Journal of Theoretical Biology 365 (2015) 197–203

Contents lists available at ScienceDirect

Journal of Theoretical Biology

journal homepage: www.elsevier.com/locate/yjtbi

Discrimination of acidic and alkaline enzyme using Chou's pseudo amino acid composition in conjunction with probabilistic neural network model

Zaheer Ullah Khan [a], Maqsood Hayat [a,*], Muazzam Ali Khan [b]

[a] Department of Computer Science, Abdul Wali Khan University Mardan, Mardan, KP, Pakistan
[b] College of Electrical and Mechanical Engineering (NUST), Islamabad, Pakistan

HIGHLIGHTS

- We develop an accurate and high throughput predictor for discrimination of acidic and alkaline.
- PseAA composition and SAAC are used as feature extraction schemes.
- Various classification algorithms are utilized.
- Two datasets were evaluated using 10-fold cross validation test.
- Best results are reported so far in the literature.

ARTICLE INFO

Article history:
Received 2 July 2014
Received in revised form
9 September 2014
Accepted 11 October 2014
Available online 22 October 2014

Keywords:
SVM
PNN
DT
K-Nearest neighbor
SAAC

ABSTRACT

Enzyme catalysis is one of the most essential and striking processes among of all the complex processes that have evolved in living organisms. Enzymes are biological catalysts, which play a significant role in industrial applications as well as in medical areas, due to profound specificity, selectivity and catalytic efficiency. Refining catalytic efficiency of enzymes has become the most challenging job of enzyme engineering, into acidic and alkaline. Discrimination of acidic and alkaline enzymes through experimental approaches is difficult, sometimes impossible due to lack of established structures. Therefore, it is highly desirable to develop a computational model for discriminating acidic and alkaline enzymes from primary sequences. In this study, we have developed a robust, accurate and high throughput computational model using two discrete sample representation methods Pseudo amino acid composition (*PseAAC*) and split amino acid composition. Various classification algorithms including probabilistic neural network (*PNN*), *K*-nearest neighbor, decision tree, multi-layer perceptron and support vector machine are applied to predict acidic and alkaline with high accuracy. 10-fold cross validation test and several statistical measures namely, accuracy, *F*-measure, and area under *ROC* are used to evaluate the performance of the proposed model. The performance of the model is examined using two benchmark datasets to demonstrate the effectiveness of the model. The empirical results show that the performance of *PNN* in conjunction with *PseAAC* is quite promising compared to existing approaches in the literature so for. It has achieved 96.3% accuracy on dataset1 and 99.2% on dataset2. It is ascertained that the proposed model might be useful for basic research and drug related application areas.

1. Introduction

Enzymes are biological catalysts, which are proteinaceous in structure. They can only work within a narrow range of temperature and pH. The pH value of underline environment greatly affects the enzyme activity. Whenever, the enzyme pH value is optimum then it is most effective. However, enzymes have a significant role in industrial applications as well as in medical areas, due to profound specificity, selectivity, and catalytic efficiency. Various factors like pH and temperature have a crucial effect on the enzymatic efficiency (Nakhil Nair et al., 2010). However, most of enzymes endure high activity in the pH range between 6 and 8. Many of acidic and alkaline enzymes derived from acidophilic and alkaliphiles make these organisms in order to survive in high acidic (usually at pH 2.0 or below) or

* Corresponding author.
E-mail addresses: zaheerkhan.cs@gmail.com (Z.U. Khan), m.hayat@awkum.edu.pk (M. Hayat).

http://dx.doi.org/10.1016/j.jtbi.2014.10.014
0022-5193/

alkaline conditions (with pH 9–11). Acidophiles and alkaliphiles have more contribution in biotechnology and industrial applications (Jordan et al., 1996; Sarethy et al., 2011).

The stability of acidic and alkaline enzymes has been studied in the biophysical and biotechnological related literature. Therefore, the stability of acidic and alkaline enzymes is essential because instability at extreme pH is one of the main bottlenecks in extending their applications (Dubnovitsky et al., 2005; Geierstanger et al., 1998; Kelch et al., 2007).

In this regard, a series of efforts have been carried out to discriminate acidic and alkaline enzymes. However, the theoretical methods have been achieved considerable success on the basis of primary and secondary enzyme sequences information at amino acids composition level, where sequences of enzymes and particular amino acids are correlated with the external environments of enzymes (Geierstanger et al., 1998; Shirai et al., 1997)

In a sequel, Zhang et al. (2009) have proposed a computational method for predicting acidic and alkaline enzymes. They have utilized random forest algorithm in conjunction with secondary structure amino acid composition. Likewise, Fan et al. (2013) have introduced a new approach for discriminating acidic and alkaline enzymes. Similarly, Lin et al. (2013) have developed a sequence-based method to discriminate acidic enzymes from alkaline enzymes. In this model, the ANOVA was applied to select the high discriminative features derived from g-gap dipeptide compositions and support vector machine was utilized to establish the prediction model. In addition, Chou (2011) has suggested a comprehensive review and published a series of publications for establishing a computational biological predictor (Chen et al., 2014c.; Fan et al., 2014; Guo et al., 2014; Liu et al., 2014; Qiu and Xiao, 2014; Xu et al., 2014). According to the comprehensive review, the first step is to construct or select a valid benchmark datasets. The second step is to formulate the biological samples with an effective mathematical expression that can truly reflect their intrinsic correlation with the target to be predicted. The third step is to introduce or develop a powerful algorithm to operate the prediction; and finally to perform cross-validation tests to objectively evaluate the anticipated accuracy of the predictor.

In spite of tremendous enhancements have been made by pattern recognition and machine learning based approaches to solve this problem, still there exists some room of improvement, which demands for more attention and exploration.

