Chun Li, Jialing Zhao, Changzhong Wang and Yuhua Yao

School of Mathematics and Statistics, Hainan Normal University, Haikou 571158, China; Department of Mathematics, Bohai University, Jinzhou 121013, China; Research Institute of Food Science, Bohai University, Jinzhou 121013, China

Abstract: Aim and Objective: The rapid increase in the amount of protein sequence data available leads to an urgent need for novel computational algorithms to analyze and compare these sequences. This study is undertaken to develop an efficient computational approach for timely encoding protein sequences and extracting the hidden information.

Methods: Based on two physicochemical properties of amino acids, a protein primary sequence was converted into a three-letter sequence, and then a graph without loops and multiple edges and its geometric line adjacency matrix were obtained. A generalized PseAAC (pseudo amino acid composition) model was thus constructed to characterize a protein sequence numerically.

Results: By using the proposed mathematical descriptor of a protein sequence, similarity comparisons among β-globin proteins of 17 species and 72 spike proteins of coronaviruses were made, respectively. The resulting clusters agreed well with the established taxonomic groups. In addition, a generalized PseAAC based SVM (support vector machine) model was developed to identify DNA-binding proteins. Experiment results showed that our method performed better than DNAbinder, DNA-Prot, iDNA-Prot and enDNA-Prot by 3.29-10.44% in terms of ACC, 0.056-0.206 in terms of MCC, and 1.45-15.76% in terms of F1M. When the benchmark dataset was expanded with negative samples, the presented approach outperformed the four previous methods with improvement in the range of 2.49-19.12% in terms of ACC, 0.05-0.32 in terms of MCC, and 3.82-33.85% in terms of F1M.

Conclusion: These results suggested that the generalized PseAAC model was very efficient for comparison and analysis of protein sequences, and very competitive in identifying DNA-binding proteins.

Keywords: Adjacency matrix, Generalized PseAAC, graph, identification of DNA-binding proteins, phylogenetic analysis, protein sequences.

1. INTRODUCTION

DNA-binding proteins (DNA-BPs) are very important functional proteins in a cell. These proteins play vital roles in various cellular processes, including DNA replication, transcription, regulation of gene expression, packaging, and other activities associated with DNA [1-5]. It is therefore substantially important to distinguish DNA-BPs from non-DNA-binding proteins (NBPs). In the past, many experimental and computational techniques have been developed for identifying DNA-BPs. Experimental techniques can provide a clear-cut answer to a query protein. However, the experimental methods are cost-intensive and time-consuming, and thus impractical for large datasets [3-7]. Computational methods can be broadly divided into two categories: structure-based method and sequence-based method. The former can discriminate DNA-binding and non-binding proteins with high accuracy, but these methods can't be employed in high throughput annotation, as they require the structure information of a query protein [1]. Though tremendous progress has been achieved in experimental determination of protein structures in the past five decades, it can't keep pace with the explosive growth of sequence information resulting from modern sequencing technology [8]. Yet as suggested by Anfinsen [9], proteins contain within their amino acid sequences enough information to determine their native conformation. Therefore, it is more promising to use sequence-based methods to identify DNA-BPs.

One of the core issues to the sequence-based methods is how to characterize protein sequences and harvest the fruits hidden in them. The most typical approach is using the amino acid composition (AAC) to formulate a protein sequence. Owing to its simplicity, the AAC model was widely applied in a number of earlier statistic-based methods. However, as pointed out in Ref [6], if we denote by
n_3, n_2, ..., n_20, the counts of 20 standard amino acids in a protein sequence, then we can see that there are a total of \(\left\lfloor \frac{n_1 + n_2 + ... + n_20}{n_3} \right\rfloor\) different sequences/strings possessing the same AAC. The reason is that AAC model neglects the order relation among elements of a sequence. To overcome this drawback, the concept of pseudo amino acid composition (PseAAC, or Chou’s PseAAC) was proposed [10-18]. The essence of PseAAC is that it not only covers AAC, but also contains additional order-correlated factors along a protein sequence. Another popular way for sequence analysis is to convert the protein primary sequence over 20 amino acids into a reduced one. The earliest and simplest reduction was the well-known HP model, in which 20 standard amino acids are divided into two types, hydrophobic (H) (or non-polar) and polar (P) (or hydrophilic). On the basis of the classic model, a detailed HP model was introduced by dividing the polar class into three subclasses: positive polar, uncharged polar and negative polar [19]. In addition, a few five-group classifications of amino acids were presented for practical purposes [20-23]. By considering property-based triples, Li et al. [6] put forward a six-letter model of amino acids. Also based on three physical-chemical properties of amino acids, Yao et al. [24] mapped the 20 standard amino acids to eight vertices of a cube with the center of origin, and thus an eight-group model of amino acids is obtained.

Motivated by the work mentioned above, we propose a generalized PseAAC which is grounded on a three-letter model and 2-D graphical representation of a protein sequence. We summarize the main work of this paper as follows: In section 2, we briefly introduce five datasets used in this study. In section 3, on the basis of two important physicochemical properties of amino acids, we cluster the 20 standard amino acids into three groups. By assigning to each group a representative symbol, we transform a protein sequence into a three-letter sequence. Then a 2-D graph without loops and multiple edges and its geometric line adjacency matrix are obtained. A sequence-derived feature vector of dimension \((25 + \lambda)\) is thus constructed to characterize a protein sequence. Our scheme is similar to, but obviously different from that of PseAAC. In section 4, we apply the presented feature vector to compare \(\beta\)-globin proteins of 17 species and 72 spike proteins of coronaviruses respectively. Also, we develop a SVM (support vector machine) model using the generalized PseAAC to identify DNA-binding and non-binding proteins on three datasets. Experiment results show that the presented method outperforms the existing methods including DNAAbinder [1], DNA-Prot [2], iDNA-Prot [3] and enDNA-Prot [4]. Finally, conclusions are given in section 5.

