Analytical Biochemistry 462 (2014) 76–83

Contents lists available at ScienceDirect

Analytical Biochemistry

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

iTIS-PseTNC: A sequence-based predictor for identifying translation initiation site in human genes using pseudo trinucleotide composition

Wei Chen [a,b,*], Peng-Mian Feng [c], En-Ze Deng [d], Hao Lin [b,d,*], Kuo-Chen Chou [a,b,e,*]

[a] Department of Physics, School of Sciences, Center for Genomics and Computational Biology, Hebei United University, Tangshan 063000, China
[b] Gordon Life Science Institute, Boston, MA 02478, USA
[c] School of Public Health, Hebei United University, Tangshan 063000, China
[d] Key Laboratory for Neuro-Information of Ministry of Education, Center of Bioinformatics, School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China
[e] Center of Excellence in Genomic Medicine Research (CEGMR), King Abdulaziz University, Jeddah 21589, Saudi Arabia

ARTICLE INFO

Article history:
Received 20 May 2014
Received in revised form 26 June 2014
Accepted 27 June 2014
Available online 10 July 2014

Keywords:
Translation initiation site
Pseudo trinucleotide composition
Physicochemical properties
Support vector machine
Web server
iTIS-PseTNC

ABSTRACT

Translation is a key process for gene expression. Timely identification of the translation initiation site (TIS) is very important for conducting in-depth genome analysis. With the avalanche of genome sequences generated in the postgenomic age, it is highly desirable to develop automated methods for rapidly and effectively identifying TIS. Although some computational methods were proposed in this regard, none of them considered the global or long-range sequence-order effects of DNA, and hence their prediction quality was limited. To count this kind of effects, a new predictor, called "iTIS-PseTNC," was developed by incorporating the physicochemical properties into the pseudo trinucleotide composition, quite similar to the PseAAC (pseudo amino acid composition) approach widely used in computational proteomics. It was observed by the rigorous cross-validation test on the benchmark dataset that the overall success rate achieved by the new predictor in identifying TIS locations was over 97%. As a web server, iTIS-PseTNC is freely accessible at http://lin.uestc.edu.cn/server/iTIS-PseTNC. To maximize the convenience of the vast majority of experimental scientists, a step-by-step guide is provided on how to use the web server to obtain the desired results without the need to go through detailed mathematical equations, which are presented in this paper just for the integrity of the new prection method.

The translation in molecular biology is an important genetic process, by which the information carried by the messenger RNA (mRNA) is decoded by ribosome complex to produce a specific protein (or peptide) chain according to the rules specified by the genetic code. As one of the key processes for gene expression, translation proceeds in four phases: (i) initiation, (ii) elongation, (iii) translocation, and (iv) termination [1].

In the first initiation process, a proper start position on the mRNA will be identified. The region at which the translation is initiated is called the Translation Initiation Site (TIS),[1] as illustrated in Fig. 1.

In eukaryotes, the translation is performed according to the scanning model that accounts for the vast majority of mRNA translations. With this model, once the translation is initiated, the 40S ribosome subunit will first attach to the 5′ end of the mRNA and move along in the 5′ to 3′ direction until it has come to the first AUG in mRNA. Subsequently, it will stall to recruit the 60S subunit to form the 80S ribosome. And the latter will perform the second elongation process. Since it is the initiation position that determines the reading frame, faulty initiation may lead to a shift in the reading frame, resulting in a complete distortion of the message. Therefore, precise identification of TIS is crucial for understanding the mechanisms of translation.

Facing the explosion of biological sequences generated in the postgenomic age, it is a challenging task to develop high-throughput tools for identifying the TIS. Actually, some computational

* Corresponding authors. Address: Department of Physics, Hebei United University, Tangshan 063000, China (W. Chen). Address: School of Life Science and Technology, University of Electronic Science and Technology of China, Chengdu 610054, China (H. Lin). Address: Gordon Life Science Institute, Boston, MA 02478, USA. Fax: +86 315 3725715 (K.-C. Chou).

E-mail addresses: chenweiimu@gmail.com, wchen@gordonlifescience.org (W. Chen), fengpengmian@gmail.com (P.-M. Feng), enzeas@gmail.com (E.-Z. Deng), hlin@uestc.edu.cn, hlin@gordonlifescience.org (H. Lin), kcchou@gordonlifescience.org (K.-C. Chou).

[1] *Abbreviations used:* PseTNC, pseudo trinucleotide composition; TIS, translation initiation site; TNC, trinucleotide composition; SVM, support vector machine.

http://dx.doi.org/10.1016/j.ab.2014.06.022
0003-2697/

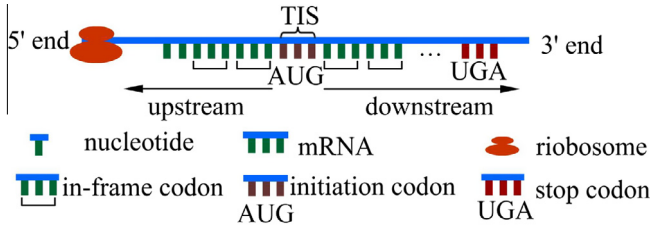

Fig.1. A schematic map to show the initiation region of translation process initiates.

methods were developed in this regard. The first model for predicting TIS was proposed by Pedersen and Nielsen [2]. In their method, the artificial neural network (ANN) approach was adopted. Subsequently, a series of TIS prediction models were developed by using different kinds of machine learning methods and sequence-coding strategies [3–8].

Stimulated by the works of these authors, we propose a new approach in hopes to further improve the prediction quality in identifying TIS.

As demonstrated by a series of recent publications [9–18] and summarized in a comprehensive review [19], to develop a really useful statistical predictor for a biological system, one needs to go through the following five steps: (i) select or construct a valid benchmark dataset to train and test the predictor; (ii) represent the samples with an effective formulation that can truly reflect their intrinsic correlation with the target to be predicted; (iii) introduce or develop a powerful algorithm to conduct the prediction; (iv) properly perform cross-validation to objectively evaluate the anticipated prediction accuracy; (v) establish a user-friendly web server for the predictor that is accessible to the public. Below, let us elaborate how to deal with these steps one by one.

Benchmark dataset

The genome coordinates of the annotated translation initiation sites in the human genome were obtained from the TIS database (TISdb) at http://tisdb.human.cornell.edu, which was built based on the multiple high-resolution Global Translation Initiation Sequencing (GTI-seq) [20]. Since the dataset is processed for DNA instead of RNA, the TIS should be ATG rather than AUG. Although the TIS also occurred with non-ATG, it was rarely happening in eukaryotes [21]. In the present study, therefore, we only considered the TIS occurring with ATG.

For different TISs, the surrounding sequence contents may be quite different, which poses a difficulty for determining the proper length of surrounding sequences. To deal with this problem, the "flexible scaled window" [22] approach was used, as shown in Fig. 2, where the size of the window is $(2\xi+3)$ with three zero scales at the center. When sliding the window along a sequence of N nucleotides, one can consecutively obtain $(N-2\xi-2)$ segments. The nucleotide segments thus obtained are called "potential TIS-containing segments" if their center is ATG, and hence will be further investigated; otherwise, disregarded.

