Loan Eligibility Prediction

Group 6: Rounak Bende, Parnavi Sen, Syona Jaimy

April 2023

1 Summary

Company Dream Housing Finance company deals in all home loans. They have presence across all urban, semi urban and rural areas. Customer first apply for home loan after that company validates the customer eligibility for loan. Predicting loan eligibility is a critical step that lenders take to evaluate potential borrowers' creditworthiness and decide whether to lend money to them. Building predictive models using past loan data and different criteria, including age, income, employment status, credit score, loan amount, and loan term, that may have an impact on loan acceptance is the procedure.

The financial crisis of 2008 highlighted the need for more precise and dependable lending methods, which is where loan eligibility prediction got its start. As a result, several financial institutions began to evaluate creditworthiness and reduce the risk of default using data-driven methodologies. Lenders today frequently utilize loan eligibility prediction models to automate the loan approval process, lower risk, expand credit availability, and guarantee fair and equitable lending practices.

Loan eligibility prediction models can assist lenders in making quicker and more informed lending choices while lowering the cost and time associated with human underwriting. They do this by automating the loan approval process and evaluating past loan data. Furthermore, loan eligibility prediction models can assist lenders see biases that might exist during the lending process and guide them in addressing them to promote fair and equitable lending practices.

All things considered, loan eligibility prediction is a crucial tool for lenders to use when making educated lending decisions and ensuring fair and equitable lending practices in the financial sector.

2 Dataset Description:

The dataset contains information on loan applicants and whether or not they were approved for a loan based on their personal and financial information.It contains 614 records and 13 columns such as loan amount applied for, marital status, education level, income of the applicant, number of dependents, loan term, credit history, property location, and loan status.

This dataset can be used for various statistical analyses and modeling techniques to understand the factors that influence loan approval decisions. The dataset is particularly useful for building machine learning models that can predict the likelihood of loan approval for new applicants.

3 Project Description:

The dataset used in the project involves in analysis of the data, some preprocessing techniques to clean the data and estimating the predictions by the criterion of accuracy, recall, F1 score of the models to provide the clear picture of the performance.

The project had 3 main goals:

1. understanding the features and their relationship with the status of the loan.
2. Testing different machine learning models for classification.
3. Identifying the models that give better accuracy and F1 score and also the features that are more efficient in predicting the status of the loan.

The team explored the relationship between Loan Status and other features, and identified models that achieved high accuracy for classifying whether a person gets a loan approved or not. Various algorithms were used for classification, including Logistic Regression, Naive Bayes, KNN, Decision trees, Ensembles, AdaBoost, XGBoost, Neural Networks and Keras(Deep learning Algorithm). The resulting models can be used by Financial organisations to predict the loan eligibility.

4 Methods:

4.1 Exploratory Data Analysis:

Figure 1: Distribution of loan status

The dataset used in this project is highly imbalanced, with approved loans outnumbering disapproved loans. To account for this imbalance during model training and evaluation, the team utilized SMOTE technique to sample the classes and bring them on to the same scale.

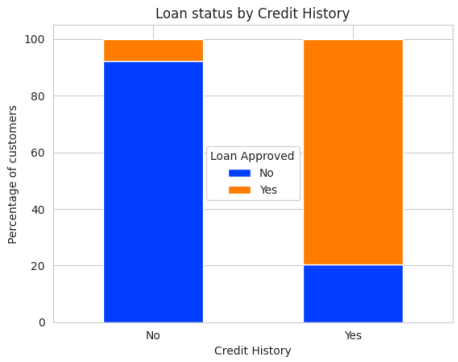


Figure 2: Distribution of loan status and credit history

the graph shows that you are more liking to get your loan approved if you have a credit history.

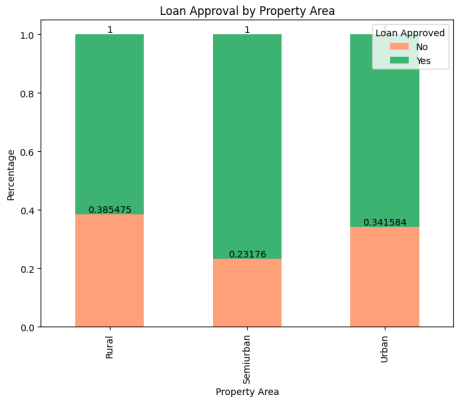


Figure 3: Distribution of loan status and property area

Semiurban area has the highest percentage of approved loans and rural has the highest rate of loans rejected.

