Hindawi
Scientific Programming
Volume 2021, Article ID 6694846, 7 pages
https://doi.org/10.1155/2021/6694846

Research Article

Prediction and Analysis of Protein Ubiquitin Sites in the Model Plant *A. thaliana*

Shujun Shan,[1,2] Yue Qi,[1] Jihong Jiang,[3] and Song Guo [4]

[1]*Department of Pharmaceutical Engineering, Jiangsu Provincial Xuzhou Pharmaceutical Vocational College, Xuzhou 221116, China*
[2]*Wanbang Biopharmaceuticals, Xuzhou 221004, China*
[3]*The Key Laboratory of Biotechnology for Medicinal Plant of Jiangsu Province, Jiangsu Normal University, Xuzhou, China*
[4]*Department of Computer Application, Shenyang Sport University, Shenyang 110102, China*

Correspondence should be addressed to Song Guo; guosong_7901@163.com

Received 10 December 2020; Revised 6 January 2021; Accepted 21 January 2021; Published 22 February 2021

Academic Editor: Wenzheng Bao

Ubiquitin is an important type of protein after translational modification. Ubiquitin has the ability to take part in several cellular regulations among several biological processions. At the same time, ubiquitin plays key roles in the enzymatic process. So as to construct the new tool to classify the ubiquitin amino acid residues, we employed the random forest model to classify the ubiquitin sites utilizing the experimentally identified ubiquitinated protein sequences of *A. thaliana*. More detailed, we utilized the *k*-spaced amino acid pair (CKSAAP) encoding and binary encoding to deal with the potential protein segments. The proposed tools can obtain 72.83% in Sp, 72.42% in Sn, 72.63% in Acc, and 0.4525 in MCC. With these performances, such tools can obtain the available results in the dataset of *Arabidopsis*.

1. Introduction

Ubiquitin is a typical type of protein after translational modification in the field of protein research [1–7]. Such modification has the ability to take part in several significant processions in the cellular level. At the same time, ubiquitin is an important element in the enzymatic process, which is a key biological procession among the life activities. When it comes to ubiquitin, such issue is attached by a special regulation protein procession [8–15]. Such issue is not only the single ubiquitin but also the chain adjoining. The majority of ubiquitin modifications appear in the lysine residues in the level of protein and peptide. Such process can be defined as three key steps, including activation, conjugation, and ligation. First of all, ubiquitin activates enzymes (E1s). And then, such protein conjugates enzymes (E2s) to complete the conjugation process. Last but not least, such protein ligases enzymes (E3s). The aforementioned steps are the main procession of ubiquitin. It is well known that ubiquitin is one of the posttranslational modifications of proteins in any cell, which is highly related to many biological processes and different kinds of diseases in plants and animals. Alzheimer's disease, for example, has been found to be downregulated and regulated by various types of signaling and endocytosis, such as Parkinson's disease and anaphylaxis. These functions are as important as other regulatory functions in biological processes [16–22].

It is well known that ubiquitin plays an important role in several protein constructions and processions. Furthermore, a great number of research studies have been carried out to reveal the molecular properties and regulation functions with ubiquitin in the whole biological process. In order to identify ubiquitin amino acid residues in the protein level, a variety of experimental methods have been proposed, such as ubiquitin antibodies (anti-ubi) and ubiquitin-binding proteins (binding-ubi), high-throughput mass spectrometry (MS), liquid chromatography, and mass spectrometry. Due to the dynamics,

rapidity, and reversibility of ubiquitin, the existing experimental methods are expensive, laborious, and time consuming. Therefore, species-specific computational methods have been proposed to identify such modification sites, which are considered to be efficient, convenient, and economical. Different calculation methods with unique characteristics have been developed using unique methods [23–29].

In a variety of species, ubiquitin amino acid can hardly be treated as conservative. Therefore, the existing predictors of ubiquitin sites are not suitable for predicting multispecies ubiquitin sites in different organisms. In previous studies, the existing predictive variables were applied to the *Arabidopsis* dataset. The results showed that the classification performance can hardly meet the need of effective identification. We can see that it is necessary to develop predictors of species-specific ubiquitin sites in order to improve the model's performance. Nevertheless, only one predictor has been developed for the model plant *Arabidopsis* species. Although the existing prediction variables are well predicted, there is still room for improvement. In this work, we try to find a new computational tool to identify ubiquitin sites with the protein sequences of *Arabidopsis* species [30–32].

In order to develop such new tools, we employed the random forest model as the classifier to deal with the potential ubiquitin amino acid residues among the protein sequences of *Arabidopsis* proteins. At the same time, the composition of k-spacing amino acid pairs (CKSAAP) was selected as the main feature of this work. The proposed RF_CKSAAP tool demonstrates better performance than other state-of-the-art algorithms in the field of machine learning. In the next section, we will introduce the whole development process of the proposed new method in detail.

