Hindawi
Scientific Programming
Volume 2020, Article ID 8894633, 7 pages
https://doi.org/10.1155/2020/8894633

Research Article

Use Chou's 5-Step Rule to Classify Protein Modification Sites with Neural Network

Chuandong Song and Bin Yang

School of Information Science and Engineering, Zaozhuang University, Zaozhuang, Shandong 277160, China

Correspondence should be addressed to Bin Yang; batsi@126.com

Received 15 April 2020; Revised 15 June 2020; Accepted 17 June 2020; Published 3 July 2020

Academic Editor: Chenxi Huang

Lysine malonylation is a novel-type protein post-translational modification and plays essential roles in many biological activities. Having a good knowledge of malonylation sites can provide guidance in many issues, including disease prevention and drug discovery and other related fields. There are several experimental approaches to identify modification sites in the field of biology. However, these methods seem to be expensive. In this study, we proposed malNet, which employed neural network and utilized several novel and effective feature description methods. It was pointed that ANN's performance is better than other models. Furthermore, we trained the classifiers according to an original crossvalidation method named Split to Equal validation (SEV). The results achieved AUC value of 0.6684, accuracy of 54.93%, and MCC of 0.1045, which showed great improvement than before.

1. Introduction

Protein post-translational modification (PTM) is a key mechanism to regulate protein functions by the covalent and generally enzymatic modification. Hundreds of types of PTMs have been discovered and reported in this field [1–6]. They played vital roles in influencing almost all aspects of cell biology and pathogenesis, e.g., gene expression, cell division, and cell signaling [7–10]. As one of a newly identified PTM type in both eukaryotic and prokaryotic, Lysine malonylation (Kmal) has wide connections with various biological processes, where some Kmal sites are potentially associated with cancer. Therefore, it is very critical to identify and understand Kmal sites in the studies of biology and diseases [11–16].

Different from traditional experimental methods, computational approach of PTMs provides a fast and low-cost strategy for experimental designing, as the PTM site prediction can be abstracted as a typical classification problem. Meanwhile, there are a list of machine learning approaches which can be successful utilized in this field. For instance, Logistic Regression (LR) was used in ModPred for 23 different modifications using sequence-based features, physicochemical properties, and evolutionary features as features [17]. Musite, which is a general and kinase-specific protein phosphorylation site prediction, applied Support Vector Machine (SVM) models utilizing three types of features: K-nearest neighbor score, disorder scores, and amino acid frequencies. In the previous work, Wei et al. presented PhosPred-RF for predicting phosphorylation sites, which utilized the evolutionary information features from position specific scoring matrices [18, 19]. Deep learning method was also applied in this area, such as the recently published tool MusiteDeep, which was utilized for general and kinase-specific phosphorylation site prediction [20].

In this article, we employed artificial neural network (ANN) classifier based on Stochastic Gradient Descent (SGD) algorithm for protein Kmal site prediction. It was pointed that we investigated a wide range of types of feature extraction schemes and finally choose EBAG + Profile and EAAC methods to train our predictors. Furthermore, we employed another two classifiers, including SVM and kNN for comparative experiments, with the same feature extraction schemes. Besides, in view of the fact that the Kmal prediction problem can be regarded as a binary classification problem, we adopted the original SEV method to solve the inherent imbalance problem of positive and negative

samples in the training set. The result of our experiment shows ANN performed better than SVM and kNN predictors. Overall, ANN can be a useful tool for identifying Kmal sites.

2. Methods and Materials

There are 4 steps in our research, which is depicted in Figure 1. The first step is dataset construction and procession, where the training set and testing set were generated. And then we encode the dataset according to two feature extraction methods. The next step is to construct three models, which were trained by the training set. Finally, all the classifiers would be tested by crossvalidation and independent testing set. Five assessment metrics would be used to evaluate the performance of our classifiers.

2.1. Dataset Construction. In this work, we derived lots of Kmal peptides from mice and human species according to a proteomic assay. Referring to the procedure established by Chen et al. [21], we built a benchmark dataset. There are 67322 Kmal sites in the training set, where the sites with high confidence were regarded as positive sites and other lysine residues were collected as negative sites. For each sample, we extracted 31-residue peptides (−15 to +15) with the lysine site in the center from the representatives. As a result, 5023 positive peptides and 62299 negative peptides were retained for further analyses. We can easily find that the ratio of positive and negative samples in training set approaches 1 to 12. Therefore, the trained models would be tested by Split to Equal validation and independent test, where 35955 peptides (including 2798 positive peptides and 33157 negative peptides) were employed as the independent testing dataset.