2. DATASETS

In this study, the following five datasets are used. For convenience, they are denoted by BetaSet, CoVSet, DNASet, DNAeSet and DNAiSet, respectively.

2.1. BetaSet

The dataset called BetaSet is composed of \(\beta\)-globin protein of 17 species: Human (ALU64020), Gorilla (P02024), Chimpanzee (P68873), Cattle (CAA25111), Banteng (BAJ05126), Goat (AAA30913), Sheep (ABC86525), European hare (CAA68429), Rabbit (CAA24251), House mouse (ADD526600), Western wild mouse (ACY03394), Spiny mouse (ACY03377), Norway rat (CAA29887), Opossum (AA330976), Guttata (ACH46399), Gallus (CAA23700), Muscovy duck (CAA33756). This dataset is used to determine the adjustable parameters in a feature vector.

2.2. CoVSet

This dataset consists of 72 spike proteins of coronaviruses (CoVs), 23 of which are MERS-CoVs, and 30 are SARS-CoVs. CoVs can be divided into three groups according to serotypes. Group \(\alpha\) (formerly known as CoV-1) and group \(\beta\) (formerly CoV-2) contain mammalian viruses, while group \(\gamma\) (formerly CoV-3) contains only avian viruses. The name, accession number, and abbreviation of the 72 sequences are listed in Table 1. According to the existing taxonomic groups, sequences 1-5 belong to the first group, sequences 6-8 belong to the third group, and the remainings belong to the second group.

2.3. DNASet

This is a benchmark dataset created in 2007 by Kumar et al. [1]. It contains 396 sequences, 146 of which are DNA-BPs (positive samples), and 250 NBPs (negative samples). In both the positive and the negative sets, the sequence similarity between any two proteins is not more than 25%.

2.4. DNAiSet

This dataset was also generated by Kumar et al. [1] which is based on the work of Wang and Brown [25]. It originally contains 92 DNA-BPs and 100 NBPs. In order to avoid overestimating a given method, those sequences having \(\geq 40\%\) sequence similarity with DNASet were removed by Xu et al. [4], and the final dataset is composed of 82 DNA-BPs and 100 NBPs.

2.5. DNAeSet

As an expanded benchmark dataset, DNAeSet was constructed in 2014 by Xu et al. [4]. According to a sequence filter criteria which is identical to DNASet, they added a number of NBPs to DNASet, and the total number of NBPs is 2125. By removing the sequence which has \(\geq 40\%\) sequence identity with DNAiSet, the current version of DNAeSet has 146 DNA-BPs and 1710 NBPs.

3. METHODS

3.1. Three-letter Sequence of Protein Sequence and its 2-D Graphical Representation

Isoelectric point (pI) and relative distance (RD) are two important physicochemical properties of the 20 standard amino acids [26-28]. Their original numerical values are listed in Table 2. As can be seen from this table, the values of \(p_l\) (isoelectric point) are in the range \([2.97, 10.76]\), while
Table 1. The accession number, name and abbreviation for 72 coronavirus spike proteins.

| No. | Accession number | Virus name/strain                             | Abbreviation |
|-----|------------------|----------------------------------------------|--------------|
| 1.  | CAB91145         | Transmissible gastroenteritis virus, genomic RNA | TGEVG        |
| 2.  | NP_058424        | Transmissible gastroenteritis virus           | TGEV         |
| 3.  | AAK38656         | Porcine epidemic diarrhea virus strain CV777 | PEDVC        |
| 4.  | NP_598310        | Porcine epidemic diarrhea virus               | PEDV         |
| 5.  | BAL45637         | Human coronavirus 229E                        | HCoV-229E    |
| 6.  | AAP92675         | Avian infectious bronchitis virus isolate BJ  | IBVBJ        |
| 7.  | AAS00808         | Avain infectious bronchitis virus strain Ca199 | IBVC         |
| 8.  | NP_040831        | Avain infectious bronchitis virus             | IBV          |
| 9.  | NP_937950        | Human coronavirus OC43                       | HCoV-OC43    |
| 10. | AAK83356         | Bovine coronavirus isolate BCoV-ENT           | BCoVE        |
| 11. | AAL57308         | Bovine coronavirus isolate BCoV-LUN           | BCoVL        |
| 12. | AAA66399         | Bovine coronavirus strain Mebus               | BCoVM        |
| 13. | AAL40400         | Bovine coronavirus strain Quebec             | BCoVQ        |
| 14. | NP_150077        | Bovine coronavirus                           | BCoV         |
| 15. | AAB86819         | Mouse hepatitis virus strain MHV-A99C12 mutant | MHVA         |
| 16. | YP_209233        | Murine hepatitis virus strain JHM             | MHVJHM       |
| 17. | AAF69334         | Mouse hepatitis virus strain Penn 97-1        | MHVP         |
| 18. | AAF69344         | Mouse hepatitis virus strain ML-10            | MHVM         |
| 19. | NP_045300        | Mouse hepatitis virus                       | MHV          |
| 20. | AAU04646         | SARS coronavirus civet007                    | civet007     |
| 21. | AAU04649         | SARS coronavirus civet010                    | civet010     |
| 22. | AAU04664         | SARS coronavirus civet020                    | civet020     |
| 23. | AAV91631         | SARS coronavirus A022                        | A022         |
| 24. | AAV49730         | SARS coronavirus B039                        | B039         |
| 25. | AAP51227         | SARS coronavirus GD01                        | GD01         |
| 26. | AAS00003         | SARS coronavirus GZ02                        | GZ02         |
| 27. | AAP30030         | SARS coronavirus BJ01                        | BJ01         |
| 28. | AAP13567         | SARS coronavirus CUHK-W1                     | CUHK-W1      |
| 29. | AAP37017         | SARS coronavirus TW1                         | TW1          |
| 30. | AAR87523         | SARS coronavirus TW2                         | TW2          |
| 31. | BAC81348         | SARS coronavirus TWH genomic RNA             | TWH          |
| 32. | BAC81362         | SARS coronavirus TWJ genomic RNA             | TWJ          |
| 33. | AAQ01597         | SARS coronavirus Taiwan TC1                  | TaiwanTC1    |
| 34. | AAQ01609         | SARS coronavirus Taiwan TC2                  | TaiwanTC2    |
| 35. | AAP97882         | SARS coronavirus Taiwan TC3                  | TaiwanTC3    |
| 36. | AAP13441         | SARS coronavirus Urbani                      | Urbani       |
| 37. | AAP72986         | SARS coronavirus HSR 1                       | HSR1         |
| 38. | AAQ94060         | SARS coronavirus AS                          | AS           |
| 39. | AAP94737         | SARS coronavirus CUHK-AG01                   | CUHK-AG01    |
| 40. | AAP94748         | SARS coronavirus CUHK-AG02                   | CUHK-AG02    |
| 41. | AAP94759         | SARS coronavirus CUHK-AG03                   | CUHK-AG03    |
| 42. | AAP30713         | SARS coronavirus CUHK-Su10                   | CUHK-Su10    |

