Vol 3, Issue 4, Oct 2016

Journal of Health Management & Informatics

jhmi.sums.ac.ir

Original Article

Prediction of Protein Thermostability by an Efficient Neural Network Approach

Jalal Rezaeenour[1,*], Mansoureh Yari Eili[2], Zahra Roozbahani[2], Mansour Ebrahimi[3]

Received 18 May 2016 ; Accepted 2 Jul 2016

ABSTRACT

Introduction: Manipulation of protein stability is important for understanding the principles that govern protein thermostability, both in basic research and industrial applications. Various data mining techniques exist for prediction of thermostable proteins. Furthermore, ANN methods have attracted significant attention for prediction of thermostability, because they constitute an appropriate approach to mapping the non-linear input-output relationships and massive parallel computing.

Method: An Extreme Learning Machine (ELM) was applied to estimate thermal behavior of 1289 proteins. In the proposed algorithm, the parameters of ELM were optimized using a Genetic Algorithm (GA), which tuned a set of input variables, hidden layer biases, and input weights, to and enhance the prediction performance. The method was executed on a set of amino acids, yielding a total of 613 protein features. A number of feature selection algorithms were used to build subsets of the features. A total of 1289 protein samples and 613 protein features were calculated from UniProt database to understand features contributing to the enzymes' thermostability and find out the main features that influence this valuable characteristic.

Results:At the primary structure level, Gln, Glu and polar were the features that mostly contributed to protein thermostability. At the secondary structure level, Helix_S, Coil, and charged_Coil were the most important features affecting protein thermostability. These results suggest that the thermostability of proteins is mainly associated with primary structural features of the protein. According to the results, the influence of primary structure on the thermostabilty of a protein was more important than that of the secondary structure. It is shown that prediction accuracy of ELM (mean square error) can improve dramatically using GA with error rates RMSE=0.004 and MAPE=0.1003.

Conclusion: The proposed approach for forecasting problem significantly improves the accuracy of ELM in prediction of thermostable enzymes. ELM tends to require more neurons in the hidden-layer than conventional tuning-based learning algorithms. To overcome these, the proposed approach uses a GA which optimizes the structure and the parameters of the ELM. In summary, optimization of ELM with GA results in an efficient prediction method; numerical experiments proved that our approach yields excellent results.

Keywords: Protein Stability, Primary and secondary structures, Extreme learning machine, Neural networks, Genetic algorithm

►*Please cite this paper as:*
Rezaeenour J, Yari Eili M, Roozbahani Z, Ebrahimi M. Prediction of Protein Thermostability by an Efficient Neural Network Approach. J Health Man & Info. 2016;3(4):102-110.

Introduction

The industrial application of a protein can be severely restricted by low thermostability. Therefore, in the last decade, there has been a growing attention to the study of thermostability of proteins in an attempt to improve the characteristics. This has become a hotspot in protein engineering and design (1-6). To successfully engineer new proteins, we must identify the factors responsible for enzyme thermosability and determine what differentiates thermophiles enzymes from mesophilic proteins (7-10).

Data mining is an important branch of research in the field of information technology and it has valuable application in various fields of biological sciences. Many data mining techniques have been used to predict protein thermostability (11, 12).

In order to comprehend the factors influencing protein thermostability, most researchers have compared homologous mesophilic and thermophilic proteins through studying their protein structures and sequences. Zhang et al. predicted mesophilic and thermophilic proteins using support vector machine (SVM) method and found that some of the dipeptide compositions are important for maintaining protein thermostability (13). Non-charged and hydrophilic residues were found to be critical to protein thermostability through decision tree and other

[1] *Department of Industrial Engineering, University of Qom, Qom, Iran*

[2] *Department of computer Engineering and IT, University of Qom, Qom, Iran*

[3] *Department of Biology, University of Qom, Qom, Iran*

[*]**Corresponding Author:** ***J Rezaeenour ,*** *Assistant Professor, Department of Industrial Engineering, University of Qom, Qom, Iran, E-mail: j.rezaee@qom.ac.ir.*

pattern recognition methods (14). Features of the primary structure of a protein, such as amino acid composition and dipeptide composition, have been shown to be the most important factor for predicting protein thermostability (15, 16). In addition, many studies have suggested that the secondary structural composition of a protein is also an important factor that affects its thermostability (17-19). Gromiha et al. predicted mesophilic and thermophilic proteins with neural networks and found that the charged residues Lys, Arg, and Glu as well as the hydrophobic residues Val and Ile have higher occurrence in thermophiles than mesophiles (20).

ANNs are Machine Learning (ML) algorithms that are frequently used in enzyme science. Over the last twenty years, the use of ANNs has increased rapidly. Furthermore, ANN methods have attracted significant attention for prediction of thermostability, because they constitute an appropriate approach to mapping the non-linear input-output relationships and massive parallel computing (21). However, major challenges imposed by ANNs include the requirement to iteratively tune model parameters, slow response of the gradient-based learning algorithm and the relatively low prediction accuracy compared to more advanced ML algorithms.

