Decision Tree Classifier

Aditya Bajracharya[1], Projan Shakya[1]
[1]Institute of Engineering, Thapathali Campus, Kathmandu

Corresponding author: Aditya Bajracharya (aditya.bajracharya01@gmail.com), Projan Shakya (projan.shakya@gmail.com).

ABSTRACT This lab report investigates the application of decision tree classifiers for predicting male fertility status based on personal and lifestyle attributes. Using the Fertility dataset from the UCI Machine Learning Repository, we implemented decision tree models in the Jupyter Notebook environment with the scikit-learn library. To address class imbalance, we applied the Synthetic Minority Over-Sampling Technique (SMOTE). Our experiments evaluated the impact of different splitting criteria (Gini impurity and entropy) and maximum tree depths on model performance. The results show that decision trees trained with Gini impurity outperformed those using entropy, achieving higher accuracy and F1-scores. After SMOTE, the optimal tree depth increased significantly, allowing the model to capture more nuanced patterns in the balanced dataset. The decision tree classifier demonstrated promising performance in predicting the minority "Altered" fertility class, highlighting its potential for identifying key factors influencing male fertility. This work underscores the value of decision trees as transparent and interpretable models for medical and health-related applications.

INDEX TERMS Decision tree classifier, Entropy, F1-score, Gini impurity, Imbalanced datasets, SMOTE (Synthetic Minority Over-Sampling Technique), Weighted F1-score

I. INTRODUCTION

Today's world of computers and artificial intelligence has revolutionized decision-making processes, thanks to advanced algorithms like decision tree. With the growing demand for AI-driven solutions, the need for efficient, accurate and understandable models is more crucial than ever.

A decision tree classifier is a type of supervised learning algorithm that is used for both classification and regression tasks. It works by splitting the dataset into subsets based on the value of input features, creating a tree-like model of decisions. Each internal node of the tree represents a feature or attribute, each branch represents a decision rule, and each leaf node represents an outcome or class label.

The simplicity and transparency of decision trees make them highly valuable in various applications. They are particularly useful when dealing with complex datasets where the relationships between features and target variables are not easily discernible. Decision trees can capture non-linear patterns and interactions between variables, making them robust and flexible.

However, decision trees also come with their own set of challenges. They are prone to overfitting, especially when dealing with noisy data or when the tree grows too deep. This can result in poor generalization to new data. To address these issues, techniques such as pruning, ensemble methods (like Random Forests and Gradient Boosting), and cross-validation are often employed.

In essence, decision tree classifiers offer a powerful tool for making informed decisions based on data. Their ability to break down complex decision-making processes into simple, interpretable rules makes them a cornerstone of many machine learning systems. By understanding and mitigating their limitations, we can harness their full potential to build robust and reliable AI models.

II. LITERATURE REVIEW

Decision trees are a popular machine learning algorithm used for both classification and regression tasks. They work by recursively partitioning the feature space into smaller regions based on the most informative features, creating a tree-like model of decisions **[1]**. Decision trees are known for their interpretability, as the decision-making process can be easily visualized and understood.

One key challenge with decision trees is their tendency to overfit, especially when dealing with noisy or imbalanced datasets. Overfitting occurs when the model becomes too complex and captures noise in the training data, leading to poor generalization to new, unseen instances. This is a common issue when the dataset has a skewed class

distribution, where one class is significantly underrepresented compared to the others.

To address the problem of imbalanced datasets, researchers have proposed various techniques. One widely used method is Synthetic Minority Over-Sampling Technique (SMOTE), which generates synthetic examples of the minority class by interpolating between existing minority instances **[2]**. This helps to balance the class distribution and improve the model's ability to learn the decision boundaries for the minority class.

In addition to oversampling techniques, ensemble methods such as Random Forests **[3]** and Gradient Boosting have also been explored to enhance the performance of decision trees on imbalanced datasets. These ensemble approaches combine multiple decision trees, often with different hyperparameters or training strategies, to create a more robust and accurate classifier.

Furthermore, the selection of appropriate evaluation metrics is crucial when dealing with imbalanced datasets. Accuracy, which is the proportion of correctly classified instances, can be misleading in such cases, as the model may achieve high accuracy by simply predicting the majority class. Metrics like precision, recall, and the F1-score, which consider both false positives and false negatives, are more suitable for evaluating the model's performance on imbalanced datasets **[4]**.

By understanding the limitations of decision trees and employing techniques to address class imbalance, researchers have been able to develop more accurate and reliable predictive models in various domains, including medical and health-related applications.

