INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

journal homepage : www.joiv.org/index.php/joiv

Handling Imbalanced Data for Acute Coronary Syndrome Classification Based on Ensemble and K-Means SMOTE Method

Muhammad Faris Muzakki [a], Rizal Dwi Prayogo [a,b,*], M Afif Rizky A [a]

[a] *School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Bandung, Indonesia*
[b] *Center for Artificial Intelligence (U-COE AI-VLB), Institut Teknologi Bandung, Bandung, Indonesia*
Corresponding author: [*]rizaldp@itb.ac.id

***Abstract*—Acute Coronary Syndrome (ACS) is a disease that has a high mortality rate with a mortality percentage of 40% after 5 years from diagnosis. Despite the high mortality rate, the conventional process of overestimating ACS can be life-threatening. For this reason, several alternatives for prediagnosis have been investigated to reduce the detection of ACS intensively, one of which is by using a machine learning approach. The machine learning-based prediagnosis approach utilizes patient medical record data as input for making detection models. This approach can produce an optimal model when there is quite a lot of data and the labels have a fairly balanced comparison. However, in machine learning-based ACS detection studies, researchers often do not have balanced data between positive and negative labels that have the potential to cause overfitting. That problem occurs because obtaining additional data with specific labels is difficult. To solve the imbalanced problem in ACS detection, we generated synthetic ACS data using the K-Means SMOTE method. The synthesis data is used as training data to build an ensemble-based machine-learning model. In this study, we obtain an increase in the F1 score of more than 10% when compared to machine learning models that do not use the K-Means SMOTE as an oversampling process. In addition to the greater F1 score, the results obtained are relatively more resistant to overfitting because the data variations in the training set are more diverse.**

***Keywords*— Acute Coronary Syndrome; imbalance learning; k-Means SMOTE.**

Manuscript received 29 Nov. 2022; revised 25 Apr. 2023; accepted 19 Jun. 2023. Date of publication 30 Nov. 2023.

I. INTRODUCTION

Acute Coronary Syndrome (ACS) occurs when part of the heart muscle does not function properly or dies due to a decrease in the supply of blood flow in the coronary arteries [1]. This is triggered by cholesterol plaques forming the inner walls of the coronary arteries (atherosclerosis) [2]. Individuals suffering from ACS have a 40% chance of dying within five years [3], so this health problem is a major concern for most countries. Although ACS is very dangerous [4], detecting ACS is very difficult, and excessive detection can be life-threatening for the patient [5]. For these reasons, several studies have been carried out to support the initial diagnosis of ACS; one is using a machine learning approach.

Machine learning (ML) is a data processing technique that can be used to classify based on existing data [6]. The study in [7] uses the artificial neural network (ANN) method with an F1 score of 0.849. Other researchers [8], [9] use a Decision Tree with an F1 score of 0.979 and the Random Forest algorithm with an accuracy of 83.45%. However, the results of previous studies were still not optimal because the dataset used to build the classification model was extremely imbalanced [10]. To solve the imbalance problem, one of the effective methods that can be used is oversampling [11]. Oversampling is a method that can be used to overcome imbalanced problems without losing any information from the original data, such as the undersampling approach [12]. Many oversampling algorithms have been formulated by researchers. One of the most stable oversampling algorithms for tabular data cases is K-means Smote [13].

K-Means SMOTE is an advanced oversampling method from SMOTE which is added with clustering and filtering processes to minimize noise and improve the quality of the resulting synthetic data. This algorithm is used in several similar studies and produces optimal results [14], [15]. In this study, we use the K-Means SMOTE method to overcome the imbalanced problem in ACS classification so that the dataset used to build the prediction model is balanced. To evaluate the results, we used the Random Forest classification algorithm to measure accuracy, F1 score, and ROC AUC.

Also, we used the Mann-Whitney U statistical test to measure whether the results of ML using K-Means SMOTE had a significant impact or not on the ML results.

II. Materials and Method

A. Dataset Description

The ACS dataset we used in this study was taken from Indonesia General Hospital and has been approved by the ethics committees of Institut Teknologi Bandung (August 26, 2022). Belmont Report and International Ethical Guideline performed all procedures. Due to the privacy policy, we cannot publicly post the dataset. This ACS dataset contains 480 instances, with 138 (28.75%) identified cases of ACS and 342 (71.25%) unidentified cases. This dataset consists of 14 features can be seen in Table 1.

