Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

Handling Imbalanced Data Sets Using SMOTE and ADASYN to Improve Classification Performance of Ecoli Data Sets

Anthony Mas Halim*, Mahendra Dwifebri P, Fhira Nhita

Fakultas Informatika, Informatika, Telkom University, Bandung, Indonesia
Email: [1,*]anthonymashalim@student.telkomuniversity.ac.id, [2]mahendradp@telkomuniversity.ac.id,
[3]fhiranhita@telkomuniversity.ac.id
Correspondence Author Email: [1]anthonymashalim@student.telkomuniversity.ac.id
Submitted: **14/06/2023**; Accepted: **27/06/2023**; Published**: 29/06/2023**

Abstract−In this digital era, machine learning is a technology that is in demand by organizations and individuals. In the data and digital information age, the ability to process data efficiently is needed. As the amount of data grows, machine learning has various problems. One of them is that a class imbalance is also often found with the increasing amount of data. Class imbalance is a condition where a class dominates another class. One example is when the positive value class has fewer numbers than the negative class. The class that is less in number is categorized as the minority class, while the class that dominates the data set is called the majority class. Class imbalance can affect classification performance incorrectly, so handling imbalanced classes is needed to improve classification results. Classification of imbalanced data using Random Forest has satisfactory results, as well as implementing SMOTE and ADASYN as sampling methods because they are prevalent and easy to implement. In this research, we use ecoli protein data set to evaluate the performance of random forest classifier with and without oversampling methods. In this research, we used the f1 score and balanced accuracy as the primary evaluation metrics. We calculated the average with the highest score of 84% for the f1 score and 90% for balanced accuracy. Both SMOTE and ADASYN perform similarly to improve the classification performance and found that balanced accuracy is a better-suited metric for imbalanced classification.

Keywords: Imbalanced Data; Random Forest; Imbalanced Ratio; SMOTE;ADASYN

1. INTRODUCTION

The world has entered the digital world. Various technologies, including the computer, facilitate various human activities. Various organizations, groups, and even individuals compete to get the latest and best information. Often the method used to obtain this information could be more effective. To make information processing more effective, one of the steps taken is using machine learning. However, sometimes the information obtained could be in its better form, and sometimes the data obtained still needs to be more efficient; this is often found in large data sets. These large data sets usually contain imbalanced classes, which may affect classification performance[1].

Data sets that contain significant differences between classes with very few instances, known as minor classes [2], and classes with sufficient instances, known as major classes, are considered imbalanced data sets. Synthetic Minority Oversampling Technique (SMOTE) is a frequently used oversampling technique to handle imbalance class problems and is also deemed a very successful technique to generate synthetic data [3]. A simple explanation of how SMOTE works. This technique starts with selecting an instance from the minority class to be selected as a point, identifying its K-nearest neighbors, then creating a line to its K-nearest neighbor. Points between this line are what is considered synthetic observation. As SMOTE generates synthetic observation on all the minority class data, this affects the original data distribution of the minority class. As an alternative to mitigate such a problem, ADASYN is proposed as a possible solution [4]. Adasyn generates synthetic observations based on a weighing technique based on more complex data points of the minority class that are harder to classify.

The classification method we are using is Random Forest Classifier. In order to create an efficient classification model, our research includes data resampling with SMOTE and ADASYN. The purpose of this research is to compare the effect of imbalanced data sets on classification performance and how an effective oversampling method can improve the performance of the model that has been built.

In 2019, Brandt et al.[4] , research to compare SMOTE and ADASYN in imbalanced data classification. The data set used in this research is the credit card fraud data set collected from Kaggle.com. This research uses random forest classifier to create the machine learning model. SMOTE and ADASYN perform efficiently for RFC, as they gain an increase in Sensitivity by 2.99% and 2.57%, respectively, for SMOTE and ADASYN. In this research, SMOTE deemed to perform better than ADASYN, with a higher increase in performance.

