OPEN ACCESS Freely available online

PLoS one

NR-2L: A Two-Level Predictor for Identifying Nuclear Receptor Subfamilies Based on Sequence-Derived Features

Pu Wang[1], Xuan Xiao[1,2]*, Kuo-Chen Chou[2]

1 Computer Department, Jing-De-Zhen Ceramic Institute, Jing-De-Zhen, China, **2** Gordon Life Science Institute, San Diego, California, United States of America

Abstract

Nuclear receptors (NRs) are one of the most abundant classes of transcriptional regulators in animals. They regulate diverse functions, such as homeostasis, reproduction, development and metabolism. Therefore, NRs are a very important target for drug development. Nuclear receptors form a superfamily of phylogenetically related proteins and have been subdivided into different subfamilies due to their domain diversity. In this study, a two-level predictor, called NR-2L, was developed that can be used to identify a query protein as a nuclear receptor or not based on its sequence information alone; if it is, the prediction will be automatically continued to further identify it among the following seven subfamilies: (1) thyroid hormone like (NR1), (2) HNF4-like (NR2), (3) estrogen like, (4) nerve growth factor IB-like (NR4), (5) fushi tarazu-F1 like (NR5), (6) germ cell nuclear factor like (NR6), and (7) knirps like (NR0). The identification was made by the Fuzzy *K* nearest neighbor (FK-NN) classifier based on the pseudo amino acid composition formed by incorporating various physicochemical and statistical features derived from the protein sequences, such as amino acid composition, dipeptide composition, complexity factor, and low-frequency Fourier spectrum components. As a demonstration, it was shown through some benchmark datasets derived from the NucleaRDB and UniProt with low redundancy that the overall success rates achieved by the jackknife test were about 93% and 89% in the first and second level, respectively. The high success rates indicate that the novel two-level predictor can be a useful vehicle for identifying NRs and their subfamilies. As a user-friendly web server, **NR-2L** is freely accessible at either http://icpr.jci.edu.cn/bioinfo/NR2L or http://www.jci-bioinfo.cn/NR2L. Each job submitted to **NR-2L** can contain up to 500 query protein sequences and be finished in less than 2 minutes. The less the number of query proteins is, the shorter the time will usually be. All the program codes for **NR-2L** are available for non-commercial purpose upon request.

Citation: Wang P, Xiao X, Chou K-C (2011) NR-2L: A Two-Level Predictor for Identifying Nuclear Receptor Subfamilies Based on Sequence-Derived Features. PLoS ONE 6(8): e23505. doi:10.1371/journal.pone.0023505

Editor: Niall James Haslam, University College Dublin, Ireland

Received February 17, 2011; **Accepted** July 19, 2011; **Published** August 15, 2011

Funding: This work was supported by the grants from the National Natural Science Foundation of China (No. 60961003), the Key Project of Chinese Ministry of Education (No. 210116), the Province National Natural Science Foundation of JiangXi (2009GZS0064 and 2010GZS0122), the Department of Education of JiangXi Province (No. GJJ09271), and the plan for training youth scientists (stars of Jing-Gang) of Jiangxi Province. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: xiaoxuan0326@yahoo.com.cn

Introduction

Nuclear receptors (NRs) are key transcription factors that regulate crucial gene networks important for cell growth, differentiation and homeostasis [1,2]. They function as ligand-activated transcription factors, thus providing a direct link between signaling molecules that control these processes and transcriptional responses. Many of these receptors are potential targets for the therapy of diseases such as breast cancer, diabetes, inflammatory diseases or osteoporosis. Nuclear receptors form a superfamily of phylogenetically-related proteins, which share a common structural organization. The N-terminal region (A/B domain) is highly variable, and contains at least one constitutionally active transactivation region (AT-1) and several autonomous transactivation domains (AD); A/B domains are variable in length, from less than 50 to more than 500 amino acids. The most conserved region is the DNA binding domain (DBD, C domain), which contains a short motif responsible for DNA-binding specificity on sequences typically containing the AGGTCT motif. A non-conserved hinge (D domain) is between the DNA-binding and ligand-binding domain, and contains the nuclear localization signal. The ligand-binding domain (LBD, E domain) is the largest domain. It is responsible for many functions, such as ligand induced, transactivation, and repression. The F domain is in the C terminus of the E domain, whose sequence is extremely variable and whose structure and function are unknown [3]. Not all the NRs contain all the six domains.

The importance of nuclear receptors has prompted the accumulation of rapidly increasing data from a great diversity of fields of research: sequences, expression patterns, three-dimensional structures, protein-protein interactions, target genes, physiological roles, mutations, etc. These collected data are very helpful for data mining and knowledge discovery. NR superfamily has been classified and assigned seven subfamilies based on the alignments of the conserved domains [3,4]. As a rising branch, the recognition of subfamilies of novel nuclear receptors is crucial for developing therapeutic strategies for the diseases mentioned above because the function of a nuclear receptor is closely correlated with its category.

Although the sequence similarity search-based tools, such as BLAST [5], are usually applied to conduct the prediction. However, this kind of approach failed to work when the query protein did not have significant sequence similarity to those of known attributes. Thus, various discrete models were proposed. The commonly used feature extraction methods are based on the concept of pseudo amino acid composition (PseAAC), which was proposed by Chou in studying protein subcellular location prediction and membrane protein type prediction [6], where a detailed description about PseAAC was elaborated.

In 2004, Bhasin and Raghava [7] have proposed a nuclear receptor subfamilies predicting method with the predictor of SVM and the input features of amino acid composition and dipeptide composition. Recently, Gao et al. [8] reconstructed the NR predicting dataset, and introduced the PseAAC [6] as the feature expression, thus enhancing the predictive quality. However, the existing predictors have the following shortcomings: (**1**) The datasets constructed to train the predictors cover very limited NRs subfamilies. For instance, the datasets constructed by these authors [7,8] only cover four subfamilies. (**2**) The cutoff threshold set by them to remove homologous sequences was 90%, meaning that the benchmark dataset thus constructed would allow inclusion of those proteins which have up to 90% pairwise sequence identity to others. To avoid homology bias, a much more stringent cutoff threshold should be adopted in constructing the benchmark datasets. (**3**) The existing predictors could not filter the irrelevant sequences, and all the input sequences would be assumed belonging to NRs regardless and hence might generate meaningless outcome. (**4**) No web-server was provided by the existing methods or the web-server provided by them is currently not working, and hence their application value is quite limited.