Figure 4: loan numerical data histograms

We can see that Loan amount, Applicant income and Co-applicant income are right skewed so we need to do some transformation to normalize these variables. Loan term is not continous so we wont do any transformations on that variable.

4.2 Data Pre-Processing:

- Data set consists of 614 samples with 13 features including the target variable.
- Data has both categorical and numerical data as well.
- Some of the samples contains empty values in their features. In order to handle the empty values we have replaced the values of the numerical data with their respective mean values of the features similarly for categorical features we have replaced the empty values with their most frequent values of the respective feature.
- There is a feature named LoanID which is unique for each and every sample but no effect in predicting the output so we had to remove the feature that is unused.
- In order to encode categorical data we have used LabelEncoder which assigns the numeric values sequentially for the total number of unique values that are present for a feature.
- Box-cox and square root transformations were used to continuous variables such as Loan amount, ApplicantIncome, and CoapplicantIncome to reduce the positive skewness that we observed and make the data more normally distributed.
- Scaling is performed on the full dataset by using Min-Max scaling technique as the features can have a standard deviation negative and might affect the predictions.

- From the observations of exploratory analysis it can be seen that there is class imbalance in the dataset. In order to handle class imbalance we have used SMOTE technique to sample the classes and bring them on to the same scale.
- Feature selection was done to improve the performance of machine learning algorithms. We used correlation, Lasso regression, random forest and Recursive Feature Elimination methods to select the best features. Education,Gender and self employed columns were dropped from the dataset.
- Feature extraction was also performed to boost the efficiency of models by minimizing data complexity. Two characteristics, total income and debt to income ratio, were created but were removed because they added no significance to the model.
- From the observations made during the exploratory data analysis there are several outliers in some features which need to be handled. For that, we have used the data between the 25th and 75th quantile ranges.

4.3 Modelling:

Our task is binary classification, we need to classify whether a Person's loan is approved or not. Description of the models used:

1. **Logistic Regression:** Logistic Regression is a widely used classification algorithm that models the probability of the binary outcome (in our case, Approved or not) as a function of the independent features. It assumes a linear relationship between the features and the log odds of the outcome. The sigmoid function is applied to the output of the linear equation to generate a probability score between 0 and 1. A threshold is used to convert the probability score into the predicted class.
2. **Stochastic Gradient Descent with Ridge Regularization:** Stochastic Gradient Descent is a classification technique which constantly iterates and converges to global minima assuming there is only one global minima for the classification. Ridge regularization is a regularization constant of L2- Norm that is applied to the cost function.
3. **Stochastic Gradient Descent with Lasso Regularization:** Stochastic Gradient Descent with Lasso is the the Descent function that works with the regularization of Lasso which is a L1- Norm that is applied to the cost function.
4. **Naïve Bayes:** Naïve Bayes is a probabilistic classifier based on Bayes' theorem, which calculates the probability of a given outcome given the presence of a particular feature. The algorithm assumes that the features are independent of each other, and thus, it is called "naïve." It calculates the prior probability of the classes and the likelihood of the features for

each class to calculate the posterior probability of each class. The class with the highest posterior probability is the predicted class.

5. **Gaussian Naïve Bayes:** The classification method known as Gaussian Naive Bayes (GNB) in Machine Learning (ML) is based on the probabilistic approach and Gaussian distribution. Each parameter (also known as a feature or a predictor) is assumed to have an independent capacity to predict the output variable by the Gaussian Naive Bayes algorithm.

6. **K-Nearest Neighbours:** KNN technique is most simple and commonly used algorithm which works in such a way that a sample is classified based on its corresponding sample classes near to them in the n-dimensional space.

7. **Decision Trees(Gini Index):** When building machine learning algorithms, decision trees are frequently employed. A decision tree's hierarchical structure takes us to the final consequence by traversing the tree's nodes. As we travel down the tree, each node contains an attribute or characteristic that is further subdivided into more nodes. In order to decide split Gini index is the criteria used and which has the highest gini index is selected as the root node.

8. **Decision Trees(Entropy):** The similarly functionality of the decision trees work but the only difference is that the criteria used for the split is Entropy. Using the entropy information Gain is calculated for each feature and the feature that gives highest information gain is selected as the root node.