III. METHODOLOGY

A. DATASET DESCRIPTION

We used the Fertility dataset **[5]** from the UCI Machine Learning Repository, designed to predict the fertility status of men based on various personal and lifestyle attributes. This dataset was collected as part of a study to understand the factors affecting male fertility. It consists of 100 instances, each characterized by 10 input features and an output class. The features include:

- Season (categorical).
- Age (integer).
- Childhood diseases (binary).
- Accident or serious trauma (binary).
- Surgical intervention (binary).
- High fever in the last year (categorical).
- Frequency of alcohol consumption (ordinal).
- Smoking habit (categorical).
- Hours spent sitting per day (integer).

The output class indicates the fertility status of the individual, categorized as either fertile or infertile. This dataset was used in the paper [6], to determine the environmental factors, as well as life habits that may affect the semen quality. It serves as an excellent resource for developing predictive models and conducting exploratory data analysis in medical and health-related research, particularly in understanding and identifying key factors influencing male fertility. Its well-defined features and clear output class make it suitable for various machine learning tasks, including classification and regression analysis.

B. PROPOSED METHODOLOGY

The methodology comprises several key steps, including data preprocessing, model training, model evaluation, and result interpretation.

Data preprocessing is essential to ensure that the dataset is suitable for training the decision tree model. This involves handling missing values, encoding categorical variables, and splitting the dataset into training and testing sets 20% data in test set. The dataset did not have any missing values and all the numerical features were already normalized. So, only the target class values were modified for better interpretability.

The target class is imbalanced with more ‘Normal’ than ‘Altered’ data as shown in Figure 1. Hence, we apply SMOTE (Synthetic Minority Over-Sampling Technique) to address this class imbalance, ensuring that the decision tree model receives clean and properly balanced input.

SMOTE is a powerful oversampling method that creates synthetic samples of the minority class by interpolating between existing minority instances as shown in Figure 3. Unlike simple oversampling, which duplicates minority class instances, SMOTE generates new, synthetic examples that lie along the line segments joining a minority instance and its nearest neighbors. This approach not only balances the class distribution but also helps to mitigate overfitting by introducing more variation in the training data. By using SMOTE, we enhance the classifier's ability to learn the decision boundaries for the minority class, which in this case are the 'Altered' fertility status instances. This step is crucial for improving the predictive performance and generalization capability of the decision tree model on imbalanced datasets.

We train the decision tree model using the training dataset and switching the splitting criterion between Gini Impurity and Entropy. The decision tree algorithm recursively splits the data based on features that provide the most information gain or decrease in impurity. During training, the decision tree grows by partitioning the feature space into smaller regions, ultimately forming a tree structure where each internal node represents a decision based on a feature, and each leaf node represents a class label.

To optimize the decision tree model, we use grid search cross-validation. Grid search systematically works through multiple combinations of hyperparameters, cross-validating as it goes to determine the best parameters. Stratified k-fold cross-validation is employed to ensure that each fold has the same proportion of class labels, providing a more reliable assessment of the model's performance. In our grid search, we tune hyperparameters such as maximum depth, minimum samples per split, and minimum samples per leaf.

Before oversampling the minority class, the grid search cross-validation provided the optimal parameters as:

Table 1: Best Parameters Before Oversampling

	Gini Impurity	Entropy
Max_depth	1	1
Min_samples_leaf	2	1
Mean_samples_split	2	2

After oversampling the minority class, the grid search cross-validation provided the optimal parameters as:

Table 2: Best Parameters After Oversampling

	Gini Impurity	Entropy
Max_depth	7	7
Min_samples_leaf	2	2
Mean_samples_split	10	5

We evaluate the model's performance using the testing dataset. Common evaluation metrics for classification tasks include accuracy, precision, recall, and F1-score. By comparing the model's predictions to the actual values in the testing dataset, we assess its ability to generalize to unseen data. Using weighted F1-score ensures that the evaluation metric accounts for the class imbalance in the dataset.

To gain insights into the decision-making process of the trained decision tree, we visualize its structure. Decision tree visualization helps us understand how the model partitions the feature space and makes predictions based on different decision paths. By examining the decision tree structure, we identify important features and understand the underlying patterns in the data.

Finally, we interpret the results of the decision tree analysis and draw conclusions about its performance and suitability for the given task. We discuss the strengths and limitations of the decision tree model and consider potential improvements.

C. MATHEMATICAL FORMULAE

In decision tree, following formulae are used to calculate the impurity of set of samples as the splitting criteria:

Entropy (H): It is a measure of impurity in a set of examples. It calculates the uncertainty or disorder of a set by computing the sum of the probability of each class label multiplied by the logarithm of that probability. Lower entropy indicates less disorder and better purity of the samples.

$$H(S) = -p_0 log_2(p_0) - p_1 log_2(p_1) \quad (1)$$

Where p_0 is the probability of class 0 and p_1 is the probability of class 1. Also, the formula can be generalized as:

$$H(S) = -\sum_{i=1}^{n} p_i log_2(p_i) \quad (2)$$

Where n is the number of classes and p_i is the probability of class i.