(Table 1) Contd....
Therefore, the normalization of these values is needed. Here, we scale them into the interval [0,1] by the formulary below:

\[ P_n^{(i)}(AA_i) = \left( \frac{P_n^{(0)}(AA_i) - \min_{j=1,...,20}[P_n^{(0)}(AA_j)]}{\max_{j=1,...,20}[P_n^{(0)}(AA_j)] - \min_{j=1,...,20}[P_n^{(0)}(AA_j)]} \right)^{1/2}, \]

\( i = 1,2,...,20, \ n = 1,2. \)

The corresponding values are listed in Table 3. The last row in this table gives the average values.

For the i-th amino acid AA_i, if \( P_n^{(i)}(AA_i) \geq \overline{P}_n^{(i)} \), then we label it by “+”, otherwise we will label it by “-”. Similarly, if property \( P_2^{(i)} \) is considered, the second label for amino acid \( AA_i \) can be obtained. In this way, each of the 20 standard amino acids has a label pair. In Table 3, the corresponding labels are also listed. Amino acids with a same label pair are viewed as members of a same group. Thus, the 20 standard amino acids are distributed to the following groups:

\[ G_1 = \{ \text{A,Y,V,Q,M,L,E} \}, \]
\[ G_2 = \{ \text{C,W,S,N,G,F,D} \}, \]
\[ G_3 = \{ \text{H,T,R,P,K} \}. \]

For each group, the first amino acid is used to stand for the group. Thus the three groups have three representative letters, they are A, C and H, respectively. The value for the property of a group is defined as the average value for the property of all members in the group. In the left-hand side of Table 4, we list the corresponding values of the three groups. Obviously, each group can be viewed as a 2-D vector. In order to make the vectors of the three groups have unit length, we further normalize them to be unit vectors, and list
Table 2. The original numerical values for properties of the 20 standard amino acids.

| Amino acid (AA) | pI \(^*\) \((P_{11}^*)\) | RD \(^*\) \((P_{22}^*)\) |
|----------------|----------------|----------------|
| A              | 6.02           | 1889           |
| C              | 5.02           | 3355           |
| D              | 2.97           | 2209           |
| E              | 3.22           | 1812           |
| F              | 5.48           | 1916           |
| G              | 5.97           | 2078           |
| H              | 7.59           | 1507           |
| I              | 6.02           | 1765           |
| K              | 9.74           | 1797           |
| L              | 5.98           | 1822           |
| M              | 5.75           | 1689           |
| N              | 5.42           | 1943           |
| P              | 6.30           | 1720           |
| Q              | 5.65           | 1538           |
| R              | 10.76          | 1697           |
| S              | 5.68           | 2000           |
| T              | 6.53           | 1469           |
| V              | 5.97           | 1680           |
| W              | 5.89           | 2317           |
| Y              | 5.66           | 1787           |

a: taken from [26-28]

Table 3. The scaled values for properties of the 20 standard amino acids.

| AA | \(P_1\) | lable1 | \(P_2\) | Lable2 |
|----|---------|--------|---------|--------|
| A  | 0.3915  | -      | 0.2227  | -      |
| C  | 0.2632  | -      | 1.0000  | +      |
| D  | 0.0321  | -      | 0.3924  | +      |
| E  | 0.3222  | -      | 0.1819  | -      |
| F  | 0.3851  | -      | 0.3229  | +      |
| G  | 0.5931  | +      | 0.0201  | -      |
| H  | 0.3915  | -      | 0.1569  | -      |
| K  | 0.8691  | +      | 0.1739  | -      |
| L  | 0.3864  | -      | 0.1872  | -      |
| M  | 0.3569  | -      | 0.1166  | -      |
| N  | 0.3145  | -      | 0.2513  | +      |
| P  | 0.4275  | +      | 0.1331  | -      |
| Q  | 0.3440  | -      | 0.0366  | -      |
| R  | 1.0000  | +      | 0.1209  | -      |
| S  | 0.3479  | -      | 0.2815  | +      |
| T  | 0.4570  | +      | 0       | -      |
| V  | 0.3851  | -      | 0.1119  | -      |
| W  | 0.3748  | -      | 0.4496  | +      |
| Y  | 0.3453  | -      | 0.1686  | -      |

\(P_n\): 0.3994 0.2283

Table 4. The values for properties of the three groups.

| Group | Representative | \(P_1\) | \(P_2\) | \(P_1\) | \(P_2\) |
|-------|----------------|---------|---------|---------|---------|
| G_{1} | A              | 0.3291  | 0.1478  | 0.9122  | 0.4097  |
| G_{II} | C              | 0.2868  | 0.4193  | 0.5646  | 0.8253  |
| G_{III} | H             | 0.6693  | 0.0896  | 0.9912  | 0.1327  |

Fig. (1). The 2-D map of the 20 standard amino acids.

