*electronics*

MDPI

*Article*

# Protein Subnuclear Localization Based on Radius-SMOTE and Kernel Linear Discriminant Analysis Combined with Random Forest

**Liwen Wu [1,2], Shanshan Huang [2,3], Feng Wu [2,3], Qian Jiang [2,3,*], Shaowen Yao [2,3,*] and Xin Jin [2,3]**

1 School of Information Science & Engineering, Yunnan University, Kunming 650000, China; wulw@mail.ynu.edu.cn
2 Engineering Research Center of Cyberspace, Yunnan University, Kunming 650091, China; huangshanshan9633@163.com (S.H.); gzwf@mail.ynu.edu.cn (F.W.); xinjin@ynu.edu.cn (X.J.)
3 School of Software, Yunnan University, Kunming 650000, China
* Correspondence: jiangqian_1221@163.com (Q.J.); yaosw@ynu.edu.cn (S.Y.); Tel.: +86-18487219630 (Q.J.)

Received: 18 August 2020; Accepted: 22 September 2020; Published: 24 September 2020


**Abstract:** Protein subnuclear localization plays an important role in proteomics, and can help researchers to understand the biologic functions of nucleus. To date, most protein datasets used by studies are unbalanced, which reduces the prediction accuracy of protein subnuclear localization—especially for the minority classes. In this work, a novel method is therefore proposed to predict the protein subnuclear localization of unbalanced datasets. First, the position-specific score matrix is used to extract the feature vectors of two benchmark datasets and then the useful features are selected by kernel linear discriminant analysis. Second, the Radius-SMOTE is used to expand the samples of minority classes to deal with the problem of imbalance in datasets. Finally, the optimal feature vectors of the expanded datasets are classified by random forest. In order to evaluate the performance of the proposed method, four index evolutions are calculated by Jackknife test. The results indicate that the proposed method can achieve better effect compared with other conventional methods, and it can also improve the accuracy for both majority and minority classes effectively.

**Keywords:** protein subnuclear localization; kernel linear discriminant analysis; SMOTE; random forest; position-specific score matrix

## 1. Introduction

A biologic cell is a highly ordered whole that can be divided into different organelles according to spatial distribution and function, such as cytoplasm, nucleus, etc. The proteins in cells strongly correlate with life activities because proteins are able to perform biologic functions only when the proteins are transported to the correct nucleus or in a cell [1,2]. The correct protein subnuclear localization can be used to annotate the structure and function of protein, and it also contributes to the development of new drugs about genetic disease, even cancer [3].

With the development of life sciences, traditional experiments such as cell fractionation, electron microscopy, cannot meet the challenge of protein subnuclear localization due to the rapid growth of protein samples in dataset [4]. To better solve this problem, computational intelligence can be used for the protein subnuclear localization [5]. The critical issues of protein subnuclear localization using computational intelligence generally include two aspects: extract the useful features of protein sequences; select appropriate classification algorithm and evaluate the results [6].

During the last two decades, many techniques about the feature extraction of protein sequences have been proposed. In 1994, Nakashima et al. established a prediction method according to a

twenty-amino acid composition (AAC) and the frequency of residue pairs [7] that can calculate the occurrence frequency of 20 amino acids in protein sequence. Afterwards, AAC has been receiving a great deal of attention from researchers due to the excellent ability of reflection for the global sequence of protein. Thus, many prediction methods based on AAC have been proposed and most of them have achieved good results. For example, Reinhardt et al. constructed the first artificial neural network prediction system based on AAC [8]; Chou et al. used the AAC to solve the classification of membrane proteins [9]. However, AAC also has a defect, in that it ignores the interaction between the order of protein sequence and the residues. Therefore, to address this shortcoming, Chou et al. proposed the pseudo-amino acid composition (PseAAC) [10] that can include the characteristics of the physicochemical properties [11] such as negative, hydrophobic, and the order of amino acids and then the PseAAC method is widely used in protein structure prediction [12,13], protein functional prediction [14], protein-protein interaction [15] and subcellular localization [16] nowadays. The pseudo-amino acid can be calculated freely at online tools (http://bioinformatics.hitsz.edu.cn/Pse-in-One/) that are established by Liu et al. [17]. Subsequently, the position-specific score matrix (PSSM)—which can find the evolution information of proteins— was first introduced by Gribskov et al. in 1987 [18]. Along with the post-genome era coming, more studies tend to combine multiple feature extraction methods together to form a new method for localization purposes, for example, Shen and Chou proposed a web service for predicting subcellular localization by fusing PseAAC composition and PsePSSM [19]; Li et al. established a method which can fuse the features of PSSM, Gene Ontology (GO) and PROFEAT to predict the subcellular localization of bacterial proteins [20], etc. All of the above mentioned methods of feature extraction are common, and more methods can be found in these papers of Yao et al. and Chou et al. [21,22].

A key step of protein subnuclear localization is building a high-quality prediction model by using a machine learning method. For example, J et al. used deep neural networks to predict the subcellular localization only according to protein sequence information [23]; Chou and Shen established a novel ensemble classifier called Hum-PLoc using the K-nearest neighbor (KNN) algorithm [24]; Yu et al. used support vector machine (SVM) to predict the subcellular localization of Gram-negative bacterial protein [4]. According to the studies mentioned above, it can be found that an excellent classifier algorithm is useful to improve the performance of prediction, such as deep neural networks [25], SVM [26,27].

The above works focused on how to extract the feature information of protein sequences and construct classification models. However, the imbalance of protein datasets is generally ignored. Because of the rapid growth of protein sequences and their imbalanced development, protein datasets are seriously imbalanced. This could severely reduce the performance of any protein subnuclear localization methods. Therefore, the imbalance of protein datasets is a concerned problem in this work. In this study, a classification model is constructed. First, obtaining a PSSM by PSI-BLAST from two high-quality datasets which are widely used. Second, the dimension reduction method of kernel linear discriminant analysis (KLDA) is used to arrive at an optimal expression. Third, an oversampling method Radius-SMOTE is proposed to generate samples of minority classes. Finally, all of the samples are classified by random forest (RF).

The following content is arranged as follows: In Section 2, the proposed method Radius-SMOTE and some vital background knowledge will be introduced. Section 3 reports experimental results and related analysis. The last is a summary of this study.

## 2. Methods

Concerning the imbalance of the datasets used in protein subnuclear localization, an effective classification model based on Radius-SMOTE was proposed for subnuclear. The proposed classification model can be divided into three parts (Figure 1). First, PSSM is selected to extract the information of protein sequence that can capture evolution information more sufficient than any other method. Second, KLDA is used to get the feature vectors by removing the redundant information of PSSM. Because

PSMM is a higher dimensional matrix that includes much redundant information and nonlinear characteristics, KLDA is much suitable for processing this kind of biologic data. Third, since the protein datasets are seriously imbalanced, the proposed Radius-SMOTE is used to alleviate the problem of imbalanced data by creating samples of minority classes. Finally, all test samples are classified by RF. RF can balance errors of different classes to a certain extent, which can further reduce the negative effects of dataset imbalance.