Recently, an extensively improved class of ML algorithms, known as Extreme Learning Machine (ELM), was proposed by Haung (22) for training Single hidden-Layer Feed-forward Neural networks (SLFN). In ELM, the hidden nodes are randomly initiated and mixed without iterative tuning. The only free parameters which need to be learned are the connections (or weights) between the hidden layer and output layer. However, ELM tends to have problems when irrelevant or correlated variables are present (23). For this reason, it is proposed in the OP-ELM methodology, to perform a pruning of the irrelevant variables, via pruning of the related neurons of the SLFN built by the ELM (24).

Additionally, Evolutionary Algorithms (EA), such as GA, can perform well for optimization of non-linear complex system. EAs are search and optimization methods based on the principles of natural evaluation and genetics which try to approximate the optimal solution of a given problem (25, 26).

Thus, the objective of this study is to present a new methodology to understand the features contributing to enzymes' thermostability and achieve high accuracy for the prediction. The predictor model used in this study is an optimized ELM algorithm which uses a GA in order to generate the hidden weights and biases; therefore, the algorithm is able to obtain the appropriate number of hidden nodes. Next, the model is trained using a few thousand mesophilic and thermophilic enzymes in order to predict thermal stability directly and accurately. Our data showed that both primary and secondary structural features contributed to the thermostability of the proteins. However, the influence of primary structural features on protein thermostability was more important than that of secondary structural features. Performance of the proposed algorithm has been shown to be comparable to another popular neural network called RBF.

Method

A. Extreme Learning Machine

This section briefly reviews ELM originally proposed by Guangbin Huang (22, 27). The main concept behind ELM lies in the random initialization of the SLFN weights and biases. Therefore, the input weights and biases do not need to be adjusted, which makes it possible to explicitly calculate the hidden layer output matrix and hence the output weights. Fig 1. shows an ELM architecture. Consider a set of M distinct samples (x_i, y_i) with $x_i€R^{d1}$ and $y_i€R^{d2}$; then, a SLFN with N hidden neurons is modeled as the following sum:

$$\sum_{i=1}^{N}\beta_i f\left(w_i^T X_j+b_i\right), \quad 1\le j\le M \qquad (1)$$

With f being the activation function, w_i the input weights, b_i the biases and β_i the output weights. ELM is constructed in a way that it perfectly approximates the given output data:

$$\sum_{i=1}^{N}\beta_i f\left(w_i^T X_j+b_i\right)=y_j, \quad 1\le j\le M \qquad (2)$$

Which writes compactly as HB = Y, with

$$H = \begin{pmatrix} f(w_1X_1+b_1) & \dots & f(w_NX_1+b_N) \\ \vdots & \ddots & \vdots \\ f(w_1X_M+b_1) & \dots & f(w_NX_M+b_N) \end{pmatrix}_{N*M} \qquad (3)$$

$$\beta = \begin{bmatrix} \beta_{11} & \dots & \beta_{1m} \\ \vdots & \ddots & \vdots \\ \beta_{M1} & \dots & \beta_{mM} \end{bmatrix}_{M*m} = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_M^T \end{bmatrix}_{M*m} \qquad (4)$$

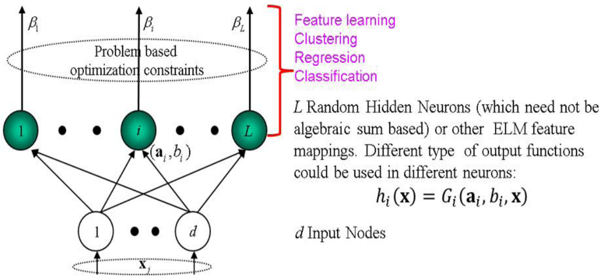
$$T = \begin{bmatrix} t_{11} & \dots & t_{1m} \\ \vdots & \ddots & \vdots \\ t_{M1} & \dots & t_{mM} \end{bmatrix}_{M*m} = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N*m} \qquad (5)$$

H is called the hidden layer output matrix of ELM. The objective function for training ELM is

$$min\|T^{\cdot}-T\| = min\|H\beta-T\| \qquad (6)$$

However, ELM tends to have problems when irrelevant or correlated variables are present (23). For this reason, it is proposed in the OP-ELM methodology to perform a pruning of the irrelevant variables, via pruning of the related neurons of the SLFN built by the ELM (24).

