Generative Multi-view features adaptive high-confidence Sampling for Class-Imbalance Learning

***Abstract*—With the continuous expansion of data availability in a number of daily fields, such as surveilance, security, and finance, people usually meet the challenge of imbalanced data, which might introduce over-fitting risk in model training. Typically, people use sampling methods in imbalanced learning applications that consists of the modification of an imbalanced dataset by some sampling approaches (e.g., random oversampling and undersampling) to provide a balanced data distribution. However, these methods may discard potentially useful samples. Feature selection can reduces irrelevant features for improve performance of model, in this condition, some key and relevant features may be delete, however, it should be retained. To solve this problem, correlation data analysis can take sing-view features divide into multi-view features to process. Therefore, multi-view data process is another task in this paper. To address these challenges, in this paper, we proposed an Multi-view features Imbalance sampling approach via Self-Paced Learning (MISPL) to effectively select the high confidences samples and separate close features for improving the robustness of the training model. Compared with other traditional sampling approaches, the results of experiments showed that our proposed MISPL approach had improved performance of classification by about 15.3%. Especially, G-mean increased 11.5% than original training result (average value of other sampling method) on the experimental datasets. Finally, our experimental results pass the Friedman test and Holm test, which indicate that our experimental processes have significant difference.**

***Index Terms*—Imbalanced classification, Multi-view adaptive sampling, Classification, Self-paced learning**

I. Introduction

In recent years, the imbalanced learning problem has drawn a significant amount of interests from academia, industry, and government funding agencies. Most standard algorithms assume or expect balanced class distributions or equal misclassification costs [1]. Thus, it is urgently needed to investigate sampling algorithms for classification. Machine learning technologies are typically used to find the relationships of instance features. In some cancer instances, it includes many number normal samples and only a few of tumor samples, which might introduce the over-fitting risk in the learning [2]. In other words, the standard algorithms fail to learn at the imbalanced datasets. In most disease prediction systems, there are many clients while a few of them are patient and the others are all healthy. So it is very common to face the imbalanced dataset problem [3].

To solve this problem, people introduce sampling methods, which include random oversampling and undersampling [4], informed undersampling [5], synthetic sampling with data generation [6]. Sampling is a class of methods that alters the size of training sets by sampling a smaller majority training set or repeating instances in the minority training set. It hope that a more balanced training set can give better results. However, the common sampling is randomly chooses samples, which might discards potentially useful data. Recently, people propose the self-paced learning approach which mimics the cognitive mechanism of humans and animals that gradually learns from easy to hard samples, and keep the samples with high confidences and delete the samples with high noises. It has been successfully applied in multi-task learning [7], image classification [8], weakly supervised object detection [9].

Multi-view learning is concerned with the problem of machine learning from data represented by multiple distinct feature sets. The recent emergence of this learning mechanism is largely motivated by the property of data from real applications where examples are described by different feature sets or different "views". For instance, the applications of multi-view learning range from dimensionality reduction [10], active learning [11], clustering [12]. A large number of promising multi-view algorithms have been developed in various fields. As we all know, previous works main is to develop clustering, i.e., unsupervised learning style. Meanwhile, these works [13–15] fuse similarity measurements from diverse views to construct a graph for multi-view examples, which successfully extends conventional multi-view spectral clustering. These works give us inspirations, therefore, in this work, we try to take multi-view transfer into supervised learning, i.e., multi-view features are generated by the between features correlation analysis. Since the increasing request that describing a stuff more and more detailed, single-view data is hard to satisfy the demand. For example, the datasets on Alzheimer's disease (AD) is usually joint different data that achieved by neuroimaging method into a long vector [16]. By this way, the data can contain more information and it can be more accurate to classify these data. However, it is also more difficult at the same time to mine useful information from the data. 1) Since every view of data is independent, it will affect the independence of every view in combined data if we directly joint all views. 2) This kind of data is usually with high-dimension, thus it is hard to build an effective model because of the curse of dimensionality [17]. Moreover, between features may exist similar, which lead to over-fitting generally. For example, human faces usually have highly varied poses in faces recognition, if training these samples directly, it lead to overfitting, because of this large variance in face pose [18, 19]. Else if use dimensionality reduction, the key feature may been deleted. In this condition, we consider to take sing-view features transfer into multi-view features for more robustness of model. Many privous works imply multiple

generative multi-view methods, such as principal component analysis (PCA) [20], co-training [21]. However, to the best of our knowledge, there are no related works how to generate multi-view, and combine with self-paced learning to apply in imbalanced problems.

In this paper, we consider to utilize the benefits of multi-view adaptive sampling and self-paced learning to solve imbalanced distribution in imbalance datasets. Specifically, we use correlation analysis to produce multi-view features data from origin data. Then, we suggest adopting the self-paced learning technique to select the initially high-confidences datasets, and thereby avoid the noisy samples effect and cold start problem as much as possible. That is to say, we use the benefits of multi-view adaptive sampling and self-paced learning to select the samples with high-confidences for more effectively and improve the robustness of the model. Experiments are conducted on twenty-six binary-class imbalanced data sets, and the results demonstrate that the proposed algorithmic framework is generally more effective and efficient than several state-of-the-art sampling methods that were specifically designed for the imbalanced classification scenario.

The remaining sections are organized as follows. Section II gives a brief survey of this paper. In section III, a multi-view imbalanced self-paced learning framework is proposed to produce multi-view features and process noisy samples for improving the performance of classification. Section IV presents the experimental results involved a pure data character and experiments of eighteen real datasets for verification. The conclusion and summary are shown in section V.

II. Related Works

As discussed above, the sampling methods of imbalanced samples are the key task in order to provide a balanced distribution. The balanced dataset provides improved overall classification performance compared to an imbalanced dataset [22]. Thus, many works try to address this challenge in the past few years. Mazurowski et al. [23] investigated a neural network method for imbalanced training data. The results show the BP model is generally preferable over evolution method for imbalanced training data especially with small data sample and large number of features. Yang et al. [24] used the evolution algorithm with multiple classifiers for remedying the class imbalance problem in medical and biological data mining. However, one of the most used techniques to deal with imbalance datasets is preprocessing the data in the learning process. For example, people [25] used evolutionary algorithms to select the most suitable generalized examples for imbalanced datasets.

