BMC Bioinformatics

RESEARCH

Open Access

Improve hot region prediction by analyzing different machine learning algorithms

Jing Hu[1,2], Longwei Zhou[1,2], Bo Li[1,2], Xiaolong Zhang[1,2]* and Nansheng Chen[3]*

From Fifteenth International Conference on Intelligent Computing (ICIC 2019)
Nanchang, China. 3-6 August 2019

*Correspondence: xiaolong.zhang@wust.edu.cn; chenn@sfu.ca
[1] School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, Hubei, China
[3] Molecular Biology and Biochemistry, Simon Fraser University, Vancouver, BC, Canada
Full list of author information is available at the end of the article

Abstract

Background: In the process of designing drugs and proteins, it is crucial to recognize hot regions in protein–protein interactions. Each hot region of protein–protein interaction is composed of at least three hot spots, which play an important role in binding. However, it takes time and labor force to identify hot spots through biological experiments. If predictive models based on machine learning methods can be trained, the drug design process can be effectively accelerated.

Results: The results show that different machine learning algorithms perform similarly, as evaluating using the F-measure. The main differences between these methods are recall and precision. Since the key attribute of hot regions is that they are packed tightly, we used the cluster algorithm to predict hot regions. By combining Gaussian Naïve Bayes and DBSCAN, the F-measure of hot region prediction can reach 0.809.

Conclusions: In this paper, different machine learning models such as Gaussian Naïve Bayes, SVM, Xgboost, Random Forest, and Artificial Neural Network are used to predict hot spots. The experiment results show that the combination of hot spot classification algorithm with higher recall rate and clustering algorithm with higher precision can effectively improve the accuracy of hot region prediction.

Keywords: Hot region, Protein–protein interaction, Hot spot, DBSCAN, SVM, Gaussian Naïve Bayes

Background

Proteins perform their corresponding biological functions by interacting with proteins or other molecules, among which the interactions between proteins are the most important. As the carrier with life activities, protein plays an important role in every link with life, such as gene regulation, signal transduction, gene expression and other basic cellular functions in life activities [1]. The binding between the two proteins is mainly based on affinity. Studies have shown that only a few residues on the protein–protein interaction surface provide the majority of the binding free energy, and these residues are called hot residues [2, 3]. At the same time, these hot residues usually gather closely on the protein interaction surface [4]. That is, hot residues often appear in the form of interaction clusters on the interaction interface, and the residues in these clusters interact with each other to form a stable network structure, which is called the hot region. In drug design, the study of hot region plays a positive role in the prediction of protein functional sites, drug target and protein design [5, 6].

The study showed that the hot region in protein–protein interaction is composed of at least three hot spot residues, which on the protein interaction interface. Starting from the machine learning method, Xia [7] based on protein sequence, structure and neighborhood features to extract feature, combined with maximum relevance and minimum redundancy algorithm, and create a hot spot prediction model based on Support Vector Machine (SVM). Then, Tuncbag [8] proposes an empirical model that combines accessible surface area and pairing propensity to predict hot spots residues, which improves the accuracy of hot spot residue prediction. To achieve better property based on structural features, Huang [9] designed an assembly learning method that combines SMOTE with data imbalance to predict hot spot residues. Hu [10] constructed a new learning hot spot prediction model based on protein sequence feature.

A better hot spot residue prediction model is beneficial to the prediction of hot region. Cukuroglu [11] analyzed hot region according to the characteristics of hot spot residues and the formation rules of hot region, and established a hot region database named Hot Region. Pons uses the small-world residue network to predict hot regions, using the small-world network method, and through the relationship between the residues, the residues can form an interconnected network [12]. Nan [13] used complex network and community detection methods to predict hot region in protein interactions. In the process of predicting hot regions, some False Positives (FP) and False Negatives (FN) in the prediction results are corrected by using the topological characteristics of residues in the network, so as to improve the accuracy of predicting hot regions. An approach based on a new clustering algorithm called Local Community Structure Detecting (LCSD) to identify the hot regions was proposed by Lin [14], with an enhanced maximum relevance minimum redundancy algorithm to upgraded prediction performance in the feature selection process of hot spot prediction.

The prediction of hot spot residues in protein interaction is the first step for predicting hot region. It is necessary to identify the hot spot residue as accurately as possible on the protein–protein interaction. Due to the limitation of amino acid mutation to alanine in the data set and the imbalance between hot spots and non-hot spots in the data set, the prediction effect of hot spots and hot regions in protein–protein interaction is not significant. With the release of the SKEMPI2.0 dataset, there were twice as many mutations to alanine in the SKEMPI2.0 dataset as there were in the previous version of SKEMPI1.0 [15, 16]. From multiple perspectives, we extracted features according to protein sequence, structure and the relationship between amino acid and built several machines learning models to predict hot spots residues. The hot spot residue prediction results of different machine learning algorithms were analyzed, and DBSCAN clustering algorithm was combined to form hot spots [17]. The experiment results show that the combination of hot spot classification algorithm with higher recall rate and clustering algorithm with higher precision can effectively improve the accuracy of hot region prediction.

Results

Dataset

The datasets used in this article are from up-to-date SKEMPI 2.0 databases (Structural database of Kinetics and Energetics of Mutant Protein Interactions). The dataset encompasses the variation data of thermodynamic parameters and kinetic rate constant parameters before and after the mutation of amino acids to alanine, leucine and other different types of amino acids. The data in the SKEMPI2.0 dataset are all from experiments or authoritative published literature. Due to different experimental environments, mutations at the same site may have multiple different values of binding free energy in the database, so we use the average value of binding free energy to replace the repeated data and eliminate the empty data. After that, 180 protein complexes were obtained from the SKEMPI2.0 database, and the corresponding structural information of each complex was obtained from the PDB database (Protein Data Bank). Each protein complex consists of a stack of interface residues whose accessible surface area is reduced by more than 1 Å during the formation of the protein complex. We defined hot and non-hot residues according to the energy changes in the alanine mutation experiment of these residues.