2. Materials and Methods

Generally speaking, our method is based on RF to predict the ubiquitin sites. It is developed based on a comparative work among the consecutive sequence coding features (CKSAAP and binary coding methods) and other typical classification methods based on optimal window size 27. After studying the two coding methods, the RF classifier based on CKSAAP coding is used to design a ubiquitin identification tool. And then, optimizing the parameters and evaluating the performance, the optimal model is accomplished [33–35].

2.1. Dataset Arrangement. In this work, the ubiquitinated protein sequence training dataset was used to train the model, and its performance was checked in an independent test dataset. The ubiquitinated protein sequence was arranged according to previously published papers [36–42]. Reports on plant cells with experimentally confirmed ubiquitin sites (lysine residues) were extracted from [42]. Ubiquitin site annotations were extracted from UniProtKB/Swiss-Prot and NCBI protein sequence database (https://www.nim.gov protein/) and were related to the model plant *Arabidopsis*. With the selection of *Arabidopsis*'s protein, we can easily find that the modified sites are far more lower than the sites that are not modified. In order to deal with such situations, we may construct several ratios among the positive and negative sample. In this work, nonubiquitin sites from all negative samples are randomly selected, and a training dataset is constructed with the ratio of positive samples to negative samples of 1 : 1, 1 : 2, and 1 : 3.

In order to test the performance of the model, cross-validation and independent test datasets are used. From the downloaded protein sequences, 250 protein sequences are randomly selected that are not included in the training dataset to construct an independent test dataset. The stability of the model is tested by considering the predictive performance of all positive and negative sample independent test datasets. However, regardless of whether the performance of all training models is tested by knife-cutting tests or cross-validation tests, the training set being evaluated contains a 1 : 1 ratio of positive and negative samples.

2.2. CKSAAP Encoding. When it comes to the composition of k-spaced residue pairs (CKSAAP), such encoding method is a typical method to demonstrate the protein sequences [43–51]. The most significant parameter of this method is k, which means the number of amino acid residues between two target amino acid residues. So, the length of the target fragments is $k-2$. For instance, if k is equal to 0, it means there is no gap between two target amino acid residues. If window size is $2r+1$, it means the real effect-identified length is $2r-1$. There are 21 types of amino acids, which include 20 types of amino acid residues and a blank. "AxxA" can play the same method to two amino acid pairs. O means the none amino acid. Other characters mean 20 types of amino acid. The detailed description is shown in the following equation:

$$\left(\frac{N_{AA}}{N_{\text{total}}}, \frac{N_{AC}}{N_{\text{total}}}, \ldots, \frac{N_{\infty}}{N_{\text{total}}}\right)_{441}. \tag{1}$$

In this work, we select the final optimum window length as 27. The final feature scale is 2646, which means $21 \times (k_{max}+1) \times 21$. In other words, one sample may contain 2646 features and one label. The dataset may transform $n \times 2647$. n means the scale of samples, and 2647 is the sum of features and the label.

2.3. Binary Encoding. In order to obtain the position-specific information in the protein sequence level, one-dimensional binary encoding was utilized. In this work, we define 21 types amino acid, including 20 types amino acid residues and one blank residue, in the model. For instance, A amino acid can be shown as 100000000000000000000. Similarly, each of the other amino acids was coded in the same way. The final scale of features is $21 \times 26 = 546$ feature dimensions without considering the central k residue.

2.4. Random Forest. When it comes to the random forest classifier, it has been widely used in several areas. Compared with other algorithms, this algorithm has strong robustness in the presence of noise and outliers. Basically, the random forest classifier is based on a decision tree classifier, where each decision tree is trained using a randomly selected subset of samples, as described in the following [52–55]. First of all, if k features and the alternative sampling scheme can be submitted, n samples are selected randomly. And then, the best split node can be selected among the input features, which can be treated as the key step to design a decision tree. Last but not least, a decision tree is generated without pruning. Usually, when all individual trees provide votes, the maximum number of votes is considered to build the forest. In our study, RF predicted two categories of positive sites (ubiquitin sites) and negative sites (nonubiquitin sites) by voting on the number of trees. If the score is greater than or equal to 0.50, the lysine (k) position is declared positive. If the score is less than 0.50, the lysine (k) position is a ubiquitin site. Lysine sites with a score closer to 1 were more accurately defined as ubiquitin sites.

2.5. Model Training. As mentioned earlier, this study used three classification algorithms, namely, RF, SVM, and naïve Bayes, to construct predictors of ubiquitin sites. In order to train these classifiers, no matter whether their prediction accuracy can distinguish well between ubiquitin sites and nonubiquitin sites, the training dataset is used. In this paper, considering the optimal value of the parameters, the support vector machine kernel radial basis function is used to realize the radio frequency signal. Through different cross-validations, the model is trained with different ratios of positive and negative samples. Among the three classification methods, RF is considered to be the best method to classify ubiquitin sites and nonubiquitin sites. We employed 5-fold cross validating in the 1 : 1 ratio of positive and negative samples.

