IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 13, 2020, accepted July 12, 2020, date of publication July 16, 2020, date of current version July 29, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3009753

PSU: Particle Stacking Undersampling Method for Highly Imbalanced Big Data

YONG-SEOK JEON AND DONG-JOON LIM
Department of Industrial Engineering, Sungkyunkwan University, Suwon 16419, South Korea
Corresponding author: Dong-Joon Lim (tgno3@skku.edu)

This work was supported by the National Research Foundation of Korea (NRF), South Korea, under Grant 2020R1F1A1066629.

ABSTRACT Imbalanced classes are a common problem in machine learning, and the computational costs required for proper resampling increases with the data size. In this study, a simple and effective undersampling method, named particle stacking undersampling (PSU) was proposed. Compared with other competing undersampling methods, PSU can significantly reduce the computational costs, while minimizing information loss to prevent a prediction bias. The performance benchmark applied on 55 binary classification problems indicated that the proposed method not only achieved an enhanced classification performance over other well-known undersampling methods (random undersampling, NearMiss-1, NearMiss-2, cluster centroid, edited nearest neighbor, condensed nearest neighbor, and Tomek Links) but also provided a computational simplicity that can be scalable to large data. Moreover, an experiment verified that two propositions forming the basis of the PSU algorithm can also be applied to other undersampling methods to achieve methodological improvements.

INDEX TERMS Data mining, imbalanced data, undersampling, big data, support vector machines.

I. INTRODUCTION

Dealing with imbalanced data is a crucial task in data mining studies. In particular, concerning the classification problems, most datasets in the real world do not contain the exact equal number of instances in each class, i.e., the classes are unequally represented, which can eventually cause significant problems while applying some algorithms.

In supervised learning, most classifiers are designed to achieve the best accuracy at the risk of being overwhelmed by an underlying class distribution [1], [2]. In the worst case, the resulting classifier becomes indiscriminate. i.e., it may be biased toward the majority class presented in the training set without having performed any feature analysis. This causes diverse ramifications based on properties of classifiers. For instance, in support vector machines (SVM), prediction performance can deteriorate owing to a) minority data that do not correspond to an ideal hyperplane, b) soft-margins invalidated by minority data, and c) support vectors dominated by majority data [3].

Among various techniques devised to address the imbalanced data issue, the resampling technique is a widely used data-level solution [4] that is generally achieved using two main approaches: undersampling and oversampling.

In the undersampling technique, instances from the majority class are eliminated to obtain a balanced training dataset. For example, random undersampling (RUS) randomly deletes instances in the majority class. However, such an approach can lead to information loss from the removed data points [2], [3]. To mitigate this side effect, Altmcay and Ergün [5] proposed the concept of cluster centroids (CC) based on adopting the k-means clustering approach. In this method, instances belonging to the majority data are grouped into a certain number of clusters (for example, as in an integrated framework of RUS and k-means clustering in [6]). To avoid the loss of potentially useful data, various heuristic undersampling methods have been proposed. Hart [7] formulated the condensed nearest neighbor (CNN) rule, and Wilson [8] introduced the concept of an edited nearest neighbor (ENN) by applying the k-NN approach to reduce the number of data points in the majority class. Similarly, Batista *et al.* [9] suggested a combination of CNN with Tomek Links [10]; in this approach, a learner first selects a subset of the majority class data, the Tomek Links method is then applied to this subset. Mani and Zhang [11] proposed multiple versions of the NearMiss method using the

The associate editor coordinating the review of this manuscript and approving it for publication was Senthil Kumar.

k-NN approach: a) NearMiss-1 generated a resampled dataset based on the mean distance from the minority class data to the k nearest points; b) NearMiss-2 that yields a resampled dataset with the mean distance to the k farthest points in the minority class data; and c) NearMiss-3 that selects k nearest neighbor points in the majority class data to the whole minority class data.

Conversely, in the oversampling technique, the instances from the minority class are duplicated to match the number of majority class instances. Random oversampling (ROS) is a typical method that randomly replicates the minority class data. Another widely used oversampling method is the synthetic minority oversampling technique (SMOTE) [12]. The basic step of the SMOTE procedure is to perform an interpolation among the neighboring minority class data to synthesize under-represented instances. The SMOTE method is considered as the standard oversampling framework to deal with imbalance datasets [13] (as in various applications using SMOTE reported in [14]–[17]).

Furthermore, algorithm-level solutions have also been proposed to address the imbalanced data issue. For instance, the cost-sensitive modeling, a popular regularization treatment, is broadly used to mitigate the class imbalance problem. In SVM, cost-sensitive SVM (CS-SVM) [18] uses differing costs considering an underlying class distribution of training data to control the sensitivity of misclassification (see the heuristic based CS-SVM proposed in [19]). Then, Lin and Wang [20] combined the fuzzy concept with SVM (F-SVM) where fuzzy membership of each input point was reformulated in SVM such that different inputs can provide different contributions to the construction of a hyperplane. Wu and Chang [21] modified the kernel function using the adaptive conformal transformation to modify the spatial resolution around the class boundary. Moreover, Li *et al.* [22] introduced an integrated framework combining AdaBoost and SVM (AdaSVM) to boost the accuracy of SVM on imbalanced data.