Gini Impurity (G): It measures the probability that newly collected, random data would be incorrectly classified if assigned a random class label based on the dataset's class distribution. Like entropy, lower Gini impurity values indicate higher purity and less disorder in the set of samples.

$$G(S) = 1 - p_0^2 - p_1^2 \quad (3)$$

Where p_0 is the probability of class 0 and p_1 is the probability of class 1. Also, the formula can be generalized as:

$$G(S) = 1 - \sum_{i=1}^{n} (p_i)^2 \quad (4)$$

Where n is the number of classes and p_i is the probability of class i.

Classification Error (CE): This measures the proportion of misclassified samples in a node.

$$CE(S) = 1 - max(p_0, p_1) \quad (5)$$

Where p_0 is the probability of class 0 and p_1 is the probability of class 1. Also, the formula can be generalized as:

$$CE(S) = 1 - max(p_0, p_1, \dots, p_n) \quad (6)$$

Where n is the number of classes and p_n is the probability of class n.

Information Gain (IG): It is the determination of the effectiveness of a feature in reducing uncertainty (entropy) in a set of samples.

$$IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} H(S_v) \quad (7)$$

Where IG (S, A) represents the information gain of feature A in dataset S, H(S) is the entropy of the parent node and $H(S_v)$ is the entropy of the child node.

Now, for performance evaluation, we have the following metrics:

Confusion Matrix: It provides a detailed breakdown of true positives, true negatives, false positives, and false negatives. It helps in understanding the performance of a classification model in more detail.

Table 3: Confusion Matrix

		Predicted	
		Positive	Negative
Actual	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

True Positives: Correctly predicted positive instances.
True Negatives: Correctly predicted negative instances.
False Positives: Incorrectly predicted positive instances.
False Negatives: Incorrectly predicted negative instances.

Accuracy: It measures the proportion of correctly classified instances out of the total instances. It is a simple and intuitive measure but may not be sufficient for imbalanced datasets.

$$Accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} \quad (8)$$

Precision: It is the ratio of correctly predicted positive observations to the total predicted positives. It is crucial when the cost of false positives is high.

$$Precision = \frac{True\ positives}{True\ positives + False\ positives} \quad (9)$$

Recall: It is the measure of ability of model to capture all the positive instances. It is important in scenarios where missing a positive instance has a significant cost.

$$Recall = \frac{True\ positives}{True\ positives + False\ Negatives} \quad (10)$$

F1 score: It is the harmonic mean of precision and recall, providing a balanced measure that considers both false positives and false negatives. It is particularly useful when the dataset has imbalanced classes.

$$F1score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (11)$$

Weighted F1 score: It is a metric used in machine learning for evaluating the performance of a classification model when dealing with imbalanced datasets. It takes into account the class distribution and provides a more informative score compared to the standard F1-score.

$$\boldsymbol{Weighted\ F1score} = \sum_{i=1}^{c} \frac{N_i}{N} \boldsymbol{F1_i} \quad (12)$$

Where c is the total number of classes, N is the total number of instances across all classes, N_i is the total number of true instances of class and $F1_i$ is the F1 score of class i.

D. INSTRUMENTATION DETAILS

In this lab, we used the Jupyter Notebook environment, an interactive computing platform, to facilitate the implementation and experimentation with decision tree models for the Fertility dataset. Jupyter Notebook provided us with a user-friendly interface that seamlessly integrates code execution, visualization, and documentation, allowing for an efficient and interactive workflow. Its support for various programming languages, including Python, enabled us to harness the powerful capabilities of the scikit-learn (sklearn) library for machine learning tasks.

Within the Jupyter Notebook environment, we utilized sklearn, a versatile machine learning library in Python, to implement and evaluate decision tree models. Sklearn offers a comprehensive suite of tools and functionalities for building, training, and evaluating machine learning models, making it well-suited for our experimentation with decision trees. By importing relevant modules from the sklearn library, such as DecisionTreeClassifier and plot_tree we seamlessly integrated decision tree functionality into our Python code, streamlining the model development process.

Furthermore, sklearn provided us with access to a rich array of preprocessing techniques, evaluation metrics, and model selection methods, enhancing the robustness and scalability of our experimentation. For instance, we utilized sklearn's train_test_split function to partition the dataset into training and testing sets, enabling us to assess the model's performance on unseen data. Additionally, we applied SMOTE to handle class imbalance in the training set, and used grid search cross-validation to tune hyperparameters effectively. Sklearn's built-in evaluation metrics, such as precision, recall, and the weighted F1 score, facilitated comprehensive performance analysis, empowering us to make informed decisions regarding model selection and parameter tuning.