2.2. Feature Encodings

2.2.1. EBAG + Profile Encoding. EBAG + Profile encoding is an integration scheme consisted of two different feature encoding method utilized by Han et al. [22]. One is Encoding Based on Attribute Grouping (EBAG) [23], which divides 20 types of amino acids into 5 groups depending on various physical and chemical properties. Table 1 shows the grouping result based on EBAG.

The other encoding method is Profile, which counts the frequency of each amino acid residue occurred in the protein peptides. And then the frequency was used as the representation of this residue in the sequence so that each peptide with 31 residues can be transformed into a 31-dimension vector. The way to combine EBAG and Profile is replacing source amino acid peptides into EGBA sequence and then encoding the sequence according to the Profile method. As a result, a peptide with 31 residues was converted to a vector of 31 dimensions as the EBAG + Profile encodings.

2.2.2. EAAC Encoding. A typical encoding scheme of prediction for PTMs was AAC encoding [24], which reflects the frequency of 20 amino acid residues surrounding the modification site. In this work, we coded each amino acid by

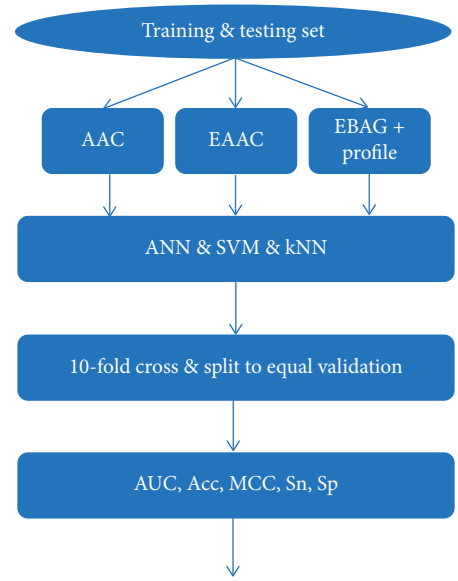

Figure 1: The working flow of our works.

Table 1: Groups of amino acid residues according to EBAG + Profile encoding.

Group	Amino acid residue	Label
C1	*A, F, G, I, L, M, P, V, W*	Hydrophobic
C2	*C, N, Q, S, T, Y*	Polar
C3	*D, E*	Acidic
C4	*H, K, R*	Basic
C5	*X*	Gaps

the EAAC method proposed by Zhen et al. [25], which is based on the AAC encoding. As 8-size window continuously slides from the N-terminus to C-terminus of each peptide in the dataset, the EAAC method counted the frequency of the 20 amino acid residues. Accordingly, the dimension of features can be calculated as follows:

$$N_s = L_p - L_s + 1,$$
$$D_{\text{eaac}} = N_s \times 20, \quad (1)$$

where L_p refers to the length of each peptide, L_s is the length of sliding windows, and D_{eaac} is the dimension of feature vector. As we set L_s to 8, a peptide with 31 residues would be corresponded to 24 $(31 - 8 + 1)$ sliding windows and converted to a matrix of 24×20 dimensions.

2.3. Construction of Classifiers

2.3.1. Artificial Neural Network. ANN is a traditional machine learning algorithm that was widely utilized in lysine PTM prediction applications. In this article, we construct an ANN model with four layers, i.e., input layer, output layer, and two hidden layers. The input layer received the feature sequence generated from different encoding method. The two hidden layers owe both 100 neurons and adopt "reLu" as their activation function. The output layer owes a single unit, outputting the probability score of each site.

2.3.2. Support Vector Machine. SVM is a well-established and commonly employed algorithm based on structural risk minimization from statistical learning theory [20]. SVM can transform the samples into a high-dimensional feature space and then construct an Optimal Separating Hyperplane (OSH) to maximize its distance from the closest training samples. Here, based on Tensorflow [26] and Scikit-learn [27], we employed SCV as our SVM model, where the applied kernel function was linear kernel.