In this concern, we propose a promising computational model to discriminate acidic and alkaline enzymes. In this model, two discrete protein sample representation methods namely *PseAAC* and split amino acid composition (*SAAC*) are used to extract numerical descriptors. *PseAAC* not only computes relative frequency of amino acid but also calculates correlation factors among amino acids. Various classification algorithms are investigated to select the best classification algorithm for this model on the same datasets. A 10-fold cross validation test is applied to assess the performance of proposed model.

The remaining paper is organized as follows: Section 2 describes materials and methods, Section 3 represents results and discussion and finally, conclusion has been drawn in Section 4.

2. Materials and methods

2.1. Benchmark datasets

In order to develop a quite promising computational model, a valid benchmark datasets are required to train the model. In this regards, we have used two benchmark datasets, the first one was originally used by Zhang et al. (2013), who collected and extracted the protein annotation information and sequences from enzyme database BRENDA (Lin et al., 2013) at http://www.brenda-enzyme.info/. In this dataset, enzymes were selected on different criteria, for acidic enzymes the optimal pH below 5.0 and for alkaline enzymes with optimal pH above 9.0. So the original dataset contains 217 enzymes including 105 of acidic enzymes and 112 of alkaline enzymes. Latter, 25% CDHit was applied to remove those sequences from dataset whose identity is more than 25%. Consequently, the second benchmark dataset contains 54 acidic and 68 alkaline enzymes, of total 122 enzymes (Fan et al., 2013; Lin et al., 2013).

2.1.1. Sample representation

In order to extract salient features from protein sequences, one best solution is to formulate or represent all the sequences with an effective strong mathematical expression that enable the sequences to exploit the key correlation with the target to be predicted. Several discrete sample representation methods namely, amino acid composition, dipeptide composition, evolutionary sample representation methods such as position specific scoring matrix, gene ontology, structure representation methods and physicochemical properties of amino acids are used for proteins formulation.

2.2. Pseudo amino acid composition

In this study, we have used pseudo amino acid composition (*PseAAC*) to extract numerical descriptors from enzymes sequences. Primary structure of protein is a polymer of 20 amino acids. However, simple amino acid composition only exhibits the occurrence frequency of each amino acid. As a result only 20 discrete attributes are extracted.

$$P_i = \frac{n_i}{L} \tag{1}$$

$$\sum_{i=1}^{u} p_i = 1 \tag{2}$$

where P_i represents the frequency of each amino acid, i indicates amino acid and L is the length of sequence.

However, each amino acid performs distinct role in formation of protein secondary structure. Therefore, information regarding the location of amino acids and sequence order is essential for discriminating acidic and alkaline enzymes. In order to incorporate correlation factors and sequence order information with simple amino acid composition Chou has introduced the concept of pseudo amino acid composition (*PseAAC*) (Chou, 2001, 2005, 2011). The concept of *PseAAC* has been adopted into almost all the fields of computational proteomics (Du et al., 2014). In addition, it has also penetrated into the area of computational genomics, such as using the pseudo *K*-tuple nucleotide composition (*PseKNC*) to formulate *DNA/RNA* sequences (Chen et al., 2012, 2014a, 2014b, 2014c; Guo et al., 2014; Qiu and Xiao, 2014). Likewise, it was used for other biological samples representation (Huang et al., 2012; Li et al., 2012). Recently three different powerful open access web-servers, called '*PseAAC*-Builder' (Du et al., 2012), 'propy' (Cao et al., 2013), and '*PseAAC*-General' (Du et al., 2014) were established to generate various modes of Chou's *PseAAC*. It can be formulated as

$$P = [p_1, \ldots p_{20}, p_{20+1}, \ldots p_{20+\lambda},]^T \tag{3}$$

where $p_1 \ldots p_{20}$ are the relative frequencies of 20 native amino acids and the remaining are the correlation factors of amino acids determined on the basis of hydrophobicity, hydrophilicity, charge

and polarity (Chou, 2001).

$$\begin{cases} \tau_1 = \frac{1}{L-1}\sum_{i=1}^{L-1} J_{i,i+1} \\ \tau_2 = \frac{1}{L-2}\sum_{i=1}^{L-2} J_{i,i+2} \\ \tau_3 = \frac{1}{L-3}\sum_{i=1}^{L-3} J_{i,i+3} \\ \cdots \\ \tau_\lambda = \frac{1}{L-\lambda}\sum_{i=1}^{L-\lambda} J_{i,i+\lambda} \end{cases} \quad (4)$$

In Eq. (4), L is the length of protein sequence τ_1 is the first rank of correlation factor, τ_2 is the second rank of correlation and τ_λ is the last rank of correlation. After investigation, we have selected the best value of lambda $\lambda=28$ means taking first 28 ranks of sequence-order correlation factors into consideration. In this work, we have utilized four physicochemical properties of amino acid hydrophobicity, hydrophilicity, charge and polarity. Consequently, $(20+112)-D=132-D$ feature space is generated.

2.3. Split amino acid composition

Sometimes the crucial information are concealed in segments, which are not possible to extract through simple composition methods. In order to extract information from segments, split amino acid composition (*SAAC*) is used. In *SAAC*, the protein sequence is divided into different parts and composition of each part is calculated separately (Chou and Shen, 2006a, b). In this work, we have divided the protein sequence into three parts; (i) 25 amino acids of *N* termini, (ii) 25 amino acids of *C* termini, and (iii) region between these two terminus. The resultant feature vector is a 60D instead of 20D as in case of *AAC* (Hayat et al., 2012).

