset namely HIV protease, trypsin, carbonic anhydrase with our features. Other than this we have also compared our features on the dataset used by Deng *et al.*[27] and Xue *et al.*[24]

Our dataset has 2864 rows and 128 columns. The rows represent the protein–ligand pair whereas the columns are their properties. Each row of the Dataset can be represented as $\boldsymbol{X}1_1$, $\boldsymbol{X}2_1$, …, $\boldsymbol{Y}1$, where $\boldsymbol{x}$ are the features and $\boldsymbol{y}$ is the class that will be predicted by our models. Our Dataset can be represented as follows

$$\begin{bmatrix} X1_1 & X2_1 & \cdots & X127_1 & Y_1 \\ X1_2 & X2_2 & \cdots & X127_2 & Y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ X1_{2864} & X2_{2864} & \cdots & X127_{2864} & Y_{2864} \end{bmatrix}$$

Machine learning algorithms will find the pattern which will fit x and create a function $f(x)$ that can predict y for a new x.

This journal is © The Royal Society of Chemistry 2018

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

Features extraction

Training a machine is highly dependent on features. Feature selection must be done with utmost care. Drug binding is an extremely selective process; it depends on the shape, size, constitutional makeup, and physicochemical properties of both drug and its target.[33] We have calculated total 127 features and all the features are very common, so they can be effortlessly calculated for any new or unexplored protein or ligand listed in ESI S2.†

For proteins

Global features refer to the features considering the entire protein. We have used the entire protein instead of features of only pockets and cavities. Our aim is to train a machine with the features that are easily calculable using merely the receptor. Calculating features of the cavity demands the information of the cavity. For the cavity information, we either need to have a co-crystallized structure of a protein with its ligand or we can try *in silico* methods. Both of the methods are well known but are time-consuming, here we are presenting a method in which can skip a few steps and reach the binding energy prediction relatively faster.

Amino acid sequence. The unique amino acid sequence of one protein is often referred to as its primary structure. Protein primary sequence is guided and specified by nucleotides present in the gene. Each amino acid is encoded by particular triplet set of codons. Native conformation of a protein is determined by the interatomic interactions along with the amino acid sequence, in a given environment.[34] Chemical reactivity of an individual protein is defined by the type and spatial orientation of surface accessible amino acid side chains.[35] Conformation, therefore, determines protein function, especially its interaction with ligand. Consequently, knowledge of primary sequence might play a crucial role to predict conformation as well as its interactive properties.

Protein secondary structure. Proteins secondary structure, stabilised through the local interactions among the adjacent residues are giving rise to a particular geometry by repetitive approach. Instead of considering coordinates of each atom present in the system, in order to minimise the computation time, we have selected a few special characteristics functional features (*e.g.* molecular weight, number of chains, number of ss bridges, and number of various types of secondary structure like helices, sheets; as mentioned in Table 1 of the protein of interest to build a prediction model towards feasibility of protein–ligand interaction. These features were calculated using programme DSSP.[36]

Accessible surface area. Solvent plays a crucial role in the interactions of proteins with their ligands. Solvent-accessible surface area (SASA) is the area of the protein that is directly in contact with solvent.[37] Interaction of protein with ligand generally involves an entropically favored displacement of solvent molecules from the protein and ligand surfaces and an enthalpically favored reorganization between the protein and ligand along with the solvent molecule.[38] SASA of the receptor is also calculated from DSSP programme.[36]

Table 1 Features details and their source

Molecule	Features	Source
Protein	Amino acid percentage	Calculated from Fasta files[31]
	Accessible surface of protein	DSSP[36]
	Number of hydrogen bonds in antiparallel bridges and parallel bridges	
	Number of hydrogen bonds of type O(I) → H–N(I-5), O(I) → H–N(I-4), O(I) → H–N(I-3), O(I) → H–N(I-2), O(I) → H–N(I-1), O(I) → H–N(I+0), O(I) → H–N(I+1), O(I) → H–N(I+2), O(I) → H–N(I+3), O(I) → H–N(I+4), O(I) → H–N(I+5)	
	Number of chains	
	Number of ss bridge	
	Number of residues	
Ligand	Atom count: C, N, O, H, S, P, Cl, F, Br, I	Padel descriptors[42]
	Bond count: number of single, double, triple bond including and excluding hydrogens	
	Ring count: number of 3, 4, 5, 6, 7, 8, 9 atom/carbon rings, aromatic rings, fused hetero rings, fused homo ring	
	Physicochemical properties: complexity, log *p*, hbond donor, hbond acceptor, topological surface area, mol. wt	Pubchem[41]

For ligands

A drug sweeps thorough blood vessel, gastrointestinal fluids, small intestine before reaching its active site. As Lipinski *et al.*[39,40] explained, a drug molecule must have the absorption, distribution, metabolism, excretion, and toxicity (ADMET) properties so as to qualify as a successful candidate. All these responses of a chemical compound are intrinsic and is a result of the combination of its various physical and chemical properties. Therefore, for defining a drug we have included all its physicochemical properties available in pubchem[41] along with that few structural properties, which were calculated using a tool Padel Descriptor.[42] We have included major 2-d properties of a small molecule along with physicochemical properties which define a molecule and differentiate it with others. List of features and their source is represented in Table 1 and details of the features are also listed in ESI S2.†

We haven't included intermolecular interaction features as we are going to use this prediction method in the very initial stage, prior to the formation of protein–ligand complex and considering the fact that binding intermolecular distance or

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

constitution can only be generated using the complex structure. The presence of the complex structure validates that either computational molecular docking or experiment is already done, hence no need to get the binding affinity using this prediction model.

Prediction models

Weka v3.8.0 (ref. 25 and 26) was used in our study. 2864 instances with 128 features were trained using Gaussian process,[43] linear regression,[44] multilayer perceptron,[45,46] SMO regression,[47,48] K-star,[49] and Random forest.[50] It is tested for 10-fold cross-validation. *i.e.* for each fold there are 286 instances for testing while rest 2578 are used for training. For the next fold another 286 instances are selected for testing and the rest used for training.