In 2019, Gameng et al. [5] performed research on a modified adaptive synthetic SMOTE for imbalanced data sets. The primary data set in this research is from an open admission program of a state college and random forest classifier applied with SMOTE and modifier ADASYN. Gameng et al. conclude that a modified ADASYN performs best when evaluated on four performance metrics (Accuracy, Precision, Recall, and F1-Score).

In 2021, Syaliman et al. [2] discussed enhancing machine learning classification performance accuracy. One example data set used is E-coli data sets have eight classes, and the approach to handle imbalanced Data is SMOTE, Gain Ratio, and the classification model used is K-NN. The proposed method performed better than the K-NN classification without SMOTE and GR, with an increase of 11.4% accuracy.

In 2021, Ramadhan et al.[6] , assessed analysis of SVM classification combined with oversampling methods SMOTE and ADASYN. The example data set used is based on diabetes examination results of Karya Medika

Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

Laboratory, that have nine classes. As a technical data set resulting in parameters that are difficult for the public to understand, another problem is the massive difference in the data for diabetics and non-diabetics. The model achieves 83% without applying oversampling methods. After applying oversampling methods, the model gained higher accuracy at 85.4% with SMOTE and 87.3% with ADASYN, with SMOTE having more errors in predicting a False Negative value.

In this research, topics, and limitations are applied to find the performance of the model by applying several research scenarios. The limitation of this research is that classification using Random Forest (RF) is carried out on imbalanced data, then applying SMOTE / ADASYN to return to RF classification. The data used is the e-coli data set obtained from the KEEL website. The data taken has an imbalanced ratio range of nine to thirteen. In this study, there are two objectives, namely, to see how the performance and accuracy of the model in analyzing imbalanced data using the SMOTE oversampling method and to compare with the ADASYN oversampling method.

The handling imbalanced data set with SMOTE and ADASYN tries to solve the imbalanced classification problems. We applied the oversampling technique to improve the class situation on the E coli data set. Previous research rarely compares the raw performance of the oversampling technique, especially on the E coli data set. We show how SMOTE and ADASYN work and evaluate how each sampling method affects the classification performance. In our research, we use the balanced accuracy score as a comparable metric that covers the whole performance of our classifier. This research aims to get a comparable result on which sampling methods work best and does every data set need to be resampled to get the best classification performance.

2. RESEARCH METHODOLOGY

2.1 Research Stages

The system built in this research is a classification system using random forest classifier on the e-coli data set. As one of the scenarios in this research, the SMOTE and ADASYN oversampling methods are also applied. The flow of this research system is depicted in Figure 1 below:

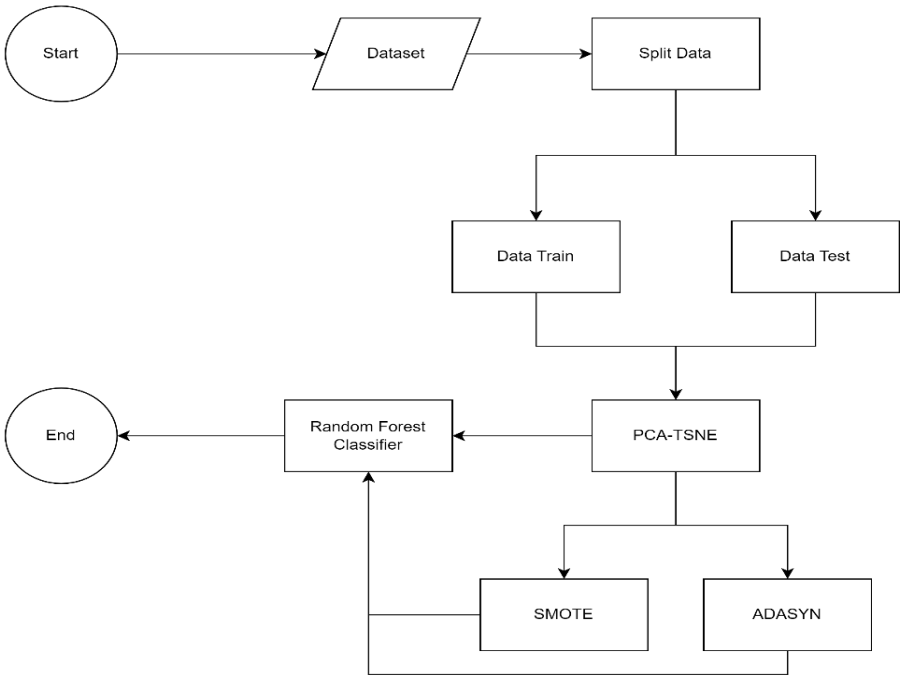

Figure 1. Research Stages

2.2 Data set

In this study, the data set used is the e-coli data set obtained from the KEEL data set repository website [7]. The e-coli data set represents classification problems for protein prediction obtained through microbiological data on Escherichia Coli (E.Coli) bacteria. The data used are five data sets with different Imbalanced Ratio ranging from nine to thirteen. In this research, positive classes are labeled as one (1), and negative classes are labeled as zero (0). The details of the data set can be seen in Table 1 below:

Table 1. Data Set Description

Data set	#Feature	#Instances	Positive Instances (%)	Negative Instances (%)	Imbalanced Ratio (IR)	Acronym

Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

Data set 1	6	203	9.85	90.15	9.15	DS1
Data set 2	6	220	9.09	90.91	10	DS2
Data set 3	6	240	8.33	91.67	11	DS3
Data set 4	6	332	7.53	92.47	12.28	DS4
Data set 5	6	280	7.14	92.86	13	DS5

2.2 Data splitting

After all, data is collected, the first stage is to split the data into two, namely, Train Data and Test Data. This study uses a split ratio of 20% test data and 80% train data. Details of the data that has been split can be seen in the Table 2 below:

Table 2. Total Train and Test Data

Data set	#Instances		#Positive Instances		#Negative Instances	
	Train	Test	Train	Test	Train	Test
DS1	162	41	17	3	145	38
DS2	176	44	16	4	160	40
DS3	192	48	13	7	179	41
DS4	265	67	20	5	245	62
DS5	224	56	16	4	208	52

2.3 Oversampling

Oversampling is one technique to overcome the class imbalance in a data set. Oversampling in the minority class aims to balance the distribution for each repeated data in the minority class. In 2019, Yahaya et al.[8] stated that a training data with huge number of noises can drastically affect classification performance. One way to overcome this problem is to reduce class imbalance by oversampling the minority class. This research will only apply oversampling methods to the training data set. This is done because both oversampling techniques might make exact copies that, if applied to the entire data set, may result in certain instances appearing both in training and testing data set that leads to biased classification [9].

2.3.1 SMOTE

SMOTE is the most used and popular sampling technique and is considered a successful technique[10]. SMOTE is an approach to minority classes. The fewer classes will be oversampled by creating data clones from existing ones. SMOTE creates additional training data by taking samples from each minority class and forming new samples based on the k minority class nearest neighbors. The synthetic classification method provides a better-created decision tree classification method. SMOTE is able to reduce the chance of overfitting than a simple random oversampling methods [11].The steps on how SMOTE works can be seen on Table 3 below:

Table 3. SMOTE Algorithm

No.	Process
1	Identify a point from the minority class
2	Identify the number of synthetic samples needed to generate
3	Identify the number of nearest neighbors to consider
4	Calculate the K-NN using Euclidean distance from the minority class samples, then select one of the K-NN randomly
5	Take the vector difference between the selected point and the nearest neighbor, then multiply the difference with a random integer between 0 and 1
6	Identify the new point on the line segment by adding the random integer to the selected point
7	Repeat the process for every identified point to satisfy the number of synthetic samples needed.

