IMPLEMENTATION OF RANDOM UNDER-SAMPLING AND SYNTHETIC MINORITY OVERSAMPLING TECHNIQUE TO EVALUATE THE PERFORMANCE OF THE CLASSIFICATION AND REGRESSION TREE METHOD

Rifandi Pratama Putra K.[1)]
Nurwan[2)]
La Ode Nashar[3)]
Djihad Wungguli[4)]
Siti Nurmardia Abdussamad[5)]
[1,3,5)]Statistics Study Program, Faculty of Mathematics and Natural Sciences, Universitas Negeri Gorontalo, Gorontalo, Indonesia
[2,4)]Mathematics Study Program, Faculty of Mathematics and Natural Sciences, Universitas Negeri Gorontalo, Gorontalo, Indonesia
e-mail: fandykasadi@gmail.com

ABSTRACT

Class imbalance in datasets poses a significant challenge in the application of classification models, including the Classification and Regression Tree (CART) method. This study aims to evaluate the performance of CART combined with two data balancing techniques: Random Under Sampling (RUS) and Synthetic Minority Oversampling Technique (SMOTE). The data set used in this research is the Heart Failure Clinical Records from Kaggle.com, which exhibits an imbalance where the number of deceased patients is 1,568 records (minority class) and the number of survivors is 3,432 records (majority class), with a total of 5,000 records. The RUS technique reduced the total number of records to 2,526, with each class containing 1,263 records. Conversely, after applying SMOTE, the total number of records increased to 5,474, with each class containing 2,737 records. Model performance evaluation was conducted using precision, recall, and F1-score metrics, both before and after implementing data balancing techniques. The results of the study showed that combining CART with SMOTE produced better performance in recognizing the minority class compared to RUS, achieving accuracy and F1-score of 88.203% and 88.195%, respectively. Meanwhile, RUS achieved an accuracy of 86.345% and an F1-score of 86.332%. Therefore, the use of SMOTE improved model accuracy by approximately **1.85%** and F1-score by **1.86%** compared to RUS. This study makes a significant contribution to improving prediction accuracy on imbalanced datasets and enriches scientific references related to the application of the CART method and data balancing techniques.

Keywords: random under sampling (RUS), synthetic minority over-sampling technique (SMOTE), classification and regression tree (CART), data imbalance.

INTRODUCTION

Data has become one of the most important assets in the digital era, used for decision-making across various fields such as business, healthcare, and governance. However, a significant challenge in data analysis is class imbalance, where the minority class contains significantly fewer records (2,654) compared to the majority class (97,345), which can affect the performance of classification models (Arifiyanti & Wahyuni, 2020). Techniques like Random Under Sampling (RUS) and Synthetic Minority Oversampling Technique (SMOTE) are employed to address this issue by balancing the class distribution in the data. RUS reduces the majority data, while SMOTE generates synthetic data for the minority class (Sutoyo & Fadlurrahman, 2020).

Data imbalance occurs when there is a significant difference between the number of samples in the majority and minority classes (Haixiang et al., 2017). This issue is commonly encountered in research involving imbalanced domains, where the classification process can result in a high misclassification rate and potentially overlook the minority class, which is often considered an outlier (Seiffert et al., 2009). An imbalanced dataset has a highly uneven data distribution, with the majority class containing significantly more samples than the minority class. The class with a much larger number of samples is referred to as the majority class, while the class with fewer samples is called the minority class. Minority class data is often mislabelled when processed in classification models. Therefore, resampling techniques such as under sampling or oversampling are necessary to address this issue (Mahmood, 2015).

RUS helps classification models like CART (Classification and Regression Tree) to focus better on minority classes by reducing bias toward the majority. RUS works by randomly removing a portion of the majority data until its proportion matches the minority class, enhancing data balance without adding new information (Saifudin & Wahono, 2015). CART itself is a well-known method capable of forming optimal decision trees based on binary structures (Wu et al., 2008). Conversely, SMOTE is used to enhance minority data by creating new synthetic samples through interpolation. This technique reduces overfitting often found in traditional oversampling methods and improves model performance by increasing the diversity of minority data. SMOTE also retains the original data without deletion, making it suitable for various classification applications, such as medical diagnoses or fraud detection (Shen et al., 2016).

CART is one of the top 10 data mining algorithms that can identify critical variables in determining classification outcomes. Previous studies have shown that CART can achieve an accuracy of up to 91% with short processing times, making it an efficient and reliable tool for various purposes, including analyzing imbalanced datasets (Wijaya, 2019). The method is advantageous for result interpretation and its low error rate (Sartono & Syafitri, 2010).