Although various approaches have been developed to cope with imbalanced data, limited attention has been paid to computational scalability. For large-scale imbalanced data, it is logical to use an undersampling method that not only adjusts the class distribution but also obtains a manageable training dataset. However, as shown in Fig. 1, the computational cost required for the undersampling process can become a critical concern as the data size increases.

Herein, a simple and effective undersampling method, referred to as particle stacking undersampling (PSU), was proposed, which can reduce the computational cost compared with other well-known undersampling methods, while minimizing the information loss to avoid a prediction bias. As elaborated in the following sections, this is enabled by achieving both data representability and peculiarity.

The remaining paper is organized as follows. Section 2 provides an overview of the key principles and computational procedures of the PSU algorithm. Section 3 presents the performance benchmarks for the proposed method against other resampling methods, both in terms of classification performance and processing time. Section 4 provides discussion focusing on the relation between the proposed principles and classification performance. Finally, Section 5 provides a summary of significant findings and future research directions.

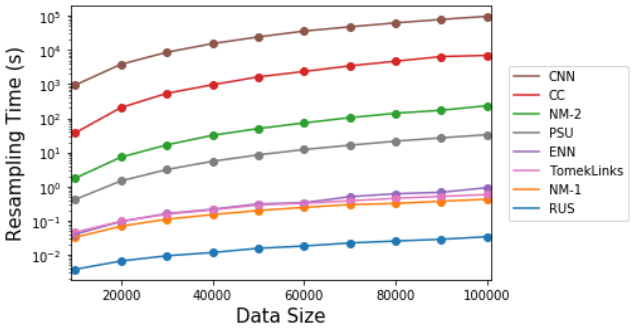

FIGURE 1. Relationship between the data size and resampling time concerning various undersampling methods on artificially generated two-dimensional data with the imbalance ratio of 10.

II. PARTICLE STACKING UNDERSAMPLING

A. PRELIMINARIES

As previously noted, the undersampling method can lead to information loss owing to the artificial removal of the majority class instances from the training set. This implies that data representability can be attained when the distribution of the original data is maintained in resampled data. To realize this, we establish the first proposition as follows.

Proposition 1: Information loss can be minimized if the sum of the distance between the resampled and original data is minimized.

However, data redundancy increases the computational complexity without improving the quality of information. Therefore, securing independence among resampled data points is desirable. This leads to the formulation of the second proposition.

Proposition 2: Information redundancy can be minimized if the sum of the distance among resampled data is maximized.

One may notice that the above propositions are consistent with the aim of the CC method but differ from that of borderline-oriented methods, such as Tomek Links. In fact, the latter emphasized more on the identification of majority data relevant to a decision boundary. This may be beneficial when classes are readily separable, otherwise susceptible to outlying data points. In highly imbalanced data, minority class data are often enclosed by the majority class data, which hinders the retention of the original distribution by relying on the borderline-oriented methods (see discussion regarding unintended outcomes from borderline-oriented methods in [23]). Furthermore, Tomek Links, by its nature, restrictively reduce the number of majority class data and therefore has a limited ability to balance the class distribution.