2.6. Model Performance Evaluation and Cross-Validation. In order to show the performance of the proposed ubiquitin site too, four typical performances were used, namely, sensitivity (Sn), specificity (Sp), accuracy (Acc), and Matthew correlation coefficient (MCC) [56–60], which can be formulated as follows:

$$\begin{aligned}
\text{accuracy (Acc)} &= \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{TN} + \text{FP}},\\
\text{sensitivity (Sn)} &= \frac{\text{TP}}{\text{TP} + \text{FN}},\\
\text{specificity (Sp)} &= \frac{\text{TN}}{\text{TN} + \text{FP}},\\
\text{Matthew correlation coefficient (MCC)} &= \frac{(\text{TP} \times \text{TN}) - (\text{FP} \times \text{FN})}{\sqrt{(\text{TP} + \text{FP}) \times (\text{TP} + \text{FN}) \times (\text{TN} + \text{FN}) \times (\text{TN} + \text{FP})}}.
\end{aligned} \tag{2}$$

TP, FP, FN, and TN represent true positive, false positive, false negative, and true negative, respectively. To provide a comprehensive performance measurement, we also consider receiver operating characteristic (ROC) curves, which are graphical representations of a function between true positive rate (i.e., sensitivity) and false positive rate (i.e., 1− specificity). In addition, the area under the ROC curve (AUC) is used to quantify the overall performance of the proposed method. The closer the AUC value is to 1, the better the performance is. In addition, folding knife and 2-fold, 5-fold cross-validation tests are also considered to examine the performance of the prediction model. Typically, in a 5-fold cross-validation test, it creates five subsets of datasets from approximately equal-sized training datasets. Four subgroups are used to train the classifier, and the remaining groups are used as test datasets to evaluate the performance of the classifier, and the classifier is therefore run five times. The prediction performance of independent test datasets also proves the performance of the model. The performance indicators Sn, Sp, and MCC are calculated at the threshold fpr = 0.20 (FP rate), while the AUC measure (the total area under the ROC curve) is calculated by the threshold-independent score.

3. Results and Discussion

3.1. Performance Assessment of the Models. In real situation, the positive samples and negative ones are not equal to 1 :1, but the imbalance situation may affect the accuracy of the classification. In order to overcome the shortcoming, we have selected three ratios, which are 1 : 1, 1 : 2, and 1 : 3, between positive and negative samples. In the following part, we show the results of the classification performances in different ratios, which include 1 : 1 in Figure 1, 1 : 2 in Figure 2, and 1 : 3 in Figure 3. At the same time, the RF model and other employed classification machine learning algorithms have been compared in these ratios.

From Table 1, we can easily get the conclusion that the neural network performances are 58.11% in Sp, 59.67% in Sn, 58.89% in Acc, and 0.1778 in MCC. The naïve Bayesian

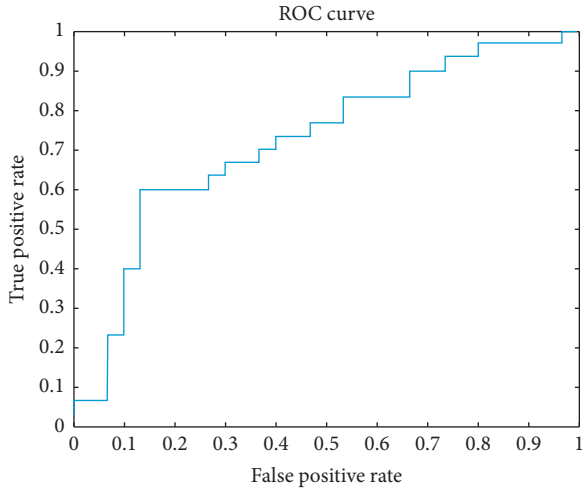

Figure 1: The ROC curve of the proposed method in 1 : 1 ratio.

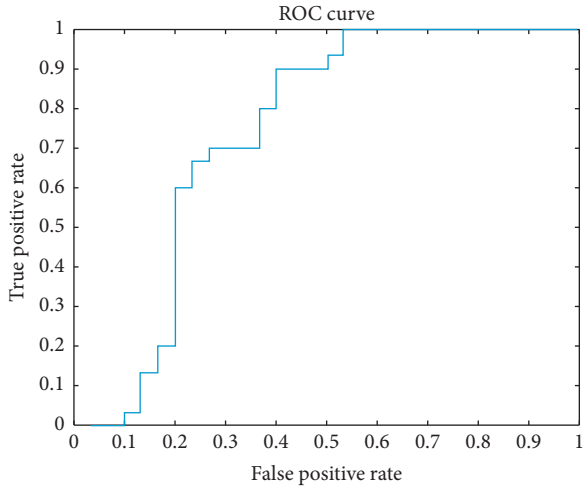

Figure 2: The ROC curve of the proposed method in 1 : 2 ratio.

