processes

MDPI

Article

Virtual Screening of Drug Proteins Based on the Prediction Classification Model of Imbalanced Data Mining

Lili Yin [1], Xiaokang Du [1], Chao Ma [1,*] and Hengwen Gu [2]

[1] School of Computer Science and Technology, Harbin University of Science and Technology, Harbin 150080, China; yinlili234@126.com (L.Y.); duxiaokang9711@126.com (X.D.)

[2] Department of Military Industry Development, No. 703 Research Institute, China State Shipbuilding Company Limited, Harbin 150078, China; guhengwen234@126.com

* Correspondence: machao8396@163.com

Abstract: We propose a virtual screening method based on imbalanced data mining in this paper, which combines virtual screening techniques with imbalanced data classification methods to improve the traditional virtual screening process. First, in the actual virtual screening process, we apply k-means and smote heuristic oversampling method to deal with imbalanced data. Meanwhile, to enhance the accuracy of the virtual screening process, a particle swarm optimization algorithm is introduced to optimize the parameters of the support vector machine classifier, and the concept of ensemble learning is brought in. The classification technique based on particle swarm optimization, support vector machine and adaptive boosting is used to screen the molecular docking conformation to improve the accuracy of the prediction. Finally, in the experimental construction and analysis section, the proposed method was validated using relevant data from the protein data bank database and PubChem database. The experimental results indicated that the proposed method can effectively improve the accuracy of virus screening and has practical guidance for new drug development. This research regards virtual screening as a problem of imbalanced data classification, which has obvious guiding significance and also provides a certain reference for the problems faced by virtual screening technology.

Keywords: imbalanced data; k-means and smote; PSO_SVM_AdaBoost; image processing

check for updates

Citation: Yin, L.; Du, X.; Ma, C.; Gu, H. Virtual Screening of Drug Proteins Based on the Prediction Classification Model of Imbalanced Data Mining. *Processes* **2022**, *10*, 1420. https://doi.org/10.3390/pr10071420

Academic Editor: Massimiliano Barolo

Received: 16 June 2022

Accepted: 18 July 2022

Published: 21 July 2022

1. Introduction

With the continuous improvement of computer hardware and software, computer storage technology is also evolving. The era of information and data has arrived. Therefore, it is an inevitable trend to use data mining, machine learning and other related technologies to extract implicit information from production, operation and daily life in favor of massive data. Among many forms of data, a special kind of dataset exists—an imbalanced dataset (IDS) [1]. It is of great practical significance and research value to study the classification probability problem of an imbalanced dataset. In the field of drug development, the problem of imbalanced data classification is gaining more and more attention, where the virtual screening technique is a typical imbalanced data classification problem.

In recent years, the continuous discovery of small molecular organic compounds such as natural compounds and synthetic compounds has provided huge data resources for drug research and development. However, traditional drug design processes usually adopt the method of clinical verification, which is inefficient in the face of the information data of tens of millions of compounds. Since the 1980s, computers have been used in the field of drug discovery because of their efficient computing power. Computers are used for guidance, design and prediction before compound synthesis, and for analysis, induction, supplement and improvement after compound synthesis, which not only greatly saves time, but also improves the accuracy of drug design. Virtual screening technology is a typical representative of computer-aided drug design [2].

Virtual screening technology simulates the drug discovery process on the computer and has gradually become one of the important auxiliary means of drug research and development. Virtual screening method is mainly divided into receptor-based virtual screening and ligand-based virtual screening. At present, LBVS (ligand-based virtual screening) method depends on the principle of “structure determination nature”, so it is often applied in practice, but small changes in active compound structure will affect its activity [3]. Although the RBVS (receptor-based virtual screening) method avoids these problems and has broad development prospects [4], it also has its own problems. For example, in commonly used molecular docking technology, the biggest problem is the accuracy and applicability of scoring functions. In order to accelerate screening, it often tends to consider using simple scoring functions and such scoring functions often do not pay enough attention to weak molecular interactions. In addition, limited by the three-dimensional structure of the receptor, when the target has a restricted experimental crystal structure, its application is limited and candidate set, inactive compound and active compound data imbalance make the candidate set mixed with wrong docking conformation directly affecting the effect of virtual screening. Therefore, considering virtual screening as a problem of imbalanced data classification is of obvious practical guiding significance. In this paper, we know from medical image processing that the imbalanced samples are processed by the oversampling or undersampling methods of images. Due to the effectiveness of the support vector machine method in medical image classification, the improved SMOTE method is introduced in the virtual screening problem and the SVM is improved by using the integrated learning idea introduced in the classification stage, and a set of preprocessing and classification methods for imbalanced data is proposed. The method has reference significance to some problems faced by virtual screening and the method has considerable reference value in medical image processing. In the first three parts of the second section, we introduce the development of the SMOTE method and the specific process of the heuristic oversampling method based on k-means and SMOTE. The fourth part introduces the data processed by the previous method. At the classification level, support vector machines, particle swarm optimization algorithms and ensemble learning methods are used for the experiments. The third section is the analysis of the experimental results, and the last section is the summary of our experiments, defects and future prospects.

2. Heuristic Oversampling Methods Based on K-Means and SMOTE

In the actual virtual screening, the number of active docking compounds tends to be much smaller than the number of inactive compounds, so the data imbalance has not been well resolved. In this paper, a k-means and SMOTE-based heuristic oversampling method is used to solve the problem of imbalanced data generated in the virtual screen-ing [5], and the k-means and SMOTE heuristics are used to oversample a small number of normal samples, so as to reduce the data imbalance ratio.

2.1. Processing Method of Imbalanced Data

Imbalanced data are mainly reflected in the imbalance of positive and counterexample data, which will lead to the counterexample tilt of most class data in the classification process.

To solve the problem of these imbalanced data, improvements are usually made from data preprocessing as well as from classification algorithms. First, in the data pre-processing stage, there are usually two sampling methods. One is the downsampling method, which reduces the imbalance ratio of the imbalanced dataset by reducing the number of counterexample sample data with large data magnitude under the condition of ensuring that the small number of positive example sample data remains unchanged [6]. The other is the upsampling method, which makes up for the gap between the positive sample data and the negative sample data by increasing the number of positive sample data [7]. In the classification algorithm, by improving the classification algorithm, the algorithm adapts more to the classification problem of the imbalanced dataset, thus improving the correctness of the imbalanced data classification based on the algorithm; for example, integrated learning can combine multiple classifiers to form a strong classifier, which can improve the classification performance of the classifier and overcome the problem of imbalanced data.