Figure 1. The structure of ELM (23)


β_1
β_i
β_L
Problem based optimization constraints
Feature learning
Clustering
Regression
Classification
1
i
L
$(\mathbf{a}_i, b_i)$
L Random Hidden Neurons (which need not be algebraic sum based) or other ELM feature mappings. Different type of output functions could be used in different neurons:
$h_i(\mathbf{x}) = G_i(\mathbf{a}_i, b_i, \mathbf{x})$
1
d
d Input Nodes
$\mathbf{x}_j$

B. Case Study: Amino Acid Data set

UniProt database (published 2014.02) was searched and protein sequences containing “temperature dependence” item in “general annotation” were downloaded. Incomplete, shorter and non-enzymatic sequences were removed. Similarity of the sequences was calculated, and those with a score larger than 95% were removed. The optimum temperature of a protein was determined based on “temperature dependence” item in protein properties or by reviewing extant literature. Based on their optimum temperature, the proteins were categorized into thermophilic (over 70 °C) or mesophilic (below 70 °C) (28, 29). After filtering, a dataset containing 1289 proteins was obtained, including 342 thermophilic proteins and 947 mesophilic proteins. The basic specifications of the datasets are shown in Table1.

Table 1. Protein features used in this study

	Feature Type	Feature Dimension	Feature
Primary structure	Amino acid composition	20	A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R,S, T,V, W, Y
	Amino acid class composition	10	polar, nonpolar, basic_polar, acidic_polar, all_ polar,neutral,charged, hydrophobic, hydrophilic,large_hydrophilic
	Dipeptide composition	400	AA, AC, AD, AE, AF, AG, AH, AI, AK, AL, AM, AN, AP, AQ, AR, AS, AT, AV, AW,AY,CA, CC, CD, CE, CF, CG,CH, CI, CK,CL, …,YY
Secondary structure		3	Coil, Extended strand, Helix
	Amino acid composition of secondary structure	60	A_Coil, A_Extended strand, A_Helix, C_Coil, C_Extended strand, C_Helix, D__Coil,D_Extended strand, D_Helix, …,Y_Coil, Y_Extendedstrand, Y_Helix
	Amino acid class composition of secondary structure	30	polar_Coil, polar_Extended strand, polar_Helix, …, large_hydrophilic_Coil, large_hydrophilic_ Extendedstrand,large_hydrophilic_ Helix
	Amino acid composition of specific secondary structure	60	Coil_A, Extended strand_A, Helix_A, Coil_C, Helix_C, Extended strand_C, D__Coil, D_Extended strand, D_Helix, …, Coil_Y, Extended strand_Y, Helix_Y
	Amino acid class composition of specific secondary structure	30	Coil_polar, Extended strand_polar, Helix_ polar, …, Coil_large_hydrophilic, Extendedstrand_large_ hydrophilic,Helix_large_hydrophilic

C. Data set Description

Amino acid composition

Amino acid composition is the most classical and most widely used protein feature in bioinformatics applications. The amino acid composition of each protein sequence was calculated using the following formula.

$$Comp(i) = \frac{n_i}{\sum_{i=1}^{20} n_i}, 1 \leq i \leq 20$$

Where i represents the type of amino acid and n_i represents the number of amino acid i contained in protein sequence.

Dipeptide composition

Dipeptide composition is another common protein sequence feature. The dipeptide composition of each protein sequence was calculated using the following formula.

$$Comp_d(i,j) = \frac{n_{ij}}{L-1}, 1 \leq i,j \leq 20$$

Where n_{ij} represents number of dipeptide ij and L represents the length of protein sequence**.**

Amino acid class composition

Amino acids can be divided into four classes based on their polarity (polar, nonpolar, basic polar and acidic polar) or charges (neutral, charged, positive charge and negative charge). Furthermore, they can also be divided into three classes based on their hydrophobicity (hydrophobic if the hydropathy is greater than 0; hydrophilic if the hydropathy index is greater than -2 but less than 0; and highly hydrophilic if the hydropathy index is less than -3). The compositions of these groups of amino acid were then calculated.

Composition of Secondary structure

Protein secondary structure was predicted with software (psipred) (30). The proportions of coil, extended strand and helix were calculated using the following formula.

$$Comp(coil) = \frac{num(coil)}{L}$$

Where num(coil) represents the number of amino acids in coil and L represents the length of protein sequence.

Amino acid composition of secondary structure

The amino acid composition of the secondary structure is the ratio of the number of a specific type of amino acid in the secondary structure to the total amino acid number. For example, the ratio of Ala in the coil to total amino acid was calculated using the following formula.

$$Comp(A\_coil) = \frac{num(A\_coil)}{L}$$

Where num(A_coil) represents the number of Ala in coil and L represents the length of protein sequence.

Amino acid class composition of secondary structure

The amino acid class composition of the secondary structure is the ratio of the total number of a specific amino acid class in the specific secondary structure to the total amino acid number. The formula used to calculate this feature is similar to that used to calculate the amino acid composition of the secondary structure.

Amino acid composition of specific secondary structure

Amino acid composition of a specific secondary structure is the ratio of the number of a specific type of amino acid to the total amino acid number in the specific secondary structure. For example, the ratio of Ala in the coil to total amino acid in the coil was calculated using the following formula.

$$Comp(coil\_A) = \frac{num(coil\_A)}{num(Coil)}$$

Where num(coil_A) represents the number of Ala in coil and num(Coil)represents the number of amino acid in coil.