Aside from the basic undersampling and oversampling methods, some famous methods are also proposed in more complex ways. Synthetic Minority Oversampling Technique (SMOTE) [26] added new synthetic minority class examples by randomly interpolating pairs of closest neighbors in the minority class. Edited Nearest Neighbours (ENN) [27] removes all instances which have been misclassified by the k-NN rule from the training set, the idea of ENN relies on the fact that one can optimally eliminate outliers and possible overlap among classes from a given training set so that the training of the corresponding classifier becomes easier in practice. EasyEnsemble [28] is chosen samples several subsets from the majority class, trains a learner using each of them, and combines the outputs of those learners.

Self-paced learning, as a new learning strategy, has been applied in many tasks in recently years [29]. Jiang et al. [30] proposed SpaR method to solve multi-instance multimedia event detection problem. Zhang et al. [31] proposed SP-MIL model to achieve saliency detection. Jiang et al. [32] proposed SPLD method to video action recognition. Xia et al. [33] proposed collaborative matrix method to predict DrugTarget interactions.

Multi-view learning has two representative techniques canonical correlation analysis [34] and co-training [35]. Co-training is originally designed for datasets with two distinct views. It trains classifiers separately on each view, and adds the most confidently predicted examples of either classifier to the training set of the other in each iteration. From the procedures of co-training, it can be found that it requires the predictions on each view to be accurate. In other words, the overall classification results may be deteri-orated if either classifier provides erroneous information to the other [36]. The canonical correlation analysis aims to learn two types of mapping functions to project two views into a common space, maximizing their correlation [37]. Also, later lots of theories and methods have been devised to investigate. Some existence data may collect from different domains, therefore, we take sing-view transfer into multi-view features. The common methods, such as principal component analysis (PCA) [20] and co-training [38], manually generating multiple views can still improve work favorably.

III. Methods

In real-world machine learning applications, there are often multiple ways to represent the features of an example. For instance, a web page can be represented by words in the page while it can also be represented by all the hyper-links referring to it from other pages. Similarly, an internet image can be represented by the visual features within it and also by the text surrounding it. Generally, the research to deal with this problem is known as multi-view learning and has attracted wide interests in recent years [39]. Various methods have been proposed to learn with multiple views better than the naive approach of using one view or concatenating all views [40, 41]. Even when there are no natural multiple views, manually generating multiple views can still improve work favorably [42]. Therefore, we use correlation analysis to generative multi-view features for improving effectiveness of model.

A. Generative multi-view features

Generally speaking, correlation analysis is wide used to analyze the linear correlation between two variables. Pearson correlation coefficient as one of various ways, is used to measure the correlation between two variable. In particular,

we define a Pearson correlation coefficient r_{uv} between two variables U and V as follows,

$$r_{uv} = \frac{\sum_{i=1}^{n}(U_i - \bar{U})(V_i - \bar{V})}{S_u S_v}, \quad (1)$$

where S_u and S_v are standard deviation of variables U and V, respectively. The r_{uv} value is in a range of $[-1, 1]$, when it is quite close to both ends, variables U and V are further cross correlation. In this situation, variables are divided into multi-view features. The more correlation variables are, the more views will been have. Each view's features will been training respectively, best result will be output value of model.

B. Multi-view sampling with self-paced learning to imbalanced samples

In order to annotate the above concerns, we integrated the self-paced learning approach and multi-view adaptive sampling technique in this paper. The process of our proposed method is shown in Algorithm 1. At first, multi-view features are generated by correlation analysis. Then, a classifier $\boldsymbol{f}$ (e.g. a Logistics regression) is trained on the $\boldsymbol{i}$-th view and the final classifier is obtained by next steps. Then, the *while* of model converge when the achieved number of majority samples not less than number of minority samples. In particular, high confidences samples $\{(\hat{\mathbf{x}}, \hat{y})\}$ are chosen firstly by MISPL for samples balanced distribution.

Algorithm 1: MISPL

Input: Training set $(x, y)_{i=1}^{m}$, initial SPL weights v_0, and a stepsize μ

Output: Model output result Re

Learn Multi-views: according to Equation (1) to obtain multi-view features F_n based various features.

if $i=1 \rightarrow n$ **then**

  Learn a classifier: solve Equation (2) to obtain L based v.

  while *not converged* **do**

    Updated λ: Find optimal pseudo label for each of selected instance by solving $\lambda^* = argmin_{\lambda} \quad v_i^* L(y_i, g(x_i; w)) + g(\lambda, v_i^*)$ *;*

    if λ^* *is small* **then**

      increase with λ by the stepsize μ;

    end

  end

  λ is obtained to solve Equation (3), next updated training samples are obtained to train a model.

  then*: result* $\boldsymbol{Re_i}$ *is obtained by test samples according to above training model.*

end

The highest Re is returned by compared to above output value ***Re****.*

Return: Re

In this work, a number of majority samples with high confidences of number equalled minority samples are chosen to train model. Therefore, giving training dataset $D = (x_i, y_i)_{i=1}^{m}$ with m samples, where $x_j \in R^d$ is the $\boldsymbol{j}$-*th* sample, y_j is the optional information according to the learning objective (e.g. y_j can be the label of x_j in a classification model). Let $\boldsymbol{f}(x_j, w)$ denotes the learned model and w is the model parameter. $L(y_j, \boldsymbol{f}(x_j, w))$ is the loss function of $\boldsymbol{j}$-*th* sample. The objective of SPL is to jointly optimize the model parameter w and latent sample weights $v = [v_1, v_2, ..., v_m]$ via the following minimization problem:

$$\min_{w,v} E(w, v; \lambda) = \sum_{j=1}^{m} v_i L(y_i, \boldsymbol{f}(x_j, w)) + g(\lambda, v_j), \quad (2)$$

where $g(\lambda, v_j)$ is self-paced regularize and λ is a penalty that controls the learning pace. Specifically, giving sample weights v, the minimization over w is a weighted loss minimization problem, independent on regularizer $g(\lambda, v)$; Giving model parameter w, the optimal weight of $\boldsymbol{j}$-*th* sample is determined in Equation (3),

$$min_{v_j} v_j L(y_j, \boldsymbol{f}(X_j, w)) + g(\lambda, v_j) \quad (3)$$

Since $l_j = L(y_j, \boldsymbol{f}(X_j, w))$ is constant once w is given, the optimal value of v_j is uniquely determined by the corresponding minimizer function $\sigma(\lambda, l_j)$ that satisfies

$$\sigma(\lambda, l_j) + g(\lambda, \sigma(\lambda, l_j)) \leq v_j l_j + g(\lambda, v_j), \forall v_j \in [0, 1] \quad (4)$$