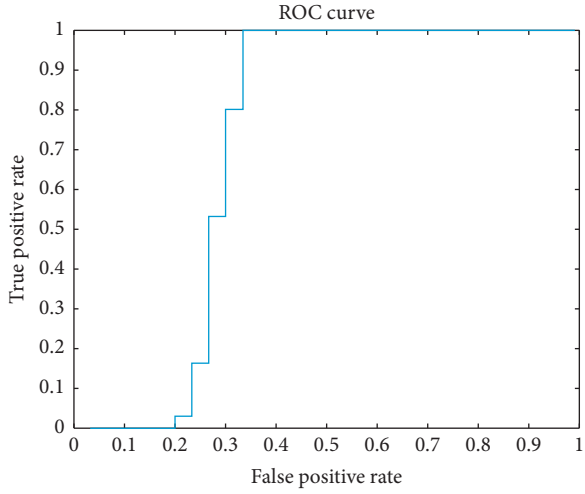

Figure 3: The ROC curve of the proposed method in 1 : 3 ratio.

Table 1: The performances in 1 : 2 ratio.

Algorithm	Sp (%)	Sn (%)	Acc (%)	MCC	F1 score
Neural network	59.84	68.46	64.15	0.2841	0.6254
Naïve Bayesian	34.31	77.07	55.69	0.1259	0.4364
Support vector machine	70.84	66.70	68.77	0.3757	0.6940
Random forest	87.65	54.15	70.90	0.4436	0.7507

model's performances are 37.08% in Sp, 75.89% in Sn, 56.49% in Acc, and 0.1407 in MCC. The support vector machine's performances are 65.29% in Sp, 59.16% in Sn, 62.23% in Acc, and 0.2450 in MCC. Finally, the random forest algorithm can obtain the results, including 72.54% in Sp, 53.84% in Sn, 63.19% in Acc, and 0.2685 in MCC.

From Table 2, we can easily get the conclusion that the neural network and naïve Bayesian model can hardly obtain the ideal performances compared to the support vector machine's and the random forest's ones. It was noted that the random forest can obtain the available results in the balance situation among the positive and negative samples.

From Table 3, we can easily get the conclusion that the neural network performances are 36.68% in Sp, 75.48% in Sn, 56.08% in Acc, and 0.1319 in MCC. The naïve Bayesian model's performances are 56.70% in Sp, 56.37% in Sn, 56.54% in Acc, and 0.1307 in MCC. The support vector machine's performances are 77.48% in Sp, 59.70% in Sn, 68.59% in Acc, and 0.3778 in MCC. Finally, the random forest algorithm can obtain the results, including 83.43% in Sp, 56.83% in Sn, 70.13% in Acc, and 0.4176 in MCC.

With the above performances, we can easily find that, with the ratio among positive and negative samples increasing, the performances of the random forest and other employed models can obtain the more available results. In other words, the proposed features and employed algorithms can be fitted in the real situation, in which the positive samples are far more lower than the negative ones.

In order to show the CKSAAP encoding is an effective feature of protein sequence processing, we have compared such features with several state-of-the-art features.

From Table 4, we can easily get the conclusion that the DNABIND performances are 61.10% in Sp, 61.96% in Sn, 61.53% in Acc, and 0.2305 in MCC. The DNAbinder performances are 56.66% in Sp, 57.73% in Sn, 54.70% in Acc, and 0.0941 in MCC. The DBD-Threader performances are 22.06% in Sp, 93.72% in Sn, 57.89% in Acc, and 0.2262 in MCC. The DNA-Prot performances are 62.56% in Sp, 47.61% in Sn, 55.09% in Acc, and 0.1029 in MCC. The iDNA-Prot performances are 62.07% in Sp, 59.71% in Sn, 60.89% in Acc, and 0.2179 in MCC. The PLMLA performances are 59.20% in Sp, 59.67% in Sn, 59.44% in Acc, and 0.1887 in MCC.

From Table 5, we can easily get the conclusion that the DNABIND performances are 63.98% in Sp, 65.13% in Sn, 64.55% in Acc, and 0.2911 in MCC. The DNAbinder performances are 52.96% in Sp, 60.62% in Sn, 56.79% in Acc, and 0.1361 in MCC. The DBD-Threader performances are 19.41%

Table 2: The performances in 1 : 1 ratio.

Algorithm	Sp (%)	Sn (%)	Acc (%)	MCC	F1 score
Neural network	58.11	59.67	58.89	0.1778	0.5857
Naïve Bayesian	37.08	75.89	56.49	0.1407	0.4601
Support vector machine	65.29	59.16	62.23	0.2450	0.6335
Random forest	72.54	53.84	63.19	0.2685	0.6634

Table 3: The performances in 1 : 3 ratio.