TABLE I
Feature of Data

No	Feature Name	Type
1	Age	Numeric [3 - 88]
2	Agina Type	Categorical
3	Angina Activity	Categorical
4	Cholesterol	Numeric [71 - 564]
5	Electrocardiographic Result	Categorical
6	Fasting Blood Sugar	Categorical
7	Gender	Categorical
8	Maximum Heart Rate	Numeric [40 - 202]
9	Num Major Vessels	Numeric [0 - 4]
10	Resting Blood Pressure	Numeric [50 - 200]
11	ST Depression ECG	Numeric [0 - 6.2]
12	ST Slope ECG	Categorical
13	Thalassemia	Categorical
14	Target	Categorical

To calculate the value of the imbalance data in this study, we used the imbalance ratio (IR) as follows:

$$IR = \frac{Number\ of\ Minority\ Class\ Sample}{Number\ of\ Majority\ Class\ Sample} = \frac{138}{342} = 0.403 \quad (1)$$

The IR value produces a range of 0 to 1, where a value of 1 means balanced and 0 is imbalanced. In this study, we will carry out an oversampling process to make the IR value close to or equal to 1. We also map data of type to continue to see the distribution of our data. Complete data can be seen in Table 2. In this study, the age range of the data used is 3-88 years, resting blood pressure 40-200, cholesterol 71-564, maximum heart rate 40-202, st depression ECG 0-6.2, and the number of major vessels 0-4.

TABLE II
Numerical Type Data

No	Feature Name	Min	Max	Std	Average
1	Age	3	88	11.76	53.19
2	Resting Blood Pressure	40	200	23.46	121.33
3	Cholesterol	71	564	64.97	213.54
4	Maximum Heart Rate	40	202	30.66	132.55
5	St Depression ECG	0	6.2	1.02	0.71
6	Num Major Vessels	0	4	1.35	1.54

B. K-Means SMOTE

K-Means SMOTE [13] is an improvement of the SMOTE algorithm [16] which still has much noise during the data generation process. This algorithm has three main steps: clustering, filtering, and oversampling.

The clustering process is carried out to separate the majority and minority classes. If, in a cluster, there is an IR value less than 1, then the oversampling process will be carried out using the SMOTE algorithm. Details of the algorithm are presented in Algorithm 1.

Algorithm 1: K-Means SMOTE

Input : Imbalanced Dataset
Output : Balanced Dataset
$clusters \leftarrow Kmeans(x)$
$filtered\_cluster \leftarrow \emptyset$
for $c \leftarrow 1$ *to* $clusters$ **do**
  $imabalance\_ratio \leftarrow \frac{majority\_count(c)+1}{minority\_count(c)+1}$
  if $imabalance\_ratio < 1$ **then**
    $filtered\_cluster \leftarrow filtered\_cluster + c$
  end
end
for $c \leftarrow 1$ *to* $clusters$ **do**
  $average\_minority\_dist(c) \leftarrow mean(euclidiandist(c))$
  $density\_factor(c) \leftarrow \frac{majority\_count(c)}{average\_minority\_dist(c)^{de}}$
  $sparsity\_factor(c) \leftarrow \frac{1}{density\_factor(c)}$
end
$sparsity\_sum \leftarrow \sum_{c=1}^{clusters} sparsity\_factor$
$sampling\_weight(c) \leftarrow \frac{\text{density\_factor(c)}}{\text{sparsity\_sum}}$
$generated\_sample \leftarrow \emptyset$
for $c \leftarrow 1$ *to* $clusters$ **do**
  $number\_of\_samples(c) \leftarrow length\ of\ c\ x\ sampling\ weight$
  $\leftarrow generated\_sample + SMOTE(c, number\_of\_samples, knn)$
end

In this study, we used K-Means Smote algorithm to produce synthetic data for the minority class, namely the positive ACS class. Thus, the data used can be balanced with IR = 1.