Random forest model

Random forest (RF), introduced by Breiman[50] is based on bagging *i.e.* bootstrap aggregation. It divides the entire dataset into subsets and builds a random tree for each subset which is called bootstrap sampling and runs prediction test on each sample tree, the final prediction result is the amalgamation of prediction of each Random Tree. In addition to bagging, RF splits the dataset on features. Each tree will be trained on a minimum of features that is K. The entire training dataset is N in number and there will be I number of random trees. For constructing a tree, a node is selected at random from the features set and growing the tree, each parent node is split into daughter nodes on the basis of best split considering information gain that is needed to be present in the data of the node.

$$\text{Information gain} = \text{entropy (parent node)} - [\text{average entropy (daughter node)}]$$

where $\text{entropy} = -\sum p_i \log_2 p_i$

Moreover, p is the probability of class.

Gaussian process model

In a multivariate data, any point in space is a vector $\vec{x}$ having components $X_1, X_2, \ldots X_n$. Gaussian Process (GP)[43] by definition is a collection of random variables of any finite number which have consistent joint Gaussian distribution which is fully specified by its mean function (μ) and covariance function (ref).

This can be represented as

$$\begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_{127} \end{bmatrix} \sim \mathrm{GP} \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_{127} \end{bmatrix}, \begin{bmatrix} C_{1,1} & C_{1,2} & \cdots & C_{1,127} \\ C_{2,1} & C_{2,2} & \cdots & C_{2,127} \\ \vdots & \vdots & \vdots & \vdots \\ C_{127,1} & C_{127,2} & \cdots & C_{127,127} \end{bmatrix}$$

where X_1, $X_2 \ldots$, X_n are the components of vector $\vec{x}$ which describe the features while μ_1, $\mu_2 \ldots$ μ_n are component of $\vec{\mu}$ which is the mean of corresponding feature. The matrix is the covariance matrix, each of its diagonal elements are variance of corresponding feature whereas the rest elements are their respective covariance. Covariance function characterizes correlations between different points in the space.

For making prediction utilizing Gaussian Process, we have used various kernels. Kernels basically calculate how 2 points $\overrightarrow{x_1}$, $\overrightarrow{x_2}$ in space are related which is termed as the covariance here. We have studied polykernel, normalised polykernel, and RBF kernel.

Polykernel. Polynomial kernel looks into the similarity of two input vectors on the basis of their dot product of the vectors. For a p degree of polynomial, it is defined as $k_{\mathrm{p}}(\overrightarrow{x_1}, \overrightarrow{x_2}) = ((\overrightarrow{x_1}, \overrightarrow{x_2}) + 1)^p$. In weka the parameter exponent controls the degree of polynomial. The default degree is set to 1, however we have toggled that to find a function which best fits our data.

Normalised polykernel. It is an extension of polykernel. This is defined as $k_{\mathrm{np}}(\overrightarrow{x_1}, \overrightarrow{x_2}) = \dfrac{k_{\mathrm{p}}(\overrightarrow{x_1}, \overrightarrow{x_2})}{\sqrt[2]{(\overrightarrow{x_1} \cdot \overrightarrow{x_1})(\overrightarrow{x_2} \cdot \overrightarrow{x_2})}}$, the parameter exponent is same as the polykernel.

RBF kernel. Radial basis function (RBF) kernel uses the squared Euclidean distance function between two feature vectors in space. This can be defined as $k_{\mathrm{R}}(\overrightarrow{x_1}, \overrightarrow{x_2}). = \mathrm{e}^{\left(-\gamma \|\overrightarrow{x_1} - \overrightarrow{x_2}\|^2\right)}$, where the parameter gamma is 0.01 by default.

Multilayer perceptron model

Multilayer perceptron (MLP)[45,46] model is an Artificial Neural Network model. Neural Network mimics biological neurons. Each element of input vector can be seen as single dendrite which have the information and it passes the information to the perceptron neuron. The perceptron forms a linear combination of inputs and their weights to computes an output and then the output calculation continues through an activation function. The classifier uses back propagation error to find the optimised weights. These perceptron makes up the hidden layer, there are more than one perceptron in a multilayer perceptron model to fit non-linearly separable data. Number of hidden layers can be defined in weka's MLP function.

SMO model

SMO model in weka 3.6.8 (ref. 25 and 26) is based upon sequential minimal optimisation (SMO) algorithm[47] for training a support vector classifier. Support vector machines (SVM) are learning algorithms that finds a hyperplane which separates the features of multiple classes of data. The points that are closest to the separator have nonzero weights and the rest have zero. The points with nonzero weights are called the support vectors because they hold up the separating plane. SVM uses kernel (same as Gaussian Process model), it implicitly maps the original data to a feature space of possibly infinite dimension in which data which is not separable in the original space becomes separable in the feature space. SMO algorithm quickly solves the SVM quadratic problem without any extra matrix storage and without invoking an iterative numerical routine for each sub-problem.

Model evaluation

Internal 10-fold cross-validation and a blind set validation was implemented for prediction of binding affinity of protein–ligand pair. Cross-validation allows each instance of the dataset to be

 This journal is © The Royal Society of Chemistry 2018

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

tested once for prediction, hence it is purely an unbiased basis for testing efficiency of a model. The performance of each model was evaluated using correlation coefficient (R^2) and Root Mean Square Error (RMSE).

$$\text{Error } (E) = \text{actual binding affinity } (A_{\text{A}}) - \text{predicted binding affinity } (A_{\text{P}})$$

Mean actual binding affinity,

$$\overline{A_{\text{A}}} = \frac{1}{N}\sum_{i=1}^{N} A_{\text{A}(i)}$$

Mean predicted binding affinity,

$$\overline{A_{P}} = \frac{1}{N}\sum_{i=1}^{N} A_{\text{P}(i)}$$

$$\text{Relative percentage error} = \frac{E}{A_{\text{A}}} \times 100,$$

Mean error,

$$\overline{E} = \frac{1}{N}\sum_{i=1}^{N} E_i,$$

$$\text{Mean Square Error (MSE)} = \frac{1}{N}\sum_{i=1}^{N} E_i^2,$$

$$\text{Root Mean Square Error (RMSE)} = \sqrt{\text{MSE}}$$

$$\text{Correlation coefficient}\left(R^2\right) = \frac{\sum_{i=1}^{N}\left(A_{\text{A}_i} - \overline{A_{\text{A}}}\right)\left(A_{\text{P}_i} - \overline{A_{\text{P}}}\right)}{\sqrt{\sum_{i=1}^{N}\left(A_{\text{A}_i} - \overline{A_{\text{A}}}\right)^2 \sum_{i=1}^{N}\left(A_{\text{P}_i} - \overline{A_{\text{P}}}\right)}}$$