Algorithm	Sp (%)	Sn (%)	Acc (%)	MCC	F1 score
Neural network	36.68	75.48	56.08	0.1319	0.4551
Naïve Bayesian	56.70	56.37	56.54	0.1307	0.5661
Support vector machine	77.48	59.70	68.59	0.3778	0.7115
Random forest	83.43	56.83	70.13	0.4176	0.7364

Table 4: Performances of different features in 1 : 1 ratio.

Method	Sp (%)	Sn (%)	Acc (%)	MCC	F1 score
DNABIND	61.10	61.96	61.53	0.2305	0.6136
DNAbinder	51.66	57.73	54.70	0.0941	0.5328
DBD-Threader	22.06	93.72	57.89	0.2262	0.3438
DNA-Prot	62.56	47.61	55.09	0.1029	0.5821
iDNA-Prot	62.07	59.71	60.89	0.2179	0.6135
PLMLA	59.20	59.67	59.44	0.1887	0.5934
This method	72.54	53.84	63.19	0.2685	0.6634

Table 5: Performances of different features in 1 : 2 ratio.

Method	Sp (%)	Sn (%)	Acc (%)	MCC	F1 score
DNABIND	63.98	65.13	64.55	0.2911	0.6435
DNAbinder	52.96	60.62	56.79	0.1361	0.5507
DBD-Threader	19.41	91.60	55.50	0.1590	0.3037
DNA-Prot	63.71	52.07	57.89	0.1589	0.6021
iDNA-Prot	62.91	62.97	62.94	0.2588	0.6293
PLMLA	57.17	61.74	59.46	0.1894	0.5851
Random forest	87.65	54.15	70.90	0.4436	0.7507

in Sp, 91.60% in Sn, 55.50% in Acc, and 0.1590 in MCC. The DNA-Prot performances are 63.71% in Sp, 52.07% in Sn, 57.89% in Acc, and 0.1589 in MCC. The iDNA-Prot performances are 62.91% in Sp, 62.97% in Sn, 62.94% in Acc, and 0.2588 in MCC. The PLMLA performances are 57.17% in Sp, 61.74% in Sn, 59.46% in Acc, and 0.1894 in MCC.

From Table 6, we can easily get the conclusion that the DNABIND performances are 65.49% in Sp, 67.14% in Sn, 66.32% in Acc, and 0.3263 in MCC. The DNAbinder performances are 56.82% in Sp, 63.07% in Sn, 59.95% in Acc, and 0.1993 in MCC. The DBD-Threader performances are 22.69% in Sp, 93.26% in Sn, 57.98% in Acc, and 0.2252 in MCC. The DNA-Prot performances are 66.95% in Sp, 52.21% in Sn, 59.58% in Acc, and 0.1936 in MCC. The iDNA-Prot performances are 64.98% in Sp, 65.37% in Sn, 65.17% in Acc, and 0.3035 in MCC. The PLMLA performances are 59.80% in Sp, 63.86% in Sn, 61.83% in Acc, and 0.2368 in MCC.

Table 6: Performances of different features in 1 : 3 ratio.

Method	Sp (%)	Sn (%)	Acc (%)	MCC	F1 score
DNABIND	65.49	67.14	66.32	0.3263	0.6603
DNAbinder	56.82	63.07	59.95	0.1993	0.5865
DBD-Threader	22.69	93.26	57.98	0.2252	0.3506
DNA-Prot	66.95	52.21	59.58	0.1936	0.6235
iDNA-Prot	64.98	65.37	65.17	0.3035	0.6511
PLMLA	59.80	63.86	61.83	0.2368	0.6104
Random forest	83.43	56.83	70.13	0.4176	0.7364

4. Conclusion

Ubiquitin is an important type of protein after translational modification. Ubiquitin has the ability to take part in several cellular regulations among several biological processions. At the same time, ubiquitin plays key roles in the enzymatic process. So as to construct the new tool to classify the ubiquitin amino acid residues, we employed the random forest model to classify the ubiquitin sites utilizing the experimentally identified ubiquitinated protein sequences of *A. thaliana*. More detailed, we utilized the *k*-spaced amino acid pair (CKSAAP) encoding and binary encoding to deal with the potential protein segments. The proposed tools can obtain 72.83% in Sp, 72.42% in Sn, 72.63% in Acc, and 0.4525 in MCC. With these performances, such tools can obtain the available results in the dataset of *Arabidopsis*.

Data Availability

All the data used to support the findings of this study are included within the manuscript.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] T. Kouzarides, "Chromatin modifications and their function," *Cell*, vol. 128, no. 4, pp. 693–705, 2007.

[2] M. Mann and O. N. Jensen, "Proteomic analysis of post-translational modifications," *Nature Biotechnology*, vol. 21, no. 3, pp. 255–261, 2003.

[3] C. Dai and W. Gu, "p53 post-translational modification: deregulated in tumorigenesis," *Trends in Molecular Medicine*, vol. 16, no. 11, pp. 528–536, 2010.