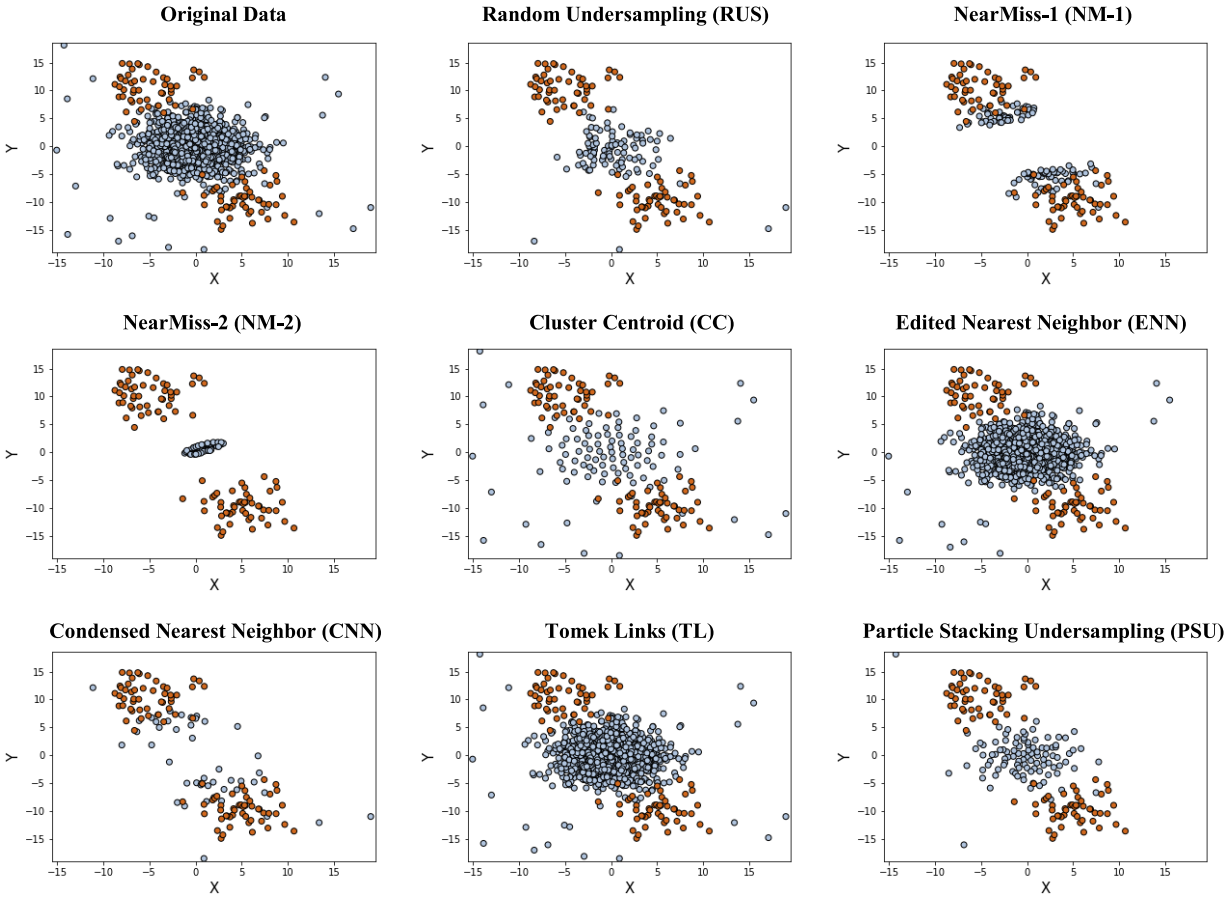

FIGURE 2. Comparison of various undersampling methods.

The CC method is prone to be affected by outlying data points and potentially converges with at locally optimal centroids. In particular, when outliers are sparsely distributed, centroids can be distorted; when they appear as a separate cluster, other clusters can be merged, both of which eventually deteriorate data representability. It is also well known that the k-means clustering method is sensitive to the choice of starting points; therefore, reproducible partitions are not always guaranteed [24]–[26]. Moreover, the CC method is severely impacted by time complexity that renders it unsuitable for large data [27], [27]–[29].

To address the aforementioned issues (see Fig. 2 for graphical insights), the proposed method focused on data representability and peculiarity while reducing the computational costs to ensure scalability to the mass of data.

B. ALGORITHMIC PROCEDURES

Concerning the two propositions and time complexity, the algorithmic procedures of PSU can be designed as presented in Algorithm 1. First, to attain data representability, data are split into multiple partitions based on the distance from the centroid of the majority class data. Each partition contains the equal number (m/n) of data points from which one sample is selected to represent the partition. Notably, PSU selects existing data points as samples; hence, it can better reflect the distribution of the original data and also

Algorithm 1 PSU

Input

a) Majority class data: $D^M = \{X_1, \ldots, X_m\}$

b) The number of minority class data points: n

Training Process

1. Calculate the centroid of majority class data: $C = (X_1 + X_2 + \ldots + X_m) / m$
2. Calculate L2-norm between C and majority class data: $D_2 = d_2(C, X_1), \ldots, d_2(C, X_m)$
3. Sort D_2 and group them into n partitions: $S = [s_1, \ldots, s_n]$
4. Set $\check{X}_1$ to be the last data point in s_1

   for $l = 2$ to n **do**

   5. Set $\check{X}_l$ to be the farthest data point in s_l in the resampled dataset: $\{\check{X}_1, \ldots, \check{X}_{n-1}\}$

   end for
6. Construct a resampled dataset: $D^R = \{\check{X}_1, \ldots, \check{X}_n\}$

Output

The resampled majority class data: D^R

save computational time compared with clustering-based methods.

When a sample is selected from a partition, the sample must be the farthest from other samples that are already selected. The aim of this criterion is to secure data peculiarity to the greatest possible extent because this prevents redundant data points from being included in the final sample set. This is also intended to facilitate data representability so that the data points from different clusters (if any) can be equally represented, even when they are included in the same partition. Moreover, it is possible for sparsely distributed data points that are largely dismissed by k-NN-based methods to be represented, unless the closely located samples are already selected. Finally, n majority data points are selected in such a way that the between-sample variation is maximized and the sample-to-original data variation is minimized.