9. **Ensemble Methods:** Ensemble methods in machine learning are techniques that combine multiple models or algorithms to improve the overall accuracy and robustness of the predictions. Ensemble methods can be used with different types of models such as decision trees, neural networks, and regression models. Hard voting and soft voting are two methods used in ensemble learning where multiple models are combined to make a final prediction. Hard voting involves taking the majority vote of the individual models while soft voting involves taking the average probability of each class across all models and selecting the class with the highest probability. Soft voting can provide more accurate results than hard voting when the models have different strengths and weaknesses or produce continuous outputs.

10. **Bagging Classifier:** The Bagging Classifier is an ensemble machine learning method that uses the Bagging technique to improve the accuracy and robustness of predictions. It creates multiple decision tree models using bootstrap samples of the training data and combines them by taking the average prediction for regression problems or the majority vote for classification problems. The Bagging Classifier reduces variance and

overfitting, is effective for unstable models, and is easy to implement. It is useful for complex datasets with high noise or variability.

11. **Random Forest:** Random Forest Classifier works by creating an ensemble of decision trees and combining their predictions to make a final prediction. The algorithm randomly selects subsets of the training data and features to create each decision tree, which helps to reduce overfitting and improve accuracy. The decision trees are constructed using the information gain or Gini index criteria to split the data based on the most informative features. During prediction, each decision tree produces a class prediction, and the Random Forest Classifier combines these predictions by taking the majority vote of all the trees. This approach helps to reduce the variance and bias of the model and produces a robust and accurate classifier that can handle complex datasets.

12. **Extra Trees Classifier:** Extra Trees Classifier is a machine learning algorithm that creates an ensemble of decision trees and combines their predictions to make a final prediction. It differs from Random Forest Classifier in that it randomly selects split points for each feature, reducing the variance of the model. The algorithm creates decision trees by randomly selecting subsets of the training data and features, and splits the data based on the most informative features using the information gain or Gini index criteria. During prediction, the algorithm combines the predictions of all the decision trees by taking the majority vote. This approach produces a robust and accurate classifier that can handle complex datasets with non-linear relationships between features and the target variable. Extra Trees Classifier is a powerful and efficient algorithm that can be applied to a wide range of applications.

13. **AdaBoost:** AdaBoost (Adaptive Boosting) is a machine learning algorithm that combines multiple weak learners (usually decision trees) to create a strong classifier. It iteratively adjusts the weights of the training instances and assigns a higher weight to misclassified instances in each iteration. The weak learners are then trained on these weighted instances to focus on the previously misclassified instances. During prediction, the algorithm combines the predictions of all the weak learners by taking a weighted majority vote. This iterative process helps to improve the accuracy of the model and creates a strong classifier that can handle complex datasets with non-linear relationships between features and the target variable. AdaBoost is particularly useful when there are many noisy or irrelevant features in the dataset.

14. **Gradient Boosting:** Gradient Boosting Classifier is a machine learning algorithm that creates a strong classifier by combining multiple weak learners (usually decision trees) in a sequential manner. During training, the algorithm adds new decision trees to the model and adjusts their weights to minimize the loss function of the model. This process continues until

the model reaches a predefined number of trees or an acceptable level of accuracy. During prediction, the algorithm combines the predictions of all the decision trees by taking a weighted majority vote. Gradient Boosting Classifier is a powerful and flexible algorithm that can produce accurate and robust predictions for a wide range of applications, particularly when there are complex interactions between features and the relationship between the features and the target variable is non-linear.

15. **XGBoost:** eXtreme Gradient Boosting (XGBoost) is an extension to the gradient boosting algorithm that uses a decision tree ensemble to improve performance and speed. It trains weak decision trees sequentially, where each new tree is built to correct the errors of the previous tree. The algorithm uses gradient descent optimization to minimize the loss function. It also incorporates regularization techniques to prevent overfitting and improve generalization.

16. **Support Vector Machines**: Support Vector Machines (SVMs) are a type of machine learning algorithm that can be used for classification or regression tasks. SVMs find the best decision boundary or best-fit line/curve to separate or predict data points, respectively. They do this by transforming the input data into a higher-dimensional feature space and maximizing the margin between the decision boundary and the closest data points, called support vectors. SVMs are powerful and versatile algorithms, but can be computationally intensive and sensitive to hyperparameters.