[4] A. J. Ruthenburg, H. Li, D. J. Patel, and C. David Allis, "Multivalent engagement of chromatin modifications by linked binding modules," *Nature Reviews Molecular Cell Biology*, vol. 8, no. 12, pp. 983–994, 2007.

[5] J. Wysocka, T. Swigut, H. Xiao et al., "A PHD finger of NURF couples histone H3 lysine 4 trimethylation with chromatin remodelling," *Nature*, vol. 442, no. 7098, pp. 86–90, 2006.

[6] J. Wysocka, T. Swigut, T. A. Milne et al., "WDR5 associates with histone H3 methylated at K4 and is essential for H3 K4 methylation and vertebrate development," *Cell*, vol. 121, no. 6, pp. 859–872, 2005.

[7] L. Zeng and M.-M. Zhou, "Bromodomain: an acetyl-lysine binding domain," *FEBS Letters*, vol. 513, no. 1, pp. 124–128, 2002.

[8] T. Jenuwein and C. D. Allis, "Translating the histone code," *Science*, vol. 293, no. 5532, pp. 1074–1080, 2001.

[9] R. Marmorstein and S. Y. Roth, "Histone acetyltransferases: function, structure, and catalysis," *Current Opinion in Genetics & Development*, vol. 11, no. 2, pp. 155–161, 2001.

[10] A. M. Bode and Z. Dong, "Post-translational modification of p53 in tumorigenesis," *Nature Reviews Cancer*, vol. 4, no. 10, pp. 793–805, 2004.

[11] G. Walsh and R. Jefferis, "Post-translational modifications in the context of therapeutic proteins," *Nature Biotechnology*, vol. 24, no. 10, pp. 1241–1252, 2006.

[12] S. Westermann and K. Weber, "Post-translational modifications regulate microtubule function," *Nature Reviews Molecular Cell Biology*, vol. 4, no. 12, pp. 938–948, 2003.

[13] C. Janke and J. Chloë Bulinski, "Post-translational regulation of the microtubule cytoskeleton: mechanisms and functions," *Nature Reviews Molecular Cell Biology*, vol. 12, no. 12, pp. 773–786, 2011.

[14] Y. Xu, X. Shao, L. Wu, N. Deng, and K. Chou, "iSNO-AAPair: incorporating amino acid pairwise coupling into PseAAC for predicting cysteine S-nitrosylation sites in proteins," *PeerJ*, vol. 1, 2013.

[15] W. Qiu, X. Xiao, W. Lin, and K. Chou, "iMethyl-PseAAC: identification of protein methylation sites via a pseudo amino acid composition approach," *BioMed Research International*, vol. 2014, Article ID 947416, 12 pages, 2014.

[16] Y. Xu, X. Wen, X.-J. Shao, N.-Y. Deng, and K.-C. Chou, "iHyd-PseAAC: predicting hydroxyproline and hydroxylysine in proteins by incorporating dipeptide position-specific propensity into pseudo amino acid composition," *International Journal of Molecular Sciences*, vol. 15, no. 5, pp. 7594–7610, 2014.

[17] X. Xiao, H.-X. Ye, Z. Liu, J.-H. Jia, and K.-C. Chou, "iROS-gPseKNC: predicting replication origin sites in DNA by incorporating dinucleotide position-specific propensity into general pseudo nucleotide composition," *Oncotarget*, vol. 7, no. 23, pp. 34180–34189, 2016.

[18] W. Chen, H. Tang, J. Ye, H. Lin, and K. Chou, "iRNA-PseU: identifying RNA pseudouridine sites," *Molecular Therapy Nucleic Acids*, vol. 5, 2016.

[19] J. Jia, Z. Liu, X. Xiao, B. Liu, and K.-C. Chou, "iCar-PseCp: identify carbonylation sites in proteins by Monte Carlo sampling and incorporating sequence coupled effects into general PseAAC," *Oncotarget*, vol. 7, no. 23, pp. 34558–34570, 2016.

[20] J. Jia, L. Zhang, Z. Liu, X. Xiao, and K.-C. Chou, "pSumo-CD: predicting sumoylation sites in proteins with covariance discriminant algorithm by incorporating sequence-coupled effects into general PseAAC," *Bioinformatics*, vol. 32, no. 20, pp. 3133–3141, 2016.

[21] Z. Liu, X. Xiao, D.-J. Yu, J. Jia, W.-R. Qiu, and K.-C. Chou, "pRNAm-PC: predicting N6-methyladenosine sites in RNA sequences via physical-chemical properties," *Analytical Biochemistry*, vol. 497, pp. 60–67, 2016.

[22] W.-R. Qiu, B.-Q. Sun, X. Xiao, Z.-C. Xu, and K.-C. Chou, "iPTM-mLys: identifying multiple lysine PTM sites and their different types," *Bioinformatics*, vol. 32, no. 20, pp. 3116–3123, 2016.

