iUbiq-Lys: prediction of lysine ubiquitination sites in proteins by extracting sequence evolution information via a gray system model

Wang-Ren Qiu\textsuperscript{a}, Xuan Xiao\textsuperscript{a,b,d,\*}, Wei-Zhong Lin\textsuperscript{a} and Kuo-Chen Chou\textsuperscript{c,d}

\textsuperscript{a}Computer Department, Jing-De-Zhen Ceramic Institute, Jing-De-Zhen 333403, China; \textsuperscript{b}Information School, ZheJiang Textile & Fashion College, NingBo 315211, China; \textsuperscript{c}Center of Excellence in Genomic Medicine Research (CEGMR), King Abdulaziz University, Jeddah 21589, Saudi Arabia; \textsuperscript{d}Bioinformatics, Gordon Life Science Institute, Boston, MA 02478, USA

Received 5 September 2014; accepted 20 September 2014

As one of the most important posttranslational modifications (PTMs), ubiquitination plays an important role in regulating varieties of biological processes, such as signal transduction, cell division, apoptosis, and immune response. Ubiquitination is also named “lysine ubiquitination” because it occurs when an ubiquitin is covalently attached to lysine (K) residues of targeting proteins. Given an uncharacterized protein sequence that contains many lysine residues, which one of them is the ubiquitination site, and which one is of non-ubiquitination site? With the avalanche of protein sequences generated in the postgenomic age, it is highly desired for both basic research and drug development to develop an automated method for rapidly and accurately annotating the ubiquitination sites in proteins. In view of this, a new predictor called “iUbiq-Lys” was developed based on the evolutionary information, gray system model, as well as the general form of pseudo-amino acid composition. It was demonstrated via the rigorous cross-validations that the new predictor remarkably outperformed all its counterparts. As a web-server, iUbiq-Lys is accessible to the public at http://www.jci-bioinfo.cn/iUbiq-Lys. For the convenience of most experimental scientists, we have further provided a protocol of step-by-step guide, by which users can easily get their desired results without the need to follow the complicated mathematics that were presented in this paper just for the integrity of its development process.

Keywords: ubiquitination; Chou’s PseAAC; gray system model; SVM; web-server, iUbiq-Lys

1. Introduction

Posttranslational modifications (PTMs) of proteins are crucial for understanding the dynamic proteome and various signaling pathways or networks in cells (Aguilar & Wendland, 2003; Herrmann, Lerman, & Lerman, 2007; Hicke & Dunn, 2003; Saghatelian & Cravatt, 2005; Welchman, Gordon, & Mayer, 2005). It has been demonstrated that the change of the ubiquitination system is closely related with cellular transformation, immune response, and inflammatory response (Schwartz & Ciechanover, 1999). In fact, there is growing evidence to indicate that ubiquitination is one of the most important reversible protein PTMs. Owing to its intimate involvement in gene regulation (Figure 1), ubiquitination plays an important role in regulating a variety of biological processes, such as signal transduction (Reinstein & Ciechanover, 2006; Sun & Chen, 2004), cell division/mitosis, apoptosis, endocytosis and the degradation of protein (Hicke, 2001; Hoeller, Hecker, & Dikic, 2006; Pickart, 2001; Reinstein & Ciechanover, 2006), gene transcription, DNA repair and replication, intracellular trafficking, and virus budding (Haglund & Dikic, 2005).

Because of its importance and complexity, identification of ubiquitination sites in proteins is one of the greatest challenges in gaining a full understanding of the regulatory roles of ubiquitination regulation and the molecular mechanism of the ubiquitin system. However, it is both time-consuming and labor-intensive to use conventional experimental approaches to determine the potential ubiquitination sites. Fortunately, some studies showed that ubiquitination occurs when an ubiquitin is covalently attached to lysine (K) residues of targeting proteins (Haglund & Dikic, 2005; Hershko & Ciechanover, 1998; Radivojac et al., 2010; Tung & Ho, 2008). Based on this, several computational approaches were developed to predict lysine ubiquitination sites. Tung and Ho (2008) built a prediction model using a set of physicochemical properties. Radivojac et al. (2010) used the random forest algorithm to develop a predictor called UbPred. Cai et al. (2012) proposed a predictor based on the nearest neighbor algorithm, in which the feature set was constructed by extracting the conservation scores and disorder scores from protein sequences, followed by utilizing the maximum relevance and minimum redundancy principle to identify the key features

\*Corresponding author. Email: xxiao@gordonlifescience.org

© 2014 Taylor & Francis
Zhao, Li, Ma, and Yin (2011) offered a computational method to identify the lysine ubiquitination sites based on the ensemble approach. Chen et al. (2011) proposed a predictor called CKSAAP_UbSite using the composition of k-spaced amino acid pairs. Although each of the aforementioned methods has its own merit and did play a role in stimulating the development of this area, they all need improvement from one or more of the following aspects: (i) The benchmark data-set used by the previous investigators needs to be either updated by incorporating some new and experiment-confirmed data, or refined by removing redundancy and duplicate sequences. (ii) The prediction quality can be further improved by introducing the state-of-the-art machine learning techniques. (iii) All the samples should be formulated purely based on the sequence information alone to lift the limitation imposed by some of the existing methods that require the availability of the structural information as well. (iv) A user-friendly web-server should be established to maximize the convenience of the most experimental scientists and to complement with those existing methods which do not have a web-server or whose web-servers are currently not working.

The present study was initiated in an attempt to develop a new predictor for identifying the ubiquitination sites in proteins from the aforementioned four aspects.

As elaborated in (Chou, 2011) and carried out in a series of recent publications (Chen, Feng, & Deng, 2014; Chen, Feng, & Lin, 2013; Ding, Deng, Yuan, Liu, & Lin, 2014; Fan, Xiao, & Min, 2014; Guo, Deng, Xu, Ding, & Lin, 2014; Liu, Xu, Lan, Xu, & Zhou, 2014; Liu, Zhang et al. 2014; Qiu, Xiao, & Chou, 2014; Xu, Wen, Shao, & Deng, 2014; Xu, Wen, Wen, & Wu, 2014), to establish a really useful statistical predictor for a biological system, we need to consider the following procedures: (i) construct or select a valid benchmark data-set to train and test the predictor; (ii) formulate the biological samples with an effective mathematical expression that can truly capture their essence and intrinsic correlation with the attribute to be predicted; (iii) introduce or develop a powerful algorithm (or engine) to operate the prediction; (iv) properly perform cross-validation tests to objectively evaluate the anticipated accuracy; (v) establish a user-friendly web-server that is accessible to the public. Below, we are to describe how to deal with these steps one-by-one.

2. Materials and methods

2.1. Benchmark data-set

To develop a statistical predictor, it is fundamentally important to establish a reliable and stringent benchmark data-set to train and test the predictor. If the benchmark data-set contains some errors, the predictor trained by it must be unreliable and the accuracy tested by it would be completely meaningless. In this study, the benchmark data-set was derived from the Swiss-Prot database (version 2013_06) at http://www.ebi.ac.uk/uniprot. Only those proteins were collected that had clear experimental annotations about their ubiquitination sites.
For facilitating the description later, we adopted the peptide formulation by Chou in studying HIV protease cleavage sites (Chou, 1993), specificity of GalNAc-transferase (Chou, 1995b), and signal peptide cleavage sites (Chou, 2001b). According to Chou’s scheme, a potential ubiquitination peptide, that is a peptide with Lys (namely K) located at its center (Figure 2), can be expressed as

\[ P_\xi(\mathbb{K}) = R_{-\xi} R_{-(\xi-1)} \cdots R_{-2} R_{-1} K R_{+1} R_{+2} \cdots R_{+(\xi-1)} R_{+\xi} \]

where the subscript \( \xi \) is an integer (cf. Equation (1)), \( R_{-\xi} \) represents the \( \xi \)-th upstream amino acid residue from the center, \( R_{+\xi} \) the \( \xi \)-th downstream amino acid residue, and so forth (Figure 2). The \((2\xi + 1)\)-tuple peptides \( P_\xi(\mathbb{K}) \) can be further classified into the following categories:

\[ P_\xi(\mathbb{K}) = \begin{cases} P_\xi^+(\mathbb{K}), & \text{if its center is a ubiquitination site} \\ P_\xi^-(\mathbb{K}), & \text{otherwise} \end{cases} \]

where \( \in \) represents “a member of” in the set theory.