The related studies demonstrate the effectiveness of the CART method in various contexts. For example, Adhitya et al. (2023) compared CART with Naïve Bayes and found that CART provided higher accuracy and F1-score in customer churn classification. Meanwhile, Irawan & Wahono (2015) used RUS to address class imbalance in software defect prediction using a neural network-based approach, achieving an average AUC of 0.82 and proving the effectiveness of RUS in improving model performance.

The strength of the previous research lies in the direct comparison between the CART and Naïve Bayes methods in customer churn classification, showing CART's superior performance in terms of higher accuracy and F1-score. This makes CART a better choice in business contexts where high-accuracy churn predictions are needed. Additionally, another study highlighted the use of Random Under Sampling (RUS) to address class imbalance in software defect prediction using a neural network, with results demonstrating the effectiveness of RUS in improving model performance, reflected in an average AUC of 0.82.

Jones & Makmun (2021) also conducted a study using the CART method to develop a computer-based application system for classifying hepatitis diagnoses. Their research findings indicated that the CART method could be an effective tool for classifying hepatitis diagnoses, achieving an accuracy rate of 94%. This demonstrates that the CART method can help improve the efficiency and effectiveness of hepatitis diagnosis while providing appropriate treatment recommendations for patients. However, one of the main challenges in using CART is its declining performance when applied to imbalanced datasets.

The dataset used in this study is derived from Kaggle.com, specifically the Heart Failure Clinical Records, which exhibits class imbalance where the number of deceased patients is fewer than the number of survivors. This provides an opportunity to evaluate the performance of methods such as CART+RUS and CART+SMOTE in addressing this imbalance (Chicco & Jurman, 2020).

In this context, the research aims to compare the classification results of imbalanced datasets using the CART+RUS and CART+SMOTE methods. The evaluation is conducted using metrics such as precision, recall, and F1-score, providing a comprehensive overview of the method's performance in handling imbalanced datasets (Prachuabsupakij & Wuttikamonchai, 2016). The advantage of this research over previous studies is the application of two data balancing techniques directly within the CART algorithm, allowing for a comprehensive evaluation of the impact of each technique on model performance. This study is expected to make a practical contribution to enhancing classification model efficiency for various applications while enriching theoretical literature on the CART method and data balancing techniques such as RUS and SMOTE, serving as a reference for future research (Siringoringo, 2018).

METHOD

The methodological framework of this research comprises several critical phases and is guided by approaches outlined in Breiman et al. (2017), Haixiang et al. (2017), and Irawan & Wahono (2015). Initially, the dataset is subjected to preprocessing with class-balancing techniques, specifically Random Under Sampling (RUS) and the Synthetic Minority Oversampling Technique (SMOTE). RUS operates by randomly removing samples from the majority class, thereby reducing its size. In contrast, SMOTE synthetically generates new samples for the minority class by interpolating between existing samples.

Once the class distribution is balanced, the refined datasets are used to train a Classification and Regression Tree (CART) model (Breiman et al., 2017). The CART algorithm identifies decision rules by recursively splitting the dataset based on feature values that yield the best classification performance.

To facilitate modeling and evaluation, programming tools such as R and Python are employed. Performance metrics such as precision, recall, and F1-score are used to evaluate the models. These metrics enable a comparative analysis of the CART model's performance on the original imbalanced dataset versus the balanced datasets created via RUS and SMOTE (Haixiang et al., 2017).

The data set used in this research is characterized by significant class imbalance, with markedly fewer instances of heart failure-related deaths compared to survivals. To mitigate modeling bias and enhance predictive power, balancing methods are integrated into the preprocessing stage.

By applying this approach, the research explores how effectively CART, in combination with class-balancing techniques, can address imbalanced class problems and improve classification accuracy in healthcare-related datasets.