Amino acid class composition of specific secondary structure

The amino acid class composition of a specific secondary structure is the ratio of the number of a specific class of amino acid to the total amino acid number for a specific secondary structure. The formula used to calculate this feature is similar to amino acid composition of specific secondary structure's formula.

D. Feature Selection

Since only a few parameters are important, feature selection algorithms can create a more manageable dataset of features by eliminating parameters which have no influence on thermostability. In order to investigate features that affect the thermostabilty of an enzyme, Weka 3.6 software suit was used. There were 613 attributes for enzymes. The selected features were classified as either important or unimportant. Several important features were identified, which play a critical role in thermostabilty of an enzyme.

Following the normalization of the dataset, each protein feature gained a value between 0 and 1, revealing the importance of that feature with regards to a target feature. The features with a weights over 0.5 were selected. A total of 50 important features which influence thermostable enzymes were identified. Table 2 shows the most effective features selected by different algorithms. The feature selection algorithms are CFS, Relief, Information Gain, Information gain ratio, and Symmetrical Uncertainty.

Table 2. The most important features selected by different feature selection algorithms

	Feature Type	**Number of features**
Primary structure	E(Glu), Q(Gln) , neutral, charged Polar, EK, basic_polar S(ser), I(Ile), EE, acidic_polar IE, K, IK, KE, C, RE T, KI, hydrophilic KK, V, SA, EI, W, H AA, QQ, LQ, DH, N, CC, hydrophobic, D, YG, AN DE, DD, DA, A, DC, VY	43
Secondary structure	large_hydrophilic, Helix_S, Coil, charged_Coil Coil_E, Coil_A, Coil_polar	7

As can be seen, the influence of primary structure on the thermostabilty of a protein is more important than that of the secondary structure, so thermostability of proteins is mainly associated with primary structural features in protein.

E. Proposed Optimization Methods

In this section, we will present the building blocks of the proposed optimization method, including the representation of the individuals, different operators of variation, and the evaluation processes. The GA used in this paper is devoted to mixed integer optimization problems (31). In (31), the authors proposed a methodology that allows solving optimization problems where the decision variables can be a combination of real, integer, and binary variables.

Initial population and its individuals

GAs work by evolving populations, i.e. sets of solutions, usually named individuals. In order to increase GA flexibility and minimize the cost functions, in the proposed approach, all variables were mapped onto continuous values between 0 and 1. The population npop*nvar population matrix for this GA is represented by

$$p = \begin{bmatrix} v_{1,1} & v_{1,2} & \cdots & v_{1,nvar} \\ v_{2,1} & v_{2,2} & \ddots & \vdots \\ \vdots & & & \\ v_{npop,1} & \cdots & & v_{npop,nvar} \end{bmatrix} \qquad (9)$$

Where v $_{m,n}$=variable n in chromosome m with 0≤v$_{m,n}$≤1. Each row is a chromosome and each chromosome represents one possible solution to the optimization problem.

Evaluation processes

Each chromosome can be evaluated using a fitness function that is specific to the problem being solved as all variables are mapped onto continuous values between 0 and 1. Prior to calculating the fitness of each individual, these values need to be converted into the actual variable values according to the domain of the problem and the corresponding true variable types.

The true value of the l-th variable (m = 1, 2. . .) of individual n (n = 1, 2 . . . m),v$_{m,n}$ is converted to real variablex$_n$, integerI$_n$, or a binary digitb$_n$.

$$x_n = (x_{max} - x_{min})v_{m,n} + x_{min} x_{min} \leq x_n \leq x_{max}$$
$$I_n = rounddown\{(I_{max} - I_{min} + 1)v_{m,n}\} + I_{min} \quad (10)$$
$$b_n = round\{v_{m,n}\}$$

Where min and max represent variable bounds. the rounddown function rounds a value to the next lowest integer, and round is a function that rounds to the nearest integer. The advantage of this approach is that scaling, quantization, and rounding are carried out in the cost function, so the GA operates independent of the variable type and operators can work with any combination of variable types.

The Fitness Function

Following the conversion of variables, the fitness of each individual can be obtained. The fitness function to evaluate a chromosome in the population can be written as:

$$fitness = \psi\ (P_k) \epsilon \mathbb{R} \quad (11)$$

Where the fitness function ψ (•) is specific to the problem being solved. Based on their fitness values, a set of individuals is selected to survive to the next generation while the remaining chromosomes are discarded. The surviving individuals form themating pool and the discarded chromosomes are replaced by new offspring. To select the parents from the mating pool, in this study, tournament selection was used (31). For each parent, five individuals from the mating pool were randomly picked and the individual with best fitness was selected to be the parent. For each pair of parents, two new individuals (offspring) were generated through crossover and mutation. The crossover operation involved producing offspring from the selected parents.