Overall, the combination of Jupyter Notebook and sklearn served as instrumental tools in our lab experimentation, offering a seamless and efficient environment for implementing, evaluating, and documenting decision tree models for fertility prediction. Through their user-friendly interfaces and extensive functionality, these tools empowered us to explore complex machine learning algorithms with ease, facilitating the advancement of our understanding and proficiency in predictive modeling.

IV. EXPERIMENTAL RESULTS

A. KFOLD CROSS-VALIDATION

The graphs in Figure 4 and Figure 5 illustrate the relationship between the decision tree depth and the weighted F1 score, highlighting the model's performance both before and after applying the Synthetic Minority Over-Sampling Technique (SMOTE) to address class imbalance in the dataset.

Initially, without SMOTE, the weighted F1 score peaks at a very shallow tree depth of 1. This suggests that the decision tree struggles to extract meaningful patterns from the imbalanced data, where the majority class (Normal) significantly outnumbers the minority class (Altered). The peak performance at such a shallow depth indicates that the model may be overfitting to the majority class, and deeper trees fail to improve the F1 score as they start to overfit the sparse minority class examples or further entrench the bias towards the majority class.

After applying SMOTE, the dataset becomes balanced, allowing the decision tree to learn more effectively from the minority class (Altered). The synthetic examples generated by SMOTE help the decision tree capture more nuanced patterns indicative of the minority class. Post-SMOTE, the optimal tree depth increases significantly to 7, as shown by the improved weighted F1 score at this depth. This indicates

that with a balanced dataset, the decision tree can afford to grow deeper and explore more complex decision boundaries, leading to better generalization and performance across both classes.

B. CONFUSION MATRIX

The confusion matrices in Figure 8, Figure 9 and Figure 11 provide a detailed breakdown of the model's performance. At depth 1, the confusion matrix for decision tree having splitting criteria Entropy and Gini Impurity are the same, indicating same predictions in both cases.

At higher depth of 7, the decision tree using Entropy made more correct predictions than the one using Gini Impurity. Also, the decision trees at greater depth did a better job of capturing the false positives than decision trees at depth 1. But only counting the true and false positives is not enough so we look into more evaluation metrics.

C. CLASSIFICATION REPORT

The classification report provides insights into the model's performance.

Table 4: Classification Report at Depth 1 using Gini

	precision	recall	F1-score	Support
Altered	1.00	0.00	0.00	2
Normal	0.9	1.00	0.95	18
Accuracy			0.90	20
Macro avg	0.95	0.50	0.47	20
Weighted avg	0.91	0.90	0.85	20
Weighted F1score	0.85			

The Table 4 indicates that the decision tree model achieves a high accuracy of 90% on the test set, with a weighted F1 score of approximately 0.85. However, it struggles to accurately classify the minority class (Altered), as evidenced by its precision of 1.00 but recall of 0.00.

Table 5: Classification Report at Depth 7 using Gini

	precision	recall	F1-score	Support
Altered	0.14	0.50	0.22	2
Normal	0.92	0.67	0.77	18
Accuracy			0.65	20
Macro avg	0.53	0.58	0.50	20
Weighted avg	0.85	0.65	0.72	20
Weighted F1score	0.72			

The Table 5 indicates that while the accuracy and F1-score can dropped compared to that in Table 4,it did a better job of predicting the minority class (Altered), as suggested by the recall of 0.50 compared to the previous 0.00.

Table 6: Classification Report at Depth 7 using Entropy

	precision	recall	F1-score	Support
Altered	0.14	0.50	0.22	2
Normal	0.92	0.67	0.77	18
Accuracy			0.65	20
Macro avg	0.53	0.58	0.50	20
Weighted avg	0.85	0.65	0.72	20
Weighted F1score	0.79			

The Table 6 shows an improvement in the decision tree model's performance compared to Table 4. The model achieved a precision of 0.20, recall of 0.50, and F1 score of 0.29 for the 'Altered' class, and a precision of 0.93, recall of 0.78, and F1 score of 0.85 for the 'Normal' class. The weighted F1 score is 0.79, which is higher than the first report's 0.72, indicating better overall performance. However, the model still struggles to accurately predict the minority 'Altered' class, suggesting the need for further improvements to handle class imbalance.

V. DISCUSSION AND ANALYSIS

The classification report shows that for lower depth both Gini Impurity and Entropy gives similar result, also seen in Figure 6 and Figure 7. Whereas at higher depth the entropy-based decision tree achieved a higher weighted F1-score of 0.79 compared to 0.72 for the Gini-based tree. This indicates the entropy model had better overall performance in accounting for the class imbalance.

For the minority "Altered" class specifically, the entropy model had a precision of 0.20 and recall of 0.50, resulting in an F1-score of 0.29. In contrast, the Gini-based tree had a lower precision of 0.14 and the same recall of 0.50 for "Altered", leading to an F1-score of only 0.22. The entropy model also had a higher overall accuracy of 0.75 compared to 0.65 for Gini. This suggests the entropy-based decision tree was better able to learn the patterns in the imbalanced dataset and generalize to the test set.