C. Random Forest

Random Forest [17] is a machine-learning algorithm built using multiple decision trees (bagging concept). Compared to conventional tree algorithms, the advantages of this algorithm are that it has better noise resistance, does not produce overfitting, and has better accuracy [18], [19]. In the Random Forest algorithm, several stages are carried out to build the model [18], which can be seen in the stages in Algorithm 2, namely:

- Conduct random sampling of data and features for each input tree.
- Build a tree model.
- voting (average, majority, etc.)

Algorithm 2: Random Forest

Input: X – input data, *n* number of trees, *z* sub sample size, *f* number of features

Output : a set of *t* Trees

initialize Random Forest ← list

for *i ← 1 to n* ***do***

    bootstrap_sample
    ← draw bootstrap from dataset (z)
    m features ← random all feature (f)
    random_forest add tree(boostrap_sample,m_feat

end

D. Autoencoder

Autoencoder is an algorithm based on the artificial neural network [20], which encodes data that does not have a label [21]. The purpose of this algorithm is to reconstruct the output so that it is close to the input. Autoencoder algorithms are generally used for data transformation and feature selection with three main layers, encoder, bottleneck, and decoder. Architectural drawings can be seen in Fig 1.

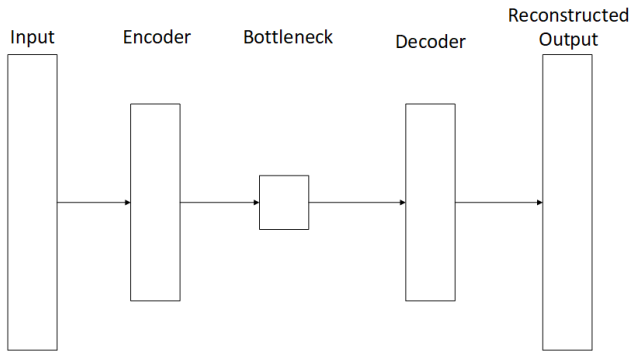

Fig. 1 Autoencoder

The function of the encoder is to receive input and transform the data into lower dimensions, proceed to the bottleneck layer, which will carry out the encoding process, and end with the decoder layer, where the data reconstruction process is carried out using the encoding results. In this study, we use an autoencoder algorithm to perform preprocessing and feature selection to make the data used as the input model more optimal.

E. Model Evaluation

Accuracy, F1 score, and ROC AUC are used in this study to measure the performance results of each experimental scenario carried out. F1 score is obtained from the precision and recall values as follows:

$$F1\ score = \frac{Precision\ x\ Recall}{Precision+Recall} \quad (2)$$

with each precision and recall formula as follows:

$$F1\ Precision = \frac{TP}{TP+FP} \quad (3)$$

A true positive value (TP) is obtained from each positive class predicted to be a positive class, while a false positive value (FP) is obtained from every negative class predicted to be a positive class.

$$recall = \frac{TP}{TP+FN} \quad (4)$$

The false negative (FN) value is obtained from the positive class, which is predicted to be a negative class.

Meanwhile, ROC AUC is obtained from the area under the ROC curve. The ROC value is obtained by comparing the True Positive Rate (TPR) and the False Positive Rate (FPR), plotting into a two-dimensional graph based on all classification thresholds. TPR and FPR formula are as follows:

$$TPR = \frac{TP}{TP+FN} \quad (5)$$

$$FPR = \frac{FP}{FP+TN} \quad (6)$$

A true negative value (TN) is obtained from each negative class that is predicted to be a negative class.

F. Scenario

In this study, we took several steps to obtain the research results, which can be seen in Fig 2. The first step is to collect the dataset and preprocess the data. The preprocessing carried out includes the disposal of unreasonable data, such as data with an age of 0 years, filling in empty columns using averages, and categorizing data with non-continuous types. After the dataset is obtained, we transform the data using an autoencoder algorithm and divide the dataset into train and test before the oversampling process. After separating the datasets, we oversampled the dataset using K-Means SMOTE algorithm. Next, we carry out the learning process with the following configurations of folds: 3, 5, 7, 9, 10, and 30The final step is to evaluate the model using the F1 Score, Accuracy, and ROC AUC parameters obtained from the machine learning model testing process. The evaluation process involves statistical processes and tests with nonparametric-based statistical tests.