As pointed out by a comprehensive review (Chou & Shen, 2007), there is no need to separate a benchmark data-set into a training data-set and a testing data-set for examining the performance of a prediction method if it is tested by the jackknife test or subsampling (K-fold) cross-validation test. Thus, the benchmark data-set for the current study can be formulated as

\[ S_\xi = S_\xi^+ \cup S_\xi^- \]

where \( S_\xi^+ \) only contains the samples of \( P_\xi^+(\mathbb{K}) \), that is the ubiquitination peptides; \( S_\xi^- \) only contains the samples of \( P_\xi^-(\mathbb{K}) \), that is the non-ubiquitination peptide (cf. Equation (2)); and \( \cup \) represents the symbol for “union” in the set theory.

Since the length of the peptide \( P_\xi(\mathbb{K}) \) is \( 2\xi + 1 \) (Equation (1)), the benchmark data-set with different values of \( \xi \) will contain peptides of different numbers of amino acid residues, as formulated by

\[ S_\xi \text{ contains the peptides of } \begin{cases} 9 \text{ residues, when } \xi = 4 \\ 11 \text{ residues, when } \xi = 5 \\ 13 \text{ residues, when } \xi = 6 \\ 15 \text{ residues, when } \xi = 7 \\ \vdots \end{cases} \]

The detailed procedures to construct \( S_\xi \) are as follows. (i) Sliding a window of \((2\xi + 1)\) amino acids (Figure 3) along each of the protein sequences taken from a protein database collected were only those peptide segments with K (lysine) at the center, that is the potential ubiquitination-site-containing peptides. (ii) If the upstream or downstream in a protein was less than \( \xi \), the lacking residue was filled with the same residue of its closest neighbor. (iii) The peptide samples thus obtained were put into the positive subset \( S_\xi^+ \) if their centers have been experimentally annotated as the ubiquitination sites, otherwise into the negative subset \( S_\xi^- \). (iv) The peptide samples thus obtained were further subject to a screening procedure to winnow those that were identical to any other. (v) Excluded from the benchmark data-set were also those that were self-conflict, namely occurring in both ubiquitination subset \( S_\xi^+ \) and non-ubiquitination subset \( S_\xi^- \).

By following the aforementioned procedures, thirteen such benchmark datasets \( S_\xi(\xi = 3, 4, 5, \ldots, 15) \) had been constructed. However, it was observed via preliminary trials (Table 1) that when \( \xi = 6 \), that is the peptide samples concerned were formed by 13 residues, the corresponding results were most promising. Accordingly, we choose \( S_{\xi=6} \) as the benchmark data-set for further investigation. Such a choice is also quite consistent with the treatment of the previous investigators (Aguilar & Wendland, 2003; Cai et al., 2012; Chen et al., 2011; Haglund & Dikic, 2005; Herrmann et al., 2007; Herskho & Ciechanover, 1998; Hicke & Dunn, 2003; Hoeller et al., 2006; Pickart, 2001; Schwartz & Ciechanover, 1999; Welchman et al., 2005; Zhao et al., 2011).

### 2.2. PseAAC vector generated via gray system model

One of the most important but also most difficult problems in computational biology today is how to effectively formulate a biological sequence with a discrete...
model or a vector, yet still keep considerable sequence-order information. This is because all the existing operation engines, such as covariance discriminant (CD) (Chen, Lin, Feng, & Ding, 2012; Wang, Yang, & Xu, 2005), neural network (Feng, Cai, & Chou, 2005), support vector machine (SVM) (Chen et al., 2013; Feng, Chen, & Lin, 2013), random forest (Kandaswamy, Martinetz, Suganthan, Sridharan, & Pugalenthì, 2011; Lin, Fang, & Xiao, 2011), conditional random field (Xu, Ding, & Wu, 2013), nearest neighbor (NN) (Cai, 2003), K-nearest neighbor (KNN) (Shen, Yang, & Chou, 2006), OET-KNN (Shen, 2009), fuzzy K-nearest neighbor (Xiao, Min, & Wang, 2013), ML-KNN algorithm (Xiao, Wu, & Chou, 2011), and SLLE algorithm (Wang et al., 2005), can only handle vector but not sequence samples. However, a vector defined in a discrete model may totally miss the sequence-order information. To deal with such a dilemma, the approach of pseudo-amino acid composition (Chou, 2001a, 2005) or Chou’s PseAAC (Lin & Lapointe, 2013) was proposed. Ever since it was introduced in 2001 (Chou, 2001a), the concept of PseAAC has been rapidly penetrated into almost all the areas of computational proteomics, such as in identifying bacterial virulent proteins (Nanni, Lumini, Gupta, & Garg, 2012), predicting super-secondary structure (Zou, He, He, & Xia, 2011), predicting anticaner peptides (Hajisharifi, Piryaiee, Mohammad Beigi, Behbahani, & Mohabatkar, 2014), predicting protein subcellular location (Kandaswamy et al., 2010; Mei, 2012a; Zhang, Zhang, Yang, Zhao, & Pan, 2008), predicting membrane protein types (Chen & Li, 2013; Han, Yu, & Anh, 2014), discriminating outer membrane proteins (Hayat & Khan, 2012), analyzing genetic sequence (Georgiou, Karakasidis, & Megaritis, 2013), identifying cyclin proteins (Mohabatkar, 2010), predicting GABA(A) receptor proteins (Mohabatkar, Mohammad Beigi, & Esmaeili, 2011), identifying antibacterial peptides (Khosravian, Faramarzi, Beigi, Behbahani, & Mohabatkar, 2013), predicting anticancer peptides (Hajisharifi et al., 2014), identifying allergenic proteins (Mohabatkar, Beigi, Abdolahi, & Mohsenzadeh, 2013), predicting metalloprotease family (Mohammad Beigi, Behjati, & Mohabatkar, 2011), predicting protein structural class (Kong, Zhang, & Lv, 2014; Sahu & Panda, 2010), identifying GPCRs and their types (Zia Ur & Khan, 2012), identifying protein quaternary structural attributes (Sun et al., 2012), predicting protein submitochondria locations (Nanni & Lumini, 2008), and identifying risk type of human papillomaviruses (Esmaeili, Mohabatkar, & Mohsenzadeh, 2010), among many others (see a long list of references cited in a 2014 article (Du et al., 2014)). Recently, the concept of PseAAC was further extended to represent the feature vectors of DNA and nucleotides (Chen et al., 2013; Chen, Lei, Jin & Lin, 2014; Guo et al., 2014; Qiu et al., 2014), as well as other biological samples (see, e.g. (Jiang, Huang, Chen, & Gao, 2013)). Because it has been widely and increasingly used, in addition to the web-server “PseAAC” (Shen, 2008) built in 2008, recently, three types of powerful open access software, called “PseAAC-Builder” (Du et al., 2012), “propy” (Cao, Xu, & Liang, 2013), and “PseAAC-General” (Du et al., 2014), were established: the former two are for generating various modes of Chou’s special PseAAC, while the 3rd one for those of Chou’s general PseAAC.