The application of SMOTE had a significant impact on the performance of both decision tree models. Before SMOTE, the optimal tree depth was just 1, indicating that deeper trees were prone to overfitting the majority class.

However, after SMOTE balanced the class distribution, the optimal depth increased to 7 for both the entropy and Gini-based trees. This allowed the models to capture more nuanced patterns in the data and improve their ability to identify the minority "Altered" instances.

The confusion matrices further illustrate this point. At depth 7, the entropy-based tree made more correct predictions than the Gini-based one, showcasing its enhanced capability to distinguish between the two classes after SMOTE.

One key advantage of decision trees is their inherent interpretability. By visualizing the learned tree structures, we can gain valuable insights into the key factors influencing fertility status. For example, the decision trees identified age,

frequency of alcohol consumption, and smoking habit as important predictors.

Examining the decision paths reveals the specific thresholds and rules the model uses to classify instances. This transparency is particularly useful in medical applications, where being able to explain the reasoning behind predictions is crucial for building trust and enabling actionable insights.

The interpretability of decision trees complements their strong performance on imbalanced datasets. By identifying the most informative features and learning simple, human-understandable rules, decision trees can effectively distinguish between the minority and majority classes, even with limited training data.

While the entropy-based decision tree demonstrated promising performance, there are still opportunities for further improvement:

- **Ensemble Methods:** Combining multiple decision trees, such as in a Random Forest or Gradient Boosting ensemble, could help to improve the overall stability and predictive power of the model.
- **Hyperparameter Tuning:** More extensive optimization of hyperparameters like maximum depth, minimum samples per split, and minimum samples per leaf could lead to even better generalization.
- **Feature Engineering:** Exploring additional relevant features or transforming the existing ones could uncover more informative patterns in the data.

By addressing these areas, future research can build upon the foundations laid in this study and develop even more robust and reliable decision tree-based models for predicting fertility status and other health-related outcomes.

VI. CONCLUSION

The report demonstrates the effectiveness of decision tree models in predicting fertility status using a dataset of environmental factors and lifestyle habits. By applying SMOTE to address the class imbalance, the decision tree models were able to significantly improve their performance in identifying the minority "Altered" fertility class.

The entropy-based decision tree model achieved the best overall performance, showcasing the advantages of using entropy as the splitting criterion when dealing with imbalanced datasets. The superior ability of the entropy model to handle the class imbalance is evidenced by its higher precision and recall for the minority "Altered" class compared to the Gini-based tree.

The increased optimal depth of 7 for both the entropy and Gini-based trees after SMOTE demonstrates the models' enhanced capacity to capture more complex patterns in the balanced dataset. This allowed them to better distinguish between the fertile and infertile instances, a crucial capability in medical applications where accurately identifying high-risk individuals is of paramount importance.

The interpretability of decision trees is a key strength, as it enables the identification of the most influential features, such as age, alcohol consumption, and smoking habits, in predicting fertility status. This transparency is invaluable in the medical domain, where being able to explain the reasoning behind predictions is essential for building trust and enabling actionable insights.

While the decision tree models showed promising performance, there are still opportunities for further improvement. Exploring ensemble methods, more extensive hyperparameter tuning, and additional feature engineering could lead to even more robust and reliable predictive models. By building upon the foundations laid in this study, future research can continue to advance the application of decision trees and other machine learning techniques in the field of fertility prediction and healthcare analytics.

REFERENCES

[1] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81-106, 1986.

[2] N. V. B. K. W. H. L. O. K. W. P. Chawla, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321-357, 2002.

[3] L. Breiman, " Random forests," *Machine learning*, vol. 45, no. 1, pp. 5-32, 2001.

[4] M. L. G. Sokolova, "A systematic analysis of performance measures for classification tasks," *A systematic analysis of performance measures for classification tasks*, vol. 45, no. 4, pp. 427-437, 2009.

[5] J. G. David Gil, "UC Irvine Machine Learning Repository," 16 January 2013. [Online]. Available: https://archive.ics.uci.edu/dataset/244/fertility. [Accessed 21 June 2024].

[6] J. L. G. J. D. J. M. J. G.-T. M. J. David Gil, "Predicting Seminal Quality with Artificial Intelligence Methods," *Expert Systems with Applications*, vol. 39, no. 16, pp. 12564-12573, 2012.

