Received: October 6, 2019. Revised: January 10, 2020.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Effectiveness Undersampling Method and Feature Reduction in Credit Card Fraud Detection

Dedy Trisanto[1]* **Nofita Rismawati[2]** **Muhamad Femy Mulya[3]** **Felix Indra Kurniadi[3]**

[1] *Polytechnic STMI Jakarta, Indonesia*
[2] *Indraprasta PGRI University, Indonesia*
[3] *Tanri Abeng University, Indonesia*
* Corresponding author's Email: d-trisanto@kemenperin.go.id

Abstract: Credit card fraud is an issue that has affected Indonesia payment system over a decade. Sometimes, the result of the fraud used for terrorism and other crimes. Financial loss is not the only problem that is affected caused by credit card fraud but also Indonesia images in international trade, e-commerce, and the merchant. Currently, a trusted and secured banking payment system is crucial for both customers and banks. The problem from credit card fraud dataset is the data have many features and imbalanced class, this problem leads the paper to propose undersampling technique and feature reduction methods. In this paper we proposed two stage-feature reduction technique because a stage feature reduction could not find the optimal features. On the other hands, we are also applied Instance Hardness Threshold sampling and Random undersampling to deal with imbalance data. The two-stage feature reduction is chosen to eliminate the ineffective feature that cannot eliminate using only one feature reduction. The model from the implemented machine learning methods is evaluated using accuracy, specificity, recall, and Matthews Correlation Coefficient. We implemented our proposed approaches in the ULB credit card fraud detection dataset. According to the result, the undersampling gives a boost in performances which improve the recall and MCC score, the IHT undersampling provide goods results, and in some cases, the result can predict all the test set correctly. However, the two-stages feature reduction fails to improve the accuracy, precision, recall, and MCC score. In one case, the method reduced the accuracy score to 0.302.

Keywords: Fraud detection, Two-stages features reduction, Under-sampling, Huber-estimation, Instance hardness threshold.

1. Introduction

Credit card fraud is an issue that has affected Indonesia payment system over a decade. Sometimes, the result of the fraud used for terrorism and other crimes [1].

In Indonesia, the total loss because credit card fraud exceeds ten billion of Indonesia rupiahs per year. Based on Indonesia Credit Card Association (AKKI), the total cases for credit card frauds from July 2003 until April 2006 are 89 cases. The financial loss from this activity surpasses $ USD 4.0 million [1].

According to Bank Indonesia records, the most common schemes for credit card fraud are Card Not Present (CNP) and Card Present Fraud. Card Not Present (CNP) scheme is the fraud technique that used phishing. While Card Present Fraud is a fraud technique that using the credit card without the holder noticed [2]. The notable problem from the CNP scheme is the careless selection of the cardholder [1].

Financial loss is not the only problem that is affected caused by credit card fraud. According to Mr. Muhammad Helmi of AKKI, credit card fraud also affects Indonesia's images in international trade and e-commerce [1]. The loss also affects the merchants who bear the cost and administrative charges [3].

The problem of credit card fraud makes banks take serious measures to prevent it happened again. Many approaches are prompted, such as a refined

DOI: 10.22266/ijies2020.0430.17

Table 1. Previous research of credit card fraud detection.

Author	Title	Methods
Randhawa et al. [3]	Credit Card Fraud Detection Using Ada Boost and Majority Voting	Adaboost with the conjunction of twelve different classifiers and using majority votings.
Zareapoor et al. [4]	Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier	K-Nearest Neighbor, Support Vector Machine, Bagging based on Decision Tree and Naïve Bayes
Awoyemi et al. [5]	Credit card fraud detection using Machine Learning Techniques: A Comparative Analysis	Hybrid sampling method, K-Nearest Neighbor, Naïve Bayes, and Logistic Regression
Namvar et al. [7]	Credit risk prediction in an imbalanced social lending environment	Logistics regression, Linear Discriminant Analysis and Random Forest

security system and a detection system which work as fast as possible [4].

A secured and trusted banking payment system is a crucial thing. It needs high-speed verification and authentication to provide better quality for user business and transactions. Therefore fraud detection has become a vital activity to reduce the impact of the forged transactions [4].