[23] W.-R. Qiu, X. Xiao, Z.-C. Xu, and K.-C. Chou, "iPhos-PseEn: identifying phosphorylation sites in proteins by fusing different pseudo components into an ensemble classifier," *Oncotarget*, vol. 7, no. 32, pp. 51270–51283, 2016.

[24] P. Feng, H. Ding, H. Yang, W. Chen, H. Lin, and K.-C. Chou, "iRNA-PseColl: identifying the occurrence sites of different RNA modifications by incorporating collective effects of nucleotides into PseKNC," *Molecular Therapy - Nucleic Acids*, vol. 7pp. 155–163, C, 2017.

[25] W. Bao, Z. Huang, C. A. Yuan, and D. S. Huang, "Pupylation sites prediction with ensemble classification model," *International Journal of Data Mining and Bioinformatics*, vol. 18, no. 2, pp. 91–104, 2017.

[26] W.-R. Qiu, S.-Y. Jiang, Z.-C. Xu, X. Xiao, and K.-C. Chou, "iRNAm5C-PseDNC: identifying RNA 5-methylcytosine sites by incorporating physical-chemical properties into pseudo dinucleotide composition," *Oncotarget*, vol. 8, no. 25, pp. 41178–41188, 2017.

[27] W. R. Qiu, B. Q. Sun, X. Xiao, D. Xu, and K. C. Chou, "iPhos-PseEvo: identifying human phosphorylated proteins by incorporating evolutionary information into general PseAAC via grey system theory," *Molecular Informatics*, vol. 36, pp. 5-6, 2017.

[28] W. R. Qiu, B. Q. Sun, X. Xiao, Z. C. Xu, J. H. Jia, and K. C. Chou, "iKcr-PseEns: identify lysine crotonylation sites in histone proteins with pseudo components and ensemble classifier," *Genomics*, vol. 110, no. 5, pp. 239–246, 2017.

[29] Y. Xu, Z. Wang, C. Li, and K. C. Chou, "iPreny-PseAAC: identify C-terminal cysteine prenylation sites in proteins by incorporating two tiers of sequence couplings into PseAAC," *Medicinal Chemistry*, vol. 13, no. 6, p. 544, 2017.

[30] W. Bao, Z. Jiang, and D. S. Huang, "Novel human microbe-disease association prediction using network consistency projection," *BMC Bioinformatics*, vol. 18, no. Suppl 16, p. 543, 2017.

[31] K.-C. Chou, "Prediction of human immunodeficiency virus protease cleavage sites in proteins," *Analytical Biochemistry*, vol. 233, no. 1, pp. 1–14, 1996.

[32] Y. D. Khan, N. Rasool, W. Hussain, S. A. Khan, and K. C. Chou, "iPhosT-PseAAC: identify phosphothreonine sites by incorporating sequence statistical moments into PseAAC," *Analytical Biochemistry*, vol. 550, 2018.

[33] B. Liu, F. Liu, X. Wang, J. Chen, L. Fang, and K.-C. Chou, "Pse-in-One: a web server for generating various modes of pseudo components of DNA, RNA, and protein sequences," *Nucleic Acids Research*, vol. 43, no. W1, pp. W65–W71, 2015.

[34] K. C. Chou, "Impacts of bioinformatics to medicinal chemistry," *Medicinal Chemistry*, vol. 11, no. 3, 2015.

[35] L. F. Yuan, C. Ding, S. H. Guo, W. Chen, and H. Lin, "Prediction of the types of ion channel-targeted conotoxins based on feature selection techniques," *Journal of Biomathematics*, 2013, http://en.cnki.com.cn/Article_en/CJFDTOTAL-SWSX201304019.htm.

[36] K. C. Chou, "An unprecedented revolution in medicinal chemistry driven by the progress of biological science," *Current Topics in Medicinal Chemistry*, vol. 17, no. 21, pp. 2337–2358, 2017.

[37] K. C. Chou, "Pseudo amino acid composition and its applications in bioinformatics, proteomics and system biology," *Current Proteomics*, vol. 6, no. 4, 2009.

[38] C. Wei, L. Hao, and K. C. Chou, "Pseudo nucleotide composition or PseKNC: an effective formulation for analyzing genomic sequences," *Molecular Biosystems*, vol. 11, no. 10, pp. 2620–2634, 2015.

[39] K.-C. Chou, "Prediction of signal peptides using scaled window," *Peptides*, vol. 22, no. 12, pp. 1973–1979, 2001.

[40] W. Chen, P.-M. Feng, H. Lin, and K.-C. Chou, "iRSpot-PseDNC: identify recombination spots with pseudo dinucleotide composition," *Nucleic Acids Research*, vol. 41, no. 6, p. e68, 2013.