Comparative study

In addition to testing the prediction models on our own dataset, we have drawn a comparison using our features with few of already published dataset. Xue *et al.*[24] used 94 drugs against human serum albumin and have used SVM models for prediction the binding affinity. Deng *et al.*[27] has used 105 diverse protein–ligand complexes and Wang *et al.*[29] used pdbbind version 2012. Wang *et al.*[29] had compared their model with various state of art prediction models with their Random forest model and concluded their model outperformed others. We have used the same list of proteins and ligands used by these authors and extracted our list of features for them. They have also tested their model on the basis R^2 and RMSE. We have chosen their best R^2 value for comparing with our result.

Results

Performance analysis

Machine learning algorithm incorporated in Weka 3.6.8 package[25,26] has been used. It has some default parameters; however, we have also tried to optimise parameters according to the need of our dataset, and hence we are able to predict our binding energy with better correlation and lesser error. Below we present a summary of the regression algorithms used.

Random forest

Random forest is an ensemble of various Decision Trees.[50] The number of trees to be generated can be defined by the user. Weka's default value of the number of tree generation is 100. We have changed the parameter and recorded the correlation coefficient for each model. We have observed an increase in the correlation coefficient with the increase in number of trees; however, it is not increasing much after 300, so we have used 400 trees for our subsequent determination of number of features to be used per tree.

When the number of features is set to 0 (default setting), then the actual number of features is calculated as $\log_2((127) + 1) = 7$. We observed that 400 iterations with 30 features in each iteration

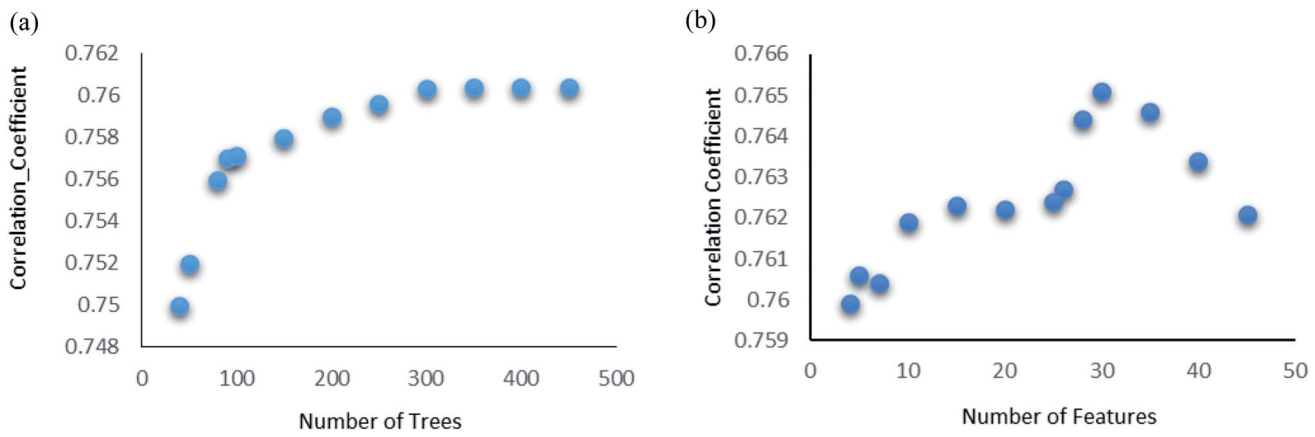

Fig. 1 (a) Random forest algorithm's performance analysis. Change in correlation coefficient with change in number of trees. (b) Random forest algorithm's performance analysis. Change in correlation coefficient with change in number of features with number of iterations fixed at 400.

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

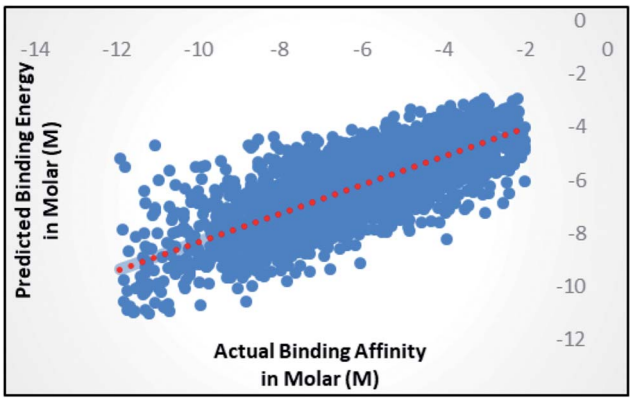

Fig. 2 Scatter plot for actual *vs.* predicted binding affinity of v2015 dataset using Random forest with 400 iterations having 30 features in each.

gave the best result, shown in Fig. 1a and b, with $R^2 = 0.7651$ and error 1.31 molar (M). Variance was fixed at default 0.001 and the number of instances per leaf was set to 1 to reduce the probabilistic conditions. The correlation plot of actual *vs.* predicted binding affinity using the above-mentioned parameters is shown in Fig. 2.

Gaussian process

Gaussian process[43] regression models utilizing normalised polykernel and RBF (Radial Basis Function) kernels were used in our study. In normalised polykernel, the degree of polynomial is assigned by a parameter named exponent. Weka 3.6.8 (ref. 25 and 26) package have 2.0 as default exponent, using that we got correlation coefficient 0.6505 and RMSE 1.53, we kept increasing stepwise and finally observed 20-degree polynomial gave the best results, correlation coefficient 0.7386 and RMSE 1.36 (Fig. 3a). In RBF kernel the parameter γ was set to 0.01 which gave correlation coefficient 0.5626 and RMSE 1.57. We further adjusted the parameter to get better results. When γ is set to 2, it gave the best results with correlation coefficient 0.7327 and RMSE 1.38 (Fig. 3b).