Many researchers have proposed several techniques for credit card fraud, but most of them still used traditional methods, which time-consuming and inefficient because the current transaction is enormous, and many variables are measured for fraud detection. Presently, a financial institution such as the banks try to tackle this problem using computational methodology such as data mining [5].

As described by Zareapoor, there are five challenges in credit card fraud detection such as handling imbalanced data, the availability of the real data, dynamic conduct of the swindler, defining the correct evaluation and the enormous amount of the dataset [4].

In this paper, the problem which need to be solved is the high dimensional features set and imbalance data from credit card fraud dataset. It can be solved using feature reduction or feature selection and Sampling method for each task consecutively. This research applied undersampling method to solve imbalance data and two stages feature selection is proposed for high dimensional features set problem. The two stages feature reduction is proposed because the high dimensional features tend to reduced the performance of classification process and finding the optimal feature set is not easy for one stage feature set [6].

The structure of this paper can be divided into six sections. Section I explained about the introduction of this research. Section II explained the previous works that handled credit card fraud detection. Section III described the dataset. Section IV talked about the research methodology. Section V explained the result of the experiment. Section VI analyzed and described the conclusion of this research and future work.

2. Previous work

Table 1 illustrated several types of research that tried to handle credit card fraud detection using data mining methodology.

Randhawa et al. proposed using Adaboost with majority voting. They used twelve different classifiers algorithms such as Support Vector Machine, Tree algorithm, K- Nearest Neighbor, and other classifiers. The research used 10-fold cross-validation to perform the evaluation and tackled the imbalanced data using the under-sampling technique. The result shows that the majority votings method gave robust performances [3]. While Randhawa give many insight towards the adaboost algorithm and the effectiveness of under-sampling techniques but the research does not show good result in public dataset especially in classification of fraud class.

Zareapoor et al. suggested using Naïve Bayes, K-Nearest Neighbor, Support Vector Machine, and Bagging classifier. This research did not perform sampling methods for tackling the imbalanced problem, but they proposed a different approach to measure the model. The result shows that Bagging classifier performs well and takes less time to compute compared to other methods [4]. Even though the Matthew Correlation Coeficient of bagging is high but the other comparison algorithms are lower.

Awoyemi et al. proposed a hybrid sampling method for handling the imbalanced data. While the classifier that they used is K-Nearest Neighbor, Naïve Bayes, and Logistic Regression, the result shows that the KNN algorithm gave significant performance compared to other classifiers, which proved the effectiveness of hybrid sampling on the performance [5]. However, in Logistic Regression algorithm, the

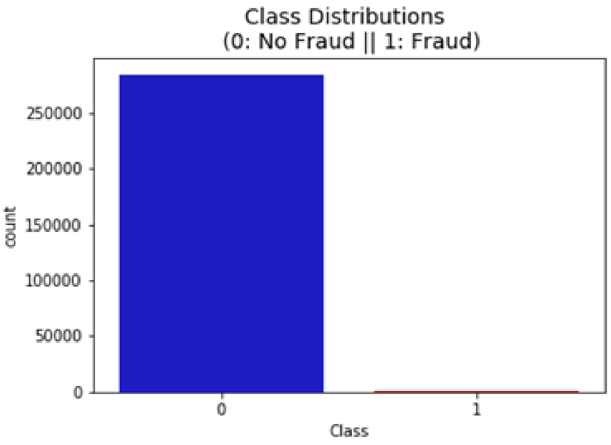

Figure. 1 Histogram credit card fraud data

under-sampling does not give boost in performance this could be happened because of the feature set have higher correlation value.

Namvar et al. proposed the imbalanced sampling approach using three different criteria such as undersampling, oversampling and hybrid. They used random undersampling, and Instance Hardness Threshold (IHT) for the undersampling approach; random oversampling, SMOTE and Adaptive Synthetic Sampling (ADASYN) for oversampling approach and the latter approach used SMOTE + Tomek links and SMOTE + Edited Nearest Neighbor (SMOTE-ENN). They adopted three different classifiers to create the model based on the sampling data. The result shows that the hybrid approach does not perform well compared to oversampling and undersampling methods. The undersampling approach shows the best result, especially in random undersampling techniques with the Random Forest as the classifier[7].