2.3.3. K-Nearest Neighbor Algorithm. kNN algorithm is another widely employed algorithm that calculates the distances of samples to cluster them [28]. If we obtain the training dataset $D = \{v_1, v_2, ..., v_n\}$and a testing sample x, we can utilize KNN to calculate the distances between x and all the instances in D. Therefore, as the nearest neighbor (shortest distance) in the training dataset, the query sample will be assigned to the same class. In this work, we also construct a kNN model implemented by Tensorflow and Scikit-learn. The parameters of our kNN were set to their default values.

3. Crossvalidation Methods

In general, when the classification model is built, researchers will divide the dataset into two parts as training set and testing set. Process of dataset usage partition is depicted in Figure 2. To make full use of the training set samples, we usually train the model through 10-fold crossvalidation. The samples in training set are divided into training set and validating set by the crossvalidation method. And then, the training set is used to train the model, while the validating set is used to verify the effect of the model and obtain the validation scores. After the crossvalidation is completed, the trained model will pass the testing set to evaluate its performance and get the testing scores.

In view of the fact that the classifier is always more sensitive to the category containing more samples and less sensitive to the category containing fewer samples in binary classification problems, it is necessary to preprocess the training set before inputting the data with unbalanced positive and negative samples into the classifier. In the previous work, we proposed a new feature extraction method named SEV (Split to Equal Validation), which can well solve the problem of imbalanced training samples in PTM sites prediction research studies. In this experiment, we also adopt the SEV method and at the same time used 10-fold crossvalidation to do comparative experiments. The working flow of SEV is as follows.

(Note: the pos means the positive and the neg means the negative in Figure 3.)

The experiment shown in Figure 3 is a part of the whole experiment, which corresponds to the process of using training set to obtain classifier and its validation scores after the dataset is divided into training set and testing set in Figure 3. In details, SE validation consists of five steps. Assuming that the ratio of negative samples to positive samples in the training set is close to n:1, (1) the first step is to divide the negative samples into n groups; (2) in the second step, each positive sample is combined with the positive

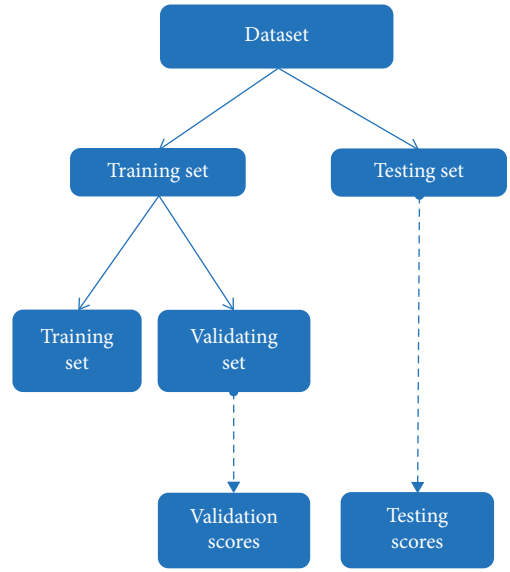

Figure 2: Process of dataset usage partition.

sample to generate n balanced subsets; (3) subsequently, model 1 will be trained by subset 1 and verified by subset 2; model 2 will be trained by subset 2 and verified by subset 3, and so on; (4) according to the n balanced subsets, n models were trained and validated; (5) finally, each model will be tested by independent testing sets, and the average of their scores will be utilized to evaluate their performance.

3.1. Performance Assessment of Predictors. There are a set of four metrics [29] directly that are often utilizing to quantitatively evaluate the performance of predictors: Sn (sensitivity), also known as TPR (True Positive Rate), reflects the proportion of true positive samples (TP) determined by the model to all the positive samples in the dataset; Sp (specificity), also known as TNR, reflects the proportion of true negative samples judged by the model in all negative samples; Acc (accuracy) is the proportion of correct samples determined by the model to the total samples; and MCC (Mathew's Correlation Coefficient) reflects the correlation coefficient between the actual predicted samples and the expected predicted samples:

$$\begin{aligned}
\mathrm{Sn} &= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \\
\mathrm{Sp} &= \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}}, \\
\mathrm{Acc} &= \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{TN} + \mathrm{FP}}, \\
\mathrm{MCC} &= \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP}) \times (\mathrm{TP} + \mathrm{FN}) \times (\mathrm{TN} + \mathrm{FN}) \times (\mathrm{TN} + \mathrm{FP})}}, \\
F1 &= \frac{2 * \mathrm{pre} * \mathrm{rec}}{\mathrm{pre} + \mathrm{rec}},
\end{aligned} \tag{2}$$

where TP, FP, TN, and FN represent the true positives, false positives, false negatives, and true negatives, respectively.