The present study was initiated in an attempt to develop a new predictor, called **NR-2L**, by addressing the above four shortcomings. To extend the coverage scope for practical application and reduce the homology bias, new benchmark datasets were constructed and a two-level predictor was developed. The new datasets cover seven subfamilies in which none of proteins included has $\geq 60\%$ pairwise sequence identity to any other in a same subset. Included in the new benchmark datasets are also the non-NR sequences for training the predictor to identify non-NR proteins. To make the predictor more powerful, more sequence-derived features were utilized. These features are capable of capturing the key information through PseAAC [6] as well as various physicochemical properties of proteins. The resulting feature vectors are finally fed into a simple yet powerful classification engine, called fuzzy K nearest neighbor algorithm, to identify NRs and their subfamilies. For the convenience of users and dealing with the situation that some link might be occasionally down, the web-server for **NR-2L** has been established at both http://icpr.jci.edu.cn/bioinfo/NR2L and http://www.jci-bioinfo.cn/NR2L, by any of which Multi-Fasta protein sequences can be input and handled in a batch mode. Furthermore, the source code of the algorithm is available for educational purposes and basic researches by e-mailing a request to the corresponding author.

To develop an effective method for identifying protein attributes such as NRs and their subfamilies, the following five things are indispensable [9]: (**1**) construct a valid benchmark dataset to train and test the predictor; (**2**) formulate the protein samples with an effective mathematical expression that can truly reflect their intrinsic correlation with the attribute to be predicted; (**3**) introduce or develop a powerful algorithm (or engine) to operate the prediction; (**4**) properly perform cross-validation tests to objectively evaluate the anticipated accuracy of the predictor; (**5**) 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.

Materials and Methods

1. Benchmark Datasets

Protein sequences were collected from the nuclear receptor data base (NucleaRDB release 5.0) at http://www.receptors.org/NR/, which is a part of a project devoted to build Molecular Class-Specific Information Systems (MCSIS) to provide, disseminate and harvest heterogeneous data [4]. The database have collected and harvested all the seven subfamilies of nuclear receptors marked with (**1**) NR1: thyroid hormone like (thyroid hormone, retinoic acid, RAR-related orphan receptor, peroxisome proliferator activated, vitamin D3-like), (**2**) NR2: HNF4-like (hepatocyte nuclear factor 4, retinoic acid X, tailless-like, COUP-TF-like, USP), (**3**) NR3: estrogen like (estrogen, estrogen-related, glucocorticoid-like), (**4**) NR4: nerve growth factor IB-like (NGFI-B-like), (**5**) NR5: fushi tarazu-F1 like (fushi tarazu-F1 like), (**6**) NR6: germ cell nuclear factor like (germ cell nuclear factor), and (**7**) NR0: knirps like (knirps, knirps-related, embryonic gonad protein, ODR7, trithorax) and DAX like (DAX, SHP). For detailed information about the database, refer to the NucleaRDB (http://www.receptors.org/NR/). Because the NucleaRDB has not provided the nuclear receptor sequences in FASTA format, we read Web content at the specified URL and extract all entries by the text-parsing method. The initial data set had 727 sequences belonging to seven subfamilies of nuclear receptors. To avoid any homology bias, a redundancy cutoff was imposed with the program CD-HIT to winnow those sequences which have $\geq 60\%$ pairwise sequence identity to any other in a same subset except for the subfamily NR6 because it contained only 5 nuclear receptor protein sequences [10]. If the redundancy-cutoff operation was also executed on this class, the samples left would be too few to have any statistical significance. The final benchmark dataset, $\mathbb{S}^{NR}$, thus obtained contains 159 sequences classified into seven different subfamilies of NRs as shown in **Table 1**, where 500 non-NRs protein sequences were also collected in $\mathbb{S}^{nNR}$for training the predictor to identifying non-NRs. The protein sequences in $\mathbb{S}^{nNR}$were randomly collected from the UniProt at http://www.uniprot.org/ according their annotations in the ''Keyword'' field, followed by undergoing the similar redundancy-cutoff operation to assure that none of the proteins in $\mathbb{S}^{nNR}$has $\geq 60\%$ pairwise sequence identity to any other. The accession numbers and sequences for the benchmark dataset thus obtained for $\mathbb{S}^{NR}$ and $\mathbb{S}^{nNR}$ are given in Supporting Information S1. Meanwhile, for the purpose of demonstrating the practical application of the current predictor, the corresponding independent testing datasets $\mathbb{S}_T^{NR}$ and $\mathbb{S}_T^{nNR}$ were also constructed (Table 1) in a way that none of proteins in the testing datasets occurs in $\mathbb{S}^{NR}$ and $\mathbb{S}^{nNR}$. The accession numbers and sequences for the independent testing datasets $\mathbb{S}_T^{NR}$ and $\mathbb{S}_T^{nNR}$are given in Supporting Information S2. It is instructive to point out that the results derived from such independent datasets are only a kind of demonstration that cannot be used to objectively measure the accuracy of a predictor; the real criterion for measuring the accuracy of the predictor should be based on the jackknife test as will be elaborated later.

2. Sequence-Derived Features

As pointed out in [9], to develop a predictor for identifying protein attributes, one of the keys is to formulate the protein samples with an effective mathematical expression that can truly reflect their intrinsic correlation with the attribute to be predicted.

Table 1. Breakdown of the learning dataset $\mathbb{S}$ and testing dataset $\mathbb{S}_T$.