APPENDIX

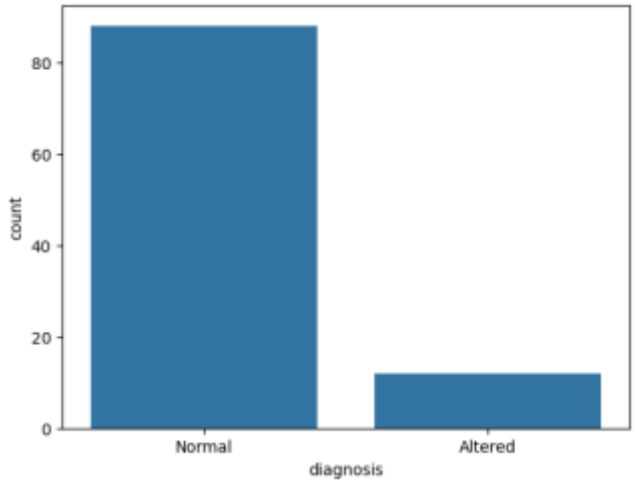

Figure 1: Countplot of Target Class in the Original Dataset

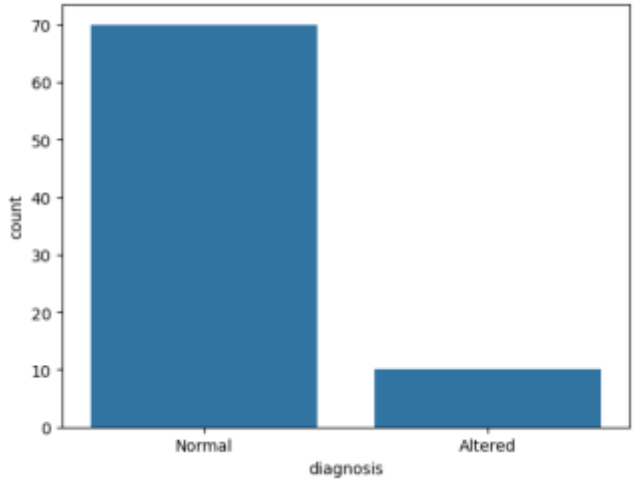

Figure 2: Count Plot of Target Class in Train Dataset

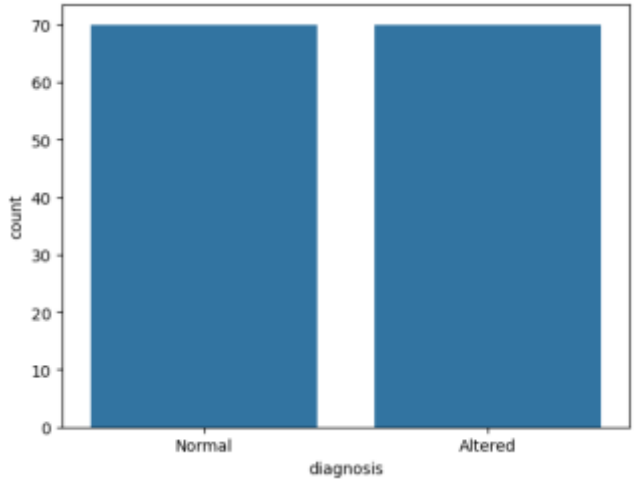

Figure 3: Count Plot of Target Train Dataset after SMOTE

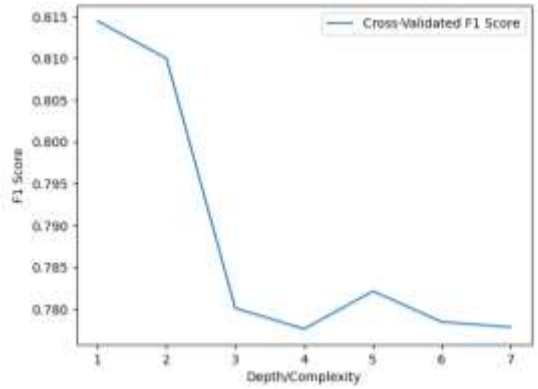

Figure 4: F1 score vs Depth graph before SMOTE