For facilitating description later, let us adopt the formulation quite similar to the one used for studying HIV protease cleavage sites [23,24], specificity of GalNAc-transferase [25], and signal

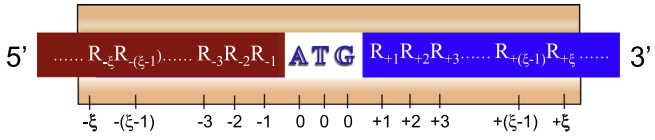

Fig.2. A schematic drawing to show how to slide the flexible scaled window along a DNA sequence to collect the potential TIS-containing segments. See Eqs. (1)–(4) and the relevant text for further explanation. Adapted from [26,72] with permission.

peptide cleavage sites [26]. According to that scheme, a potential TIS-containing segment, i.e., a nucleotide sequence with ATG located at its center can be expressed as

$$\mathbf{D}_\xi(\mathbb{ATG}) = \mathbf{R}_{-\xi}\mathbf{R}_{-(\xi-1)}\cdots\mathbf{R}_{-2}\mathbf{R}_{-1}\mathbb{ATG}\mathbf{R}_{+1}\mathbf{R}_{+2}\cdots\mathbf{R}_{+(\xi-1)}\mathbf{R}_{+\xi} \quad (1)$$

where the subscript ξ is an integer, $R_{-\xi}$ represents the ξ-th upstream nucleotide residue from the center, R_ξ the ξ-th downstream nucleotide residue, and so forth. The $(2\xi+3)$ -nucleotide-long sequence $\mathbf{D}_\xi(\mathbb{ATG})$ can be further classified into the following categories:

$$\mathbf{D}_\xi(\mathbb{ATG}) \in \begin{cases} \mathbf{D}_\xi^+(\mathbb{ATG}), & \text{if its center is TIS site} \\ \mathbf{D}_\xi^-(\mathbb{ATG}), & \text{otherwise} \end{cases} \quad (2)$$

where $\mathbf{D}_\xi^+(\mathbb{ATG})$ represents a true TIS-containing nucleotide segment, $\mathbf{D}_\xi^-(\mathbb{ATG})$ a false TIS-containing nucleotide segment, and $\in$ represents "a member of" in the set theory.

As pointed out by a comprehensive review [27], there is no need to separate a benchmark dataset into a training dataset and a testing dataset for examining the performance of a prediction method if it is tested by the jackknife test or subsampling (K-fold) cross-validation test since the outcome thus obtained is actually from a combination of many different independent dataset tests. Thus, the benchmark dataset for the current study can be formulated as

$$\mathbb{S}_\xi = \mathbb{S}_\xi^+ \cup \mathbb{S}_\xi^- \quad (3)$$

where $\mathbb{S}_\xi^+$ only contains the samples of $\mathbf{D}_\xi^+(\mathbb{ATG})$, i.e., the true TIS-containing nucleotide segments; $\mathbb{S}_\xi^-$ only contains the samples of $\mathbf{D}_\xi^-(\mathbb{ATG})$, i.e., the false TIS-containing nucleotide segments (cf. Eq. (2); and $\cup$ represents the symbol for "union" in the set theory.

Since the length of the peptide $\mathbf{D}_\xi(\mathbb{ATG})$ is $2\xi+3$ (cf. Eq. (1)), the benchmark dataset with different values of ξ will contain sequences of different numbers of nucleotide residues, as formulated by

$$\mathbb{S}_\xi \text{ contains the sequences of } \begin{cases} 393 \text{ residues}, & \text{when} \xi = 195 \\ 395 \text{ residues}, & \text{when} \xi = 196 \\ 397 \text{ residues}, & \text{when} \xi = 197 \\ 399 \text{ residues}, & \text{when} \xi = 198 \\ 401 \text{ residues}, & \text{when} \xi = 199 \\ \vdots & \vdots \end{cases} \quad (4)$$

The detailed procedures to construct $\mathbb{S}_\xi$ are as follows: (i) From the Human Genome Sequence (HG19), take 1159 DNA sequences that have clearly experiment-confirmed TIS annotations. (ii) Slide the $(2\xi+3)$-nucleotide-long flexible scaled window (Fig. 2) along each of the 1159 DNA sequences. (iii) Collect only those sequence segments with their central ATG being the annotated TIS location. (iv) The nucleotide sequence segments thus obtained were put into the positive subset $\mathbb{S}_\xi^+$.

The negative samples for $\mathbb{S}_\xi^-$ were derived from the 1,267,443 non-TIS sequences determined by Saeys et al. [28] in human chromosome 21. We processed all these false TIS-containing nucleotide sequences to make them have exactly the same format as the true TIS-containing sequences; i.e., each had $(2\xi+3)$-nucleotide-long sequence with ATG located at the center as well. To balance out the numbers between the negative and the positive samples for machine learning, we randomly picked 1159 samples from the 1,267,443 non-TIS sequences and used them as the negative samples.

By following the aforementioned procedures, five such benchmark datasets ($\mathbb{S}_{\xi=195}$, $\mathbb{S}_{\xi=196}$, $\mathbb{S}_{\xi=197}$, $\mathbb{S}_{\xi=198}$, and $\mathbb{S}_{\xi=199}$) had been constructed. Each of these datasets contained 1159 true TIS-con-

taining nucleotide sequences and 1159 false TIS-containing nucleotide sequences.

However, it was observed via preliminary trials that when $\xi = 198$, i.e., the samples concerned were formed by 198 nucleotide residues, the corresponding results were most promising. Accordingly, we choose $\mathbb{S}_{\xi=198}$ as the benchmark dataset for further investigation. Thus, Eq. (3) can be reduced to

$$\mathbb{S} = \mathbb{S}^+ \cup \mathbb{S}^- \tag{5}$$

where $\mathbb{S} = \mathbb{S}_{198}$, $\mathbb{S}^+ = \mathbb{S}^+_{198}$, and $\mathbb{S}^- = \mathbb{S}^-_{198}$. The detailed nucleotide sequences are given in Supporting Information S1.

DNA sequence formulation

Given a DNA or protein sequence, how do we formulate it for statistical prediction? This is a fundamental problem in computational biology. Roughly speaking, there are two kinds of models: one is the sequential model and the other is the discrete model.

In the sequential model, the most straightforward way to express a DNA sample **D** with L nucleic acid residues is just to use its sequence; i.e.,

$$\mathbf{D} = \mathrm{R}_1\mathrm{R}_2\mathrm{R}_3\mathrm{R}_4\mathrm{R}_5\mathrm{R}_6\mathrm{R}_7\cdots\mathrm{R}_\mathrm{L} \tag{6}$$

where R_1 represents the first nucleic acid residue at position 1, R_2 the second nucleic acid residue at position 2, and so forth. Using a sequential model like Eq. (6), one can adopt various sequence-similarity-search-based tools, such as BLAST [29], to perform statistical analysis. Unfortunately, this kind of straightforward and intuitive approach failed to work when a query sample did not have significant similarity to any of the character-known sequences.

To overcome the aforementioned difficulty, investigators resorted to the discrete model. Actually, another important reason for many investigator to shift their focus to the discrete or vector model is because all the existing operation engines, such as covariance discriminant (CD) [30,31], optimization approach [32], correlation coefficient method [33], neural network [34], support vector machine (SVM) [11,35], random forest [36], conditional random field [37], nearest neighbor (NN) [38]; K-nearest neighbor (KNN) [39,40], OET-KNN [41], Fuzzy K-nearest neighbor [39], ML-KNN algorithm [42], and SLLE algorithm [43], can be directly used to handle only vector but not sequence samples.