In SKEMPI2.0 data set, the average value of ΔΔG are used as the final result for the binding free energy of the same site under different experiments. At present, most of the research on hot spot residues adopts such a definition standard: In the alanine experiment, the interface residues with binding free energy change greater than 2 kcal /mol were regarded as hot spot residues, and the interface residues with binding free energy change less than 0.4 kcal /mol were regarded as non-hot spot residues, and the data that the binding free energy varies between 0.4 kcal/mol and 2.0 kcal/mol are discarded. We found that using 2.0 kcal/mol as the hot spot definition standard of the SKEMPI2.0 data set would cause the data to be extremely unbalanced, resulting in a sharp drop in the recall rate of the hot spot prediction model. The strategy of using 1.0 kcal/mol as the threshold can make better use of the entire data set. Therefore, we define more than 1.0 kcal/mol as hot spot residues, and less than 1.0 kcal/mol as non-hot spot residues.

Using 1.0 kcal/mol as the standard for defining hot and non-hot spots, we finally obtained 2326 interface residues from 180 protein complexes about SKEMPI2.0 database, including 1513 non-hot and 813 hot spot residues. Table 1 shows the specific distribution of 20 amino acids in the data set, which indicates that amino acids with aromatic side chains are more likely to have hot spots residues and TYR, ARG, LYS, and GLU are easier to exist in the hot spot residues. Otherwise, TYR, SER, ARG, and GLU are more likely to appear in the interface residues.

More than two-fifths of the data in the SKEMPI 2.0 database come from the previous version SKEMPI 1.0. Since a lot of research has been done in SKEMPI 1.0, the main work in this article focuses on the dataset extended by SKEMPI 2.0. In order to enhance the stability of the prediction model, in the SKEMPI 2.0 expansion data, we added the protein complex containing the number of interface residues less than 3 into the training set, and put the remaining data of expansion data the test set. The rest of data in SKEMPI 2.0 is regarded as the testing set. Table 2 shows the detailed data of training set and test set.

Table 1 Distribution of data in SKEMPI 2.0

Amino acid	Non-hot spots	Hot spots	All residues	Ratio of hot spots	property of side chain
SER	143	21	164	0.128	Hydroxyl-containing
CYS	6	1	7	0.143	Sulfur-containing
GLN	107	29	136	0.213	Amid
THR	114	31	144	0.214	Hydroxyl-containing
PRO	42	17	59	0.288	Cyclic
ASN	101	41	142	0.289	Amid
GLY	48	20	68	0.294	Aliphatic
VAL	70	30	100	0.3	Aliphatic
GLU	154	68	222	0.306	Acid
HIS	60	28	88	0.318	Basic aromatic
MET	23	11	34	0.326	Sulfur-containing
LYS	131	64	195	0.328	Basic
ARG	146	80	226	0.354	Basic
ASP	101	41	142	0.409	Acid
LEU	48	41	89	0.419	Aliphatic
ILE	72	52	124	0.461	Aliphatic
PHE	51	55	106	0.519	Aromatic
TYR	75	104	179	0.581	Aromatic
TRP	21	50	71	0.704	Aromatic
All	1513	813	2623	0.35	none

Table 2 Training set and testing set

Dataset	Non-hot spots	Hot spots	All residues	Complexes
Training set	864	390	1254	101
Testing set	649	423	1072	79

Experimental results

In the hot spot prediction stage, we need to predict the hot spot residues as precisely as possible. The optimal feature subset was obtained through feature selection by mRMR (Max-Relevance and Min-Redundancy) algorithm. The calculated score and details for each feature are in Additional file 1. In this section, we used several traditional machine learning methods to build the hot spot prediction model. such as RF (Random Forest), Xgboost (eXtreme Gradient Boosting Decision Tree), ANN (Artificial Neural Network) and SVM (Support Vector Machine) [18–21], and compared these widely used methods in hot spot research with Gaussian Bayes. For the ANN algorithm, we build a five-layer neural network, in which the activation function of each layer is ReLU (rectified linear units) [22]. Because neural networks have automatic feature fusion functions, in addition to ANN, other machine learning methods all use mRMR feature selection algorithm for feature selection in the feature selection process.

Table 3 shows the prediction results of different machine learning algorithms in the hot spot prediction experiment. According to Table 3, it can be concluded that the accuracy and precision of the Xgboost method is the best compared to other methods. The performance of GNB method on recall and F-measure is the highest. The experimental

Table 3 Comparison of results with different methods to predict hot spots

Methods	Hot Spot			
	Accuracy	Recall	Precision	F-measure
GNB	0.674	**0.792**	0.561	**0.657**
SVM	0.696	0.6	0.618	0.609
Xgboost	**0.726**	0.596	**0.672**	0.632
RF	0.693	0.596	0.615	0.605
ANN	0.715	0.679	0.632	0.655

The highest value in each column is shown in bold

Table 4 Comparison of results with different methods to predict hot regions

Methods	Hot Region		
	Recall	Precision	F-measure
GNB	**0.766**	**0.923**	**0.809**
SVM	0.617	0.725	0.667
Xgboost	0.574	0.6	0.587
RF	0.617	0.644	0.63
ANN	0.596	0.538	0.567

The highest value in each column is shown in bold

data showed that GNB could correctly predict more hot spot residues, and Xgboos could correctly predict more non-hot spot residues. Due to hot spot residues are fewer than non-hot spot residues on the training set, the accuracy of GNB is lower than that of Xgboost. The more hot spots residues are predicted, the more hot regions we can get during clustering process.