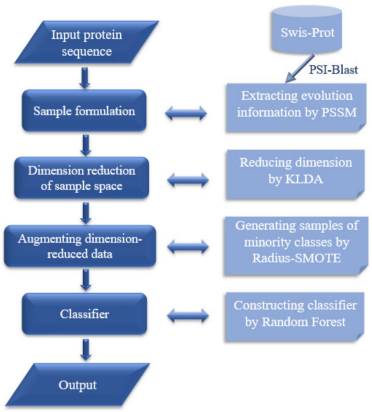



**Figure 1.** Flowchart of prediction process.

### 2.1. Position-Specific Score Matrix

Many homologous proteins may have the same structures and functions, thus PSSM is used to find the evolution information of protein sequence [28,29] in this study. PSSM, denoted by $P$, is defined by Equation (1).

$$P = \begin{bmatrix} E_{1\to1} & E_{1\to2} & \dots & E_{1\to j} & \dots & E_{1\to20} \\ E_{2\to1} & E_{2\to2} & \dots & E_{2\to j} & \dots & E_{2\to20} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ E_{i\to1} & E_{i\to2} & \dots & E_{i\to j} & \dots & E_{i\to20} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ E_{L\to1} & E_{L\to2} & \dots & E_{L\to j} & \dots & E_{L\to20} \end{bmatrix}, \quad (1)$$

where $E_{i\to j}$ represents the score of amino acid residue in the $i-$th protein $P$ being mutated to amino acid type $j$ during the evolution process; $L$ is the length of protein sequence; the numbers 1 to 20 are used to represent the native amino acid types.

In this work, $P$ can be generated by PSI-BLAST according to protein sequence and non-redundant (NR) database and the parameters of $E$-value and iteration are set as 0.001 and 3, respectively. According to Equation (1), $P$ is a $L \times 20$ matrix, because the value of $L$ is different for different protein sequences, Chou et al. proposed a representation method to standardize the representation of $P$ for each sample [19], which is shown in Equations (2) and (3).

$$\overline{P} = \left(\overline{E_1} \quad \overline{E_2} \quad \dots \overline{E_{20}}\right)^T, \quad (2)$$

$$\overline{E_j} = \frac{1}{L}\sum_{i-1}^{L} E_{i\to j}(j = 1, 2, \dots, 20), \quad (3)$$

where $\overline{E_j}$ represents the average score of the amino acid residues in the $j-$th protein $P$. According to the Equations (2) and (3), the generated $\overline{P}$ is a $20 \times 20$ matrix [30].

### 2.2. Kernel Linear Discriminant Analysis

Kernel linear discriminant analysis (KLDA) [31], a dimensionality reduction algorithm based on kernel method, is used to solve the problem of data linear inseparability in the original space and

it is a nonlinear extension of linear discriminate analysis (LDA). LDA is a dimensionality reduction algorithm and its essence is to map the data from the high dimensional space to the low dimensional linear subspace by the linear combination of features. However, the useful features of data in the real world are usually not the linear combinations of the original features. When there are a large number of nonlinear structures in the datasets, the mapping of linear dimensionality method cannot preserve these structures information, so kernel method is proposed to transform the problem of linear inseparability in the original space into a linear separable problem in the high dimensional eigenspace.

Next, the idea of KLDA is represented in detail. In this study, X is used to represent the protein dataset that contains $N$ samples classified in $k$ classes and $X = \{x_1, x_2, \ldots, x_N\} = C_1 \cup C_2 \ldots \cup C_k$, where $x_i (i = 1, 2, \ldots, N)$ and $C_i (i = 1, 2, \ldots, k)$ represent each sample and class of X, respectively. The process of dimension reduction can be divided into three steps:

1. Map input samples $x_1, x_2, \ldots, x_N$ to a higher dimensional space $F$ by nonlinear mapping function $\varnothing$ and the mapped samples can be expressed as $\varnothing(x_1), \varnothing(x_2), \ldots, \varnothing(x_N) \in F$;
2. Calculate the mean $m^{\varnothing}$ of all mapped samples and the mean $m_i^{\varnothing}$ of the mapped samples for class $C_i$ by the following formulas:

$$m^{\varnothing} = \frac{1}{N} \sum_{i=1}^{k} \sum_{x \in C_i} \varnothing(x), \tag{4}$$

$$m_i^{\varnothing} = \frac{1}{N_i} \sum_{x \in C_i} \varnothing(x), \tag{5}$$

   where $N_i$ means the number of samples belonging to the class $C_i$.
3. Calculate intraclass covariance matrix $S_{intra}^{\varnothing}$ and the interclass covariance matrix $S_{inter}^{\varnothing}$ for the whole mapped samples using the follow formulas:

$$S_{intra}^{\varnothing} = \sum\nolimits_{i=1}^{k} \sum\nolimits_{x \in C_i} \left[\varnothing(x) - m_i^{\varnothing}\right]\left[\varnothing(x) - m_i^{\varnothing}\right]^T, \tag{6}$$

$$S_{inter}^{\varnothing} = \sum\nolimits_{i=1}^{k} N_i \left[m_i^{\varnothing} - m^{\varnothing}\right]\left[m_i^{\varnothing} - m^{\varnothing}\right]^T, \tag{7}$$

4. Find the optimal projection direction $v$ by minimizing the intraclass distance and maximizing the interclass distance and the process can be expressed as Equation (8).

$$\max J_F(v) = \frac{v^T S_{inter}^{\varnothing} v}{v^T S_{intra}^{\varnothing} v}, \tag{8}$$

   Moreover, $v$ is the linear combination of $\varnothing(x_1), \varnothing(x_2), \ldots, \varnothing(x_N)$, which can be expressed as follows:

$$v = \sum_{i=1}^{N} a_i \varnothing(x_i), \tag{9}$$

   In Equation (8), $\varnothing$ is unknown and feature space $F$ may not be unique, which means $v$ cannot be computed directly. Thus, the kernel trick $K(x, y) = < \varnothing(x), \varnothing(y) >$ is introduced to solve this problem and Equations (4) and (5) can be transcribed as Equations (10) and (11).

$$v^T m^{\varnothing} = \frac{1}{N} \sum\nolimits_{j=1}^{N} \sum\nolimits_{n=1}^{N} a_j K\left(x_j, x_n\right) = a^T M, \tag{10}$$

$$v^T m_i^{\varnothing} = \frac{1}{N_i} \sum\nolimits_{j=1}^{N} \sum\nolimits_{n=1}^{N_i} a_j K\left(x_j, x_n\right) = a^T M_i, \tag{11}$$