PSU is intended to be a heuristic and deterministic undersampling method, i.e., the PSU method seeks a limited but representative set of majority data points with minimal operations, so that it can be applied to large data efficiently. Moreover, unlike the CC method, samples identified using the PSU method are reproducible, implying that the same unique solution can be obtained regardless of the experimental setting.

III. EXPERIMENTAL EVALUATION

In this section, seven well-known undersampling methods (RUS, NM-1, NM-2, CC, ENN, CNN, and Tomek Links) are compared with PSU by applying two popular kernels (linear and RBF) to SVM on 55 highly imbalanced datasets (imbalance ratio greater than 9) obtained from the KEEL repository [30] and, the comparison results are presented. Noted that 14 multi-class datasets were decomposed into 55 binary classification problems. Table 1 summarizes the description of these datasets.

The experiment was designed to perform 100 times repeated test for each dataset to further reduce variations in random splits (Fig. 3). In particular, the optimal parameters of the linear and RBF kernels, namely C and γ, were determined based on a five-fold cross-validation on the train set with respective undersampling methods. The classification performance of each fold was obtained by training SVM with the optimal parameters on the resampled train set using the same undersampling method that was used for the parameter optimization, and subsequently applying the trained model to test set. Note that the area under the curve (AUC) and geometric mean (G-mean) were used as performance measures as both of them were deemed as comprehensive and balanced metrics to better reflect the classification performance on imbalanced data [19], [31].

Table 2 summarizes the results of the experiment.[1] On average, in terms of both AUC and G-mean, CC achieved the most accurate classifiers followed by PSU and RUS that significantly outperformed other undersampling methods. However, the CC method had significantly larger time complexity than those of competing algorithms, except CNN. This was

[1]The results presented here can be reproduced at https://github.com/YongSeok-Jeon/IEEE.2020.PSU

TABLE 1. Description of the dataset.

Dataset	Number of instances	Number of attributes	Imbalance ratio
ecoli-0-3-4_vs_5	200	8	9.0
ecoli-0-6-7_vs_3-5	222	8	9.1
yeast-0-2-5-6_vs_3-7-8-9	1004	9	9.1
yeast-0-2-5-7-9_vs_3-6-8	1004	9	9.1
yeast-2_vs_4	514	9	9.1
yeast-0-3-5-9_vs_7-8	506	9	9.1
ecoli-0-2-3-4_vs_5	202	8	9.1
ecoli-0-4-6_vs_5	203	7	9.2
ecoli-0-3-4-6_vs_5	205	8	9.2
ecoli-0-1_vs_2-3-5	244	8	9.2
ecoli-0-2-6-7_vs_3-5	224	8	9.2
ecoli-0-3-4-7_vs_5-6	257	8	9.3
yeast-0-5-6-7-9_vs_4	528	9	9.4
ecoli-0-6-7_vs_5	220	7	10.0
vowel0	988	14	10.0
ecoli-0-1-4-7_vs_2-3-5-6	336	8	10.6
led7digit-0-2-4-5-6-7-8-9_vs_1	443	8	11.0
ecoli-0-1_vs_5	240	7	11.0
ecoli-0-1-4-7_vs_5-6	332	7	12.3
ecoli-0-1-4-6_vs_5	280	7	13.0
shuttle-c0-vs-c4	1829	10	13.9
yeast-1_vs_7	459	8	14.3
ecoli4	336	8	15.8
page-blocks-1-3_vs_4	472	11	15.9
abalone9-18	731	9	16.4
dermatology-6	358	35	16.9
yeast-1-4-5-8_vs_7	693	9	22.1
yeast-2_vs_8	482	9	23.1
flare-F	1066	12	23.8
car-good	1728	7	24.0
car-vgood	1728	7	25.6
kr-vs-k-zero-one_vs_draw	2901	7	26.6
kr-vs-k-one_vs_fifteen	2244	7	27.8
yeast4	1484	9	28.1
winequality-red-4	1599	12	29.2
kddcup-guess_passwd_vs_satan	1642	42	30.0
yeast-1-2-8-9_vs_7	947	9	30.6
yeast5	1484	9	32.7
kr-vs-k-three_vs_eleven	2935	7	35.2
abalone-17_vs_7-8-9-10	2338	9	39.3
yeast6	1484	9	41.4
winequality-white-3_vs_7	900	12	44.0
kddcup-land_vs_portsweep	1061	42	49.5
abalone-19_vs_10-11-12-13	1622	9	49.7
kr-vs-k-zero_vs_eight	1460	7	53.1
winequality-white-3-9_vs_5	1482	12	58.3
poker-8-9_vs_6	1485	11	58.4
shuttle-2_vs_5	3316	10	66.7
abalone-20_vs_8-9-10	1916	9	72.7
kddcup-buffer_overflow_vs_back	2233	42	73.4
kddcup-land_vs_satan	1610	42	75.7
kr-vs-k-zero_vs_fifteen	2193	7	80.2
poker-8-9_vs_5	2075	11	82.0
kddcup-rootkit-imap_vs_back	2225	42	100.1
abalone19	4174	9	129.4

TABLE 2. Performance benchmarks.