After the hot spot residue prediction, DBSCAN clustering algorithm was used to predict the hot region. The clustering results are presented in Table 4. The F-measure represents the balance between recall and precision, using F-measure as the evaluation criterion, the two parameters "Min" and "ε" in the DBSCAN algorithm are determined by the grid search method. Experiment results show that the hot region prediction model combined with GNB and DBSCAN algorithms is significantly better than other methods. The Fig. 1 showed that the number of hot spot residues correctly predicted by GNB in a single hot region of 47 standard hot regions is close to the performance of other algorithms. The more true positive hot spot residues are predicted, the more hot region will be correctly constructed. Otherwise, few hot spots will cause the recall rate of hot region prediction to decrease. Besides, because of the lack of true positive residues, some hot regions are incorrectly clustered in the process of forming true positive hot regions with relatively unconstrained parameters.

Standard hot regions and predicted hot regions

In this paper, we define the standard hot region according to Keskin [23]. The definition of a standard hot region: Each hot region is composed of at least three hot spot residues, each hot spot residue is assumed to be a regular sphere, and the Cα atom of each hot spot residue is considered to be the center of the sphere. Calculate the radius of the sphere from the volume of the hot spot residue sphere [24]. If the distance between the

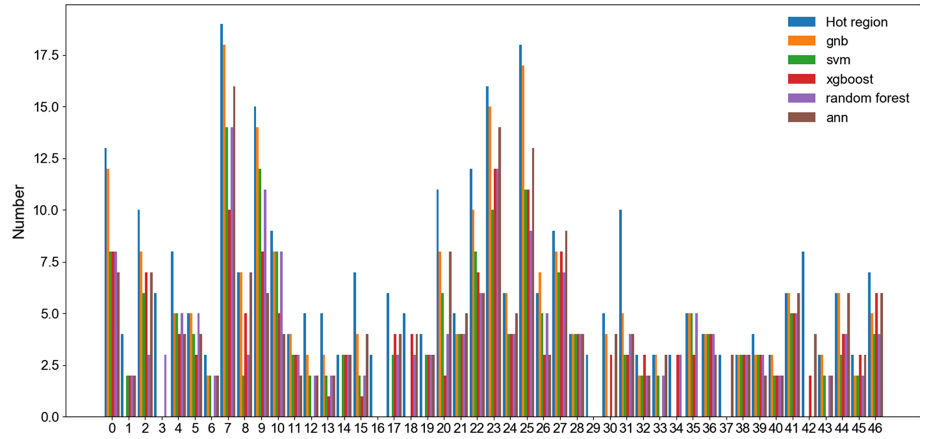


Fig. 1 Distribution of hot spots in hot regions. The x-axis corresponds to the 47 standard hot regions in Additional file 2. The height of the bar shows the number of hot spots in those 47 standard hot regions and the number of true positive hot spots in the predicted hot regions

centers of two spheres (two Cα-atoms of two hot spots) is less than the sum of the radius of the two spheres plus a tolerance distance (2 Å), the two hot spot residues are flagged to be clustered and to form a network in the hot region. The prediction accuracy of the prediction model for different amino acid mutations is compared with the standard hot regions. Finally, 47 standard hot regions were detected. For detailed results, see Additional file 2.

Because of a hot region contains at least three hot spot residues, and the maximum distance between two contacting amino acids in the same hot region is 9.5 Å, in the process of setting the parameters of "Min" and "ε" of DBSCAN algorithm, we set "Min" to be greater than 3 and set "ε" to be less than 9.5 Å. According to the results of all methods clustering, when "Min" is 3 and "ε" is 9.5 Å, the DBSCAN algorithm could achieves the optimal F-measures in other methods except the GNB. However, under this parameter setting, GNB recognizes more non-hot spots as hot spots. With "Min" set to 4 and "ε" set to 8.5 Å, the GNB could more accurately label non-hot spot as noises. The clustering algorithm will perform well when the factors that form hot regions are improved.

Methods

The hot spot residue that was predicted by different machine learning algorithm are clustered to form hot region with DBSCAN algorithm. Then, we evaluate the hot region prediction by comparing hot regions from predicted models with the standard hot regions in dataset.

Binding free energy changes

Data on single amino acids mutated into alanine was extracted from the SKEMPI 2.0 (https://life.bsc.es/pid/skempi2/) which contains affinities of wild type complexes and affinities of mutated complexes measured by biological experiments in the scientific literature. Because binding free energy changes of multiple mutated residues has not been accumulated based on such single mutated residues, more samples from multiple mutated samples cannot be deduced. The data with affinity could not be measured were deleted. We calculate the bind free energy of amino acid mutations according to Formula (1), where the binding affinity (K_d) is determined according to biological experiments such as Surface Plasmon Resonance and Isothermal Titration Calorimetry [25, 26], and R is the gas constant (8.314/4184) kcal/(K*mol) (1 kcal = 4.184 kJ) and T is the experimental temperature (in the range of 273 K to 323 K). The change of bind free energy ($\Delta\Delta G$) can be calculated based on the ΔG_{mut} and ΔG_{wt}, which be calculated from Eqs. (1), (2).