17. **Perceptron:** The perceptron is a basic type of neural network used for binary classification problems. It takes in input data and assigns weights to each feature, then computes a weighted sum. The result is passed through an activation function, typically a step function, to produce a binary output of 1 or 0. The perceptron learning algorithm adjusts the weights based on the error between the predicted output and the actual output. It continues to iterate over the training data until the algorithm converges or a stopping criterion is met. Perceptrons can be used for simple classification tasks but are limited in their ability to handle more complex problems.

18. **Multi-Layer Perceptron:**A multilayer perceptron (MLP) is a deep, artificial neural network. It is composed of more than one perceptron. They are composed of an input layer to receive the signal, an output layer that makes a decision or prediction about the input, and in between those two, an arbitrary number of hidden layers that are the true computational engine of the MLP.

19. **Deep Neural Networks:**Deep Neural Networks have multiple layers of interconnected artificial neurons or nodes that are stacked together. Each of these nodes has a simple mathematical function—usually a linear function that performs extraction and mapping of information.

5 Results:

We have built models using the algorithms mentioned above. Hyperparameters, plots, and metricsvalues of those models are provided below:

Model	Hyperparameters
Logistic Regression	penalty: l2, solver: sag
Naive Bayes	alpha: 10
KNeighborsClassifier	nearest neighbors: 3
DecisionTreeClassifier	max depth: 7, min samples leaf: 1, min samples split: 5
Voting Classifier	3 classfiers(logisitic, knn, Decision Trees)
Bagging Classifier	estimators: 400
RandomForestClassifier	estimators: 1000
ExtraTreesClassifier	estimators: 1200
AdaBoostClassifier	estimators: 400
GradientBoostingClassifier	learning rate: 0.1, max depth: 6, estimators: 1000
XGBClassifier	estimators: 600
Support Vector machine	C: 10000, kernel: rbf
Multi-Layer Perceptron	hidden layer size: 250

Model	AUC-ROC	Accuracy	Precision	Recall	F1 Score
Logistic Regression	0.68	0.70	0.73	0.75	0.74
Gaussian Naive Bayes	0.69	0.66	0.74	0.63	0.68
KNeighborsClassifier	0.74	0.73	0.80	0.70	0.75
Lasso regularization	0.56	0.61	0.60	0.95	0.74
Ridge regularization	0.68	0.69	0.72	0.75	0.73
DT using gini	0.75	0.74	0.81	0.73	0.76
DT using entropy	0.71	0.70	0.77	0.68	0.72
DT using regularization	0.71	0.70	0.77	0.68	0.72
Voting Classifier(hard voting)	0.77	0.76	0.83	0.73	0.77
Voting Classifier(soft voting)	0.75	0.74	0.82	0.70	0.76
Bagging Classifier	0.71	0.76	0.83	0.73	0.77
RandomForestClassifier	0.84	0.84	0.87	0.85	0.86
ExtraTreesClassifier	0.81	0.81	0.85	0.82	0.83
AdaBoostClassifier	0.68	0.67	0.75	0.65	0.70
GradientBoostingClassifier	0.83	0.83	0.91	0.78	0.84
XGBClassifier	0.84	0.84	0.91	0.78	0.84
SVM	0.77	0.77	0.82	0.78	0.8
Perceptron	0.62	0.60	0.72	0.51	0.60
MLPClassifier	0.71	0.71	0.75	0.75	0.75
Deep Learning(Keras)	0.82	0.82	0. 89	0.78	0.82

- AUC-ROC: It is the area under the receiver operating characteristic curve, which is a measure of how well the model can distinguish between positive and negative classes. A value of 1 indicates perfect performance, while a value of 0.5 indicates random guessing.

- Accuracy: It is the proportion of correctly classified instances out of the total number of instances. While accuracy is a common metric, it can be misleading in cases of imbalanced datasets.

- Recall: It is the proportion of true positive instances (correctly classified positive instances) out of the total number of actual positive instances. It measures the ability of the model to correctly identify positive instances.

- Precision: It is the proportion of true positive instances out of the total number of instances that the model classified as positive. It measures the ability of the model to avoid false positives. 5. F1 Score: It is the harmonic mean of precision and recall, which provides a balanced measure of both metrics.

Looking at the performance metrics in the table 2, we can see that Random forest and XGBoost has the highest AUC-ROC score, indicating that it performs best at distinguishing between positive and negative instances. It also has the highest F1 score, indicating that it has a balanced performance between precision and recall. On the other hand, KNeighbors Classifier and Gaussian Naive Bayes have low recall scores, which suggests that they have trouble correctly identifying positive instances. Gradient Boost and Extra trees also perform well on this dataset.