Attribute	Training dataset $\mathbb{S}$			
	Set	Subfamily	Subset	Number
NR	$\mathbb{S}^{NR}$	NR1	$\mathbb{S}_1^{NR}$	50
		NR2	$\mathbb{S}_2^{NR}$	36
		NR3	$\mathbb{S}_3^{NR}$	37
		NR4	$\mathbb{S}_4^{NR}$	7
		NR5	$\mathbb{S}_5^{NR}$	12
		NR6	$\mathbb{S}_6^{NR}$	5
		NR0	$\mathbb{S}_0^{NR}$	12
Non-NR	$\mathbb{S}^{nNR}$	N/A	N/A	500
	Independent testing dataset $\mathbb{S}_T$			
NR	$\mathbb{S}_T^{NR}$	NR1	$\mathbb{S}_{T1}^{NR}$	231
		NR2	$\mathbb{S}_{T2}^{NR}$	127
		NR3	$\mathbb{S}_{T3}^{NR}$	148
		NR4	$\mathbb{S}_{T4}^{NR}$	23
		NR5	$\mathbb{S}_{T5}^{NR}$	33
		NR6	$\mathbb{S}_{T6}^{NR}$	0
		NR0	$\mathbb{S}_{T0}^{NR}$	6
Non-NR	$\mathbb{S}_T^{nNR}$	N/A	N/A	500

doi:10.1371/journal.pone.0023505.t001

A protein sequence **P** with L amino acid residues can be expressed as

$$\mathbf{P} = R_1R_2R_3R_4R_5R_6\cdots R_L \quad (1)$$

In order to capture as much useful information from a protein sequence as possible, we are to approach this problem from four different angles, followed by incorporating the feature elements thus obtained into the general form of PseAAC [9].

2.1 Amino Acid Composition (AAC)

As mentioned in the introduction, AAC was widely used to transform protein sequences into 20-D (dimensional) numerical vectors (see, e.g., [11,12,13,14]). The AAC of a protein is defined as the normalized occurrence frequencies of 20 amino acids in that protein; i.e.,

$$\mathrm{AAC} = [f_1, f_2, \cdots, f_{20}]^{\mathbf{T}} \quad (2)$$

where $f_i = n_i/L$ with each $i(=1,2,\cdots,20)$ corresponding to one of the 20 native amino acid types, and n_i the number of type i amino acids in the protein; while **T**is the transpose operator.

2.2 Dipeptide Composition (DC)

Traditional dipeptide (amino acid pair) composition was used to capture the local-order information of a protein sequence, which gives a fixed pattern length of 400 (20×20) [15]. The fraction of each dipeptide was formulated as

$$\text{Fraction of dip}(u) = \frac{\text{Total number of dip}(u)}{\text{Total number of all possible dipeptides}} \quad (3)$$

where $\mathrm{dip}(u)$ $(u=1,2,\cdots,400)$is the u-th dipeptide. In addition, to express the interaction of the amino acid for a pair with higher sequence gap than for the dipeptide pair (**Fig.1**), let us consider the following general equation

$$\text{Fraction of dip}^g(u) = \frac{\text{Total number of dip}^g\ (u)}{\text{Total number of all possible } g\text{-gap dipeptides}} \quad (4)$$

where $g=0$, 1, 2, or larger, and $\mathrm{dip}^g(u)(u=1,2,\cdots,400)$ is the u-th dipeptide with g gap between the two residues. When $g=0$, Eq.4 is reduced to Eq.3, the formulation for the conventional dipeptide. Accordingly, the dipeptide compositions with different gaps can be generally formulated as

$$\mathrm{DC}^g = \left[d_1^g, d_2^g, \cdots, d_{400}^g\right]^{\mathrm{T}} \quad (5)$$

where $d_u^g(u=1,2,\cdots,400)$is the$u$-thnormalized occurrence fre-

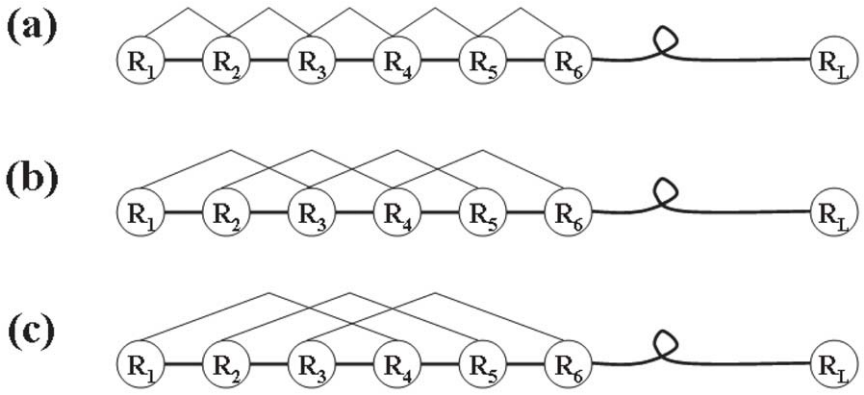

Figure 1. Schematic drawing to show dipeptides with different gaps along a protein chain. (a) The traditional (0-gap) dipeptide, (b) the 1-gap dipeptide, and (c) the 2-gaps dipeptide, where represents the amino acid residue at the sequence position 1, at position 2, and so forth. Adapted with permission from Chou [6].
doi:10.1371/journal.pone.0023505.g001

quency of the dipeptide of gap$g$. Since the couple effects among the local residues are usually stronger than those among the distant ones [16,17], here let us just consider the cases of $g=0$ and 1 as denoted by DC(0) and DC(1) respectively. Thus, we obtain $400\times 2=800$elements for using DC to formulate the protein sample, in which 400 elements are from DC(0) and 400 from DC(1).

2.3 Complexity Factor (CF)

A protein sequence is actually a symbolic sequence for which the complexity measure factor can be used to reflect its sequence feature or pattern and has been successfully used in some protein attribute prediction [18]. Among the known measures of complexity, the Lempel-Ziv (LZ) complexity [19] reflects the order that is retained in the sequence, and hence was adopted in this study.