2.3.2 ADASYN

ADASYN is a minority class approach that has similarities with SMOTE. ADASYN differs from SMOTE in the number of samples created. ADASYN will sample more minority classes within the k-nearest neighbor area. ADASYN uses weighted distribution for each minority class sample based on each class learning difficulty[12]. ADASYN works by calculating the degree of class imbalance, then calculating the number of synthetic data examples that are needed to be generated. The steps on how ADASYN works can be seen on Table 4 below:

Table 4. ADASYN Algorithm

No.	Process
1	Calculate the ratio of minority instances to majority instances (d)
2	Calculate the total number of synthetic observations to generate (G)

Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

3	Find the K-NN for each minority point and calculate a value that indicates how many of the neighbors come from the majority class (r_i)
4	Apply normalization of the r_i to make it equal to 1
5	Calculate the amount of synthetic observation to generate for each neighborhood (G_i)
6	Generate G_i number of data for each neighborhood, then generate the new synthetic observation (S_i)

$$d = \frac{m_s}{m_l} \quad (1)$$

Description :
d = Ratio minority instances to majority instances
m_s = #Minority instances
m_l = #Majority instances

$$G = (m_l - m_s)\beta \quad (2)$$

Description :
G = #Synthetic data to generate
m_s = #Minority instances
m_l = #Majority instances
β = Desired d value (1)

$$r_i = \frac{\#Majority}{k} \quad (3)$$

Description :
r_i = #Of neighbors that come from the majority class
K = #Of desired K-NN

$$\hat{r}_i = \frac{r_i}{\sum r_i} \quad (4)$$

Description :
$\hat{r}_i$ = Normalized r_i Value
r_i = #Of neighbors that come from the majority class

$$G_i = G\hat{r}_i \quad (5)$$

Description :
G_i = amount of synthetic observation to generate for each neighborhood
$\hat{r}_i$ = Normalized r_i Value

$$s_i = (x_i + x_{zi} - x_i)\lambda \quad (6)$$

Description :
s_i = New synthetic observation/data
x_i = Minority samples of a neighborhood
x_{zi} = Randomly selected minority example from the same neighborhood
λ = Randomly generated integer between zero and one

2.4 Random Forest

Random forest is an ensemble learning model. Multiple models are trained and combined to make predictions. In random forests, those individual models are known as decision trees [13]. A decision tree works by predicting the designated rule of sequences on the input features. Each decision tree in a random forest is trained by randomly selecting a subset of the original data set. The training process recursively splits data based on the selected feature until reaching a leaf node that represents class labels. Each decision tree makes individual predictions based on the trained model in the prediction process. Each of these predictions is considered a vote for a particular label. RF Classifier is a popular model due to its ability to handle complex tasks like missing values, and imbalanced data set [14]s. The steps on how Random Forest Classifier can be seen on Table 5 below:

Table 5. Random Forest Algorithm

No.	Process

Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

1	Random Forest (RF) accept input of a labelled data set with that has been split into training set and testing set
2	RF run a bootstrap sampling to randomly select subsets of training data to create bootstrap samples
3	For every bootstrap sample created, RF construct a decision tree.
4	Random forest select a random subset of features from the total feature
5	Once the decision tree construction finished, RF made a prediction by each tree on the testing set.

2.5 Evaluation

The confusion matrix is used as an evaluation metric in this research to find factual information and classification prediction results [15]. The confusion matrix is one of the evaluation methods to measure the performance of the classification system. The confusion matrix table can be seen in Table 6 below:

Table 6. Confusion matrix

	Predicted Negative	Predicted Positive
Actual Negative	TN	FP
Actual Positive	FN	TP

Description:
TP = positive predicted data and positive factual data (true positive)
FN = negative predictive data and positive factual data (false negative)
FP = positive predictive data and negative factual data (false positive)
TN = negative predicted data and negative factual data (true negative)

To measure the performance of the classification model built, the information obtained from the confusion matrix will be used to calculate the precision, recall, and F1-Score values. The formula for calculating the evaluation value is:
Precision measures the accuracy of predicting positive value by considering all positive value such as TP and FP [16]. The formula for Precision Score is as follows:

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

Recall measures the accuracy of predicting all positive values. The recall is obtained by dividing TP by all positively classified data [17]. The formula is as follows:

$$Recall = \frac{TP}{TP+FN} \quad (8)$$

F1 Score is the harmonic mean of precision and recall, where FP and FN are considered equally weighted [18]. The formula of the F1 Score is as follows:

$$F1\ Score = \frac{2 \times (recall \times precision)}{recall+precision} \quad (9)$$