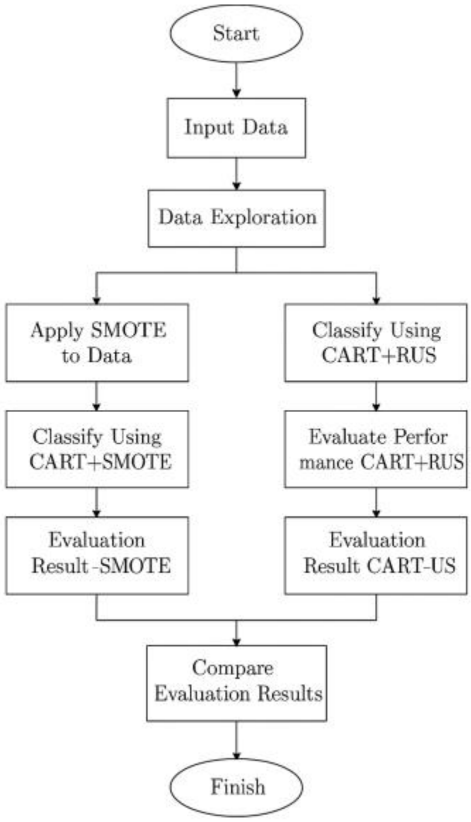

Figure 1. Flow Diagram

RESULTS AND DISCUSSION

Data Explorations

The classification data of 5,000 heart failure patients reveals an imbalance between those who died and those who survived. Based on medical records from Chicco & Jurman (2020), the data is divided into two groups: "Death" and "Survive," with the number of cases in each group shown in the figure below:

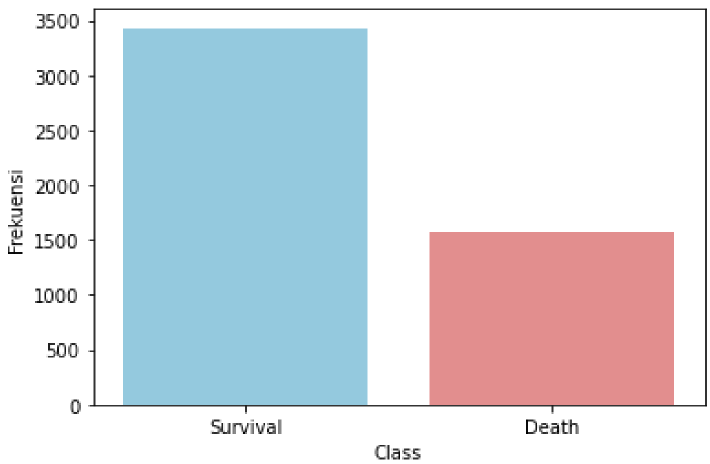

Figure 2. Class Frequency Diagram

The figure above shows an imbalance in the data distribution, where the "Survive" class dominates with 3,432 cases (68.64%), while the "Death" class only accounts for 1,568 cases (31.36%). This imbalance can negatively affect the performance of classification models, particularly in predicting the minority class, unless proper handling methods such as resampling are applied.

RUS Technique

RUS is a widely adopted method in data preprocessing to tackle class imbalance in datasets. This technique involves identifying the gap in sample sizes between the majority and minority classes and subsequently reducing the number of majority class samples through random elimination until both classes reach a comparable size (Saifudin & Wahono, 2015). RUS offers multiple advantages in managing class imbalance, particularly due to its simplicity and ease of implementation. One of its core strengths is its ability to efficiently balance class distributions by discarding excess data from the dominant class. By doing so, it contributes to a more even distribution of instances, which can enhance classification model performance. Moreover, reducing the size of the dominant class also helps lower the likelihood of overfitting, thereby improving the model's ability to generalize to new, unseen data.

The RUS procedure starts by examining the dataset to determine which classes are overrepresented and underrepresented. This step often involves the use of basic statistical tools or visual aids such as histograms to visualize class proportions relative to the target variable. Once this assessment is complete, the algorithm randomly removes samples from the class with more data points until the quantity aligns with that of the underrepresented class. Following this balancing process, a predictive model can then be built and assessed using performance indicators like precision, recall, and the F1-score. Although RUS is capable of improving a model's sensitivity to the minority class, there is a notable trade-off: the process might result in the removal of informative data, which could negatively impact the model's overall accuracy.

RUS is used in this study to address class imbalance in the dataset with two approaches: manual and automated. The automated approach was chosen because it is faster and more efficient, especially for large datasets. The first step is to identify the majority and minority classes in the dataset, as seen in the example where the "Survive" class is the majority with 3432 samples, and the "Death" class is the minority with 1568 samples. Once the majority and minority classes are identified, samples from the majority class are randomly removed until the numbers are balanced with the minority class. The automated approach uses the imbalanced-learn package and the Random Under Sampler algorithm in Python, enabling the process to be carried out more efficiently. The result of the RUS process shows a more balanced class distribution, ready for further analysis.