The LZ complexity of a sequence $\mathbf{P}$ can be measured by the minimal number of steps required for its synthesis in a certain process. For each step only two operations were allowed in the process: either generating an additional symbol that ensures the uniqueness of each component $\mathbf{P}[i_{k-1}:i_k]$, or copying the longest fragment from the part of a synthesized sequence. Its substring is expressed by

$$\mathbf{P}[i:j]=\mathrm{R}_i\mathrm{R}_{i+1}\mathrm{R}_{i+2}\cdots\mathrm{R}_j(1\le i<j\le L) \tag{6}$$

The complexity measure factor, $\mathrm{CF}(\mathbf{P})$, of a nonempty sequence synthesized according to the following procedure is defined by

$$\mathrm{Syn}(\mathbf{P})=\mathbf{P}[1:i_1]\bullet\mathbf{P}[i_1+1:i_2]\bullet\cdots\bullet\mathbf{P}[i_{m-1}+1:L] \tag{7}$$

Let us assume that $\mathbf{P}=\mathrm{R}_1\mathrm{R}_2\mathrm{R}_3\mathrm{R}_4\mathrm{R}_5\mathrm{R}_6\cdots\mathrm{R}_L$ has been reconstructed by the program up to the residueR_r, and R_r has been newly inserted. The string up to R_r will be denoted by$\mathbf{P}[1:r]\bullet$, where the dot denotes that R_r is newly inserted to check whether the rest of the string $\mathbf{P}[r+1:L]$ can be reconstructed by a simple copying. First, suppose $q=\mathrm{R}_{r+1}$, and see whether q is reproducible from $\mathbf{P}[1:r]q\pi$, which means deleting the last character from the string $\mathbf{P}[1:r]q$. If the answer is "no," then we insert q into the sequence followed by a dot. Thus, it could not be obtained by the copying operation. If the answer is "yes," then no new symbol is needed and we can go on to proceed with $q=R_{r+1}R_{r+2}$ and repeat the same procedure. The LZ complexity is the number of dots (plus one if the string is not terminated by a dot). For example, for the sequence $\mathbf{P}=\mathrm{TMPPPETPSEGRQPSPSPSPTT}$, the LZ schema of synthesis generates the following components $\mathrm{Syn}(\mathbf{P})$and the corresponding complexity $\mathrm{CF}(\mathbf{P})$:

$$\begin{cases}\mathrm{Syn}(\mathbf{P})=\mathrm{T}\bullet\mathrm{M}\bullet\mathrm{P}\bullet\mathrm{PPE}\bullet\mathrm{TP}\bullet\mathrm{S}\bullet\mathrm{EG}\bullet\mathrm{R}\bullet\mathrm{Q}\bullet\mathrm{PSP}\bullet\mathrm{SPSPT}\bullet\mathrm{T}\\ \mathrm{CF}(\mathbf{P})=12\end{cases} \tag{8}$$

2.4 Fourier Spectrum Components (FSC)

Given a protein sequence $\mathbf{P}$, suppose $H(R_1)$is the certain physicochemical property value of the 1st residueR_1, $H(R_2)$ that of the 2nd residueR_2, and so forth. In terms of these property values the protein sequence can be converted to a digit signal$[H(R_1),H(R_2),\cdots,H(R_L)]$, for which we implement the discrete Fourier transform, obtaining the frequency-domain values,

$$X[k]=\sum_{l=1}^{L}H(R_i)\exp\left[-j\left(\frac{2\pi l}{L}\right)k\right],\quad (k=1,2,\cdots,L) \tag{9}$$

where j represents the imaginary number. For each $X[k]$ we can calculate its amplitude components F_k and phase components Φ_k

$$F_k=\mathrm{abs}(X[k]) \tag{10}$$

$$\Phi_k=\mathrm{angle}(X[k]) \tag{11}$$

Where **abs** gets the complex magnitude and **angle** gets the phase angle. Thus we can generate 2L discrete Fourier spectrum numbers as given below:

$$\{F_1,F_2,\cdots,F_L,\Phi_1,\Phi_2,\cdots,\Phi_L\} \tag{12}$$

The $2L$ Fourier spectrum numbers contain substantial information about the digit signal, and thereby can also be used to reflect characters of the sequence order of a protein. Furthermore, in the L phase components $\{\Phi_1,\Phi_2,\cdots,\Phi_L\}$, the high-frequency components are noisier and hence only the low-frequency components are more important. This is just like the case of protein internal motions where the low-frequency components are functionally more important [20]. For certain physicochemical property, accordingly, we only need to consider the 1st 10 phase components as well as their corresponding amplitudes, i.e.

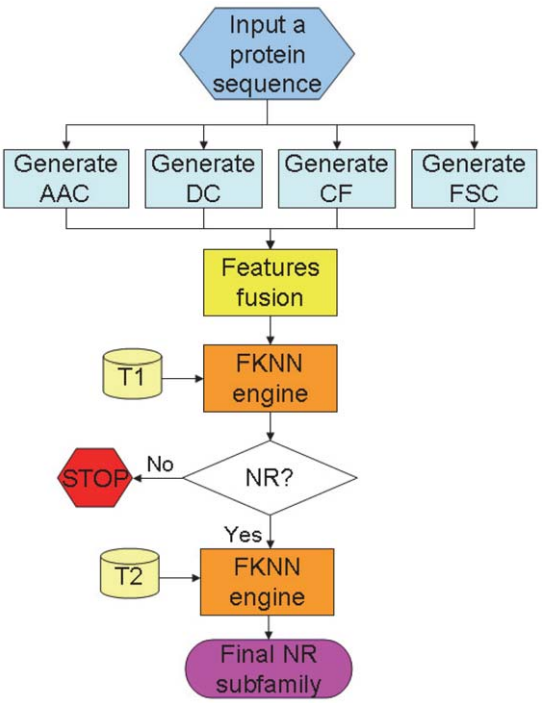

Figure 2. Flowchart to show the operation process of NR-2L. T1 represents the data taken from the Supporting Information S1 for training the 1st level prediction; T2 represents those from the Supporting Information S1 for training the 2nd level prediction. See the text for further explanation.
doi:10.1371/journal.pone.0023505.g002