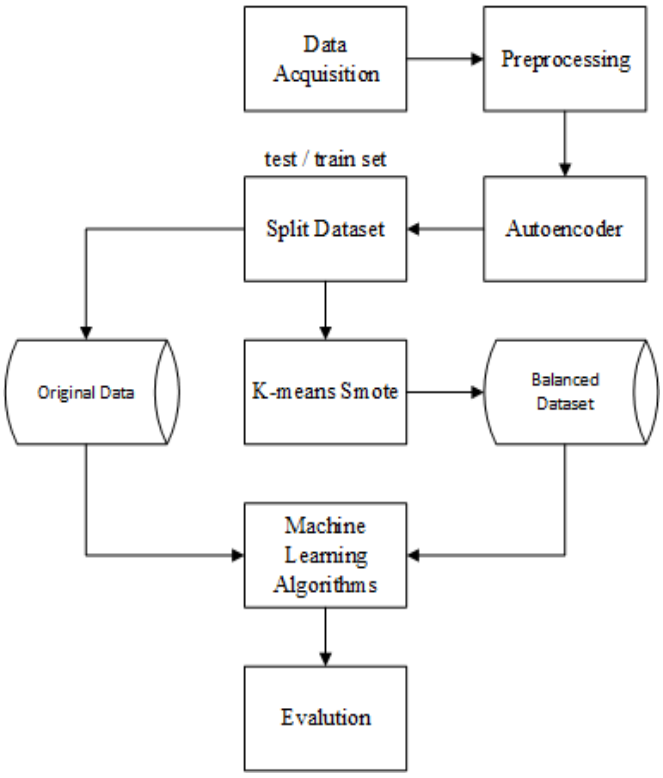

Fig. 2 Scenario

III. Result and Discussion

A. Feature Analysis

In this study, we ranked the features with the most significance on the ACS labeling Gini importance [22]. Gini importance, also known as impurity importance, is obtained from the value of impurity reduction carried out in the feature tree splitting process. The value obtained from each tree will be averaged against the number of trees in a Random Forest to compare the values between the variables. The higher the Gini value, the more significant the feature is on the target. The results of the ranking can be seen in Fig 3