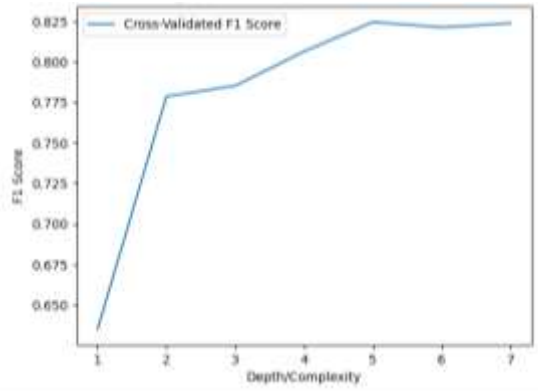

Figure 5: F1 score vs Depth graph after SMOTE

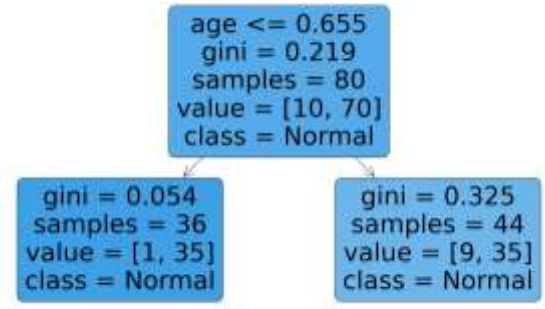

Figure 6: Decision Tree at Depth 1 using Gini Impurity

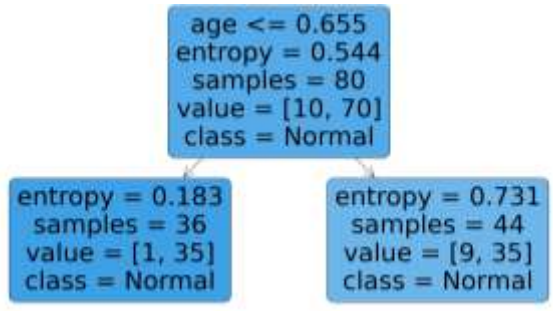

Figure 7: Decision Tree at Depth 1 using Entropy

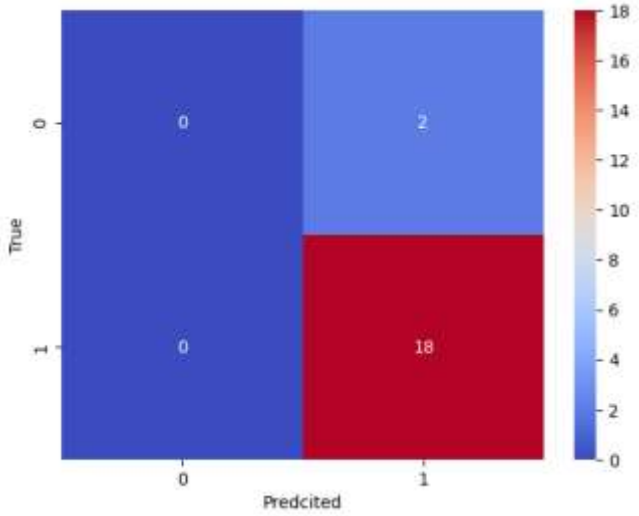

Figure 8: Confusion Matrix at Depth 1