For example, if $g(\lambda, v_j) = -\lambda v_j$ [29], the optimal v_j^* is calculated by

$$v_j^* = \sigma(\lambda, l_j) = \begin{cases} 1 & \textit{if } l_j \leq \lambda \\ 0 & \textit{otherwise} \end{cases} \quad (5)$$

By gradually increasing the value of λ so that μ more samples will be added in the next iteration, increasingly low quality samples are included into the training process until dataset balanced distribution. Many existing researches have been devoted to learning appropriate minimizer functions [43, 44], which are classified as SPL with explicit regularizers, since they usually require the explicit form of regularizer $g(\lambda, v)$. $\sigma(\lambda, l)$ is then derived from the form of $g(\lambda, v)$. Finally, undersampling samples with high confidences make positive and negative samples consensus.

IV. Experimental Results

In this section, in order to further evaluate the proposed method, experiments are conducted and analyzed in details to explore the performance improvement. This original dataset contains the expression profiles are from internet public variable datasets, such as US National Library of Medicine National Institutes of Health and the UC Irvine Machine Learning Repository, furthermore details are shown in Table I.

Table I shows details of training samples, including multiple types datasets, such as biomedical, finance, recognition. The biggest imbalanced ratio is 129.4. The breast cancer dataset, as a example, includes 569 samples, among 212 minority samples and 357 majority samples, their imbalance sample ratio is 1.68, each sample also includes 30 attributes. With the increase of training model parameter λ, the accurate training rate is steadily changed. A group of samples with balanced

TABLE I
26 PUBLICLY AVAILABLE DATASETS USED IN THE EXPERIMENTS, #min/#maj is the size of minority and majority class

Dataset	Samples	#min/#maj	Ratio	Attributions
Ionosphere	315	126 / 189	1.50	34
Vehicle1	846	217 / 629	2.90	18
Vehicle2	846	218 / 628	2.88	18
Vehicle3	846	212 / 634	2.99	18
Wine-white	4898	1060 / 3838	3.62	11
Glass-small_vs_large	214	51 / 163	3.20	9
Ecoli1	336	77 / 259	3.36	7
Ecoli2	336	52 / 284	5.46	7
Car-good	1728	69 / 1659	24.04	6
Glass6	214	29 / 185	6.38	9
Yeast3	1484	163 / 1321	8.10	8
Ecoli3	336	35 / 301	8.60	7
Page-blocks0	5472	559 / 4913	8.79	10
Abalone19	4174	32 / 4142	129.44	8
Network security detection	8793	1824 / 6969	3.82	444
Vowel0	988	90 / 898	9.98	13
Glass5	214	9 / 205	22.78	9
Yeast6	1484	35 / 1449	41.40	8
Zoo-3	101	5 / 96	19.20	16
Winequality-red-6	1599	217 / 1382	6.37	11
Letter Recognition	20000	789 / 19211	24.3	16
Cardiotocography_C1	2126	384 / 1742	4.5	22
Mfeat-mor0	1808	1207 / 601	2.0	216
Credit card clients	30000	6636 / 23364	3.0	23
Magic	19020	6688/ 12332	1.8	11
Breast cancer	569	212 /357	1.68	30

distribution is selected to training next cycle under the multi-view features in the MISPL.

Before star our training, 70% of samples are used in training set, 30% of samples are used in testing set. In order to assess the accuracy of classification, AUC, F1-Score, and G-mean are also used in the experiment. Some famous sampling methods are used in this paper for comparing the advantage of our proposal method, including Edited Nearest Neighbours (ENN) [27], Synthetic Minority Oversampling Technique (SMOTE) [26], EasyEnsemble [28], One Sided Selection [45], and Neighborhood Cleaning Rule (NCL) [46].

Table II by line of comparison shows that F1-score of MISPL is more than other sampling approaches clearly, such as EasyEnsemble, SMOTE. The comparison results of proposed sampling method with other previous methods show that the proposed MISPL method always outperforms others in term of F1-Score. The best F1-score value achieved by MISPL model is 96.49% on the Vehicle2 dataset, which is about 2% higher than the worst F1-score value with EasyEnsemble (94.98%). The highest improved F1-score value with MISPL is 75.57% in network security detection, previous value is 11.67% with One Sided Selection, i.e. increased about 64% of F1-score by MISPL. On other hand, standard deviation average value of MISPL is higher than others, which indicates that our proposed MISPL method is robustness in general. In particularly, comparison with different models in term of F1-Score in various data, result of MISPL common is better than other models, that implying MISPL model effectiveness better than other sampling methods.

Meanwhile, Table III shows that average G-mean of MISPL value is higher than other approaches. The highest improved G-mean value with MISPL is 94.48% in Yeast3, previous value is 12.37% with ENN, i.e. increased highest about 82% of G-mean by MISPL. On other hand, standard deviation average value of MISPL is higher than others, which indicates that our proposed MISPL method is robustness in general. Simultaneously, we find that standard deviation value of G-mean is higher than F1-score, which indicates that MISPL pay attention to true positive value.

ROC (Receiver Operating Characteristic) is one of styles to assess performance of the model, experimental results with multiple training times are shown in Table IV. Table IV also shows that average AUC of MISPL is higher than others. However, standard deviation value of MISPL is slightly higher than ENN, which indicates robustness of MISPL is less than ENN.

Table II,III,IV show that average performance with AUC, F1-Score and G-mean are compared in seven sampling methods on 26 datasets. Our MISPL method is highest in six performance measure, which indicates that MISPL method are effectively and robustly. Average F1-score increased about 20% than One sided selection training, average G-mean increased about 19% than ENN training, and average AUC increased about 15% than ENN training. Meanwhile, other models, such as SMOTE, are also improving performance of model.