2.2. SMOTE Algorithm Introduction

Upsampling technology to expand a few classes of samples, and there are often two ways to replicate existing observations, is the other way to generate artificial data that may work unknowingly, randomly selecting replication or using basic data samples, or knowingly directing it to the areas considered most effective in upsampling. In informed cases, clustering methods are sometimes employed to identify sample regions for gene screening. Sample instances of a few classes are randomly oversampled to the desired distribution. The method has been increasingly adopted due to its simplicity as well as ease of implementation, but the method is a simple data replication, which may make the classifier trained on randomly upsampled data at the risk of overfitting.

Chawla et al. proposed the SMOTE algorithm in 2002 in order to avoid the overfitting that arises from the on-the-fly upsampling [8]. This proposed algorithm does not simply replicate the original data, but will generate artificial samples, as shown in Figure 1. The SMOTE algorithm is implemented by randomly selecting one of the few classes of sample data, and then selecting several of its neighboring sample data by linear interpolation. The SMOTE algorithm generates artificial samples in three steps, first selecting a few random samples $\vec{\mathrm{A}}$. Select instance $\vec{\mathrm{B}}$ from its K-nearest minority class neighbors. Finally, a new sample is created by randomly interpolating two samples $\vec{\mathrm{x}},\vec{\mathrm{x}} = \vec{\mathrm{A}} + \omega \times \left(\vec{\mathrm{B}} \times \vec{\mathrm{A}}\right)$, where ω is the random weight in [0, 1].

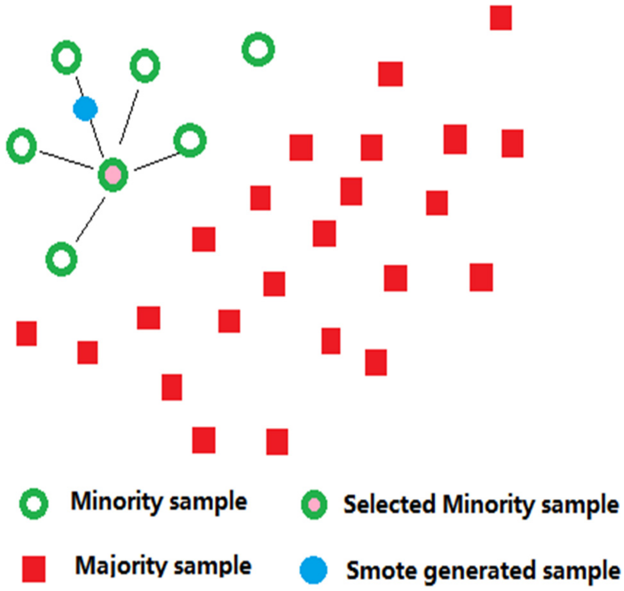


Figure 1. A small number of samples randomly selected by SMOTE linear interpolation and one of its k = 5 nearest neighbors.

However, the algorithm still has a few problems in handling imbalanced data and noise. In the presence of noise, SMOTE may produce some samples in most regions, while most non-noisy samples are produced in a few dense regions, which creates an imbalance within the class. Despite such drawbacks, SMOTE is widely adopted by researchers because of its simplicity and the extra value of random upsampling.

Currently, many researchers have proposed many improvements to the technique as well as ways to extend it, aiming to eliminate its shortcomings. These modifications focused on decision boundaries, with Borderline-SMOTE in the method category that emphasizes class regions, dividing a few class samples into noise, hazard and safety points, first removing noise points and only using hazard points for sample synthesis. Borderline-SMOTE not only uses a few samples but also a majority during generation, allowing the boundary between classes to be strengthened [9].

Cluster-SMOTE is an alternative approach to highlight the technical categories of certain class regions, which clusters a few classes using K-means before applying SMOTE to the discovered classes [10]. The stated goal of the method is to enhance the minority class regions by creating artificial samples in the initial minority class regions. The method does not specify the number of samples to be generated in each post-clustering population, nor does it determine the final optimal number of clusters.

Nekooeimehr and Lai-Yuen proposed an adaptive semi-unsupervised weighted oversampling (A-SUWO), which belongs to the same class. A-SUWO is a technique based on hierarchical clustering that uses clustering to improve the quality of oversampling and its purpose is to oversample hard-to-learn instances near the decision boundary [11].

The self-organizing mapping oversampling (SOMO) algorithm uses self-organizing mapping to transform input data into a two-dimensional space in which to identify safe and efficient regions for the data [12]. The SMOTE technique is then used within the clusters found in the low-dimensional space and between adjacent clusters to correct for the intra- and inter-class imbalances.

The entire input space was clustered using the K-means by Santos et al. Less-represented clusters were selected using SMOTE for oversampling [13]. The algorithm differs from most oversampling methods because the application of SMOTE is independent of category labels. The class labels of the generated sample come from the most recent copy of the two parent classes. The algorithm therefore targets dataset imbalances, not between classes or within classes, and cannot be used to resolve class disequilibrium.

2.3. Heuristic Upsampling Methods Based on K-Means and SMOTE

In this paper, the combination of the simple popular k-means clustering algorithm with SMOTE upsampling is used to balance the skewed dataset, which manages to avoid noise generation by only upsampling in safe regions. At the same time, focus on the imbalance between classes and within classes, and solve the problem of small separation by expanding the area of a few classes.

The sample classification of this method is based on cluster density. More samples are generated in sparse minority class areas than in dense majority class areas, so as to eliminate the imbalance within the class. In addition, the highlight of the method used is that it clusters regardless of the category label, so as to detect the upsampling security region and finally offset the overfitting by generating new samples rather than copying them.

The k-means SMOTE algorithm used in this paper is composed of three parts: clustering, filtering and upsampling. First, in the clustering process, k-means clustering is used to cluster the input data space into k groups, followed by a filtering step to select clusters for upsampling and retain a small number of clusters with a high proportion of samples. Then, it assigns the number of synthetic samples to be generated and assigns a larger number of samples to a few clusters with sparse sample distribution. Up to the upsampling step, SMOTE is applied in each of the selected clusters to achieve the target ratio of minority and majority instances. This is shown in Figure 2.