Crossover Operator

In the proposed algorithm, the top 50% of the chromosomes survive to be part of the mating pool. Tournament selection with two chromosomes per tournament was used. Roulette wheel selection with rank ordering would give nearly equivalent results (31). At this point, mating between two selected chromosomes can be done using one of the many different real or binary crossovers. Uniform crossover provides a larger exploration of the cost surface than other approaches (30), so it is selected for the purpose of this algorithm. First, a random binary mask with the same length of the individuals is created. In this approach, only one offspring is created for each pair of parents. So, each offspring receives values of variables from the first or second parent depending on whether the value of the mask bit is zero or one: offspring 1/(2), receives the values from parent 1/(2) if the respective mask bit is one and receives the values from parent 2/(1) if the respective mask bit is zero. Consider the following example:

Parent1	=	p_{1_1}	p_{1_2}	p_{1_3}		p_{1_4}
Parent2	=	p_{2_1}	p_{2_2}	p_{2_3}		p_{2_4}
Mask	=	1	0	1	0	
Offspring1	=	p_{1_1}	p_{2_2}	p_{1_3}		p_{2_4}
Offspring2	=	p_{2_1}	p_{1_2}	p_{2_3}		p_{1_4}

This type of crossover results in a variety of outcomes if the values are binary, but only interchanges ones between chromosomes if the values are integer or continuous. Consequently, the mutation introduces new values within the population of continuous values.

Mutation operator

One approach to mutation is to randomly select variables in the population and replace them with uniform random values. Another approach is to add a random correction factor, which may be created by multiplying each element within a chromosome by a random number (-1≤β$_{rm}$≤1) and multiplying the entire chromosome by a mutation factor(0≤α$_r$≤1).

$$\text{Chrom}^* = rem\{\alpha_r[\beta_{m1}v_{m1}\beta_{m2}v_{m2} \quad \cdots \quad \beta_{m8}v_{m8}] + [v_{m1}v_{m2} \quad \cdots \quad v_{m8}]\} \qquad (12)$$

Optimized-ELM uses the mutation operator to maintain the diversity of the population and prevent the algorithm from being trapped in local minima. For each new offspring, a random number r is generated and $r<r_m$, where r_m is the mutation probability; this offspring is mutated. Mutation acts a 2 step operator. First, a random element of the individual is replaced with a uniform random value within the interval [0, 1]. Being $p_k=[p_{k1},p_{k2}\ ,p_{k3},p_{k4}\ ]^T$ the offspring, if the second chromosome is to be replaced, the mutated chromosome is given by:

$$p_k^1 = \left[p_{k1}\,,p_{k2}^{/}\,,p_{k3}\,,p_{k4}\right]^T \qquad (13)$$

Where $p_{k2}/$ is a new random value within the interval [0, 1]. In a second step, a random adjustment factor is added to the chromosome, whichis obtained by multiplying each elementl within the previously mutated chromosome p_k^1 by a random number ($-1\leq\beta_{k1}\leq1$) and multiplying the resulting chromosome by a mutation factor($-1\leq\eta_k\leq1$) so that:

$$p_k^c = \ \eta_k\beta_{k1}p_{k1},\beta_{k1}p_{k2}^{/},\beta_{k3}p_{k3},\beta_{k4}p_{k4}]^T \qquad (14)$$

Finally, the mutated chromosome is given by:

$$P_k^2 = rem\left(P_k^1 + P_k^c\right) \qquad (15)$$

Where rem is the remainder function (digits to the left of the decimal point are dropped). This type of mutation modifies the entire chromosome rather than a single variable.

Once the best solution is found, all answers need to be converted into their true value (31). As mentioned before, all decision variables are mapped onto real variables within the interval [0, 1]. Binary variables si,i = 1...n, are converted using I_n.Eq.10 while integer variables sλ j,j = 1...h, are converted using I_n.Eq.10. Furthermore, Eq. 14 is utilized to convert the input weights wijand bias bj, considering that the lower and upper bounds are -1 and 1. Finally, the regularization parameter is also converted using Eq. 13, since the lower and upper bounds are 0 and 100.

Experimental results

This section presents experimental results using our model on the protein sequence dataset described in the previous section. Our findings are shown in several tables and plots. The performance of the proposed ELM is compared with that of an ANN by evaluating numerical computations. All simulations were conducted in Matlab R2010b environment running on a PC with a 2.5 GHz Core™ i5 CPU and 6 GB RAM.

In this paper, in order to reduce the number of features and the computational time, the most effective features for thermostability of protein were extracted and selected using feature selection algorithms according to methods defined in section3. Then, these data with 50 features and 1289 samples were trained with ANN and our optimal algorithm, ELM, for prediction. For this, datasets were divided into two parts for training and test purposes. The former was used to train the model, while the latter computed predictions and compared them with original values. 80% of the instances were used for training and 20% for testing purposes.

First, to estimate the accuracy of the prediction model and compare the performance of the models, generalization errors should be calculated. Thus, the proposed ML algorithm was statistically evaluated using the following score metrics or prediction error indicators: Root Mean Square Error (RMSE), and Mean Absolute Percentage Error (MAPE):

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(x_T - x_P)^2} \qquad (16)$$

$$MAPE = \sum_{i=1}^{n}\frac{1}{N}\left|\frac{x_T - x_P}{x_T}\right| (17)$$

Where x_T and x_P are the predicted and observed values, respectively.