A. Friedman test

To compare the statistical significance of performance of our proposed with existing methods, we adopted the Friedman test used in the study [47]. Based on the performance ranking of different approaches in eight datasets, the Friedman test can measure the statistical differences in multiple methods. The Friedman estimator F_F, followed by a Fisher distribution, is shown in Equation (6).

$$F_F = \frac{(N_D - 1)\chi_r^2}{N_D(N_M - 1) - \chi_r^2},$$
$$\chi_r^2 = \frac{12N_D}{N_M(N_M+1)}\left(\sum_{i=1}^{N_M} R_i^2 - \frac{N_M(N_M+1)^2}{4}\right),$$
$$R_i = \frac{\sum_{j=1}^{N_D} r_{ij}}{N_D}, \quad (6)$$

where N_M is the number of approaches, N_D is the number of data sets, R_i is the average ranking in the i^{th} method, and r^{ij} represents the ranking of i^{th} approaches on the j^{th} dataset. For each dataset style evaluation, the AUC, F1-Score, G-mean and standard deviation (Std.) value of MISPL method are ranked from one to the number of methods, respectively.

TABLE II
COMPARISON WITH DIFFERENT MODELS IN TERM OF F1-SCORE ON DIFFERENT DATASETS

Dataset	ENN	SMOTE	EasyEnsemble	One side selection	NCL	ISPL	MIPSL
Ionosphere	.7543 ± .036	.7628 ± .062	.7724 ± .050	.7758 ± .052	.7904 ± .060	.7982 ± .048	**.8204 ± .030**
Vehicle1	.7828 ± .039	.7562 ± .032	.8027 ± .025	.632 ± .040	.7709 ± .027	.8162 ± .025	**.8248 ± .018**
Vehicle2	.952 ± **.007**	.9499 ± .020	.9498 ± .017	.9558 ± .011	.9503 ± .017	.955 ± .011	**.9649** ± .015
Vehicle3	.7278 ± .045	.7614 ± **.004**	.7534 ± .043	.5915 ± .053	.7288 ± .037	.7939 ± .018	**.8255** ± .017
Wine-white Glass-small vs large Ecoli1	.5041 ± .056	.6973 ± .041	.7226 ± **.010**	.3572 ± .044	.495 ± .075	.7479 ± .015	**.7557** ± .014
Ecoli2	.8428 ± .054	.9073 ± .033	.8796 ± .074	.8531 ± .082	.9097 ± .047	.9445 ± .042	**.9604 ± .015**
Car-good	.7629 ± .042	.8859 ± .064	.8903 ± .070	.7243 ± .093	.8166 ± **.014**	.8963 ± .039	**.9111 ± .035**
Glass6	.3488 ± .077	.8713 ± .028	.8472 ± .041	.3078 ± .116	.3117 ± .078	.84 ± .037	**.8734 ± .013**
Yeast3	.4287 ± .112	.8859 ± **.009**	.8903 ± .046	.7243 ± .419	.8166 ± .072	.8963 ± .023	**.9111 ± .036**
Ecoli3	.8622 ± .075	.8912 ± **.039**	.8912 ± .039	.8724 ± .034	.8853 ± .073	.919 ± .066	**.9306** ± .064
Page-blocks0	.1237 ± .055	.9206 ± .035	.8833 ± .035	.0719 ± .032	.1246 ± **.031**	.9067 ± .138	**.9448** ± .059
Abalone19	.8809 ± **.009**	.8982 ± .072	.8601 ± .074	.8439 ± .068	.8486 ± .036	.9109 ± .050	**.9241** ± .033
Network security detection	.8349 ± .040	.8698 ± **.037**	.8283 ± .107	.6227 ± .043	.7905 ± .115	.8679 ± .051	**.9357** ± .049
Vowel0	.8669 ± .018	.8775 ± .021	.8785 ± **.010**	.8371 ± .054	.7448 ± .034	.8841 ± .014	**.9166** ± .023
Glass5	.3308 ± .080	.7091 ± .146	.7226 ± **.010**	.1167 ± .032	.3012 ± .026	.689 ± .012	**.7557** ± .014
Yeast6	.8546 ± .054	**.948** ± .028	.9222 ± .031	.8318 ± .084	.9013 ± .039	.826 ± .019	.8827 ± **.019**
Zoo-3	.5381 ± .065	.8747 ± .057	.8968 ± .025	.5234 ± .046	.5726 ± .076	.933 ± .052	**.9603 ± .008**
Winequality-red-6	.6000 ± .363	.8511 ± .091	.8511 ± .089	.76 ± .146	.8267 ± .167	.88 ± .110	**.96 ± .089**
Letter Recognition	.6288 ± .425	.7906 ± .141	.8072 ± .089	.5685 ± .073	.5275 ± .060	.843 ± **.028**	**.8611** ± .073
Cardiotocography	.4716 ± .077	.7487 ± .095	.7558 ± .042	.2791 ± .103	.4834 ± .084	.782 ± .025	**.8025 ± .023**
Mfeat-mor0	.8953 ± .024	.9149 ± .022	.9002 ± .012	.8970 ± .017	.8971 ± .014	.9035 ± .012	**.9212 ±.007**
Credit card clients	.7146 ± .034	.8332 ± .031	.8308 ± .034	.6030 ± .054	.7166 ± .042	.8153 ± .029	**.8419 ±.047**
Magic Brest cancer Avg.	.9105 ± .043	.9213 ± .022	.9248 ± .019	.9070 ± .043	.9206 ± .044	.9330 ± .025	**.9345 ±.010**
	.5934 ± .023	.6446 ± .019	.6609 ± .017	.5276 ±**.0097**	.5914 ± .010	.6822 ± .020	**.6862** ± .014
	.7544 ± .007	.7600 ± .008	.7595 ±**.0055**	.6892 ± .008	.7613 ± .006	.8102 ± .013	**.8207** ± .011
	.9561 ± .031	.9592 ± .022	.9468 ± .021	.9554 ± .012	.9253 ± .018	.9323 ± .030	**.9639 ±.007**
	.6893 ± .073	.8419 ± .045	.8396 ± .040	.6472 ± .068	.7080 ± .050	.8541 ± .037	**.8804 ±.029**