2.3.1. Classification hypothesis

Classification is the sub-parts of machine learning and pattern recognition, where the data is classified into established classes on the basis of attributes. It is also called supervised learning where the targets of these classes are known in advance. In a classification process, the novel instance is classifying on the basis of already known pattern, which is predefined for each class. In this study, we utilized various classification hypotheses in order to select the best one among in the given hypothesis for discriminating acidic and alkaline. Framework of proposed model illustrated in Fig. 1. Detailed description of each hypothesis is mentioned below.

2.4. K-Nearest neighbor (KNN)

KNN is a simple and well-known classification algorithm, which is mostly used in classification, regression estimation, and pattern recognition (Hayat and Khan, 2011, 2012; Zahoor et al., 2008). In spite of its simplicity, *KNN* gives competitive performance without priori assumption about the training samples among various supervised learning algorithms.

It is also known as distance base model. In *KNN* classification, instances are classified on the basis of *K* nearest neighbor in the sample space. It uses Euclidian distance to calculate the distance between the instances. Calculated distance values are sorted in ascending order $d_i \le d_i+1$, $i=1, 2, 3\ldots k$, where k is the number of examples.

$$d(x,y) = \sqrt{\sum_{m=1}^{D}(x_{im}-x_{jm})^2} = = \sqrt{\|x_i\|^2 + \|x_j\|^2 - 2x_{i^t}x_j} \quad (5)$$

$$|\|X_i\|^2 = \sqrt{\sum_{m=1}^{D} x_{im}^2} \quad (6)$$

where $\|X_i\|^2$ is norm of x_i that represents length of vector x.

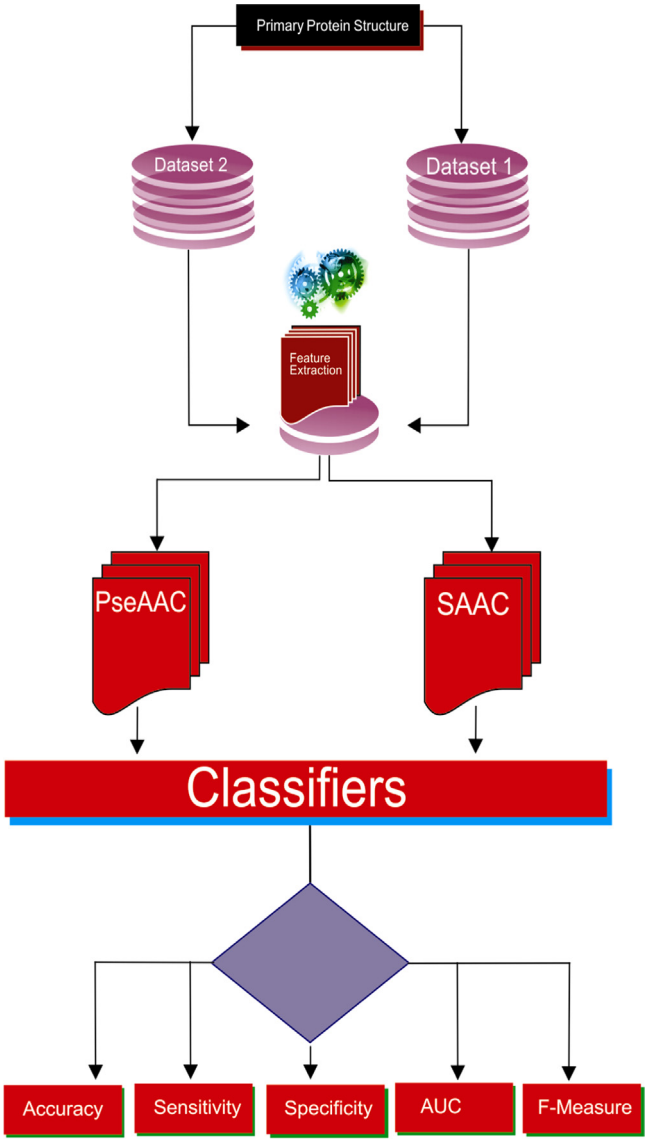

Fig. 1. Framework of proposed model.

After computing distance between training examples and testing example only *K* examples are selected from the feature space, which have minimum distance from the testing example. Consequently, the most frequently occurring class is assigned to the testing example. In case of tie randomly assigns the class.

However, the main issue in *KNN* is the selection of *k* neighbors. Mostly it size is data dependent or heuristic based. Usually, the value of *k* is odd to make ties less likely. In case of small value of *k*, *KNN* comes with small regions for each class and leads to non-smooth decision boundaries and overfitting. While choosing large value of *k* the time complexity of the *KNN* is liner, but computational cost is extensive. However, the discrimination power of Euclidian distance function is poor in case of large attributes number. It creates fever smother regions, as a result underfitting is occurring. So the value of *k* should be chosen between maximum and minimum. It is illustrated in Fig. 2.

2.5. Support vector machine

SVM is a very powerful statistical learning method, based on statistical theory (Hayat and Iqbal, 2014; Jordan et al., 1996; Vapnik, 2005, 1998). It has been extensively applied in the field

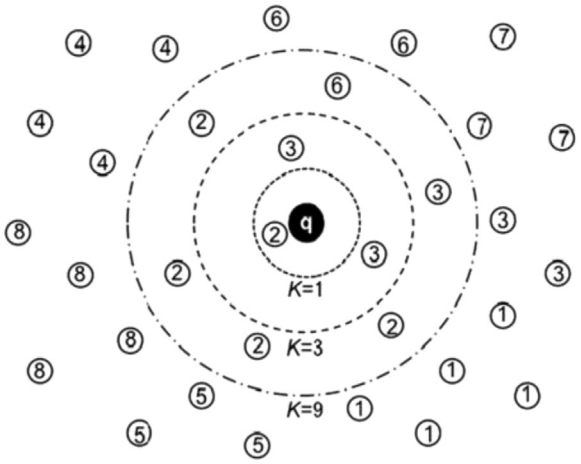

Fig. 2. General structure of KNN.

of bioinformatics, machine learning, and non-linear regression. The basic idea behind the *SVM* is to draw a parallel line to the hyperplane that determines the distance between the dividing line and the closest points in the training dataset; the points are called support vectors and the distance is called margin (Vapnik, 1998). *SVM* utilizes various types of kernel including linear, polynomial, radial basis kernel function (*RBF*), and sigmoid function.