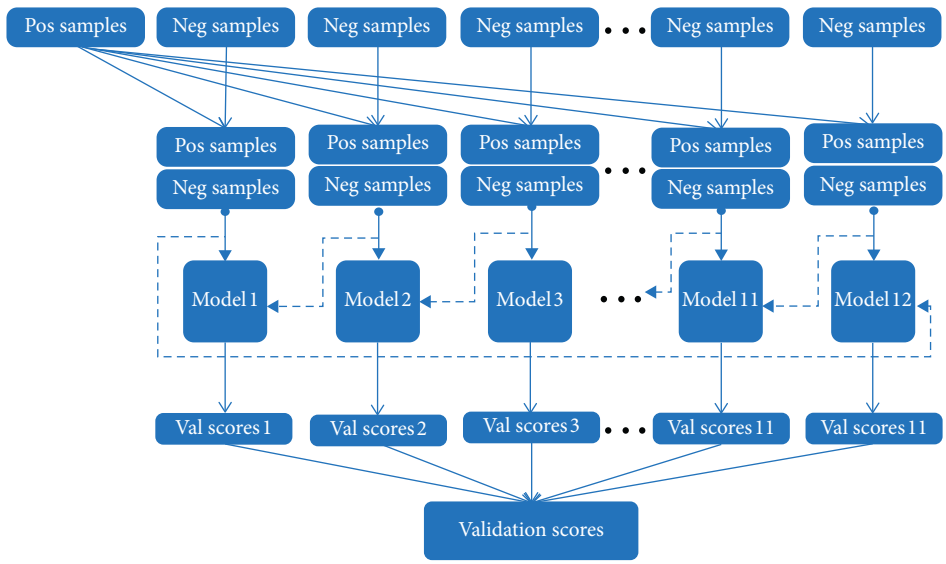

Figure 3: The working flow of Split to Equal validation.

The pre means precision and the rec means recall in the classification model. On the contrary, ROC curves and AUC value were also adopted to evaluate the performance of the predictors.

4. Results and Discussion

4.1. Performance of the Three Classification Models Based on Different Encoding Schemes. In this study, we firstly constructed three machine learning models, i.e., ANN, SVM, and kNN algorithm, and then trained them according to Amino Acid Composition (AAC) encoding scheme that considered the hydrophobicity and charged character of the amino acid. Split to Equal Validation (SEV) and independent training sets were utilized to assess the performance of models above, where AUC, Acc, MCC, Sn, and Sp were adopted as assessment metrics. The results of the independent testing results were depicted in Table 2.

Based on the results of this experiment, we speculate that feature extraction schemes are very important factors affecting the final classification accuracy. Therefore, we adopt the EBAG + Profile encoding method, which utilized the physical and chemical properties of amino acids. EAAC encoding method, which is based on AAC encoding and accords to the probability of occurrence of specific amino acids in the peptide sequence, was also adopted in this experiment. The testing scores were depicted in Tables 3 and 4, respectively.

As we know, a larger AUC value means that the current classification algorithm is more likely to rank positive samples in front of negative samples, so as to get better classification results. Therefore, it is obvious that classifiers under EAAC encoding scheme plays better performance than the other two schemes for getting higher AUC values. And other experimental results such as MCC and Acc value and EAAC encoding also obtained higher scores and showed similar advantages than others.

On the contrary, there is one thing certain that ANN's classification effect is better than SVM and kNN under EAAC encoding scheme. As for independent test, when taking EAAC, the AUC value of ANN is **0.7471**, while SVM and kNN algorithms obtain the AUC value of 0.6322 and 0.6317. All of these results in Tables 2–4 show that different types of classifiers have great impact on the prediction performance. In this work, ANN is the best classifier.

4.2. Comparing the Results of SEV with 10-Fold Cross-validation and Scaling the Number of Training Samples by SEV. In this research, we utilize the SEV verification method to preprocess the training samples, and thus train the classifier model. In addition, on the premise that other conditions remain unchanged, we used 10-fold cross-validation instead of SEV to do the same experiment. The experimental results are shown in Table 5:

Among them, the experiment is based on the neural network model, using 10-fold crossvalidation and SEV methods, respectively. It can be seen that although the Acc equivalent of the 10-fold cross is too high, its AUC value does not perform well. This is because, in the case of extremely unbalanced positive and negative samples, the classifier will guess the kind of samples with higher probability in the training set, but its classification ability is not outstanding. The SEV verification method can overcome this problem well. Although Acc equivalence is not as good as 10-fold cross verification, SEV has more advantages in the AUC value, which can best represent the classification ability of the model in the real sense.