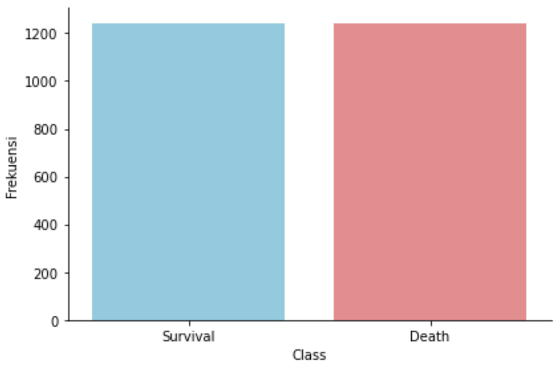

Figure 3. Data after RUS

Based on the image above, the implementation of Random Under Sampling (RUS) successfully balanced the distribution between the SURVIVE and DEATH classes. Each class now contains 1,263 samples, representing an equal proportion of 50% per class from the total of 2,526 data points. This balance eliminates the dominance of the majority class (SURVIVE) and matches it with the minority class (DEATH), allowing the classification model to perform optimally and generate more reliable predictions

The Application of the CART Method on RUS Data

After applying the Random Under Sampling (RUS) technique to the imbalanced data, the next step is to implement the CART method to evaluate the model's performance.

1. Formation of the Classification Tree

In the manual implementation, the researcher calculates the Gini Index or Information Gain to determine the best split point in the decision tree. This process is repeated until a stopping condition is reached. For large datasets, the researcher automates this process using Python with the Decision Tree Classifier, resulting in outcomes as shown:

Table 1. Gini Index of RUS Data

VARIABEL	INDEX GINI
age	0.0367
sex	0.0195
anaemia	0.0277
high_blood_pressure	0.0745
creatinine_phosphokinase	0.0223
ejection_fraction	0.1361
diabetes	0.0468
platelets	0.0493
serum_creatinine	0.0378
serum_sodium	0.0733
smoking	0.0316
time	0.4443

Next, the selection of a splitter or candidate node to become the parent node or root node is carried out using the goodness of split criterion. To calculate the value of goodness of split, the following is obtained:

Table 2. Goodness of Split Data RUS

Calon Simpul	Value
Time	1.6169
High blood pressure	0.8784
Serum sodium	0.8239
Serum creatinine	0.7112
Diabetes	0.6009
Platelets	0.5032
Smoking	0.4761

Calon Simpul	**Value**
Age	0.4326
Anaemia	0.3065
Creatinine phosphokinase	0.2748
Sex	0.2544
Ejection fraction	0.1909

Based on the calculation of the goodness of split values in the table above for each candidate node, it was found that the highest candidate node is **Time** with a value of 1.6169. Therefore, this candidate node will become the parent node. This node will branch into a left branch, where **Time < 0.5**, and a right branch, where **Time > 0.5**.

2. Decision Tree Pruning

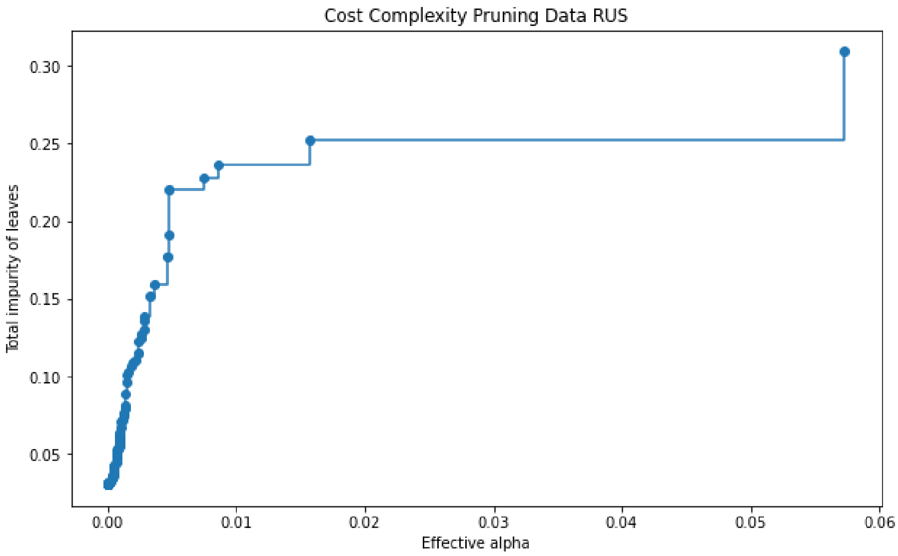

Figure 4. Plot Complexity Parameter

Pruning is performed on the subtree with a minimum complexity parameter value of approximately 0.002, as observed in the plot. This point is selected to reduce tree complexity while maintaining optimal performance. Subsequently, the optimal classification tree will be determined by balancing accuracy and model complexity.