$$\Delta G = -RT\ln(Kd) \tag{1}$$

$$\Delta\Delta G = \Delta G_{mut} - \Delta G_{wt} \tag{2}$$

Feature selection

In this paper, the structural information on protein complexes is from PDB (Protein Data Bank) [27]. We extracted features including solvent accessible surface area, protrusion index, relative accessible surface area, binding sites and the depth index with aimed amino acid from PSAIA [28], and calculate RctASA, RctmPI by formula (3), (4), conservation scores from the ConSurf server [29], the attributes of amino acid side chains, the hydrophobic index, and the interaction numbers between two amino acids. The detailed features are shown in Additional file 1. The additional file indicated that the attribution of the amino acid side chain is a discrete variable, we encode the feature with one-hot, which is a feature extraction method that can deal with discontinuous numerical features. In total, we collected 83 features from protein structural, sequence and energy. Since not every feature contributes the same to the model, we determined the optimal feature subset by combining mRMR algorithm (minimum Redundancy Maximum Relevance) [30]; the mutual information I (x, y) is labeled as:

$$RctASA = \frac{[unbound\ total\ ASA] - [bound\ total\ ASA]}{[unbound\ total\ ASA]} \tag{3}$$

$$RctmPI = \frac{[unbound\ total\ mean\ PI] - [bound\ total\ mean\ PI]}{[unbound\ total\ mean\ PI]} \tag{4}$$

$$I(x,y) = \iint P(x,y)\log\frac{P(x,y)}{p(x)p(y)}dxdy \tag{5}$$

and the maximum correlation criterion and the minimum redundancy criterion are defined as:

$$\max D(F,c), D = \frac{1}{F}\sum_{Xi\in F} I(Xi,c) \tag{6}$$

$$\min R(F), R = \frac{1}{F^2}\sum_{Xi,Xj\in F} I(Xi,Xj) \tag{7}$$

According to the training set, a feature list is produced. To discover the highest F-score combination, we applied incremental feature selection and got a rank of all features in descending order. Every time we combined the feature ranked at the top with its next one to obtain the F-measure in machine learning models, then selected the set of features with the best F-measure.

Prediction of hot regions

There are non-hot spot residues and hot spot residues in the dataset. However, we needed to detect as many hot spots as possible in the hot regions which contain hot spot residues.

Naïve Bayes classifier

We constructed a Gaussian Naïve Bayes classifier given a set of training examples with class labels and then used the model to distinguish between non-hot spot residues and hot spot residues [31–33]. One example is a tuple of features $(x_1, x_2, \ldots, x_n)$ of one sample (x_i) and the class label c of the sample, so X is all samples and C is the classification variable. In our experiment, we assumed that there are two classes: c = 0 (non-hot spot residue) and c = 1 (hot spot residue). According to the Bayes Rule, the probability of one sample E = $(x_1, x_2, \ldots, x_n)$, being class c of sample E is:

$$p(c|E) = \frac{p(E|c)p(c)}{p(E)} \tag{8}$$

Sample E is classified as class 1 if and only if $f_b(E)$ is more than 1, otherwise, sample E will be classified as class 0:

$$f_b(E) = \frac{p(C=1|E)}{p(C=0|E)} \geq 1 \tag{9}$$

where $f_b(E)$ is called a Bayesian classifier.

Assuming that all features are independent given the value of the class variable, that is

$$p(E|c) = p(x_1, x_2, \ldots, x_n|c) = \prod_{i=1}^{n} p(x_i|c) \tag{10}$$

the resulting classifier is then:

$$f_{nb}(E) = \frac{p(C=1)}{p(C=0)} \prod_{i=1}^{n} \frac{p(x_i|C=1)}{p(x_i|C=0)} \tag{11}$$

The function $f_{nb}(E)$ is called a Naïve Bayesian classifier, or simply Naïve Bayes (NB). When we used the Gaussian distribution to calculate the $p(x_i|C)$, the classifier becomes Gaussian Naïve Bayes. Given probability distribution is under Gaussian distribution, the function is:

$$g(x_i, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x_i-\mu)}{2\sigma^2}} \tag{12}$$

Support vector machine

A support vector machine constructs a hyper-plane or set of hyper-planes in a high or infinite-dimensional space, which can be used for classification, regression or other tasks. Intuitively, a good separation is achieved by the hyper-plane that has the largest distance to the nearest training data points of any class (so-called functional margin), since in general the larger the margin the lower the generalization error of the classifier. Given training dataset $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ the goal of the classification is to find a maximum-margin hyperplane $w^t x + b = 0$.

Used as the output of SVM in binary classification:

$$f(x, W) = \operatorname{sgn}\left(\sum_{i=1}^{N} w_i K(x_j, x_i) + b\right) \tag{13}$$

optimized objective function:

$$\begin{aligned} &J = W^T W = \|W\|^2 \\ &\text{s.t.: } y_J \left[\sum_{i=1}^{N} w_i K(x_j, x_i) + b\right] \geq 1, j = 1, \ldots, N \end{aligned} \tag{14}$$

where N is sample size, W is the output adjustable parameter vector of support vector machine, $K(x_j, x_i)$ is kernel function.