By substituting each amino acid with its representative letter, a protein primary sequence is reduced into a three-letter sequence. For example, the three-letter sequence of the sequence segment EKAAVTGFWGKVKVDEVGAEA is AHAAAHCCCHAAAAACAAA. To obtain the graphical representation of a reduced sequence, we start from the origin \((0,0)\) and move in \(xoy\)-plane in the direction dictated by Fig. (1). In mathematics, one can let \(S = S_1 S_2 \cdots S_L\) be a given three-letter sequence. And then one has a map \(\emptyset\), which maps \(S\) into a plot set. Explicitly, \(\emptyset(S) = \emptyset(S_1) \emptyset(S_2) \cdots \emptyset(S_L)\), and \(\emptyset\) is given by

\[
\emptyset(S_i) = (x_i, y_i)^T = \left(\sum_{k=1}^{i} S_{1k}^1, \sum_{k=1}^{i} S_{2k}^2\right)^T
\]

where, \(T\) represents the transpose of a matrix, \(S_{jk}^i (j=1,2)\) represents the \(j\)-th component of the unit vector corresponding to \(S_k\) (cf. Fig. 1 and Table 4). Connecting all points of the plot set in turn, a 2-D curve is drawn. In Fig. (2), we show the 2-D graphical representation of sequence AAAAAHCCCHAAHACACAAA. It is not difficult to find that the 2-D graphical representation has no degeneracy, and thus is a simple graph, that is, a graph without loops and multiple edges.
thus the modified Zagreb index has been proposed \[38\]:

Zagreb index is defined as

less than or equal to one. The higher order \(kL/kL\) matrix is the elements are zero. Obviously, the entries in a \(L/L\) matrix are defined as a quotient of the Euclidean distance between two vertices of the graph and the sum of geometrical lengths of edges between the two vertices. By definition all diagonal elements are zero. Obviously, the entries in a \(L/L\) matrix are less than or equal to one. The higher order \(1/L/1\) matrix is the matrix whose \((i,j)\)-entry is \([L/L]_{ij}^k\). As the exponent \(k\) approaches positive infinity, \(1/L/1\) converges to a \((0,1)\) matrix (denoted by \(1/L/1\)). With respect to the proposed 2-D graph, \([1/L/1]_{i,i}=1\) if and only if the two corresponding vertices lie on a straight line in the curve, including the cases of adjacency and non-adjacency. In this sense, we call such a matrix a geometric line adjacency matrix (GLAM), or simply a generalized adjacency matrix (GAM), generated by a graph, and denote it by \(M_G\).

The first Zagreb index is a well-known vertex-degree-based molecular structure descriptor. This index was first time considered by Gutman and Trinajstic about 45 years ago, and since then discussed and used in numerous studies (see \[38-40\] and the references cited therein). The first Zagreb index is defined as

\[Z_{g1} = Z_{g1}(G) = \sum_{u \in V} d_u^2\]  \hspace{1cm} \text{(2)}

where \(d_u\) denotes the degree (=number of first neighbors) of the vertex \(u\) in graph \(G\). If \(G\) is a simple graph (i.e. without loops and multiple edges), \(Z_{g1}\) can be also obtained directly from its adjacency matrix since the row-sums of this matrix are equal to degrees of the corresponding vertices.

It should be mentioned that the Zagreb index gives greater weights to inner vertices and edges than to outer vertices and edges of a graph \[38\]. One way to amend it is to insert inverse values of the vertex-degree into Eq.(2), and thus the modified Zagreb index has been proposed \[38\]:

\[mZ_{g1} = mZ_{g1}(G) = \sum_{u \in V} (d_u^2)^{-1}\]  \hspace{1cm} \text{(3)}

Clearly, \(mZ_{g1}\) gives greater weights to outer vertices/edges than to inner ones in a graph.

At the same time, on the basis of our geometric line adjacency matrix, we can count the vertex-pair with generalized adjacency relationship. It should be noted that, in our case, the 'neighbors' include not only the conventional neighbors, i.e. the first neighbors, but also the second neighbors, the third neighbors, and so on. We call the corresponding number of graph \(G\) a line-adjacency index, and denote it by \(L(G)\). Then we have a graph-based index:

\[f_{g1} = \frac{L(G)}{Z_{g1}(G) + mZ_{g1}(G)}\]

For a symmetric matrix, eigenvalue-based indices, such as the leading eigenvalue \[29-33, 35\] and the graph energy \[17\], are often used as the matrix invariants. Moreover, in our previous paper \[41\], an alternative invariant called ‘ALE-index’ was proposed. The ALE-index is defined by the following formula:

\[\chi = \frac{1}{2} \left( \frac{1}{L} \| \|_{m1} + \sqrt{\frac{L-1}{L} \| \|_{F}} \right)\]  \hspace{1cm} \text{(4)}

where \(L\) is the order of the matrix, \(\| \|_{m1}\) and \(\| \|_{F}\) are the \(m1\)- and \(F\)-norms of a matrix respectively. In order to reduce variations caused by comparison of matrices with different sizes, we consider a normalized ALE-index \(\chi'(M_G) = \frac{\chi(M_G)}{\sqrt{L}}\) instead of \(\chi(M_G)\). For convenience, we denote this matrix-based index by \(f_{g2}\).

In addition, with respect to three-letter sequence \(S = S_1S_2\cdots S_t\), we define a coupling mode function \(g_n(S, k)\) by

\[g_n(S, k) = \sqrt{\sum_{i=1}^{k} \left( P_n(S_i) - P_n(S_{i+k}) \right)^2}, \hspace{0.5cm} (n=1, 2)\]  \hspace{1cm} \text{(5)}

where \(P_1\) and \(P_2\) are values for properties of the corresponding representative letter (group), integer \(k\) represents the counted rank (or tier) of the coupling mode. Then, following the similar procedures in \[10, 11\], we can extract global sequence-order information of the three-letter sequence \(S\) by

\[\theta_1 = \frac{1}{L-1} \times \frac{1}{2} \sum_{n=1}^{2} g_n(S, 1),\]

\[\theta_2 = \frac{1}{L-2} \times \frac{1}{2} \sum_{n=1}^{2} g_n(S, 2),\]

\[\vdots\]

\[\theta_\lambda = \frac{1}{L-\lambda} \times \frac{1}{2} \sum_{n=1}^{2} g_n(S, \lambda),\]

where \(\theta_k (k = 1, 2, \ldots, \lambda)\) is called the \(k\)-th tier correlation factor. Clearly, \(\theta_1\) reflects the coupling mode between the most contiguous elements along three-letter sequence \(S\), \(\theta_2\) is the coupling mode between the second most contiguous, \(\theta_3\) the third most contiguous, and so forth.

