Balancing Data Within Incremental Semi-supervised Fuzzy Clustering for Credit Card Fraud Detection

*Gabriella Casalino[a] and Giovanna Castellano[a] and Nicola Marvulli[a]
[a]Department of Computer Science, University of Bari,
Via E. Orabona 4, Bari, Italy, name.surname@uniba.it

Abstract

As the number of online financial transactions increases, the problem of credit card fraud detection has become quite urgent. Machine learning methods, including supervised and unsupervised approaches, have been proven to be effective to detect fraudulent activities. In our previous work presented at EUSFLAT2019 we proposed the use of an incremental semi-supervised fuzzy clustering that processes both labeled and unlabeled data as a stream to create a classification model for credit card fraud detection. However, we observed that the results of the method were affected by data unbalancement. Indeed credit card fraud data are highly imbalanced since the number of fraudulent activities is far less than the genuine ones. In this work, to deal with the high data unbalance, different resampling methods are investigated and their empirical comparison is reported.

Keywords: Web Economy, Cyber Security, Credit Card Fraud Detection, Stream Data Mining, Semi-supervised learning, Resampling algorithms.

1 Introduction

The integration of information technology solutions in the financial field, such as in many other domains, has led to a massive transformation over the recent years. The term *Web Economy* is used to refer to the digital transformation of the economy that has changed the way businesses are structured and consumers access services, information, and goods [25]. Payment transactions are one of the most affected fields, indeed digital payments have seen exponential growth in the last few years. The total value of digital payments in 2020 was estimated at USD $4,934,741$ million [1].

If from one hand digital transformation has resulted in many positive effects such as enhancement of productivity, reduction of distances, customized services, from the other hand, cyber crimes have spread exponentially. The number of victims of identity fraud in 2018 is about 14.4 million, according to the *Identity Fraud Study* by Javelin Strategy and Research [2]. Among several kinds of cybercrimes in the economic domain, digital frauds on credit card transactions are the most common, due to the rise of the use of credit cards for digital payments for both online and physical shopping.

Thus fraud management has become a critical factor for the banking and commerce industries. It involves three activities, namely, prevention, detection, and resolution. In this work, we focus on the second one. Once a fraud has occurred, the aim is to detect it among the majority of genuine transactions. However, according to Fraud Benchmark Report [3] more than 80% of the North American businesses still conducts manual reviews, just on a low percentage of orders (about 30%). Manual screenings are conducted for discriminating genuine from fraudulent activities. But a manual review is time expensive, prohibitive for a big amount of data and, could lead to a high rate of both false negatives and false positives.

As a result, machine learning algorithms have been ef-

[1] Reports Statista: https://www.statista.com/outlook/dmo/fintech/digital-payments/worldwide

[2] 2019 Identity Fraud Study: https://www.javelinstrategy.com/coverage-area/2019-identity-fraud-report-fraudsters-seek-new-targets-and-victims-bear-brunt

[3] 2016 Fraud Benchmark Report: https://www.cybersource.com/content/dam/cybersource/NA_2016_Fraud_Benchmark_Report.pdf (accessed January 2017)

fectively used for a first screening of the whole data, and then, only when suspicious frauds are detected, human experts are involved in the analysis for review. When applied to fraud detection, machine learning methods have been proved to be fast, able to scale, and efficient [20]. Learning approaches for fraud detection can be divided into supervised and unsupervised ones [24]. The first learns the customer model from historical transactions, and then predictions are performed on new data. Even if supervised methods have been proved to be effective [14], they require labeled data that, as discussed before, are time-consuming and labor-intensive to obtain. For this reason, unsupervised algorithms, that do not use a-priori information, are often used to perform *anomaly detection* tasks. Using unsupervised approaches, fraudulent transactions are selected when changes are detected in the customers' normal behavior. Unsupervised methods are also able to discover unseen types of frauds since they do not rely on past transactions. However, completely unsupervised methods can not exploit available expert knowledge that can be injected into the model by labeling transactions as fraud/non-fraud.

Hybrid or semi-supervised methods can embed the benefits of both supervised and unsupervised approaches in the task of credit card fraud detection [5]. Indeed they combine the information coming from the few available data, and from the unknown patterns hidden in data.