According to a comprehensive review (Chou, 2011), PseAAC can be generally formulated as

$$\mathbf{P} = [\psi_1, \psi_2, \cdots, \psi_n, \cdots, \psi_\Omega]^T$$  \hspace{1cm} (5)

where $\mathbf{T}$ is the transpose operator, while $\Omega$ an integer to reflect the vector’s dimension. The value of $\Omega$ as well as the components $\psi_n \ (u = 1, 2, \cdots, \Omega)$ in Equation (5) will depend on how to extract the desired information from a protein/peptide sequence. Below, we are to describe how to extract the useful information from the benchmark datasets to define the peptide samples via Equation (5).

Biology is a natural science with historic dimension. All biological species have developed beginning from a very limited number of ancestral species. It is true for protein/peptide sequence as well (Chou, 2004). Their evolution involves changes of single residues, insertions and deletions of several residues (Chou, 1995a), gene doubling, and gene fusion. To incorporate this kind of evolution information into Equation (5), we may consider the following.

According to (Schafer et al., 2001), the sequence evolution information for a peptide with $2^\xi + 1$ amino acid residues can be expressed by a $(2^\xi + 1) \times 20$ matrix, as given by

### Table 1. Preliminary test results with different size of $2^\xi + 1$ window.$^a$

| Window size $2^\xi + 1$ | Acc (%) | MCC | Sn (%) | Sp (%) |
|------------------------|---------|-----|--------|--------|
| 7                      | 59.22   | .1903 | 51.05  | 67.36  |
| 9                      | 56.85   | .1391 | 55.09  | 58.57  |
| 11                     | 59.10   | .1863 | 65.34  | 52.84  |
| 13                     | 89.35   | .7980 | 97.60  | 81.04  |
| 15                     | 67.92   | .3587 | 64.30  | 71.46  |
| 17                     | 63.10   | .2712 | 71.21  | 55.18  |
| 19                     | 63.96   | .2788 | 65.50  | 62.36  |
| 21                     | 64.08   | .2899 | 58.85  | 69.51  |
| 23                     | 63.14   | .2715 | 56.16  | 70.41  |
| 25                     | 63.64   | .2830 | 68.67  | 58.83  |
| 27                     | 64.69   | .3015 | 70.58  | 59.12  |
| 29                     | 65.35   | .3154 | 59.57  | 71.49  |
| 31                     | 65.17   | .3064 | 62.63  | 67.86  |

$^a$See Equation (15) for the metrics listed in the table.
where \( m^0_{ij} \) represents the original score of amino acid residue in the \( i \)-th \((i = 1, 2, \ldots, 2\xi + 1)\) sequential position of the peptide that is being changed to amino acid type \( j \((j = 1, 2, \ldots, 20)\) during the evolution process. Here, the numerical codes 1, 2, \ldots, 20 are used to denote the 20 native amino acid types according to the alphabetical order of their single character codes (Wu, Xiao, & Chou, 2011). The \((2\xi + 1) \times 20\) scores in Equation (6) were generated using PSI-BLAST (Schaffer et al., 2001) to search the UniProtKB/Swiss-Prot database (Release 2011_05) through three iterations with .001 as the E-value cutoff for multiple sequence alignment against the sequence of the peptide \( P \). In order to make every element in the matrix of Equation (6) within the range of 0–1, a conversion was performed through the standard sigmoid function to make it become

\[
P_{PSSM}^{0} = \begin{bmatrix} m^0_{1,1} & m^0_{1,2} & \cdots & m^0_{1,20} \\ m^0_{2,1} & m^0_{2,2} & \cdots & m^0_{2,20} \\ \vdots & \vdots & \ddots & \vdots \\ m^0_{2\xi+1,1} & m^0_{2\xi+1,2} & \cdots & m^0_{2\xi+1,20} \end{bmatrix} \quad (6)
\]

\[m_{ij} = \frac{1}{1 + e^{-m^0_{ij}}} \quad (1 \leq i \leq 2\xi + 1, 1 \leq j \leq 20) \quad (8)
\]

Now, we can use the gray model approach to extract more useful information from Equation (7) to define the components of Equation (5). According to the gray system theory (Deng, 1989), if the information of a system investigated is fully known, it is called a “white system”; if completely unknown, a “black system”; if partially known, a “gray system”. The model developed on the basis of such a theory is called “gray model”, which is a kind of nonlinear and dynamic model formulated by a differential equation. The gray model is particularly useful for solving complicated problems that are lack of sufficient information, or need to process uncertain information and to reduce random effects of acquired data. Following the same approach as done by Lin, Fang, and Xiao (2013), we can set the dimension of Equation (5) is \( \Omega = 3 \times 20 = 60 \) and each of its 60 components is defined by

\[
\psi_u = \begin{bmatrix} a^1_u & \text{when } u = 1 \\ a^2_u & \text{when } u = 2 \\ b^1_u & \text{when } u = 3 \\ \vdots & \vdots \\ a^{20}_u & \text{when } u = 58 \\ a^{20}_u & \text{when } u = 59 \\ b^{30}_u & \text{when } u = 60 \end{bmatrix} \quad (9)
\]

In the above equation

\[
\begin{align*}
\mathbf{a}^j & = \begin{bmatrix} a^j_1 \\ a^j_2 \\ \vdots \\ a^{20}_j \\ b^j_1 \\ b^j_2 \\ \vdots \\ b^{30}_j \end{bmatrix} = (\mathbf{B}^j_1 \mathbf{B}_j)^{-1} \mathbf{B}_j^T \mathbf{u}_j \quad (j = 1, 2, \ldots, 20) \\
\mathbf{B}_j & = \begin{bmatrix} -m_{2,j} & -m_{1,j} - 0.5m_{2,j} & 1 \\ -m_{3,j} & - \sum_{i=1}^{2} m_{i,j} - 0.5m_{3,j} & 1 \\ \vdots & \vdots & \vdots \\ -m_{k,j} & - \sum_{i=1}^{2} m_{i,j} - 0.5m_{k,j} & 1 \\ \vdots & \vdots & \vdots \\ -m_{2\xi+1,j} & - \sum_{i=1}^{2} m_{i,j} - 0.5m_{2\xi+1,j} & 1 \\
\end{bmatrix} \\
\mathbf{u}_j & = \begin{bmatrix} m_{2,j} - m_{1,j} \\ m_{3,j} - m_{2,j} \\ \vdots \\ m_{k,j} - m_{k-1,j} \\ m_{2\xi+1,j} - m_{2\xi,j} \end{bmatrix} \\
\end{align*}
\quad (11) \quad (12)
\]

As explained in Section 2.1, the value for parameter \( \xi \) in Equations (6)–(8) and (11)–(12) was set at 6. In other words, each of peptide samples considered in this study contains 13 amino residues with K at its center (see also Equation (1) and Supporting Information S1). Thus, Equation (3) can be reduced to

\[
\mathbb{S} = \mathbb{S}^+ \cup \mathbb{S}^- \quad (13)
\]

where \( \mathbb{S} = \mathbb{S}_6 \) contained 7855 peptides, of which 659 samples were of ubiquitination belonging to the positive data-set \( \mathbb{S}^+ = \mathbb{S}^+_6 \), while 7196 of non-ubiquitination belonging to the negative data-set \( \mathbb{S}^- = \mathbb{S}^-_6 \). The detailed sequences of the 659 + 7196 = 7855 peptide samples and their positions in proteins are given in Supporting Information S1.
Now, each of these peptide sequences in the benchmark data-set $S$ can be formulated by a 60-D vector via Equation (9), which is obtained by incorporating sequence evolutionary information into the general PseAAC (Equation (5)) using the gray system model (Equations (10)–(12)).