Furthermore, if the respective counts of the three representative letters (A, C and H) in sequence \(S\) are \(c_A\), \(c_C\), \(c_H\), respectively, then we can obtain a so-called group composition (GC):
**4. RESULTS AND DISCUSSION**

In this section, we will discuss the use of the generalized PseAAC. As can be seen from Eqs (7) and (8), the present mathematical descriptor contains four uncertain parameters: $\lambda$, $w_1$, $w_2$, and $w_3$. Here $\lambda$ represents the total number of correlation ranks counted (cf. Eq(6)), which is an integer. Generally speaking, the greater the value of $\lambda$, the more sequence-order effects will be incorporated. However, if the value is too large, it might cause the overfitting problem or ‘high dimension disaster’ [15], therefore, we endeavour to limit the value of $\lambda$ to a small integer. In this study, the five datasets (BetaSet, CoVSet, DNASet, DNAeSet and DNAiSet) are arranged into two groups: one contains BetaSet, the other includes the rest. The first group is used for determining the four adjustable parameters, and the second group for testing purpose.

**4.1. Parameter Determination**

According to the method mentioned above, we first associate each of 17 protein sequences in BetaSet with a \( \lambda = 25 + \lambda \) dimensional vector (cf. Eqs (7) and (8)), and then calculate the pair-wise Euclidean distance between any two of the 17 protein sequences via their \( m \)-D vectors. Thus a \( 17 \times 17 \) real symmetric matrix \( D_{17} \) is obtained. Based on the basis of the achieved distance matrix \( D_{17} \), a UPGMA tree is constructed using MEGA4 package. The result will depend on values of the rank $\lambda$ and the three weight factors. It is found that when $\lambda = 6$, $w_1 = 0.6$, $w_2 = 1.13$, and $w_3 = 1.0$, the three non-mammals (Muscovy duck, Gallus and Guttata) form a separate branch and stay outside of the mammals. Moreover, in the subtree of mammals, primate species (Human, Chimpanzee, Gorilla) are grouped closely. Also, rodent species (Norway rat, Spiny mouse, House mouse, Western wild mouse) and lagomorph species (Rabbit, European hare) are situated at independent branches, respectively. While Goat, Sheep, Cattle and Banteng appear to cluster together (Fig. 3). This result is analogous to that reported in the literature [6, 29, 30, 35, 36]. Accordingly, the four numerical values are respectively used for the four uncertain parameters, and a 31-D feature vector is thus obtained.

**Fig. (3).** The relationship tree of 17 species.

**4.2. Test I: Phylogenetic Analysis of Coronavirus Spike Proteins**

In order to evaluate the effectiveness of our method, we test it by phylogenetic analysis on the CoVSet dataset. Coronaviruses (CoVs) belong to the genus Coronavirus of family Coronaviridae [42]. The first coronavirus (HCoV-229E) was isolated from humans in 1965. Until 2003, coronaviruses attracted little interest beyond causing mild upper respiratory tract infections. However, this phenomenon changed dramatically with the emergence of SARS-CoV and MERS-CoV. As of July 2017, 2040 laboratory-confirmed cases of MERS-CoV infection were reported in over 27 countries, and at least 710 individuals have died (crude CFR 34.8%) [43].

Using the above-determined values for parameters $\lambda$, $w_1$, $w_2$, and $w_3$, we calculate the 31-D feature vectors of 72 coronavirus spike proteins and their Euclidean distance matrix; then the corresponding phylogenetic tree (Fig. 4) is constructed. Observing Fig. (4), we find that the 72 coronavirus spike proteins are clustered into three groups: one contains the five *alpha* coronaviruses (PEDV, PDEV, TGEVG, TGEV, and HCoV-229E), the second includes the three *gamma* coronaviruses (IBV, IBVBJ, IBVC), and the third corresponds to the group *beta*. A closer look at the subtree of *beta* coronaviruses shows that MERS-CoVs are clearly clustered together, so it is with SARS-CoVs, while MIVH, MIVMA, MIVMH, MIVHPJH, BCoV, BCoVE, BCoVI, BCoVM, BCoVQ and HCoV-OC43 are situated at an independent branch. The resulting cluster agrees well with the established taxonomic groups.
4.3. Test II: Identification of DNA-binding Proteins

To further assess the effectiveness of the proposed method, we conduct a series of experiments of identification of DNA-binding proteins on three datasets: DNASet, DNAeSet and DNAiSet. Among them, DNASet serves as training datasets, while DNAiSet serves as an independent testing dataset.

Support vector machine (SVM) is employed as the classifier, and R package ‘e1071’ v1.6-8 [44] is used to implement SVM. For a given set of binary-labeled training examples, SVM maps the input space into a higher-dimensional space and seeks a hyperplane to separate the positive samples from the negative ones [25]. The optimal hyperplane maximizes the separation margin between the two classes of training data. The distance measurement between the data points in the high-dimensional space is defined by the kernel function. In this study, we use the radial basis function (RBF) kernel $K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2}$. This model involves two tunable parameters: the kernel width $\gamma$ and the penalty parameter $C$. Prediction performance can be assessed using some quality indices including Accuracy (ACC), Sensitivity (Se), Specificity (Sp), F-measure (F1M) and Matthews correlation coefficient (MCC) [2, 4, 5, 25, 37, 45]:

$S_e = \frac{TP}{TP+FN}$,

$S_p = \frac{TN}{TN+FP}$,

$ACC = \frac{TP+TN}{TP+TN+FP+FN}$,

$MCC = \frac{TP\times TN - FP \times FN}{\sqrt{(TP+FN)(TP+FP)(TN+FP)(TN+FN)}}$,

$P = \frac{TP}{TP+FP}$,

$R = \frac{TP}{TP+FN}$,

$F1M = \frac{2\times P \cdot R}{P + R}$,

where TP, TN, FP, and FN are defined as the numbers of true positive, true negative, false positive, and false negative samples obtained from the prediction respectively, while P and R denote Precision value and Recall value, respectively. One can also use the alternative definition by a series of studies published recently [15, 46-48]. The higher the values of these measurements, the better the quality of prediction.