2.6. Decision tree

Decision tree is a predictive model (Hayat and Khan, 2011, 2012; Hayat et al., 2012) that can be viewed as a tree of induction rules. Each branch tip indicates an outcome of the test and class feature variable. Those features with high informative value are placed high up in the tree. Entropy and information gain ration are used for calculating worthy feature set.

$$H(S) = -\sum_{c \in C} P_c \log_2 P_c \tag{7}$$

In the above equation $H(S)$ is the calculated entropy for c classes, whereas the information Gain Ratio are given in the equation below

$$IG(S,F) = H(S) - \sum_{f \in F} \frac{|S_f|}{|S|} H(S_f) \tag{8}$$

where S is the number of elements with F features set.

2.7. Multi-layer perceptron (MLP)

A multi-layer perceptron is a feed forward artificial neural network that accepts a set of input data onto a set of appropriate outputs. *MLP* utilizes a supervised learning technique called back propagation for training the model that is able to classify non-linearly separable data.

MLP consist of 3 or more layers, input layer, output layer and one or more hidden layer. Each node of one layer is connecting with certain weight w^{ij} to every node of the following layer (Rosenblatt, 1961).

2.8. Probabilistic neural network

Probabilistic neural network (*PNN*) is a very powerful data mining tool that is able to adopt and represent any input/output complex relationship (Jyosthna et al., 2012). *PNN* is not only reflect neural network paradigm, but also adapt statistical Bayesian decision rule (Hayat et al., 2012). General structure of *PNN* contains

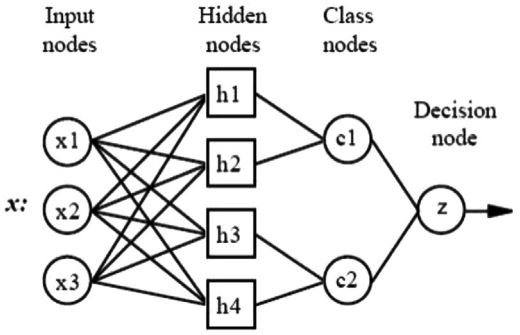

Fig. 3. General structure of PNN.

four layers shown in Fig. 3. The first layer is called input layer, the dimension of the input layer dictated by the dimension (p) of input vector. The second layer is called pattern layer. The dimension of the pattern layer is equal to the dimension of number of examples in the training set. Third layer is summation layer, which has the dimension (k) equal to the number of classes in the set of examples. The fourth layer is called decision layer, which categorize test example into predefined classes.

Suppose an input vector $x=\{x_1,x_2,x_3,\ldots,x_m\}$ is given to *PNN* model, the input layer precedes these m values to next layer. Pattern layer is fully connected to the input layer with one neuran for each pattern in the training set. In each node of the pattern layer, the distance Z_i is calculated between the input vector X_i and target class vector X_t. The calculated distance is then input to the transfer function, and the output of that transfer function is output of the node. Parzen window probability density function estimator with a Gaussian kernel function is used as transfer function. There are many kernel function, but the Gaussian kernel function is most widely used in the machine learning.

The radial base Gaussian function (*RBF*) computes the distance of a point being evaluated to each other points.

$$\text{Weight} = RBF(\text{distance})$$

Sigma values are used to control the influence and spread of *RBF* function. We have used conjugate gradient algorithm to compute the optimum sigma values.

2.9. Validation check methods

In cross validation tests, Jackknife test is deemed the most trusted and objective test in statistical prediction due to its unique results (Chang and Lin, 2011; Chou and Shen, 2008; Hayat and Khan, 2011; Hayat et al., 2013; Xu et al., 2013). Therefore, it has been extensively applied by research community for investigating the performance of their predictors. On the other hand, it is computational expensive because it performs n iterations. In order to incorporate the distinguishable attributes of jackknife test along with minimum computational cost, we have carried out 10-fold cross validation test.

2.10. Waikato environment for knowledge analysis (Weka)

In this study, we have utilized various software including Matlab, Bioweka and Weka 3.7.10. Weka is an open source package, implemented in java. Weka is developed at the University of Waikato New Zealand of machine learning algorithms (Takami and Horikoshi, 2000).

2.11. Evaluation matrices

Sometimes accuracy is insufficient for measuring the performance of predictor, because in case of unbalanced dataset, it bias toward majority class. Therefore, we have used various performance measures to show the strength of proposed model.

$$\text{Sensitivity} = \frac{TP}{TP+FN} \tag{9}$$

$$\text{Specificity} = \frac{TN}{TN+FP} \tag{10}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{11}$$

$$\text{Recall} = \frac{TP}{TP+FN} \tag{12}$$

$$F = 2 \times \left(\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}\right) \tag{13}$$

$$\text{ACC} = \frac{TP+TN}{TP+FN+FP+TN} \tag{14}$$

where *F* stands for *F*-measure, which is the harmonic mean of precision and recall by giving the overall measure of the quality of prediction, and *AUC* (Area under *ROC* curve).

3. Results and discussion

In this work, we have analyzed the performance of several classification algorithms used two feature spaces namely: *PseAAC* and *SAAC*. The empirical results of various classification algorithms are discussed below.