### 2.3. SVM operation engine

We used the SVM (Hearst, 1998; Vapnik, 1999) as the operation engine to perform predictions. SVM is a powerful and popular method for pattern recognition that has been widely used in computational biology as reflected by a series of publications (see, e.g. (Cai, Zhou, & Chou, 2003; Fan et al., 2014; Guo et al., 2014; Han et al., 2014; Liu, Wang, Zou, Dong, & Chen, 2013; Liu, Zhang et al., 2014; Mohabatkar et al., 2011; Qiu et al., 2014; Wan, Mak, & Kung, 2013; Xie, Fu, & Nie, 2013). The basic idea of SVM is to transform the data into a high dimensional feature space and then determine the optimal separating hyperplane using a kernel function. To handle a multi-class problem, “one-versus-one (OVO)” and “one-versus-rest (OVR)” are generally applied to extend the traditional SVM. For a brief formulation of SVM and how it works, see the papers (Cai et al., 2003). For more details about SVM, see a monograph (Cristianini & Shawe-Taylor, 2000).

The SVM software was downloaded from the LIBSVM (Chang & Lin, 2011), which provided a simple interface. Due to its advantages, the users can easily perform classification prediction by properly selecting the built-in parameters $c$ and $\gamma$. In order to maximize the performance of the SVM algorithm, the two parameters in the RBF kernel were preliminarily optimized through a grid search strategy. To obtain the optimized parameters, the search function SVMegForClass was used that was downloaded from http://www.matlabsky.com.

The predictor obtained via the aforementioned procedures is called iUbiq-Lys. To provide an intuitive picture, a flowchart is provided in Figure 4 to illustrate the prediction process of iUbiq-Lys.

How to properly and objectively evaluate the anticipated accuracy of a new predictor and how to make it easily accessible and user-friendly are the two key issues that will have important impacts on its application value. Below, we are to address these problems.

### 2.4. Metrics for scoring prediction quality

In the literature, the following four metrics are often used to score the quality of a predictor at four different angles

\[
\begin{align*}
\text{Sn} &= \frac{\text{TP}}{\text{TP} + \text{FN}} \\
\text{Sp} &= \frac{\text{TN}}{\text{TN} + \text{FP}} \\
\text{Acc} &= \frac{\text{TP} + \text{TN} + \text{FP} + \text{FN}}{\text{TP} \times \text{TN} + \text{FP} \times \text{FN}} \\
\text{MCC} &= \frac{(\text{TP} \times \text{TN} - \text{FP} \times \text{FN})}{\sqrt{\text{TP} + \text{FP}}(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}
\end{align*}
\]

where TP represents the number of the true positive; TN, the number of the true negative; FP, the number of the false positive; FN, the number of the false negative; Sn, the sensitivity; Sp, the specificity; Acc, the accuracy; MCC, the Mathew’s correlation coefficient. To most biologists, unfortunately, the four metrics as formulated in Equation (14) are not quite intuitive and easy to understand, particularly the equation for MCC. Therefore, we prefer to use the formulation proposed recently in (Chen et al., 2013; Xu, Ding, & Wu, 2013; Xu, Shao, & Wu, 2013) based on the symbols introduced by Chou (Chou, 2001b) in predicting signal peptides. According to the formulation, the same four metrics can be expressed as

\[
\begin{align*}
\text{Sn} &= 1 - \frac{N^+}{N^+ + N^-}, \quad 0 \leq \text{Sn} \leq 1 \\
\text{Sp} &= 1 - \frac{N^-}{N^+ + N^-}, \quad 0 \leq \text{Sp} \leq 1 \\
\text{Acc} &= 1 - \frac{N^+ + N^-}{N^+ + N^-}, \quad 0 \leq \text{Acc} \leq 1 \\
\text{MCC} &= \frac{1 - \left( \frac{N^+ + N^-}{N^+ + N^-} \right)}{\sqrt{\left( \frac{N^+ + N^-}{N^+ + N^-} \right) \left( \frac{N^+ + N^-}{N^+ + N^-} \right)}}, \quad -1 \leq \text{MCC} \leq 1
\end{align*}
\]

where $N^+$ is the total number of the ubiquitination peptides investigated, while $N^-$ is the number of the ubiquitination peptides incorrectly predicted as the non-ubiquitination peptides; $N^-$ is the total number of the non-ubiquitination investigated, while $N^-_+$ is the number of the non-ubiquitination incorrectly predicted as the ubiquitination peptides (Chou, 2001b).

Now, it is crystal clear from Equation (15) that when $N^+ = 0$ meaning none of the ubiquitination peptides was incorrectly predicted to be a non-ubiquitination peptide, we have the sensitivity $\text{Sn} = 1$. When $N^+ = N^+$ meaning that all the ubiquitination peptides were incorrectly predicted as the non-ubiquitination peptides, we have the sensitivity $\text{Sn} = 0$. Likewise, when $N^- = 0$ meaning none of the non-ubiquitination peptides was incorrectly predicted to be the ubiquitination peptide, we have the specificity $\text{Sp} = 1$; whereas $N^+ = N^-$ meaning all the non-ubiquitination peptides were incorrectly predicted as the ubiquitination peptides, we have the specificity $\text{Sp} = 0$. When $N^+ = N^- = 0$ meaning that none of the ubiquitination peptides in the positive data-set $S^+$ and
none of the non-ubiquitination peptides in the negative data-set \( S^- \) was incorrectly predicted, we have the overall accuracy \( \text{Acc} = 1 \) and \( \text{MCC} = 1 \); when \( N^+ = N^+ \) and \( N^- = N^- \) meaning that all the ubiquitination peptides in the positive data-set \( S^+ \) and all the non-ubiquitination peptides in the negative data-set \( S^- \) were incorrectly predicted, we have the overall accuracy \( \text{Acc} = 0 \) and \( \text{MCC} = -1 \); whereas when \( N^+ = N^+ / 2 \) and \( N^- = N^- / 2 \) we have \( \text{Acc} = .5 \) and \( \text{MCC} = 0 \) meaning no better than random prediction. As we can see from the above discussion based on Equation (15), the meanings of sensitivity, specificity, overall accuracy, and Mathew’s correlation coefficient have become much more intuitive and easier to understand.

It is instructive to point out; however, the set of metrics in Equations (14) and (15) is valid only for the single-label systems. For the multi-label systems, such as those for the subcellular localization of multiplex proteins (see, e.g. (Lin et al., 2013; Wu et al., 2011; Xiao et al., 2011)) where a protein may have two or more locations, and those for the functional types of antimicrobial peptides (see, e.g. (Xiao, Wang, Lin & Jia, 2013)) where a peptide may possess two or more functional types, a completely different set of metrics is needed as elaborated in (Chou, 2013).

2.5. Jackknife cross-validation

With a set of clear and valid metrics as defined in Equation (15), to measure the quality of a predictor, the next thing we need to consider is how to objectively derive the values of these metrics for a predictor.