First of all, in this algorithm, k-means is one of the very popular iterative algorithms that essentially works by iteratively repeating two structures [14]. The first step is to assign each observation to the nearest one of the k clustering primes, and the second step is to update the positions of the primes so that they are located in the center between the observations assigned to them. The algorithm converges when the observations are no longer reassigned, ensuring convergence to the typical local optimum within a finite number of iterations. For large datasets, where k-means converges slowly, an efficient implementation can be used for the clustering step of k-means SMOTE. All hyperparameters of k-means are also hyperparameters of the proposed algorithm, the most significant being k, the number of clusters. A suitable k is important for the effectiveness of k-means SMOTE. First of all, for k-means, the k value must be given, that is, the number of clusters must be specified. It is often difficult to pre-estimate and give the k value at the beginning. Secondly, the k center points at the beginning are randomly given. Recalculations are performed in subsequent iterations until convergence. However, according to the steps of the algorithm, it is not difficult to see that the final result often depends, to a large extent, on the positions of the k center points at the beginning, which also means that the results have great randomness and the algorithm needs to constantly adjust the object and continuously calculate the adjusted new cluster center points, so when the amount of data is very large, the algorithm overhead will be relatively large.

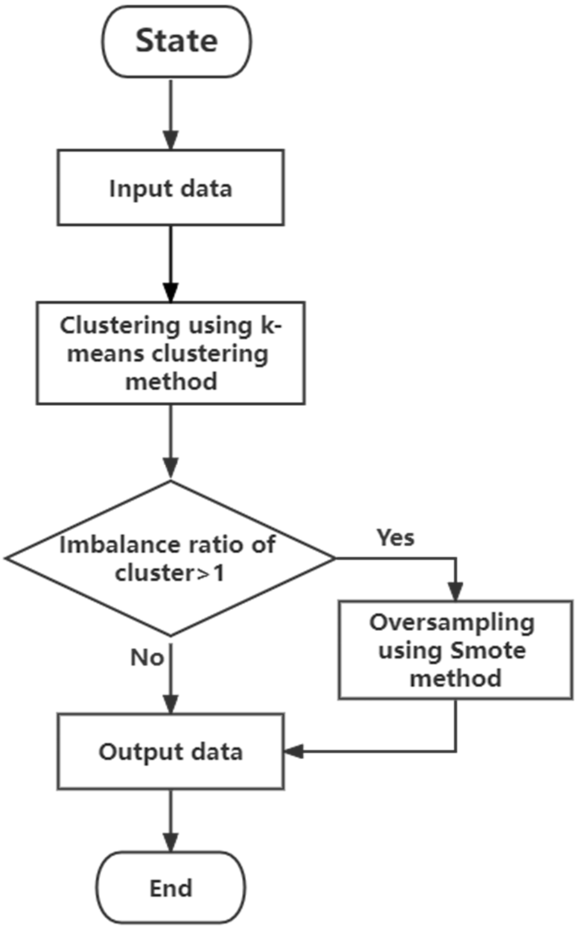

State
Input data
Clustering using k-means clustering method
Imbalance ratio of cluster>1
Yes
No
Oversampling using Smote method
Output data
End

Figure 2. K-means SMOTE upsamples the security area and eliminates the imbalance within the class.

When determining the number of clusters in the k-means method, we compared the popular elbow method and the silhouette coefficient method, because the optimal k value determined by the silhouette coefficient method is not necessarily the best; therefore, it is sometimes necessary to use the sum of squared errors (SSE) to assist the selection, which is slightly cumbersome compared to the elbow method. Thus, the elbow method is finally selected. The core idea of the elbow method is that as the number of clusters k increases, the sample division will become more refined, the degree of aggregation of each cluster will gradually increase and thus the SSE will naturally gradually become smaller. Moreover, when k is less than the real number of clusters, since the increase in k will greatly increase the degree of aggregation of each cluster, the decline in SSE will be very large. In addition, when k reaches the real number of clusters, the degree of aggregation returns will decrease rapidly. Thus, the decline in SSE will decrease sharply, becoming flat as the value of k continues to increase. Therefore, the relationship between SSE and k is in the shape of an elbow, and this elbow the k value is corresponding to is the true number of clusters of the data.

The filtering operation after clustering selects the clusters to be upsampled and determines the number of artificial samples to be generated in each cluster. This is carried out mainly to upsample clusters in which the minority is dominant, because using SMOTE in the minority region is less likely to generate noise risk. The filtering operation will assign more generated samples to the sparse minority class clusters, in which the choice of upsampling clusters is based on the ratio of minority to majority instances, usually at least 50% of the ratio should be selected for upsampling, which is a hyperparameter of k-means SMOTE and can be adjusted by adjusting the imbalance ratio threshold (irt).

Because the distribution of the generated samples needs to be determined, the clusters after the filtering operation are assigned weights between [0, 1]. High sampling weights correspond to low-density few-class samples and produce more generated samples. To achieve this idea, the sampling weights depend on the density of individual clusters compared to all selected average densities. When measuring the density of clusters, considering only the distance between a few instances, the calculation of sampling weights can be represented by five suboperations:

1. The Euclidean distance matrix is calculated for each filtered cluster F, without considering most samples;
2. The average distance within each cluster is calculated by summing all non-diagonal elements of the distance matrix and dividing by the number of non-diagonal elements;
3. To obtain a measure of density, divide each cluster's number of minority instances by its average minority distance raised to the power of the number of features m:

$$\mathrm{density(f)} = \frac{\mathrm{minority\ count(f)}}{\mathrm{average\ minority\ distance(f)}^{\mathrm{m}}};$$

4. The sparsity metric is obtained by inverting the density metric: sparsity $(\mathrm{f}) = \frac{1}{\mathrm{density(f)}}$;
5. The sparsity factor of the clusters is divided by the sum of the sparsity factors of all clusters to define the sampling weights of each cluster.

Therefore, it can be seen that the sum of all the sampling weights is 1. Thus, the number of samples to be generated in each particular cluster is determined by multiplying the sampling weights of the clusters by the total number of samples to be generated in total.

In the upsampling process of the k-means SMOTE algorithm, it is used to upsample each filtered cluster, and for each cluster sampled, all points of the cluster are given and the number of samples generated is $\|\text{sampling weight (f)} \times \text{n}\|$, n represents the total number of generated samples.