The objective function J is to ensure the optimality of classification, and the constraint condition is to ensure the correctness of classification. In order to eliminate the influence of noise and abnormal samples, relaxation variables are introduced as follows:

$$\mathrm{J} = \frac{1}{2} W^T W + C \sum_{i=1}^{N} \xi_j \tag{15}$$

$$y_j \left[\sum_{i=1}^{N} w_i K(x_j, x_i) + b\right] \geq 1 - \xi_j, j = 1, \ldots N, \xi_j \geq 0 \tag{16}$$

Xgboost

Xgboost (eXtreme Gradient Boosting) is one type of ensemble learning. The boosting method, by combining multiple weak learners, gives the final learning result. We used the theory of regression tasking to build the optimal Boosting model, regardless of the classification or regression.

$$\mathrm{Obj} = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \tag{17}$$

The objective function consists of two parts, the first part is used to measure the difference between the predicted score and the real score, and the second part is the regularization term.

The newly generated tree is to fit the residual error predicted last time, that is, when t trees are generated, the prediction score can be written as:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(\gamma_i) \tag{18}$$

where t is the number of leaf nodes, and w is the fraction of leaf nodes. Gamma can control the number of leaf nodes; a lambda can control the number of leaf nodes so they do not get too large as this avoids overfitting.

According to (17) (18) the objective function:

$$\text{Obj}^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(X_i)\right) + \Omega\left(f_t\right) \tag{19}$$

in Xgboost is used to approximate it using its Taylor second-order expansion at $f_t = 0$. Therefore, the objective function is approximate:

$$\text{Obj}^{(t)} \simeq \sum_{i=1}^{n} \left[l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(X_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega\left(f_t\right) \tag{20}$$

where g_i is a derivative, and h_i is the second derivative.

The objective function can be simplified as:

$$\widetilde{\text{Obj}}^{(t)} \simeq \sum_{i=1}^{n} \left[g_i f_t(X_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega\left(f_t\right) \tag{21}$$

Random forest

The advantage of a random forest algorithm is that it combines several weak classifiers into one strong classifier, which can resist the over-fitting of the decision tree by using a voting mechanism. Random forest has a strong generalization ability and high efficiency and accuracy for multidimensional data classification. The Gini coefficient is defined as follows:

$$\text{Gini}(p) = \sum_{k=1}^{K} p_k(1 - p_k) = 1 - \sum_{k=1}^{K} p_k^2 \tag{22}$$

where p_k is the probability that the sample is in class k. The probability of misclassification is (1-p_k). The Gini was calculated for each feature in each sample, and then we selected the feature with the optimal Gini coefficient, theta θ^*.

$$\theta^* = \min Gini(\theta_i) \tag{23}$$

where θ_i represents the feature of the i in the sample.

Artificial neural network

An artificial neural network (ANN) is a computational model based on the structure and functions of biological neural networks. Information that flows through the network affects the structure of the ANN because a neural network changes or learns, based on the input and output. We constructed networks containing five layers with activation function rectified linear units (ReLu) and the input layer size corresponds to the 83 features obtained from feature selection.

Evaluation

In this paper, we adopt the following general evaluation indicators to evaluate the performance of the prediction model for hot spots and hot region.

$$\text{Recall} = \text{TP}/(\text{TP} + \text{FN}) \tag{24}$$

$$\text{Precision} = \text{TP}/(\text{TP} + \text{FP}) \tag{25}$$

$$\text{F-measure} = 2 * \text{Recall} * \text{Precision}/(\text{Recall} + \text{Precision}) \tag{26}$$

When predicting hot spots, the following notations are used:

> True Positive (TP): The number of hot spots in predicted hot regions and also in standard hot regions;
> False Negative (FN): The number of hot spots that are not in predicted hot regions but in standard hot regions;
> False Positive (FP): The number of hot spots in predicted hot regions but not in standard hot regions;
> Precision represents the accuracy of the hot spot prediction, and Recall represents the coverage of predicted hot spots in standard hot regions. With a good balance between Precision and Recall, the F-measure offers a better overall accuracy of hot spot prediction.

However, for prediction of hot regions, the following notations are used:

> True Positive (TP): The number of hot regions in predicted hot regions and also in standard hot regions;
> False Negative (FN): The number of hot regions that are not in predicted hot regions but in standard hot regions;
> False Positive (FP): The number of hot regions in predicted hot regions but not in standard hot regions;