Based on the previous researches, this paper implemented undersampling using IHT and random undersampling technique, and for classification process, we used Random Forest, Support Vector Machine, Logistic Regression, and K Nearest Neighbour and Naïve Bayes as the classifiers. We are also implementing a feature engineering process to reduce the features using our proposed method: Two-stage feature reduction. The idea of introducing two - stages feature reduction because of a problem from using one stage feature reduction. One stage feature reduction does not give an optimal feature and finding the right features is not an easy task for one stage feature selection.

3. Data

The dataset is obtained from the ULB Machine Learning Group, and the description of this dataset explained in [8]. The dataset consists of credit card transactions from European cardholders in the year 2013. The dataset has 248,807 sales, and it has occurred in two days. The fraud cases in percentages wise hold 0.172% of the total transactions [5]. Fig. 1 shows how much the imbalance class from the dataset, while the non-fraud (0) class has most data compared to fraud (1) class.

The data have multivariable features which consist of 30 input features. The features represent amount, times, and 28 principal components which do not explain because of the confidential information [5].

4. Research methodology

The research methodology in this paper consist of four essential steps: outlier detection, undersampling, feature reduction and classification process. Fig. 2 illustrates the proposed system diagram and the detailed for each step will explain in each subsection.

4.1 Outlier detection

Outlier detection is a vital process in the data mining process, especially in a massive dataset like Credit Card Fraud dataset from ULB. The presence of the outliers in the data will generate a weak model that will mispredict the outcome of the class. Many types of research tend to ignore the outlier when making the model, but in most of the cases, ignoring the outlier will make information on the outlier data affect the model [9].

A robust method is proposed to manage the issue. In general, the robust model is used for managing the data peculiarity, detecting and eliminating the outliers, and also treating the outliers [9]. In this paper, we implemented the Minimum Covariance Determinant (MCD) algorithm for outlier detection. The MCD is chosen because of an affine equivariant and robust estimator for multivariate location [10].

```
The algorithm of the MCD [11], [12] :
Assume the set of observation $A = \{a_1, a_2, \dots, a_n\}$ With $n$ data and take a random sample with size $k$.
1   Determine random subset of $H_0$ moreover, with $k$ observation
2   Repeat:
        a) Determine covariance ( $Cov$ ) and mean of the $H_0$
        b) Determine distance $d(A_i)$ for all $A_i$ relative to $H$ using Mahalanobis Distance
        c) Choose the $k$ smallest distance and create a new subset $H_1$
```