Kernel	Method	AUC	G-mean	Mean Rank (AUC)	Mean Rank (G-mean)	Train Ratio (%)	Resampling Time (s)
Linear	RUS	0.8508	0.8454	3.93	3.81	10.70	0.20
	NM-1	0.8005	0.7857	5.21	5.10	10.70	0.69
	NM-2	0.7918	0.7733	5.39	5.27	10.70	0.90
	CC	0.8588	0.8514	3.51	3.46	10.70	44.96
	ENN	0.8057	0.7425	4.65	4.82	95.97	5.75
	CNN	0.8101	0.7489	4.63	4.75	13.29	307.15
	Tomek Links	0.7879	0.7169	5.13	5.29	99.36	5.45
	PSU	0.8577	0.8505	3.55	3.49	10.70	0.61
RBF	RUS	0.8614	0.8557	4.08	3.98	10.70	0.20
	NM-1	0.8160	0.8022	5.34	5.14	10.70	0.69
	NM-2	0.7986	0.7819	5.73	5.55	10.70	0.90
	CC	0.8730	0.8670	3.57	3.50	10.70	44.96
	ENN	0.8337	0.7836	4.32	4.52	95.97	5.75
	CNN	0.8341	0.7937	4.47	4.64	13.29	307.15
	Tomek Links	0.8193	0.7539	4.75	5.00	99.36	5.45
	PSU	0.8673	0.8607	3.74	3.67	10.70	0.61

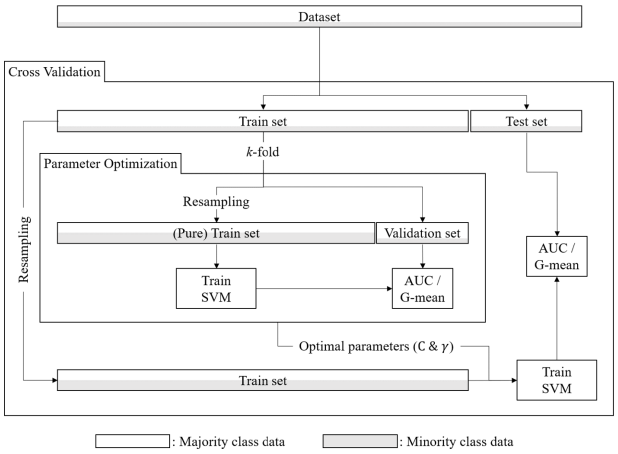

FIGURE 3. Experimental framework.

TABLE 3. Post-hoc test (Wilcoxon) results (p-value) compared with PSU.

Benchmark Methods	Linear		RBF	
	AUC	G-mean	AUC	G-mean
RUS	0.0000	0.0000	0.0000	0.0000
NM-1	0.0000	0.0000	0.0000	0.0000
NM-2	0.0000	0.0000	0.0000	0.0000
CC	0.9482	0.4899	0.0000	0.0000
ENN	0.0000	0.0000	0.0000	0.0000
CNN	0.0000	0.0000	0.0000	0.0000
Tomek Links	0.0000	0.0000	0.0000	0.0000

tolerable in the conducted experiment; however, when a large number of centroids need to be discovered in big data, the required computational load can become a critical drawback, as shown in Fig. 1. However, PSU achieved competitive resampling performance in a relatively short processing time, approximately, seventy times faster than CC.

To examine the statistical significance of the difference between the methods, the Friedman omnibus test [32] was first conducted on the rank values of classification performances for each undersampling method across the datasets. Consequently, the p-value was found to be less than the alpha risk of 0.05, indicating the existence of exceptional undersampling method(s). The Wilcoxon signed-rank test was then performed as a *post-hoc* analysis to facilitate the pairwise comparison of the undersampling methods with the adjusted alpha risk of 0.0017 ($\approx 0.05/28$) [33], [34].