Since credit card frauds financially affect both customers and organizations, almost real-time analyses are required to limit the damages as soon as possible [29, 4]. In this context, stream data algorithms have been proven to be effective, since they are able to process data as they are available and to adapt the model to the new incoming data [19]. In [7] we proposed the use of an incremental semi-supervised clustering method for credit card fraud detection. The results showed that the method, called DISSFCM (Dynamic Incremental Semi-Supervised FCM) can be an effective technique to recognize fraudulent activities in credit card transaction data streams. However, data unbalancement affects the results, since data belonging to the minority class are difficult to detect.

In this paper, we extend the work presented in [7] by applying resampling algorithms to the data stream. To this aim, a benchmark dataset for credit card fault detection has been used, and different resampling methods have been combined with DISSFCM and the results have been compared. The rest of this paper is structured as follows. Section 2 briefly describes DISSFCM and the resampling methods that have been compared. Results are discussed in section 3 and conclusions and future works are detailed in section 4.

2 Methods

In this work Dynamic Incremental Semi-Supervised Fuzzy C-Means (DISSFC) algorithm has been combined with several resampling methods for unbalanced data. DISSFCM is a classification algorithm, that sequentially analyses subsets of data (chunks), that can be partially labeled. Indeed, it is based on the Semi-supervised fuzzy C-Means (SSFCM) proposed by Pedrycz, [26]. At each time T, the current chunk is analyzed by SSFCM, and labeled prototypes, together with hard cluster assignments are returned. The clustering process is based on a minimization process of the objective function in eq. 1, that from one hand is able to optimize the model according to the geometrical structure hidden in data (unsupervised approach), and from the other hand embeds the *a-priori* information about the known classes in the learning process.

$$J = \sum_{k=1}^{K} \sum_{j=1}^{N_t} u_{jk}^2 d_{jk}^2 + \alpha \sum_{k=1}^{K} \sum_{j=1}^{N_t} \left(u_{jk} - b_j f_{jk}\right)^2 d_{jk}^2 \quad (1)$$

where $K \geq C$ is the number of clusters, $N_t = |X_t|$ is the cardinality of the t-th chunk in the data stream, $u_{jk} \in [0,1]$ is the membership degree of a sample $\mathbf{x}_j$ in the k-th cluster, d_{jk} is the Euclidean distance between jth sample and center $\mathbf{c}_k$ of the k-th cluster, $\alpha \geq 0$ is a regularization parameter for the second part of the objective function that exploits class information, $b_j = b(\mathbf{x}_j)$ and $f_{jk} = 1$ iff the j-th sample belongs has the same class label of the k-th cluster. For each cluster center $\mathbf{c}_k$, a prototype $\mathbf{p}_k$ is derived as a medoid.

Moreover, fuzzy clustering has been preferred to hard clustering, due to its capability to better represent changes in data, which is a critical factor for stream data [1]. Indeed, for this reason, several extensions of fuzzy clustering algorithms have been proposed for data stream [10, 17, 27].

A first incremental version of the Semi-supervised fuzzy C-Means (ISSFCM), was proposed in [6], where historical information on previous data was injected from one chunk to the subsequent, through the labeled prototypes. This algorithm has then been extended, with its dynamic and adaptive version (DISSFCM), which is able to modify the model if it is no more able to correctly represent the original data, by increasing the number of clusters [8]. To this aim reconstruction error is used to quantitatively evaluate the goodness of the model, and a split mechanism is activated to increase the model granularity, for a better representation of the data in the current chunk.

Since DISSFCM analyses on chunk per time, resampling algorithms have been applied to the current

Oversampling	**Undersampling**	**Hybrid**
Oversampling	Undersampling	SMOTEENN [3]
SMOTE [9]	CondensedNearestNeighbour [13]	SMOTETomek [2]
BorderlineSMOTE [12]	EditedNearestNeighbours [30]	
KMeansSMOTE [11]	RepeatedEditedNearestNeighbours [28]	
RandomOverSampler [22]	AllKNN [28]	
SVM-SMOTE [23]	NearMiss [21]	
	NeighbourhoodCleaningRule [18]	
	OneSidedSelection [16]	
	RandomUnderSampler	
	TomekLinks [28]	
	ClusterCentroids	

Table 1: Resampling methods.

data. Resampling algorithms can be grouped in three main classes: oversampling, undersampling and hybrid methods [15]. Oversampling methods use data augmentation to add samples to the minority class; conversely, undersampling methods reduce the number of samples belonging to the majority class. Hybrid methods combine both oversampling and undersampling to obtain balanced classes.