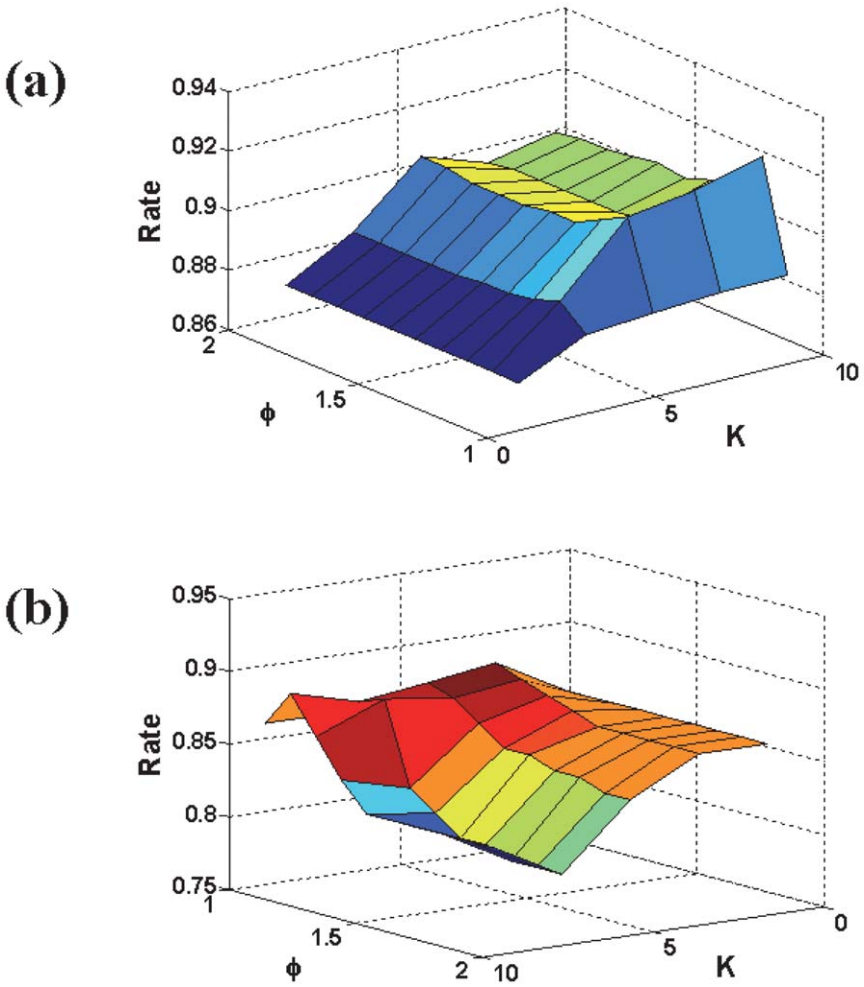

Figure 3. 3D graph to show the jackknife success rates with the different parameters. (a) The results obtained by the 1st level prediction, and (b) the results obtained by the 2nd level prediction, where the parameters and are defined in Eq.16.
doi:10.1371/journal.pone.0023505.g003

$$\mathrm{FSC}=[F_1,F_2,\cdots,F_{10},\Phi_1,\Phi_2,\cdots,\Phi_{10}]^T \quad (13)$$

As for the physicochemical property values, we adopted the hydrophobicity of each constituent amino acid, and its hydrophilicity and side-chain mass as done in [6]. These values can be obtained from the web-site at http://www.csbio.sjtu.edu.cn/bioinf/PseAAC/PseAAReadme.htm. Thus, we can obtain the 60 Fourier spectrum components.

2.5 Features Fusion into Pseudo Amino Acid Composition (PseAAC)

Finally, we obtained a total of 881 feature elements, of which 20 are from AAC, 800 from DC, 1 from CF, and 60 from FSC. Thus, according to the general formulation of PseAAC (cf. Eq.6 of [9]), a protein sample can be formulated as an 881-D vector given by

$$\mathbf{P}=[\psi_1,\psi_2,\cdots,\psi_{881}]^{\mathbf{T}} \quad (14)$$

Table 2. Prediction success rate and MCC index in identifying NR and non-NR by the jackknife test and independent dataset test.

Attribute	Jackknife test		Independent dataset test	
	ACC	MCC	ACC	MCC
NR	$\frac{156}{159}=98.11\%$	0.83	$\frac{566}{568}=99.65\%$	0.96
Non-NR	$\frac{454}{500}=90.80\%$	0.83	$\frac{481}{500}=96.20\%$	0.96
Overall	$\frac{610}{659}=92.56\%$		$\frac{1047}{1068}=98.03\%$	

doi:10.1371/journal.pone.0023505.t002

where

$$\psi_k = \begin{cases} \dfrac{f_k}{\sum_{i=1}^{20} f_i + \sum_{j=1}^{861} w_j p_j}, (1 \le k \le 20) \\ \dfrac{w_{(k-20)} p_{(k-20)}}{\sum_{i=1}^{20} f_i + \sum_{j=1}^{861} w_j p_j}, (21 \le k \le 881) \end{cases} \quad (15)$$

where $f_i (i = 1,2,\cdots,20)$ are the amino acid composition, $p_j (j = 1,2,\cdots,861)$ are the remaining 861 (=881-20) feature elements from dipeptide composition, complexity factor and Fourier spectrum components; w_j are the weight factors. In this study, the weight factor was set at 20 for all the feature elements from DC, $10^{-3}$for those from CF, and $10^{-4}$for those from FSC.

2.6 The Fuzzy K Nearest Neighbor (FKNN) Classifier

The *K*-nearest neighbor (*K*-NN) rule [21] is one of the simplest but quite powerful methods for performing nonparametric classification. The main idea of *K*-NN can be stated as following: Given a test sample with unknown label, its label is assigned according to the labels of its *K* nearest neighbors in the training set. Recently, the *K*-NN classifier has been successfully used to predict protein subcellular localization [22], membrane protein type, protease type, among many other protein attributes (see a long list of papers cited in a recent review [9]). For an intuitive illustration of how *K*-NN classifier works, see Fig.5 of [9].

Fuzzy *K*-NN classification method [23] is a special variation of the *K*-NN classification family. Instead of roughly assigning the label based on a voting from the *K* nearest neighbors, it attempts to estimate the membership values that indicate how much degree the query sample belongs to the classes concerned, Obviously, it is impossible for any characteristic description to contain complete information, which would make the classification ambiguous. In view of this, the fuzzy principle is very reasonable and particularly useful under such a circumstance.