3. Determining Optimal Decision Trees

The pruned decision tree results in a simpler and more efficient model. The variable *Time* serves as the primary splitter with a gini index of 0.5, indicating its most significant role in separating the data into two classes: class 0 (deceased patients) and class 1 (surviving patients). Subsequent splits involve key variables such as *Ejection Fraction*, *High Blood Pressure*, *Age*, and *Serum Sodium*, which contribute to assessing patient conditions. Orange nodes predominantly represent class 0, while blue nodes represent class 1, providing clear visual insights.

With a streamlined structure, the tree is not only easier to interpret but also more effective in identifying key risk factors. The combination of medical factors, such as high blood pressure, serum creatinine levels, and diabetes, offers critical insights into predicting patient outcomes after a heart failure attack. This model serves as a valuable tool for medical analysis and supports improved decision-making in healthcare.

Classification Results of CART + RUS

Table 3. Confusion Matrix

Class	Predictive Positive	Predictive Negative
Actual Positive	220	22
Actual Negative	46	210

$$Accuracy = (220 + 210)/(220 + 22 + 46 + 210) = 0.86345$$
$$Recall = 220/(220 + 22) = 0.90909$$
$$Precision = 220 / (220 + 46) = 0.82706$$
$$F1 - score = 0.86332$$

The confusion matrix for classification results shows that the model correctly predicted 220 cases for the Death class (True Positive) and 210 cases for the Survive class (True Negative). However, 22 instances of the Death class were incorrectly predicted as Survive (False Negative), and 46 instances of the Survive class were misclassified as Death (False Positive). Based on these results, several evaluation metrics were calculated: an accuracy of 86.345%, a recall for the Death class of 90.909%, a precision of 82.706%, and an F1-Score of 86.332%.

The recall value (90.909%) indicates that the model effectively identifies data belonging to the Death class, capturing most positive instances. The precision value (82.706%) reflects the proportion of correct positive predictions made by the model, highlighting cases where some surviving patients were incorrectly classified as deceased (false positives).

The CART model with the RUS technique achieved an accuracy of 86.345% and an F1-Score of 86.332%, demonstrating strong performance in predicting patient outcomes (deceased or surviving). The accuracy metric indicates the model's capability to correctly classify the majority of patients, despite the reduction of majority class data. Meanwhile, the F1-Score highlights the balance between the model's ability to detect deceased patients (recall) and ensure the accuracy of those predictions (precision).

SMOTE

Imbalanced data presents a significant challenge in developing predictive models using machine learning. One common strategy to address this issue is through oversampling the underrepresented class. While helpful, oversampling has some disadvantages, including a higher risk of overfitting and potential loss of meaningful information. A basic form of oversampling typically involves copying existing data from the minority class, which can amplify the overfitting problem, especially in smaller datasets. Alternatively, under sampling works by randomly discarding data from the majority class, but this method risks discarding important data patterns. To overcome these limitations, one advanced solution is to apply the SMOTE, which generates artificial data points to enhance the minority class.

The SMOTE technique begins by identifying the minority class within the dataset. It then calculates how many additional samples are needed to achieve a more balanced class distribution.

Unlike RUS, which lowers the sample count in the dominant class by removing data points, SMOTE boosts the representation of the minority class by generating synthetic instances without eliminating any existing data. After the target number of new samples is determined, SMOTE creates these synthetic data points by identifying the closest neighbors of a given instance in the minority class and generating new values through interpolation between them. Once the new dataset is created, a machine learning model can be trained and evaluated accordingly. A key benefit of using SMOTE is its ability to equalize class proportions without discarding majority class data. However, it should be noted that improper application of SMOTE may still lead to overfitting, especially if the synthetic samples do not adequately reflect data variation.

Following the application of Random Under Sampling (RUS), the researcher implemented SMOTE (Synthetic Minority Over-sampling Technique) to further address the imbalance between class labels. Unlike RUS, which reduces the majority class size by removing samples, SMOTE generates new synthetic samples to increase the representation of the minority class. It does this by interpolating between existing minority class samples to introduce new, varied data points. This method is particularly effective in large datasets, especially when using tools like Python and specialized libraries such as imbalanced-learn. The first step involves clearly defining which class is the majority and which is the minority; in this study, "Survive" was treated as the majority class and "Death" as the minority. SMOTE then enhances the minority class by producing synthetic data to improve its overall representation within the dataset:

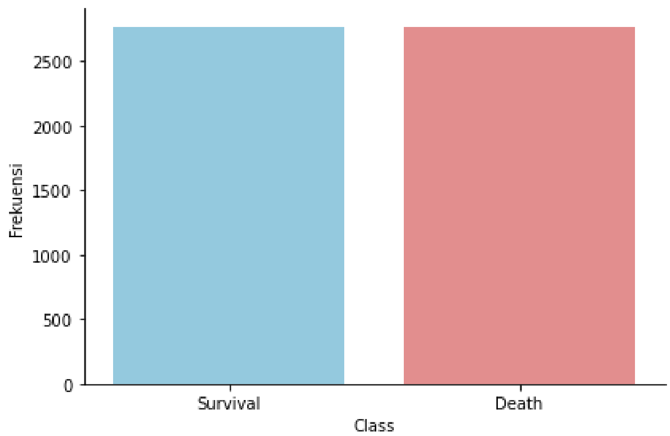

Figure 5. Data After SMOTE

Based on the image above, the application of SMOTE successfully balanced the distribution between the SURVIVE and DEATH classes, with each class now having a total of 2737 data, bringing the total dataset to 5474. SMOTE effectively increased the number of samples in the minority class (DEATH) by generating synthetic data that replicates the pattern of the original samples. With a balanced distribution, the classification model can learn more effectively from both classes, improving overall accuracy and prediction performance. This SMOTE technique ensures that the model is not solely focused on the majority class but is also sensitive to predictions in the minority class, resulting in more accurate and precise outcomes.

Application of the CART Method on SMOTE Data

After applying the SMOTE technique to balance the data, the next step is to apply the CART method using the balanced data. With the SMOTE-generated data, the CART model is trained to build a decision tree that more accurately represents both classes, SURVIVE and DEATH.

1. Formation of the Classification Tree

Similar to the RUS technique, the researcher calculates the Gini Index or Information Gain on the SMOTE-processed data to determine the best split point in the decision tree. This process is repeated until a stopping condition is reached. For large datasets, the researcher automates this process using Python with the Decision Tree Classifier, producing results as shown.

Table 4. Gini Index of SMOTE Data

Variabel	**Gini Index**
age	0.0106
sex	0.0008
anaemia	0.0265
high_blood_pressure	0.0189
creatinine_phosphokinase	0.0090
ejection_fraction	0.1874
diabetes	0.0348
platelets	0.0404
serum_creatinine	0.0058
serum_sodium	0.0547
smoking	0.0003
time	0.6108

Next, the selection of a splitter or candidate node to become the parent node or root node is carried out using the goodness of split criterion. To calculate the value of goodness of split, the following is obtained:

Table 5. Goodness of Split Data SMOTE

Calon Simpul	**Value**
Time	0.2528
High blood pressure	0.2453
Ejection fraction	0.2277
Platelets	0.2026
Serum sodium	0.1574
Age	0.1297
Diabetes	0.1108
Anaemia	0.0301
Serum creatinine	0.0168
Creatinine phosphokinase	0.0139
Sex	0.0053
Smoking	0.0011

Based on the calculation of the goodness of split values in the table above for each candidate node, it was found that the highest candidate node is **Time** with a value of 0.2528. Therefore, this candidate node will become the parent node. This node will branch into a left branch, where **Time < 0.5**, and a right branch, where **Time > 0.5.**

2. Decision Tree Pruning

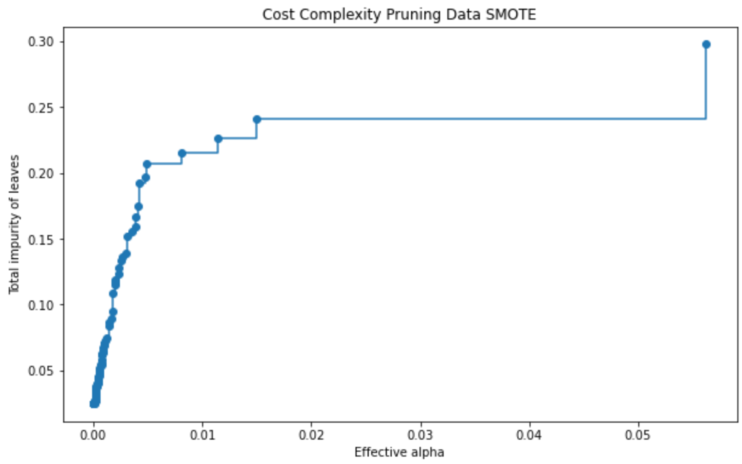

Figure 6. Plot Complexity Parameter

Pruning is performed on the subtree with a minimum complexity parameter value of approximately 0.002, as observed in the plot. This point is selected to reduce tree complexity while maintaining optimal performance. Subsequently, the optimal classification tree will be determined by balancing accuracy and model complexity.