In this works different oversampling, undersampling and hybrid methods have been compared in order to find the best pre-processing technique for unbalanced stream classification with DISSFCM. Imbalanced learn Python toolbox has been used for algorithm implementations [4] and default values, for the algorithm parameters, have been used. Table 1 lists the adopted algorithms for each category and their references.

3 Results

A set of experiments has been conducted in order to compare the effectiveness of different resampling methods in improving the classification performance of DISSFCM, when dealing with strongly unbalanced data. To this aim, we considered a dataset containing the transactions made by European credit cards[5]. It contains sequential transactions that occurred in two days, and it is an example of a highly unbalanced dataset. Indeed, as it can be expected, the positive class (*frauds*) accounts for 0.172% of all transactions (i.e., 492 fraudulent activities out of $284,807$ normal transactions) [5]. The *fraud* class (i.e. the target class) is identified by 1 and the *normal* class by 0. Principal Component Analysis (PCA) was used to anonymize the 28 features describing each transaction, whilst time and amount of expense are clear.

Since transactions arrive continuously, on one hand, it is crucial to detect possible fraudulent activities, as soon as possible, on the other hand, processing data as they arrive is computationally expensive. Thus, in order to simulate a real scenario, we considered an interval of twelve hours as processing units. Data has been split into four chunks, containing sequential transactions, that have been sequentially evaluated through DISSFCM.

For each chunk, 70% of samples were considered as training set and the remaining 30% as test set. Since in stream scenarios, data arriving at time $t+1$ are assumed to be unknown at time t, resampling algorithms have been applied to each chunk, separately, and, training sets only have been resampled. Finally, for a fair comparison, the same samples in each chunk have been considered for training and test sets.

Moreover, since DISSFCM is a semi-supervised classification algorithm, the influence of the labeling percentage on the classification performance has been studied by considering four different labeling ratios, that is 25%, 50%, 75%, and 100%.

Standard classification measures, such as accuracy, precision, recall, and F1-score have been used to evaluate the algorithm performances in detecting the fraudulent class.

Figures 1- 4 visualize the average classification performance of DISSFCM on the four chunks, varying the labeling percentage and the resampling methods. The first row reports values obtained without applying any resampling method. Since the aim of the classification problem is to correctly identify the fraudulent activities, precision, recall, and F1-score values of the sole target class (*frauds*) are reported in the figures.

[4] Imbalanced learn toolbox: `https://imbalanced-learn.org/stable/`

[5] Dataset: `https://www.kaggle.com/mlg-ulb/creditcardfraud`

Moreover, a color scale has been used to easily identify the values of each measure. Red color corresponds to 0 values, and blue to the maximum value 1. Different shades are used for values in the interval $(0,1)$. This representation helps to highlight low and high values and to compare the different resampling methods, varying the number of total labels in data.

While accuracy and F1-score return an overall evaluation of the classification performance, in the case of fraud detection major attention should be given to precision and recall measures. Indeed precision indicates the ratio of relevant items among all those returned $(\frac{TP}{TP+FP})$ whilst recall indicates the ratio of relevant items that have been identified among all the relevant items $(\frac{TN}{TP+FN})$. In this context TP is the true positive value (fraudulent sample correctly classified); TN is the true negative value (genuine activity correctly classified); FP is the false positive value (genuine sample that has been incorrectly assigned to the fraudulent class) and FN is the false negative value (fraudulent activity that has been incorrectly classified as genuine).

Either FP and FN should be avoided since they cause problems for both the bank and the client. But, when an FP occurs, the emergency mode is activated, and perhaps, the customer's bank account is temporarily blocked, until further checks are performed. However, even this is an annoying situation for the customer, and for this reason, it should be avoided and the number of FP should be minimized, in this case, there is no loss of money. On the contrary, when an FN occurs, fraudulent activity is entered into the system, and it has not been recognized. This is a very critical scenario for the bank and the customer, worse than the previous one. For this reason, in the following analysis, special attention will be given to precision and recall measures.