Other models

Other than Random forest and Gaussian Process, we also used multilayer perceptron, SMO, and K-star prediction models. multilayer perceptron and Linear regression did not suit our data well. We observed that the maximum number of instances have relative error less than 20%. Out of 2864, the number of instances that gave relative error less than 20% is 1938, 1932, 1896 and 1525 respectively in Random forest,[50] SMO,[47] Gaussian process[43] and multilayer perceptron[45,46] (Fig. 4). For less than 250 instances, we are getting more than 50% relative error. SMO and Random forest models are equally good with our data, however a slight difference is that 2 instances had 200% relative error in Random forest model and the number is 8 for SMO and 9 for Gaussian Process. This makes Random forest model better than the rest. We can see that multilayer perceptron is not performing well, 361 instances had relative error more than 50%. Out of 2864 instances, 2570 instances had error less than 0.2 log units of actual energy, which implies a significant prediction result.

Comparative results

We have fine-tuned the algorithms to find out the best performance. We have observed that Random forest[50] is outperforming all other algorithms. Wang *et al.*,[29] Deng *et al.*[27] and Xue *et al.*[24] studied prediction of protein–ligand binding affinity. We have used the instances from their datasets and calculated the features that we used in our study.

Xue *et al.*[24] had used 94 drugs against human serum albumin, we have also used human serum albumin against 91 drugs as 3 drugs were obsolete at present and we could not find their information. Xue *et al.*[24] used Support Vector Machine (SVM) regression using RBF kernel for their study, so for this dataset we have also used SMO models, which is the SVM regression algorithm incorporated in Weka 3.6.8.[25,26] We got a better correlation coefficient 0.86 as compared to their reported correlation coefficient 0.63 for 10-fold cross-validation (Fig. 5a). They have trained the machine for drugs against single protein human serum albumin, we can conclude that ML

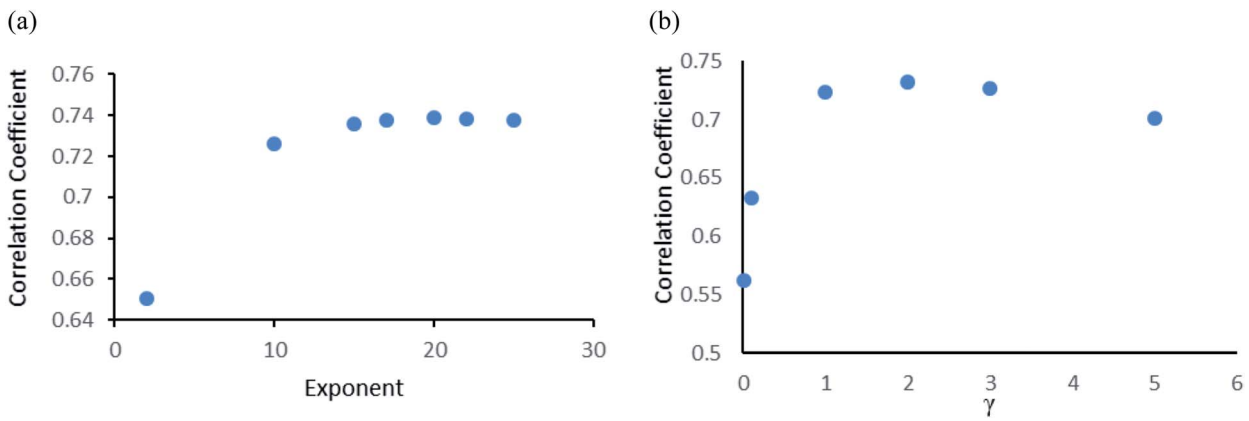

Fig. 3 (a) Gaussian process algorithm's performance analysis. Change in the correlation coefficient with change in exponent value of normalised polykernel, (b) change in the correlation Coefficient with change in γ value of RBF kernel.

This journal is © The Royal Society of Chemistry 2018

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

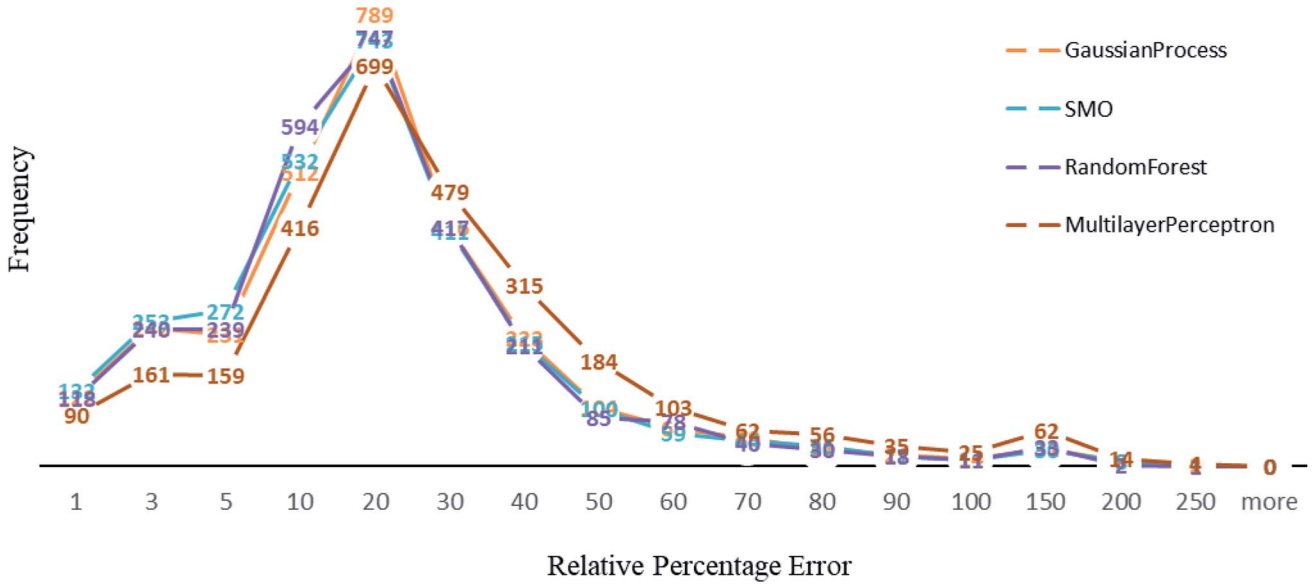

Fig. 4 A comparison of percentage of relative error among algorithms used.