The simplest discrete model used to represent a DNA sample is its nucleic acid composition or NAC, as given below

$$\mathbf{D} = [f(\mathrm{A}) \quad f(\mathrm{C}) \quad f(\mathrm{G}) \quad f(\mathrm{T})]^{\mathbf{T}} \tag{7}$$

where $f(\mathrm{A})$, $f(\mathrm{C})$, $f(\mathrm{G})$ and $f(\mathrm{T})$ are the normalized occurrence frequencies of adenine (A), cytosine (C), guanine (G), and thymine (T) in the DNA sequence, respectively; the symbol **T** is the transpose operator. As we can see from Eq. (7), however, if using NAC to represent a DNA sample, all its sequence order information would be completely lost.

Now, how can we formulate a biological sequence with a discrete model or vector, yet still keep considerable sequence order information? Actually, it is one of the most important but also most difficult problems in computational biology.

One way to cope with such a problem is to represent the DNA sequence with the *k*-tuple nucleotide composition; i.e.,

$$\mathbf{D} = [f_1^{\text{K-tuple}} \quad f_2^{\text{K-tuple}} \quad \cdots \quad f_i^{\text{K-tuple}} \quad \cdots \quad f_{4^k}^{\text{K-tuple}}]^{\mathbf{T}}, \tag{8}$$

where f_i^{K}-tuple is the normalized occurrence frequency of the *i*-th *k*-tuple nucleotide in the DNA sequence. As we can see from Eq. (8), when $k > 3$ the number of components therein will rapidly increase. This will cause the high-dimension disaster [44] by facing the following consequences: (i) the overfitting problem that will make the prediction with extremely low capacity for tolerating deviation [45]; (ii) the information redundancy problem that will lead to serious bias [27] and misrepresentation; (iii) the overprediction or underprediction problem [42] that will significantly reduce the prediction accuracy.

To avoid the high-dimension disaster, here we used the 3-tuple nucleotide or trinucleotide composition (TNC) to formulate the DNA sample, as given by

$$\begin{aligned}\mathbf{D} &= \left[f_1^{\text{3-tuple}} \quad f_2^{\text{3-tuple}} \quad f_3^{\text{3-tuple}} \quad f_4^{\text{3-tuple}} \quad \cdots \quad f_{64}^{\text{3-tuple}}\right]^{\mathbf{T}} \\ &= [f(\mathrm{AAA}) \quad f(\mathrm{AAC}) \quad f(\mathrm{AAG}) \quad f(\mathrm{AAT}) \quad \cdots \quad f(\mathrm{TTT})]^{\mathbf{T}},\end{aligned} \tag{9}$$

where $f_1^{\text{3-tuple}} = f(\mathrm{AAA})$ is the normalized occurrence frequency of AAA in the DNA sequence; $f_2^{\text{3-tuple}} = f(\mathrm{AAC})$, that of AAC; $f_3^{\text{3-tuple}} = f(\mathrm{AAG})$, that of AAG; and so forth. By doing so, we can only incorporate the local sequence order information between the most contiguous nucleotides, but none of the global or long-range sequence-order information can be reflected.

Actually, a similar problem also occurred in computational proteomics, where the dipeptide or tripeptide composition approach could only contain the local or short-range sequence-order information of a protein chain. In order to incorporate its global or long-range sequence-order information, the pseudo amino acid composition [46,47] or Chou's PseAAC [48] was proposed. Since the concept of PseAAC was proposed in 2001 [46], it has been penetrating into almost all fields of protein attribute predictions (see, e.g., [49–54] as well as a long list of publications cited in a recent review [55]). Because it has been widely and increasingly used, recently three types of powerful open access software, called "PseAAC-Builder" [56], "propy" [57], and "PseAAC-General" [55], were established: the former two are for generating various modes of Chou's special PseAAC; while the third one for those of Chou's general PseAAC.

Inspired by the success of using PseAAC to represent protein samples, here we introduce the pseudo trinucleotide composition (PseTNC); i.e., instead of Eq. (9), the DNA sequence of Eq. (6) is formulated by

$$\mathbf{D} = [d_1 \quad d_2 \quad \cdots \quad d_{64} \quad d_{64+1} \quad \cdots \quad d_{64+\lambda}]^{\mathbf{T}}, \tag{10}$$

where

$$d_u = \begin{cases} \dfrac{f_u^{\text{3-tuple}}}{\sum_{i=1}^{64} f_i^{\text{3-tuple}} + w\sum_{j=1}^{\lambda}\theta_j}, & 1 \leqslant u \leqslant 64 \\ \dfrac{w\theta_{u-64}}{\sum_{i=1}^{64} f_i^{\text{3-tuple}} + w\sum_{j=1}^{\lambda}\theta_j}, & (64+1) \leqslant u \leqslant (64+\lambda) \end{cases} \tag{11}$$

In Eq. (11), $f_u^{\text{3-tuple}} (u = 1, 2, \cdots, 64)$ has exactly the same meaning as in Eq. (9), w is the weight factor, and the correlation factor θ_j is given by

$$\theta_j = \frac{1}{\mathrm{L}^* - j}\sum_{i=1}^{\mathrm{L}^*-j}\Theta(\mathrm{T}_i; \mathrm{T}_{i+j}) \quad (j = 1, 2, \cdots, \lambda < \mathrm{L}^*), \tag{12}$$

where $\mathrm{L}^* = \mathrm{Int}[\mathrm{L}/3]$ meaning to take the integer part of the number in the brackets, $\mathrm{T}_i = \mathrm{R}_i\mathrm{R}_{i+1}\mathrm{R}_{i+2}$ and $\mathrm{T}_{i+j} = \mathrm{R}_{i+j}\mathrm{R}_{i+j+1}\mathrm{R}_{i+j+2}$ (cf. Eq. (6)). Accordingly, θ_1 is called the first-tier correlation factor that reflects the sequence order correlation between all the most contiguous trinucleotides along a DNA sequence (Fig. 3a); θ_2, the second-tier correlation factor between all the second most contiguous trinucleotides (Fig. 3b); θ_3, the third-tier correlation factor between all the third most contiguous trinucleotides (Fig. 3c); and so forth. In Eqs. (11) and (12), λ represents the number of the total ranks or tiers to be counted [46,47]. Both λ and the weight factor w are the uncertain parameters, and their values will be further discussed later. The correlation function $\Theta(\mathrm{T}_i; \mathrm{T}_{i+j})$ in Eq. (12) is given by