The first row of Figures 1- 4 reports the baseline scenario, where no resampling algorithms have been applied. Red squares for precision and recall suggest that the DISSFCM algorithm is not able to correctly detect fraudulent activities. In this case, labeled data do not help to improve the results. Several algorithms return similar results, such as TomekLinks, RepeatedEditedNearestNeighbours, OneSidedSelection, NeighbourhoodCleaningRule, EditedNearestNeigbours and AllKNN. Even with 100% labeling, they are not able to detect fraudulent activities. Other algorithms, such as SVMSMOTE and CondensedNearestNeighbour return high values for recall (i.e., most of the fraudulent activities are identified) and medium values for precision (i.e., several genuine activities have been wrongly reported as frauds) for 100% labeling, but they are influenced by the percentage of available labels. The best results have been returned by ClusterCentroids and KMeansSMOTE. Indeed for 100% labeling, they both collect very high values of recall (0.99), that means no false negatives have been returned, and all the true positives have been identified. This is a noticeable result. On the other hand, precision values, for both the algorithms, are close to 0.6 that is a quite good value. However, this means that some false positives have been returned together with the true positive. Furthermore, it is worth noting that these results are stable, varying the labeling percentage. Even with a low labeling percentage (e.g. 25%) DISSFCM is able to detect all the fraudulent activities on data that have been resampled by ClusterCentroids, and also KMeansSMOTE returns quite good results.

It is interesting to note that one oversampling method and one undersampling method, both of them based on the K-means algorithm, have returned the best values. This result is not surprising. Indeed KMeansSMOTE applies K-means clustering on the entire input space and then distributes the number of new synthetic samples by using SMOTE oversampling method across clusters. On the other hand, ClusterCentroids under samples the majority class by replacing subsets of samples, grouped with K-means algorithm, with their respective cluster centroids. Thus, resampled data presents a geometrical structure that well fits with the fuzzy c-means clustering algorithm, DISSFCM is based on. However, these are preliminary results. Further investigations are required, to verify this behavior on other synthetic and real unbalanced data.

4 Conclusion

Credit card fraud detection is a critical task for banks and companies, since with the money dematerialization, credit card usage has strongly increased, together with fraudulent activities on physical and online customers' transactions. This big amount of daily generated data has made unusable manual analysis of these transactions. Machine learning algorithms have been used in literature to address this problem. Both supervised and unsupervised algorithms have been proved to be effective, in different ways, in detecting fraudulent activities. However, semi-supervised algorithms are more suitable for dealing with this kind of data, that have a limited amount of labels. Moreover, high unbalancing affects these data, since fraudulent activities are much rarer than genuine ones. Finally, almost real-time analysis is necessary in order to detect as soon as possible, eventual frauds, by avoiding a computational overload.

In this work, several resampling methods have been combined with DISSFCM, a Dynamic Incremental Semi-Supervised classification algorithm based on Fuzzy C-Means. The best results have been obtained

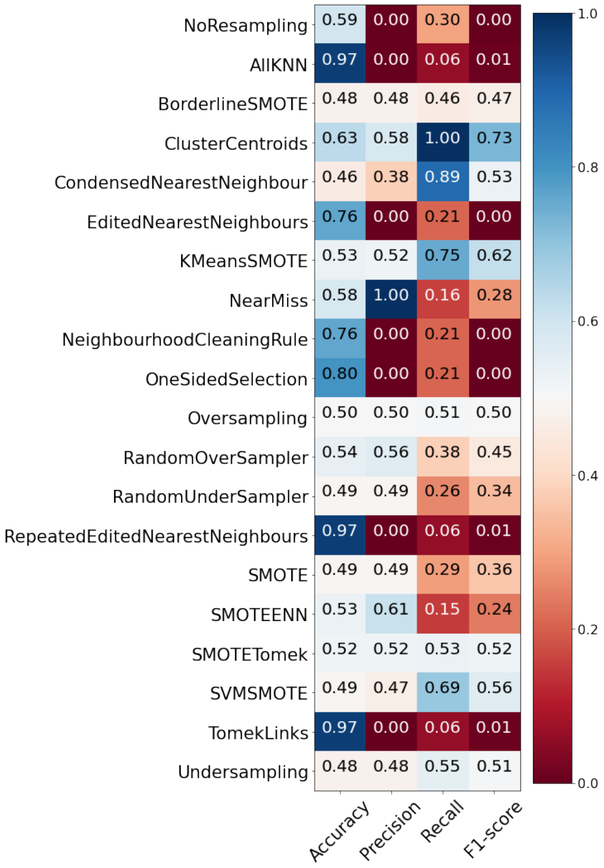

Figure 1: Classification performance of the target class fraudulent - 25% labeling.