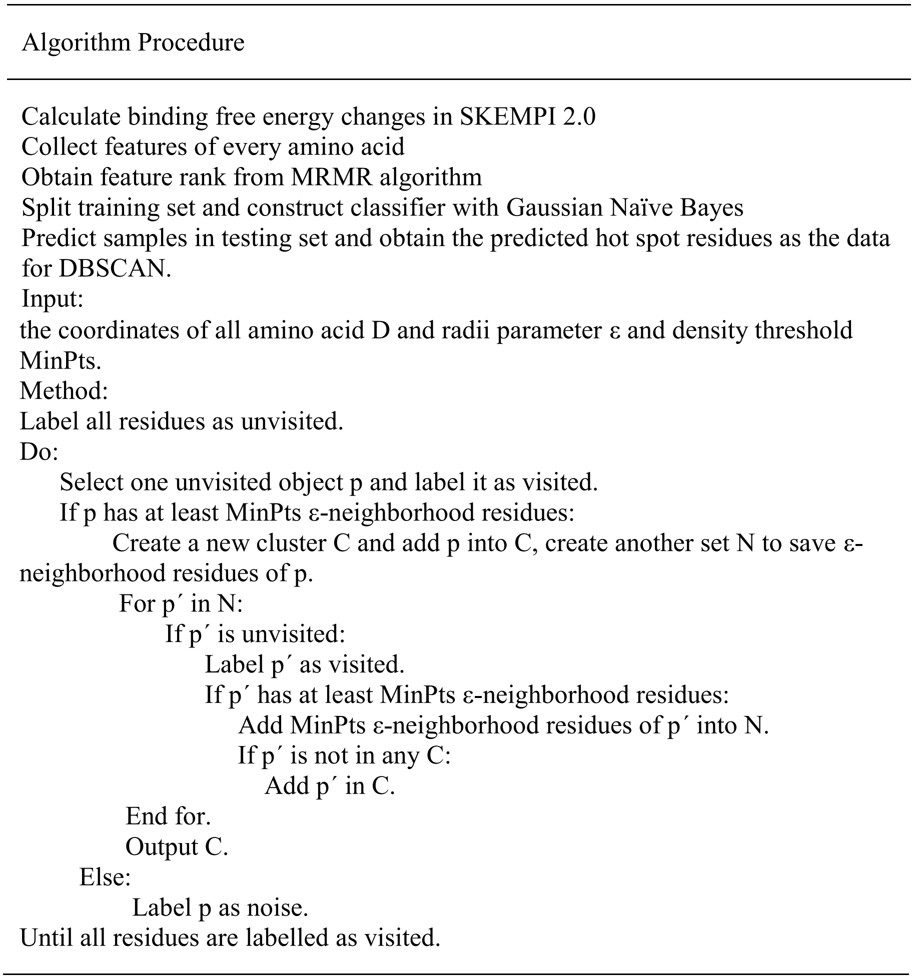
Similarly, Precision represents the accuracy of the hot region prediction, and Recall represents the coverage of predicted hot regions in standard hot regions. With a good balance between Precision and Recall, the F-measure offers a better overall accuracy in predicting hot regions than using either Precision or Recall solely.

Cluster

Cluster analysis is abbreviated as clustering, which is the process of dividing a data object into subsets. Each subset is a cluster. The clustering process makes the objects in the clusters similar to each other, but not similar to the objects in other clusters. Because the hot spot residues in protein–protein interactions are not evenly distributed on the interface of protein interactions, they are tightly gathered in a dense area. We use the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm to predict hot spots based on the characteristic that hot spot residues gather on the protein interaction interface in a flowing structure. The DBSCAN algorithm is suitable to cluster this kind of data. There are two hyperparameters "Min" and "ε" to be measured in this algorithm. "Min" represents the density of residue measured by the number of residues of it, and "ε" represents the radius of residue O as the center of a circle.

For a dataset D composed of residues with the parameters of "Min" and "ε", the residues with more than or equal to "Min" will be regarded as the core residue in its ε-neighborhood. After checking all residues, the core residues and their ε-neighborhood residues will make up the dense regions, which are the clusters we need.

The detailed process about clustering is that all residues should be defined as "unvisited" in the first stage. Then, we need to select randomly one residue p as the center of the circle and calculate the number of residues in its neighborhood to distinguish whether the residue is a core residue or not. If it is a core residue, we labelled the residue as "visited" and selected the neighborhood residues as the next detected objects. If existing core residues are in them, the process continues until the cluster C cannot be extended. Then we return to the beginning, select randomly one residue p in the remaining residues that was labeled as "unvisited" as the center of a circle, and repeat the process until all residues are "visited". Eventually, we will obtain several clusters.



```
Algorithm Procedure

Calculate binding free energy changes in SKEMPI 2.0
Collect features of every amino acid
Obtain feature rank from MRMR algorithm
Split training set and construct classifier with Gaussian Naïve Bayes
Predict samples in testing set and obtain the predicted hot spot residues as the data for DBSCAN.
Input:
the coordinates of all amino acid D and radii parameter ε and density threshold MinPts.
Method:
Label all residues as unvisited.
Do:
    Select one unvisited object p and label it as visited.
    If p has at least MinPts ε-neighborhood residues:
        Create a new cluster C and add p into C, create another set N to save ε-neighborhood residues of p.
        For p´ in N:
            If p´ is unvisited:
                Label p´ as visited.
                If p´ has at least MinPts ε-neighborhood residues:
                    Add MinPts ε-neighborhood residues of p´ into N.
                    If p´ is not in any C:
                        Add p´ in C.
        End for.
        Output C.
    Else:
        Label p as noise.
Until all residues are labelled as visited.
```

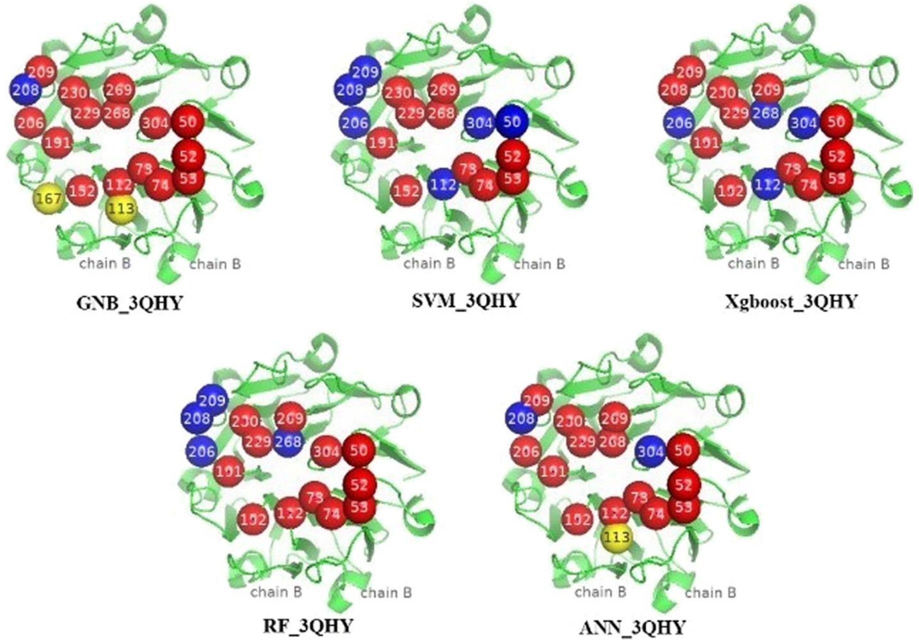


Fig. 2 Visual prediction results with PyMol of 3HQY using different methods. All residues are in chain B. Red spheres are predicted correctly in the standard hot region, Blue spheres are not predicted in the standard hot region, yellow spheres are mistaken as hot spot residues