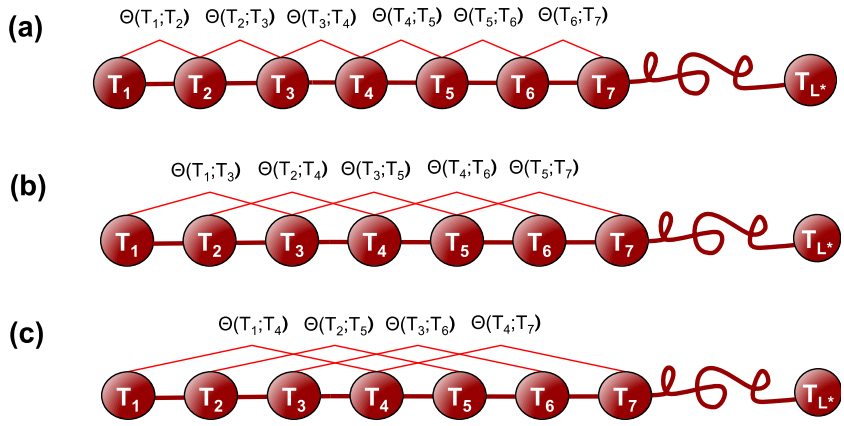

Fig.3. A schematic illustration to show the correlations of trinucleotides along a DNA sequence. (a) The first-tier correlation (blue line) reflects the sequence-order mode between all the most contiguous nonoverlapping trinucleotide. (b) The second-tier correlation (red line) reflects the sequence-order mode between all the second-most contiguous nonoverlapping trinucleotide. (c) The third-tier correlation (purple line) reflects the sequence-order mode between all the third-most contiguous nonoverlapping trinucleotide. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$\Theta(\mathrm{T}_i;\mathrm{T}_{i+j}) = \frac{1}{\mu}\sum_{v=1}^{\mu}[P_v(\mathrm{T}_i) - P_v(\mathrm{T}_{i+j})]^2, \tag{13}$$

where μ is the number of physicochemical properties considered that is equal to 3 in this study as will be explained later; $P_v(\mathrm{T}_i)$, the numerical value of the v-th physicochemical property for the trinucleotide T_i at position i; and $P_v(\mathrm{T}_{i+j})$, the corresponding value for the trinucleotide T_{i+j} at position $i+j$.

Selection of physicochemical properties

According to the famous 3 → 1 genetic rule, a codon of three nucleotides in DNA will define an amino acid in protein, as illustrated in Fig. 4. Thus, similar to the treatment in [58] where the three physicochemical properties of amino acids (i.e., their numerical values of hydrophobicity [59], hydrophilicity [60], and side-chain mass) were used to define the PseAAC for a protein sequence, here we can also use the three physicochemical properties to generate the PseTNC. In other words, we can assign the numerical values of the three amino acid physicochemical properties to each of the 64 nucleotides according to the 3 → 1 genetic rule, as shown in Table 1, where the value of zero was assigned to the trinucleotides TAA, TAG, and TGA since they are not corresponding to an amino acid but a "termination codon" or "stop" sign (Fig. 4).

Note that before substituting the values of $P_v(\mathrm{T}_i)$ in Table 1, they were all subjected to a standard conversion, as defined by [27]

$$P_v(\mathrm{T}_i) \Leftarrow \frac{P_v(\mathrm{T}_i) - \langle P_v(\mathrm{T}_i)\rangle}{\mathrm{SD}\langle P_v(T_i)\rangle} \quad (v = 1,2,3), \tag{14}$$

where the symbol < > means taking the average of the quantity therein over the 64 different trinucleotides (see Eq. (9)), and SD means the corresponding standard deviation. The converted values thus will have a zero mean value over the 64 different dinucleotides, and will remain unchanged if going through the same conversion procedure again. Listed in Table 2 are the values of $P_v(\mathrm{T}_i)$ obtained via the standard conversion of Eq. (14) from those of Table 1.

SVM operation engine

The SVM classification algorithm has been widely used in the realm of bioinformatics (see, e.g., [9,11,14,35,61]). Its basic principle is to transform the input vector into a high-dimension Hilbert space and seek a separating hyperplane with the maximal margin in this space by using the decision function

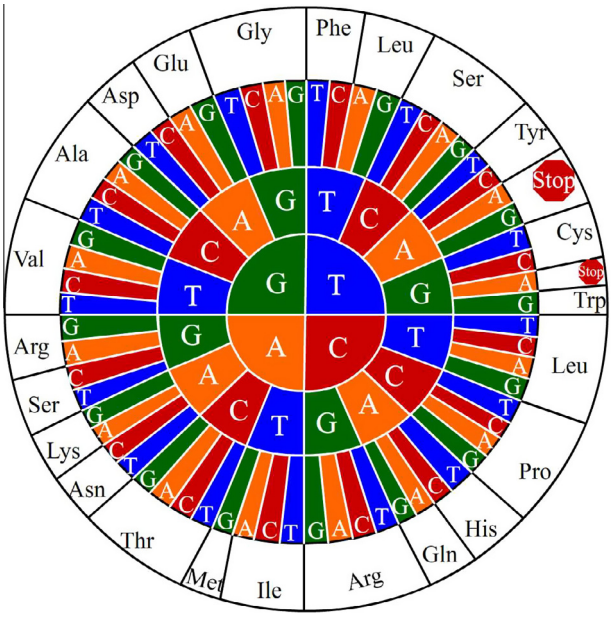

Fig.4. A schematic illustration to show how a codon of three nucleotides in DNA defines an amino acid in protein according to the 3 → 1 genetic rule. The characters in the first three rings from the center represent four bases in DNA, while those in the fourth ring represent the 3-letter codes of the 20 native amino acids in protein. See the text for more explanation.

$$f(\vec{X}) = \mathrm{sgn}\left(\sum_{i=1}^{N} y_i\alpha_i \cdot K(\vec{X},\vec{X}_i) + b\right), \tag{15}$$

where α_i is the Lagrange multipliers; b is the offset; $\vec{X}_i$ is the i-th training vector; y_i represents the type of the i-th training vector. $K(\vec{X},\vec{X}_i)$ is a kernel function which defines an inner product in a high-dimensional feature space, and sgn is the sign function. Due to its effectiveness and speed in the nonlinear classification process, the radial basis kernel function (RBF) $K(\vec{X}_i,\vec{X}_j) = \exp(-\gamma||\vec{X}_i - \vec{X}_j||^2)$ was used in the current study.

The package LIBSVM 2.84 (http://www.csie.ntu.edu.tw/~cjlin/libsvm/) by Chang and Lin was used to perform SVM in the current study. The regularization parameter C and the kernel width param-

Table 1
The original values for the 64 trinucleotides taken from their corresponding amino acids according to their hydrophobicity, hydrophilicity, and side-chain mass, respectively.[a]