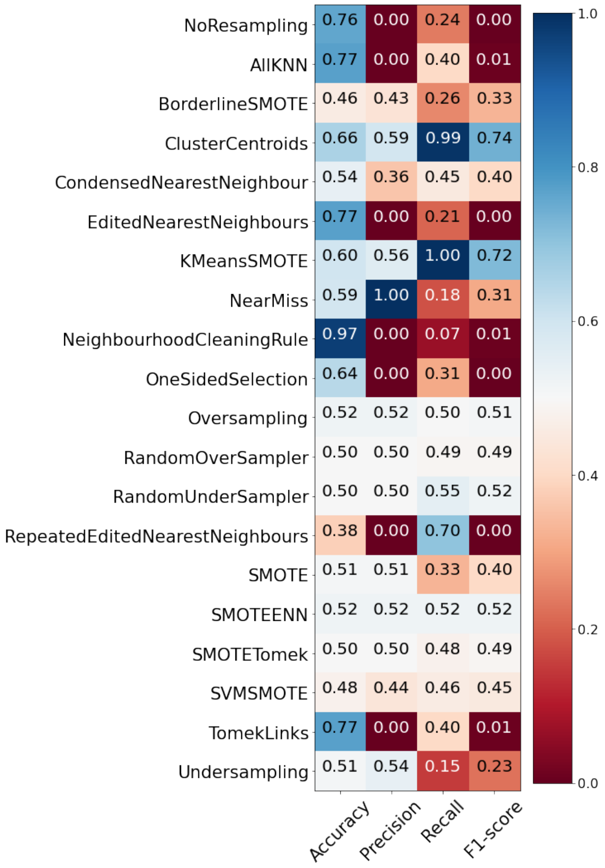

Figure 2: Classification performance of the target class fraudulent - 50% labeling.

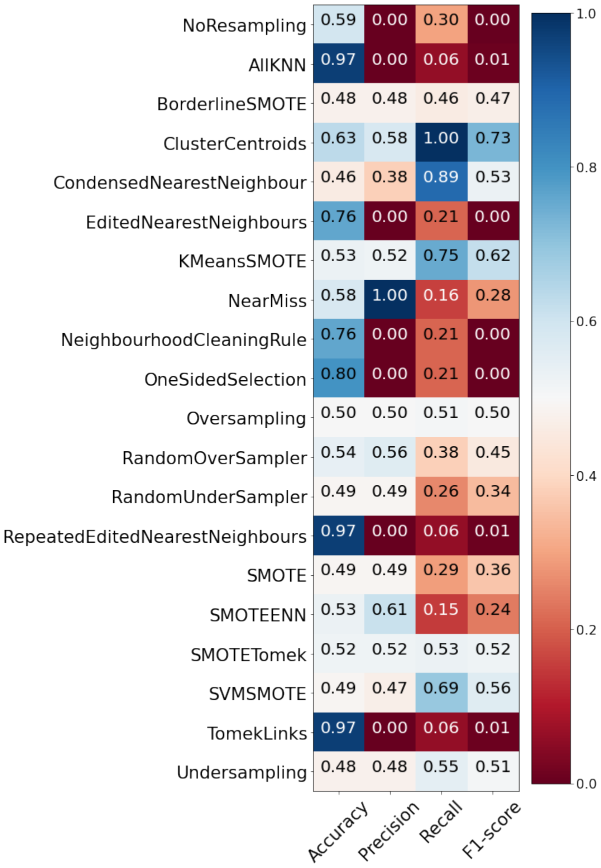

Figure 3: Classification performance of the target class fraudulent - 75% labeling.