In statistical prediction, the following three cross-validation methods are often used to calculate the metrics of Equation (15) for evaluating the quality of a predictor: independent data-set test, subsampling test, and jackknife test (Chou, 2011). However, of the three test methods, the jackknife test is deemed the least arbitrary that can always yield a unique result for a given benchmark data-set. The reasons are as follows. (i) For the independent data-set test, although all the samples used to test the predictor are outside the training data-set used to train it so as to exclude the “memory” effect or bias, the way of how to select the independent samples to test the predictor could be quite arbitrary unless the number of independent samples is sufficiently large. This kind of arbitrariness might result in completely different conclusions. For instance, a predictor achieving a higher success rate than the other predictor for a given independent testing data-set might fail to keep so when tested by another independent testing data-set. (ii) For the subsampling test, the concrete procedure usually used in the literatures is the fivefold, sevenfold or 10-fold cross-validation. The problem with this kind of subsampling test is that the number of possible selections in dividing a benchmark data-set is an astronomical figure even for a very simple data-set, as demonstrated by Equations (28)–(30) in (Chou, 2011). Therefore, in any actual subsampling cross-validation tests, only an extremely small fraction of the possible selections are taken into account. Since different selections will always lead to different results even for a same benchmark data-set and a same predictor, the subsampling test cannot avoid the arbitrariness either. A test method unable to yield a unique outcome cannot be deemed as a good one. (iii) In the jackknife test, all the samples in the benchmark data-set will be singled out one-by-one and tested by the predictor trained by the remaining samples. During the process of jackknifing, both the training data-set and testing data-set are actually open, and each sample will be in turn moved between the two. The jackknife test can exclude the “memory” effect. Also, the arbitrariness problem as mentioned above for the independent data-set test and subsampling test can be avoided because the outcome obtained by the jackknife cross-validation is always unique for a given benchmark data-set. Accordingly, the jackknife test has been increasingly used and widely recognized by investigators to examine the quality of various predictors (see, e.g. (Du et al., 2006; Hajisharifi et al., 2014; Kandaswamy et al., 2011; Mei, 2012b; Mohabatkar et al., 2011; Mohabatkar et al., 2013; Niu, Hu, Zheng, Huang, & Feng, 2012; Sahu & Panda, 2010; Shen, 2010)). Accordingly, in this study we also used the jackknife cross-validation method to calculate the metrics in Equation (15) although it would take more computational time.

3. Results and discussion

As shown in the Supporting Information S1, the number of samples in the negative data-set \( S^- \) is almost 11 times the size of the positive data-set \( S^+ \) (cf. Equation (13)). To make the tests under a balance situation, we divided the negative data-set \( S^- \) into 11 subsets; that is

\[
S^- = S^- (1) \cup S^- (2) \cup S^- (3) \cup \cdots \cup S^- (11) = \bigcup_{i=1}^{11} S^- (i)
\]

(16)

where the subset \( S^- (i) \) (\( i = 1, 2, \ldots , 10 \)) contained 659 non-ubiquitination peptides, while \( S^- (11) \) contained 7196 \(- 659 \times 10 = 606 \) non-ubiquitination peptides. The detailed sequence data for the 11 negative subsets are given in the Supporting Information S2.

Listed in Table 2 are the jackknife test results in identifying the ubiquitination sites by iUbiq-Lys, respectively, on the benchmark datasets \( S(i) = S^+ \cup S^- (i) (i = 1, 2, \cdots , 11) \) and \( S = S^+ \cup S^- \). As we can see from the table, the average results by the current iUbiq-Lys predictor on the 11 benchmark datasets is 90.06% for Acc, .82 for MCC, 80.99% for Sn, and 99.06% for Sp.
Table 2. The four metrics rates obtained with iUbiq-Lys by the jackknife test on the benchmark datasets with different negative datasets.

| Benchmark data-set | Acc (%) | MCC | Sn (%) | Sp (%) |
|--------------------|---------|-----|--------|--------|
| S(1) = S⁺ U S⁻ (1) | 90.48   | .82 | 81.65  | 99.24  |
| S(2) = S⁺ U S⁻ (2) | 89.95   | .81 | 80.43  | 99.39  |
| S(3) = S⁺ U S⁻ (3) | 90.26   | .82 | 80.76  | 99.70  |
| S(4) = S⁺ U S⁻ (4) | 89.64   | .81 | 80.12  | 99.09  |
| S(5) = S⁺ U S⁻ (5) | 89.41   | .80 | 79.51  | 95.24  |
| S(6) = S⁺ U S⁻ (6) | 89.27   | .80 | 82.42  | 96.97  |
| S(7) = S⁺ U S⁻ (7) | 88.27   | .78 | 77.83  | 98.63  |
| S(8) = S⁺ U S⁻ (8) | 89.80   | .81 | 80.00  | 99.54  |
| S(9) = S⁺ U S⁻ (9) | 90.02   | .82 | 80.58  | 99.39  |
| S(10) = S⁺ U S⁻ (10)| 92.08   | .85 | 85.02  | 99.09  |
| S(11) = S⁺ U S⁻ (11)| 91.01   | .83 | 82.57  | 99.39  |
| Average           | 90.06   | .82 | 80.99  | 99.06  |
| S = S⁺ U S⁻ (Integrated) | 82.14   | .50 | 80.56  | 99.39  |

*The positive data-set S⁺ contains 659 peptides with an ubiquitination site at each of their centers (cf. Equations (1)-(2)), and the negative data-set S⁻ contains 7196 peptides with a non-ubiquitination site at each of their centers (see Supporting Information S1). Since the latter is about 11 times the size of the former, and hence is randomly divided into 11 subsets (Equation (16)) where, except S⁻ (11) containing 606 negative samples, all the other 10 negative subsets each contain 659 samples (see Supporting Information S2). See the text for further explanation.

Compared with S(i), although the results obtained from the benchmark data-set S are basically about the same except for MCC, which is dropped down from .82 to .50. This is because the positive data-set and negative data-set are extremely uneven in size; the latter is 11 times the size of the former. Even though, however, it is still higher than or comparable to the MCC values reported by the existing methods, as shown in Table 3. Particularly, as we can see from the table, most of the existing methods did not provide a web-server, and hence, their practical usage is quite limited. To overcome such a shortcoming, we have established a web-server for the current predictor with the website at http://www.jci-bioinfo.cn/iUbiq-Lys. Moreover, for the convenience of the vast majority of experimental scientists, below we are to give a step-to-step guide for how to use the web-server to get the desired results.

4. Web-server and user guide

**Step 1.** Open the web-server at http://www.jci-bioinfo.cn/iUbiq-Lys and you will see the top page of the predictor on your computer screen, as shown in Figure 5. Click on the Read Me button to see a brief introduction about iUbiq-Lys predictor and the caveat when using it.

**Step 2.** Either type or copy/paste the sequences of query proteins into the input box shown at the center of Figure 5. The input sequence should be in the FASTA format. All the input sequences should be in the FASTA format. A sequence in FASTA format consists of a single initial line beginning with the symbol “>” in the first column, followed by lines of sequence data in which amino acids are represented using single-letter codes. Except for the mandatory symbol “>”, all the other characters in

Table 3. A comparison of the current predictor iUbiq-Lys with the existing methods for identifying lysine ubiquitination sites in proteins.

| Method              | Acc (%) | MCC | Sn (%) | Sp (%) | Web-server | Number of positive samples | Number of negative samples |
|---------------------|---------|-----|--------|--------|------------|---------------------------|----------------------------|
| iUbiq-Lys (Average) | 90.06   | .82 | 80.99  | 99.06  | Yes        | 659                       | 7196                       |
| iUbiq-Lys (Integrated) | 82.14   | .50 | 80.56  | 99.39  | Yes        | 659                       | 7196                       |
| IPM                 | 72.19   | N/A | N/A    | N/A    | Yes        | 151                       | 151                        |
| mRMR+NN             | N/A     | .142| N/A    | N/A    | No         | 364                       | 1092                       |
| Ensemble classifier  | 76.82   | .519| 76.85  | 76.91  | No         | 298                       | 563                        |
| CKSAAP UbSite       | 73.49   | .469| 69.85  | 76.96  | Yes        | 263                       | 4345                       |
| Binary encoding     | 58.14   | .163| 56.23  | 60.04  | No         | 263                       | 4345                       |

*See Tung and Ho (2008).
*See Cai et al. (2012).
*See Zhao et al. (2011).
*See Chen et al. (2011).
the single initial line are optional and only used for the purpose of identification and description. The sequence ends if another line starting with the symbol “>” appears; this indicates the start of another sequence. Example sequences in FASTA format can be seen by clicking on the Example button. Note that if your input protein sequences contain any symbol other than the 20 native amino acid codes (ACDEFGHIKLMNPQRSTVWY), an error message will be prompted on your screen.