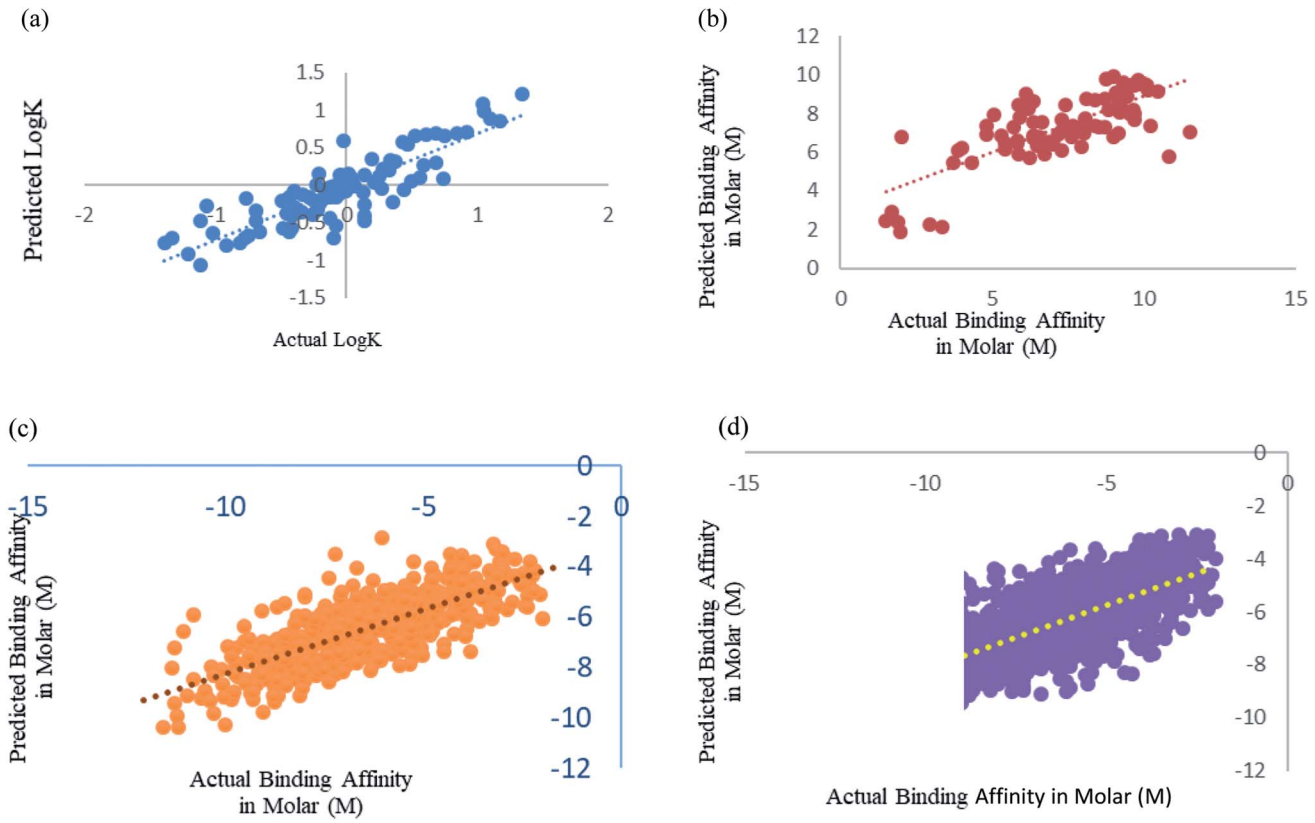

Fig. 5 (a) Scatter plot for actual *vs.* predicted log *K* of Xue's dataset using SMO utilising RBF kernel over 10-fold cross-validation. (b) Scatter plot for actual *vs.* predicted binding affinity of Deng's dataset using Random forest over 10-fold cross validation. (c) Scatter plot for actual *vs.* predicted binding affinity of Wang's dataset using Random forest. (d) Scatter plot for actual *vs.* predicted binding affinity of Kramer's dataset using Random forest.

models can predict protein–ligand interaction with one protein and many ligand with very minimal error, RMSE 0.114.

Deng *et al.*[27] were one of the initiators of carrying out prediction study over diverse class of proteins. They had used 105 diverse protein–ligand complex and we also have used the same complexes with our calculated features. We have used Random forest prediction models[50] for prediction test. We got a better correlation coefficient 0.75 as compared to their

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

Table 2 Comparison table

Authors	Training instances	Test/method	Their result	Our result
C. X. Xue, *J. Chem. Inf. Comput. Sci.*, 2004 (ref. 24)	Human serum albumin; 95 drugs	Training set Supplied test set Cross validation	$R^2 = 0.94$: RMSE = 0.134 $R^2 = 0.89$: RMSE = 0.222 $R^2 = 0.63$	**$R^2 = 0.1$; RMSE = 0.0059** **$R^2 = 0.987$: RMSE = 0.114** **$R^2 = 0.867$**
Wei Deng, *J. Chem. Inf. Comput. Sci.*, 2004 (ref. 27)	105 (diverse) complexes	Cross validation	$R^2 = 0.64$	**$R^2 = 0.756$**
Christian Kramer and Peter Gedeck, *J. Chem. Inf. Model.*, 2011 (ref. 28)	Pdbbind v2009; 1387 complexes	Cross validation	$R^2 = 0.69$	**$R^2 = 0.72$**
Yu Wang, *J. Comput.–Aided Mol. Des.*, 2014 (ref. 29)	Hiv protease 136	Supplied test set 34	$R^2 = 0.728$: RMSE = 1.05	RF: $R^2 = 0.69$: RMSE = 1.07 **DT: $R^2 = 0.74$: RMSE = 1.08**
	Trypsin 88	Supplied test set 22	$R^2 = 0.871$: RMSE = 0.61	RF: $R^2 = 0.85$: RMSE = 0.69 **Kstar: $R^2 = 0.873$: RMSE = 0.65**
	Carbonic anhydrase 100 V2012 2318	Supplied test set 26 Supplied test set 579	$R^2 = 0.790$: RMSE = 0.92 $R^2 = 0.678$: RMSE = 1.46	**$R^2 = 0.8078$: RMSE = 0.8867** **$R^2 = 0.7564$: RMSE = 1.299**