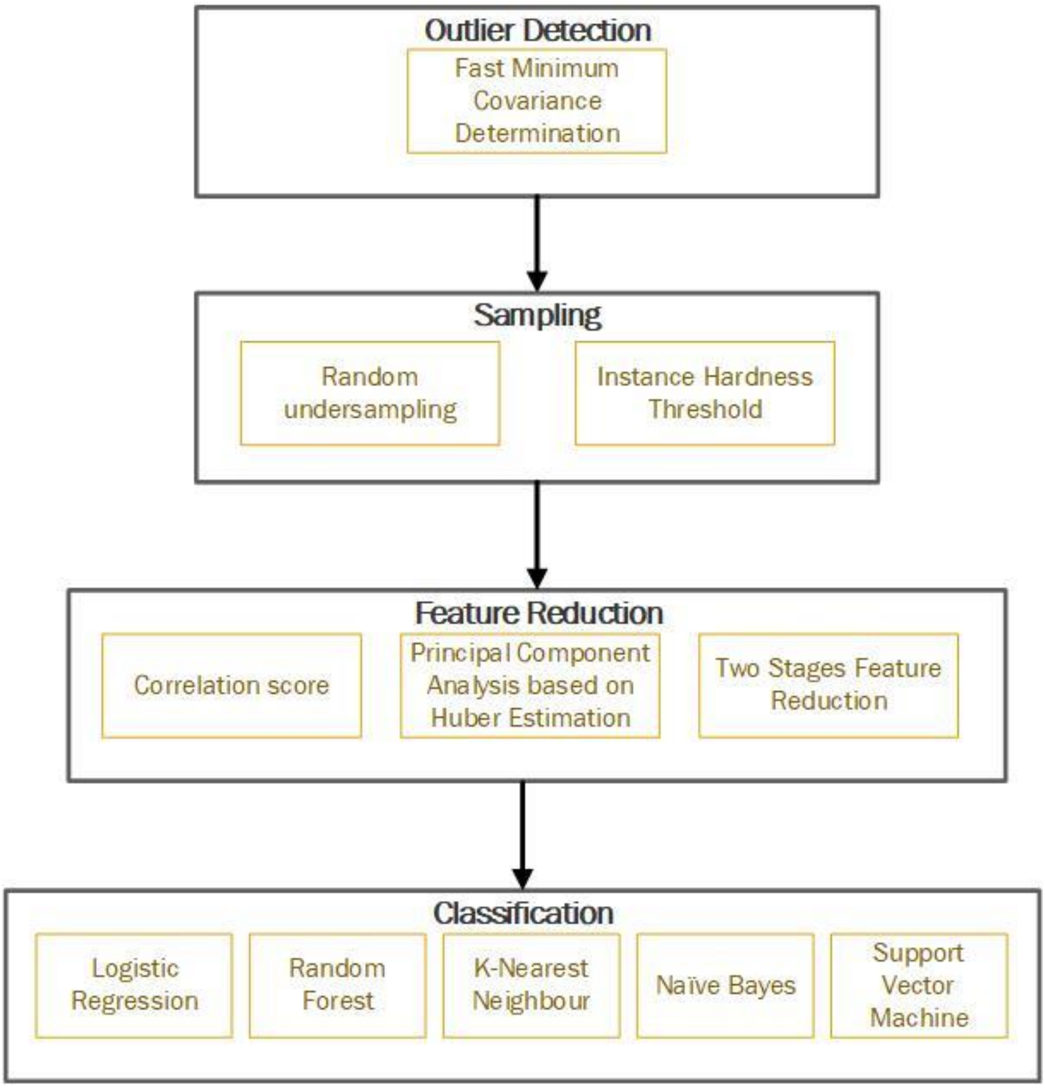

Figure. 2 System diagram of Credit Card Fraud Detection

d) Repeat the *step a to c* until H_0 and $H_1 = 0$

3 Evaluate from 1 to h times (possibly 500) and determine the subset who have the smallest volume

4.2 Sampling

The data, as stated in Section 3, has an imbalanced distribution. The problem will generate an error in the classification process because it will predict the majority class. One of the prominent techniques to solve the problem is using the resampling. Resampling techniques are an approach to generates an adjust training dataset before building the classification model [7].

According to the research which has done by Namvar et al. [7], we implemented the undersampling approach using random undersampling (RUS) and Instance Hardness Threshold (IHT) undersampling. The idea behind using these two undersampling methods is that the undersampling techniques work well in credit card fraud compared to other approaches [7].

Instance Hardness is an algorithm that used an instance in a dataset, and the instance has a hardness property, which implies the misclassified likelihood. In the undersampling approach, the Instance Hardness algorithm acts as a filter that removes suspected outlier or noise data, and this algorithm is called Instance Hardness Threshold (IHT) [13]. The definition of the Instance Hardness concerning [13]:

$$IH_h(X_i, Y_i) = 1 - p(Y_i | X_i, h) \qquad (1)$$

Where IH, (X_i, Y_i), $p(Y_i | X_i, h)$ represents Instance Hardness value, training dataset, the probability which h assigns the label Y_i respectively.

Random undersampling is a method of data sampling which randomly selects most of the class instances and removes them until the desired class distribution is attained [14].