Combined with Equations (6)–(11), the final criterion function of dimension reduction can be rewritten as follows:

$$\max J_{(a)} = \frac{a^T \widetilde{M} a}{a^T \widetilde{L} a}, \tag{12}$$

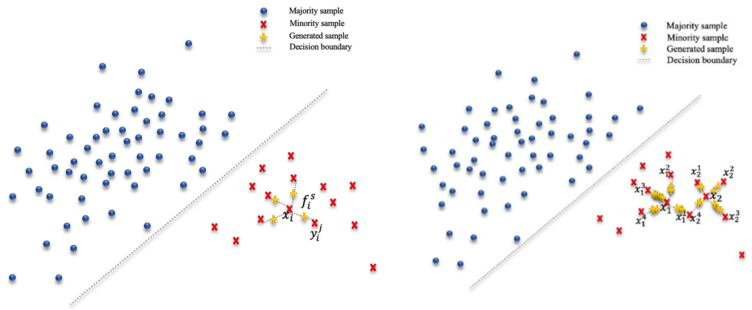
where $\widetilde{M} = \sum_{i=1}^{k} N_i (M_i - M)(M_i - M)^T$, $\widetilde{L} = \sum_{i=1}^{k} K_i \left(I - 1_{N_i}\right) K_i{}^T$.

5. Obtain the finally rank-reduction projective matrix Y by $Y = (a)^T X$.

### 2.3. The Proposed Radius-SMOTE

SMOTE was proposed by Chawla et al. in 2002 [32] and it is used to solve the problem of imbalanced data in this study. Supposing that each sample in minority class can be expressed as $x_i (i = 1, 2, \ldots, n)$, the $k (k < n)$ nearest neighbors of $x_i$ can be expressed as $x_i^j (j = 1, 2, \ldots, k)$, the imbalance rate of minority class is represented by $r$ and the generated sample can be represented as $y_i^s (s = 1, 2, \ldots, r)$.

Original SMOTE executes three steps to generate a new instance, as shown in Figure 2a. First, it chooses a random minority sample $x_i$; among its $k$ nearest minority class neighbors, selecting an instance $x_i^j$ randomly; finally, a new instance $y_i^s$ is generated between $x_i$ and $x_i^j$ by $y_i^s = x_i + rand(0,1) \times \left(x_i^j - x_i\right)$. The generated sample $y_i^s$ can only fall on the line between $x_i$ and $x_i^j$, this leads to $y_i^s$ containing few useful features and generated samples will overlap as shown in Figure 2b.




(**a**) (**b**)

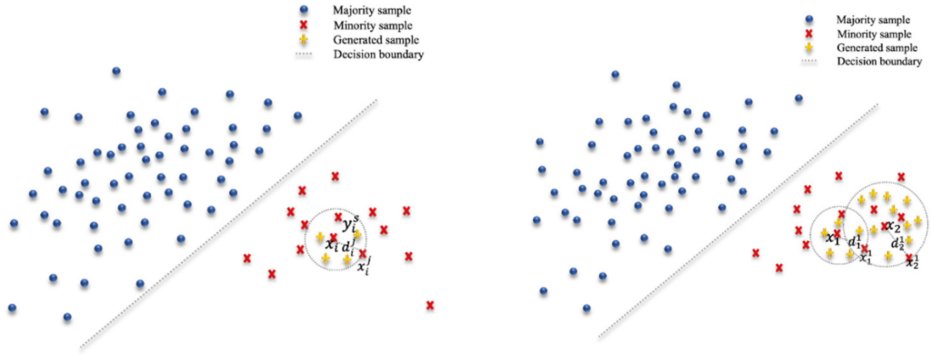
**Figure 2.** (**a**) Minority samples synthesized by SMOTE. (**b**) Overlap problem of minority samples synthesized by SMOTE.

To address this problem mentioned above, Radius-SMOTE is proposed to synthesize better samples of minority classes. The process of synthesizing new samples by Radius-SMOTE can be divided into five steps:

1. Calculate the imbalance rate $r$ by $r = \frac{n_{Majority} - n_{Minority}}{n_{Minority}}$;
2. Select a sample of minority $x_i$, calculate its $k$ nearest neighbors of minority class and select a neighbor at random represented as $x_i^j$;
3. Calculate the distance $d_i^j$ between $x_i$ and $x_i^j$ by Euclidean distance;
4. Randomly take values from $(0, d_i^j)$ to generate a vector $\left(d_{i,1}^j, d_{i,2}^j, \ldots, d_{i,c}^j\right)$, where $c$ means the characteristics dimension of $x_i$. Taking $x_i$ as the center and $d_i^j$ as the radius, thus a circle can be defined as shown in Figure 3a; at the same time, a new sample $y_i^s$ can be inserted within this circle by Equation (13).

$$y_i^s = (x_i^1, x_i^2, \ldots, x_i^c) + \left(d_{i,1}^j, d_{i,2}^j, \ldots, d_{i,c}^j\right) 0 \le d_{i,t}^j \le d_i^j, t = (1, 2, \ldots, c) \tag{13}$$

5. According to the imbalance rate, repeat Steps 2 to 4 $r$ times. Finally, $n \times r$ samples can be synthesized by Radius-SMOTE, as shown in Figure 3b.




(**a**) (**b**)

**Figure 3.** (**a**) Minority samples synthesized by Radius-SMOTE. (**b**) Reasonable distribution of minority samples synthesized by Radius-SMOTE.

### *2.4. Random Forest*

Random forest (RF) is an ensemble-based algorithm [33] that constructs many decision trees using original data and classifies the samples by combining the result of each decision tree. The construction of RF can be divided into five steps:

1. Randomly select the training subsets from the original dataset;
2. Set up a decision tree for each training subset, in which each decision tree does not need to be pruned;
3. Construct RF model by formed forest that is composed of tens or hundreds of decision trees;
4. Classify a new sample, each decision tree in the forest gives an individual result;
5. Calculate the votes of each class and get the final class which has the supreme votes.

## 3. Experiments

In this section, the experimental results of two datasets based on our proposed method are introduced and analyzed.

### *3.1. Datasets*

Two different benchmark datasets were chosen to conduct the numerical experiments in this work and the information could be found in Tables 1 and 2. The first dataset was named Dataset 1 [19], constructed by Shen and Chou. This set contains 714 protein sequences located at nine subnuclear. The second dataset was named Dataset 2 [34], constructed by Kumar et al. and contains 669 protein sequences located at ten subnuclear sites.