Trinucleotide	P_1 (T_i)	P_2 (T_i)	P_3 (T_i)	Trinucleotide	P_1 (T_i)	P_2 (T_i)	P_3 (T_i)
AAA	−1.5	3	73	GAA	−0.74	3	73
AAC	−0.78	0.2	58	GAC	−0.9	3	59
AAG	−1.5	3	73	GAG	−0.74	3	73
AAT	−0.78	0.2	58	GAT	−0.9	3	59
ACA	−0.05	−0.4	45	GCA	0.62	−0.5	15
ACC	−0.05	−0.4	45	GCC	0.62	−0.5	15
ACG	−0.05	−0.4	45	GCG	0.62	−0.5	15
ACT	−0.05	−0.4	45	GCT	0.62	−0.5	15
AGA	−2.53	3	101	GGA	0.48	0	1
AGC	−0.18	0.3	31	GGC	0.48	0	1
AGG	−2.53	3	101	GGG	0.48	0	1
AGT	−0.18	0.3	31	GGT	0.48	0	1
ATA	1.38	−1.8	57	GTA	1.08	−1.5	43
ATC	1.38	−1.8	57	GTC	1.08	−1.5	43
ATG	0.64	−1.3	75	GTG	1.08	−1.5	43
ATT	1.38	−1.8	57	GTT	1.08	−1.5	43
CAA	−0.85	0.2	72	TAA	0	0	0
CAC	−0.4	−0.5	82	TAC	0.26	−2.3	107
CAG	−0.85	0.2	72	TAG	0	0	0
CAT	−0.4	−0.5	82	TAT	0.26	−2.3	107
CCA	0.12	0	42	TCA	−0.18	0.3	31
CCC	0.12	0	42	TCC	−0.18	0.3	31
CCG	0.12	0	42	TCG	−0.18	0.3	31
CCT	0.12	0	42	TCT	−0.18	0.3	31
CGA	−2.53	3	101	TGA	0	0	0
CGC	−2.53	3	101	TGC	0.29	−1	47
CGG	−2.53	3	101	TGG	0.81	−3.4	130
CGT	−2.53	3	101	TGT	0.29	−1	47
CTA	1.06	−1.8	57	TTA	1.19	−2.5	91
CTC	1.06	−1.8	57	TTC	1.19	−2.5	91
CTG	1.06	−1.8	57	TTG	1.06	−1.8	57
CTT	1.06	−1.8	57	TTT	1.06	−1.8	57

[a] The following symbols were used to represent the three physicochemical properties of the corresponding amino acids: P_1, hydrophobicity; P_2, hydrophilicity; P_3, side-chain mass.

Table 2
The normalized values obtained from Table 1 via the standard conversion.

Trinucleotide	P_1 (T_i)	P_2 (T_i)	P_3 (T_i)	Trinucleotide	P_1 (T_i)	P_2 (T_i)	P_3 (T_i)
AAA	−1.37	1.78	0.62	GAA	−0.66	1.78	0.62
AAC	−0.70	0.16	0.15	GAC	−0.81	1.78	0.18
AAG	−1.37	1.78	0.62	GAG	−0.66	1.78	0.62
AAT	−0.70	0.16	0.15	GAT	−0.81	1.78	0.18
ACA	−0.01	−0.19	−0.27	GCA	0.61	−0.25	−1.22
ACC	−0.01	−0.19	−0.27	GCC	0.61	−0.25	−1.22
ACG	−0.01	−0.19	−0.27	GCG	0.61	−0.25	−1.22
ACT	−0.01	−0.19	−0.27	GCT	0.61	−0.25	−1.22
AGA	−2.33	1.78	1.51	GGA	0.48	0.04	−1.67
AGC	−0.14	0.21	−0.71	GGC	0.48	0.04	−1.67
AGG	−2.33	1.78	1.51	GGG	0.48	0.04	−1.67
AGT	−0.14	0.21	−0.71	GGT	0.48	0.04	−1.67
ATA	1.32	−1.00	0.11	GTA	1.04	−0.83	−0.33
ATC	1.32	−1.00	0.11	GTC	1.04	−0.83	−0.33
ATG	0.63	−0.71	0.69	GTG	1.04	−0.83	−0.33
ATT	1.32	−1.00	0.11	GTT	1.04	−0.83	−0.33
CAA	−0.76	0.16	0.59	TAA	0.03	0.04	−1.70
CAC	−0.34	−0.25	0.91	TAC	0.27	−1.29	1.70
CAG	−0.76	0.16	0.59	TAG	0.03	0.04	−1.70
CAT	−0.34	−0.25	0.91	TAT	0.27	−1.29	1.70
CCA	0.14	0.04	−0.36	TCA	−0.14	0.21	−0.71
CCC	0.14	0.04	−0.36	TCC	−0.14	0.21	−0.71
CCG	0.14	0.04	−0.36	TCG	−0.14	0.21	−0.71
CCT	0.14	0.04	−0.36	TCT	−0.14	0.21	−0.71
CGA	−2.33	1.78	1.51	TGA	0.03	0.04	−1.70
CGC	−2.33	1.78	1.51	TGC	0.30	−0.54	−0.20
CGG	−2.33	1.78	1.51	TGG	0.79	−1.93	2.43
CGT	−2.33	1.78	1.51	TGT	0.30	−0.54	−0.20
CTA	1.02	−1.00	0.11	TTA	1.14	−1.41	1.19
CTC	1.02	−1.00	0.11	TTC	1.14	−1.41	1.19
CTG	1.02	−1.00	0.11	TTG	1.02	−1.00	0.11
CTT	1.02	−1.00	0.11	TTT	1.02	−1.00	0.11

eter γ were optimized via an optimization procedure using a grid search, and their actual values thus obtained in the current study were $C = 8$ and $\gamma = 0.125$, respectively. The probability score obtained from SVM was used to make predictions. If the probability score >0.5, an ATG will be predicted as TIS, otherwise, non-TIS.

The predictor obtained via the above procedures is called iTIS-PseTNC, where "i" stands for "identifying," "TIS" for "translation initiation site," "Pse" for "pseudo," "T" for "tri," "N" for "nucleotide," and "C" for "composition."

Performance evaluation method

In statistical prediction, the following three cross-validation methods are often used to examine a predictor for its effectiveness in practical application: independent dataset test, subsampling test, and jackknife test [62]. However, of the three methods, the jackknife test is deemed the least arbitrary that can always yield a unique result for a given benchmark dataset as elaborated in [63] and demonstrated by Eqs. (28)–(30) therein. Accordingly, the jackknife test has been increasingly and widely used by investigators to examine the quality of various predictors (see, e.g., [15,52,64–67]). Accordingly, the jackknife test was used in this study to examine the quality of the current predictor.

Also, in the literature a set of four metrics called the sensitivity (Sn), specificity (Sp), accuracy (Acc), and Mathew's correlation coefficient (MCC) are often used to measure the test quality. To make these metrics in a more intuitive and easier-to-understand formulation, let us use the following equations to represent them as done in a series of recent publications (see, e.g., [14,15,68])

$$\begin{cases} \mathrm{Sn} = 1 - \frac{N_-^+}{N^+}, & 0 \leqslant \mathrm{Sn} \leqslant 1 \\ \mathrm{Sp} = 1 - \frac{N_+^-}{N^-}, & 0 \leqslant \mathrm{Sp} \leqslant 1 \\ \mathrm{Acc} = 1 - \frac{N_-^+ + N_+^-}{N^+ + N^-}, & 0 \leqslant \mathrm{Acc} \leqslant 1 \\ \mathrm{MCC} = \frac{1 - \left(\frac{N_-^+}{N^+} + \frac{N_+^-}{N^-}\right)}{\sqrt{\left(1 + \frac{N_+^- - N_-^+}{N^+}\right)\left(1 + \frac{N_-^+ - N_+^-}{N^-}\right)}}, & -1 \leqslant \mathrm{MCC} \leqslant 1 \end{cases} \quad (16)$$

where N^+ is the total number of the TIS locations investigated while N_-^+ the number of the TIS locations incorrectly predicted as the non-TIS locations; N^- the total number of the non-TIS locations investigated while N_+^- the number of the non-TIS locations incorrectly predicted as the TIS locations. Using Eq. (16) would make the meaning of the four metrics crystal clear even for experimental scientists as elaborated in [10,12,69,70].