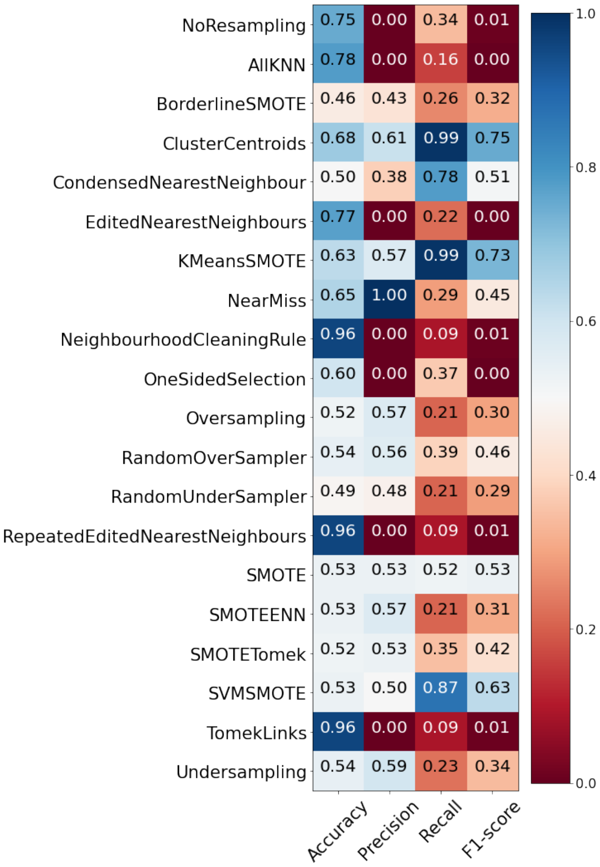

Figure 4: Classification performance of the target class fraudulent - 100% labeling.

by resampling the training data with two algorithms based on K-means clustering. Particularly the oversampling method KMeansSMOTE and the undersampling method ClusterCentroids had been able to detect almost all the fraudulent activities (recall=0.99), thus avoiding false negatives, and also quite good precision values have been returned (0.6). Furthermore, they have been proven to be robust, since labeling percentage had not affected too much the results. Indeed, with 25% of labels, ClusterCentroids had been still able to detect all the frauds, and KMeansSMOTE performances had been slightly affected. These are encouraging results that suggest the necessity of integrating resampling methods within DISSFCM. Further analyses on different unbalanced datasets are necessary to better study the influence of the resampling methods on DISSFCM results.

Acknowledgement

Gabriella Casalino acknowledges funding from the Italian Ministry of Education, University and Research through the European PON project AIM (Attraction and International Mobility), nr. 1852414, activity 2, line 1. G. Casalino and G. Castellano are members of the INdAM GNCS research group. G. Casalino and G. Castellano are with the CITEL - Centro Interdipartimentale di Telemedicina, University of Bari Aldo Moro.

References

[1] A. Abdullatif, F. Masulli, S. Rovetta, Clustering of nonstationary data streams: A survey of fuzzy partitional methods, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8 (4) (2018) e1258.

[2] G. E. Batista, A. L. Bazzan, M. C. Monard, Balancing training data for automated annotation of keywords: a case study., in: WOB, 2003, pp. 10–18.

[3] G. E. Batista, R. C. Prati, M. C. Monard, A study of the behavior of several methods for balancing machine learning training data, ACM SIGKDD explorations newsletter 6 (1) (2004) 20–29.

[4] F. Carcillo, Y.-A. Le Borgne, O. Caelen, G. Bontempi, Streaming active learning strategies for real-life credit card fraud detection: assessment and visualization, International Journal of Data Science and Analytics 5 (4) (2018) 285–300.

[5] F. Carcillo, Y.-A. Le Borgne, O. Caelen, Y. Kessaci, F. Oblé, G. Bontempi, Combining unsupervised and supervised learning in credit card fraud detection, Information Sciences.

[6] G. Casalino, G. Castellano, C. Mencar, Incremental adaptive semi-supervised fuzzy clustering for data stream classification., in: Proc. of the 2018 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS 2018), 2018, pp. 1–7.

[7] G. Casalino, G. Castellano, C. Mencar, Credit card fraud detection by dynamic incremental semi-supervised fuzzy clustering, in: Atlantis Studies in Uncertainty Modelling, 11th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT 2019), Atlantis Press, 2019, pp. 198–204.