3.1. Prediction performance of classification algorithms using SAAC feature space

The success rates of classification algorithms in conjunction with *SAAC* feature space using dataset1 are reported in Table 1. In this study, we have examined two types of *SVM* such as *C-SVC* and *nu-SVC*. The performances of both types are evaluated on the basis of three kernels namely: linear, *RBF*, and sigmoid, respectively. The empirical results show that the accuracy of *C-SVC* is higher than that of *nu-SVC*. *C-SVC* has achieved the highest accuracy 84.33% using linear kernel, whereas 83.41% and 80.65% in case of *RBF* and sigmoid. On the other hand, *nu-SVC* has obtained the highest accuracy 82.95% using *RBF*. However, the accuracy of *C-SVC* is higher than *nu-SVC* while the *AUC* of *nu-SVC* is better than that of *C-SVC*. On the other hand, the performance of *KNN* is 81.57% accuracy, 86.49% sensitivity, 76.42% specificity, 0.82*F*-measure, and 0.90 *AUC*. Similarly, *MLP* has obtained 85.71%, 87.39%, 83.96%, 0.86, and 0.88, accuracy, sensitivity, specificity, *F*-measure, and *AUC*, respectively. The predicted results of Decision Tree are 84.33% accuracy, 85.59% sensitivity, 83.02% specificity, 0.84 *F*-measure, and 0.86 *AUC*. In contrast, the accuracy, sensitivity, specificity, *F*-measure, and *AUC* of *PNN* are 94.47%, 95.5%, 93.4%, 0.94, and 0.93, respectively. The predicted outcomes of classification algorithms on dataset2 are listed in Table 2. In the case of *C-SVC*, the highest accuracy 78.69% is achieved by linear kernel whereas, in case of *nu-SVC*, both the linear and *RBF* achieved the same accuracy, which is 77.87%. Similarly to dataset1, the accuracy of *C-SVC* is higher than *nu-SVC*, in contrast, the *AUC* of *nu-SVC* is higher than *C-SVC*. *KNN* has yielded 72.95% accuracy, 88.24% sensitivity, 53.7% specificity, and 0.8036 *AUC*. The sensitivity of *KNN* is recorded well whereas the specificity is worse. Likewise, *MLP* has obtained 74.59% accuracy, 88.24% sensitivity, 57.81% specificity, and 0.83 *AUC*. Decision Tree has achieved 71.31% accuracy and 0.80 *AUC*. On the other hand, *PNN* has yielded the highest success rates among using classification algorithms. Its accuracy, sensitivity, specificity, *F-measure*, and *AUC* are 92.62%, 94.12%, 90.74%, 0.93, and 0.89, respectively. On both the datasets, the performance of *PNN* is better than that of other used algorithms.

Table 1
Success rates of classifiers using SAAC feature space of dataset1.

Classifier		Acc (%)	Sn (%)	Sp (%)	*F*-measure	AUC
SVM						
C-SVC	Linear	84.33	88.29	80.19	0.85	0.88
	RBF	83.41	86.49	80.19	0.84	0.89
	Sigmoid	80.65	85.59	75.47	0.82	0.85
nu-SVC	Linear	80.65	82.88	78.30	0.81	0.89
	RBF	82.95	84.68	81.13	0.83	0.91
	Sigmoid	71.43	65.77	77.36	0.70	0.73
KNN		81.57	86.49	76.42	0.83	0.90
MLP		85.71	87.39	83.96	0.86	0.88
Decision tree		84.33	85.59	83.02	0.85	0.86
PNN		**94.47**	**95.5**	**93.4**	**0.94**	**0.93**

Acc=accuracy, Sn=sensitivity, Sp=specificity-measure, AUC=area under ROC curve.

Table 2
Success rate of classifiers using *SAAC* feature space of dataset2.

Classifiers		Acc (%)	Sen (%)	Sp (%)	*F*-measure	AUC
SVM						
C-SVC	Linear	78.69	86.76	68.52	0.8200	0.8300
	RBF	74.41	80.88	68.52	0.7800	0.8300
	Sigmoid	73.41	82.35	62.96	0.7800	0.7400
nu-SVC	Linear	77.87	83.82	70.37	0.8100	0.8500
	RBF	77.87	83.82	70.37	0.8100	0.8500
	Sigmoid	60.66	69.12	50.00	0.6600	0.5900
KNN		72.95	88.24	53.7	0.7843	0.8000
MLP		74.59	88.24	57.41	0.7947	0.8300
Decision tree		71.31	76.47	64.81	0.7482	0.8000
PNN		**92.62**	**94.12**	**90.74**	**0.9343**	**0.9000**

Table 3
Success rates of different classifiers using *PseAAC* composition feature space on dataset1.

		Acc (%)	Sn (%)	Sp (%)	*F*-measure	AUC
SVM						
C-SVC	Linear	81.57	81.28	82.08	0.8182	0.8995
	RBF	86.18	85.59	86.79	0.8636	0.9253
	Sigmoid	82.95	84.68	81.13	0.8356	0.9165
nu-SVC	Linear	59.59	60.36	79.25	0.6700	0.7957
	RBF	84.79	83.78	85.85	0.8493	0.9245
	Sigmoid	70.05	74.77	65.09	0.7186	0.7593
KNN		84.64	88.29	84.91	0.8711	0.9304
MLP		96.2	87.39	84.91	0.8661	0.8844
Decision tree		84.8	85.59	83.96	0.8520	0.8731
PNN		**96.31**	**96.40**	**96.23**	**0.9640**	**0.9521**

3.2. Prediction performance of classification algorithms using PseAAC feature space

The predicted outcomes of various classification algorithms using dataset1 are shown in Table 3. In the case of both the types of *SVM, RBF* kernel has achieved better results compared to other kernels. *KNN* has obtained 84.64% accuracy whereas Decision Tree

Table 4
Success rates of different classifiers using *PseAAC* composition feature space on dataset2.