**Table 1.** Constitutions of protein benchmark Dataset 1.

| Class | Subnuclear Localization Name | Number |
|---|---|---|
| 1 | Chromatin proteins (Ca) | 99 |
| 2 | Heterochromatin proteins (Ht) | 22 |
| 3 | Nuclear envelope proteins (Ne) | 61 |
| 4 | Nuclear matrix proteins (Nm) | 29 |
| 5 | Nuclear pore complex proteins (Nc) | 79 |
| 6 | Nuclear speckle proteins (Ns) | 67 |
| 7 | Nucleolus proteins (Nl) | 307 |
| 8 | Nucleoplasm proteins (Np) | 37 |
| 9 | Nuclear PML body proteins (Nb) | 13 |
| Sum | | 714 |

**Table 2.** Constitutions of protein benchmark Dataset 2.

| Class | Subnuclear Localization Name | Number |
|---|---|---|
| 1 | Centromere proteins (Cn) | 86 |
| 2 | Chromosome proteins (Co) | 113 |
| 3 | Nuclear speckle proteins (Ns) | 50 |
| 4 | Nucleolus proteins (Nl) | 294 |
| 5 | Nuclear envelope proteins (Ne) | 17 |
| 6 | Nuclear matrix proteins (Nm) | 18 |
| 7 | Nucleoplasm proteins (Np) | 30 |
| 8 | Nuclear pore complex proteins (Nc) | 12 |
| 9 | Nuclear proteins (Na) | 12 |
| 10 | PML body Telomere(Pb) | 37 |
| Sum | | 669 |

### 3.2. Evaluation Indexes

In this work, the Jackknife test was used to examine the performance of the proposed model. This model is considered the most reasonable cross-validation method [35,36]. In a dataset, the Jackknife test supposes $n$ samples which will be selected as test samples one-by-one, and the remaining $n-1$ data used as training samples simultaneously.

In order to evaluate the degree of dataset imbalance and the performance of the proposed method, five different indices of the Jackknife test can be defined as follows:

$$IR = \frac{\min(n_i)}{\max(n_i)} (i = 1,\ 2,\ \ldots, k), \tag{14}$$

where $n_i$ means the $i-$ th class, $IR$ means the degree of dataset imbalance and the smaller $IR$ means that the data are more imbalance.

$$Sensitivity(Se) = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \tag{15}$$

$$Specificity(Sp) = \frac{\mathrm{TN}}{\mathrm{FP} + \mathrm{TN}}, \tag{16}$$

$$Matthews\ Correlation\ Codficient(MCC) = \frac{\mathrm{TP} \times \mathrm{TN} - \mathrm{FP} \times \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}}, \tag{17}$$

$$Accuracy(ACC) = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FN} + \mathrm{FP} + \mathrm{TN}}, \tag{18}$$

where TP and TN denote the number of true positive and true negative, which are correctly labeled, and FP and FN denote the number of false positives and false negatives that are incorrectly labeled. *Se* denotes the rate of positive samples that are correctly labeled in all positive samples, *Sp* denotes the rate of negative samples that are correctly labeled in all negative samples; in addition, *ACC* denotes the rate of all correctly labeled samples, and the range of *Se*, *Sp* and *ACC* is [0, 1]. The *MCC* denotes the relationship between original classes and prediction classes, and its range is [−1, 1], when the value of *MCC* is equal to 1, the performance of prediction is perfect; when the value of *MCC* is equal to 0, the performance of prediction is random; when the value of *MCC* is equal to −1, the performance of prediction is worst.

### 3.3. The Analysis of Unbalance Datasets

The imbalance degree of Dataset 1 was calculated in both cases and shown in Figure 4. In the first case, the value of *IR* was 0.042 based on the original data; in the second case, the value of *IR* was 0.769 based on the expanded data by SMOTE or Radius-SMOTE. The *IR* of Dataset 1 increased to 0.727, which means that the Radius-SMOTE was effective for reducing the dataset imbalance.

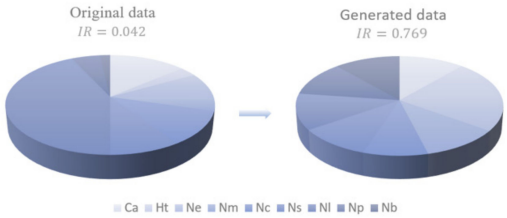



**Figure 4.** Analysis of imbalanced Dataset 1.

Based on Dataset 2, the values of *IR* were calculated (Figure 5), which can be found in the last column, where 0.041 means the imbalance degree of original data and 0.769 means the imbalance degree of expanded data by SMOTE or Radius-SMOTE. The *IR* of Dataset 2 increased by 0.728. According to the process of Radius-SMOTE, the imbalance of Dataset 2 is solved.

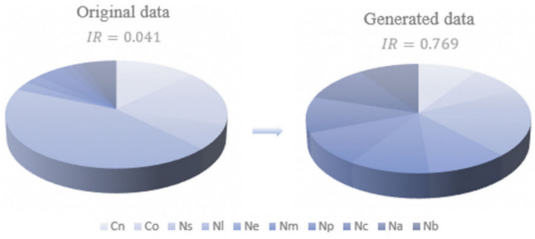



**Figure 5.** Analysis of imbalanced Dataset 2.

### *3.4. The Overall Accuracy Analysis of the Proposed Method*

In this study, the features of protein sequences were extracted by PSSM and the prediction of subnuclear localization was obtained by RF. To demonstrate the performance of the proposed method, three experiments were performed on each of two benchmark datasets. The experimental result by the Jackknife cross-validation is shown in Figure 6.

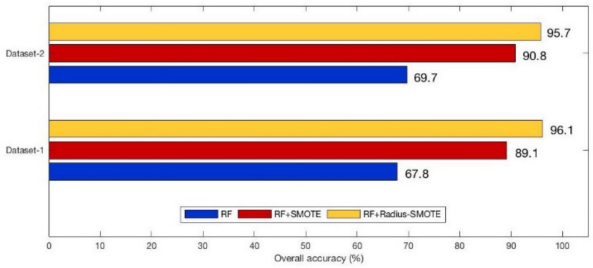



**Figure 6.** Overall accuracy of each dataset.

From Figure 6, the maximum accuracies of Dataset 1 and Dataset 2 were 96.1% and 95.7%. The classification accuracy of RF + SMOTE were all higher than RF, which means the oversampling method could effectively improve the classification accuracy. The accuracy performed by RF + Radius-SMOTE were all higher than the accuracy calculated by RF with SMOTE, which means the method of Radius-SMOTE could achieve good classification performance.

#### 3.4.1. The Relationship between *k* in Radius-SMOTE with Overall Accuracy

In the process of expanding datasets, the selection of *k* nearest neighbor plays an important role in classification. The experimental results influenced by different *k* in Radius-SMOTE are shown in Figure 7. The blue polyline denotes the overall accuracy of Dataset 1; the red polyline denotes the overall accuracy of Dataset 2. In this study, the interval of *k* is set to [1:9]. For Dataset 1, the minimum accuracy is 84.8 ($k = 1$); along with the increase of *k*, the maximum accuracy is 96.1% ($k = 7$). For Dataset

2, the minimum accuracy was 84.3% ($k = 1$) and the maximum accuracy was 95.7% ($k = 5$). In general, the accuracy of each dataset reaches the lowest when $k = 1$, which means the selection of nearest neighbor in Radius-SMOTE was significant.