TABLE III
COMPARISON WITH DIFFERENT MODELS IN TERM OF G-MEAN ON DIFFERENT DATASETS

Dataset	ENN	SMOTE	EasyEnsemble	One side selection	NCL	ISPL	MIPSL
Ionosphere	.7782 ± **.0298**	.7815 ± .050	.7906 ± .041	.7933 ± .044	.8049 ± .052	.8042 ± .044	**.8254** ± .030
Vehicle1	.7811 ± .040	.7616 ± **.026**	**.7981** ± .027	.6742 ± .033	.7715 ± .029	.7508 ± .042	.7626 ± .031
Vehicle2	.9520 ± **.006**	.9505 ± .019	.9490 ± .018	.9565 ± .011	.9504 ± .018	.9550 ± .010	**.9645** ± .016
Vehicle3	.7416 ± .039	.7593 ± **.011**	.7560 ± .035	.6433 ± .038	.7370 ± .034	.7171 ± .055	**.7809** ± .044
Wine-white Glass-small vs large Ecoli1	.5779 ± .041	**.7099** ± .031	.7047 ± .023	.4658 ± .035	.5693 ± .056	.6784 ± .028	.7059 ± **.008**
Ecoli2	.8505 ± .050	.9108 ± .031	.8842 ± .070	.8600 ± .074	.9118 ± .045	.9445 ± .040	**.9595 ± .015**
Car-good	.7779 ± .035	.8751 ± .076	.8847 ± .075	.7535 ± .076	.8265 ± **.011**	.8734 ± .054	**.9012** ± .042
Glass6	.4585 ± .063	**.8652 ± .024**	.8446 ± .039	.4240 ± .093	.4282 ± .064	.8445 ± .035	.8593 ± .027
Yeast3	.5214 ± .087	.8751 ± **.013**	.8847 ± .069	.7535 ± .338	.8265 ± .056	.8734 ± .030	**.9012** ± .045
Ecoli3	.8693 ± .068	.8956 ± .037	.8956 ± .037	.8799 ± **.031**	.8926 ± .064	.9094 ± .073	**.9214** ± .072
Page-blocks0	.2512 ± .067	.9220 ± **.032**	.8808 ± .036	.1894 ± .045	.2563 ± .035	.9156 ± .121	**.9474** ± .056
Abalone19	.8863 ± **.009**	.9024 ± .065	.8262 ± .109	.8551 ± .058	.8581 ± .032	.9010 ± .062	**.9192** ± .040
Network security detection	.8444 ± .036	.8673 ± .046	.8035 ± .114	.6725 ± **.034**	.8016 ± .109	.8552 ± .057	**.9257** ± .062
Vowel0	.8746 ± .016	.8749 ± .018	.8826 ± **.008**	.8490 ± .047	.7705 ± .028	.8893 ± .012	**.9167** ± .022
Glass5	.4430 ± .062	.7004 ± .115	.7047 ± .023	.2474 ± .035	.4200 ± .021	.5785 ± .028	**.7059 ± .008**
Yeast6	.8644 ± .048	**.9479 ± .027**	.9169 ± .030	.8453 ± .071	.9062 ± .036	.7594 ± .037	.8560 ± .028
Zoo-3	.6069 ± .050	.8768 ± .054	.8909 ± .032	.5954 ± .036	.6335 ± .059	.9315 ± .056	**.9588 ± .008**
Winequality-red-6	.6243 ± .371	.8570 ± **.084**	.8330 ± .100	.7657 ± .131	.8243 ± .160	.8243 ± .160	**.9414** ± .131
Letter Recognition	.6909 ± .355	.8106 ± .117	.8059 ± .075	.6304 ± .056	.5986 ± .046	.8444 ± **.036**	**.8630** ± .064
Cardiotocography	.5543 ± .059	.7509 ± .080	.7590 ± .037	.3980 ± .093	.5639 ± .065	.7323 ± .057	**.7689 ± .030**
Mfeat-mor0	.9004 ± .022	.9172 ±**.021**	.9048 ± .011	.9019 ± .016	.9020 ± .013	.9063 ± .012	**.9233 ± .007**
Credit card clients	.7429 ±**.029**	.8162 ± .037	.8084 ± .040	.6565 ± .042	.7436 ± .036	.7857 ± .058	**.8460** ± .045
Magic Brest cancer Avg.	.9131 ± .041	.9219 ± .022	.9240 ± .020	.9096 ± .041	.9225 ± .041	.9275 ± .034	**.9345 ± .010**
	.6463 ± .018	.6780 ± .009	.6868 ± .0075	.5978 ± .008	.6452 ± .008	.6774 ± .020	**.6940 ± .0072**
	.7666 ± .006	.7697 ± .006	.7699 ±**.0050**	.7214 ± .007	.7693 ± .0053	.7826 ±**.040**	**.8091** ± .023
	.9568 ± .030	.9585 ± .022	.9474 ± .021	.9555 ± .012	.9251 ± .017	.9295 ± .031	**.9634 ± .007**
	.7260 ± .066	.8445 ± .041	.8360 ± .042	.6921 ± .058	.7407 ± .044	.8304 ± .047	**.8675 ± .034**

In this comparison, seven different methods are considered. $r_{ij}^{\alpha} = 1$ denotes the highest AUC, F1-Score, or G-mean and $r_{ij}^{\alpha} = 7$ denotes the worst AUC, F1-Score, or G-mean. For model diagnosis variance test, $r_{ij}^{\alpha} = 1$ denotes the lowest diagnosis AUC, F1-Score, or G-mean variation, and $r_{ij}^{\alpha} = 7$ represents the model with the highest AUC, F1-Score, or G-mean variation. In this test, F_F follows a Fisher distribution with N_M - 1 and (N_M - 1)(N_D - 1) degrees of freedom, and the confidence level α is set as 0.05. Meanwhile, N_M = 7 and N_D = 26, with degree of freedom N_M-1 = 6 and ($N_M - 1$)($N_D - 1$) = 28 are applied, which obtain a critical value of the Fisher distribution $F(6, 150) = 2.09$.