Step 3. Click on the Submit button to see the predicted result. For example, if you use the two query protein sequences in the Example window as the input, after clicking the Submit button, you will see the following on your screen. (i) The 1st protein contains 6 K residues; of which only the one located at the sequence position 254 is of ubiquitination site, while all the others are of non-ubiquitination site. (ii) The 2nd protein contains 23 K residues; of which only the two located at the sequence positions 485 and 490 belong to the ubiquitination site, while all the others belong to non-ubiquitination site. All these results are fully consistent with experimental observations.

Step 4. As shown on the lower panel of Figure 5, you may also choose the batch prediction by entering your e-mail address and your desired batch input file (in FASTA format) via the “Browse” button. To see the sample of batch input file, click on the button Batch-example. The maximum number of the query proteins for each batch input file is 50. After clicking the button Batch-submit, you will see “Your batch job is under computation; once the results are available, you will be notified by e-mail.”

Step 5. Click on the Supporting Information button to download the benchmark data-set used to train and test the iUbiq-Lys predictor (Figure 5).

Conclusion
As one of the most important PTMs, lysine ubiquitination plays an important role in regulating a variety of biological processes.

A new predictor was developed for identifying the lysine ubiquitination sites in proteins based on a set of 13-tuple peptides generated as follows. Sliding a window of 13 amino acids along each of the protein sequences taken from a protein database, collected were only those peptide segments with K (lysine) at the center, that is the potential ubiquitination-site-containing peptides.

The new predictor is called iUbiq-Lys, in which each of the potential ubiquitination-site-containing peptides was formulated with a 60-D vector formed by incorporating the sequence evolutionary information into the general form (Chou, 2011) of pseudo-amino acid composition (Chou, 2001a, 2005) or Chou’s PseAAC (Cao et al., 2013; Du et al., 2014; Lin & Lapointe, 2013) via a gray system model.

It was observed from the rigorous cross-validations that the iUbiq-Lys outperformed any of the existing
predictors in this area, as reflected by remarkably higher scores in a set of four metrics generally used to measure the accuracy of a predictor.

Furthermore, for the convenience of most experimental scientists, the web-server of iUbiq-Lys has been established at http://www.jci-bioinfo.cn/iUbiq-Lys, with which users can easily get their desired results without the need to follow the complicated mathematics that were presented in this paper just for the integrity of the predictor.

It has not escaped our notice that the current approach can also be used to develop various methods for identifying the sites of other PTMs in proteins.

Supplementary material

The supplementary material for this paper is available online at http://dx.doi.org/10.1080/07391102.2014.968875.

Acknowledgment

The authors wish to express their gratitude to the two anonymous reviewers for their constructive comments, which were very helpful in strengthening the presentation of this study.

Funding

This work was partially supported by the National Nature Science Foundation of China [grant number 61261027]; the Key Project of Chinese Ministry of Education [grant number 210116]; Natural Science Foundation of Jiangxi Province, China [grant number 20142BAB207013, [grant number 20122BAB201044]; the Foundation of Jiangxi Education Committee, China [grant number GJJ14651].

References

Aguilar, R. C., & Wendland, B. (2003). Ubiquitin: Not just for proteasomes anymore. Current Opinion in Cell Biology, 15, 184–190.

Cai, Y. D. (2003). Nearest neighbour algorithm for predicting protein subcellular location by combining functional domain composition and pseudo-amino acid composition. Biochemical and Biophysical Research Communications, 305, 407–411.

Cai, Y., Huang, T., Hu, L., Shi, X., Xie, L., & Li, Y. (2012). Prediction of lysine ubiquitination with mRMR feature selection and analysis. Amino Acids, 42, 1387–1395.

Cai, Y. D., Zhou, G. P., & Chou, K. C. (2003). Support vector machines for predicting membrane protein types by using functional domain composition. Bioophysical Journal, 84, 3257–3263.

Cao, D. S., Xu, Q. S., & Liang, Y. Z. (2013). Propy: A tool to generate various modes of Chou’s PseAAC. Bioinformatics, 29, 960–962.

Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. ACM Transactions on Intelligent Systems and Technology, 2, 1–27.

Chen, Z., Chen, Y. Z., Wang, X. F., Wang, C., Yan, R. X., & Zhang, Z. (2011). Prediction of ubiquitination sites by using the composition of k-spaced amino acid pairs. PLoS ONE, 6, e22930.

Chen, W., Feng, P. M., & Deng, E. Z. (2014). iTIS-PseTNC: A sequence-based predictor for identifying translation initiation site in human genes using pseudo trinucleotide composition. Analytical Biochemistry, 462, 76–83.

Chen, W., Feng, P. M., & Lin, H. (2013). iRSpot-PseDNC: Identify recombination spots with pseudo dinucleotide composition. Nucleic Acids Research, 41, e68.

Chen, W., Lei, T. Y., Jin, D. C., & Lin, H. (2014). PseKNC: A flexible web server for generating pseudo K-tuple nucleotide composition. Analytical Biochemistry, 456, 53–60.

Chen, Y. K., & Li, K. B. (2013). Predicting membrane protein types by incorporating protein topology, domains, signal peptides, and physicochemical properties into the general form of Chou’s pseudo amino acid composition. Journal of Theoretical Biology, 318, 1–12.

Chen, W., Lin, H., & Cai, N. (2012). iNuc-PhysChem: A sequence-based predictor for identifying nucleosomes via physicochemical properties. PLoS ONE, 7, e47843.

Chou, K. C. (1993a). A vectorized sequence-coupling model for predicting HIV protease cleavage sites in proteins. Journal of Biological Chemistry, 268, 16938–16948.

Chou, K. C. (1995a). The convergence–divergence duality in lectin domains of selectin family and its implications. FEBS Letters, 363, 123–126.

Chou, K. C. (1995b). A sequence-coupled vector-projection model for predicting the specificity of GalNAc-transferase. Protein Science, 4, 1365–1383.

Chou, K. C. (2001a). Prediction of protein cellular attributes using pseudo amino acid composition. PROTEINS: Structure, Function, and Genetics (Erratum: ibid., 2001, Vol. 44, 60), 43, 246–255.

Chou, K. C. (2001b). Using subsite coupling to predict signal peptides. Protein Engineering Design and Selection, 14, 75–79.

Chou, K. C. (2004). Structural bioinformatics and its impact to biomedical science. Current Medicinal Chemistry, 11, 2105–2134.

Chou, K. C. (2005). Using amphiphilic pseudo amino acid composition to predict enzyme subfamily classes. Bioinformatics, 21, 10–19.

Chou, K. C. (2011). Some remarks on protein attribute prediction and pseudo amino acid composition. Journal of Theoretical Biology, 273, 236–247.

Chou, K. C. (2013). Some remarks on predicting multi-label attributes in molecular systems. Molecular BioSystems, 9, 1092–1100.

Chou, K. C., & Shen, H. B. (2007). Recent progress in protein subcellular location prediction. Analytical Biochemistry, 370, 1–16.

Cristianini, N., & Shawe-Taylor, J. (2000). *An introduction to support vector machines and other Kernel-based learning methods*. Cambridge: Cambridge University Press.

Deng, J. L. (1989). Introduction to grey system theory. The Journal of Grey System, 1, 1–24.