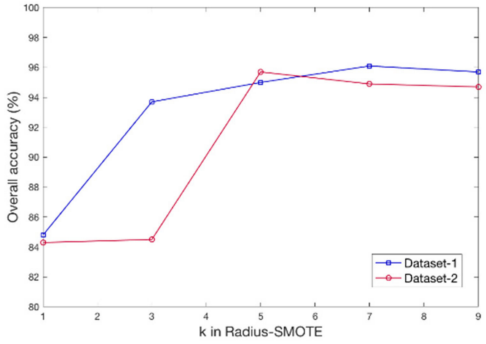



**Figure 7.** Relationship between overall accuracy of each dataset and the k in Radius-SMOTE.

#### 3.4.2. The Relationship between *k* in RF with Overall Accuracy

RF was used for classification in this study. The parameter *k* of RF represents the number of trees in the process of classification and it had influence on the overall accuracy of the proposed method. In Figure 8, the interval of *k* was set in the range of [30:210]. For Dataset 1, the overall accuracy was lowest 95.1% ($k = 30$); in addition, the accuracy gradually turns to be stable when *k* was greater than 30 and achieves the highest value 96.71% ($k = 210$). For Dataset 2, the overall accuracy was highest 95.7% ($k = 150$); in addition, the difference of accuracy caused by *k* was 4.5 higher than that of Dataset 1, which means the effect of *k* on Dataset 2 was more obvious than Dataset 1. Figure 8 shows that the blue polyline was above the red polyline, which means that Dataset 1 had a better classification performance than Dataset 2.

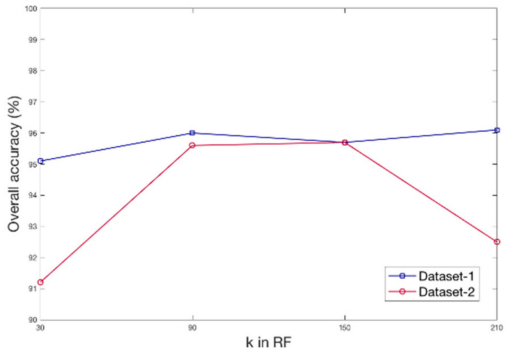



**Figure 8.** Relationship between overall accuracy of each dataset and the k in random forest (RF).

### *3.5. The Analysis for Evaluation Indexes of Different Methods*

In this part, three experiments were performed on each of two benchmark datasets. All datasets were mapped into low-dimensional space by KLDA and then classified by RF. Four evolution indices of datasets were calculated based on the Jackknife tests and shown in Tables 3 and 4. In the first experiment, the value of evolution indices is found in the third column of Tables 3 and 4, based on original data; In the second experiment, the value of evolution indices are found in the fourth column of Tables 3 and 4, based on expanded data by SMOTE; In the third experiment, the value of evolution indices is found in the fifth column of Tables 3 and 4, based on expanded data by Radius-SMOTE.

As shown in Table 3, for the first experiment, the class Nl (307 samples) had the highest Se value of 0.958 and the lowest Sp value of 0.766; and the class Nb (13 samples) had the lowest Se value of 0.154 and the highest Sp value of 1, which means that the classification results were easily affected by

the number of samples in different categories. In addition, most of the value of MCC was less than 0.5, which means that the classification performance of RF without oversampling was not good. For the second experiment, the values of MCC were more than 0.83 (only the MCC of Nl was 0.76) and the values of ACC were more than 0.94, which demonstrated that the classification performance with SMOTE was good. For the third experiment, the Se of all classed were higher than 0.92 (only the Se of Nl was 0.876) and the values of Sp were higher than 0.988. The values of ACC were higher than 0.98 which were very close to 1 and the values of MCC were higher 0.9, which showed that the classification performance with Radius-SMOTE was better than that of others.

As shown in Table 4, for the first experiment, the Se values of Ne and Nm were 0.118 and 0.11, respectively, which was approximately equal to 0.11; the MCC values of all classes were lower than the 0.52 (only the MCC of Nl was 0.759), which means that the classification performance without oversampling was bad. For the second experiment, the ACC values of all classes were higher than 0.94 and the MCC values of Cn, Co and Nl were between 0.74 to 0.88; and the remaining seven types of dataset were higher than 0.92. It was found that the MCC and ACC of the first dataset were larger than that of the second class, which illustrated that the classification with SMOTE was better.

**Table 3.** Four evolution indices of each class in Dataset 1.

| Dataset 1 | Index | Original Data (KLDA + RF) | SMOTE (KLDA + RF) | Radius-SMOTE (KLDA + RF) |
|---|---|---|---|---|
| Ca | Se | 0.546 | 0.815 | 0.953 |
| | Sp | 0.890 | 0.985 | 0.988 |
| | ACC | 0.832 | 0.964 | 0.984 |
| | MCC | 0.423 | 0.830 | 0.925 |
| Ht | Se | 0.455 | 0.980 | 0.984 |
| | Sp | 0.996 | 0.973 | 0.999 |
| | ACC | 0.972 | 0.975 | 0.997 |
| | MCC | 0.604 | 0.897 | 0.985 |
| Ne | Se | 0.508 | 0.882 | 0.980 |
| | Sp | 0.940 | 0.988 | 0.992 |
| | ACC | 0.891 | 0.974 | 0.991 |
| | MCC | 0.451 | 0.883 | 0.958 |
| Nm | Se | 0.31 | 0.921 | 0.979 |
| | Sp | 0.986 | 0.994 | 0.996 |
| | ACC | 0.947 | 0.985 | 0.994 |
| | MCC | 0.393 | 0.928 | 0.971 |
| Nc | Se | 0.519 | 0.798 | 0.924 |
| | Sp | 0.919 | 0.992 | 0.998 |
| | ACC | 0.863 | 0.972 | 0.991 |
| | MCC | 0.436 | 0.839 | 0.946 |
| Ns | Se | 0.388 | 0.873 | 0.974 |
| | Sp | 0.927 | 0.996 | 0.993 |
| | ACC | 0.863 | 0.982 | 0.991 |
| | MCC | 0.326 | 0.907 | 0.955 |
| Nl | Se | 0.958 | 0.886 | 0.876 |
| | Sp | 0.766 | 0.949 | 0.994 |
| | ACC | 0.872 | 0.941 | 0.98 |
| | MCC | 0.747 | 0.761 | 0.901 |
| Np | Se | 0.432 | 0.865 | 0.983 |
| | Sp | 0.985 | 0.994 | 0.997 |
| | ACC | 0.945 | 0.978 | 0.995 |
| | MCC | 0.522 | 0.897 | 0.977 |
| Nb | Se | 0.154 | 0.983 | 1.000 |
| | Sp | 1.000 | 0.996 | 0.999 |
| | ACC | 0.978 | 0.994 | 0.999 |
| | MCC | 0.388 | 0.975 | 0.994 |