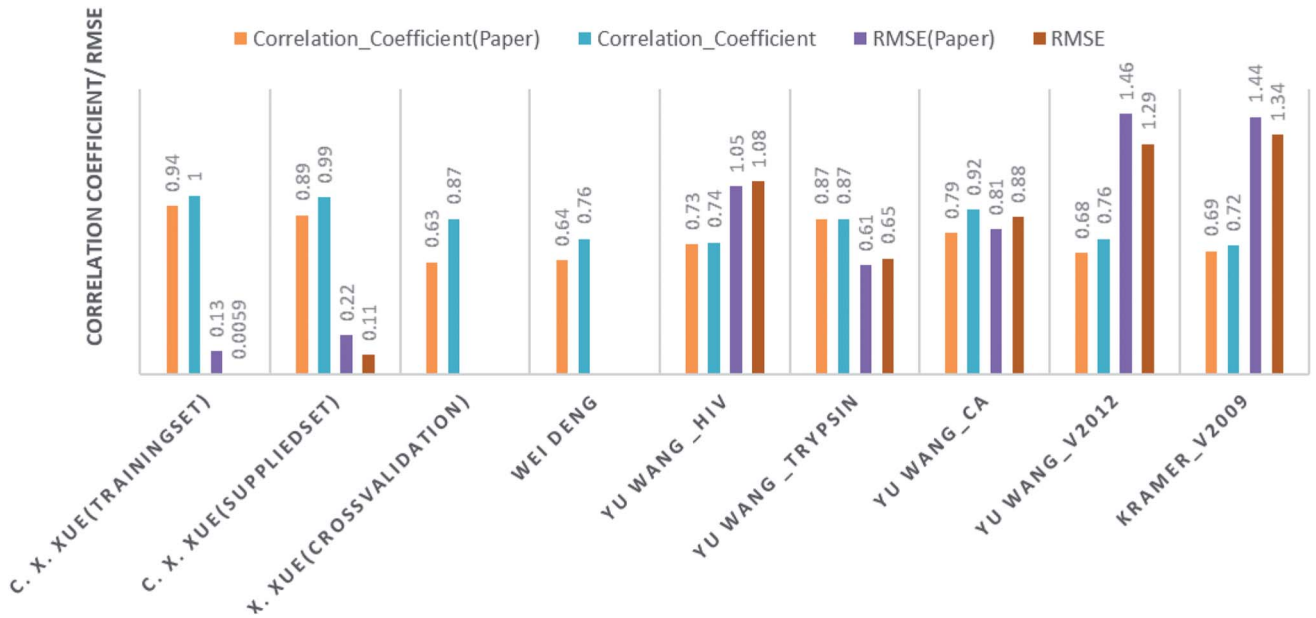

Fig. 6 Bar graph for the correlation coefficient and RMSE (both rounded off to two decimal places) of the prediction models. A comparison of results using our prediction model and the results published by the respective authors.

reported correlation coefficient 0.64 for 10-fold cross-validation (Fig. 5b).

Wang *et al.*[29] used the PdbBind Database[30] of the year 2012 for their study and had reported an appreciable prediction results. We have used the exact same protein ligand pairs in test and training set as used by them. We have used Random forest Model[50] for the dataset, as Wang *et al.*[29] concluded that the Random forest models outperformed others. We got a better correlation coefficient 0.75 and RMSE 1.29 as compared to their reported correlation coefficient 0.67 and RMSE 1.46 (Fig. 5c). Other than the generic dataset used by Wang we have also compared family specific dataset used by them.

Kramer and Gadeck[28] had also used the PdbBind Database[30] of the year 2009. They have used refined set of protein–ligand binding data which comprised 1741 complexes out of which they had excluded the complex in which the ligands which were polymer, peptides and ATPs. We have also eliminated those and 1387 complexes were used. Random forest algorithm with 400 iterations and 30 features in each iteration was used to predict. Prediction was analyzed using 10-fold cross validation (Fig. 5d). As they had already drawn a comparison among other models and programs which are used in prediction and stated their performance was significantly better, so we have considered their method as benchmark for our comparison an observed our method and features outperformed theirs.

Comparison of our results with the published one is listed in Table 2 and represented in Fig. 6.

Blind set validation

We have also validated our prediction model using external test set. Out of the 2864 instances we have randomly sampled 80%

 This journal is © The Royal Society of Chemistry 2018

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

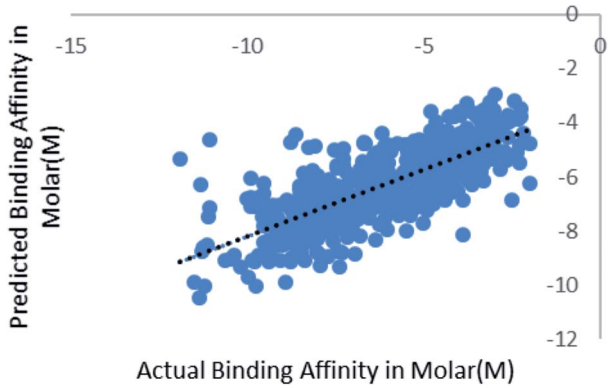

Fig. 7 Scatter plot for actual *vs.* predicted binding affinity of external dataset using Random forest.

Table 3 Result of prediction of feasibility of protein–ligand interaction

TP rate	0.968
FP rate	0.056
Precision	0.968
Recall	0.968
F-Measure	0.967
MCC	0.927
ROC area	0.994
PRC area	0.994

Table 4 Change in true positive rate of protein binding prediction using Random forest algorithm with respect to decrease in number of features

Number of attributes	TP rate
127	0.968
93	0.969
82	0.965
72	0.966
64	0.967
58	0.966
54	0.968
48	0.965
38	0.963
28	0.96
18	0.96
10	0.93

of the data in training set and rest 20% in the test set. Model was trained with 2291 instances and was supplied with 573 test instances. The blind validation also gave promising results, $R^2 = 0.75$ and RMSE = 1.38. The result is represented in Fig. 7.