A comparison of the performance based on statistical analysis of error of the predicted output with the observed values for each model was conducted. Results of the experiments, training and testing instance, Error Rates, number of hidden nodes of the ELM and ANN are given in Table 4. Also, the performance comparison of RBF with G-ELM, is shown in Table 3. The optimal method produces suitable results in terms of accuracy as well as RMSE error. It is clearly observed that G-ELM testing accuracy is higher than RBF (because $RMSE_{G\text{-}ELM} < RMSE_{RBF}$). It is clearly proved that the ELM produces efficient and suitable results compared to popular ANN predictor.

Fig 2 presents the GA convergence curves of our method from the analysis of the figure. Seen from this figure, the GA converges with sufficient speed. Fig 3 shows the true and the approximated function of the ELM learning algorithm. The red line is Expected values and the blue line is actual values. As shown in Fig 3a, once trained, the model has the capability to estimate the termostability in protein sequences well with satisfactory accuracy. The estimated values of the proposed model are plotted in Fig3b. As can be seen, expected values and actual values are close to each other.

Cross-validation techniques can be used to determine how well the prediction method will work. While a model may minimize the Root-Mean Squared Error on the training data, it can be optimistic in its predictive error. The partitions used in cross-validation help to get a better assessment of a model's predictive performance. Each data set is first randomly divided into a training and test subset for five-fold cross-validation (32). The average $RMSE_{ELM}$ obtained from cross-validation is about 0.08. In Fig 4, we plotted the error rate for each value of k, which helps us to see in what region there might be a minimal error rate.

Table 3. Error rates of ELM

	Number of instance	Number of hidden nodes	RMSE		MAPE
G-ELM	1031 training 258 test	25	Train	Test	0.10031
			0.050	0.041	
RBF	1031 training 258 test	3 layers, each layer 10 neurons	Train	Test	0.1096
			0.923	0.899	

Figure 2. GA convergence curve

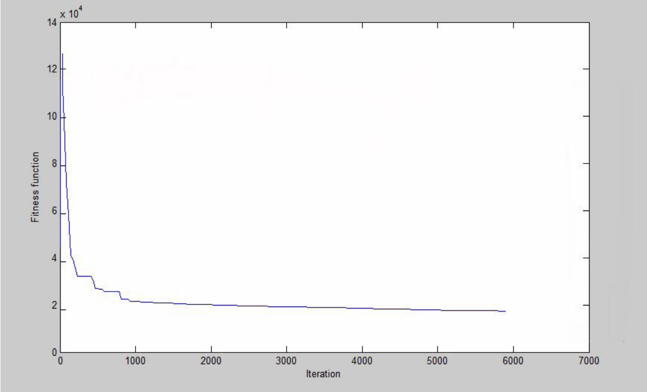


Figure 3. Measured and predicted values by ELM model (3.a shows the training process and 3.b shows the test phase)

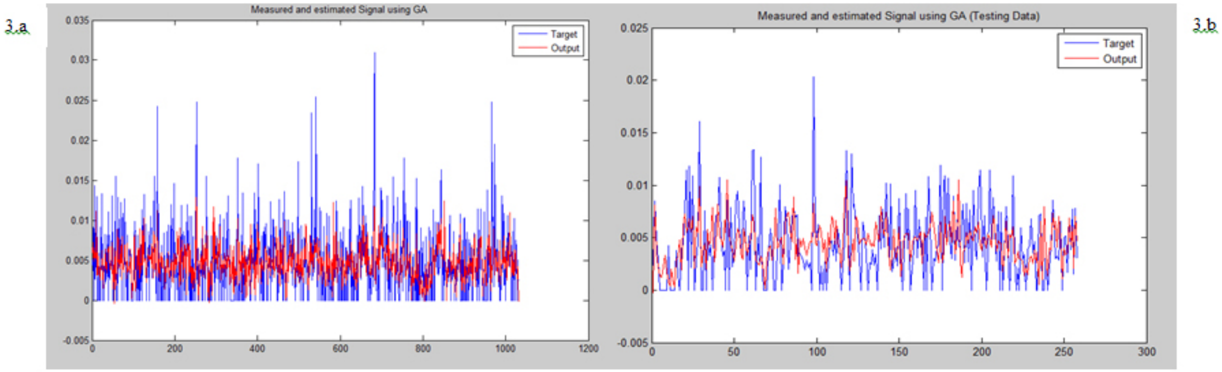
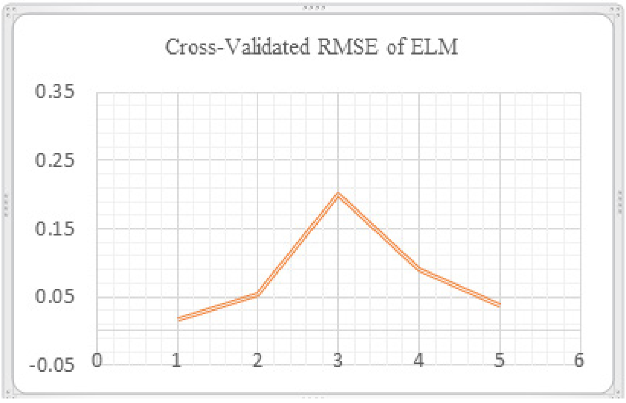


Figure 4. Error rate for each value of k cross validation



From the experimental result analysis, it is clearly apparent that the G-ELM provides superior forecasting performance when tuned with genetic algorithm in comparison to RBF in prediction of protein thermosability ; therefore, it can be considered a viable option to replace ANN for predicting and designing thermostable proteins.