1. Random Forest model outputs:

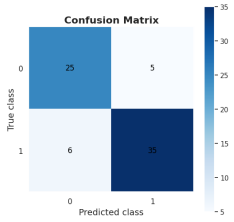


Figure 5: Confusion matrix

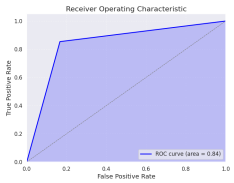


Figure 6: AUC graph

2. XGBoost model outputs:

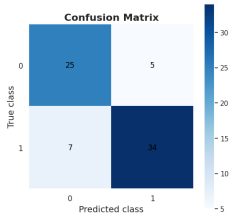


Figure 7: Confusion matrix

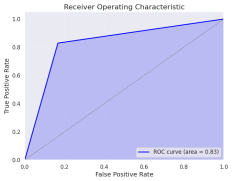


Figure 8: AUC graph

3. GradientBoost model outputs:

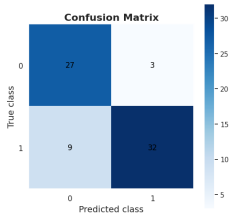


Figure 9: Confusion matrix

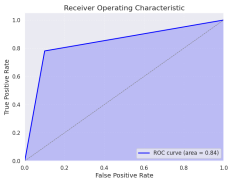


Figure 10: AUC graph

4. Extra trees model outputs:

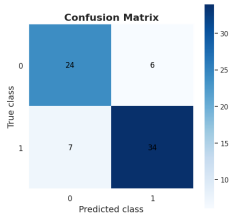


Figure 11: Confusion matrix

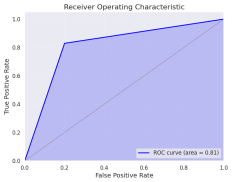


Figure 12: AUC graph

6 Discussion:

In the context of predicting the decision of the loan approval we found that Applicant income, Dependents, Property area, co-applicant income and married features contributes largely in predicting the outcome. The F1 score and also AUC-ROC which are a widely used metric to evaluate the performance of binary classification models. It measures the ability of the model to distinguish between positive and negative classes, where a higher F1 score indicates better

performance. F1 score is impacted by 2 factors that are precision and recall. More the precision and recall higher the F1 score. But in our case precision is an important factor. It implies that we can slightly compromise on recall but precision must be greater and overall F1 score should also be high. Similarly, higher AUC-ROC score indicates better performance.

Among the ML models tested in this task, Random Trees Classifier, Extra Trees, and XGBoost Classifier have shown the best performance based on the F1-score metric and also AUC-ROC.

7 Future Scope Of Work:

- Exploring the use of alternative data sources: While traditional credit scores and employment history are important predictors of loan defaults, alternative data sources such as social media activity, online purchasing behavior, and mobile phone usage could provide additional insights into borrower behavior and help improve lending decisions.
- Applying more deep learning techniques to get better performance.
- Using stacking techniques and trying to get a higher level of blending to improve our model.
- Expanding the dataset to include more diverse borrower profiles: While the loan prediction dataset provides valuable insights, it is important to expand the dataset to include more diverse borrower profiles, including those with limited credit history, immigrants, and other under-served populations. This could provide new insights into borrower behavior and help improve lending practices for these communities.

8 Statement of Contribution:

1. **Syona Jaimy:** Exploratory Data analysis, Hyper-parameter tuning and visualization, model evaluation, performance metrics, Decision Trees(Gini-index, Entropy, Max-depth), project report.
2. **Rounak Bende:** Data pre-processing, Logistic Regression, Stochastic Gradient Descent with Ridge Regularization, Stochastic Gradient Descent with Lasso Regularization, Gaussian Naive Bayes, Naive Bayes, K-Nearest Neighbours, project report.
3. **Parnavi Sen:** Ensemble Classifier (Hard voting, Soft voting), Bagging Classifier, Random Forest Classifier, Extra-trees classifier, AdaBoost, Gradient Boosting, XGBoost, Support Vector Machines, Perceptron, Multi-layered Perceptron, Keras(deep-learning). Project presentation.