Parameter determination

As we can see from Eqs. (11) and (12), the current prediction model was based on two parameters w and λ. The former is the weight factor usually within the range from 0 to 1, and the latter is the number of the correlation tiers to be counted for the global sequence order information. Generally speaking, the greater the λ, the more global sequence-order information the model will contain. However, if λ is too large, it would reduce the cluster-tolerant capacity [45] so as to lower down the cross-validation accuracy due to overfitting or "high-dimension disaster" [44] problem. Therefore, our search for the optimal values of the two parameters was confined in the range:

$$\begin{cases} w \in [0, 1] \\ \lambda \in [1, 10] \end{cases}. \quad (17)$$

Furthermore, to reduce the computational time in searching for the optimal values of the two parameters, the 5-fold cross-validation approach was utilized. Once the optimal values were determined, the rigorous jackknife test was performed to evaluate the anticipated accuracy of the predictor.

Results and discussion

Success rates by jackknife test

The jackknife rates obtained by the iTIS-PseTNC predictor on the benchmark dataset (Supporting Information S1) for the four metrics of Eq. (16) are listed in Table 3, where, for facilitating comparison, listed are also the corresponding results by StartScan [28], the best of the existing predictors in this area. As we can see from the table, the current predictor outperformed the StartScan predictor in all four metrics, indicating that iTIS-PseTNC may become a useful high-throughput tool in identifying TIS locations, or at the very least play a complementary role to the existing predictors in this area.

Demonstration on an independent dataset

As elucidated under Performance evaluation method, the jackknife test is the most objective cross-validation approach, and hence there is no need to conduct an independent dataset test again. However, as a demonstration to show how to practically use the current predictor, we also constructed an independent dataset according to the following criteria: (i) included were only those TIS (or non-TIS) sequences confirmed by experiments, and (ii) none of the included TIS (or non-TIS) sequences occurs in the dataset used to train the iTIS-PseTNC predictor. By strictly following the above criteria, we randomly picked 200 TIS sequences from [28] to form an independent dataset $\mathbb{S}_{\mathrm{Ind}}$. The detailed sequences are given in Supporting Information S2. Tested on such an independent dataset, the iTIS-PseTNC predictor correctly identified 195 TIS and 194 non-TIS with Sn = 195/200 = 97.50% and Sp = 194/200 = 97.00%, which is fully consistent with the rates obtained by the jackknife test as shown in Table 3.

In addition, the current predictor trained with the benchmark dataset from the human genome was further used to identify the TISs in the mouth genome. Using similar procedures as described under Benchmark dataset, we collected 1300 experimentally confirmed TIS sequences from the TIS database (TISdb) and the mouse genome (mm 10) to form a second independent dataset $\mathbb{S}_{\mathrm{Mou}}$, as given in Supporting Information S3. Of the 1300 TISs in $\mathbb{S}_{\mathrm{Mou}}$, 1298 were correctly identified, indicating that the iTIS-PseTNC is really quite promising and holds a high potential to analyze the TIS locations for the genes from other species as well.

Web-server guide

For the convenience of the vast majority of experimental scientists, a web server for iTIS-PseTNC has been established. By following the procedures below, users can easily obtain their desired results without the need to worry about the mathematics involved during its development.

Table 3
The scores of the four metrics (cf. Eq. (16)) by the iTIS-PseTNC via the jackknife tests on the benchmark dataset.[a]

Predictor	Sn (%)	Sp (%)	Acc (%)	MCC
iTIS-PseTNC	97.49	98.42	97.92	0.958
StartScan	95.32[b]	96.43[b]	96.02[b]	0.921[b]

[a] See Supporting Information S1.
[b] Result obtained by StartScan [28] on the same benchmark dataset.

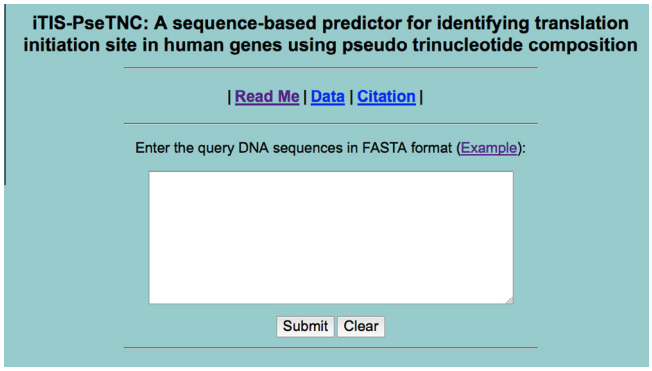

Fig.5. A screenshot to show the top page of the iTIS-PseTNC web server. Its website address is at http://lin.uestc.edu.cn/server/iTIS-PseTNC.

Step 1. Open the web server at http://lin.uestc.edu.cn/server/iTIS-PseTNC and you will see the top page of iTIS-PseTNC on your computer screen, as shown in Fig. 5. Click on the Read Me button to see a brief introduction about the predictor and the caveat when using it.

Step 2. Either type or copy/paste the query DNA sequences into the input box at the center of Fig. 5. The input sequence should be in the FASTA format. A sequence in FASTA format consists of a single initial line beginning with a greater-than symbol (">") in the first column, followed by lines of sequence data. The words right after the ">" symbol in the single initial line are optional and only used for the purpose of identification and description. All lines should be no longer than 120 characters and usually do not exceed 80 characters. The sequence ends if another line starting with a ">" appears; this indicates the start of another sequence. Example sequences in FASTA format can be seen by clicking on the Example button right above the input box.

Step 3. Click on the Submit button to see the predicted result. For example, if you use the query DNA sequences in the Example window as the input, you will see the following shown on the screen of your computer. (i) The outcome for the first query example is: there are 2 "ATG," the first one is TIS and the second one is non-TIS. (ii) The outcome for the second query sample is: there is 1 "ATG" and it is TIS. (iii) The outcome for the third query sample is: there is 1 "ATG" and it is non-TIS. All these results are fully consistent with the experimental observations.

Step 4. Click on the Data button to download the datasets used to train and test the iTIS-PseTNC predictor.

Step 5. Click on the Citation button to find the relevant papers that document the detailed development and algorithm of iTIS-PseTNC.

Caveats. Each of the input query sequences must be 399 bp or longer and they should only contain valid characters: 'A', 'C', 'G', and 'T'.

Conclusions

Knowledge of the translation initiation site is very important for performing in-depth genome analysis. A new predictor, called iTIS-PseTNC, was developed for identifying the TIS locations in human genes by taking into account both the local and the global sequence-order information of DNA sequences via the well-known PseAAC approach. The new predictor is very promising as reflected by the high success rates obtained by the rigorous jackknife tests. Although the current iTIS-PseTNC was trained by the benchmark dataset from human genome, it holds high potential to analyze the genomes of other species as well, as reflected by the quite promising outcome in a preliminary test of using it to identify the TIS locations of the mouse genes.

Since publicly accessible web servers represent the direction for developing practically more useful predictor [71], a user-friendly web server for iTIS-PseTNC has been established at http://lin.uestc.edu.cn/server/iTIS-PseTNC, by which users can easily obtain their desired results. Compared with the StartScan [28], the best of the existing web servers for predicting TIS, the current web server not only yielded higher prediction quality, but also was able to handle multiple query sequences with a single submission, which is beyond the reach of the StartScan web server.