Discussion

Features Contributingto Thermostability

Feature selection algorithms were used to select the features that contribute to protein thermostability and the results of all algorithms (CFS, Relief, Information Gain, Information gain ratio, Symmetrical Uncertainty) were merged into one dataset. After merging the results and removing the duplicate features, a new dataset containing 50 features was obtained (43 primary structure features and 7secondary structural features).

In the CFS algorithm, Gln was selected as the most important feature. In the second level, Glu and polar were selected as the split attributes. Recent studies (14) have shown that the frequency of Gln and Glu is important to protein thermostability; these findings are consistent with the results of this study that Glu and Gln are the main amino acids responsible protein thermostability and the average compositions of Glu in thermophilic proteins are higher than that in mesophilic proteins. It is known that charged residues appear more frequently in thermophiles (20). Glutamic acid is a negatively charged amino acid residue, and it may enhance protein thermostability via forming more hydrogen bond or salt bridge to increase protein structure stability. Thus, protein thermostability can be enhanced by increasing the content of Glu or decreasing the content of Gln in a protein sequence.

The number of protein secondary structural features was lower in the feature selection algorithms and most of them were in the middle or the bottom of the 50 selected features. This suggested that the influence of protein primary structural features on protein thermostability could be more important than the influence of secondary structural features. From the results, Helix_S, was selected as the first, and Coil was selected as the second important attribute.

Performance analysis of G-ELM model

Significant studies have been done in the past for better generalization, faster learning and rate of convergence. But, unfortunately, ELM also suffers with some limitations as outliners, irrelevant variables (in the presence of irrelevant input variables, a reduction of performance is exhibited) and number of hidden nodes (ELM tends to require more neurons in the hidden-layer than conventional tuning-based learning algorithms). To overcome these limitations of ELM, constructive and heuristic approaches have been proposed in various studies. We used a GA which optimized the structure and the parameters of the ELM. In summary, optimization of ELM with GA results in an efficient prediction method;numerical experiments proved that our approach obtains excellent results with MAPE and RMSE values equal to 0.0041 and 0.10031, respectively.

Conclusion and future work

There are a number of challenges in applying ML models in protein engineering. The aim of this paper was to demonstrate the performance of an ANN-based ELM. This study analyzed a large number of protein sequences (1289) with 430 primary structure features and 183 secondary structure features through ELM neural network and genetic algorithms.

We have made a detailed statistical analysis on amino acid composition and found that Gln, Glu and polar were the main amino acids responsible for protein thermostability. Regarding the secondary structural features, Helix_S, Coil, charged_Coil were the most important features with respect to protein thermostability.

The results of the proposed approach for prediction obtained from our models suggested that the primary structural features of a protein may exert a stronger influence on its thermostability than the influence of secondary structural features. This would help the researchers to avoid repeating related experiments on protein secondary structure. Our findings may provide the theoretical support for enhancing the thermostability of proteins for industrial application, such as microbial enzymes to be used in the food industry.

Conflict of Interest

None declared.