9 References:

- https://www.kaggle.com/datasets/vikasukani/loan-eligible-dataset
- https://scikit-learn.org/stable/modules/generated/sklearn.linear model.LogisticRegression.html
- https://xgboost.readthedocs.io/en/stable/
- https://www.tensorflow.org/guide/keras/functional
- https://www.mygreatlearning.com/blog/xgboost-algorithm/
- https://towardsdatascience.com/perceptrons-the-first-neural-network-model-8b3ee4513757
- https://wiki.pathmind.com/multilayer-perceptron
- https://www.v7labs.com/blog/deep-learning-guide: :text=What

10 Appendix:

10.1 Results

1. **Graph of loan vs dependents:**

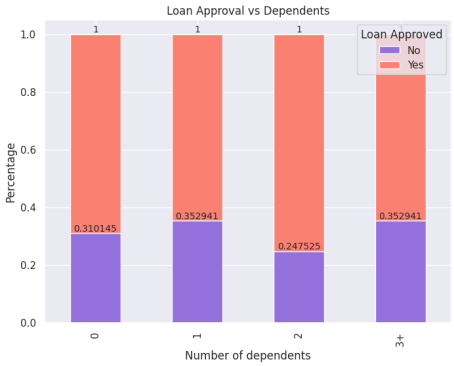


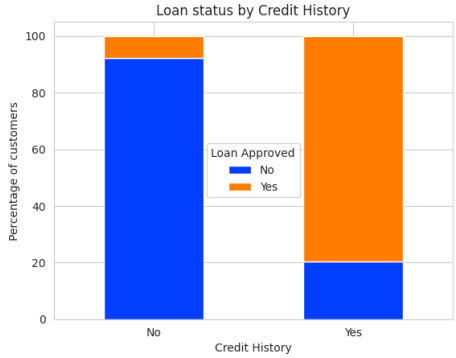
Figure 13: Distribution of loan status and property area:

the graph tells us that there is higher rate of approvals where there are no dependents and the number of dependents are 2 versus having 3 or 4 number of dependents

2. **Graph of loan status Vs married:**

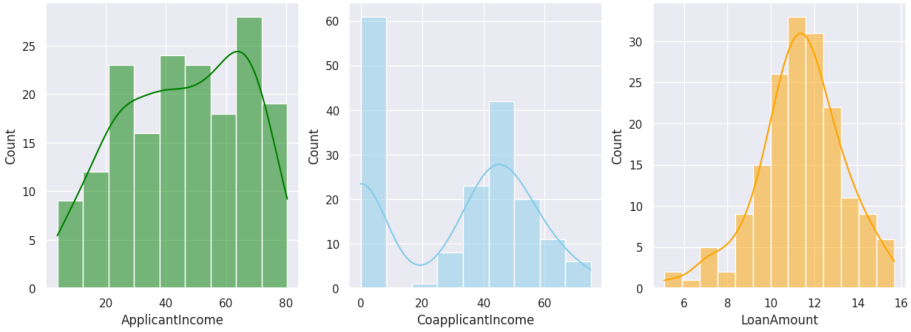
the graph shows that most of the individuals are married compared to unmarried. There is also a higher rate of loan approval for married individuals.

3. **Graph of Loan status Vs credit history**



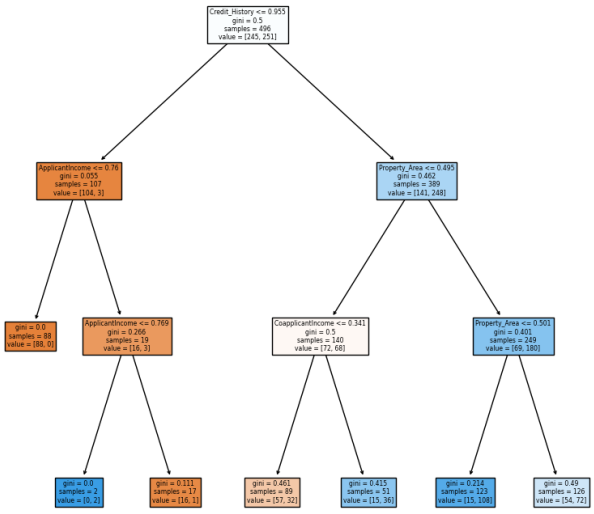
the graph tells that most of the individuals have credit history and they are more likely to get their loan approved

4. **Result of Numerical variables after normalization**



We have used Box Cox transformation on Applicant income. Square root transformation was used on Loan amount and Co applicant income. We choose these transformation specifically because they give yes the best results.

5. **Decision Tree**



Representation of decision tree on our dataset with max depth =3

Code for our project: Code Hyperlink