Table 3 lists the p-values obtained from the *post-hoc* test; the value smaller than the adjusted alpha risk indicates that there existed a statistically significant difference between PSU and the corresponding benchmark method. Based on the result, it was confirmed that in the linear kernel, there was no dominance between PSU and CC, i.e., they equally yielded superior classification performance compared with the other methods in terms of both AUC and G-mean. However, in the RBF kernel, CC outperformed PSU, while they both maintained superiority to others. One possible interpretation of this is that the RBF kernel tends to map data to a higher dimensional space; thus, unlike the linear kernel, it can better handle the case when isolated centroids represent sparsely distributed data points, considering that some of them could contain important relations between classes. Finally, the aggressive identification of CC can be supplemented using the RBF mapping, which can also provide an opportunity to discover information from data. However, this is associated with the cost of additional computing resources

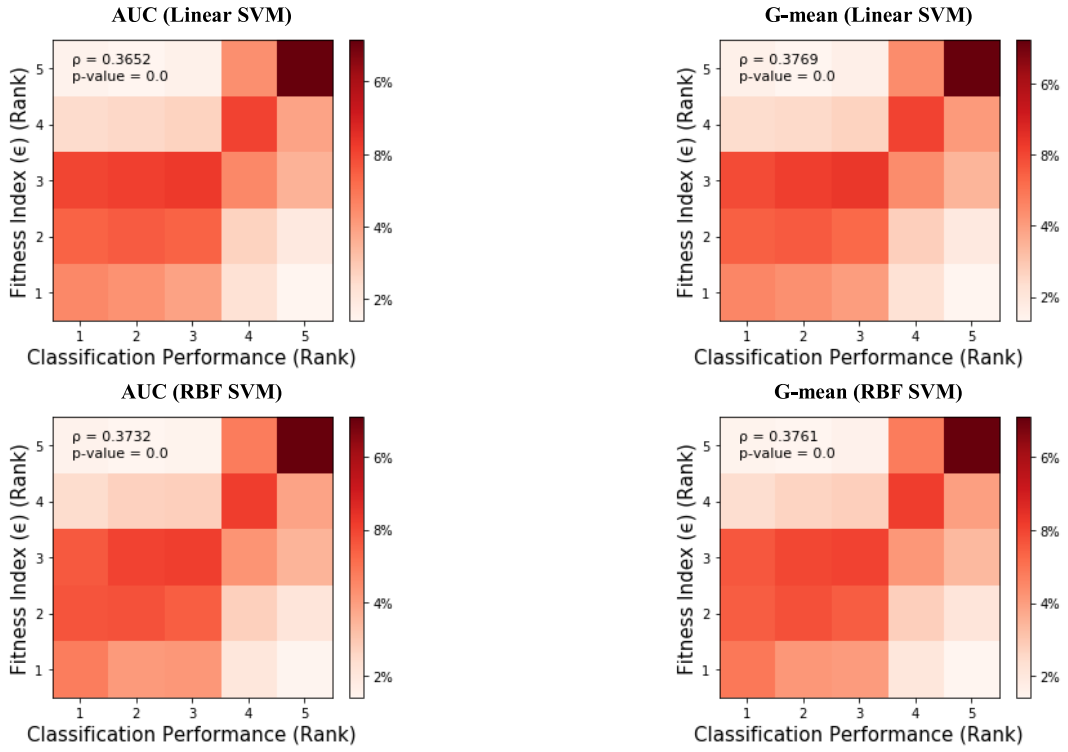

FIGURE 4. Relationships between the fitness index (ϵ) and the classification performance.

required for the complicated mapping and parameter optimization; in our experiment, the average execution time for the RBF kernel (40.85s) was three times more than that of the linear kernel (12.68s). Below, key implications of the experiment are summarized in three points:

1) CC and PSU outperformed the other methods; however, PSU was considerably more scalable, concerning that its time complexity was significantly lower than that of CC.
2) RBF-SVM in conjunction with CC may still be preferred if the processing time is tolerable, notwithstanding the high data complexity.
3) Borderline-oriented methods, such as ENN, CNN, and Tomek Links, were demonstrated to be relatively underperforming in terms of both resampling time and classification performance.

IV. DISCUSSION

In this section, a follow-up experiment was conducted to verify whether our propositions can serve as legitimate criteria in the undersampling practice. To enable comparing the extent to which the two propositions have been satisfied using different undersampling methods within a dataset, we introduce the fitness index (ϵ), which is defined as the sum of the distance between the resampled and original data divided by the sum of the distance among the resampled data. Note that by definition, a lower index corresponds to greater extent of satisfying the propositions using the resampling process.

An ordinal association between the fitness index rank and classification performance rank was investigated for each dataset and then collated to present the overall pattern (see Fig. 4). Note that we focused on five undersampling methods (RUS, CC, NM-1, NM-2, and PSU) because the borderline-oriented methods (ENN, CNN, and Tomek Links) were not intended to balance the number of major/minor classes. The obtained results indicated a statistically significant positive rank correlation: the Spearman rank-order correlation coefficient between the performance rank and fitness index rank was found to be greater than 0.35 with the p-value of 0.

Notably, the identified correlation was derived based on the five undersampling methods, i.e., when a lower fitness index was achieved using any method, it was likely that the resulting classification performance surpassed those of the other methods. This implied that the two formulated propositions serve as common principles for the five undersampling methods, and accordingly, they can be further applied when there is a need to achieve methodological improvements.