More importantly, in order to further explore the influence of imbalance between positive and negative samples in the training set, we used the SEV method to scale the training samples. We verified the fact that unbalanced training data would finally lead to very low Sn and very high Sp of the classifier, further causing these evaluation metrics to lose its significance. In order to further explore how far

Table 2: The testing results of three models based on AAC encoding.

AAC	Classifier	AUC	Acc (%)	MCC	Sn (%)	Sp (%)	*F*1
Testing scores	ANN	0.5833	54.94	0.0598	56.35	54.83	55.58
	SVM	0.6149	53.92	0.0856	62.86	53.17	57.61
	kNN	**0.6224**	47.64	0.0906	71.20	45.67	55.65

Table 3: The testing results of three models based on EBPR encoding.

EAPR	Classifier	AUC	Acc (%)	MCC	Sn (%)	Sp (%)	*F*1
Testing scores	ANN	**0.6552**	56.57	0.1226	67.23	55.68	60.91
	SVM	0.5041	82.71	0.0056	12.06	88.61	21.23
	kNN	0.5874	64.69	0.0705	46.38	66.21	54.55

Table 4: The testing results of three models based on EAAC encoding.

EAAC	Classifier	AUC	Acc (%)	MCC	Sn (%)	Sp (%)	*F*1
Testing scores	ANN	**0.7471**	63.54	0.2002	74.16	62.65	67.92
	SVM	0.6322	56.21	0.1028	63.61	55.59	59.33
	kNN	0.6317	43.60	0.0931	76.19	40.88	53.21

Table 5: The testing scores of 10-fold crossvalidation and Split to Equal Validation.

Scores	Validation methods	AUC	ACC (%)	MCC	SN (%)	SP (%)	F1
Validation scores	10F CV	0.5751	21.11	0.0776	95.23	15.14	26.13
	SEV	**0.8465**	64.35	0.3260	100.00	61.48	76.15
Testing scores	10F CV	0.6965	90.12	0.1060	10.91	96.73	19.61
	SEV	**0.7471**	63.54	0.2002	74.16	62.65	67.92

reaching the positive and negative sample ratios affect the classifier's performance, we calculate the five metrics, i.e., AUC, Acc, MCC, Sn, and Sp by adjusting different positive and negative sample ratios from 1 : 1 to 1 : 12 under SEV.

As can be seen from Table 6, as the proportion of positive and negative samples in the training set increases, the AUC value of the model gradually decreases from 0.7471 to about 0.7000, the MCC value gradually decreases to about 0.1300, while the ACC value continuously increases to 89.43%. This also verifies our previous conclusion: when the positive and negative samples of the training set are extremely unbalanced, the classifier will tend to guess the kind with more samples, but its classification effect is not good.

In addition, Table 6 also shows another information, that is, when the ratio of positive and negative samples in the training set reaches 1 : 9, the AUC value of the classifier will also tend to be stable, only fluctuating around 0.7000 without further decline. This means that, in this experiment, although the ratio of positive and negative samples has been changing towards a more unbalanced direction, the performance of the classifier will not decrease indefinitely, but will tend to be stable after reaching a certain threshold.

In a word, through this experiment, we can further verify that the positive and negative sample ratios have far-reaching influence on the results in the binary classification problem, and the SEV method can solve this problem well.

Table 6: The testing scores based on different proportion of positive and negative samples.