Discussion

Comparison of prediction results visualization

We assumed that a hot region is predicted correctly only when 60 percent of hot spot residues in the standard hot regions occur in the predicted hot region. Therefore, if the results predicted by GNB are clustered to form a hot region that are regarded as true positive hot regions, the evaluation of the results of other machine learning methods for recall will be reduced, and the increase of mistaken hot regions will lead to the evaluation of precision is decrease. We visualized the hot spot residues and predicted hot spot residues of the protein complex 3HQY with PyMol software [34] in Fig. 2. In the protein complex 3QHY, there are 16 hot spot residues in the standard hot region. The GNB algorithm can correctly predict 15 true positive hot spot residues in the standard hot region and only two non-hot spot residues come within the predicted hot region. In addition to ANN, other models have higher accuracy for non-hot spot residues, but cannot predict more hot spot residues in the standard hot region, so the recall is low. ANN can correctly predict 14 hot spot residues in the standard hot region, but a non-hot spot residue was incorrectly predicted as a hot spot residue.

Conclusion

In this paper, we collect alanine mutations data from the latest presented SKEMPI 2.0 database. When we use 1.0 kcal/mol as the threshold for hot spot and non-hot spot residues, it shows that amino acids of aromatic are more likely to become hot spots residues. Furthermore, hot spot residues are 70.4% from TRP. In the first stage, we used the mRMR algorithm to rank the importance of every feature based on mutual information and RctmPI is the most important feature. In the next stage of predicting hot spot residues, the performance of all methods about F-measure is close, but Gaussian Naïve Bayes (GNB) has the best performance for recall, so that hot regions can be made up of enough true positive hot spot residues. In the final stage, the DBSCAN algorithm was selected to cluster the data for forming hot regions.

The combined method with Gaussian Naïve Bayes (GNB) and DBSCAN can effectively improve hot region predictions. Though several machine learnings methods are applied to test the performance, the limitation of the method is barely biological experiments involved. Thus, the next step is to collect and apply more biological data to verify the model.

Abbreviations

PPIs: Protein–Protein Interactions; PDB: Protein Data Bank; GNB: Gaussian Naïve Bayes; SVM: Support Vector Machine; Xgboost: EXtreme Gradient Boosting Decision Tree; RF: Random Forest; ANN: Artificial Neural Network.

Supplementary Information

The online version contains supplementary material available at https://doi.org/10.1186/s12859-021-04420-0.

Additional file 1: Result of MRMR algorithm rank.

Additional file 2: Standard hot regions and detailed experimental results for hot spot and hot region prediction.

Acknowledgements

The author would like to thank the editors and reviewers for their suggestions, which is a great help for this article, and the members of Machine Learning and Artificial Intelligence Laboratory, School of Computer Science and Technology, Wuhan University of Science and Technology, for their helpful discussion within seminars.

About this supplement

This article has been published as part of BMC Bioinformatics Volume 22 Supplement 3, 2021: Proceedings of the 2019 International Conference on Intelligent Computing (ICIC 2019): bioinformatics. The full contents of the supplement are available online at https://bmcbioinformatics.biomedcentral.com/articles/supplements/volume-22-supplement-3.

Authors' contributions

HJ, ZXL and ZLW contributed to the conception of the study; HJ and LB contributed significantly to analysis and manuscript preparation; ZLW and LB performed the data analyses and wrote the manuscript; CNS, ZXL and HJ helped perform the analysis with constructive discussions. All authors read and approved the final manuscript.

Funding

This work is supported by the National Natural Science Foundation of China (No. 61972299, No.61572381), Science and Technology Research Plan of Hubei Ministry of Education (Q20171105), and China Scholarship Council. Publication costs are funded by National Natural Science Foundation of China (No. 61972299). The funding bodies played no role in the design of the study or in collection, analysis, or interpretation of data, or in writing the manuscript.

Availability of data and materials

The SKEMPI2.0 database is available at (https://life.bsc.es/pid/skempi2/). In the research, part of the core code and data are available at (https://github.com/nsiakjdw/Paper-datasets.git). The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]School of Computer Science and Technology, Wuhan University of Science and Technology, Wuhan, Hubei, China. [2]Hubei Province Key Laboratory of Intelligent Information Processing and Real-Time Industrial System, Wuhan 430065, Hubei, China. [3]Molecular Biology and Biochemistry, Simon Fraser University, Vancouver, BC, Canada.