4.3.1. Predictive Performance on Benchmark Dataset

This experiment is made on DNASet itself. To obtain a reliable result with few error, the SVM model on DNASet is established by 5-fold cross-validation (5CV) with 3 runs. Here the 31-D feature vector of a protein sequence serves as the input for SVM. In a 5CV, the positive and negative samples are randomly distributed into five subsets or the so-called folds, and the test is repeated five times. In each of the five iterations, one subset is used as the testing set, while the remaining four subsets are combined together and used to build a classifier (training). The predictions made for the test data instances in all the five iterations yield the final result. The sensitivity, specificity, ACC, MCC and F1M are calculated for each run, and the corresponding results and their average values are listed in Table 5. As can be seen...
from this table, we achieve the accuracy (ACC) of 89.65%, with MCC of 0.776 and F1M of 84.91%. This result shows that our SVM model performs well on the benchmark dataset DNASet.

| Test | 1       | 2       | 3       | Average |
|------|---------|---------|---------|---------|
| Se(%)| 78.77   | 78.77   | 79.45   | 79.00   |
| S_p(%)| 96.00   | 96.00   | 95.60   | 95.87   |
| Acc(%)| 89.65   | 89.65   | 89.65   | 89.65   |
| MCC  | 0.7761  | 0.7761  | 0.7758  | 0.776   |
| F1M(%)| 84.87   | 84.87   | 84.98   | 84.91   |

4.3.2. Predictive Performance on Blind Dataset

It is important to examine the performance of the newly developed method on an independent dataset. In this experiment, we establish the classifier with the benchmark dataset DNASet and then test it on the independent dataset DNAiSet. To decide the parameter pair $(γ, C)$, we utilize a systematic grid search for $γ = 2^i$ and $C = 2^j$, where integers $i$ and $j$ are in ranges $[-3, 3]$ and $[0, 3]$, respectively. It is found that $γ = 0.125$ and $C = 2$ are the optimal values for DNASet. With the best pair $(γ, C)$, DNAiSet is fed to the SVM. As a result, our model correctly predicts 68 out of 82 DNA-BPs and 92 out of 100 NBPs. The ACC arrives at 87.91%, with the MCC, sensitivity, specificity, and F1M of 0.756, 82.93%, 92.00% and 86.07%, respectively (see Table 6). This demonstrates that our SVM model performs equally well on independent dataset.

For convenience of comparison, results of some existing methods including DNAbinder [1], DNA-Prot [2], iDNA-Prot [3] and enDNA-Prot [4] are also listed in Table 6. DNAbinder developed by Kumar et al. [1] can extract evolutionary information in form of position specific scoring matrix (PSSM) from the corresponding protein sequence. PSSM-21 and PSSM-400 are two feature vectors generated by means of PSSM, whose dimensions are 21 and 400, respectively. In [1], PSSM-400 based SVM model was mainly used for predicting DNA-BPs. DNA-Prot [2] is a Random Forest based method, in which the feature vector includes sequence information and structure information, such as the composition of 20 standard amino acids, composition of 10 amino acid groups, and secondary structure information predicted from a protein sequence. iDNA-Prot [3] constructs the feature vector via the grey model, and Random Forest is also used as the operation engine. EnDNA-Prot [4] is a predictor which encodes a protein sequence into a feature vector with dimension of 188 and adopts an ensemble classifier constructed with four types of machine learning classifiers. All these methods are tested on the same datasets to make an unbiased comparison with our method. Observing Table 6, we can see that the current approach outperforms other methods by 3.29-10.44% in terms of ACC, 0.056-0.206 in terms of MCC, and 1.45-15.76% in terms of F1M. This result indicates that our method achieves highly comparable performance.

### Table 6. Performance of different methods (trained on DNASet and tested on DNAiSet).

| Method          | ACC(%) | MCC  | F1M(%) | Se(%) | Sp(%) |
|-----------------|--------|------|--------|-------|-------|
| This work       | 87.91  | 0.756| 86.07  | 82.93 | 92.00 |
| DNAbinder(PSSM-21) | 79.00  | 0.61 | 70.31  | 54.87 | 98.08 |
| DNAbinder(PSSM-400) | 80.11  | 0.62 | 72.73  | 58.53 | 97.97 |
| DNA-Prot        | 84.61  | 0.69 | 81.08  | 73.17 | 94.00 |
| iDNA-Prot       | 77.47  | 0.55 | 75.73  | 78.05 | 77.00 |
| enDNA-Prot      | 84.62  | 0.70 | 84.62  | 73.18 | 94.00 |

4.3.3. Impact of the Number of Negative Samples

When the size of positive samples is comparable to that of negative samples, many machine learning algorithms should have better performance. However, in real life, the number of non-binding proteins is much greater than that of DNA-BPs, i.e.,

$$TN + FP \gg TP + FN.$$  \hspace{1cm} (10)

In this case, the frequency of NBPs is generally much greater than that of the binding ones in the predictions, that is,

$$TN \gg FP.$$ \hspace{1cm} (11)

Eqs (10) and (11) lead to that the value of ACC defined by Eq (9) tends towards 1. To solve this problem, instead of using the definition of ACC in Eq (9), here we use the alternative definition [49, 50]:

$$\text{acc} = \frac{1}{2} (S_e + S_p).$$ \hspace{1cm} (12)