Specificity measures the accuracy of predicting all negative values [19] (true-negative rate), obtained by dividing TN by all negatively classified data. The formula is as follows:

$$Specificity = \frac{TN}{TN+FP} \quad (10)$$

Balanced accuracy(BA) is an arithmetic mean of specificity and recall [20]. BA is gained by averaging recall and specificity as the formula is shown:

$$Balanced\ Accuracy = \frac{Recall+Specificity}{2} \quad (11)$$

3. RESULT AND DISCUSSION

Evaluation in this research is testing the classification model that has been built. The evaluation will refer to several predetermined metrics and balanced accuracy, and the f1-score is chosen as the main comparable score. The system starts by splitting the data with a ratio of 80% for the training set and 20% for the testing set. In this study, three scenarios are applied. The first scenario is to compare the performance of the model against 5 data sets without using the oversampling method and classification with a random forest classifier. The second scenario compares the performance of the model against 5 data sets that have been applied SMOTE oversampling and classification with a random forest classifier. The third scenario compares the performance of the model against 5 data sets that have been applied ADASYN oversampling and classification with a random forest classifier. The detailed description of the scenario can be seen on table 7 below:

Table 7. Research Scenario

No.	Scenario Description

Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

1	Performance evaluation of random forest classifier without oversampling methods
2	Performance evaluation of random forest classifier combined with SMOTE oversampling method
3	Performance evaluation of random forest classifier combined with ADASYN oversampling method

3.1 Performance result without oversampling method

In the first scenario, the goal is to compare five data sets without using undersampling. The results of the first scenario can be seen in Table 8 below:

Table 8. Results of 1st Scenario

Data set	Precision	Recall	Specificity	F1-Score	Balanced Accuracy
DS1	66%	66%	97.30%	66%	82.01%
DS2	**100%**	75%	**100%**	85.7%	87.50%
DS3	**100%**	85.71%	**100%**	92.3%	92.85%
DS4	**100%**	60%	**100%**	75%	80%
DS5	**100%**	**100%**	**100%**	**100%**	**100%**
Average	93%	77%	99%	84%	88%

3.2 Performance result with SMOTE oversampling method

The details of the data distribution before and after applying SMOTE can be seen in Table 9 below:

Table 9. Data Set Distribution Before and After SMOTE

Data set	Training Data Before SMOTE		Training Data After SMOTE	
	#Positive	#Negative	#Positive	#Negative
DS1	17	145	145	145
DS2	16	160	160	160
DS3	13	179	179	179
DS4	20	245	245	245
DS5	16	208	208	208

In the second scenario, the goal is to compare five data sets using SMOTE oversampling method. The results of the second scenario can be seen in Table 10 below:

Table 10. Results of 2nd Scenario

Data set	Precision	Recall	Specificity	F1-Score	Balanced Accuracy
DS1	75%	**100%**	97.30%	85.71%	98.60%
DS2	80%	**100%**	97.50%	**88.88%**	**98.75%**
DS3	**100%**	71.40%	**100%**	83.3%	85.71%
DS4	66%	40%	98.30%	50%	69.19%
DS5	66%	**100%**	96.10%	80%	98.07%
Average	77%	82%	98%	78%	90%

3.3 Performance result with ADASYN oversampling method

The details of the data distribution before and after applying ADASYN can be seen in Table 11 below:

Table 11. Data Set Distribution Before and After ADASYN

Data set	Training Data Before ADAYSN		Training Data set After ADASYN	
	#Positive	#Negative	#Positive	#Negative
DS1	17	145	146	145
DS2	16	160	166	160
DS3	13	179	179	179
DS4	20	245	248	245
DS5	16	208	209	208

In the third scenario, the goal is to compare five data sets using ADASYN oversampling method. The results of the third scenario can be seen in Table 12 below:

Table 12. Results of 3rd Scenario

Data set	Precision	Recall	Specificity	F1-Score	Balanced Accuracy
DS1	75%	**100%**	97.30%	85.7 1%	98.68%

Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

DS2	80%	**100%**	97.50%	**88.88%**	98.75%
DS3	**100%**	71.4%	**100%**	83.3%	85.71%
DS4	50%	40%	96.70%	44.4%	68.38%
DS5	80%	**100%**	98.07%	**88.8%**	**99.03%**
Average	77%	82%	98%	76%	90%

3.4 Analysis of experiment result

We compare the performance of imbalanced data with the oversampling method. We use two main performance metrics, f1-score and balanced accuracy. Based on the f1 score, as seen in Figure 2, DS1 and DS2 experience an increase in F1 scores when SMOTE and ADASYN are applied. However, in DS3, the effect of SMOTE and ADASYN is less significant and even less for DS4 and DS5. Imbalanced classification without sampling methods has a higher f1 score than the balanced classification, even though IR is not too far apart between data sets.

For DS1, DS2, and DS3, SMOTE and ADASYN have similar performance. Furthermore, for DS4, SMOTE perform slightly better than ADASYN, and for DS5, ADASYN perform slightly better than SMOTE. With a slight difference in imbalanced ratio, and DS3, DS4, and DS5 are data sets with higher IR, SMOTE and ADASYN perform less significantly than when applied on DS1 and DS2 with lower IR. From this result, we conclude that based on the F1 score, IR is not the only parameter to decide if a data set needs oversampling. The figure 2 below shows F1 score of all scenarios on all five data sets.

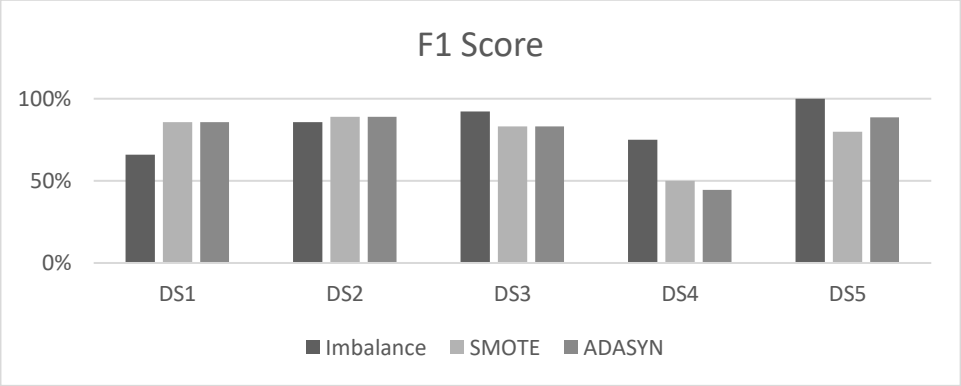

Figure 2. F1 Score Results

Analysis based on balanced accuracy, as seen in Figure 3. DS1 and DS2 SMOTE and ADASYN experience an increase in balanced accuracy when we apply SMOTE and ADASYN. However, less significant for DS3 and DS4, DS5 experienced a slight decrease in balanced accuracy.

For DS1, DS2, and DS3, SMOTE and ADASYN have similar performance, SMOTE performs better than ADASYN in DS4, and ADASYN performs better than SMOTE in DS5. With a slight difference in imbalanced ratio, and DS3, DS4, and DS5 are data sets with higher IR, SMOTE and ADASYN perform less significantly than when applied on DS1 and DS2 with lower IR. From this result, we conclude that based on balanced accuracy, IR is not the only parameter to decide if a data set needs oversampling. The figure 3 below shows balanced accuracy of all scenarios on all five data sets.