Negative set validation

The algorithm and features were also validated against non-binders. Non-binding or decoy data was used from the DUD-E database.[51] We have used 12 proteins with pdb id (1e66, 2oi0, 3d0e, 1bcd, 3odu, 3ccw, 3g0e, 2ojg, 3bgs, 2azr, 1ype, 1sqt) from the DUD-E database which contribute in 2249 non-binders and 1110 binders. The same features were calculated for all the binders and non-binders. Random forest algorithm was trained using 10-fold cross validation in prediction. The prediction results are shown in Table 3. The correctly classified instances are 3250 means 96.755% accuracy and incorrectly classified instances 109 3.245%.

Feature selection

We have used 127 features of protein and ligand for training. We have further explored and employed attribute selection filter of Weka package[25,26] which ranks the attributes on the basis of their information gain. Out of 127, we have observed a set of 93 attributes giving the best prediction results shown in Table 4. However, there is not significant difference in the results, so top 18 (9 of ligand and 9 of protein) attributes can be used. Ranking list of the attributes is given in ESI 4.†

Conclusion

Discovery of potential drug or lead through protein–ligand interaction is a herculean task. As the interaction between specific ligand and protein depend on some characteristic features, determining a particular feature of ligand and protein plays a crucial role in identifying the interaction. The aim of our study is to predict whether an unknown ligand can interact with a protein, which may be utilised as a potential lead. Towards this end we exploit the binding energy in terms of dissociation constant K_d and inhibition constant K_i using Machine Learning algorithms with a few significant features of interaction. This method reduces the running time in comparison of state-of-the art computational techniques. Out of various machine learning algorithms like multilayer perceptron, SVM and Gaussian Process, Random forest model best suited the protein–ligand binding energy prediction problem. RF model's performance is highly dependent on the number of iterations (trees) and number of features used to build each tree, on the other hand machine learning relies upon the number and significance of features on which model is trained upon. Use of too many non-significant features may cause the machine to learn fuzzy patterns, leading to poor prediction. We have performed extensive experimentation in order to select optimum parameters of the ML algorithms which not only reduced the run time but also increase the accuracy of prediction. Further comparative study revealed that our strategy and the RF-model perform much better in the diverse dataset towards prediction of the unknown interaction. These models have the potential to identify the binding site for the interaction of protein and ligand based on their structural, physicochemical, and coordinate features.

Conflicts of interest

There are no conflicts to declare.

Acknowledgements

Indra Kundu and Raja Banerjee would like to acknowledge DBT-BIF (WBUT), Govt. of India (Sanction No. BT/BI/25/020/2012

(BIF)), for partial financial support and computational facility at WBUT.

References

1 A. D. Joseph, W. H. Ronald and G. G. Henry, The price of innovation: new estimates of drug development costs, *J. Health. Econ.*, 2003, **22**(2), 151–185.
2 D. M. Lorber, Computational drug design, *Chem. Biol.*, 1999, **6**(8), 227–228.
3 A. C. Anderson, The process of structure-based drug design, *Chem. Biol.*, 2003, **10**, 787–797.
4 B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts and J. D. Watson, *Molecular Biology of The Cell*, Garland Science, 4th edn, New York, 2002.
5 H. Frauenfelder, S. Sligar and P. Wolynes, The energy landscapes and motions of proteins, *Science*, 1991, **254**, 1598–1603.
6 A. Ostermann, R. Waschipky and F. G. Parak, Ligand binding and conformational motions inmyoglobin, *Nature*, 2000, **404**, 205–208.
7 G. Sliwoski, S. Kothiwale, J. Meiler and E. Lowe, *Computational Methods in Drug Discovery. Pharmacol. Rev.*, 2014, **66**, 334–395.
8 P. M. Colman, Structure-based drug design, *Curr. Opin. Struct. Biol.*, 1994, **4**, 868–874.
9 L. M. Amzel, Structure-based drug design, *Curr. Opin. Biotechnol.*, 1998, **9**, 366–369.
10 I. Muegge and M. Rarey, Small Molecule Docking and Scoring, *Rev. Comput. Chem.*, 2001, **17**, 1–60.
11 N. Brooijmans and I. D. Kuntz, Molecular recognition and docking algorithms, *Annu. Rev. Biophys. Biomol. Struct.*, 2003, **32**, 335–373.
12 I. Halperin, B. Ma, H. Wolfson and R. Nussinov, Principles of Docking: An Overview of Search Algorithms and a Guide to Scoring Functions, *Proteins: Struct., Funct., Genet.*, 2002, **47**, 409–443.
13 E. Alpaydın, *Introduction to machine learning*, MIT Press, 2nd edn, Cambridge, MA, 2009.
14 T. M. Mitchell, *Machine Learning*, McGraw Hill, 1st edn, New York, 1997.
15 C. Manly, S. Louise-May and J. Hammer, The impact of informatics and computational chemistry on synthesis and screening, *Drug Discovery Today*, 2001, **6**, 1101–1110.
16 A. Smith and C. Tony. Introducing Machine Learning Concepts with WEKA, in *Statistical Genomics: Methods and Protocols*, Springer New York, New York, NY, 2016.
17 I. Dubchak, I. Muchnik, S. R. Holbrook and S.-h. Kim, Prediction of protein folding class using global description of amino acid sequence, *Proc. Natl. Acad. Sci. U. S. A.*, 1995, **92**, 8700–8704.
18 X.-W. Chen and M. Liu, Prediction of protein–protein interactions using random decision forest framework, *Bioinformatics*, 2005, **21**, 4394–4400.
19 J. R. Bock and D. A. Gough, Predicting protein–protein interactions from primary structure, *Bioinformatics*, 2001, **17**, 455–460.
20 Z. Yuan, K. Burrage and J. S. Mattick, Prediction of protein solvent accessibility using support vector machines, *Proteins*, 2002, **48**, 566–570.
21 Y.-D. Cai, X.-J. Liu, X.-b. Xu and K.-C. Chou, Prediction of protein structural classes by support vector machines, *Comput. Chem.*, 2002, **26**, 293–296.
22 L. Jacob and J. P. Vert, Protein–ligand interaction prediction: an improved chemogenomics approach, *Bioinformatics*, 2008, **24**, 2149–2156.
23 M. Yarimizu, C. Wei, Y. Komiyama, K. Ueki, S. Nakamura, K. Sumikoshi, T. Terada and K. Shimizu, Tyrosine Kinase Ligand-Receptor Pair Prediction by Using Support Vector Machine, *Adv. Bioinf.*, 2015, 1–5.
24 C. X. Xue, R. S. Zhang, H. X. Liu, X. J. Yao, M. C. Liu, Z. D. Hu and B. T. Fan, QSAR Models for the Prediction of Binding Affinities to Human Serum Albumin Using the Heuristic Method and a Support Vector Machine, *J. Chem. Inf. Comput. Sci.*, 2004, **44**, 1693–1700.
25 E. Frank, M. Hall, I. H. Witten and C. J. Pal, The WEKA Workbench. Online Appendix for *Data Mining: Practical Machine Learning Tools and Techniques*, 4th edn, Morgan Kaufmann, 2016.
26 M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, *The WEKA Data Mining Software: An Update. SIGKDD Explorations*, 2009, 11, pp. 10–18.
27 W. Deng, C. Breneman and M. J. Embrechts, Predicting Protein−Ligand Binding Affinities Using Novel Geometrical Descriptors and Machine-Learning Methods, *J. Chem. Inf. Model.*, 2004, **44**, 699–703.
28 C. Kramer and P. Gedeck, Global Free Energy Scoring Functions Based on Distance-Dependent Atom-Type Pair Descriptors, *J. Chem. Inf. Model.*, 2011, **51**, 707–720.
29 Y. Wang, Y. Guo, Q. Kuang, X. Pu, Y. Ji, Z. Zhang and M. Li, A comparative study of family-specific protein–ligand complex affinity prediction based on random forest approach, *J. Comput.-Aided Mol. Des.*, 2015, **29**, 349–360.
30 R. Wang, X. Fang, Y. Lu and S. Wang, Collection of Binding Affinities for Protein−Ligand Complexes with Known Three-Dimensional Structures, *J. Med. Chem.*, 2004, **47**, 2977–2980.
31 R. Wang, X. Fang, Y. Lu, C. Y. Yang and S. Wang, The PDBbind Database: Methodologies and updates, *J. Med. Chem.*, 2005, **48**, 4111–4119.
32 H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne, The Protein Data Bank, *Nucleic Acids Res.*, 2000, **28**, 235–242.
33 V. Lee, *Peptide and Protein Drug Delivery*, CRC Press, 1st edn, USA, 1990.
34 C. B. Anfinsen, Principles that govern the folding of protein chains, *Science*, 1973, **181**, 223–230.
35 S. Jones and J. M. Thornton, Principles of protein–protein interactions, *Proc. Natl. Acad. Sci. U. S. A.*, 1996, **93**, 13–20.
36 W. Kabsch and C. Sander, Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features, *Biopolymers*, 1983, **22**, 2577–2637.
37 G. E. Schulz and R. H. Schirmer, *Principles of Protein Structure*, Springer, 1st edn, New York, 1979.