**Table 4.** Four evolution indices of each class in Dataset 2.

| Dataset 2 | Index | Original Data (KLDA + RF) | SMOTE (KLDA + RF) | Radius-SMOTE (KLDA + RF) |
|---|---|---|---|---|
| Cn | Se | 0.698 | 0.849 | 0.969 |
| | Sp | 0.858 | 0.993 | 0.991 |
| | ACC | 0.834 | 0.978 | 0.989 |
| | MCC | 0.479 | 0.878 | 0.939 |
| Co | Se | 0.726 | 0.774 | 0.854 |
| | Sp | 0.846 | 0.992 | 0.992 |
| | ACC | 0.822 | 0.973 | 0.980 |
| | MCC | 0.515 | 0.821 | 0.869 |
| Ns | Se | 0.320 | 0.900 | 0.988 |
| | Sp | 0.953 | 0.993 | 0.994 |
| | ACC | 0.893 | 0.984 | 0.994 |
| | MCC | 0.310 | 0.908 | 0.963 |
| Nl | Se | 0.932 | 0.881 | 0854 |
| | Sp | 0.817 | 0.948 | 0.995 |
| | ACC | 0.881 | 0.940 | 0.979 |
| | MCC | 0.759 | 0.742 | 0.890 |
| Ne | Se | 0.118 | 0.965 | 0.983 |
| | Sp | 0.998 | 0.997 | 0.999 |
| | ACC | 0.967 | 0.993 | 0.998 |
| | MCC | 0.271 | 0.967 | 0.988 |
| Nm | Se | 0.111 | 0.962 | 0.983 |
| | Sp | 0.992 | 0.976 | 0.996 |
| | ACC | 0.959 | 0.992 | 0.995 |
| | MCC | 0.175 | 0.961 | 0.975 |
| Np | Se | 0.233 | 0.926 | 0.993 |
| | Sp | 0.979 | 0.990 | 0.997 |
| | ACC | 0.934 | 0.983 | 0.996 |
| | MCC | 0.278 | 0.911 | 0.980 |
| Nc | Se | 0.333 | 0.962 | 1.000 |
| | Sp | 0.996 | 0.997 | 0.999 |
| | ACC | 0.979 | 0.993 | 0.999 |
| | MCC | 0.462 | 0.965 | 0.994 |
| Na | Se | 0.167 | 0.951 | 1.000 |
| | Sp | 0.998 | 0.998 | 0.999 |
| | ACC | 0.977 | 0.993 | 0.999 |
| | MCC | 0.326 | 0.964 | 0.994 |
| Pb | Se | 0.460 | 0.915 | 0.996 |
| | Sp | 0.980 | 0.993 | 0.996 |
| | ACC | 0.941 | 0.985 | 0.996 |
| | MCC | 0.519 | 0.920 | 0.979 |

### *3.6. Comparisons with Other Methods*

We compared the performance between our proposed method and state-of-arts methods based on two benchmark datasets. Contrast experiments were divided into two categories. These were then compared with methods of SMOTE-variants and methods of protein subnuclear localization. Based on the same dataset, the Jackknife test was used to compare the performance of the proposed method with other methods previously introduced in detail in this section.

#### 3.6.1. Comparison of Dataset 1

Based on Dataset 1, the proposed method was first compared with other oversampling methods, as shown in Table 5. It was found that Radius-SMOTE had better performance for protein subnuclear localization.

**Table 5.** Prediction results of Dataset 1 based on different oversampling methods.

| Oversampling Methods | Overall Accuracy (%) |
|---|---|
| Borderline_SMOTE1 [37] | 92.4 |
| Borderline_SMOTE2 [37] | 93.3 |
| SVM_balance [38] | 93.1 |
| NRAS [39] | 89.3 |
| The proposed Radius-SMOTE | 96.1 |

Then the proposed method was compared with six state-of-arts methods of protein subnuclear localization, as shown in Table 6. Through the comparison, the highest classification accuracy of Dataset 1 was 96.1% obtained by the proposed method.

**Table 6.** Prediction results of Dataset 1 obtained by different methods of protein subnuclear localization.

| Methods (Jackknife Test) | Overall Accuracy (%) |
|---|---|
| Fusion of PsePSSM and PseAAC-KNN [19] | 67.4 |
| PseAAPSSM-LDA-KNN [6] | 88.1 |
| DipPSSM-LDA-KNN [6] | 95.9 |
| AACPSSM with fused kernel-KLDA-KNN [40] kernel | 94.7 |
| PseAAC-A hybrid-classifier-based SVM [41] classifier | 81.2 |
| CoPSSM-KLDA-based DGGA-KNN [5] | 90.3 |
| The proposed PSSM-Radius-SMOTE-KLDA-RF | 96.1 |

The comparison of MCC between proposed method and other prediction methods of protein subnuclear localization can be found in Figure 9. The biggest difference of MCC for different methods were 0.39, 0.69, 0.45 and 0.093, respectively. The smallest difference was 0.093 obtained by the proposed method, which means that the classification results of majority and minority classes are more accurate than that of other methods (Figure 9).

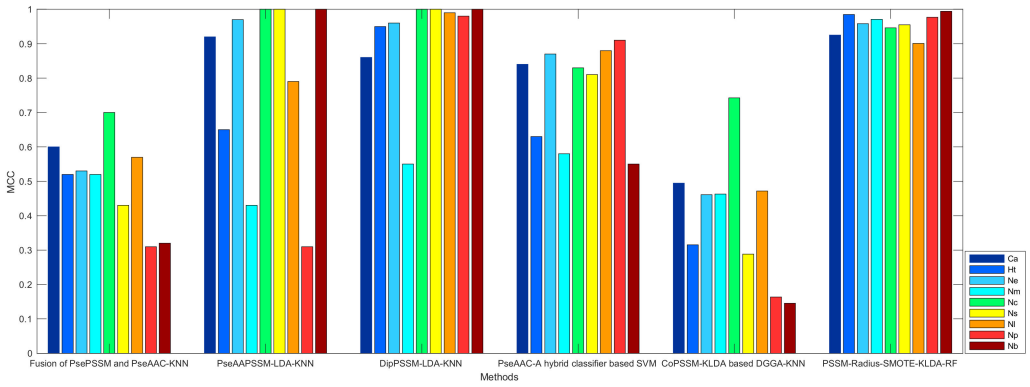



**Figure 9.** MCC of each class of Dataset 1 by different methods of protein subnuclear localization.

### 3.6.2. Comparison of Dataset 2

For Dataset 2, the four comparison methods of oversampling were the same as those in Dataset 1 (Table 7). The performance of Radius-SMOT on Dataset 2 was also the best, which illustrates the effectiveness of Radius-SMOTE.