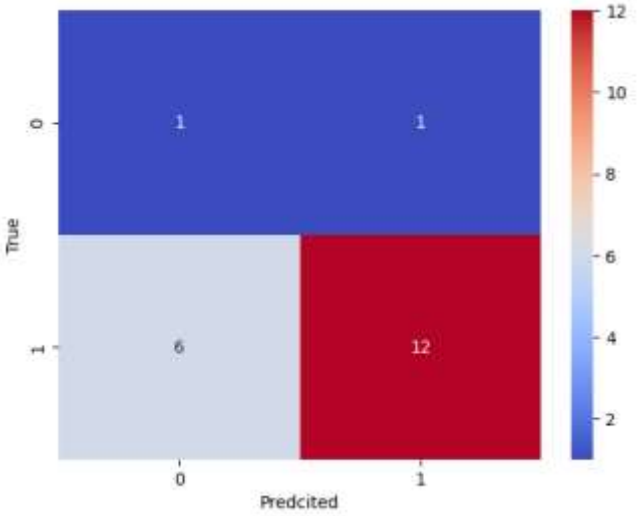

Figure 11: Confusion Matrix at Depth 7 using Gini Impurity

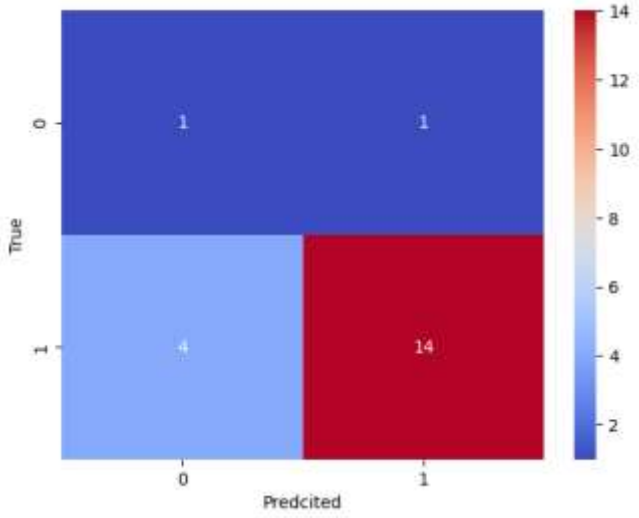

Figure 9: Confusion Matrix at Depth 7 using Entropy

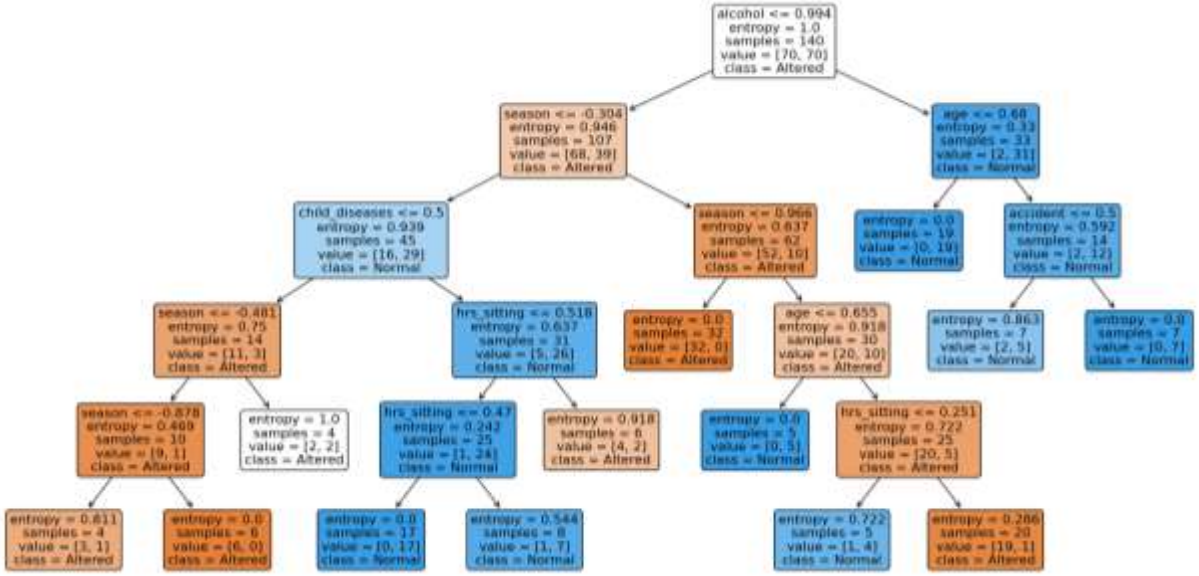

Figure 10: Decision Tree at Depth 7 using Entropy

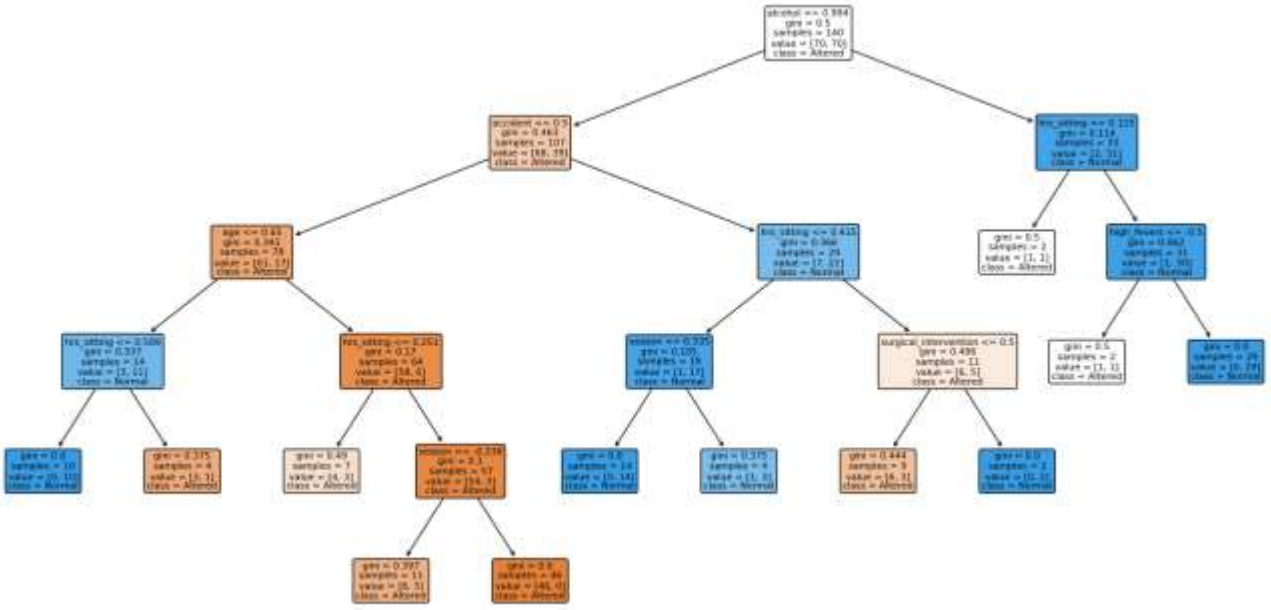

Figure 12: Decision Tree at Depth 7 using Gini