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

 This journal is © The Royal Society of Chemistry 2018

Open Access Article. Published on 28 2018. Downloaded on 13.05.26 14:03:04.
This article is licensed under a Creative Commons Attribution 3.0 Unported Licence.

38 M. C. Chervenak and E. J. Toone, A Direct Measure of the Contribution of Solvent Reorganization to the Enthalpy of Binding, *J. Am. Chem. Soc.*, 1994, **23**, 10533–10539.
39 C. A. Lipinski, F. Lombardo, B. W. Dominy and P. J. Feeney, Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings, *Adv. Drug Delivery Rev.*, 2001, **46**, 3–26.
40 C. A. Lipinski, Lead- and drug-like compounds: the rule-of-five revolution, *Drug Discovery Today: Technol.*, 2004, **1**, 337–341.
41 S. Kim, P. A. Thiessen, E. E. Bolton, J. Chen, G. Fu, A. Gindulyte, L. Han, J. He, S. He, B. A. Shoemaker, J. Wang, B. Yu, J. Zhang and S. H. Bryant, PubChem Substance and Compound databases, *Nucleic Acids Res.*, 2016, **44**, 1202–1213.
42 C. W. Yap, PaDEL-descriptor: An open source software to calculate molecular descriptors and fingerprints, *J. Comput. Chem.*, 2011, **32**, 1466–1474.
43 D. J. C. Mackay, *Introduction to Gaussian Processes*, Cambridge University, UK, 1998.
44 H. Akaike, A new look at the statistical model identification, *IEEE Trans. Autom. Control*, 1974, **19**, 716–723.
45 W. S. Sarle, Neural Networks and Statistical Models, *Proceedings of the Nineteenth Annual SAS Users Group International Conference*, 1994, pp. 1–13.
46 L. V. Fausett, *Fundamentals of Neural Networks: Architectures, Algorithms and Applications*, Prentice-Hall, USA, 1994.
47 J. Platt, Fast Training of Support Vector Machines Using Sequential Minimal Optimization, in *Advances in Kernel Methods – Support Vector Learning*, ed. B. Schoelkopf, C. Burges, A. Smola, MIT Press, 1998.
48 S. S. Keerthi, S. K. Shevade, C. Bhattacharyya and K. R. K. Murthy, Improvements to Platt's SMO Algorithm for SVM Classifier Design, *Neural Comput.*, 2001, **13**, 637–649.
49 J. G. Cleary and L. E. Trigg. An Instance-based Learner Using an Entropic Distance Measure, *12th International Conference on Machine Learning*, 1995, pp. 108–114.
50 L. Breiman, Random Forests, *Mach. Learn.*, 2001, **45**, 5–32.
51 M. M. Mysinger, M. Carchia and J. J. Irwin, Shoichet BK Directory of Useful Decoys, Enhanced (DUD-E): Better Ligands and Decoys for Better Benchmarking, *J. Med. Chem.*, 2012, 6582–6594.