For each composite sample to be generated, SMOTE selects a random minority sample in the cluster $\overrightarrow{\mathrm{a}}$ and finds a few random adjacent instances of point $\overrightarrow{\mathrm{b}}$. The new samples $\overrightarrow{\mathrm{x}}$ were also determined by randomly interpolating $\overrightarrow{\mathrm{a}}$ and $\overrightarrow{\mathrm{b}}$. Geometrically, the new point $\overrightarrow{\mathrm{x}}$ will be placed somewhere on the line from $\overrightarrow{\mathrm{a}}$ to $\overrightarrow{\mathrm{b}}$. The procedure is repeated until the number of samples to be generated is reached.

The method used in this paper is different from other related techniques, the method is to cluster the whole dataset without considering the class labels, this can find overlapping class regions and help prevent oversampling of unsafe regions. In addition, the k-means method can find clusters with different densities, but they are usually of the same size, so the way to combat intra-class imbalance is to distribute the samples according to the cluster density. This method finally uses SMOTE to prevent the problem of overfitting from arising. The Algorithm 1 used is as follows:

Algorithm 1: K-means SMOTE algorithm used in this paper.

```
Input: Sample matrix X; target vector y; number of samples to generate n; k-means the number of
    clusters to find k; imbalance ratio threshold ir-t; number of nearest neighbors considered by
    SMOTE knn; index used to calculate density d.
Output: Generated samples.
Step 1: The input sample set is first clustered and then clusters with more minority class instances
    than majority class instances are filtered.
Step 2: For each filtered cluster, the sampling weight is calculated according to its minority
    population density.
Step 3: SMOTE is used to upsample each filtered cluster and the sampling weight is used to
    calculate the number of samples to be generated.
```

2.4. Classification Algorithms Used in Virtual Screening of Drug Proteins

In Section 2.3, a heuristic upsampling method based on k-means SMOTE has been used to reduce the imbalanced data, but the SVM classification algorithm is adopted. To further enhance the accuracy of virtual screening, the optimization algorithm and integrated learning ideas are introduced, and an integrated learning model is constructed to improve the robustness of the imbalanced data classification.

2.4.1. Support Vector Machine

Generally speaking, SVM is a typical two-class classification model [15]. When the sample is linearly indistinguishable in the original space, the sample can be mapped from the original space to a higher dimensional feature space, making the sample linearly divisible in this feature space. Additionally, after introducing such a mapping, the required solution of the pairwise problem does not need to solve the true mapping function, but only needs to know its kernel function. Suppose a given set S with N samples, $S = \{(x_i, y_i)|i = 1, 2, \ldots, N\}$, the expressions and objective functions of the classification hyperplane are shown in the following two equations:

$$f(x) = \omega \cdot x + b \tag{1}$$

$$\min_{w,b} \frac{1}{2}||w||^2 + C\sum_{i=1}^{n} \xi_i \tag{2}$$

$$\text{s.t. } y_i(\omega \cdot x + b) \geq 1 - \xi_i$$

$$\xi_i \geq 0, i = 1, 2, \ldots, N$$

where w is the normal vector of the hyperplane; b is the translation distance of the hyperplane; ξ_i is the non-negative relaxation vector used to improve the generalization ability of the model; and C is the penalized factor used to trade-off the relationship between the classification loss and the maximum interval. The kernel function selects the RBF. The kernel function is shown in the following equation:

$$K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \tag{3}$$

σ is the width parameter of the function which controls the radial scope of the function. The kernel function used in the SVM classifier is a Gaussian kernel function, so C and the Gaussian kernel radius gamma affect the classification effect of the classifier; thus, it is essential to continuously optimize its parameters during the training stage. The commonly used algorithms include genetic algorithm and gravity search algorithm. PSO has a memory, with all particles of good solution knowledge are saved, while the genetic algorithm has no memory, previous knowledge is destroyed as the population changes. In the genetic algorithm, chromosomes share information with each other, so the movement of the entire population is relatively even towards the optimal area. The particles in the PSO only share information through the current search for the optimal point; thus, to a large extent, this is a single-item information sharing mechanism, and the entire search and update process follows the process of the current optimal solution. In most cases, all particles may converge to the optimal solution faster than the evolutionary individuals in the genetic algorithm. At the same time, the PSO algorithm has a simpler principle, fewer parameters and easier implementation than the gravitational search algorithm. In this paper, the optimization of the parameters of the SVM classifier is carried out using the particle swarm optimization (PSO) algorithm.

2.4.2. Particle Swarm Optimization Algorithm

PSO algorithm is an intelligent algorithm which finds the most valuable particles in the population according to the cooperation and competition mechanism in the evolution process of particles in the population. During parameter optimization, the PSO algorithm first initializes the particle swarm to determine the population size and iteration times, and then sets the speed and position of particles [16]. The following two equations are used to iterate the particle swarm optimization to find out individual extremes and the overall extremes, and then enter the next iteration to determine the speed and position of the next generation of particles. Through the iterative process, when a certain number of iterations or the set convergence accuracy is reached, the global optimal solution in the search space can be obtained.

$$v_{ij}^{k+1} = w(k)v_{ij}^{k} + c_1r_1\left(p_{ij} - x_{ij}^{k}\right) + c_2r_2\left(p_{gj} - x_{ij}^{k}\right) \tag{4}$$

$$x_{ij}^{k+1} = x_{ij}^{k} + v_{ij}^{k+1} \tag{5}$$

where v_{ij}^{k} is the velocity of the j-th variable of the ith particle in the k-th iteration; x_{ij}^{k} is the position of the j-th variable of the particle in the k-th iteration; c_1 and c_2 are the learning factors; $w(k)$ is inertia weight; r_1 and r_2 are random numbers evenly distributed in [0, 1]; and p_{ij} and p_{gj} are the i-th individual and the j-th variable in the global optimal particle, respectively.

2.4.3. AdaBoost Algorithm

In order to better solve the classification problem of imbalanced data in this paper, the idea of integrated learning is introduced, which can improve the classification ability of traditional machine learning algorithms and reduce the generalization error. The most commonly used methods are mainly boosting and bagging [17,18]; therefore, in this paper, we study the AdaBoost algorithm in the boosting method.