**Table 7.** Prediction results of Dataset 2 based on different oversampling methods.

| Oversampling Methods | Overall Accuracy (%) |
|---|---|
| Borderline_SMOTE1 [37] | 93.8 |
| Borderline_SMOTE2 [37] | 93.6 |
| SVM_balance [38] | 95.6 |
| NRAS [39] | 88.1 |
| The proposed Radius-SMOTE | 95.7 |

At the same time, the proposed method was also compared with three methods of protein subnuclear localization (Table 8). The results illustrate that the proposed method had better performance than other methods.

**Table 8.** Prediction results of Dataset 2 obtained by different methods of protein subnuclear localization.

| Methods (Jackknife Test) | Overall Accuracy (%) |
|---|---|
| SSLD and AAC-SVM [34] | 81.5 |
| PseAAPSSM-LDA-KNN [6] | 84 |
| CoPSSM-KLDA-based DGGA-KNN [5] | 87.4 |
| The proposed PSSM-Radius-SMOTE-KLDA-RF | 95.7 |

From Figure 10, the MCC of each class obtained by the proposed method was higher than that of other three methods. Thus, the classification performance of the proposed method in Dataset 2 was better than other methods.

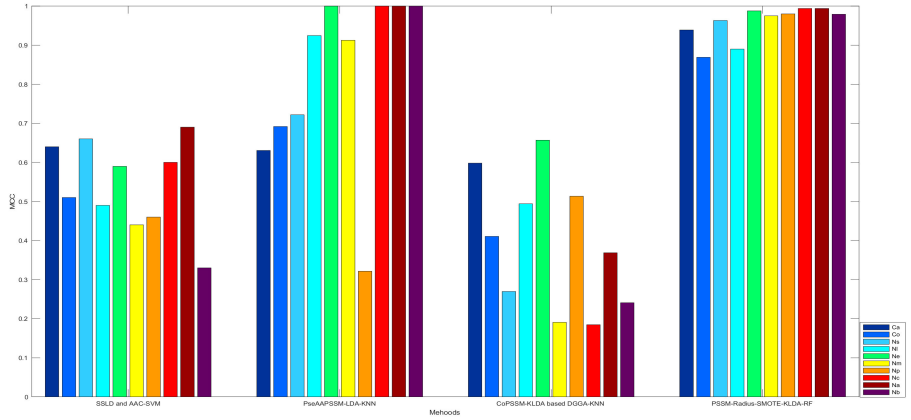



**Figure 10.** MCC of each class on Dataset 2 by different methods of protein subnuclear localization.

## 4. Conclusions

This study proposes an effective protein subnuclear localization method, with the aim of overcoming the imbalance of protein datasets and improving the prediction accuracy of protein subnuclear localization. First, the features of protein are represented by PSSM, which can extract the evolution information of proteins. Second, the dimensions of feature vector are reduced by KLDA, which can reduce the redundant information of protein dataset. Third, Radius-SMOTE, which is based on SMOTE, is used to solve the imbalance problem of protein dataset. Finally, the subnuclear localization of proteins is predicted by RF.

According to the Jackknife test, the overall accuracy of the proposed method in two benchmark datasets can reach 96.1% and 95.7%. From the experimental results, the following conclusions can be drawn:

1. The imbalance of protein datasets has a great impact on the prediction accuracy of protein subnuclear localization;

2. The proposed method can efficiently improve the prediction accuracy of protein subnuclear localization by solving the imbalanced problem of protein datasets;
3. The combination of KLDA and RF can improve the classification accuracy of protein at the subnuclear level.

**Author Contributions:** Conceptualization, L.W.; data curation, S.H.; formal analysis, Q.J. and X.J.; methodology, L.W.; project administration, S.Y.; resources, F.W.; software, L.W. and F.W.; supervision, Q.J. and X.J.; validation, L.W.; writing—original draft, L.W.; writing—review & editing, S.H. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China (No. 61863036), China Postdoctoral Science Foundation (2020T130564, 2019M653507), Postdoctoral Science Foundation of Yunnan Province in China and Yunnan University's Research Innovation Fund for Graduate Students (No. 2019164).

**Conflicts of Interest:** The authors declare no conflict of interest to publish this study.

## References

1. Garapati, H.S.; Male, G.; Mishra, K. Predicting subcellular localization of proteins using protein-protein interaction data. *Genomics* **2020**, *112*, 2361–2368. [CrossRef]
2. Javed, F.; Hayat, M. Predicting subcellular localization of multi-label proteins by incorporating the sequence features into Chou's PseAAC. *Genomics* **2019**, *111*, 1325–1332.
3. Gardy, J.L.; Brinkman, F.S. Methods for predicting bacterial protein subcellular localization. *Nat. Rev. Microbiol.* **2006**, *4*, 741–751.
4. Yu, B.; Li, S.; Chen, C.; Xu, J.; Qiu, W.; Wu, X.; Chen, R. Prediction subcellular localization of Gram-negative bacterial proteins by support vector machine using wavelet denoising and Chou's pseudo amino acid composition. *Chemom. Intell. Lab. Syst.* **2017**, *167*, 102–112.
5. Wang, S.; Yue, Y. Protein subnuclear localization based on a new effective representation and intelligent kernel linear discriminant analysis by dichotomous greedy genetic algorithm. *PLoS ONE* **2019**, *13*, e0195636. [CrossRef]
6. Wang, S.; Liu, S. Protein sub-nuclear localization based on effective fusion representations and dimension reduction algorithm LDA. *Int. J. Mol. Sci.* **2015**, *16*, 30343–30361.
7. Nakashima, H.; Nishikawa, K. Discrimination of intracellular and extracellular proteins using amino acid composition and residue-pair frequencies. *J. Mol. Biol.* **1994**, *238*, 54–61.
8. Reinhardt, A. Using neural networks for prediction of the subcellular location of proteins. *Nucleic Acids Res.* **1998**, *26*, 2230–2236. [CrossRef]
9. Chou, K.C.; Shen, H.B. Predicting eukaryotic protein subcellular location by fusing optimized evidence-theoretic K-nearest neighbor classifiers. *J. Proteome Res.* **2006**, *5*, 1888–1897. [CrossRef]
10. Chou, K.C. Prediction of protein cellular attributes using pseudo-amino acid composition. *Proteins* **2001**, *43*, 246–255. [CrossRef]
11. Chou, K.C. Prediction of protein subcellular locations by incorporating quasi-sequence-order effect. *Biochem. Biophys. Res. Commun.* **2000**, *278*, 477–483. [CrossRef] [PubMed]
12. Hayat, M.; Iqbal, N. Discriminating protein structure classes by incorporating Pseudo Average Chemical Shift to Chou's general PseAAC and Support Vector Machine. *Comput. Methods Programs Biomed.* **2014**, *116*, 184–192. [CrossRef]
13. Nanni, L.; Brahnam, S.; Lumini, A. Prediction of protein structure classes by incorporating different protein descriptors into general Chou's pseudo amino acid composition. *J. Theor. Biol.* **2014**, *360*, 109–116. [CrossRef] [PubMed]
14. Zhou, X.B.; Chen, C.; Li, Z.C.; Zou, X.Y. Using Chou's amphiphilic pseudo-amino acid composition and support vector machine for prediction of enzyme subfamily classes. *J. Theor. Biol.* **2007**, *248*, 546–551. [CrossRef] [PubMed]
15. Liu, L.; Cai, Y.; Lu, W.; Feng, K.; Peng, C.; Niu, B. Prediction of protein-protein interactions based on PseAA composition and hybrid feature selection. *Biochem. Biophys. Res. Commun.* **2009**, *380*, 318–322. [CrossRef] [PubMed]