4.3 Two-stage feature reduction

The proposed two-stage feature reduction method adopts the idea that was suggested by Zhao et al., to find the optimal features for the classification process [6]. The differences between our proposed method and Zhao method is in the approach. While Zhao using Information Gain and Binary Particle Swarm Optimization. We used Correlation-based measures to find the similarity between feature and Principal Component Analysis to reduce the features from high dimensional data. Feature reduction is a prominent step in pre-processing.

In the first stage of feature reduction, we used correlation-based measured to analyze the correlation between our features in the dataset. The most straightforward measure for correlation value is the linear correlation coefficient (r)[15].

$$r = \frac{\sum(X_i - \bar{X_\iota})\,(Y_i - \bar{Y_\iota})}{\sqrt{\sum(X_i - \bar{X_\iota})}\sqrt{\sum(Y_i - \bar{Y_\iota})}} \tag{2}$$

Where X, Y is the pair features and $\bar{X}, \bar{Y}$ is the mean of the features.

After we find the correlation coefficients for each feature, we removed the feature, that have a correlation coefficient of more than 80%. The higher correlation scores mean the features are linearly dependent, and the features have the same effect for each other.

In the second stage, we implemented a Robust Principal Component Analysis (PCA) with Huber-estimation. PCA was used to reduce the dimensionality of the data by minimizing the mean square error of the subspaces [16]. However, the PCA is susceptible to the outliers. To handle the outlier problem, we used M-estimator to reweight the covariance matrix. The M-Estimator is used because the

The Huber M-estimator[17]:

$$x = argmin \sum_{i=1}^{N} h(|x_i - x|) \tag{3}$$

Where:

$$y(t) = \begin{cases} y^2/2 & for\ |y| \le \Delta \\ \Delta|y| - \Delta^2/2 & for\ |y| > \Delta \end{cases} \tag{4}$$

Where Δ is the threshold to measure the outlier of the data; $h(t)$is the Huber M-estimator.

In this case our Robust PCA Algorithm is [17], [18]:

1. Compute the mean of the matrix:

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{5}$$

2. Find the value of threshold (Δ) for Huber M-estimator from the data. The threshold is chosen by using percentile.
3. Compute the covariance matrix C with Huber M-Estimator.
4. Solve the eigen value problem
5. Perform dimensional reduction using eigenvector, from eigenvalue in decreasing order

4.4 Classification

A total of five machine learning algorithms are used in this research. The algorithms that we used are Naïve Bayes\, K-Nearest Neighbor, Support Vector Machine, Random Forest, and Logistic Regression. Each machine learning represents the different approaches for making the model.

Naïve Bayes (NB) is a probabilistic machine learning that works better in supervised especially classification tasks. The Naïve Bayes algorithm based on Bayes theorem and assumes the independence of each class [3].

$$P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)} \tag{8}$$

Logistic Regression (LR) is an algorithm used in practice because of the simplification and balance distribution towards the error. The logistic regression usually used for binary classification and the formulation [7]:

$$Ln\left(\frac{F(x)}{1 - F(x)}\right) = b_0 + \sum_{i=1}^{n} b_i x_i \tag{9}$$

K-Nearest Neighbor (KNN) is an algorithm that assumes a similar class will exist near to each other. KNN takes the idea of similarity. The similarity usually based on the distance between the data. KNN is generally used for supervised learning in classification and regression [19].

Table 2. The summary of the machine learning method and the design principle [20]

Machine Learning Method	Type	Design principle
KNN	Distance-based	Finding the class based on the nearest k-neighbour
SVM	Distance-based	The hyperplane separated the classes
LR	Probabilistic	Estimates the probability of a binary response predictor features
NB	Probabilistic	Learning the probabilistic belongs to a specific vector
RF	Tree-based	An ensemble of a decision tree

Table 3. The proposed techniques

Outlier Detection	Undersampling	Feature Reduction	Classifier
MCD	Random Undersampling	Correlation	KNN
			SVM
		PCA using Huber estimation	LR
	Instance Hardness Threshold		NB
		Two-stage feature reduction	RF

$$d(x_i, y_i) = \sqrt{\sum (y_i - x_i)^2} \qquad (10)$$

Support Vector Machine (SVM) is an algorithm that can tackle supervised and regression. The idea of SVM is building the new model by assigning new samples to category and creating the non-probabilistic binary classifier.