AdaBoost is an iterative promotion algorithm. The core idea is for the same training set, repeatedly using the most common learning algorithm (such as decision tree, naive Bayes, etc.) training, to obtain different classifiers (weak classifier), and then the weak classifiers by weighted voting constitute a stronger final classifier (strong classifier). For a binary classification problem, set the input m samples: $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$, where the training sample is x_i, $y_i \in \{0, 1\}$, representing negative and positive samples, respectively. The AdaBoost integration learning algorithm steps are as follows [19]:

1. The initial weight distribution of m samples is

$$D_k(i) = \frac{1}{m} \tag{6}$$

2. Weak predictor prediction. When training the k-th weak predictor, the error sum of the prediction function g(k) and ξ_i is obtained:

$$\xi_k = \sum_i D_k(i) \cdot i = 1, 2, 3 \ldots, m \tag{7}$$

where i shall meet $\left|\frac{g_{k(x_i)-y_i}}{y_i}\right| > \varphi$; $g_{k(x_i)}$ is the prediction output; y_i is the desired output; and φ is a manually set threshold.

3. Reference error and ξ_k. The weights a_k were calculated using the following equation:

$$a_k = \ln\left(\frac{1}{{\xi_k}^2}\right) \tag{8}$$

4. Regarding the weight update, refer to a_k updates the weight distribution of training samples in the next iteration. The update equation is as follows:

$$D_{k+1}(i) = \frac{D_k(i)}{b}\begin{cases} {\xi_k}^2 \; \frac{g_{k(x_i)-y_i}}{y_i} \le \varphi \\ 1 \; \frac{g_{k(x_i)-y_i}}{y_i} > \varphi \end{cases} \tag{9}$$

where b is the normalization factor.

5. After T iterations, T groups of weak prediction sequences $f(g_k,\ a_k)$ are obtained and a_k is the weight of the prediction sequence. Normalization using the first equation. Finally, the strong predictor function is obtained, as shown in the second equation below:

$$a_k = \frac{a_k}{\sum_{T=1}^{k} a_k} \tag{10}$$

$$y(x_i) = \sum_{T=1}^{k} a_k g_{k(x_i)} \tag{11}$$

where $g_{k(x_i)}$ is the predicted value of the k-th weak predictor function.

2.4.4. PSO-SVM-AdaBoost Classification Model

Through the above theory, in order to better solve the imbalance classification problem of imbalanced data, make full use of the global optimization ability of the PSO algorithm to optimize the penalty parameter C and Gaussian kernel radius gamma of SVM. The PSO-SVM classifier with optimized parameters is used as a weak classifier, and the AdaBoost algorithm is used to train the selected PSO-SVM base classifier for the same group of samples [20]. Finally, it is weighted and combined into a strong classifier to enhance the prediction accuracy of the model.

3. Experimental Verification and Analysis

3.1. Evaluation Criterion

In the current mainstream methods for evaluating the effect of virtual screening and machine learning, the enrichment factor (EF) is the most common standard for evaluating the effect of virtual screening. EF is often used to evaluate the early recognition attributes of virtual screening methods. The calculation formula of enrichment factor is shown in the following equation:

$$EF = \frac{Hits_s}{N_s} / \frac{Hits_t}{N_t} \tag{12}$$

where $Hits_s$ is the number of active compounds in the sample and Hit_t is the number of all compounds in the test set; N_s is the number of all compounds sampled; and N_t is the number of all compounds.

In the actual drug development and discovery process, a large number of compounds will be involved, but only a small number of compounds will be screened out by a computer in the end. For this reason, before conducting experiments, a 10% EF value is selected to reduce the bias of the preliminary evaluation. The false positive rate (TPR) is the standardized horizontal coordinate, the true positive rate (FPR) is the vertical coordinate and the AUC value is the area of the lower part of the ROC curve. The formula for calculating the AUC value is shown below.

$$\mathrm{AUC} = \int_0^1 \frac{\mathrm{TP}}{\mathrm{P}} \mathrm{d}\frac{\mathrm{FP}}{\mathrm{N}} = \frac{1}{\mathrm{P}\cdot\mathrm{N}} \int_0^{\mathrm{N}} \mathrm{TPdFP} \tag{13}$$

The AUC value is between 0.5 and 1. If it is a perfect model, the AUC value is 1 and the ROC curve and AUC value are not affected by the distribution of data imbalance.

3.2. Experimental Data Acquisition

In this paper, HIV-1 protease and Src kinase were selected as target proteins in virtual screening [21] and small molecular compounds were selected from the PubChem library [22]. HIV-1 protease is the most important target protein in the treatment of AIDS. HIV-1 protease is a specific aspartyl protease with C2D symmetry axis structure encoded by HIV gene, so it has two identical peptide chains. Therefore, as long as the fusion of these two identical polypeptide chains can be inhibited or the amino acids existing in the peptide chain can be hydrolyzed, the purpose of inhibiting enzyme activity can be achieved. The crystal structure of HIV-1 protease selected in this paper is directly obtained from the PDB database [23]. The PID is 5COP and its resolution is 2.00 Å. The structural diagram is shown in Figure 3. Src kinase is an important target for anti-tumor drugs. It has a high level of expression in lung cancer, breast cancer, rectal cancer and pancreatic cancer. SRC is a kind of non-receptor tyrosine kinase family with molecular weight of 60 kDa. The structure of Src kinase family is similar. SRC can participate in signal transduction pathways in vivo through various receptors, so as to promote the occurrence of a series of chemical reactions, and this kinase is associated with the occurrence of a variety of cancers in the human body. Researchers found that the SRC pathway was activated in more than half of all cancers and therefore could be used to treat cancer by keeping SRC active. The Src kinase crystal selected for this paper was extracted from the PDB database. Its PID is 2H8H and its resolution is 2.20 Å. Figure 4 shows its structure.

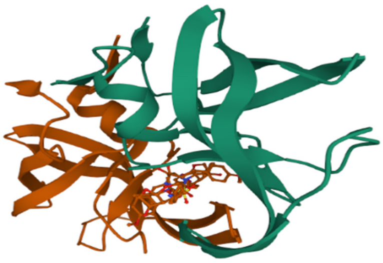


Figure 3. The PID is a crystal of the HIV-1 protease complex of the 5COP.