Table V shows the Friedman test results for AUC, F1-score and G-mean. The average rankings for MISPL (R_{MISPL})

TABLE IV
COMPARISON WITH DIFFERENT MODELS IN TERM OF AUC ON DIFFERENT DATASETS

Dataset	ENN	SMOTE	EasyEnsemble	One side selection	NCL	ISPL	MIPSL
Ionosphere	.8701 ± .032	.8866 ± **.023**	.8611 ± .032	.8669 ± .029	.8875 ± .039	.8714 ± .058	**.9064** ± .027
Vehicle1	.8728 ± .039	.8589 ± **.015**	.8749 ± .034	.8639 ± .031	.8573 ± .035	.8775 ± .017	**.8863** ± .023
Vehicle2	.9816 ± .014	.9896 ± .009	.9798 ± .015	.9877 ± **.003**	**.9904** ± .01	.9865 ± .008	.9891 ± .01
Vehicle3	.8281 ± .022	.8306 ± **.019**	.8447 ± .036	.8453 ± .021	.8385 ± .03	.8466 ± .03	**.8486** ± .035
Wine-white Glass-small vs large Ecoli1	.7366 ± .052	.7621 ± .018	.7605 ± .025	.7337 ± **.01**	**.7726** ± .017	.7425 ± .033	.7706 ± .021
Ecoli2	.9396 ± .038	.9591 ± **.01**	.9689 ± .029	.9644 ± .027	.9422 ± .065	.9742 ± .02	**.9813** ± .019
Car-good	.9285 ± .018	.9285 ± .06	.9543 ± .042	.9372 ± .026	.9592 ± .012	.9395 ± .02	**.9709** ± **.011**
Glass6	.9573 ± **.01**	.9189 ± .028	.9004 ± .023	.9244 ± .043	**.9662** ± .026	.9422 ± .028	.9536 ± .054
Yeast3	.8220 ± **.023**	.9285 ± .028	.9543 ± .074	.9372 ± .075	.9592 ± .037	.9395 ± .027	**.9709** ± .055
Ecoli3	.9188 ± .062	.9188 ± .051	.9188 ± .051	.8688 ± .085	.9750 ± **.026**	.9500 ± .051	**.9875** ± .028
Page-blocks0	.9741 ± **.006**	.9752 ± .013	.9387 ± .028	.9668 ± .014	.9616 ± .019	.9719 ± .039	**.9813** ± .034
Abalone19	.9125 ± **.009**	.9208 ± .066	.9140 ± .073	.9087 ± .057	.9236 ± .044	.9165 ± .054	**.9262** ± .043
Network security detection	.8710 ± .025	.9110 ± .051	.8900 ± .077	.8412 ± **.013**	.8800 ± .141	.8760 ± .067	**.9290** ± .057
Vowel0	.9197 ± .022	.9347 ± .011	.9256 ± .03	.9227 ± .009	.8545 ± .008	.9372 ± **.007**	**.9479** ± .008
Glass5	.6966 ± **.009**	**.837** ± .122	.7605 ± .025	.6476 ± .011	.6607 ± .011	.6256 ± .017	.7706 ± .021
Yeast6	.9797 ± .022	**.9923** ± **.008**	.9849 ± .01	.9813 ± .013	.9871 ± .01	.9896 ± .014	.9877 ± .014
Zoo-3	.9728 ± .02	.9663 ± .013	.9660 ± .014	**.9813** ± **.012**	.9646 ± .026	.9435 ± .057	.9621 ± .018
Winequality-red-6	.8300 ± .144	.9220 ± .087	.9380 ± **.051**	.8080 ± .187	.8900 ± .108	.8500 ± .137	**.9500** ± .112
Letter Recognition	**.9463** ± .054	.8826 ± .079	.8930 ± .08	.8230 ± .046	.8250 ± .054	.8670 ± .054	.8920 ± **.043**
Cardiotocography	**.8228** ± **.019**	.7883 ± .095	.7870 ± .036	.7555 ± .062	.7443 ± .071	.7774 ± .054	.8052 ± .04
Mfeat-mor0	.9326 ± .005	.9229 ± .010	.9355 ±**.0039**	.9054 ± .005	.9038 ± .004	.9254 ± .014	**.9375** ± .011
Credit card clients	.8391 ± .025	.8257 ± .034	.8260 ± .031	.7878 ± .017	.8011 ±**.015**	.8421 ± .053	**.8994** ± .034
Magic Brest cancer Avg.	.9678 ± .021	.9736 ± .019	.9736 ± .014	.9686 ± .029	.9748 ± .021	.9835 ± .0064	**.9846** ±**.0063**
	.7403 ± .012	.7457 ± .009	.7530 ± .008	.7482 ±**.0038**	.7451 ± .009	.6874 ± .040	.7440 ± .008
	.8415 ± .0044	.8406 ± .005	.8425 ± .005	.8361 ± .006	.8409 ±**.0042**	.8724 ± .022	**.8866** ± .013
	.9523 ± .007	.9518 ± .008	.9532 ± .003	.9519 ± .004	.9443 ± .004	.9610 ± .018	**.9859** ± **.003**
	.8867 ± .027	.8989 ± .034	.8961 ± .033	.8755 ± .032	.8835 ± .033	.8883 ± .036	**.9175** ± .029

TABLE V
FRIEDMAN TEST RESULTS

Test item	R_{MISPL}	X_F^2	$F_F(2.09)$	Decison
F1-Score value	1.1	103.5	49.3	Positive
F1-Score Std. value	2.7	23.02	4.32	Positive
G-mean value	1.4	88.5	32.8	Positive
G-mean Std. value	3	8.25	1.40	Negative
AUC value	1.8	35.8	7.4	Positive
AUC Std. value	3.6	-1.66	-0.26	Negative

are the best on seven sampling methods in terms of the three performance measures. Additional, the results show that experimental results have significant statistical difference of the ranking for AUC, F1-score Std. value, F1-Score, and G-mean as well. However, we can know from Table V that G-mean Std. value and AUC Std. value are not significant, their results cannot to measure ranking.