		Acc (%)	Sn (%)	Sp (%)	*F*-measure	AUC
SVM						
C-SVC	Linear	81.97	88.24	74.07	0.8400	0.8800
	RBF	81.15	88.24	72.22	0.8400	0.8900
	Polynomial	77.05	80.88	72.22	0.8000	0.8500
	Sigmoid	74.59	79.41	68.52	0.7700	0.8400
nu-SVC	Linear	78.69	86.76	68.52	0.8200	0.7700
	RBF	80.33	88.24	70.37	0.8300	0.7900
	Polynomial	80.33	88.24	70.37	0.8300	0.7900
	Sigmoid	80.33	88.24	70.37	0.8300	0.7900
KNN		72.13	73.53	70.37	0.7400	0.7700
MLP		80.33	85.29	74.07	0.8300	0.8700
Decision tree		64.00	64.00	59.00	0.6500	0.6300
PNN		**99.18**	**98.53**	**100**	**0.9900**	**0.9900**

Table 5
Comparison of proposed method with existing methods on dataset1.

Methods	Acc (%)	Sn (%)	Sp (%)	AUC
Zhang et al. (2009)	90.70	88.60	92.80	0.9500
Ours	**96.31**	**96.43**	**96.00**	**0.9500**

has yielded 84.8% accuracy. The performance of *MLP* is 96.2% accuracy, 87.4% sensitivity, 84.9% specificity, 0.86 *F*-measure, and 0.88 *AUC*. Among classification algorithms the highest success rates have been achieved by *PNN*. It has obtained 96.3% accuracy, 96.4% sensitivity, 96% specificity, 0.96 *F*-measure, and 0.95 *AUC*. *PNN* performances on different folds are also shown in supplementary. The success rates of classification algorithms using dataset2 are listed in Table 4. Still, *PNN* has obtained remarkable performance compared to other classification algorithms. It has obtained 99.2% accuracy, 98.5% sensitivity, 100% specificity, 0.99 *F*-measure, and 0.98 *AUC*. The second highest results have been obtained by *MLP*. Its accuracy is 80.33%, sensitivity is 85.29%, specificity is 74.04, *F*-measure is 0.82, and *AUC* is 0.87. In case of *SVM, C-SVC* has generated different results on different kernels. In case of linear it has obtained 82.0% accuracy while it *AUC* is 0.87. Similarly, in case of *RBF* and sigmoid, it has achieved 81.15% and 74.59% accuracy, and 0.88 and 0.84 *AUC*. In *nu-SVC*, linear kernel has achieved 78.7% accuracy whereas *RBF* and sigmoid have obtained similar accuracy 80.33%. *KNN* has yielded 72.13% accuracy and 0.76 *AUC*. Likewise, the performance of Decision Tree is 64.0% accuracy and 0.63 *AUC*.

3.3. Performance comparison with existing methods

A series of efforts have been carried out for discriminating acidic and alkaline enzymes. In order to compare the performance of our proposed model with already existing methods a comparison has been drawn in Table 5 using dataset1. In this application, the pioneer work has been performed by Zhang et al. (2009), who has proposed the first computational model for discriminating acidic and alkaline enzymes. Zhang et al. (2009) developed model has achieved 90.7% accuracy and 0.95 *AUC*. In contrast, our proposed model have yielded 96.3% accuracy while 0.95 *AUC*. The predicted outcomes of our proposed model are 5.6% higher than Zhang et al. (2009) model. The performance comparison of proposed model with existing models using dataset2 is listed in Table 6. In case of dataset2, Fan et al. have developed a computational model for discriminating acidic and alkaline enzymes. Their developed model has achieved 94.0% accuracy and 0.96 *AUC* (Fan et al., 2013). Further, Lin et al. (2013), proposed a model, which has

Table 6
Comparison of proposed method with existing methods on dataset2.

Methods	Acc (%)	Sn (%)	Sp (%)	AUC
Lin et al. (2013)	94.40	94.60	94.30	0.9700
Fan et al. (2013)	94.00	92.40	95.50	0.9600
Ours	**99.18**	**98.53**	**100.00**	**0.9900**

obtained 94.4% accuracy and 0.97 *AUC*. In this regard, our proposed model has yielded 99.2% accuracy and 0.98 *AUC*. Our proposed model obtained 5.2% higher accuracy than that of Fan et al. accuracy, and 4.8% higher than that of Lin et al., accuracy. We know that feature extraction is one of the indispensable and very important task of the whole prediction process, because extraction of discriminative features play a significant role in enhancing the generalization capability of classification algorithms. On the other hand, this performance is also ascribed with *PNN* because the performance of *PNN* generates more accurate predicted target probability scores compared to multilayer perception networks. In addition, it is relative insensitive to outliers and adopting Bayes optimal classification. The empirical results revealed that our proposed model might be a very handy tool in bioinformatics and both in the basic research and drug design. It could be also very helpful in the drug development and structural bioinformatics (Chou, 2004a, 2004b), molecular docking (Wang et al., 2008; Zheng et al., 2007), and also enzyme identification and their classes (Shen and Chou, 2007). This in turn could be highly appreciated and will be extensively welcomed by science community.

Since user-friendly and publicly accessible web-servers are essential for a new prediction method to demonstrate it effectively (Lin and Lapointe, 2013). In this regards a series of papers published very recently (Chen et al., 2014b; Ding et al., 2014; Fan et al., 2014; Guo et al., 2014; Liu et al., 2014; Qiu and Xiao, 2014; Xu et al., 2014), we will make efforts to establish a web-server for the method presented in this paper in near future.