Received: 2 September 2021 Accepted: 8 September 2021
Published online: 25 October 2021

References

1. Chothia C, Janin J. Principles of protein–protein recognition. Nature. 1975;256(5520):705–8.
2. Clackson T, Wells JA. A hot spot of binding energy in a hormone-receptor interface. Science. 1995;267(5196):383–6.
3. Bogan AA, Thorn KS. Anatomy of hot spots in protein interfaces. J Mol Biol. 1998;280(1):1–9.
4. Xiang L, Keskin O, Ma B, et al. Protein-protein interactions: hot spots and structurally conserved residues often locate in complemented pockets that pre-organized in the unbound states: implications for docking. J Mol Biol. 2004;344(3):781–95.
5. Gul S, Hadian K. Protein–protein interaction modulator drug discovery: past efforts and future opportunities using a rich source of low- and high-throughput screening assays. Expert Opin Drug Discov. 2014;9(12):1393–404.
6. Cukuroglu E, Engin HB, Gursoy A, et al. Hot spots in protein-protein interfaces: towards drug discovery. Prog Biophys Mol Biol. 2014;116(2):165–73.
7. Xia J, Zhao X, Song J, et al. APIS: accurate prediction of hot spots in protein interfaces by combining protrusion index with solvent accessibility. BMC Bioinformatics. 2010;11(1):174–174.
8. Tuncbag N, Gursoy A, Keskin O, et al. Identification of computational hot spots in protein interfaces. Bioinformatics. 2009;25(12):1513–20.
9. Huang Q, Zhang X. An improved ensemble learning method with SMOTE for protein interaction hot spots prediction. Bioinform Biomed. 2016;10:1584–9.
10. Hu S, Chen P, Wang B, et al. Protein binding hot spots prediction from sequence only by a new ensemble learning method. Amino Acids. 2017;49(10):1773–85.
11. Cukuroglu E, Gursoy A, Keskin O, et al. Analysis of hot region organization in hub proteins. Ann Biomed Eng. 2010;38(6):2068–78.
12. Pons C, Glaser F, Fernandezrecio J, et al. Prediction of protein-binding areas by small-world residue networks and application to docking. BMC Bioinform. 2011;12(1):378–378.
13. Nan D, Zhang X. Prediction of hot regions in protein-protein interactions based on complex network and community detection. Bioinform Biomed. 2013;10:17–23.
14. Lin X, Zhang X. Prediction of hot regions in PPIs based on improved local community structure detecting. IEEE/ACM Trans Comput Biol Bioinf. 2018;15(5):1470–9.
15. Moal IH, Fernandezrecio J. SKEMPI: a Structural Kinetic and Energetic database of Mutant Protein Interactions and its use in empirical models. Bioinformatics. 2012;28(20):2600–7.
16. Jankauskaitė J, Jimenezgarcia B, Dapkūnas J, et al. SKEMPI 20: an updated benchmark of changes in protein-protein binding energy, kinetics and thermodynamics upon mutation. Bioinformatics. 2019;35(3):462–9.
17. Ester M, Kriegel H, Sander J, et al. A density-based algorithm for discovering clusters in large spatial Databases with Noise. Knowl Discov Data Min. 1996;10:226–31.
18. Chen T, Guestrin C. XGBoost: A Scalable Tree Boosting System. Knowledge discovery and data mining, 2016: 785–794.
19. Cutler A, Cutler DR, Stevens JR. Random forests. Mach Learn. 2011;45(1):157–76.
20. Chang C, Lin C. LIBSVM: A library for support vector machines. ACM Trans Intell Syst Technol. 2011;2:1–27.
21. Pao Y. Adaptive pattern recognition and neural networks. Reading Addison Wesley, 1989, 12(May), 31–67.
22. Nair V, Hinton GE. Rectified linear units improve restricted Boltzmann machines. In: International conference on machine learning, 2010: 807–814.
23. Keskin O, Ma B, Nussinov R, et al. Hot regions in protein-protein interactions: the organization and contribution of structurally conserved hot spot residues. J Mol Biol. 2005;345(5):1281–94.
24. Miller S, Lesk AM, Janin J, et al. The accessible surface area and stability of oligomeric proteins. Nature. 1987;328(6133):834–6.
25. Pierce MM, Raman CS, Nall BT, et al. Isothermal titration calorimetry of protein-protein interactions. Methods. 1999;19(2):213–21.
26. Wang Y, Shen B, Sebald W, et al. A mixed-charge pair in human interleukin 4 dominates high-affinity interaction with the receptor alpha chain. Proc Natl Acad Sci USA. 1997;94(5):1657–62.
27. Berman HM, Battistuz T, Bhat TN, et al. The protein data bank. Acta Crystallographica Sect D Biol Crystallography. 2002;58(6):899–907.
28. Mihel J, Sikic M, Tomic S, et al. PSAIA—protein structure and interaction analyzer. BMC Struct Biol. 2008;8(1):21–21.
29. Ashkenazy H, Abadi S, Martz E, et al. ConSurf 2016: an improved methodology to estimate and visualize evolutionary conservation in macromolecules. Nucleic Acids Res. 2016;8:W344-350.
30. Peng H, Long F, Ding C, et al. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell. 2005;27(8):1226–38.
31. Chan TF, Golub GH, LeVeque RJ. Updating formulae and a pairwise algorithm for computing sample variances. In: COMPSTAT 1982 5th Symposium held at Toulouse 1982. Physical-Verlag HD; 1982.
32. Hierons R M. Machine learning. Tom M. Mitchell. Published by McGraw-Hill, Maidenhead, U.K., International Student Edition, 1997. ISBN: 0-07-115467-1, 414 pages. Price: U.K. £22.99, soft cover. Software Testing, Verification & Reliability, 1999, 9(3): 191–193.
33. Zhang H. The Optimality of Naive Bayes. The florida ai research society, 2004: 562–567.
34. Python Molecule, https://pymol.org/2/, Accessed 2 May 2020.

Publisher's Note