B. Holm test

We know from Table V that the results show statistically significant differences. Furthermore, the Holm test is used to compare significant differences in the performance of MISPL with other methods [48]. The Holm test is a ranking of multiple methods under multiple datasets according to accuracy rate.

The test statistics for comparing the i-th and m-th approach using these methods are as follow:

$$z = \frac{(R_i - R_m)}{\sqrt{\frac{N_M(N_M+1)}{6N_D}}}, \tag{7}$$

where N_M is the number of classifiers, N_D is the number of datasets, and $R = \frac{1}{N_D}\sum_{i=1}^{N_D} r_i^d$, r_i^j represents the ranking of the i-th classifier of the j-th dataset. The z value is used to find the corresponding probability from the normal distribution table, which is then compared with an appropriate α. The tests differ in the way they adjust the value of α to compensate for multiple comparisons. More details are described in [48].

According to the following description, the F1-Score, G-mean and AUC are used to ranking these results. Therefore, the average ranking for each method, F1-score as a instance, is $R_{OneSideSection}$=6.3, R_{ENN}=5.5, R_{NCL}=5.2, $R_{EasyEnsemble}$=3.7, R_{SMOTE}=3.5, R_{ISPL}=2.7 and R_{MISPL}=1.1, respectively. The Holm test results are shown in Table VI. We know from Table VI that the Holm procedure rejects the first to sixth hypotheses since the corresponding p values are smaller than the adjusted αs. These results indicate that pass significant test statistic.

V. CONCLUSION

Imbalanced size data is very common in the widely daily application, which causes great challenges to existing mining and learning algorithms. With the rapid growth of the imbalanced ratio, it leads to loss lots of key information, which greatly reduces the performance of machine learning algorithms, increases computational complexity, and reduces the prediction ability of learning model.

This paper proposed a self-paced learning sampling method for imbalanced classification in twenty-six real applications. Our MISPL could reduce noise of imbalance samples to improve the performance of classification, that is, removes some irrelevant and redundant samples and finds the suitable subset. Compared with conventional methods, our MISPL method can achieved improved F1-score about 15.3%, and G-mean increased 11.5%.

TABLE VI
HOLM TEST RESULTS

i	method	$z = \frac{(R_i - R_{ISPL})}{\sqrt{\frac{N_M(N_M+1)}{6N_D}}}$	p	$\alpha/(N_M - i)$
F1-score				
1	OneSideSection	8.68	<0.0001	0.0083
2	ENN NCL	7.34	<0.0001	0.0100
3	EasyEnsemble	6.84	<0.0001	0.0125
4	SMOTE	4.34	<0.0001	0.0167
5	ISPL	4.01	0.0001	0.0250
6		2.67	0.0038	0.5000
G-mean				
1	OneSideSection	8.18	<0.0001	0.0083
2	ENN	6.68	<0.0001	0.0100
3	NCL	6.18	<0.0001	0.0125
4	ISPL	3.84	<0.0001	0.0167
5	EasyEnsemble	3.5	0.0002	0.0250
6	SMOTE	2.84	0.0023	0.5000
AUC				
1	OneSideSection	6.01	<0.0001	0.0083
2	ENN	4.84	<0.0001	0.0100
3	NCL	4.17	<0.0001	0.0125
4	SMOTE	3.54	<0.0001	0.0167
5	EasyEnsemble	3.48	0.0001	0.0250
6	ISPL	3.41	0.0003	0.5000

This paper using imbalance samples to classification have become a major concern in the field of machine learning. In addition to that, this model can continue to work in the absence of any manual labeling for saving much time and cost. It will be efficient tool to make solutions for feature selection because of its high reliability and strong anti-noise and outliers in some dataset.

VI. ACKNOWLEDGMENTS

The authors would like to thank reviewers for their constructive comments.

REFERENCES

[1] Q. Li and Y. Mao, "A review of boosting methods for imbalanced data classification," *Pattern Analysis and Applications*, vol. 17, no. 4, pp. 679–693, 2014.

[2] Y. Liu, X. Yu, J. X. Huang, and A. An, "Combining integrated sampling with svm ensembles for learning from imbalanced datasets," *Information Processing & Management*, vol. 47, no. 4, pp. 617–631, 2011.

[3] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[4] E. Ramentol, "Smote-rs: a hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using smote and rough sets theory," *Knowledge and Information Systems*, vol. 33, no. 2, pp. 245–265, 2012.

[5] S. J. Yen and Y. S. Lee, "Cluster-based under-sampling approaches for imbalanced data distributions," *Expert Systems with Applications*, vol. 36, no. 3, pp. 5718–5727, 2009.

[6] R. Blagus and L. Lusa, "Smote for high-dimensional class-imbalanced data," *BMC Bioinformatics,14,1(2013-03-22)*, vol. 14, no. 1, pp. 1–16, 2013.

[7] C. Li, F. Wei, J. Yan, W. Dong, Q. Liu, and H. Zha, "Self-paced multi-task learning," *Thirty-First AAAI Conference on Artificial Intelligence*, pp. 2175–2181, 2017.

[8] Y. Tang, Y. B. Yang, and Y. Gao, "Self-paced dictionary learning for image classification," in *ACM International Conference on Multimedia*, 2012, pp. 833–836.

[9] D. Zhang, J. Han, L. Zhao, and D. Meng, "Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework," *International Journal of Computer Vision*, pp. 1–18, 2018.

[10] M. White, X. Zhang, D. Schuurmans, and Y.-l. Yu, "Convex multi-view subspace learning," in *Advances in Neural Information Processing Systems*, 2012, pp. 1673–1681.

[11] Q. Zhang and S. Sun, "Multiple-view multiple-learner active learning," *Pattern Recognition*, vol. 43, no. 9, pp. 3113–3119, 2010.

[12] X. Chang, D. Tao, and X. Chao, "Multi-view self-paced learning for clustering," in *International Conference on Artificial Intelligence*, 2015.

[13] L. Huang, H.-Y. Chao, and C.-D. Wang, "Multi-view intact space clustering," *Pattern Recognition*, vol. 86, pp. 344–353, 2019.