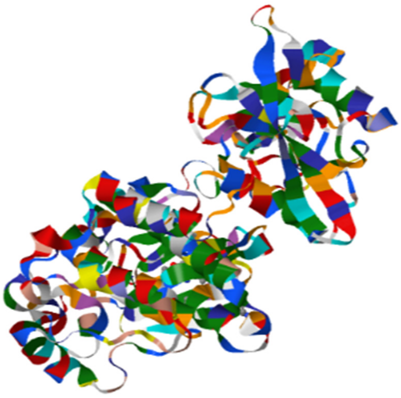


Figure 4. PID is a Src kinase complex-binding crystal of 2H8H.

Small-molecule ligands perform molecular docking with the above two target proteins using the SP patterns in the GLIDE docking in the Maestro software. The target protein was first dehydrogenation using protein preparation and predictions were made using https://www.playmolecule.com/ (accessed on 15 December 2021) when predicting the binding pockets of the target proteins [24]. Preparation for small-molecule ligands was carried out using LigPrep, paired boxes were then generated in receptor grid generation using the binding pocket positions obtained above [25]. Finally, the SP mode of GLIDE was selected for molecular docking in the ligand docking.

This paper proposes two specific technologies, namely, heuristic upsampling method based on k-means and SMOTE and classification model based on PSO AdaBoost SVM. Both techniques used HIV-1 protease and Src kinase as experimental data.

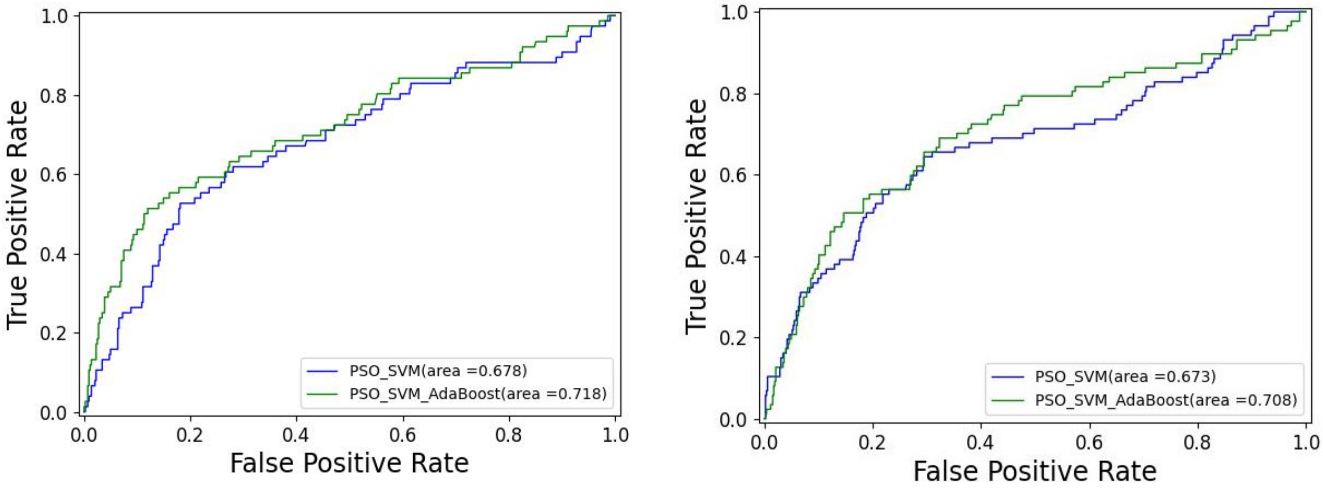
In this paper, the PSO-SVM-AdaBoost-integrated learning classification algorithm is proposed, which is compared with the SVM classification method only using particle swarm optimization algorithm [26]. At the same time, predictive contrasts were also performed on the postsampled dataset. Through cross validation, it is concluded that the number of iterations of the particle swarm optimization algorithm is 40 and the population size is 100.

This paper divides the experiment into four cases: the experimental comparison between PSO-SVM and PSO-SVM-AdaBoost classification model before sampling and the experimental comparison between PSO-SVM and PSO-SVM-AdaBoost classification model after using the upsampling method proposed in Section 2. The experimental comparison results of the two classification methods when the two datasets of HIV-1 protease and Src kinase are not sampled are shown in Table 1 below.

Table 1. Index comparison of two classification methods.

Protein Name	Algorithm	10% EF	AUC
HIV-1 protease	PSO-SVM	5.78	0.678
HIV-1 protease	PSO-SVM-AdaBoost	6.03	0.718
Src kinase	PSO-SVM	5.75	0.673
Src kinase	PSO-SVM-AdaBoost	5.98	0.708

The ROC curve is shown in Figure 5.



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

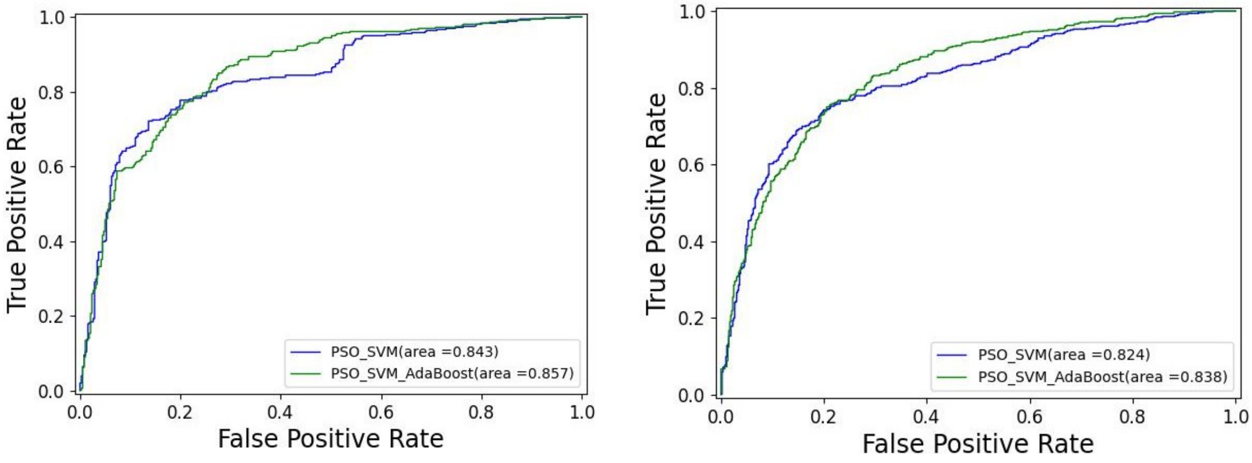
Figure 5. Comparison of ROC curves of the two datasets: (**a**) ROC curves of the first two classification methods without sampling of HIV-1 protease; (**b**) ROC curves of the first two classification methods without sampling of SRC protease.