Suppose $\{\mathbf{P}_1,\mathbf{P}_1,\cdots,\mathbf{P}_N\}$ is a set of vectors representing N proteins in the training set which has been classified into M classes: $\{C_1,C_2,\cdots,C_M\}$, where C_i denotes the i-th class. Thus, for a query protein $\mathbf{P}$, its fuzzy membership value for the i-th class is given by:

$$\mu_i(\mathbf{P}) = \frac{\sum_{j=1}^{K} \mu_i(\mathbf{P}_j) \mathrm{d}(\mathbf{P},\mathbf{P}_j)^{-2/(\varphi-1)}}{\sum_{j=1}^{K} \mathrm{d}(\mathbf{P},\mathbf{P}_j)^{-2/(\varphi-1)}} \quad (16)$$

where K is the number of the nearest neighbors counted; $\mu_i(\mathbf{P}_j)$is the fuzzy membership value of the protein $\mathbf{P}_j$ to the i-th class (it is set to 1 if the real label of $\mathbf{P}_j$ is C_i; otherwise, 0); $\mathrm{d}(\mathbf{P},\mathbf{P}_j)$ is the distance between the query protein $\mathbf{P}$ and its j-th nearest protein $\mathbf{P}_j$ in the training dataset; and $\varphi(>1)$ is the fuzzy coefficient for determining how heavily the distance is weighted when calculating each nearest neighbor's contribution to the membership value. Various metrics can be chosen for $\mathrm{d}(\mathbf{P},\mathbf{P}_j)$, such as Euclidean distance, Hamming distance, and Mahalanobis distance [11,24]. In this paper, the Euclidean metric was used. The values of φ and K will be mentioned later. After calculating all the memberships for a query protein, it is assigned to the class with which it has the highest membership value; i.e., the predicted class for the query protein $\mathbf{P}$ should be

Table 3. Prediction success rate and MCC index in identifying NR subfamilies by the jackknife test and independent test.

NR subfamily	Jackknife test		Independent dataset test	
	ACC	MCC	ACC	MCC
NR1	$\frac{43}{50}=86.00\%$	0.88	$\frac{229}{231}=99.13\%$	0.99
NR2	$\frac{31}{36}=86.11\%$	0.85	$\frac{127}{127}=100\%$	1.00
NR3	$\frac{37}{37}=100\%$	0.86	$\frac{148}{148}=100\%$	1.00
NR4	$\frac{6}{7}=85.71\%$	0.70	$\frac{23}{23}=100\%$	0.98
NR5	$\frac{10}{12}=83.33\%$	0.86	$\frac{33}{33}=100\%$	0.98
NR6	$\frac{5}{5}=100\%$	1	N/A	N/A
NR0	$\frac{9}{12}=75.00\%$	0.86	$\frac{6}{6}=100\%$	1.00
Overall	$\frac{141}{159}=88.68\%$		$\frac{566}{568}=99.65\%$	

doi:10.1371/journal.pone.0023505.t003

$$C_u = \mathbf{argmax}_i\{\mu_i(\mathbf{P})\} \quad (17)$$

where u is the argument of i that maximizes $\mu_i(\mathbf{P})$.

The predictor thus established is called **NR-2L**, where "2L" means the prediction consisting of two layers. The 1st layer is to identify a query protein as NR or not; if it is a NR, the 2nd layer will be automatically continued to further identify the NR among the seven subfamilies. To provide an intuitive picture, a flowchart to show the process of how the classifier works is given in **Fig.2**.

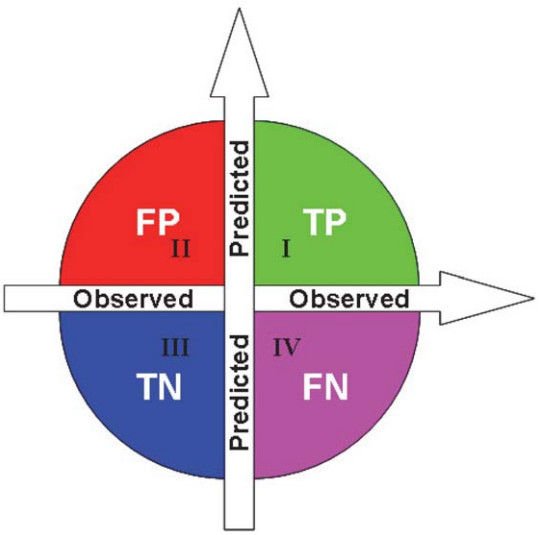

Figure 4. Distribution of predicted results in four quadrants. (I) TP, the true positive quadrant (green) for correct prediction of positive dataset, (II) FP, the false positive quadrant (red) for incorrect prediction of negative dataset; (III) TN, the true negative quadrant (blue) for correct prediction of negative dataset; and (IV) FN, the false negative quadrant (pink) for incorrect prediction of positive dataset.
doi:10.1371/journal.pone.0023505.g004

Table 4. The jackknife success rates obtained in identifying the NR subfamilies by separately using different features on the benchmark dataset of Supporting Information S1.

Feature mode	AAC	AAC+DC(0)	AAC+DC(1)	AAC+CF	AAC+FSC
Success rate	66.67%	81.76%	80.50%	72.33%	73.58%

doi:10.1371/journal.pone.0023505.t004

Results and Discussion

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 [25]. However, as elucidated and demonstrated by Eqs.28-32 of [9], among the three cross-validation methods, the jackknife test has the least arbitrary that can always yield a unique result for a given benchmark dataset, and hence has been increasingly and widely used by investigators to examine the accuracy of various predictors (see, e.g., [26,27,28,29,30,31,32]). Accordingly, the jackknife test was also adopted here to examine the quality of the present predictor.''

The values of parameter φ and K in **Eq.16** were determined by optimizing the overall jackknife success rate thru a 2-D search (**Fig.3**). It was found that the highest overall jackknife rate was obtained when $\varphi = 1.11$ and$K = 9$ in the first level, while $\varphi = 1.11$ and$K = 3$ in the second level. Thus, with the optimized parameters, predictions were further made for proteins in the independent data set. The success rates obtained by the jackknife test and independent test are given in **Table 2** and **Table 3** for the first and second level, respectively. The prediction result by the jackknife test for each of the proteins in the benchmark dataset $\mathbb{S} = \mathbb{S}^{\rm NR} + \mathbb{S}^{\rm nNR}$ is given in Supporting Information S3, and the prediction result for each of the proteins in the independent test set $\mathbb{S}_{\rm T} = \mathbb{S}_{\rm T}^{\rm NR} + \mathbb{S}_{\rm T}^{\rm nNR}$is given in Supporting Information S4.