In order to analyze the influence of the number of negative samples in a benchmark dataset on the predictive performance of the current method, we construct a series of subsets $(S_k)$ of DNAeSet and use them as training set in turn, while DNAiSet is always used as the testing set. Each subset $S_k$ contains all the 146 DNA-BPs and a part of NBPs in DNAeSet. In detail, if the set of NBPs in $S_k$ is denoted by $S^-_k$, $k=1, 2, ..., 30$, then $S^-_1$ consists of 250 NBPs randomly selected from DNAeSet. And $S^-_{k+1}$ is obtained by adding 50 NBPs to $S^-_k$, until 1700 NBPs are contained in it. For each subset $S_k$, $k=1, 2, ..., 30$, we develop the SVM model by 5CV with 3 runs. The results averaging over the three runs are given in Fig. (5). From Fig. (5) we can see that the curves of ACC and acc visibly split with each other when $n$, the size of $S^-_1$, is larger. With increasing of $n$, ACC increases rapidly, while acc tends to be steady. The value of ACC seems higher and higher on the surface, but it cannot correctly reflect the performance because it is nothing but a false appearance.

In order to show the advantage of their method, Xu et al. [4] created a dataset called expanded benchmark dataset1100 with all the 146 positive samples and 1100 negative samples in DNAeSet, which is employed as another training dataset to evaluate the predictive performance on the independent dataset DNAiSet. For convenience of comparison, we also
select the expanded benchmark dataset S_{18} (n = 1100) to establish the classifier and test it on DNAiSet. Repeating this procedure five times, the average results are given in Table 7 (the first row). Results obtained by the other four methods (DNAbind, DNA-Prot, iDNA-Prot and enDNA-Prot) trained on the expanded benchmark dataset with n=1100 are also listed in Table 7. From this table we see that the overall accuracy of our method is about 92%, with MCC of 0.84 and F1M of 91.24%, which outperforms other methods with improvement in the range of 2.49-19.12% in terms of ACC, 0.05-0.32 in terms of MCC, and 3.82-33.85% in terms of F1M. This suggests that our method performs well on unbalanced datasets.

CONCLUSION

Based on two important physicochemical properties, 20 standard amino acids were distributed into three groups, and to each of which a representative symbol was assigned. By replacing each amino acid with its representative letter, a protein primary sequence was converted into a three-letter sequence, which can be viewed as a coarse-grained description of the protein primary sequence. On the basis of the three-letter sequence, a graph without loops and multiple edges was obtained. By taking the advantage of the 2-D graph, we constructed a geometric line adjacency matrix (GLAM) and then the corresponding ALE-index, the line-adjacency index, the first Zagreb index and its modification were calculated. In addition, λ order-correlated factors were extracted via the reduced sequence. By combining these elements with the frequencies of occurrence of 20 standard amino acids and their three representative letters, a generalized PseAAC model of a protein sequence was constructed. On five popular datasets, the proposed method was tested by phylogenetic analysis and identification of DNA-binding proteins. The results illustrated the better performance of our method.

CONSENT FOR PUBLICATION

Not applicable.

CONFLICT OF INTEREST

The authors declare no conflict of interest, financial or otherwise.

ACKNOWLEDGEMENTS

The authors' greatest gratitude goes to the anonymous referees for their insightful suggestions and generous support.

The authors are also indebted to the previous programs: the Natural Science Foundation of Liaoning Province (201602005), the Program for Liaoning Innovative Research Team in University (LT2014024), and the National Natural Science Foundation of China (61762035).

REFERENCES

[1] Kumar, M.; Gromiha, M.M.; Raghava, G.P.S. Identification of DNA-binding proteins using support vector machines and evolutionary profiles. BMC Bioinformatics. 2007, 8, 463.

[2] Kumar, K.K.; Pugalenthi, G.; Suganthan, P.N. DNA-prot: identification of DNA binding proteins from protein sequence information using random forest. J. Biomol. Struct. Dyn., 2009, 26(6), 679-686.

[3] Lin, W.Z.; Fang, J.A.; Xiao, X.; Chou, K.C. iDNA-prot: identification of DNA binding proteins using random forest with grey model. PLoS One. 2011, 6(9), e24756.

[4] Xu, R.F.; Zhou, J.Y.; Liu, B.; Yao, L.; He, Y.; Zou, Q.; Wang, X. enDNA-Prot: identification of DNA-binding proteins by applying ensemble learning. Biomed. Res. Int., 2014, 2014, 294279.

[5] Zhang, Y.P.; Wuyunqijie; Zheng, W.; Liu, S.Y.; Zhao, C.G. gDNA-Prot: predict DNA-binding proteins by employing support vector machine and a novel numerical characterization of Protein sequence. J. Theor. Biol., 2016, 406, 8-16.

[6] Li, C.; Li, X.Q.; Lin, Y.N. Numerical characterization of protein sequences based on the generalized Chou’s pseudo amino acid composition. Appl. Sci., 2016, 6(12), 406.

[7] Liu, Z.C.; Shino, D.S.; S洛克ouhim, M.; Lee, K.N.; Yoo, B.W.; Kim, Y.K.; Lee, Y.S. Light-directed synthesis of peptide nucleic acids (PNAs) chips. Biosens. Bioelectron. 2007, 22, 2891-2897.

[8] Marks, D.S.; Hopf, T.A.; Sander, C. Protein structure prediction from sequence variation. Nat. Biotechnol., 2012, 30(11), 1072-1080.

[9] Anfinsen, C.B. Principles that govern the folding of protein chains. Science, 1973, 181, 225-230.

[10] Chou, K.C. Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins: Struct. Funct. Genet., 2001, 43, 246-255.

[11] Chou, K.C. Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics, 2005, 21, 10-19.
[12] Chou, K.C. Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology. *Curr. Proteomics*, 2009, 6, 262-274.

[13] Tahiri, M.; Hayat, M. iNuc-STNC: a sequence-based predictor for identification of nucleosome positioning in genomes by extending the concept of SAAC and Chou's PseAAC. *Mol. Biosyst.*, 2016, 12, 2587-2593.