[8] G. Casalino, G. Castellano, C. Mencar, Data stream classification by dynamic incremental semi-supervised fuzzy clustering, International Journal on Artificial Intelligence Tools 28 (08) (2019) 1960009.

[9] N. V. Chawla, K. W. Bowyer, L. O. Hall, W. P. Kegelmeyer, Smote: synthetic minority oversampling technique, Journal of artificial intelligence research 16 (2002) 321–357.

[10] P. Ducange, F. Marcelloni, R. Pecori, Fuzzy hoeffding decision tree for data stream classification, International Journal of Computational Intelligence Systems.

[11] D. Georgios, B. Fernando, L. Felix, Oversampling for imbalanced learning based on k-means and smote, Inf. Sci. 465 (2018) 1–20.

[12] H. Han, W.-Y. Wang, B.-H. Mao, Borderline-smote: a new over-sampling method in imbalanced data sets learning, in: International conference on intelligent computing, Springer, 2005, pp. 878–887.

[13] P. Hart, The condensed nearest neighbor rule (corresp.), IEEE transactions on information theory 14 (3) (1968) 515–516.

[14] S. Khatri, A. Arora, A. P. Agrawal, Supervised machine learning algorithms for credit card fraud detection: a comparison, in: 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), IEEE, 2020, pp. 680–683.

[15] S. Kotsiantis, D. Kanellopoulos, P. Pintelas, et al., Handling imbalanced datasets: A review, GESTS International Transactions on Computer Science and Engineering 30 (1) (2006) 25–36.

[16] M. Kubat, S. Matwin, et al., Addressing the curse of imbalanced training sets: one-sided selection, in: Icml, Vol. 97, Citeseer, 1997, pp. 179–186.

[17] S. Laohakiat, V. Sa-ing, An incremental density-based clustering framework using fuzzy local clustering, Information Sciences 547 (2021) 404–426.

[18] 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.

[19] D. Leite, I. Škrjanc, F. Gomide, An overview on evolving systems and learning from stream data, Evolving Systems (2020) 1–18.

[20] Y. Lucas, J. Jurgovsky, Credit card fraud detection using machine learning: A survey, arXiv preprint arXiv:2010.06479.

[21] I. Mani, I. Zhang, knn approach to unbalanced data distributions: a case study involving information extraction, in: Proceedings of workshop on learning from imbalanced datasets, Vol. 126, 2003.

[22] G. Menardi, N. Torelli, Training and assessing classification rules with imbalanced data, Data mining and knowledge discovery 28 (1) (2014) 92–122.

[23] H. M. Nguyen, E. W. Cooper, K. Kamei, Borderline over-sampling for imbalanced data classification, International Journal of Knowledge Engineering and Soft Data Paradigms 3 (1) (2011) 4–21.

[24] X. Niu, L. Wang, X. Yang, A comparison study of credit card fraud detection: Supervised versus unsupervised, arXiv preprint arXiv:1904.10604.

[25] S. Paiva, M. A. Ahad, G. Tripathi, N. Feroz, G. Casalino, Enabling technologies for urban smart mobility: Recent trends, opportunities and challenges, Sensors 21 (6) (2021) 2143.

[26] W. Pedrycz, Algorithms of Fuzzy Clustering with Partial Supervision, Pattern Recogn. Lett. 3 (1) (1985) 13–20.
URL http://dx.doi.org/10.1016/0167-8655(85)90037-6

[27] R. Tabbussum, A. Q. Dar, Comparison of fuzzy inference algorithms for stream flow prediction, Neural Computing and Applications 33 (5) (2021) 1643–1653.

[28] I. Tomek, Two modifications of cnn, In Systems, Man, and Cybernetics, IEEE Transactions on 6 (2010) 769–772.

[29] P. H. Tran, K. P. Tran, T. T. Huong, C. Heuchenne, P. HienTran, T. M. H. Le, Real time data-driven approaches for credit card fraud detection, in: Proceedings of the 2018 international conference on e-business and applications, 2018, pp. 6–9.

[30] D. L. Wilson, Asymptotic properties of nearest neighbor rules using edited data, IEEE Transactions on Systems, Man, and Cybernetics (3) (1972) 408–421.