Random Forest (RF) is an ensemble algorithm based on a decision tree algorithm. The idea behind RF is multiple decision trees that have been trained on bootstrap samples, and the algorithm chooses a subset randomly to building the tree, and after that, the voting function is used to generate the model [7].

A summary of each classifier can be seen in Table 2.

4.5 Evaluation

The performance of the classification methods is evaluated based on accuracy, specificity, recall, and Matthew Correlation Coefficient (MCC). These evaluation metrics are applied because of the relevance in assessing the imbalanced classification problem [5].

Table 4. The amount of data before and after the cleaning phase using MCD

	Before Cleaning	After Cleaning
Fraud	492	399
Non-Fraud	284,315	281,559
Total	284,807	281,958

Accuracy is the initial evaluation for the classification process, but in an imbalance dataset, the accuracy does not perform well. The evaluation tends to accentuate the majority class [7]. Specificity and recall are the proper evaluation for binary classification because it gives the accuracy for each category. MCC is an excellent evaluation for an imbalanced dataset because it consists of True Positive (TP), False Negative (FN), True Negative (TN) and False Positive (FP) in the evaluation. The MCC value is usually between -1 to 1, where 1 represents the proper classification, while -1 represents the distinction between classification and process [5].

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \qquad (11)$$

$$recall = \frac{TP}{TP + FN} \qquad (12)$$

$$specificity = \frac{TN}{TP + FN} \qquad (13)$$

$$MCC = \frac{(TP \times TN) - (FP \times FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \qquad (14)$$

5. Experiment and result

In this evaluation, we assessed the performance of the classifier in combination with various approaches. Table 3 shows the combinations for our approaches. However, we implemented two different scenarios.

Before the proposed techniques, we normalized the dataset feature values, especially the “Amount” and “Time” using Robust scaler from scikit-learn [21].

The next step is using MCD to find the outliers from the data and removing it. This step is done to make the data cleaner. The amount of data after the cleaning phase can be seen in Table 4.

Table 5. Result without using feature reduction and undersampling

Classifier	accuracy	recall	specificity	MCC
KNN	0.99954	0.81579	0.99979	0.82651
SVM	0.99949	0.71053	0.99988	0.79284
LR	0.99933	0.65789	0.9997	0.72807
NB	0.99933	0.68421	0.99975	0.73389
RF	0.99959	0.77632	0.99989	0.83924

Table 6. Result using Random Undersampling [7], without feature reduction

Classifier	accuracy	recall	specificity	MCC
KNN	0.97368	0.96053	0.98684	0.94770
SVM	0.96053	0.94737	0.97368	0.92137
LR	0.96053	0.96053	0.96053	0.92105
NB	0.92763	0.86842	0.98684	0.86132
RF	0.94737	0.92105	0.97368	0.89598

Table 7. Result using Random Undersampling [7], and feature reduction: correlation [15]

Classifier	accuracy	recall	Specificity	MCC
KNN	0.96053	0.93421	0.98684	0.92233
SVM	0.96711	0.94737	0.98684	0.93494
LR	0.94079	0.93421	0.94737	0.88166
NB	0.90789	0.85526	0.96053	0.82035
RF	0.93421	0.93421	0.93421	0.86842

According to Table 4, the amount of data that detect as outliers are 93 for the fraud transaction and 2,576 for the non-fraud transaction.

The next phase will divide into two different scenarios and using ratio 80: 20 for train and test. The first scenario is without using undersampling and feature reduction. Table 5 illustrates the result of this scenario.

Based on Table 5, the accuracy and specificity score show the appropriate result. However, the recall and MCC score does not show any excellent performance. The problem happens because the model tends to misclassify the fraud class. This can be seen in lower recall score compared to specificity. The low recall and MCC score problem can be solved using sampling techniques. we used undersampling rather than using hybrid or oversampling.

The second scenario is applying undersampling and feature reduction. In this scenario, we compared four cases for feature reduction: without any feature reduction, using correlation score, using PCA with Huber estimation and using two-stage feature reduction.