As can be seen from the **Table 2** and **Table 3**, the success rates in identifying NRs and their subfamilies by both jackknife test and independent dataset test are very high, indicating that the **NR-2L** predictor is quite promising in generating reliable results for both basic research and drug development.

To further evaluate the performance of **NR-2L**, the Matthew's correlation coefficient (MCC) index, another widely used criterion in statistics, was also used. The definition of MCC index is given by

$$\mathrm{MCC} = \frac{(\mathrm{TP})(\mathrm{TN}) - (\mathrm{FP})(\mathrm{FN})}{\sqrt{[\mathrm{TP}+\mathrm{FP}][\mathrm{TP}+\mathrm{FN}][\mathrm{TN}+\mathrm{FP}][\mathrm{TN}+\mathrm{FN}]}} \qquad (18)$$

where TP represents the true positive; TN, the true negative; FP, the false positive; and FN, the false negative (see Fig.4). The corresponding MCC values thus obtained are also given in Table 2 and Table 3, from which we can see that NR-2L not only possess high accuracy but also quite stable even though the subset sizes are very different.

Also, it is instructive to see the results in Table 4, where the success rates obtained by using different features are separately listed. It can be seen from the table that, among the five feature combinations, the contribution from AAC+DC(0) is the highest to the successful prediction.

The results listed in Tables 2, 3, and 4 were obtained for the benchmark dataset with 60% cutoff threshold to exclude those protein sequences that have $\geq 60\%$pairwise sequence identity to any other in a same subset. To show the impact of such threshold values to the predicted results, an extensive study was performed on the datasets constructed by following exactly the same procedures as described in the ''Benchmark Datasets'' section with, however, cutoff thresholds 40%, 50%, 60%, 70%, respectively. The results thus obtained are given in Table 5, from which we can see that the larger the cutoff threshold value, the less stringent the benchmark dataset, and the higher the overall success rate by the jackknife test, fully in consistency with the elucidation as elaborated in [9].

Table 5. The jackknifing success rates obtained in identifying NR subfamilies with different redundancy reduction cutoff thresholds[a].

Redundancy / Subfamily	40%	50%	60%	70%
NR1	$\frac{22}{30} = 73.33\%$	$\frac{31}{37} = 83.78\%$	$\frac{43}{50} = 86\%$	$\frac{60}{65} = 92.31\%$
NR2	$\frac{11}{21} = 52.38\%$	$\frac{24}{29} = 82.76\%$	$\frac{31}{36} = 86.11\%$	$\frac{42}{46} = 91.30\%$
NR3	$\frac{13}{16} = 81.25\%$	$\frac{22}{22} = 100\%$	$\frac{37}{37} = 100\%$	$\frac{48}{48} = 100\%$
NR4	$\frac{1}{4} = 25\%$	$\frac{1}{4} = 25\%$	$\frac{6}{7} = 85.71\%$	$\frac{7}{8} = 87.50\%$
NR5	$\frac{4}{7} = 57.14\%$	$\frac{7}{9} = 77.78\%$	$\frac{10}{12} = 83.33\%$	$\frac{12}{14} = 85.71\%$
NR6	$\frac{5}{5} = 100\%$	$\frac{5}{5} = 100\%$	$\frac{5}{5} = 100\%$	$\frac{5}{5} = 100\%$
NR0	$\frac{3}{9} = 33.33\%$	$\frac{5}{10} = 50\%$	$\frac{9}{12} = 75\%$	$\frac{11}{14} = 78.57\%$
Overall	$\frac{59}{92} = 64.13\%$	$\frac{95}{116} = 81.90\%$	$\frac{141}{159} = 88.68\%$	$\frac{185}{200} = 92.50\%$

[a]We did not eliminate the redundancy of NR6 subfamily because it contained only 5 nuclear receptors. If the redundancy-cutoff operation was also executed on this class, the samples left would be too few to have any statistical significance.

Owing to the functional importance of NRs and the rapid increasing of their sequences, it is important and feasible to develop a reliable predictor for identifying NRs and their subfamilies based on the sequence information. The NR-2L predictor developed in this study can be used to address this kind of problems. The high success rates achieved by NR-2L have once again indicated that it is indeed an effective approach by fussing several different kinds of sequence-derived features into PseAAC to formulate protein samples for identifying their attributes. It is anticipated that NR-2L may become a useful tool in speeding up the pace of characterizing newly found nuclear receptor proteins or at least may play an important complementary role to the other methods in this regard. For the convenience of biologists and pharmacologists in using NR-2L, a user-friendly web-server for NR-2L has been established at http://icpr.jci.edu.cn/bioinfo/NR2L, by which users can easily obtain the desired results in a short period of time even for a large number of query protein sequences. Furthermore, as a backup, the web-server for NR-2L can also be accessed at http://www.jci-bioinfo.cn/NR2L in case the former link is down. All the program codes for NR-2L are available for non-commercial purpose upon request.

Supporting Information

Supporting Information S1 The training dataset S contains 500 non-NR proteins and 159 NR proteins classified into the following 7 main subfamilies according to NucleaRDB (http://www.receptors.org/NR/): (1) NR1: thyroid hormone like; (2) NR2: HNF4-like; (3) NR3: estrogen like; (4) NR4: nerve growth factor IB-like; (5) NR5: fushi tarazu-F1 like; (6) NR6: germ cell nuclear factor like; and (7) NR0: knirps and DAX like. Both the accession numbers and sequences are given. None of the proteins included has $\geq$60% pairwise sequence identity to any other in the same subset except the NR6 subfamily.
(PDF)

Supporting Information S2 The independent testing dataset ST contains 500 non-NR proteins and 568 NR proteins classified into the following 7 main subfamilies according to NucleaRDB (http://www.receptors.org/NR/): (1) NR1:thyroid hormone like; (2) NR2: HNF4-like; (3) NR3: estrogen like; (4) NR4: nerve growth factor IB-like; (5) NR5: fushi tarazu-F1 like; (6) NR6: germ cell nuclear factor like; and (7) NR0: knirps and DAX like. Both the accession numbers and sequences are given. None of the proteins included here occurs in the training dataset S.
(PDF)

Supporting Information S3 List of the jackknifing results obtained by NR-2L on the 159 NRs and 500 non-NRs in the dataset S (cf. Supporting Information S1), and the corresponding observed results as annotated in NucleaRDB or UniProt.
(PDF)

Supporting Information S4 List of the results obtained by NR-2L on the 568 NRsand 500 non-NRs in the independent testing dataset ST (cf. Supporting Information S2), and the corresponding observed results as annotated in NucleaRDB or UniProt.
(PDF)

Acknowledgments

The authors wish to thank Professor Niall Haslam for his constructive suggestions. The authors also wish to thank the two anonymous reviewers for their valuable comments, which are very helpful for strengthening the presentation of the paper.