Generally, the PSU method lies in the middle between the sophisticated undersampling methods that may be precise yet not practically applicable to large data and the intuitive undersampling methods that are straightforward yet rely on arbitrary procedures without strictly formulated principles. It is therefore important to examine the nature of data to achieve the maximum effect of the proposed method. For instance, when a given dataset is separable without difficulty,

simpler methods can be preferred by providing high priority to the time complexity. However, when the construction of a representative set of virtual data points is desirable, more advanced methods may have to be used to handle the classification complexity.

V. CONCLUSION

In this paper, a simple and effective undersampling method, named PSU, was proposed. Compared with other competing undersampling methods, PSU can significantly reduce the computational cost, while minimizing information loss to avoid a prediction bias. This was achieved by realizing both data representability and peculiarity in the proposed algorithm. The performance benchmark indicated that the proposed method not only reached a competitive classification performance over other well-known undersampling methods but also provided a computational simplicity that can be scalable to large data. Further, we experimentally verified that two propositions that form the basis of the PSU algorithm can also be applied to other undersampling methods to achieve methodological improvements.

In practice, data are mostly imbalanced, and the computational cost required for proper resampling increases with the data size. To address this problem, we focused on a data-level undersampling solution; however, some algorithm-level solutions can achieve the same goal in other ways. In this regard, a hybrid approach can be considered to assess complementary interactions between resampling methods and characteristics of a classifier. In addition, a proper size of the partition and/or the number of samples to be drawn from each partition can be determined considering the distribution of a dataset. Lastly, the differential application of multiple approaches, including delicate resampling applied to the data located near the decision boundary, while aggressive resampling is applied to other data, can be implemented to further improve the efficiency and effectiveness of the resampling process.