Table 6 until Table 9 explained the result of undersampling using Random Undersamplimg methods. Table 10 until Table 13 explained the result of undersampling using Instance Hardness Threshold.

Table 8. Result using Random Undersampling [7], and feature reduction: PCA with Huber estimation [17]

Classifier	accuracy	recall	Specificity	MCC
KNN	0.44737	0.89474	0	-0.2357
SVM	0.50000	1	0	Nan
LR	0.30263	0.60526	0	-0.4959
NB	0.48026	0.96053	0	-0.1419
RF	0.76974	0.85526	0.68421	0.54754

Table 9. Result using Random Undersampling [7], and feature reduction: two-stage feature reduction (our proposed method)

Classifier	accuracy	recall	specificity	MCC
KNN	0.30921	0.59211	0.02632	-0.4628
SVM	0.47368	0.92105	0.02632	-0.1179
LR	0.26316	0.52632	0	-0.5571
NB	0.35526	0.71053	0	-0.4114
RF	0.73684	0.57895	0.89474	0.4992

Table 10. Result using Instance Hardness Threshold [13], without feature reduction

Classifier	accuracy	recall	specificity	MCC
KNN	1	1	1	1
SVM	0.99346	1.	0.98701	0.98701
LR	1	1	1	1
NB	0.99346	1.	0.98701	0.98701
RF	1	1	1	1

Table 11. Result using Instance Hardness Threshold [13], and feature reduction: correlation [15]

Classifier	acc	recall	specificity	MCC
KNN	1	1	1	1
SVM	0.99346	1	0.98701	0.98701
LR	1	1	1	1
NB	0.98693	1	0.97403	0.97419
RF	1	1	1	1

Table 12. Result using Instance Hardness Threshold[13], and feature reduction: PCA with Huber estimation[17]

Classifier	accuracy	recall	specificity	MCC
KNN	0.96078	0.94737	0.97403	0.92187
SVM	0.66667	1.00000	0.33766	0.44952
LR	1	1	1	1
NB	1	1	1	1
RF	1	1	1	1

Table 13. Result using Instance Hardness Threshold [13], and feature reduction: two-stage feature reduction (our proposed method)

Classifier	accuracy	Recall	specificity	MCC
KNN	1	1	1	1
SVM	0.60131	1	0.20779	0.33952
LR	0.85621	0.92105	0.79221	0.71881
NB	0.98693	1	0.97403	0.97419
RF	0.98039	1	0.96104	0.96153

According to Table 6 until Table 13, the result shows that the undersampling techniques improved the recall, specificity and MCC value compared to without using the undersampling technique. It is shown that the effectiveness of using undersampling for credit card classification.

In the feature reduction, the two-step feature reduction does not perform well compared to the correlation coefficient method. The problem of the proposed method laid in the second stage. While PCA fails to differentiate the outlier using Huber Estimator.

6. Conclusion

This paper investigates the effectiveness of undersampling and feature reduction for credit card fraud detection. This paper also proposed a two-stages feature reduction using correlation coefficient score and Principal Component Analysis with Huber estimation. The result shows that using undersampling technique especially Instance Hardness Threshold method boost performance of model. However, the two-stage feature reduction does not perform well in the dataset. The problem come from the second stage of our proposed method. The Robust PCA using Huber estimator failed to differentiate the outlier from the data.

References

[1] H. Y. Prabowo, "A better credit card fraud prevention strategy for Indonesia", *Journal of Money Laundering Control*, Vol. 15, No. 3, pp. 267–293, 2012.

[2] B. Buonaguidi, "Credit card fraud: What you need to know", [Online]. Available: https://www.bbc.com/worklife/article/20170711-credit-card-fraud-what-you-need-to-know. [Accessed: 27-Aug-2019].

[3] K. Randhawa, C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, "Credit Card Fraud Detection Using AdaBoost and Majority Voting", *IEEE Access*, Vol. 6, pp. 14277–14284, 2018.