[14] S. Huang, Z. Kang, and Z. Xu, "Self-weighted multi-view clustering with soft capped norm," *Knowledge-Based Systems*, vol. 158, pp. 1–8, 2018.

[15] Z. Zhang, L. Liu, F. Shen, H. T. Shen, and L. Shao, "Binary multi-view clustering." *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2018.

[16] D. Zhang and D. Shen, "Multi-modal multi-task learning for joint prediction of multiple regression and classification variables in alzheimer's disease," *Neuroimage*, vol. 59, no. 2, pp. 895–907, 2012.

[17] X. Cheng, Y. Zhu, J. Song, G. Wen, and H. Wei, "A novel low-rank hypergraph feature selection for multi-view classification," *Neurocomputing*, vol. 253, no. C, pp. 115–121, 2017.

[18] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Aggregate channel features for multi-view face detection," *IEEE international joint conference on biometrics*, pp. 1–8, 2014.

[19] J. Deng, G. Trigeorgis, Y. Zhou, and S. Zafeiriou, "Joint multi-view face alignment in the wild," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3636–3648, 2019.

[20] G. Chao and S. Sun, "Semi-supervised multi-view maximum entropy discrimination with expectation laplacian regularization," *Information Fusion*, vol. 45, pp. 296–306, 2019.

[21] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual conference on Computational learning theory*. ACM, 1998, pp. 92–100.

[22] A. Estabrooks, T. Jo, and N. Japkowicz, "A multiple resampling method for learning from imbalanced data sets," *Computational intelligence*, vol. 20, no. 1, pp. 18–

36, 2004.
[23] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, "Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance," *Neural networks*, vol. 21, no. 2-3, pp. 427–436, 2008.
[24] P. Yang, L. Xu, B. B. Zhou, Z. Zhang, and A. Y. Zomaya, "A particle swarm based hybrid system for imbalanced medical data sampling," *BMC Genomics*, vol. 10, no. 3, pp. 1–14, 2009.
[25] S. Garcı, I. Triguero, C. J. Carmona, F. Herrera *et al.*, "Evolutionary-based selection of generalized instances for imbalanced classification," *Knowledge-Based Systems*, vol. 25, no. 1, pp. 3–12, 2012.
[26] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.
[27] D. Guan, W. Yuan, Y.-K. Lee, and S. Lee, "Nearest neighbor editing aided by unlabeled data," *Information Sciences*, vol. 179, no. 13, pp. 2273–2282, 2009.
[28] X. Y. Liu, J. Wu, and Z. H. Zhou, "Exploratory undersampling for class-imbalance learning," in *International Conference on Data Mining*, 2006, pp. 965–969.
[29] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Advances in Neural Information Processing Systems*, 2010, pp. 1189–1197.
[30] L. Jiang, D. Meng, T. Mitamura, and A. G. Hauptmann, "Easy samples first: Self-paced reranking for zero-example multimedia search," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 547–556.
[31] D. Zhang, D. Meng, and J. Han, "Co-saliency detection via a self-paced multiple-instance learning framework," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 5, pp. 865–878, 2017.
[32] L. Jiang, D. Meng, S.-I. Yu, Z. Lan, S. Shan, and A. Hauptmann, "Self-paced learning with diversity," in *Advances in Neural Information Processing Systems*, 2014, pp. 2078–2086.
[33] L.-Y. Xia, Z.-Y. Yang, H. Zhang, and Y. Liang, "Improved prediction of drugtarget interactions using self-paced learning with collaborative matrix factorization," *Journal of Chemical Information and Modeling*, vol. 59, no. 7, pp. 3340–3351, 2019.
[34] D. R. Hardoon, S. Szedmak, and J. Shawetaylor, "Canonical correlation analysis: an overview with application to learning methods." *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, 2004.
[35] X. Xu, W. Li, D. Xu, and I. W. Tsang, "Co-labeling for multi-view weakly labeled learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 6, pp. 1113–1125, 2015.
[36] M. L. Zhang and Z. H. Zhou, "Cotrade: Confident co-training with data editing," *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society*, vol. 41, no. 6, pp. 1612–26, 2011.
[37] J. Li, H. Yong, B. Zhang, M. Li, L. Zhang, and D. Zhang, "A probabilistic hierarchical model for multi-view and multi-feature classification," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
[38] J. Zhao, X. Xie, X. Xu, and S. Sun, "Multi-view learning overview: Recent progress and new challenges," *Information Fusion*, vol. 38, pp. 43–54, 2017.
[39] N. Chen, J. Zhu, F. Sun, and E. P. Xing, "Large-margin predictive latent subspace learning for multiview data analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 12, pp. 2365–2378, 2012.
[40] W. Wang and Z.-H. Zhou, "A new analysis of co-training." in *ICML*, vol. 2, 2010, p. 3.
[41] Y. Wang, L. Wu, X. Lin, and J. Gao, "Multiview spectral clustering via structured low-rank matrix factorization," *IEEE transactions on neural networks and learning systems*, vol. 29, no. 10, pp. 4833–4843, 2018.
[42] M.-F. Balcan, A. Blum, and K. Yang, "Co-training and expansion: Towards bridging theory and practice," in *Advances in neural information processing systems*, 2005, pp. 89–96.
[43] D. Zhang, D. Meng, L. Zhao, and J. Han, "Bridging saliency detection to weakly supervised object detection based on self-paced curriculum learning," *IJCAI*, 2017.
[44] Y. Fan, R. He, J. Liang, and B.-G. Hu, "Self-paced learning: An implicit regularization perspective." in *AAAI*, vol. 3, 2017, p. 4.
[45] C. Jia and Y. Zuo, "S-sulfpred: A sensitive predictor to capture s-sulfenylation sites based on a resampling one-sided selection undersampling-synthetic minority over-sampling technique," *Journal of theoretical biology*, vol. 422, pp. 84–89, 2017.
[46] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in *Conference on Artificial Intelligence in Medicine in Europe*. Springer, 2001, pp. 63–66.
[47] M. Friedman, "The use of ranks to avoid the assumption of normality implicit in the analysis of variance." *Publications of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1939.
[48] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, no. Jan, pp. 1–30, 2006.