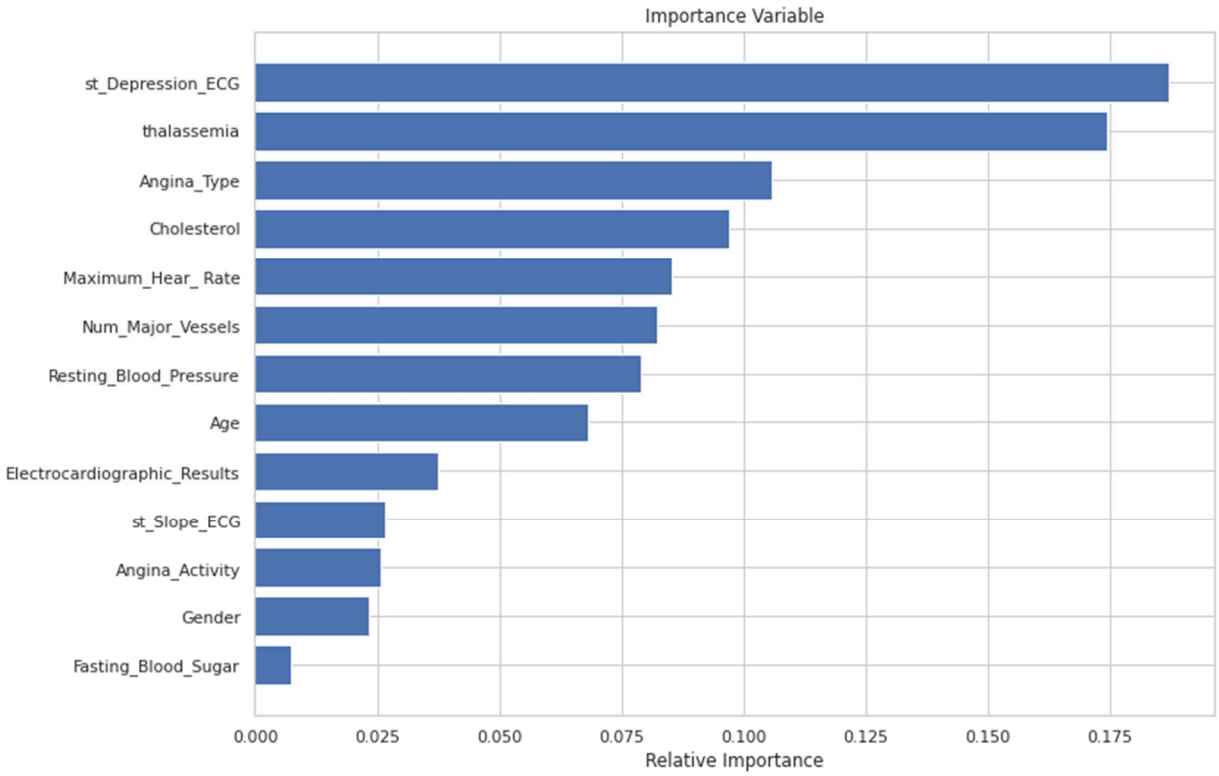

Fig. 3 Ranking of Features Using Relative Importance

In Fig 3, we can see that the most significant features on the ACS label are st depression ECG and thalassemia, which account for more than 17% of the total features. Meanwhile, the resting blood pressure, num major vessels, maximum heart rate, angina type, and cholesterol features affect between 7.8% and 10.5%. Other features only have an impact of less than 7.5% each.

B. Autoencoder Result

In this research, we use two layers of an encoder with one additional bottleneck layer and end with two layers of a decoder. The output of this process is the transformed data. The auto-encoder layer and parameters can be seen in Fig 4. The autoencoder was run for 200 epochs and produced the lowest training loss value of 0.0181 and the lowest loss validation value of 0.0149. The loss value of each epoch can be seen in Fig 5.

C. K-Means SMOTE Impact on Machine Learning Models

In this study, all machine learning models built using the k-Means SMOTE data train had better F1 scores, accuracy, and ROC AUC scores compared to models built using the original data train. The entire distribution of results can be seen in Table 3. The best F1 score obtained is 0.8515 with 30-fold configurations. In comparison, the lowest F1 score is 0.7087 with 3-fold configuration and uses the original training data as a modeling material.

Layer (type)	Output Shape	Param #
input_3 (InputLayer)	[(None, 13)]	0
dense_12 (Dense)	(None, 26)	364
batch_normalization_8 (BatchNormalization)	(None, 26)	104
leaky_re_lu_8 (LeakyReLU)	(None, 26)	0
dense_13 (Dense)	(None, 13)	351
batch_normalization_9 (BatchNormalization)	(None, 13)	52
leaky_re_lu_9 (LeakyReLU)	(None, 13)	0
dense_14 (Dense)	(None, 6)	84
dense_15 (Dense)	(None, 13)	91
batch_normalization_10 (BatchNormalization)	(None, 13)	52
leaky_re_lu_10 (LeakyReLU)	(None, 13)	0
dense_16 (Dense)	(None, 26)	364
batch_normalization_11 (BatchNormalization)	(None, 26)	104
leaky_re_lu_11 (LeakyReLU)	(None, 26)	0
dense_17 (Dense)	(None, 13)	351

Total params: 1,917
Trainable params: 1,761
Non-trainable params: 156

Fig. 4 Autoencoder Layers and Parameters

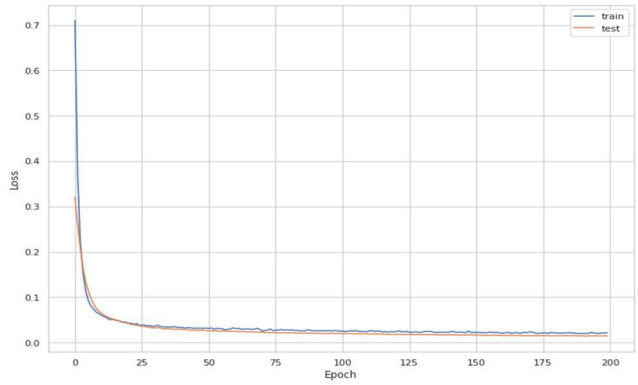

Fig. 5 Autoencoder Training Plot

In this study, we also compared the results of the k-Means SMOTE with several other oversampling algorithms such as SMOTE [16], ADASYN [23], Gaussian SMOTE [24], Cure SMOTE [25], SMOTE PSO [26] and Borderline SMOTE [27]. The results of each algorithm can be seen in Table IV. In the experimental results, K-Means SMOTE gives the highest results for all scenarios when compared to the results of other oversampling algorithms.