[4] M. Zareapoor and P. Shamsolmoali, "Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier", *Procedia Computer Science*, Vol. 48, pp. 679–685, 2015.

[5] J. O. Awoyemi, A. O. Adetunmbi, and S. A. Oluwadare, "Credit card fraud detection using machine learning techniques: A comparative analysis", In: *Proc. of 2017 International Conference on Computing Networking and Informatics (ICCNI)*, pp. 1–9, 2017.

[6] X. Zhao, D. Li, B. Yang, H. Chen, X. Yang, C. Yu, and S. Liu, "A two-stage feature selection method with its application", *Computers & Electrical Engineering*, Vol. 47, pp. 114–125, 2015.

[7] A. Namvar, M. Siami, F. Rabhi, and M. Naderpour, "Credit risk prediction in an imbalanced social lending environment", *International Journal of Computational Intelligence Systems*, Vol. 11, No. 1, p. 925, 2018.

[8] A. D. Pozzolo, O. Caelen, R. A. Johnson, and G. Bontempi, "Calibrating Probability with Undersampling for Unbalanced Classification", In: *Proc. of 2015 IEEE Symposium Series on Computational Intelligence*, Cape Town, South Africa, pp. 159–166, 2015.

[9] N. Gusriani and Firdaniza, "Linear regression based on Minimum Covariance Determinant (MCD) and TELBS methods on the productivity of phytoplankton", *IOP Conf. Ser.: Mater. Sci. Eng.*, Vol. 332, p. 012037, 2018.

[10] M. Hubert and M. Debruyne, "Minimum covariance determinant: Minimum covariance determinant", *WIREs Comp. Stat.*, Vol. 2, No. 1, pp. 36–43, 2010.

[11] P. J. Rousseeuw and K. V. Driessen, "A Fast Algorithm for the Minimum Covariance Determinant Estimator", *Technometrics*, Vol. 41, No. 3, pp. 212–223, 1999.

[12] J. Shore, "Minimum Covariance Determinant", [Online]. Available: https://tr8dr.wordpress.com/2010/09/24/minim

um-covariance-determination/. [Accessed: 15-Sep-2019].

[13] M. R. Smith, T. Martinez, and C. Giraud-Carrier, “An instance level analysis of data complexity”, *Mach Learn*, Vol. 95, No. 2, pp. 225–256, 2014.

[14] J. Prusa, T. M. Khoshgoftaar, D. J. Dittman, and A. Napolitano, “Using Random Undersampling to Alleviate Class Imbalance on Tweet Sentiment Data”, In: *Proc. of 2015 IEEE International Conference on Information Reuse and Integration*, pp. 197–202, 2015.

[15] E. C. Blessie and E. Karthikeyan, “Sigmis: A Feature Selection Algorithm Using Correlation Based Method”, *Journal of Algorithms & Computational Technology*, Vol. 6, No. 3, pp. 385–394, Sep. 2012.

[16] T. Zhang and G. Lerman, “A Novel M-Estimator for Robust PCA”, *Journal of Machine Learning Research*, Vol. 15, pp. 749–808, 2014.

[17] B. T. Polyak and M. V. Khlebnikov, “Principle component analysis: Robust versions”, *Autom. Remote Control*, Vol. 78, No. 3, pp. 490–506, 2017.

[18] P. Wiriyathammabhum and B. Kijsirikul, “Robust Principal Component Analysis Using Statistical Estimators”, In: *Proc. of the 6th International Joint Conference on Computer Science and Software Engineering*, pp. 1–6, 2009.

[19] O. Harrison, “Machine Learning Basics with the K-Nearest Neighbors Algorithm”, [Online]. Available: https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761. [Accessed: 18-Sep-2019].

[20] M. Pichler, V. Boreux, A. Klein, M. Schleuning, and F. Hartig, “Machine learning algorithms to infer trait-matching and predict species interactions in ecological networks”, *Methods in Ecology and Evolution*, pp. 1–13, 2019.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, and D. Cournapeau, “Scikit-learn: Machine Learning in Python”, *Journal of Machine Learning Research*, Vol. 12, pp. 2826–2830, 2011.