The heuristic upsampling method based on k-means and SMOTE proposed in this paper increases the amount of data, reduces the imbalance ratio, observes the changes in virtual screening results before and after sampling and further compares the experimental performance of the two algorithms. The experimental comparison results of the two classification methods are shown in Table 2.

Table 2. After sampling, the indexes of the two classification methods are compared.

Protein Name	Algorithm	10% EF	AUC
HIV-1 protease	PSO-SVM	6.64	0.843
HIV-1 protease	PSO-SVM-AdaBoost	6.73	0.857
Src kinase	PSO-SVM	6.44	0.824
Src kinase	PSO-SVM-AdaBoost	6.51	0.838

The ROC curve is shown in Figure 6.



(a) (b)

Figure 6. Comparison of ROC curves of the two datasets: (**a**) The ROC curves of the two methods after sampling HIV-1 protease; (**b**) The ROC curves of the two methods after sampling Src kinase.

In order to prove the effectiveness of the method, two groups of representative data are used for the model comparison test. The PSO-SVM and PSO-SVM AdaBoost models before and after the resampling of HIV-1 protease and Src kinase data are compared.

The data after the virtual screening of HIV-1 protein and SRC kinase were compared to the classification effect before and after the heuristic oversampling method based on k-means and SMOTE, as well as the comparison of the classification effect before and after using ensemble learning, to express the used methods for resampling and the effectiveness of introducing ensemble learning methods.

Before resampling, in the virtual screening for HIV-1 protease, the penalty parameter C and Gaussian kernel radius gamma of SVM classifier are optimized by the particle swarm optimization algorithm and the classification effect of virtual screening is still low. The 10% EF is 5.78 and the AUC value is 0.678. After ensemble learning, the accuracy of PSO-SVM AdaBoost classifier is improved. The 10% EF is 6.03 and the AUC value is increased to 0.718. In the virtual screening experiment for Src kinase, through the PSO-SVM AdaBoost algorithm, the 10% EF was increased from 5.75 to 5.98 in PSO-SVM, and the AUC value was also increased from 0.673 to 0.708. After sampling, the SVM classification effect is significantly improved and the virtual screening effect is also strengthened. The enrichment factor of HIV-1 protease virtual screening using the PSO-SVM classification model reached 6.64 and AUC value was 0.843. PSO-SVM AdaBoost algorithm also improved the effect of virtual screening and the enrichment factor increased to 6.73. At the same time, the AUC value is increased to 0.857. In the virtual screening experiment for Src kinase, the PSO-SVM classification enrichment factor after sampling is 6.44 and the AUC value is 0.824. After the PSO-SVM AdaBoost algorithm, the enrichment factor is increased to 6.51 and the AUC value is also increased to 0.838. Through the analysis of the virtual screening effect of the two target proteins, the screening effect after sampling is improved. At the same time, the PSO-SVM AdaBoost classification model of integrated learning is also used to improve the virtual screening effect. It is concluded that the heuristic upsampling method based on k-means and SMOTE improves the accuracy of virtual screening by resampling the training set. Secondly, AdaBoost ensemble learning is used to improve the accuracy and stability of single PSO-SVM classification.

4. Conclusions

This paper studies the virtual screening based on molecular docking, analyzes the problems faced by the current virtual screening, improves it through the combination of a machine learning method and virtual screening technology and considers the virtual screening as an imbalanced data problem. The research of this paper is mainly reflected in the following two aspects.

Firstly, for the imbalanced data generated in virtual screening, the heuristic upsampling method based on k-means and SMOTE is used for data preprocessing [27]. More samples are generated in sparse minority class areas than in dense majority class areas, which is no longer a simple random copy of samples, so as to reduce the imbalance ratio of imbalanced data and avoid overfitting to a certain extent [28].

Secondly, the particle swarm optimization (PSO) algorithm is used to improve the parameters of the optimized SVM classifier, and combined with the idea of integrated learning, it further improves the classification effect of the classifier. HIV-1 and Src kinase are selected as target proteins for virtual screening, veryfing the effectiveness of the drug protein virtual screening method based on imbalanced data classification and prediction model proposed in this paper.

In protein virtual screening, it is necessary to prepare data carefully and select parameters carefully. At present, our research still needs to pay attention to the processing of large-scale massive data and high attribute dimension. In the future, we need to conduct further research on the processing method of imbalanced data, not only limited to the improvement of the SMOTE method, to improve the protein virtual screening method.

Author Contributions: Project administration, L.Y.; Writing—original draft, C.M.; Writing—review and editing, X.D.; Supervision, H.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation, China (No. 62172123) and the special projects for the central government to guide the development of local science and technology, China (No. ZY20B11).

Conflicts of Interest: The authors declare no potential conflict of interest with respect to the research, authorship and/or publication of this article.

Abbreviations:

SVM	Support vector machine
PSO	Particle swarm optimization
AdaBoost	Adaptive boosting
PDB	Protein data bank
RBVS	Receptor-based virtual screening
LBVS	Ligand-based virtual screening
SSE	Sum of squared errors
IDS	Imbalanced dataset