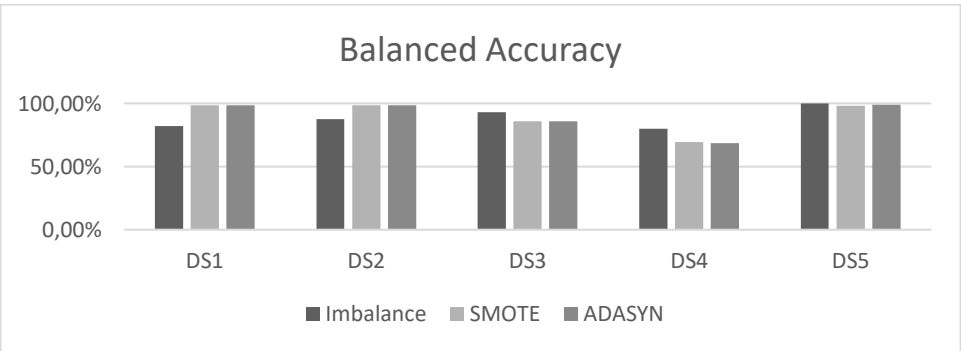

Figure 3. Balanced Accuracy Results

Based on Table 11, imbalanced classification has a better average in the f1 score, and balanced accuracy has a better average for balanced classification. We conclude that a balanced accuracy score is a better-suited performance metric to evaluate imbalanced data sets. SMOTE and ADASYN have a higher average for balanced accuracy and are performing better than imbalance classification, with an average of balanced accuracy of 90% both for SMOTE and ADASYN. The average of the f1 score and balanced accuracy can be seen on table 13 below:

Building of Informatics, Technology and Science (BITS)
Volume 5, No 1, June 2023 Page: 246−253
ISSN 2684-8910 (media cetak)
ISSN 2685-3310 (media online)
DOI 10.47065/bits.v5i1.3647

Table 13. Average of Balanced Accuracy and F1 Score

Average	Imbalanced	SMOTE	ADASYN
F1 Score	84%	78%	76%
Balanced	88%	90%	90%

4. CONCLUSION

We have several conclusions based on the analysis carried out in this research on Handling Imbalanced Data Sets Using SMOTE and ADASYN to Improve Minor Class Classification Performance. Based on the f1 score SMOTE and ADASYN performs less significantly. Classification performed better when no oversampling method was applied, with an 84% f1 score, higher than SMOTE and ADASYN at 78% and 76%. Comparing SMOTE and ADASYN performance, both have similar performance and are only slightly better than one another. Based on balanced accuracy, SMOTE and ADASYN perform better with higher averages than imbalanced classification. Classification performs better when the data set is preprocessed with an oversampling method. By the average of balanced accuracy, SMOTE and ADASYN are better at 90% than imbalance classification at 88%. Another conclusion is that an imbalanced ratio is not the only parameter to decide if a data set needs to be resampled. Furthermore, this research concludes that balanced accuracy is a better-suited performance metric for imbalanced learning.