4. Conclusion

In this research study, we have established an effective model for discriminating acidic and alkaline enzyme using two discrete protein sequence representation methods including *PseAAC* and *SAAC*. Various classification algorithms namely: *KNN, SVM, MLP*, Decision Tree, and *PNN* were utilized to select the best classification algorithm among these algorithms for discriminating acidic and alkaline enzymes. Two benchmarked datasets were investigated to show the strength of predictive model. 10-fold cross validation is applied to assess the performance of the proposed model. Among used classification algorithms, *PNN* has achieved outstanding performance in conjunction with *PseAAC* on both datasets. The performance of *PNN* is 96.3% accuracy on dataset1 and 99.2% accuracy on dataset2, respectively. The performance is ascribed to the high discriminative feature extraction method *PseAAC* by using four important physicochemical properties of amino acids and *PNN*. It is anticipated that the proposed model might be very powerful and useful tool in the field of studying enzyme and its adaptation to acidic and alkaline environment.

Appendix A. Supporting information

Supplementary data associated with this article can be found in the online version at http://dx.doi.org/10.1016/j.jtbi.2014.10.014.

References

Cao, D.S., Xu, Q.S., Liang, Y.Z., 2013. Propy: a tool to generate various modes of Chou's PseAAC. Bioinformatics 29, 960–962.

Chang, C.C., Lin, C.J., 2011. LIBSVM: a library for support vector machines. ACM Trans. Intell. Syst. Technol. 2, 21–27.

Chen, W., Lei, T.Y., Jin, D.C., 2014a. PseKNC: a flexible web-server for generating pseudo *K*-tuple nucleotide composition. Anal. Biochem. 456, 53–60.

Chen, W., Feng, P.M., Lin, H., 2014b. iSS-PseDNC: identifying splicing sites using pseudo dinucleotide composition. Biomed. Res. Int., 623149.

Chen, W., Lin, H., Feng, P.M., Ding, C., 2012. iNuc-PhysChem: a sequence-based predictor for identifying nucleosomes via physicochemical properties. PLoS One 7, e47843.

Chen, W., Feng, P.M., Deng, E.Z., Lin, H., 2014c. iTIS-PseTNC: a sequence-based predictor for identifying translation initiation site in human genes using pseudo trinucleotide composition. Anal. Biochem. 14, 281–284.

Chou, K.C., 2001. Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins 43, 246–255.

Chou, K.C., 2004a. Molecular therapeutic target for type-2 diabetes. J. Proteome Res. 3, 1284–1288.

Chou, K.C., 2004b. Review: structural bioinformatics and its impact to biomedical science. Curr. Med. Chem. 11, 2105–2134.

Chou, K.C., 2005. Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics 21, 10–19.

Chou, K.C., 2011. Some remarks on protein attribute prediction and pseudo amino acid composition. J. Theor. Biol. 273, 236–247.

Chou, K.C., Shen, H.B., 2008. Cell-PLoc: a package of web servers for predicting subcellular localization of proteins in various organisms. Nat. Protoc. 3, 153–162.

Chou, K.C., Shen, H.B., 2006a. Predicting eukaryotic protein subcellular location by fusing optimized evidence-theoretic K-nearest neighbor classifiers. J. Proteome Res. 5, 1888–1897.

Chou, K.C., Shen, H.B., 2006b. Hum-PLoc: a novel ensemble classifier for predicting human protein subcellular localization. Biochem. Biophys. Res. Commun. 347, 150–157.

Ding, H., Deng, E.Z., Yuan, L.F., Liu, L., 2014. iCTX-type: a sequence-based predictor for identifying the types of conotoxins in targeting ion channels. Biomed. Res. Int., 286419.

Du, P., Gu, S., Jiao, Y., 2014. PseAAC-general: fast building various modes of general form of Chou's pseudo-amino acid composition for large-scale protein datasets. Int. J. Mol. Sci. 15, 3495–3506.

Du, P., Wang, X., Xu, C., Gao, Y., 2012. PseAAC-builder: a cross-platform stand-alone program for generating various special Chou's pseudo-amino acid compositions. Anal. Biochem. 425, 117–119.

Dubnovitsky, A.P., Kapetaniou, E.G., Papageorgiou, A.C., 2005. Enzyme adaptation to alkaline pH: atomic resolution (1.08 A) structure of phosphoserine aminotransferase from *Bacillus* alkaliphiles. Protein Sci. 14, 97–110.

Fan, G.L., Li, Q.Z., Zuo, Y.C., 2013. Predicting acidic and alkaline enzymes by incorporating the average chemical shift and gene ontology information's into the general form of Chou's PseAAC. Process Biochem. 48, 1048–1053.

Fan, Y.N., Xiao, X., Min, J.L., 2014. iNR-Drug: predicting the interaction of drugs with nuclear receptors in cellular networking. Int. J. Mol. Sci. 15, 4915–4937.

Geierstanger, B., Jamin, M., Volkman, B.F., Baldwin, R.L., 1998. Protonation behavior of histidine 24 and histidine 119 in forming the pH 4 folding intermediate of apomyoglobin. Biochemistry 37, 4254–4265.

Guo, S.H., Deng, E.Z., Xu, L.Q., Ding, H., Lin, H., 2014. iNuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo *k*-tuple nucleotide composition. Bioinformatics 30, 1522–1529.

Hayat, M., Khan, A., 2011. Discriminating outer membrane proteins with fuzzy *K*-nearest neighbor algorithms based on the general form of Chou's PseAAC. Protein Pept. Lett. 11, 411–421.

Hayat, M., Khan, A., 2012. Prediction of membrane protein types by using dipeptide and pseudo amino acid composition based composite features. IET Commun. 6, 3257–3264.