D. Comparison with Previous Studies

In Table 5, we present a comparison of our research with previous studies. In the study of [7]–[9] and this study, ACS cases were only under 32% of the total data, whereas in the [28] study ACS cases had 97% of the total data. The composition of the ACS data in these studies is imbalanced. However, in [29] study, the ACS case had 50% of the composition of all data, which means the data in the study were balanced. Although several previous studies had an F1 score that was better than ours, this cannot be a measure of the quality of the predictive model because of the bias in the type and quality of data used in each study.

E. Statistical Evaluation

To obtain more accurate and unbiased comparison results, we use nonparametric-based statistical tests [30] to compare the results between the model built with the original data with the K-Means SMOTE based on Table 3. The use of this method can provide an exact description of the distribution of values. In this study, we used Mann-Whitney U [31] for statistical tests with the following as follows:

$$U_1 = n_1 n_2 + \left(\frac{n_1(n_1+1)}{2}\right) - r_1 \quad (7)$$

$$U_2 = n_1 n_2 + \left(\frac{n_2(n_2+1)}{2}\right) - r_2 \quad (8)$$

$$p = min(U_1, U_2) \quad (9)$$

where n_1 and n_2 are the values of the F1 score from the results of the k-means smote data and the original data. r_1 and r_2 are the rank of sum in the groups. To measure the null hypothesis, we use a threshold $\alpha = 0.05$. Based on the calculation result, we obtained a p-value of 0.00216. that result can be concluded that H_0 is rejected. The results mean that the use of the K-Means SMOTE has a significant effect on increasing the F1 score.

TABLE III
MACHINE LEARNING RESULT

Fold	Accuracy		Weighted Average		Roc Auc	
	Original	K-Means Smote	Original	K-Means Smote	Original	K-Means Smote
5	0.7125	0.8295	0.7087	0.8293	0.8105	0.8737
7	0.7292	0.8336	0.7249	0.8323	0.8235	0.8876
9	0.7483	0.8382	0.7432	0.8378	0.8350	0.8803
10	0.7505	0.8485	0.7464	0.8483	0.8284	**0.8925**
30	0.7521	**0.8506**	0.7432	**0.8486**	0.8392	0.8915
Avg	0.7542	**0.8534**	0.7446	**0.8515**	0.8440	**0.8810**

TABLE IV
OVERSAMPLING COMPARISON

Fold	Weighted Average						
	K-Means SMOTE	SMOTE	ADASYN	Gaussian SMOTE	Cure SMOTE	Borderline SMOTE	SMOTE PSO
3	0.8522	0.7949	0.7896	0.7746	0.8259	0.7737	0.7234
5	0.8397	0.8125	0.8105	0.8182	0.8332	0.8187	0.7231
7	0.8648	0.8266	0.8329	0.8203	0.8307	0.844	0.7216
9	0.8555	0.8464	0.8150	0.8301	0.8345	0.8458	0.7288
10	0.8547	0.8367	0.8337	0.8213	0.8379	0.8422	0.7352
30	**0.8560**	0.8381	0.8301	0.8290	0.8361	0.8479	0.7491
Avg	**0.8538**	0.8259	0.8186	0.8156	0.8331	0.8287	0.7302

TABLE V
PREVIOUS STUDIES

No	Studies	Data Input	Data Size	IR	Methods	Result (%)
1	[7]	40 features (medical, physical exam, and ECG histories)	2204	16.3	ANN	F1 = 84.9
2	[8]	9 features (medical histories)	887	27.8	ANN	F1 = 95.2
3	[28]	37 features (age, sex, and laboratory tests)	189	97	Decision Tree	F1 = 97.9
4	[29]	20 features (medical, physical exam, ECG, echocardiography, and troponin histories)	228	50	SVM, ANN, and naïve bayes	F1 = 98.9
5	[9]	13 features (medical, physical exam, laboratory tests, and ECG histories)	444	31	Random forest	Acc = 83.45
6	This Study	13 features (medical, physical exam, laboratory tests, and ECG histories)	480	28.7	Random forest	F1 = 85.34