[14] Chen, W.; Zhang, X.; Brooker, J.; Lin, H.; Zhang, L.; Chou, K.C. PseKNC-General: a cross-platform package for generating various modes of pseudo nucleotide compositions. *Bioinformatics*, 2015, 31, 119-120.

[15] Chen, W.; Feng, P.M.; Lin, H.; Chou, K.C. iSpot-PseDNC: Identify combination spots with pseudo dinucleotide composition. *Bioinformatics*, 2013, 41, e68.

[16] Li, L.Q.; Yu, S.J.; Xiao,W.D.; Li, Y.S.; Huang, L.; Zheng, X.Q.; Zhou, S.W.; Yang, H. Sequence-based identification of recombination spots using pseudo nucleic acid representation and recursive feature extraction by linear kernel SVM. *BMC Bioinformatics*, 2014, 15, 340.

[17] Yu, L.L.; Zhang, Y.S.; Gutman, I.; Shi, Y.T.; Dehmer, M. Protein sequence comparison based on physicochemical properties and the position-feature energy matrix. *Sci. Rep.*, 2017, 7, 46237.

[18] He, P.A.; Tao, H.; Ma, T.T.; Dai, Q.; Yao, Y. A Novel protein characterization based on pseudo amino acids composition and star-like graph topological indices. *Comb. Chem. High Throughput Screen.*, 2017, 20(4), 328-337.

[19] Yu, Z.G.; Anh, V.; Lau, K.S. Chaos game representation of protein sequences and its application. *Nature Struct. Biol.*, 1999, 6: 1033-1038.

[20] Wang, J.; Wang, W. A computational approach to simplifying the protein folding problem. *Nature Struct. Biol.*, 2004, 11, 613-616.

[21] Li, C.; Wang, W. Modeling study on the validity of a possibly simplified representation of proteins. *Phys. Rev. E.*, 2005, 71, 031904.

[22] Yin, H.; Chen, Y.; Wang, W. A new invariant of DNA sequences. *J. Chem. Inf. Sci.*, 2004, 44, 1577-1580.

[23] Yao,Y.H.; Yan, S.; Han, J.; Dai, Q.; He, P.A. A novel descriptor of protein sequences and its application. *J. Theor. Biol.*, 2014, 347, 1069-1117.

[24] Wang, L.J.; Brown, S.J. BindN: a web-based tool for efficient prediction of DNA and RNA binding sites in amino acid sequences. *Nucleic Acids Res.*, 2006, 34, W243-W248.

[25] Grantham, R. Amino acid difference formula to help explain protein. *Science, 1974*, 185, 862-864.

[26] Ma, F.; Wu, Y.T.; Xu, X.F. Correlation analysis of some physical chemistry properties among genetic codons and amino acids. *J. Anhui Agric. Univ.*, 2003, 30, 439-445.

[27] Li, C.; Wang, J.; Zhang, Y.; Wang, J. Similarity analysis of protein sequences based on the normalized relative entropy. *Comb. Chem. High Through Screen.*, 2008, 11, 477-481.

[28] Randic, M.; Vracko, M.; Nandy, A.; Basak, S.C. On 3-D graphical representation of DNA primary sequences and their numerical characterization. *J. Chem. Inf. Comput. Sci.*, 2000, 40, 1235-1244.

[29] Randic, M.; Vracko, M.; Lers, N.; Plavšić, D. Analysis of similarity/dissimilarity of DNA sequences based on novel 2-D graphical representation. *Chem. Phys. Lett.*, 2003, 371, 202-207.

[30] Randic, M.; Novic, M.; Plavšić, D. Milestones in graphical bioinformatics. *Int. J. Quantum Chem., 2013*, 113, 2413-2446.

[31] Randic, M.; Zupan, J.; Balaban, A.T.; Vikić-Topić, D.; Plavšić, D. Graphical representation of proteins. *Chem. Rev.*, 2011, 111, 790-862.

[32] Novic, M.; Randic, M. Representation of proteins as walks in 20-D space. *SAfR. QSAIR. Environ. Res.*, 2008, 19, 317-337.

[33] Li, C.; Yang, Y.; Jia, M.D.; Zhang, Y.; Yu, X.; Wang, C. Phylogenetic analysis of DNA sequences based on k-word and rough set theory. *Physica A*, 2014, 398, 162-171.

[34] Randic, M.; Guo, X.F.; Basak, S.C. On the characterization of DNA primary sequences by triplet of nucleic acid bases. *J. Chem. Inf. Comput. Sci.*, 2001, 41, 619-626.

[35] Zhang, Z.J. DV-Curve: A novel intuitive tool for visualizing and analyzing DNA sequences. *Bioinformatics*, 2009, 25(9), 1112-1117.

[36] Liao, B.; Liao, B.Y.; Sun, X.M.; Zeng, Q.G. A Novel method for similarity analysis and protein sub-cellular localization prediction. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 2010, 7(2), 2678-2683.

[37] Nikolic, S.; Kovacevic, G.; Milicevic, A.; Trinajstic, N. The Zagreb indices 30 years after. * Croat. Chem. Acta*, 2003, 76, 113-124.

[38] Doslic, T.; Furtula, B.; Graovac, A.; Gutman, I.; Moradi, S.; Yarahmadi, Z. On vertex-degree-based molecular structure descriptors. *MATCH Commun. Math. Comput. Chem.*, 2011, 66, 613-626.

[39] Su, G.F.; Tu, J.H.; Das, K.-Ch. Graphs with fixed number of pendant vertices and minimal zeroth-order general Randic index. *Discrete Appl. Math.*, 2014, 172, 105-114.

[40] Randic, M.; Zupan, J.; Balaban, A.T.; Vikić-Topić, D.; Plavšić, D. Molecular graphs and structural similarity/dissimilarity of DNA sequences based on novel 2-D graphical representation. *Chem. Phys. Lett.*, 2003, 371, 202-207.