Ding, H., Deng, E. Z., Yuan, L. F., Liu, L., & Lin, H. (2014). iCTX-Type: A sequence-based predictor for identifying the types of conotoxins in targeting ion channels. *Biomed Research International* 2014, 286419.

Du, X., Gu, S., & Jiao, Y. (2014). PseAAC-general: Fast building various modes of general form of Chou’s pseudo-amino acid composition. Journal of Theoretical Biology, 3506, 3495–3506.

Du, P., Wang, X., Xu, C., & Gao, Y. (2012). PseAAC-builder: A cross-platform stand-alone program for generating various approaches Chou’s pseudo-amino acid compositions. Analytical Biochemistry, 425, 117–119.
Du, Q. S., Jiang, Z. Q., He, W. Z., & Li, D. P. (2006). Amino acid principal component analysis (AAPCA) and its applications in protein structural class prediction. Journal of Biomolecular Structure and Dynamics, 23, 635–640.

Esmaeili, M., Mohabatkar, H., & Mohsenzadeh, S. (2010). Using the concept of Chou’s pseudo amino acid composition for risk type prediction of human papillomaviruses. Journal of Theoretical Biology, 263, 203–209.

Fan, Y. N., Xiao, X., & Min, J. L. (2014). iNR-Drug: Predicting the interaction of drugs with nuclear receptors in cellular networking. International Journal of Molecular Sciences, 15, 4915–4937.

Feng, P. M., Chen, W., & Lin, H. (2013). iHSP-PseRAAAC: Identifying the heat shock protein families using pseudo reduced amino acid alphabet composition. Analytical Biochemistry, 442, 118–125.

Georgiou, D. N., Karakasidis, T. E., & Megaritis, A. C. (2013). A short survey on genetic sequences, Chou’s pseudo amino acid composition and its combination with fuzzy set theory. The Open Bioinformatics Journal, 7, 41–48. Retrieved from http://www.benthamscience.com/open/tobioij/articles/V007/S0025TOBIOJ/41T0BIOJ.pdf

Guo, S. H., Deng, E. Z., Xu, L. Q., Ding, H., & Lin, H. (2014). iNuc-PseKNC: A sequence-based predictor for predicting nucleosome positioning in genomes with pseudo k-tuple nucleotide composition. Bioinformatics, 30, 1522–1529.

Haglund, K., & Dikic, I. (2005). Ubiquitylation and cell signaling. The EMBO Journal, 24, 3353–3359.

Hajishari, Z., Piryaeie, M., Mohammad Beigi, M., Behbahani, M., & Mohabatkar, H. (2014). Predicting anticancer peptides with Chou’s pseudo amino acid composition and investigating their mutagenicity via Ames test. Journal of Theoretical Biology, 341, 34–40.

Han, G. S., Yu, Z. G., & Anh, V. (2014). A two-stage SVM method to predict membrane protein types by incorporating amino acid classifications and physicochemical properties into a general form of Chou’s PseAAC. Journal of Theoretical Biology, 344, 31–39.

Hayat, M., & Khan, A. (2012). Discriminating outer membrane proteins with fuzzy K-nearest neighbor algorithms based on the general form of Chou’s PseAAC. Protein and Peptide Letters, 19, 411–421.

Heard, M. A. (1998). Support vector machines. IEEE Intelligent Systems, 13, 18–28.

Herrmann, J., Lerman, L. O., & Lerman, A. (2007). Ubiquitin and ubiquitin-like proteins in protein regulation. Circulation Research, 100, 1276–1291.

Hershko, A., & Ciechanover, A. (1998). The ubiquitin system. Annual Review of Biochemistry, 67, 425–479.

Hicke, L. (2001). Protein regulation by monoubiquitin. Nature Reviews Molecular Cell Biology, 2, 195–201.

Hicke, L., & Dunn, R. (2003). Regulation of membrane protein transport by ubiquitin and ubiquitin-binding proteins. Annual Review of Cell and Developmental Biology, 19, 141–172.

Hoeller, D., Hecker, C. M., & Dikic, I. (2006). Ubiquitin and ubiquitin-like proteins in cancer pathogenesis. Nature Reviews Cancer, 6, 776–788.

Jiang, Y., Huang, T., Chen, L., & Gao, Y. F. (2013). Signal propagation in protein interaction network during colorectal cancer progression. BioMed Research International, 2013, 287019.

Kandaswamy, K. K., Martinetz, T., Suganthan, P. N., Sridharan, S., & Pugalenthi, G. (2011). Afp-Pred: A random forest approach for predicting antifreeze proteins from sequence-derived properties. Journal of Theoretical Biology, 270, 56–62.

Kandaswamy, K. K., Pugalenthi, G., Möller, S., Hartmann, E., Kalies, K. U., Suganthan, P. N., & Martinetz, T. (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.

Khosravian, M., Faramarzi, F. K., Beigi, M. M., Behbahani, M., & Mohabatkar, H. (2013). Predicting antibacterial peptides by the concept of Chou’s pseudo-amino acid composition and machine learning methods. Protein and Peptide Letters, 20, 180–186.

Kong, L., Zhang, L., & Lv, J. (2014). Accurate prediction of protein structural classes by incorporating predicted secondary structure information into the general form of Chou’s pseudo amino acid composition. Journal of Theoretical Biology, 344, 12–18.

Lin, S. X., & Lapointe, J. (2013). Theoretical and experimental biology in one. Journal of Biomedical Science and Engineering, 6, 435–442.

Lin, W. Z., Fang, J. A., & Xiao, X. (2011). iDNA-Prot: Identification of DNA binding proteins using random forest with grey model. PLoS ONE, 6, e24756.

Lin, W. Z., Fang, J. A., & Xiao, X. (2013). iLoc-Animal: A multi-label learning classifier for predicting subcellular localization of animal proteins. Molecular BioSystems, 9, 634–644.

Liu, B., Wang, X., Zou, Q., Dong, Q., & Chen, Q. (2013). Protein remote homology detection by combining Chou’s pseudo amino acid composition and profile-based protein representation. Molecular Informatics, 32, 775–782.

Liu, B., Xu, J., Lan, X., Xu, R., & Zhou, J. (2014). iDNA-Protidis: Identifying DNA-binding proteins by incorporating amino acid distance-pairs and reduced alphabet profile into the general pseudo amino acid composition. PLoS ONE, 9, e106691.

Liu, B., Zhang, D., Xu, R., Xu, J., Wang, X., & Chen, Q. (2014). Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection. Bioinformatics, 30, 472–479.

Mei, S. (2012a). Predicting plant protein subcellular multi-localization by Chou’s PseAAC formulation based multi-label homology knowledge transfer learning. Journal of Theoretical Biology, 310, 80–87.

Mei, S. (2012b). Multi-kernel transfer learning based on Chou’s PseAAC formulation for protein submitochondria localization. Journal of Theoretical Biology, 293, 121–130.

Mohabatkar, H. (2010). Prediction of cyclin proteins using Chou’s pseudo amino acid composition. Protein and Peptide Letters, 17, 1207–1214.

Mohabatkar, H., Beigi, M. M., Abdolahi, K., & Mohsenzadeh, S. (2013). Prediction of allergenic proteins by means of the concept of Chou’s pseudo amino acid composition and a machine learning approach. Medicinal Chemistry, 9, 133–137.

Mohabatkar, H., Mohammad Beigi, M., & Esmaeili, A. (2011). Prediction of GABA(A) receptor proteins using the concept of Chou’s pseudo-amino acid composition and support vector machine. Journal of Theoretical Biology, 281, 18–23.