IV. Conclusion

In this study, we addressed the problem of data imbalance in the ACS classification case by using the K-Means SMOTE algorithm to oversample the training data. Our simulations showed that all models built using K-Means SMOTE oversampling data increased F1 scores in all scenarios, with an average increase of 10.07%. We also compared the performance of other oversampling algorithms and found that K-Means SMOTE had the most significant increase in F1 scores.

Our study's findings suggest that oversampling algorithms can improve the output of machine learning models on imbalanced ACS datasets. However, we acknowledge that our research has some limitations, such as using only one dataset and an oversampling algorithm. Therefore, future research could explore other oversampling algorithms, feature engineering processes, and advanced machine learning algorithms to improve the output of these models further.

In conclusion, our research provides insight into the use of oversampling algorithms to address data imbalance in the ACS classification case. Our findings can be used as a foundation for future research to improve the output of machine-learning models on imbalanced ACS datasets.

Acknowledgment

This work was supported by the PPMI KSE Research Group grant, School of Electrical Engineering and Informatics, Institut Teknologi Bandung (grant number 799a/IT1.C12/KU/2022). We thank the reviewers for their valuable comments and suggestions to improve our manuscript.

References

[1] E. A. Amsterdam et al., "2014 AHA/ACC guideline for the management of patients with non-ST-elevation acute coronary syndromes: executive summary: a report of the American College of Cardiology/American Heart Association Task Force on Practice Guidelines," Circulation, vol. 130, no. 25, pp. 2354–2394, 2014.

[2] P. Libby, G. Pasterkamp, F. Crea, and I. K. Jang, "Reassessing the Mechanisms of Acute Coronary Syndromes: The 'vulnerable Plaque' and Superficial Erosion," Circulation Research, vol. 124, no. 1, pp. 150–160, 2019.

[3] N. Makki, T. M. Brennan, and S. Girotra, "Acute coronary syndrome," Journal of Intensive Care Medicine, vol. 30, no. 4, pp. 186–200, 2015.

[4] E. A. Dziedzic, J. S. Gasiorą, A. Tuzimek, M. Dabrowskią, and P. Jankowski, "Neutrophil-to-Lymphocyte Ratio Is Not Associated with Severity of Coronary Artery Disease and Is Not Correlated with Vitamin D Level in Patients with a History of an Acute Coronary Syndrome," Biology, vol. 11, no. 7, pp. 1–12, 2022.

[5] P. A. Iannattone, X. Zhao, J. VanHouten, A. Garg, and T. Huynh, "Artificial Intelligence for Diagnosis of Acute Coronary Syndromes: A Meta-analysis of Machine Learning Approaches," Canadian Journal of Cardiology, vol. 36, no. 4, pp. 577–583, 2020.

[6] M. F. Muzakki, J. A. Utama, R. Priyatikanto, and L. S. Riza, "Detection System of Solar Flare Occurrence in PROBA2 SWAP Images Using Seeded Region Growing and Machine Learning," vol. 62, no. 07, pp. 3329–3342, 2020.

[7] W. G. Baxt, F. S. Shofer, F. D. Sites, and J. E. Hollander, "A neural network aid for the early diagnosis of cardiac ischemia in patients presenting to the emergency department with chest pain," Annals of Emergency Medicine, vol. 40, no. 6, pp. 575–583, 2002.

[8] A. M. Bulgiba and M. Razaz, "How well can signs and symptoms predict AMI in the Malaysian population?," International Journal of Cardiology, vol. 102, no. 1, pp. 87–93, 2005.

[9] E. P. Cynthia, M. Afif Rizky A., A. Nazir, and F. Syafria, "Random Forest Algorithm to Investigate the Case of Acute Coronary Syndrome," Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), vol. 5, no. 2, pp. 369–378, 2021.