[41] X. Cheng, X. Xiao, and K. C. Chou, "pLoc-mPlant: predict subcellular localization of multi-location plant proteins by incorporating the optimal GO information into general PseAAC," *Gene*, vol. 13, no. 9, 2017.

[42] X. Cheng, S.-G. Zhao, W.-Z. Lin, X. Xiao, and K.-C. Chou, "pLoc-mAnimal: predict subcellular localization of animal proteins with both single and multiple sites," *Bioinformatics*, vol. 33, no. 22, p. 3524, 2017.

[43] X. Cheng, X. Xiao, and K. C. Chou, "pLoc-mGneg: predict subcellular localization of Gram-negative bacterial proteins by deep gene ontology learning via general PseAAC," *Genomics*, vol. 110, no. 4, pp. 231–239, 2018.

[44] C. Xiang, X. Xuan, and K. C. Chou, "pLoc-mEuk: predict subcellular localization of multi-label eukaryotic proteins by extracting the key GO information into general PseAAC," *Genomics*, vol. 110, no. 1, pp. 50–58, 2018.

[45] B. Wenzheng, C. Yuehui, and W. Dong, "Prediction of protein structure classes with flexible neural tree," *Bio-medical Materials and Engineering*, vol. 24, no. 6, pp. 3797–3806, 2014.

[46] W. Bao, D. Wang, and Y. Chen, "Classification of protein structure classes on flexible neutral tree," *IEEE/ACM Transactions on Computational Biology & Bioinformatics*, vol. 14, no. 5, p. 1, 2017.

[47] Y. Chen, B. Yang, J. Dong, and A. Abraham, "Time-series forecasting using flexible neural tree model," *Information Sciences*, vol. 174, no. 3-4, pp. 219–235, 2005.

[48] Y. Chen, A. Abraham, and B. Yang, "Hybrid flexible neural-tree-based intrusion detection systems," *International Journal of Intelligent Systems*, vol. 22, no. 4, pp. 337–352, 2010.

[49] Y. Chen, A. Abraham, and B. Yang, "Feature selection and classification using flexible neural tree," *Neurocomputing*, vol. 70, no. 1–3, pp. 305–313, 2006.

[50] A. Szilágyi and J. Skolnick, "Efficient prediction of nucleic acid binding function from low-resolution protein structures," *Journal of Molecular Biology*, vol. 358, no. 3, pp. 922–933, 2006.

[51] K. K. Kumar, G. Pugalenthi, and P. N. Suganthan, "DNA-Prot: identification of DNA binding proteins from protein sequence information using random forest," *Journal of Biomolecular Structure and Dynamics*, vol. 26, no. 6, pp. 679–686, 2009.

[52] W. Z. Lin, J. A. Fang, X. Xiao, and K. C. Chou, "iDNA-Prot: identification of DNA binding proteins using random forest with grey model," *PLoS One*, vol. 6, no. 9, Article ID e24756, 2011.

[53] L. Song, D. Li, X. Zeng, Y. Wu, L. Guo, and Q. Zou, "nDNA-prot: identification of DNA-binding proteins based on unbalanced classification," *BMC Bioinformatics*, vol. 15, no. 1, p. 298, 2014.

[54] S.-P. Shi, J.-D. Qiu, X.-Y. Sun, S.-B. Suo, S.-Y. Huang, and R.-P. Liang, "PLMLA: prediction of lysine methylation and lysine acetylation by combining multiple features," *Molecular Biosystems*, vol. 8, no. 5, pp. 1520–1527, 2012.

[55] G. Florian, R. Shubin, C. Chunaram, C. Jürgen, and M. Matthias, "Predicting post-translational lysine acetylation using support vector machines," *Bioinformatics*, vol. 26, no. 13, p. 1666, 2010.

[56] L. Songling, L. Hong, L. Mingfa, S. Yu, X. Lu, and L. Yixue, "Improved prediction of lysine acetylation by support vector machines," *Protein & Peptide Letters*, vol. 16, no. 8, 2009.

[57] Y. Xu, X.-B. Wang, J. Ding, L.-Y. Wu, and N.-Y. Deng, "Lysine acetylation sites prediction using an ensemble of support vector machine classifiers," *Journal of Theoretical Biology*, vol. 264, no. 1, pp. 130–135, 2010.

[58] S. B. Suo, "Position-specific analysis and prediction for protein lysine acetylation based on multiple features," *Plos One*, vol. 7, no. 11, Article ID e49108, 2012.

[59] J. Shao, D. Xu, L. Hu et al., "Systematic analysis of human lysine acetylation proteins and accurate prediction of human lysine acetylation through bi-relative adapted binomial score Bayes feature representation," *Molecular Biosystems*, vol. 8, no. 11, pp. 2964–2973, 2012.

[60] Y. Li, "Accurate in silico identification of species-specific acetylation sites by integrating protein sequence-derived and functional features," *Scientific Reports*, vol. 4, p. 5765, 2014.