Author Contributions

Conceived and designed the experiments: XX. Performed the experiments: PW. Analyzed the data: KCC. Contributed reagents/materials/analysis tools: XX. Wrote the paper: XX KCC.

References

1. Altucci L, Gronemeyer H (2001) Nuclear receptors in cell life and death. Trends in Endocrinology and Metabolism 12: 460–468.
2. Mangelsdorf DJ, Thummel C, Beato M, Herrlich P, Schutz G, et al. (1995) The nuclear receptor superfamily: the second decade. Cell 83: 835–839.
3. Robinson-Rechavi M, Garcia HE, Laudet V (2003) The nuclear receptor superfamily. J Cell Sci 116: 585–586.
4. Florence H, Gerrit V, Fred EC (2001) Collecting and harvesting biological data: the GPCRDB and NucleaRDB information systems. Nucleic Acids Research 29: 346–349.
5. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25: 3389–3402.
6. Chou KC (2001) Prediction of protein cellular attributes using pseudo-amino acid composition. Proteins 43: 246–255.
7. Bhasin M, Raghava GPS (2004) Classification of Nuclear Receptors Based on Amino Acid Composition and Dipeptide Composition. Journal of Biological Chemistry 279: 23262–23266.
8. Gao QB, Jin ZC, Ye XF, Wu C, He J (2009) Prediction of nuclear receptors with optimal pseudo amino acid composition. Analytical Biochemistry 387: 54–59.
9. Chou KC (2011) Some remarks on protein attribute prediction and pseudo amino acid composition (50th Anniversary Year Review). Journal of Theoretical Biology 273: 236–247.
10. Li W, Godzik A (2006) Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. Bioinformatics 22: 1658–1659.
11. Chou KC (1995) A novel approach to predicting protein structural classes in a (20–1)-D amino acid composition space. Proteins: Structure, Function, and Bioinformatics 21: 319–344.
12. Nakashima H, Nishikawa K, Ooi T (1986) The folding type of a protein is relevant to the amino acid composition. J Biochem 99: 153–162.
13. Zhou GP (1998) An intriguing controversy over protein structural class prediction. Journal of Protein Chemistry 17: 729–738.
14. Zhou GP, Doctor K (2003) Subcellular location prediction of apoptosis proteins. PROTEINS: Structure, Function, and Genetics 50: 44–48.
15. Liu W, Chou KC (1999) Protein secondary structural content prediction. Protein Engineering 12: 1041–1050.
16. Chou KC (1993) A vectorized sequence-coupling model for predicting HIV protease cleavage sites in proteins. Journal of Biological Chemistry 268: 16938–16948.
17. Chou KC (2000) Review: Prediction of tight turns and their types in proteins. Analytical Biochemistry 286: 1–16.
18. Xiao X, Shao SH, Huang ZD, Chou KC (2006) Using pseudo amino acid composition to predict protein structural classes: approached with complexity measure factor. Journal of Computational Chemistry 27: 478–482.
19. Gusev VD, Nemytikova LA, Chuzhanova NA (1999) On the complexity measures of genetic sequences. Bioinformatics 15: 994–999.
20. Chou KC (1988) Review: Low-frequency collective motion in biomacromolecules and its biological functions. Biophysical Chemistry 30: 3–48.
21. Cover T, Hart P (1967) Nearest neighbor pattern classification. IEEE transactions on information theory 13: 21–27.
22. Chou KC, Wu ZC, Xiao X (2011) iLoc-Euk: a multi-label classifier for predicting the subcellular localization of singleplex and multiplex eukaryotic proteins. PLoS ONE 6: e18258.
23. Keller JM, Gray MR, Givens JAJ (1985) A frezzy K-nearest neighbor algorithm. IEEE transactions on systems, man, and cybernetics 15: 580–585.
24. Mahalanobis PC (1936) On the generalized distance in statistics. Proc Natl Inst Sci India 2: 49–55.
25. Chou KC, Zhang CT (1995) Review: Prediction of protein structural classes. Critical Reviews in Biochemistry and Molecular Biology 30: 275–349.
26. Liu T, Jia C (2010) A high-accuracy protein structural class prediction algorithm using predicted secondary structural information. Journal of Theoretical Biology 267: 272–275.
27. Masso M, Vaisman II (2010) Knowledge-based computational mutagenesis for predicting the disease potential of human non-synonymous single nucleotide polymorphisms. Journal of Theoretical Biology 266: 560–568.
28. Wang T, Xia T, Hu XM (2010) Geometry preserving projections algorithm for predicting membrane protein types. Journal of Theoretical Biology 262: 208–213.
29. Joshi RR, Sekharan S (2010) Characteristic peptides of protein secondary structural motifs. Protein & Peptide Letters 17: 1198–1206.
30. Kandaswamy KK, Pugalenthi G, Moller S, Hartmann E, Kalies KU, et al. (2010) Prediction of Apoptosis Protein Locations with Genetic Algorithms and

Support Vector Machines Through a New Mode of Pseudo Amino Acid Composition. Protein and Peptide Letters 17: 1473–1479.
31. Liu T, Zheng X, Wang C, Wang J (2010) Prediction of Subcellular Location of Apoptosis Proteins using Pseudo Amino Acid Composition: An Approach from Auto Covariance Transformation. Protein & Peptide Letters 17: 1263–1269.
32. Mohabatkar H (2010) Prediction of cyclin proteins using Chou's pseudo amino acid composition. Protein & Peptide Letters 17: 1207–1214.