[10] S. Calderon-ramirez et al., "Correcting data imbalance for semi-supervised COVID-19 detection using X-ray chest images," Applied Soft Computing, vol. 111, p. 107692, 2021.

[11] V. Karia, W. Zhang, A. Naeim, and R. Ramezani, "Gensample: A genetic algorithm for oversampling in imbalanced datasets," arXiv preprint arXiv:1910.10806, 2019.

[12] R. Mohammed, J. Rawashdeh, and M. Abdullah, "Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results," 2020 11th International Conference on Information and Communication Systems, ICICS 2020, pp. 243–248, 2020.

[13] F. Last, G. Douzas, and F. Bacao, "Oversampling for Imbalanced Learning Based on K-Means and SMOTE," pp. 1–19, 2017.

[14] X. W. Liang, A. P. Jiang, T. Li, Y. Y. Xue, and G. T. Wang, "LR-SMOTE—An improved unbalanced data set oversampling based on K-means and SVM," Knowledge-Based Systems, vol. 196, p. 105845, 2020.

[15] Q. Wang, L. Li, B. Jiang, Z. Lu, J. Liu, and S. Jian, "Malicious domain detection based on k-means and smote," in International Conference on Computational Science, 2020, pp. 468–481.

[16] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," Journal of Artificial Intelligence, vol. 16, pp. 321–357, 2002.

[17] L. Breiman, "Random forests," Machine learning, vol. 45, no. 1, pp. 5–32, 2001.

[18] K. Zhang, X. Wu, R. Niu, K. Yang, and L. Zhao, "The assessment of landslide susceptibility mapping using random forest and decision tree methods in the Three Gorges Reservoir area, China," Environmental Earth Sciences, vol. 76, no. 11, 2017.

[19] R. G. Leiva, A. F. Anta, V. Mancuso, and P. Casari, "A novel hyperparameter-free approach to decision tree construction that avoids overfitting by design," IEEE Access, vol. 7, pp. 99978–99987, 2019.

[20] M. Tschannen, O. Bachem, and M. Lucic, "Recent Advances in Autoencoder-Based Representation Learning," no. NeurIPS, pp. 1–25, 2018.

[21] W. Xu and Y. Tan, "Semisupervised Text Classification by Variational Autoencoder," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 1, pp. 295–308, 2020.

[22] S. Nembrini, I. R. König, and M. N. Wright, "The revival of the Gini importance?," Bioinformatics, vol. 34, no. 21, pp. 3711–3718, 2018.

[23] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in 2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence), 2008, pp. 1322–1328.

[24] H. Lee, J. Kim, and S. Kim, "Gaussian-based SMOTE algorithm for solving skewed class distributions," International Journal of Fuzzy Logic and Intelligent Systems, vol. 17, no. 4, pp. 229–234, 2017.

[25] L. Ma and S. Fan, "CURE-SMOTE algorithm and hybrid algorithm for feature selection and parameter optimization based on random forests," BMC bioinformatics, vol. 18, no. 1, pp. 1–18, 2017.

[26] F. R. Torres, J. A. Carrasco-Ochoa, and J. F. Mart\'\inez-Trinidad, "SMOTE-D a deterministic version of SMOTE," in Mexican Conference on Pattern Recognition, 2016, pp. 177–188.

[27] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," in international conference on intelligent computing, pp. 878–887, 2005.

[28] S. H. Ha and S. H. Joo, "A hybrid data mining method for the medical classification of chest pain," International Journal of Computer and Information Engineering, vol. 4, no. 1, pp. 99–104, 2010.

[29] G. B. Berikol, O. Yildiz, and T. Özcan, "Diagnosis of Acute Coronary Syndrome with a Support Vector Machine," Journal of Medical Systems, vol. 40, no. 4, pp. 1–8, 2016.

[30] R. D. Prayogo and S. A. Karimah, "Feature Selection and Adaptive Synthetic Sampling Approach for Optimizing Online Shopper Purchase Intent Prediction," 2021.

[31] M. P. Perme and D. Manevski, "Confidence intervals for the Mann–Whitney test," Statistical Methods in Medical Research, vol. 28, no. 12, pp. 3755–3768, 2019.