3. Determining Optimal Decision Trees

The decision tree above predicts a patient's condition after a heart attack. Class 0 indicates deceased patients, while class 1 indicates survivors. The classification begins with the variable **Time**, and then splits based on **Ejection Fraction**, **Serum Sodium**, and **High Blood Pressure**. These four variables are key to distinguishing between survival and death outcomes.

In the diagram, **orange nodes** represent a majority of class 0 (deceased), and **blue nodes** represent class 1 (survived). Terminal nodes with **low Gini values** indicate better classification performance and more homogeneous groups. For example, branches dominated by class 0 or class 1 with Gini values near 0 suggest that those splits are effective.

Overall, the model shows that Ejection Fraction, Serum Sodium, and High Blood Pressure strongly influence patient outcomes. This decision tree can support clinical judgment by helping predict post-heart attack mortality risk more accurately and efficiently.

Classification Results of CART + RUS

Table 6. Confusion Matrix

Class	**Predictive Positive**	**Predictive Negative**
Actual Positive	459	73
Actual Negative	57	513

$$Accuracy = (459 + 513)/(459 + 73 + 57 + 513) = 0.88203$$
$$Recall = 459 / (459 + 73) = 0.90909$$
$$Precision = 459 / (459 + 57) = 0.82706$$
$$F1 - score = 0.88195$$

The confusion matrix illustrates that the model accurately identified 459 cases as positive (True Positives) and 513 cases as negative (True Negatives). Meanwhile, 73 actual positive instances were misclassified as negative (False Negatives), and 57 actual negatives were incorrectly predicted as positive (False Positives). From these results, the model recorded an **accuracy of 88.203%**, a **precision of 88.953%**, a **recall of 86.278%**, and an **F1-score of 88.195%**. The recall score of **86.278%** highlights the model's ability to successfully detect the majority of actual Death class instances, which represent the positive class. The precision score of **88.953%** indicates how accurate the model's positive predictions were—though some patients who actually survived were incorrectly marked as deceased (false positives). By using the CART model enhanced with the SMOTE approach, the system attained an **accuracy of 88.203%** and an **F1-score of 88.195%**, indicating strong predictive performance. The high accuracy suggests the model effectively classifies both classes correctly in most cases. However, in the context of imbalanced datasets, accuracy alone may not fully capture model performance. Therefore, the elevated F1-score is especially important, as it reflects a strong balance between recall and precision. This balance demonstrates the model's capability not only to capture most of the positive instances (high recall) but also to ensure the reliability of those positive predictions (high precision). Such performance makes the CART+SMOTE model particularly suitable for applications—like in healthcare—where minimizing prediction errors is essential.

Comparison of CART+RUS and CART+SMOTE Classification Results

The comparison of evaluation results between the CART model formed with the RUS data balancing technique and the CART model formed with the SMOTE data balancing technique can be seen in the previous table. Based on the table, it can be observed that the CART classification with the RUS-processed data yields slightly better results, with an Accuracy of 88.203% and an F-1 Score of 88.195%, compared to the CART model's performance on SMOTE-processed data, which has an Accuracy of 86.345% and an F-1 Score of 86.332%.

Table 7. Comparison of CART+RUS and CART+SMOTE

Model	Accuracy	F1-Score
CART + RUS	86.35%	86.33%
CART + SMOTE	88.20%	88.20%

Based on the comparison between CART+RUS and CART+SMOTE, the CART+SMOTE method performs better, with an accuracy of 88.203% compared to 86.345% for CART+RUS. This suggests that SMOTE performs better in addressing data imbalance by creating artificial samples that better reflect the characteristics of the minority group, thereby achieving a more balanced distribution between class labels. The improved accuracy is further reinforced by evaluation indicators like precision, recall, and F1-score, which demonstrate consistent results, particularly in detecting the minority class (DEATH). Therefore, CART+SMOTE proves to be more effective in producing an accurate and unbiased classification model.

CONCLUSION

According to the study results, applying data balancing methods notably influences the effectiveness of the CART classification algorithm. Using Random Under Sampling (RUS), the model recorded an accuracy of 86.345% alongside an F1-score of 86.332%. Although its performance is lower compared to the SMOTE approach is considered to be more straightforward and efficient in its application Through the elimination of excess data points from the dominant class, RUS enables the algorithm to better concentrate on key patterns in the dataset. On the other hand, SMOTE achieved an accuracy of 88.203% and an F1-score of 88.195%, demonstrating better performance in handling imbalanced data. By generating representative synthetic data, SMOTE helps the model learn patterns from both classes more effectively, making it more suitable for datasets with extreme imbalance.