Hayat, M., Iqbal, N., 2014. Discriminating of protein structure classes by incorporating pseudo average chemical shift and support vector machine. J. Comput. Methods Programs Biomed. 116, 184–192.

Hayat, M., Khan, A., Yeasin, M., 2012. Prediction of membrane proteins using split amino acid composition and ensemble classification. J. Amino Acids 42, 2447–2460.

Hayat, M., Tahir, M., Khan, S.A., 2013. Prediction of protein structure classes using hybrid space of multi-bayes profiles and bi-gram probability feature spaces. J. Theor. Biol. 346C, 8–15.

Huang, T., Wang, J., Cai, Y.D., Yu, H., 2012. Hepatitis C virus network based classification of hepatocellular cirrhosis and carcinoma. PLoS One 7, e34460.

Jordan, M.A., McGinness, S., Phillips, C.V., 1996. Acidophilic bacteria—their potential mining and environmental applications. Min. Eng. 9, 169–181.

Jyosthna, D.C.h., Syam Prasad Reddy, B., Vagdhan Kumar, K., Musala Reddy, B., Raja Nayak, N., 2012. ANN approach for weather prediction using back propagation. In: The International Conference on Advances in ICT for Emerging Regions 3, 176-18.

Kelch, B.A., Eagen, K.P., Erciyas, F.P., Humphris, E.L., Thomason., A.R., Mitsuiki, S., 2007. Structural and mechanistic exploration of acid resistance: kinetic stability facilitates evolution of extremophilic behavior. J. Mol. Biol. 368, 870–883.

Li, B.Q., Huang, T., Liu, L., Cai, Y.D., 2012. Identification of colorectal cancer related genes with mRMR and shortest path in protein–protein interaction network. PLoS One 7, e33393.

Lin, H., Chen, W., Ding, H., 2013. AcalPred: a sequence-based tool for discriminating between acidic and alkaline enzymes. PLoS One 8 (10), 75726. http://dx.doi.org/10.1371/journal.pone.0075726.

Lin, S.X., Lapointe, J., 2013. Theoretical and experimental biology in one. J. Biomed. Sci. Eng. (JBiSE) 6, 435–442.

Liu, B., Zhang, D., Xu, R., Xu, J., Wang, X., Chen, Q., 2014. Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection. Bioinformatics 30, 472–479.

Nakhil Nair, U., Carl, A.D., Zhao, H., 2010. Engineering of enzymes for selective catalysis. Curr. Org. Chem. 14, 1870–1872.

Qiu, W.R., Xiao, X., 2014. iRSpot-TNCPseAAC: identify recombination spots with trinucleotide composition and pseudo amino acid components. Int. J. Mol. Sci. 15, 1746–1766.

Rosenblatt, Frank, 1961. Principles of Neurodynamics. Perceptron and the Theory of Brain Mechanisms. Spartan Books, Washington DC.

Sarethy, I.P., Saxena, Y., Kapoor, A., Sharma, M., Sharma, S.K., Gupta, V., 2011. Alkaliphilicbacteria: applications in industrial biotechnology. J. Ind. Microbiol. Biotechnol. 38, 769–790.

Shen, H.B., Chou, K.C., 2007. EzyPred: a top-down approach for predicting enzyme functional classes and subclasses. Biochem. Biophys. Res. Commun. 364, 53–59.

Shirai, T., Suzuki, A., Yamane, T., Ashida, T., Kobayashi, T., Hitomi, J., 1997. High resolution crystal structure of M-protease: phylogeny aided analysis of the high-alkaline adaptation mechanism. Protein Eng. 10, 627–634.

Takami, H., Horikoshi, K., 2000. Analysis of the genome of an alkaliphilic Bacillus strain from an industrial point of view. Extremophiles 4, 99–108.

Vapnik, 2005. Introduction to Data Mining. Addison–Wesley Longman Publishing Co Inc, Boston, MA (©2005).

Vapnik, V., 1998. Statistical Learning Theory. Wiley-Interscience, New York.

Wang, J.F., Wei, D.Q., Chen, C., Li, Y., Chou, K.C., 2008. Molecular modeling of two CYP2C19 SNPs. Protein Pept. Lett. 15 (1), 27–32.

Xu, Y., Ding, J., Wu, L.Y., Chou, K.C., 2013. iSNO-PseAAC: predict cysteine *S*-nitrosylationsites in proteins by incorporating position specific amino acid propensity in to pseudo amino acid composition. PLoS One 8 (8), e55844.

Xu, Y., Wen, X., Shao, X.J., Deng, N.Y., 2014. iHyd-PseAAC: predicting hydroxyproline and hydroxylysine in proteins by incorporating dipeptide position-specific propensity into pseudo amino acid composition. Int. J. Mol. Sci. 15, 7594–7610.

Zahoor, J., Abrar, M., Hussain, D.M.A., 2008. Seasonal to Inter-annual Climate Prediction Using Data Mining KNN Technique. 20. Springer-Verlag, Berlin Heidelberg, pp. 40–51 (IMTIC 2008, CCIS 20, 2008. ©).

Zhang, G., Li, H., Fang, B., 2009. Discriminating acidic and alkaline enzymes using a random forest model with secondary structure amino acid composition. Process Biochem. 44, 654–660.

Zhang, Li, Q., Liang Fan, G., Chun Zuo, Y., 2013. Predicting acidic and alkaline enzymes by incorporating the average chemical shift and gene ontology information's into the general form of Chou's PseAAC. Process Biochem. 48, 1048–1053.

Zheng, H., Wei, D.Q., Zhang, R., Wang, C., Wei, H., Chou, K.C., 2007. Screening for new agonists against Alzheimer's disease. Med. Chem. 3, 488–493.