16. Li, B.; Cai, L.; Liao, B.; Fu, X.; Bing, P.; Yang, J. Prediction of Protein Subcellular Localization Based on Fusion of Multi-view Features. *Molecules* **2019**, *24*, 919. [CrossRef] [PubMed]
17. Liu, B.; Liu, F.; Wang, X.; Chen, J.; Fang, L.; Chou, K.C. Pse-in-One: A web server for generating various modes of pseudo components of DNA, RNA, and protein sequences. *Nucleic Acids Res.* **2015**, *43*, 65–71. [CrossRef]
18. Gribskov, M.; McLachlan, A.; Eisenberg, D. Profile analysis: Detection of distantly related proteins. *Proc. Natl. Acad. Sci. USA* **1987**, *84*, 4355–4358. [CrossRef]
19. Shen, H.B.; Chou, K.C. Nuc-PLoc: A new web-server for predicting protein subnuclear localization by fusing PseAA composition and PsePSSM. *Protein Eng. Des. Sel.* **2007**, *20*, 561–567. [CrossRef]
20. Li, L.; Yu, S.; Xiao, W.; Li, Y.; Li, M.; Huang, L.; Zheng, X.; Zhou, S.; Yang, H. Prediction of bacterial protein subcellular localization by incorporating various features into Chou's PseAAC and a backward feature selection approach. *Biochimie* **2014**, *104*, 100–107. [CrossRef]
21. Yao, Y.; Xu, H.; He, P.; Dai, Q. Recent advances on prediction of protein subcellular localization. *Mini-Rev. Org. Chem.* **2015**, *12*, 481–492.
22. Chou, K.C.; Shen, H.B. Recent progress in protein subcellular location prediction. *Anal. Biochem.* **2007**, *370*, 1–16.
23. Armenteros, J.J.; Sonderby, C.K.; Sonderby, S.K.; Nielsen, H.; Winther, O. DeepLoc: Prediction of protein subcellular localization using deep learning. *Bioinformatics* **2017**, *33*, 3387–3395.
24. Chou, K.C.; Shen, H.B. Hum-PLoc: A novel ensemble classifier for predicting human protein subcellular localization. *Biochem. Biophys. Res. Commun.* **2006**, *347*, 150–157.
25. Science, C.; Trust, W. Accurate Classification of Protein Subcellular Localization from High-Throughput Microscopy Images Using Deep Learning. *Genes Genomes Genet.* **2017**, *7*, 1385–1392.
26. Hasan, M.A.; Ahmad, S.; Molla, M.K. Protein subcellular localization prediction using multiple kernel learning based support vector machine. *Mol. Biosyst.* **2017**, *13*, 785–795.
27. Tu, Y.K.; Hong, Y.Y.; Chen, Y.C. Finite element modeling of kirschner pin and bone thermal contact during drilling. *Life Sci. J.* **2009**, *6*, 23–27.
28. Li, Y.; Li, L.P.; Wang, L.; Yu, C.Q.; Wang, Z.; You, Z.H. An Ensemble Classifier to Predict Protein–Protein Interactions by Combining PSSM-based Evolutionary Information with Local Binary Pattern Model. *Int. J. Mol. Sci.* **2019**, *20*, 3511.
29. Xiao, X.; Wu, Z.C.; Chou, K.C. iLoc-Virus: A multi-label learning classifier for identifying the subcellular localization of virus proteins with both single and multiple sites. *J. Theor. Biol.* **2011**, *284*, 42–51.
30. Chou, K.C.; Wu, Z.C.; Xiao, X. iLoc-Euk: A multi-label classifier for predicting the subcellular localization of singleplex and multiplex eukaryotic proteins. *PLoS ONE* **2011**, *6*, e18258.
31. Mika, S.; Ratsch, G.; Weston, J.; Scholkopf, B.; Mullers, K.R. Fisher discriminant analysis with kernels. *IEEE Signal. Process. Soc. Workshop* **1999**, 41–48. [CrossRef]
32. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic minority over-sampling technique. *J. Artif. Intell. Res.* **2002**, *16*, 321–357.
33. Gajowniczek, K.; Grzegorczyk, I.; Ząbkowski, T.; Bajaj, C. Weighted Random Forests to Improve Arrhythmia Classification. *Electronics* **2020**, *9*, 99.
34. Kumar, R.; Jain, S.; Kumari, B.; Kumar, M. Protein sub-nuclear localization prediction using SVM and Pfam domain information. *PLoS ONE* **2014**, *9*, e98345. [CrossRef]
35. Chou, K.C.; Liu, W.M.; Maggiora, G.M.; Zhang, C.T. Prediction and classification of domain structural classes. *Proteins Struct. Funct. Genet.* **1998**, *31*, 97–103. [CrossRef]
36. Cheng, X.; Zhao, S.G.; Lin, W.Z.; Xiao, X.; Chou, K.C. PLoc-mAnimal: Predict subcellular localization of animal proteins with both single and multiple sites. *Bioinformatics* **2017**, *33*, 3524–3531.
37. Han, H.; Wang, W.Y.; Mao, B.H. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. *Adv. Intell. Comput.* **2005**, *3644*, 878–887.
38. Farquad, M.A.H.; Bose, I. Preprocessing unbalanced data using support vector machine. *Decis. Support. Syst.* **2012**, *53*, 226–233.
39. William, A.R. Noise Reduction A Priori Synthetic Over-Sampling for class imbalanced data sets. *Inf. Sci.* **2017**, *408*, 146–161.

40. Yue, Y.; Wang, S. Protein subnuclear location based on KLDA with fused kernel and effective fusion representation. In Proceedings of the 6th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 21–22 October 2017.
41. Song, C. Protein Subnuclear Localization Using a Hybrid Classifier Combined with Chou's Pseudo Amino Acid Composition. In Proceedings of the 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Beijing, China, 13–15 October 2018.