References

1. Asial I, Cheng YX, Engman H, Dollhopf M, Wu B, Nordlund P, et al. Engineering protein thermostability using a generic activity-independent biophysical screen inside the cell. Nat Commun. 2013;4:2901.
2. Chitturi B, Shi S, Kinch LN, Grishin NV. Compact Structure Patterns in Proteins. J Mol Biol. 2016 Aug 4.
3. Kumwenda B, Litthauer D, Bishop OT, Reva O. Analysis of protein thermostability enhancing factors in industrially important thermus bacteria species. Evol Bioinform Online. 2013;9:327-42.
4. Meysman P, Zhou C, Cule B, Goethals B, Laukens K. Mining the entire Protein DataBank for frequent spatially cohesive amino acid patterns. BioData Min. 2015;8:4.
5. Movahedi M, Zare-Mirakabad F, Arab SS. Evaluating the accuracy of protein design using native secondary sub-structures. BMC Bioinformatics. 2016;17(1):353.
6. Pucci F, Dhanani M, Dehouck Y, Rooman M. Protein thermostability prediction within homologous families using temperature-dependent statistical potentials. PLoS One. 2014;9(3):e91659.
7. Ebrahimi M, Ebrahimie E, Ebrahimi M, Deihimi T, Delavari A, Mohammadi-dehcheshmeh M. Application of neural networks methods to define the most important features contributing to xylanase enzyme thermostability. CEC 2009: IEEE Congress on Evolutionary Computation. 2009:18-21, 5-2891.
8. Ebrahimi M, Ebrahimie E. Sequence-Based Prediction of Enzyme Thermostability Through Bioinformatics Algorithms. Current Bioinformatics. 2010;5(3):195-203.
9. Satpathy R, Konkimalla V, Ratha J. Propensity based classification: Dehalogenase and non-dehalogenase enzymes. Journal of AI and Data Mining. 2015;3(2):209-15.
10. Zhao W, Wang X, Deng R, Wang J, Zhou H. Discrimination of thermostable and thermophilic lipases using support vector machines. Protein Pept Lett. 2011 Jul;18(7):707-17.
11. Ebrahimie E, Ebrahimi M, Deihimi T, Ebrahimi M. Using neural networks expert system to predict protein thermostability. 2011.
12. Huang L-T, Wu C-C, Lai L-F, Gromiha MM, Wang C-S, Chen Y-R. Data mining application in biomedical informatics for probing into protein stability upon double mutation. Appl Math. 2014;8(1L):125-32.
13. Zhang G, Fang B. Application of amino acid distribution along the sequence for discriminating mesophilic and thermophilic proteins. Process Biochemistry. 2006;41(8):1792-8.
14. Ebrahimi M, Lakizadeh A, Agha-Golzadeh P, Ebrahimie E. Prediction of thermostability from amino acid attributes by combination of clustering with attribute weighting: a new vista in engineering enzymes. PLoS One. 2011;6(8):e23146.

15. Xu J, Chen Y. Discrimination of Protein Thermostability Based on a New Integrated Neural Network. 2011;7062:107-12.
16. Wu L-C, Lee J-X, Huang H-D, Liu B-J, Horng J-T. An expert system to predict protein thermostability using decision tree. Expert Systems with Applications. 2009;36(5):9007-14.
17. Szilágyi A, Závodszky P. Structural differences between mesophilic, moderately thermophilic and extremely thermophilic protein subunits: results of a comprehensive survey. Structure. 2000;8(5):493-504.
18. Vogt G, Woell S, Argos P. Protein thermal stability, hydrogen bonds, and ion pairs. Journal of Molecular Biology. 1997;269(4):631-43.
19. Vogt G, Argos P. Protein thermal stability: hydrogen bonds or internal packing? Folding and Design. 1997;2:S40-S6.
20. Gromiha MM, Suresh MX. Discrimination of mesophilic and thermophilic proteins using machine learning algorithms. Proteins. 2008 Mar;70(4):1274-9.
21. Amini M, Rezaeenour J, Hadavandi E. Effective intrusion detection with a neural network ensemble using fuzzy clustering and stacking combination method. Journal of Computing and Security. 2015;1(4).
22. Huang G-B, Zhu Q-Y, Siew C-K. Extreme learning machine: theory and applications. Neurocomputing. 2006;70(1):489-501.
23. Luo J, Vong CM, Wong PK. Sparse Bayesian extreme learning machine for multi-classification. IEEE Trans Neural Netw Learn Syst. 2014 Apr;25(4):836-43.
24. Matias T, Araújo R, Antunes CH, Gabriel D, editors. Genetically optimized extreme learning machine. 2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA); 2013: IEEE.
25. Marvi H, Esmaileyan Z, Harimi A. Estimation of LPC coefficients using evolutionary algorithms. Journal of AI and Data Mining. 2013;1(2):111-8.
26. Eftekhari M, Eftekhari M, Majidi M. Securing interpretability of fuzzy models for modeling nonlinear MIMO systems using a hybrid of evolutionary algorithms. Iranian Journal of Fuzzy Systems. 2012;9(1):61-77.
27. Huang GB, Chen L, Siew CK. Universal approximation using incremental constructive feedforward networks with random hidden nodes. IEEE Trans Neural Netw. 2006 Jul;17(4):879-92.
28. Brown DK, Militzer W, Georgi CE. The effect of growth temperature on the heat stability of a bacterial pyrophosphatase. Archives of Biochemistry and Biophysics. 1957;70(1):248-56.
29. Lauwers AM, Heinen W. Thermal properties of enzymes from Bacillus flavothermus, grown between 34 and 70 degrees C. Antonie Van Leeuwenhoek. 1983 Jun;49(2):191-201.
30. Jones DT. Protein secondary structure prediction based on position-specific scoring matrices. J Mol Biol. 1999 Sep 17;292(2):195-202.
31. Haupt RL. Antenna design with a mixed integer genetic algorithm. IEEE Transactions on Antennas and Propagation. 2007;55(3):577-82.
32. Gohari M, Baghestani A, Purhosseigholi M, Orooji A. [Evaluation of parametric models with estimation of prediction error by the cross validation method in analyzing survival of colorectal patients]. Razi journal of Medical Science. 2016;23:45.