References

1. Alibeigi, M.; Hashemi, S.; Hamzeh, A. DBFS: An effective Density Based Feature Selection scheme for small sample size and high dimensional imbalanced data sets. *Data Knowl. Eng.* **2012**, *81–82*, 67–103. [CrossRef]
2. Johnson, D.K.; Karanicolas, J. Ultra-High-Throughput Structure-Based Virtual Screening for Small-Molecule Inhibitors of Protein-Protein Interactions. *J. Chem. Inf. Model.* **2016**, *56*, 399. [CrossRef] [PubMed]
3. Roy, A.; Srinivasan, B.; Skolnick, J. PoLi: A Virtual Screening Pipeline Based on Template Pocket and Ligand Similarity. *J. Chem. Inf. Model.* **2015**, *55*, 1757–1770. [CrossRef] [PubMed]
4. Dai, W.; Guo, D. A Ligand-Based Virtual Screening Method Using Direct Quantification of Generalization Ability. *Molecules* **2019**, *24*, 2414. [CrossRef] [PubMed]
5. Georgios, D.; Fernando, B.; Felix, L. Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE. *Inf. Ences* **2018**, *465*, 1–20.
6. Zheng, X.; Tang, Y.Y.; Zhou, J.; Wang, P. Improving Unbalanced Downsampling via Maximum Spanning Trees for Graph Signals. In Proceedings of the 2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, Hungary, 9–12 October 2016.
7. Beermann, M.; Ohm, J.R. Non-Linear Up-Sampling for Image Coding in a Spatial Pyramid. In Proceedings of the SPIE—The International Society for Optical Engineering, San Jose, CA, USA, 29 January 2007; p. 65082w.
8. Chawla, N.V.; Bowyer, K.W.; Hall, L.O.; Kegelmeyer, W.P. SMOTE: Synthetic Minority Over-sampling Technique. *J. Artif. Intell. Res.* **2011**, *2002*, 16. [CrossRef]
9. Han, H.; Wang, W.Y.; Mao, B.H. Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning. In Proceedings of the 2005 International Conference on Advances in Intelligent Computing, Hefei, China, 23–26 August 2005.
10. Agrawal, A.; Viktor, H.L.; Paquet, E. SCUT: Multi-Class Imbalanced Data Classification using SMOTE and Cluster-based Undersampling. In Proceedings of the International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management, Lisbon, Portugal, 12–14 November 2015.
11. Iman, N.; Lai-Yuen, S.K. Adaptive semi-unsupervised weighted oversampling (A-SUWO) for Imbalanced Datasets. *Expert Syst. Appl.* **2016**, *46*, 405–416.
12. Bacao, F.; Douzaz, G. Self-Organizing Map Oversampling (SOMO) for imbalanced data set learning. *Expert Syst. Appl.* **2017**, *82*, 40–52.
13. Santos, M.S.; Henriques Abreu, P.; García-Laencina, P.J.; Simão, A.; Carvalho, A. A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients. *J. Biomed. Inform.* **2015**, *58*, 49–59. [CrossRef] [PubMed]
14. Basel, S.; Gopakumar, K.U.; Prabhakara, R.R. Classification of countries based on development indices by using K-means and grey relational analysis. *GeoJournal* **2021**, in press. [CrossRef]
15. Pang, S.; Kasabov, N. Inductive vs Transductive Inference, Global vs Local Models: SVM, TSVM, and SVMT for Gene Expression Classification Problems. In Proceedings of the 2004 IEEE International Joint Conference on Neural Networks, Budapest, Hungary, 25–29 July 2004; IEEE: Piscataway, NJ, USA, 2004; Volume 2.
16. Zhang, Z.; Guo, H. Research on Fault Diagnosis of Diesel Engine Based on PSO-SVM. In *Proceedings of the 6th International Asia Conference on Industrial Engineering and Management Innovation*; Atlantis Press: Paris, France, 2016.
17. Freund, Y.; Schapire, R.E. A decision-theoretic generalization of on-line learning and an application to boosting. *J. Comput. Syst. Sci.* **1997**, *55*, 119–139. [CrossRef]
18. Louppe, G.; Geurts, P. Ensembles on Random Patches. In Proceedings of the Machine Learning and Knowledge Discovery in Databases, Bristol, UK, 24–28 September 2012; pp. 346–361.
19. Nakamura, M.; Hiroki, N.; Kuniaki, U. Improvement of boosting algorithm by modifying the weighting rule. *Ann. Math. Artif. Intell.* **2004**, *41*, 95–109. [CrossRef]
20. Hao, G.; Bin, J. Fault Diagnosis of Wind Turbines' Bearing Based on PSO-AdaBoostSVM. In Proceedings of the 2018 3rd International Conference on Electrical, Automation and Mechanical Engineering (EAME 2018), Xi'an, China, 26–27 July 2018; Atlantis Press: Paris, France, 2018.
21. Peng, L.; Yin, L.; Zhao, B.; Sun, Y. Virtual Screening of Drug Proteins Based on Imbalance Data Mining. *Math. Probl. Eng.* **2021**, *2021*, 585990.
22. Kim, S.; Thiessen, P.A.; Bolton, E.E.; Chen, J.; Fu, G.; Gindulyte, A.; Han, L.; He, J.; He, S.; Shoemaker, B.A.; et al. PubChem substance and compound databases. *Nucleic Acids Res.* **2016**, *44*, D1202–D1213. [CrossRef] [PubMed]
23. Chakraborty, S.; Phu, M.; de Morais, T.P.; Nascimento, R.; Goulart, L.R.; Rao, B.J.; Asgeirsson, B.; Dandekar, A.M. The PDB database is a rich source of alpha-helical anti-microbial peptides to combat disease causing pathogens. *F1000Research* **2014**, *3*, 295. [CrossRef] [PubMed]
24. Soufan, O.; Ba-alawi, W.; Magana-Mora, A.; Essack, M.; Bajic, V.B. DPubChem: A web tool for QSAR modeling and high-throughput virtual screening. *Sci. Rep.* **2018**, *8*, 9110. [CrossRef] [PubMed]
25. Hidaka, T.; Imamura, K.; Hioki, T.; Takagi, T.; Giga, Y.; Giga, M.-H.; Nishimura, Y.; Kawahara, Y.; Hayashi, S.; Niki, T.; et al. Prediction of Compound Bioactivities Using Heat-Diffusion Equation. *Patterns* **2020**, *1*, 100140. [CrossRef] [PubMed]
26. Hussin, S.K.; Abdelmageid, S.M.; Alkhalil, A.; Omar, Y.M.; Marie, M.I.; Ramadan, R.A. Handling Imbalance Classification Virtual Screening Big Data Using Machine Learning Algorithms. *Complexity* **2021**, *2021*, 15. [CrossRef]

27. Revathi, M.; Ramyachitra, D. A Modified Borderline Smote with Noise Reduction in Imbalanced Datasets. *Wirel. Pers. Commun.* **2021**, *121*, 1659–1680. [CrossRef]
28. Duan, H.; Wei, Y.; Liu, P.; Yin, H. A Novel Ensemble Framework Based on K-Means and Resampling for Imbalanced Data. *Appl. Sci.* **2020**, *10*, 1684. [CrossRef]