It is anticipated that the new iTIS-PseTNC predictor may become a useful high-throughput tool for identifying the TIS locations, or, at the very least, it can play a complementary role to the existing methods in this area.

Acknowledgments

The authors thank the anonymous reviewers for their constructive comments, which were very helpful for strengthening the presentation of this study. The authors are also very much indebted to Dr. Yvan Saeys for his help in using StartScan. This work was supported by the National Nature Scientific Foundation of China (Nos. 61100092 and 61202256), the Nature Scientific Foundation of Hebei Province (No. C2013209105), and Science and Technology Department of Hebei Province (No. 132777133).

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.ab.2014.06.022.

References

[1] R.J. Jackson, C.U. Hellen, T.V. Pestova, The mechanism of eukaryotic translation initiation and principles of its regulation, Nat. Rev. Mol. Cell Biol. 11 (2010) 113–127.
[2] A.G. Pedersen, H. Nielsen, Neural network prediction of translation initiation sites in eukaryotes: perspectives for EST and genome analysis, Proc. Int. Conf. Intell. Syst. Mol. Biol. 5 (1997) 226–233.
[3] A.G. Hatzigeorgiou, Translation initiation start prediction in human cDNAs with high accuracy, Bioinformatics 18 (2002) 343–350.
[4] A.A. Salamov, T. Nishikawa, M.B. Swindells, Assessing protein coding region integrity in cDNA sequencing projects, Bioinformatics 14 (1998) 384–390.
[5] M. Tech, P. Meinicke, An unsupervised classification scheme for improving predictions of prokaryotic TIS, BMC Bioinformatics 7 (2006) 121.
[6] A. Zien, G. Ratsch, S. Mika, B. Scholkopf, T. Lengauer, K.R. Muller, Engineering support vector machine kernels that recognize translation initiation sites, Bioinformatics 16 (2000) 799–807.
[7] H. Li, T. Jiang, A class of edit kernels for SVMs to predict translation initiation sites in eukaryotic mRNAs, J. Comput. Biol. 12 (2005) 702–718.
[8] Y. Wang, H. Ou, F. Guo, Recognition of translation initiation sites of eukaryotic genes based on an EM algorithm, J. Comput. Biol. 10 (2003) 699–708.
[9] W. Chen, P.M. Feng, H. Lin, IRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition, Nucleic Acids Res. 41 (2013) e69.
[10] J.L. Min, X. Xiao, IEzy-Drug: a web server for identifying the interaction between enzymes and drugs in cellular networking, Biomed Res. Int. 2013 (2013) 701317.
[11] B. Liu, D. Zhang, R. Xu, J. Xu, X. Wang, Q. Chen, Q. Dong, Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection, Bioinformatics 30 (2014) 472–479.
[12] X. Xiao, J.L. Min, P. Wang, ICDI-PseFpt: identify the channel–drug interaction in cellular networking with PseAAC and molecular fingerprints, J. Theor. Biol. 337C (2013) 71–79.
[13] Y. Xu, X.J. Shao, L.Y. Wu, N.Y. Deng, ISNO-AAPair: incorporating amino acid pairwise coupling into PseAAC for predicting cysteine S-nitrosylation sites in proteins, PeerJ 1 (2013) e171.
[14] S.H. Guo, E.Z. Deng, L.Q. Xu, H. Ding, H. Lin, W. Chen, INuc-PseKNC: a sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition, Bioinformatics 30 (2014) 1522–1529.
[15] W.R. Qiu, X. Xiao, IRSpot-TNCPseAAC: identify recombination spots with trinucleotide composition and pseudo amino acid components, Int. J. Mol. Sci. 15 (2014) 1746–1766.