Scale-up ratio	AUC	Acc (%)	MCC	Sn (%)	Sp (%)	*F*1
1:1	0.7471	63.54	0.2002	74.16	62.65	67.92
1:2	0.7399	76.67	0.1982	52.99	78.65	63.32
1:3	0.7324	82.60	0.1828	38.61	86.28	53.35
1:4	0.7290	84.33	0.1689	32.65	88.64	47.72
1:5	0.7228	85.54	0.1623	28.93	90.26	43.82
1:6	0.7205	86.71	0.1476	24.13	91.94	38.23
1:7	0.7196	87.27	0.1499	23.12	92.62	37.00
1:8	0.7193	86.64	0.1539	25.18	91.77	39.52
1:9	0.7012	87.32	0.1340	20.91	92.86	34.13
1:10	0.6984	88.08	0.1382	19.61	93.80	32.44
1:11	0.6972	88.96	0.1337	16.87	94.97	28.65
1:12	0.7033	89.43	0.1292	15.10	95.64	26.08

5. Conclusions

The currently available PTM prediction approaches are mainly based on ML that requires preprocessing amino acid data into digital features. Here, we adopted two feature extraction schemes according to different ideas of physical and chemical characteristics and occurrence frequency and then constructed three ML classifiers, while the application of the SEV method solves the problem of sample imbalance in binary classification. The results not only showed that

feature extraction methods and classifier types play important roles on prediction results but also indicated the direction of our next work. In addition to proposing new feature encoding schemes, more classifiers can be utilized in this field, including depth learning (DL) classifiers such as CNN or RNN. In addition, the SEV method will be improved and applied to the new machine learning model. In total, the outstanding performance of ML in prediction of Kmal sites suggests that computational methods can be applied widely to this field.

Data Availability

The data utilized to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no competing interests.

Acknowledgments

This work was supported by the talent project of "Qingtan scholar" of Zaozhuang University, Shandong Provincial Natural Science Foundation, China (no. ZR2015PF007), the PhD research startup foundation of Zaozhuang University, and Zaozhuang University Foundation (nos. 2014BS13 and 2015YY02).

References

[1] C. Peng, Z. Lu, Z. Xie et al., "The first identification of lysine malonylation substrates and its regulatory enzyme," *Molecular & Cellular Proteomics*, vol. 10, no. 12, p. M111.012658, 2011.

[2] H.D. Matthew and D. Yingming, "Metabolic regulation by lysine malonylation, succinylation, and glutarylation," *Molecular & Cellular Proteomics Mcp*, vol. 14, 2015.

[3] H. Mujahid, X. Meng, S. Xing, X. Peng, C. Wang, and Z. Peng, "Malonylome analysis in developing rice (Oryza sativa) seeds suggesting that protein lysine malonylation is well-conserved and overlaps with acetylation and succinylation substantially," *Journal of Proteomics*, vol. 170, pp. 88–98, 2018.

[4] T. Arendt, H. G. Zveuintshva, and T. A. Lkontovich, "Dendritic changes in the basal nucleus of meynert and in the diagonal band nucleus in alzheimer's disease-a quantitative golgi investigation," *Neuroscience*, vol. 19, no. 4, pp. 1265–1278, 1986.

[5] X. Bao, Q. Zhao, T. Yang, Y. M. E. Fung, and X. D. Li, "A chemical probe for lysine malonylation," *Angewandte Chemie*, vol. 125, no. 18, pp. 4983–4986, 2013.

[6] P. Boevink, K. Oparka, C. S. Santa, B. Martin, A. Betteridge, and C. Hawes, "Stacks on tracks: the plant Golgi apparatus traffics on an actin/ER network," *Plant Journal for Cell & Molecular Biology*, vol. 15, pp. 441–447, 2010.

[7] M. Bretscher and S. Munro, "Cholesterol and the golgi apparatus," *Science*, vol. 261, no. 5126, pp. 1280-1281, 1993.

[8] M. Canuel, S. Lefrancois, J. Zeng, and C. R. Morales, "AP-1 and retromer play opposite roles in the trafficking of sortilin between the golgi apparatus and the lysosomes," *Biochemical and Biophysical Research Communications*, vol. 366, no. 3, pp. 724–730, 2008.

[9] K. Caroline, M. Katy, A. Shireen et al., "Foot-and-mouth disease virus replication sites form next to the nucleus and close to the golgi apparatus, but exclude marker proteins associated with host membrane compartments," *Journal of General Virology*, vol. 86, 2005.

[10] L. Citores, L. Bai, V. Sørensen, and S. Olsnes, "Fibroblast growth factor receptor-induced phosphorylation of STAT1 at the golgi apparatus without translocation to the nucleus," *Journal of Cellular Physiology*, vol. 212, no. 1, pp. 148–156, 2007.