REFERENCES

[1] X. Jiang and Z. Ge, "Data Augmentation Classifier for Imbalanced Fault Classification," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 3, pp. 1206–1217, 2021, doi: 10.1109/TASE.2020.2998467.
[2] K. U. Syaliman, "Enhance the Accuracy of K-Nearest Neighbor (K-Nn) for Unbalanced Class Data Using Synthetic Minority Oversampling Technique (Smote) and Gain Ratio (Gr)," *J. Infokum*, vol. 10, no. 1, pp. 188–195, 2021.
[3] D. Elreedy and A. F. Atiya, "A Comprehensive Analysis of Synthetic Minority Oversampling Technique (SMOTE) for handling class imbalance," *Inf. Sci. (Ny).*, vol. 505, pp. 32–64, 2019, doi: 10.1016/j.ins.2019.07.070.
[4] J. Brandt and E. Lanzén, "A Comparative Review of SMOTE and ADASYN in Imbalanced Data Classification," p. 42, 2020.
[5] H. A. Gameng, B. D. Gerardo, and R. P. Medina, "A Modified Adaptive Synthetic SMOTE Approach in Graduation Success Rate Classification A Modified Adaptive Synthetic SMOTE Approach in Graduation Success Rate Classification," no. December 2019, 2020, doi: 10.30534/ijatcse/2019/63862019.
[6] N. G. Ramadhan, "Comparative Analysis of ADASYN-SVM and SMOTE-SVM Methods on the Detection of Type 2 Diabetes Mellitus," *Sci. J. Informatics*, vol. 8, no. 2, pp. 276–282, 2021, doi: 10.15294/sji.v8i2.32484.
[7] J. Alcalá-Fdez *et al.*, "KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework," *J. Mult. Log. Soft Comput.*, vol. 17, no. 2–3, pp. 255–287, 2011.
[8] S. W. Yahaya, A. Lotfi, and M. Mahmud, "A Consensus Novelty Detection Ensemble Approach for Anomaly Detection in Activities of Daily Living," *Appl. Soft Comput. J.*, vol. 83, p. 105613, 2019, doi: 10.1016/j.asoc.2019.105613.
[9] J. L. P. Lima, D. MacEdo, and C. Zanchettin, "Heartbeat Anomaly Detection using Adversarial Oversampling," *Proc. Int. Jt. Conf. Neural Networks*, vol. 2019-July, no. July, pp. 1–7, 2019, doi: 10.1109/IJCNN.2019.8852242.
[10] P. Soltanzadeh and M. Hashemzadeh, "RCSMOTE: Range-Controlled synthetic minority over-sampling technique for handling the class imbalance problem," *Inf. Sci. (Ny).*, vol. 542, pp. 92–111, 2021, doi: 10.1016/j.ins.2020.07.014.
[11] J. Park, S. Kwon, and S. P. Jeong, "A study on improving turnover intention forecasting by solving imbalanced data problems: focusing on SMOTE and generative adversarial networks," *J. Big Data*, vol. 10, no. 1, 2023, doi: 10.1186/s40537-023-00715-6.
[12] A. O. Technique *et al.*, "DAD-Net : Classification of Alzheimer ' s Disease Using Neural Network," pp. 1–21, 2022.
[13] V. Jackins, S. Vimal, M. Kaliappan, and M. Y. Lee, "AI-based smart prediction of clinical disease using random forest classifier and Naive Bayes," *J. Supercomput.*, vol. 77, no. 5, pp. 5198–5219, 2021, doi: 10.1007/s11227-020-03481-x.
[14] I. Ullah, B. Raza, A. K. Malik, M. Imran, S. U. Islam, and S. W. Kim, "A Churn Prediction Model Using Random Forest: Analysis of Machine Learning Techniques for Churn Prediction and Factor Identification in Telecom Sector," *IEEE Access*, vol. 7, no. c, pp. 60134–60149, 2019, doi: 10.1109/ACCESS.2019.2914999.
[15] I. Prayoga and M. D. P, "Sentiment Analysis on Indonesian Movie Review Using KNN Method With the Implementation of Chi-Square Feature Selection," vol. 7, pp. 369–375, 2023, doi: 10.30865/mib.v7i1.5522.
[16] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, pp. 1–13, 2020, doi: 10.1186/s12864-019-6413-7.
[17] M. Grandini, E. Bagli, and G. Visani, "Metrics for Multi-Class Classification: an Overview," pp. 1–17, 2020, [Online]. Available: http://arxiv.org/abs/2008.05756
[18] R. Arora, C. T. Tsai, K. Tsereteli, P. Kambadur, and Y. Yang, "A semi-Markov structured support vector machine model for high-precision named entity recognition," *ACL 2019 - 57th Annu. Meet. Assoc. Comput. Linguist. Proc. Conf.*, no. 2005, pp. 5862–5866, 2020, doi: 10.18653/v1/p19-1587.
[19] N. Munsch *et al.*, "Diagnostic accuracy of web-based COVID-19 symptom checkers: Comparison study," *J. Med. Internet Res.*, vol. 22, no. 10, 2020, doi: 10.2196/21299.
[20] D. Chicco, N. Tötsch, and G. Jurman, "The matthews correlation coefficient (Mcc) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation," *BioData Min.*, vol. 14, pp. 1–22, 2021, doi: 10.1186/s13040-021-00244-z.