Mohammad Beigi, M., Behjati, M., & Mohabatkar, H. (2011). Prediction of metalloproteinase family based on the concept of Chou’s pseudo amino acid composition using a machine learning approach. Journal of Structural and Functional Genomics, 12, 191–197.
Nanni, L., & Lumini, A. (2008). Genetic programming for creating Chou’s pseudo amino acid based features for sub-mitochondria localization. *Amino Acids*, 34, 653–660.

Nanni, L., Lumini, A., Gupta, D., & Garg, A. (2012). Identifying bacterial virulent proteins by fusing a set of classifiers based on variants of Chou’s pseudo amino acid composition and on evolutionary information. *IEEE/ACM Transactions on Computational Biology and Informatics*, 9, 467–475.

Niu, S., Hu, L. L., Zheng, L. L., Huang, T., & Feng, K. Y. (2012). Predicting protein oxidation sites with feature selection and analysis approach. *Journal of Biomolecular Structure & Dynamics (JBSD)*, 29, 650–658.

Pickart, C. M. (2001). Ubiquitin enters the new millennium. *Molecular Cell*, 8, 499–504.

Qiu, W. R., Xiao, X., & Chou, K. C. (2014). iRSpot-TNCpseAAC: Identify recombination spots with trinucleotide composition and pseudo amino acid components. *International Journal of Molecular Sciences*, 15, 1746–1766.

Radivojac, P., Vacic, V., Haynes, C., Cocklin, R. R., Mohan, A., Heyen, J. W., … Jakoucheva, L. M. (2010). Identification, analysis, and prediction of protein ubiquitination sites. *Proteins: Structure, Function, and Bioinformatics*, 78, 365–380.

Reinstein, E., & Ciechanover, A. (2006). Narrative review: Protein degradation and human diseases: The ubiquitin connection. *Annals of Internal Medicine*, 145, 676–684.

Saghatelyan, A., & Cravatt, B. F. (2005). Assignment of protein function in the postgenomic era. *Nature Chemical Biology*, 1, 130–142.

Sahu, S. S., & Panda, G. (2010). A novel feature representation method based on Chou’s pseudo amino acid composition for protein structural class prediction. *Computational Biology and Chemistry*, 34, 320–327.

Schafer, A. A., Aravind, L., Madden, T. L., Shavirin, S., Spouge, J. L., Wolf, Y. I., … Alschul, S. F. (2001). Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. *Nucleic Acids Research*, 29, 2994–3005.

Schwartz, A. L., & Ciechanover, A. (1999). The ubiquitin-proteasome pathway and pathogenesis of human diseases. *Annual Review of Medicine*, 50, 57–74.

Shen, H. B. (2008). PseAAC: A flexible web server for generating various kinds of protein pseudo amino acid composition. *Analytical Biochemistry*, 373, 386–388.

Shen, H. B. (2009). A top-down approach to enhance the power of predicting human protein subcellular localization: Hum-mPloc 2.0. *Analytical Biochemistry*, 394, 269–274.

Shen, H. B. (2010). Virus-mPloc: A fusion classifier for viral protein subcellular location prediction by incorporating multiple sites. *Journal of Biomolecular Structure and Dynamics*, 28, 175–186.

Shen, H. B., Yang, J., & Chou, K. C. (2006). Fuzzy KNN for predicting membrane protein types from pseudo amino acid composition. *Journal of Theoretical Biology*, 240, 9–13.

Sun, L., & Chen, Z. J. (2004). The novel functions of ubiquitination in signaling. *Current Opinion in Cell Biology*, 16, 119–126.

Sun, X. Y., Shi, S. P., Qiu, J. D., Suo, S. B., Huang, S. Y., & Liang, P. R. (2012). Identifying protein quaternary structural attributes by incorporating physicochemical properties into the general form of Chou’s PseAAC via discrete wavelet transform. *Molecular BioSystems*, 8, 3178–3184.

Tung, C. W., & Ho, S. Y. (2008). Computational identification of ubiquitination sites from protein sequences. *BMC Bioinformatics*, 9, 310.

Vapnik, V. N. (1999). An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10, 988–999.

Wan, S., Mak, M. W., & Kuang, S. Y. (2013). GOASVM: A subcellular location predictor by incorporating term-frequency gene ontology into the general form of Chou’s pseudo-amino acid composition. *Journal of Theoretical Biology*, 323, 40–48.

Wang, M., Yang, J., & Xu, Z. J. (2005). SLLE for predicting membrane protein types. *Journal of Theoretical Biology*, 232, 7–15.

Welchman, R. L., Gordon, C., & Mayer, R. J. (2005). Ubiquitin and ubiquitin-like proteins as multifunctional signals. *Nature Reviews Molecular Cell Biology*, 6, 599–609.

Wu, Z. C., Xiao, X., & Chou, K. C. (2011). iLoc-Plant: A multi-label classifier for predicting the subcellular localization of plant proteins with both single and multiple sites. *Molecular BioSystems*, 7, 3287–3297.

Xiao, X., Min, J. L., & Wang, P. (2013). iGPCR-Drug: A web server for predicting interaction between GPCRs and drugs in cellular networking. *PLoS ONE*, 8, e72234.

Xiao, X., Wang, P., Lin, W. Z., & Jia, J. H. (2013). iAMP-2L: A two-level multi-label classifier for identifying antimicrobial peptides and their functional types. *Analytical Biochemistry*, 436, 168–177.

Xiao, X., Wu, Z. C., & Chou, K. C. (2011). iLoc-Virus: A multi-label learning classifier for identifying the subcellular localization of virus proteins with both single and multiple sites. *Journal of Theoretical Biology*, 284, 42–51.

Xie, H. L., Fu, L., & Nie, X. D. (2013). Using ensemble SVM to identify human GPCRs N-linked glycosylation sites based on the general form of Chou’s PseAAC. *Protein Engineering, Design & Selection*, 26, 735–742.

Xu, Y., Ding, J., & Wu, L. Y. (2013). iSNOPseAAC: Predict cysteine S-nitrosylation sites in proteins by incorporating position specific amino acid propensity into pseudo amino acid composition. *PLoS ONE*, 8, e55844.

Xu, Y., Shao, X. J., & Wu, L. Y. (2013). iSNOPseAAC: Predicting amino acid pairwise coupling into PseAAC for predicting cysteine S-nitrosylation sites in proteins. *PeerJ*, 1, e171.

Xu, Y., Wen, X., Shao, X. J., & Deng, N. Y. (2014). iHydPseAAC: Predicting hydroxyproline and hydroxylysine in proteins by incorporating dipeptide position-specific propensity into pseudo amino acid composition. *International Journal of Molecular Sciences*, 15, 7594–7610.

Xu, Y., Wen, X., Wen, L. S., & Wu, L. Y. (2014). iNitro-Tyr: Prediction of nitrotyrosine sites in proteins with general pseudo amino acid composition. *PLoS ONE*, 9, e105018.

Zhang, S. W., Zhang, Y. L., Yang, H. F., Zhao, C. H., & Pan, Q. (2008). Using the concept of Chou’s pseudo amino acid composition to predict protein subcellular localization: An approach by incorporating evolutionary information and von Neumann entropies. *Amino Acids*, 34, 565–572.

Zhao, X., Li, X., Ma, Z., & Yin, M. (2011). Prediction of lysine ubiquitylation with ensemble classifier and feature selection. *International Journal of Molecular Sciences*, 12, 8347–8361.

Zia Ur, R., & Khan, A. (2012). Identifying GPCRs and their variants of Chou’s PseAAC via discrete wavelet transform. *PeerJ*, 8, 3005.

Zou, D., He, Z., He, J., & Xia, Y. (2011). Supersecondary structure prediction using Chou’s pseudo amino acid composition. *Journal of Computational Chemistry*, 32, 271–278.