This study shows that SMOTE is more effective in maintaining model quality for imbalanced datasets, achieving higher accuracy and F1-score compared to RUS. However, the choice of data balancing technique should consider the level of data imbalance and analytical priorities. If simplicity and computational efficiency are required, RUS can be used; but for better model performance in highly imbalanced conditions, SMOTE is more recommended.

REFERENCE

Adhitya, R. R., Wina Witanti, & Rezki Yuniarti. (2023). Perbandingan metode CART dan Naïve Bayes untuk klasifikasi customer churn. *INFOTECH Journal*, *9*(2), 307–318. https://doi.org/10.31949/infotech.v9i2.5641

Arifiyanti, A. A., & Wahyuni, E. D. (2020). SMOTE: Metode penyeimbang kelas pada klasifikasi data mining. *SCAN - Jurnal Teknologi Informasi Dan Komunikasi*, *15*(1), 34–39. https://doi.org/10.33005/scan.v15i1.1850

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (2017). *Classification and regression trees* (1st Edition). Routledge. https://doi.org/10.1201/9781315139470

Chicco, D., & Jurman, G. (2020). Machine learning can predict survival of patients with heart failure from serum creatinine and ejection fraction alone. *BMC Medical Informatics and Decision Making*, *20*(1), 16. https://doi.org/10.1186/s12911-020-1023-5

Haixiang, G., Yijing, L., Shang, J., Mingyun, G., Yuanyue, H., & Bing, G. (2017). Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, *73*, 220–239. https://doi.org/10.1016/j.eswa.2016.12.035

Irawan, E., & Wahono, R. S. (2015). Penggunaan Random Under Sampling untuk penanganan ketidakseimbangan kelas pada prediksi cacat software berbasis neural network. *LmuKomputer.Com Journal of Software Engineering*, *1*(2), 92–100.

Jones, A. H. S., & Makmun, M. S. (2021). Implementasi metode CART untuk klasifikasi diagnosis penyakit hepatitis pada anak. *Journal of Informatics Information System Software Engineering and Applications (INISTA)*, *3*(2), 61–70.

Langguth, F., Sunkavalli, K., Hadap, S., & Goesele, M. (2016). Shading-Aware Multi-view Stereo. In *Lecture Notes in Computer Science: Vol. vol 9907* (pp. 469–485). Springer. https://doi.org/10.1007/978-3-319-46487-9_29

Mahmood, A. M. (2015). Class Imbalance Learning in Data Mining – A Survey. *International Journal of Communication Technology for Social Networking Services*, *3*(2), 17–36. https://doi.org/10.21742/ijctsns.2015.3.2.02

Pourmoradi, N. (2024). *Heart failure clinical records*. https://www.kaggle.com/datasets/nimapourmoradi/heart-failure-clinical-records

Prachuabsupakij, W., & Wuttikamonchai, O. (2016). An intelligent system to predict student's graduation. *International Conference on Science, Technology and Innovation for Sustainable*

Well-Being (STISWB VIII), 1–6.

Saifudin, A., & Wahono, R. S. (2015). Penerapan teknik ensemble untuk menangani ketidakseimbangan kelas pada prediksi cacat software. *Journal of Software Engineering*, *1*(1), 28–37.

Sartono, B., & Syafitri, U. D. (2010). Metode pohon gabungan: solusi pilihan untuk mengatasi kelemahan pohon regresi dan klasifikasi tunggal. *Indonesian Journal of Statistics and Its Applications*, *15*(1), 1–7.

Seiffert, C., Khoshgoftaar, T. M., & Van Hulse, J. (2009). Hybrid sampling for imbalanced data. *Integrated Computer-Aided Engineering*, *16*(3), 193–210. https://doi.org/10.3233/ICA-2009-0314

Siringoringo, R. (2018). Klasifikasi data tidak seimbang menggunakan algoritma SMOTE dan K-Nearest Neighbor. *Journal Information System Development*, *3*(1), 44–49.

Sutoyo, E., & Fadlurrahman, M. A. (2020). Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Television Advertisement Performance Rating Menggunakan Artificial Neural Network. *Jurnal Edukasi Dan Penelitian Informatika (JEPIN)*, *6*(3), 379–385. https://doi.org/10.26418/jp.v6i3.42896

Wijaya, J. (2019). *Implementasi algoritma pohon keputusan cart untuk menentukan klasifikasi data evaluasi mobil* [Thesis (Skripsi)]. Universitas Sanata Dharma.

Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Yu, P. S., Zhou, Z.-H., Steinbach, M., Hand, D. J., & Steinberg, D. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems*, *14*(1), 1–37. https://doi.org/10.1007/s10115-007-0114-2