REFERENCES

[1] N. Thai-Nghe, Z. Gantner, and L. Schmidt-Thieme, "Cost-sensitive learning methods for imbalanced data," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2010, pp. 1–8, doi: 10.1109/IJCNN.2010.5596486.
[2] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009, doi: 10.1109/TKDE.2008.239.
[3] R. Akbani, S. Kwek, and N. Japkowicz, "Applying support vector machines to imbalanced datasets," in *Proc. Eur. Conf. Mach. Learn.* Berlin, Germany: Springer, 2004, pp. 39–50.
[4] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, "A survey on addressing high-class imbalance in big data," *J. Big Data*, vol. 5, no. 1, p. 42, Nov. 2018, doi: 10.1186/s40537-018-0151-6.
[5] H. Altınçay and C. Ergün, "Clustering based under-sampling for improving speaker verification decisions using AdaBoost," in *Structural, Syntactic, and Statistical Pattern Recognition*, vol. 3138, A. Fred, T. M. Caelli, R. P. W. Duin, A. C. Campilho, and D. de Ridder, Eds. Berlin, Germany: Springer, 2004, pp. 698–706.
[6] Y.-P. Zhang, L.-N. Zhang, and Y.-C. Wang, "Cluster-based majority under-sampling approaches for class imbalance learning," in *Proc. 2nd IEEE Int. Conf. Inf. Financial Eng.*, Sep. 2010, pp. 400–404, doi: 10.1109/ICIFE.2010.5609385.
[7] P. Hart, "The condensed nearest neighbor rule (corresp.)," *IEEE Trans. Inf. Theory*, vol. 14, no. 3, pp. 515–516, May 1968.
[8] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-2, no. 3, pp. 408–421, Jul. 1972.
[9] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newslett.*, vol. 6, no. 1, pp. 20–29, Jun. 2004.
[10] I. Tomek, "Two modifications of CNN," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, no. 11, pp. 769–772, Nov. 1976, doi: 10.1109/TSMC.1976.4309452.
[11] I. Mani and I. Zhang, "KNN approach to unbalanced data distributions: A case study involving information extraction," in *Proc. Workshop Learn. Imbalanced Datasets*, vol. 126, 2003, pp. 1–7.
[12] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.
[13] A. Fernandez, S. Garcia, F. Herrera, and N. V. Chawla, "SMOTE for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary," *J. Artif. Intell. Res.*, vol. 61, pp. 863–905, Apr. 2018, doi: 10.1613/jair.1.11192.
[14] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: A new over-sampling method in imbalanced data sets learning," in *Proc. Int. Conf. Intell. Comput.*, 2005, pp. 878–887.
[15] K. Li, W. Zhang, Q. Lu, and X. Fang, "An improved SMOTE imbalanced data classification method based on support degree," in *Proc. Int. Conf. Identificat., Inf. Knowl. Internet Things*, Oct. 2014, pp. 34–38, doi: 10.1109/IIKI.2014.14.
[16] E. Ramentol, Y. Caballero, R. Bello, and F. Herrera, "SMOTE-RSB*: A hybrid preprocessing approach based on oversampling and undersampling for high imbalanced data-sets using SMOTE and rough sets theory," *Knowl. Inf. Syst.*, vol. 33, no. 2, pp. 245–265, Nov. 2012, doi: 10.1007/s10115-011-0465-6.
[17] J. A. Sáez, J. Luengo, J. Stefanowski, and F. Herrera, "SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering," *Inf. Sci.*, vol. 291, pp. 184–203, Jan. 2015, doi: 10.1016/j.ins.2014.08.051.
[18] G. Fumera and F. Roli, "Cost-sensitive learning in support vector machines," in *Proc. 8th Convegno Associazione Italiana Per L'Intelligenza Artificiale (AIIA)*, Siena, Italy, Sep. 2002.
[19] P. Cao, D. Zhao, and O. Zaiane, "An optimized cost-sensitive SVM for imbalanced data learning," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*. Berlin, Germany: Springer, 2013, pp. 280–292.
[20] C.-F. Lin and S.-D. Wang, "Fuzzy support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 464–471, Mar. 2002, doi: 10.1109/72.991432.
[21] G. Wu and E. Y. Chang, "Adaptive feature-space conformal transformation for imbalanced-data learning," in *Proc. 20th Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 816–823.
[22] X. Li, L. Wang, and E. Sung, "AdaBoost with SVM-based component classifiers," *Eng. Appl. Artif. Intell.*, vol. 21, no. 5, pp. 785–795, Aug. 2008, doi: 10.1016/j.engappai.2007.07.001.
[23] J. Ha and J.-S. Lee, "A new under-sampling method using genetic algorithm for imbalanced data classification," in *Proc. 10th Int. Conf. Ubiquitous Inf. Manage. Commun. (IMCOM)*, 2016, pp. 1–6.
[24] K. A. Nazeer and M. P. Sebastian, "Improving the accuracy and efficiency of the k-means clustering algorithm," in *Proc. World Congr. Eng.*, vol. 1, 2009, pp. 1–3.
[25] K. Arai and A. R. Barakbah, "Hierarchical K-means: An algorithm for centroids initialization for K-means," *Rep. Fac. Sci. Eng.*, vol. 36, no. 1, pp. 25–31, 2007.
[26] M. Yedla, S. R. Pathakota, and T. M. Srinivasa, "Enhancing K-means clustering algorithm with improved initial center," *Int. J. Comput. Sci. Inf. Technol.*, vol. 1, no. 2, pp. 121–125, 2010.
[27] G. Frahling and C. Sohler, "A fast k-means implementation using coresets," *Int. J. Comput. Geometry Appl.*, vol. 18, no. 6, pp. 605–625, Dec. 2008.
[28] J. Z. C. Lai, T.-J. Huang, and Y.-C. Liaw, "A fast -means clustering algorithm using cluster center displacement," *Pattern Recognit.*, vol. 42, no. 11, pp. 2551–2556, Nov. 2009.
[29] M. Shindler, A. Wong, and A. W. Meyerson, "Fast and accurate k-means for large datasets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2375–2383.
[30] J. Alcalá-Fdez, A. Fernández, J. Luengo, J. Derrac, and S. García, "KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Multiple-Valued Logic Soft Comput.*, vol. 17, nos. 2–3, pp. 255–287, 2011.

[31] J. Akosa, "Predictive accuracy: A misleading performance measure for highly imbalanced data," in *Proc. SAS Global Forum*, 2017, pp. 2–5.
[32] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Jan. 2006.
[33] A. Benavoli, G. Corani, and F. Mangili, "Should we really use post-hoc tests based on mean-ranks?" *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 152–161, 2016.
[34] R. A. Armstrong, "When to use the Bonferroni correction," *Ophthalmic Physiol. Opt.*, vol. 34, no. 5, pp. 502–508, Sep. 2014.

YONG-SEOK JEON received the bachelor's degree in systems management engineering from Sungkyunkwan University, South Korea. He is currently a Junior Researcher of the Industrial Engineering Department, Sungkyunkwan University. His current research interests include tree-based ensemble modeling, optimization modeling, support vector machines, heuristic modeling, and big data.

DONG-JOON LIM received the B.S. and M.S. degrees in industrial engineering from Sungkyunkwan University, South Korea, and the Ph.D. degree in engineering and technology management from Portland State University, Portland, OR, USA. He is currently an Assistant Professor with the Department of Systems Management Engineering, Sungkyunkwan University. His current research interests include technological forecasting, optimization modeling, productivity analysis, and data mining. He is also a developer of an open-source R package "DJL" which implements various decision support tools related to econometrics and technometrics. His academic honors include the Emerald Literati Network Award (outstanding author), the ENI Award (finalist for renewable and non-conventional energy), the Marie Brown Award, and various fellowships from PSU, SKKU, and A&P, among others.