[16] Y.N. Fan, X. Xiao, J.L. Min, INR-drug: predicting the interaction of drugs with nuclear receptors in cellular networking, Int. J. Mol. Sci. 15 (2014) 4915–4937.
[17] Y. Xu, X. Wen, X.J. Shao, N.Y. Deng, IHyd-PseAAC: predicting hydroxyproline and hydroxylysine in proteins by incorporating dipeptide position-specific propensity into pseudo amino acid composition, Int. J. Mol. Sci. 15 (2014) 7594–7610.
[18] W.R. Qiu, X. Xiao, W.Z. Lin, IMethyl-PseAAC: identification of protein methylation sites via a pseudo amino acid composition approach, BioMed Res. Int. 2014 (2014) (ID 947416).
[19] K.C. Chou, Some remarks on protein attribute prediction and pseudo amino acid composition [50th Anniversary Year Review], J. Theor. Biol. 273 (2011) 236–247.
[20] J. Wan, S.B. Qian, TISdb: a database for alternative translation initiation in mammalian cells, Nucleic Acids Res. 42 (2014) D845–D850.
[21] M. Kozak, Initiation of translation in prokaryotes and eukaryotes, Gene 234 (1999) 187–208.
[22] K.C. Chou, H.B. Shen, Signal-CF: a subsite-coupled and window-fusing approach for predicting signal peptides, Biochem. Biophys. Res. Commun. 357 (2007) 633–640.
[23] K.C. Chou, A vectorized sequence-coupling model for predicting HIV protease cleavage sites in proteins, J. Biol. Chem. 268 (1993) 16938–16948.
[24] K.C. Chou, Review. Prediction of human immunodeficiency virus protease cleavage sites in proteins, Anal. Biochem. 233 (1996) 1–14.
[25] K.C. Chou, A sequence-coupled vector-projection model for predicting the specificity of GalNAc-transferase, Protein Sci. 4 (1995) 1365–1383.
[26] K.C. Chou, Prediction of signal peptides using scaled window, Peptides 22 (2001) 1973–1979.
[27] K.C. Chou, H.B. Shen, Review. Recent progresses in protein subcellular location prediction, Anal. Biochem. 370 (2007) 1–16.
[28] Y. Saeys, T. Abeel, S. Degroeve, Y. Van de Peer, Translation initiation site prediction on a genomic scale: beauty in simplicity, Bioinformatics 23 (2007) i418–i423.
[29] J.C. Wootton, S. Federhen, Statistics of local complexity in amino acid sequences and sequence databases, Comput. Chem. 17 (1993) 149–163.
[30] K.C. Chou, Prediction of G-protein-coupled receptor classes, J. Proteome Res. 4 (2005) 1413–1418.
[31] G.P. Zhou, K. Doctor, Subcellular location prediction of apoptosis proteins, Proteins Struct. Funct. Genet. 50 (2003) 44–48.
[32] C.T. Zhang, An optimization approach to predicting protein structural class from amino acid composition, Protein Sci. 1 (1992) 401–408.
[33] C.T. Zhang, A correlation coefficient method to predicting protein structural classes from amino acid compositions, Eur. J. Biochem. 207 (1992) 429–433.
[34] T.B. Thompson, C. Zheng, Neural network prediction of the HIV-1 protease cleavage sites, J. Theor. Biol. 177 (1995) 369–379.
[35] Y.D. Cai, G.P. Zhou, Support vector machines for predicting membrane protein types by using functional domain composition, Biophys. J. 84 (2003) 3257–3263.
[36] K.K. Kandaswamy, T. Martinetz, S. Moller, P.N. Suganthan, S. Sridharan, G. Pugalenthi, AFP-Pred: a random forest approach for predicting antifreeze proteins from sequence-derived properties, J. Theor. Biol. 270 (2011) 56–62.
[37] Y. Xu, J. Ding, L.Y. Wu, ISNO-PseAAC: predict cysteine S-nitrosylation sites in proteins by incorporating position specific amino acid propensity into pseudo amino acid composition, PLoS One 8 (2013) e55844.
[38] H.B. Shen, K.C. Chou, Using optimized evidence-theoretic K-nearest neighbor classifier and pseudo amino acid composition to predict membrane protein types, Biochem. Biophys. Res. Commun. 334 (2005) 288–292.
[39] X. Xiao, P. Wang, GPCR-2L: predicting G protein-coupled receptors and their types by hybridizing two different modes of pseudo amino acid compositions, Mol. BioSyst. 7 (2011) 911–919.
[40] P. Wang, X. Xiao, NR-2L: a two-level predictor for identifying nuclear receptor subfamilies based on sequence-derived features, PLoS One 6 (2011) e23505.
[41] K.C. Chou, H.B. Shen, Euk-mPLoc: a fusion classifier for large-scale eukaryotic protein subcellular location prediction by incorporating multiple sites, J. Proteome Res. 6 (2007) 1728–1734.
[42] K.C. Chou, Some remarks on predicting multi-label attributes in molecular biosystems, Mol. BioSyst. 9 (2013) 1092–1100.
[43] M. Wang, J. Yang, Z.J. Xu, SLLE for predicting membrane protein types, J. Theor. Biol. 232 (2005) 7–15.
[44] T. Wang, J. Yang, H.B. Shen, Predicting membrane protein types by the LLDA algorithm, Protein Pept. Lett. 15 (2008) 915–921.
[45] K.C. Chou, A key driving force in determination of protein structural classes, Biochem. Biophys. Res. Commun. 264 (1999) 216–224.
[46] K.C. Chou, Prediction of protein cellular attributes using pseudo amino acid composition, Proteins Struct. Funct. Genet. 43 (2001) 246–255 (Erratum: ibid., 2001, Vol.44, 60).
[47] K.C. Chou, Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes, Bioinformatics 21 (2005) 10–19.
[48] S.X. Lin, J. Lapointe, Theoretical and experimental biology in one, J. Biomed. Sci. Eng. (JBiSE) 6 (2013) 435–442.
[49] L. Nanni, A. Lumini, Genetic programming for creating Chou's pseudo amino acid based features for submitochondria localization, Amino Acids 34 (2008) 653–660.
[50] D.N. Georgiou, T.E. Karakasidis, J.J. Nieto, A. Torres, Use of fuzzy clustering technique and matrices to classify amino acids and its impact to Chou's pseudo amino acid composition, J. Theor. Biol. 257 (2009) 17–26.
[51] M. Mohammad Beigi, M. Behjati, H. Mohabatkar, Prediction of metalloproteinase family based on the concept of Chou's pseudo amino acid composition using a machine learning approach, J. Struct. Funct. Genomics 12 (2011) 191–197.
[52] Z. Hajisharifi, M. Piryaiee, M. Mohammad Beigi, M. Behbahani, H. Mohabatkar, Predicting anticancer peptides with Chou's pseudo amino acid composition and investigating their mutagenicity via Ames test, J. Theor. Biol. 341 (2014) 34–40.
[53] M. Khosravian, F.K. Faramarzi, M.M. Beigi, M. Behbahani, H. Mohabatkar, Predicting antibacterial peptides by the concept of Chou's pseudo-amino acid composition and machine learning methods, Protein Pept. Lett. 20 (2013) 180–186.
[54] H. Mohabatkar, M.M. Beigi, K. Abdolahi, S. Mohsenzadeh, Prediction of allergenic proteins by means of the concept of Chou's pseudo amino acid composition and a machine learning approach, Med. Chem. 9 (2013) 133–137.
[55] P. Du, S. Gu, Y. Jiao, 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 (2014) 3495–3506.
[56] P. Du, X. Wang, C. Xu, Y. Gao, PseAAC-Builder: a cross-platform stand-alone program for generating various special Chou's pseudo-amino acid compositions, Anal. Biochem. 425 (2012) 117–119.
[57] D.S. Cao, Q.S. Xu, Y.Z. Liang, Propy: a tool to generate various modes of Chou's PseAAC, Bioinformatics 29 (2013) 960–962.
[58] K.C. Chou, Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology, Curr. Proteomics 6 (2009) 262–274.
[59] C. Tanford, Contribution of hydrophobic interactions to the stability of the globular conformation of proteins, J. Am. Chem. Soc. 84 (1962) 4240–4247.
[60] T.P. Hopp, K.R. Woods, Prediction of protein antigenic determinants from amino acid sequences, Proc. Natl. Acad. Sci. U.S.A. 78 (1981) 3824–3828.
[61] K.C. Chou, Y.D. Cai, Using functional domain composition and support vector machines for prediction of protein subcellular location, J. Biol. Chem. 277 (2002) 45765–45769.
[62] K.C. Chou, C.T. Zhang, Prediction of protein structural classes, Crit. Rev. Biochem. Mol. Biol. 30 (1995) 275–349.
[63] K.C. Chou, H.B. Shen, Cell-PLoc 2.0: an improved package of web-servers for predicting subcellular localization of proteins in various organisms, Nat. Sci. 2 (2010) 1090–1103.
[64] Y.K. Chen, K.B. Li, Predicting membrane protein types by incorporating protein topology, domains, signal peptides, and physicochemical properties into the general form of Chou's pseudo amino acid composition, J. Theor. Biol. 318 (2013) 1–12.
[65] H. Mohabatkar, M. Mohammad Beigi, A. Esmaeili, Prediction of GABA(A) receptor proteins using the concept of Chou's pseudo-amino acid composition and support vector machine, J. Theor. Biol. 281 (2011) 18–23.
[66] S.S. Sahu, G. Panda, A novel feature representation method based on Chou's pseudo amino acid composition for protein structural class prediction, Comput. Biol. Chem. 34 (2010) 320–327.
[67] X.Y. Sun, S.P. Shi, J.D. Qiu, S.B. Suo, S.Y. Huang, R.P. Liang, Identifying protein quaternary structural attributes by incorporating physicochemical properties into the general form of Chou's PseAAC via discrete wavelet transform, Mol. BioSyst. 8 (2012) 3178–3184.
[68] W. Chen, P.M. Feng, H. Lin, ISS-PseDNC: identifying splicing sites using pseudo dinucleotide composition, Biomed. Res. Int. 2014 (2014) (ID 623149).
[69] X. Xiao, J.L. Min, P. Wang, IGPCR-Drug: a web server for predicting interaction between GPCRs and drugs in cellular networking, PLoS One 8 (2013) e72234.
[70] P.M. Feng, W. Chen, H. Lin, IHSP-PseRAAAC: identifying the heat shock protein families using pseudo reduced amino acid alphabet composition, Anal. Biochem. 442 (2013) 118–125.
[71] K.C. Chou, H.B. Shen, Review: recent advances in developing web-servers for predicting protein attributes, Nat. Sci. 2 (2009) 63–92.
[72] K.C. Chou, Using subsite coupling to predict signal peptides, Protein Eng. 14 (2001) 75–79.