[11] G. Werner and K. Werner, "Changes in the nucleus, endoplasmic reticulum, golgi apparatus, and acrosome during spermiogenesis in the waterstrider, Gerris najas deg. (Heteroptera: gerridae)," *International Journal of Insect Morphology and Embryology*, vol. 22, no. 5, pp. 521–534, 1993.

[12] W. G. Whaley and M. Dauwalder, "The golgi apparatus, the plasma membrane, and functional integration," *International Review of Cytology*, vol. 58, pp. 199–245, 1979.

[13] I. H. Witten and E. Frank, "Data mining: practical machine learning tools and techniques," *Acm Sigmod Record*, vol. 31, pp. 76-77, 2011.

[14] J.-Y. Xu, Z. Xu, Y. Zhou, and B.-C. Ye, "Lysine malonylome may affect the central metabolism and erythromycin biosynthesis pathway in saccharopolyspora erythraea," *Journal of Proteome Research*, vol. 15, no. 5, pp. 1685–1701, 2016.

[15] R. Yang, C. Zhang, R. Gao, and L. Zhang, "A novel feature extraction method with feature selection to identify golgi-resident protein types from imbalanced data," *International Journal of Molecular Sciences*, vol. 17, no. 2, p. 218, 2016.

[16] S. Ya and P.-F. Jiao, "Predicting golgi-resident protein types using pseudo amino acid compositions: approaches with positional specific physicochemical properties," *Journal of Theoretical Biology*, vol. 391, 2016.

[17] M. Bujnicki, Dunin-Horkawicz, de Stanislaw et al., "tRNA-modpred: a computational method for predicting posttranscriptional modifications in tRNAs," *Methods A Companion to Methods in Enzymology*, vol. 107, 2016.

[18] L. Wei, P. Xing, J. Tang, and Q. Zou, "PhosPred-RF: a novel sequence-based predictor for phosphorylation sites using sequential information only," *IEEE Transactions on Nanobioscience*, vol. 16, no. 4, pp. 240–247, 2017.

[19] S. Banerjee, S. Basu, D. Ghosh, and M. Nasipuri, "PhospredRF: prediction of protein phosphorylation sites using a consensus of random forest classifiers," in *Proceedings of the International Conference & Workshop on Computing & Communication*, Kassel, Germany, March 2015.

[20] D. Wang, S. Zeng, C. Xu et al., "MusiteDeep: a deep-learning framework for general and kinase-specific phosphorylation site prediction," *Bioinformatics*, vol. 33, 2017.

[21] C. Zhen, Z. Yuan, Z. Zhang, and J. Song, "Towards more accurate prediction of ubiquitination sites: a comprehensive review of current methods, tools and features," *Briefings in Bioinformatics*, vol. 4, 2014.

[22] R. Z. Han, D. Wang, Y. H. Chen, L. K. Dong, and Y. L. Fan, "Prediction of phosphorylation sites based on the integration of multiple classifiers," *Genetics & Molecular Research*, vol. 16, 2017.

[23] S. C. Bagley and R. B. Altman, "Characterizing the microenvironment surrounding protein sites," *Protein Science*, vol. 4, no. 4, pp. 622–635, 2008.

[24] L.-N. Wang, S.-P. Shi, H.-D. Xu, P.-P. Wen, and J.-D. Qiu, "Computational prediction of species-specific malonylation sites via enhanced characteristic strategy," *Bioinformatics*, vol. 33, p. btw755, 2016.

[25] C. Zhen, H. Ningning, H. Yu et al., "Integration of a deep learning classifier with A random forest approach for predicting malonylation sites," *Genomics Proteomics & Bioinformatics*, vol. 16, 2018.
[26] L. Rampasek and A. Goldenberg, "TensorFlow: biology's gateway to deep learning?" *Cell Systems*, vol. 2, no. 1, pp. 12–14, 2016.
[27] F. Pedregosa, G. Varoquaux, A. Gramfort et al., "Scikit-learn: machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
[28] Y. Cai, T. Huang, L. Hu, X. Shi, L. Xie, and Y. Li, "Prediction of lysine ubiquitination with mRMR feature selection and analysis," *Amino Acids*, vol. 42, no. 4, pp. 1387–1395, 2012.
[29] J. Chen, H. Liu, J. Yang, and K.-C. Chou, "Prediction of linear B-cell epitopes using amino acid pair antigenicity scale," *Amino Acids*, vol. 33, no. 3, pp. 423–428, 2007.