Loan Approval

Overview

The loan approval dataset is a collection of financial records and associated information used to determine the eligibility of individuals or organizations for obtaining loans from a lending institution. It includes various factors such as cibil score, income, employment status, loan status. This dataset is commonly used in machine learning and data analysis to develop models and algorithms that predict the likelihood of loan approval based on the given features.

Project Phases

- Problem Statement
- Data Collection
- Data Cleaning and Preprocessing
- Exploratory Data Analysis (EDA)
- Feature Engineering
- Model Training & Selection
- Model Deployment
- Documentation and Reporting

Problem Statement

Classification Problem

- **Statement**: Predict whether a loan application will be approved or rejected based on the applicant's Income, number of dependents, loan amount, and other relevant features.
- **Objective**: Build a classification model that outputs a binary decision (approved or rejected) for each loan application.

Data Collection

I obtained the dataset from Kaggle, which is provided in the form of a CSV file, containing all the necessary information for the analysis.

The data consists of:

- Loan id: the id of the loan
- No of dependents: Number of Dependents of the Applicant
- Education: Education of the Applicant
- Self-employed: Employment Status of the Applicant
- Income annum: Annual Income of the Applicant
- Loan amount: The total sum requested for borrowing
- Loan term: Duration over which the loan must be repaid in Years
- Cibil score: Credit Score
- Residential assets value: The total value of an applicant's residential properties.
- Commercial assets value: The total value of an applicant's commercial properties.
- Luxury assets value: The total value of an applicant's high-end or luxury possessions, such as expensive cars, jewelry, or art
- Bank asset value**:** The total value of an applicant's assets held in bank accounts
- Loan status: Loan Approval status

Data Collection

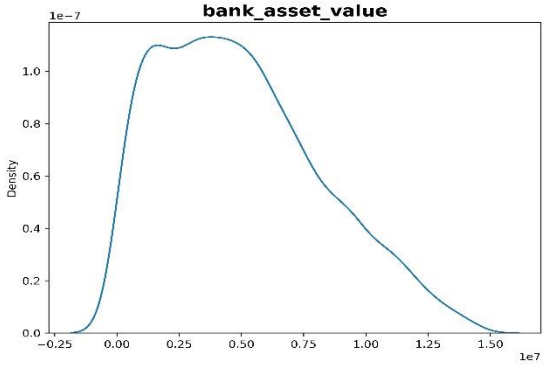
Initially, I conducted an exploratory analysis of the dataset to assess its completeness and structure. I checked for missing values and confirmed that there were none. Next, I examined the dataset for any duplicate records and found that there were no duplicates present.

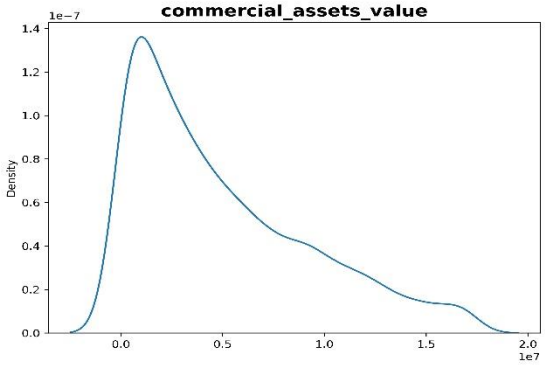
Data Cleaning

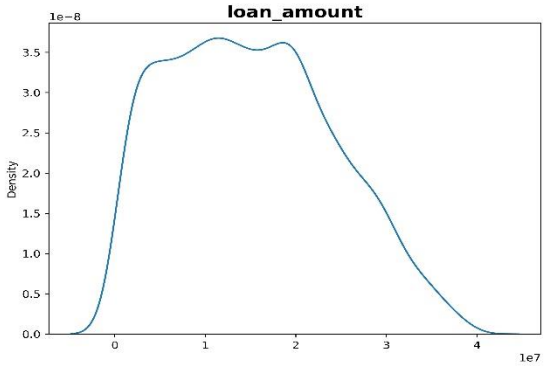
- Removed the redundant spaces from strings using strip method
- Deleted Loan ID column as it's redundant
- Removed redundant spaces from column names
- Replaced Outliers with the upper bound using IQR method

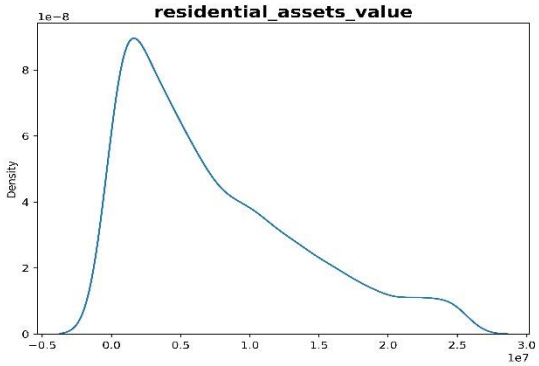
EDA

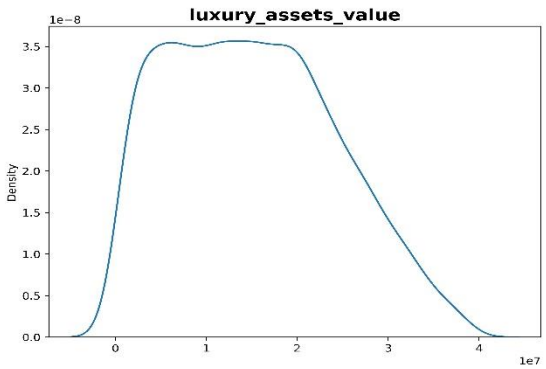
Numerical Features Distributions

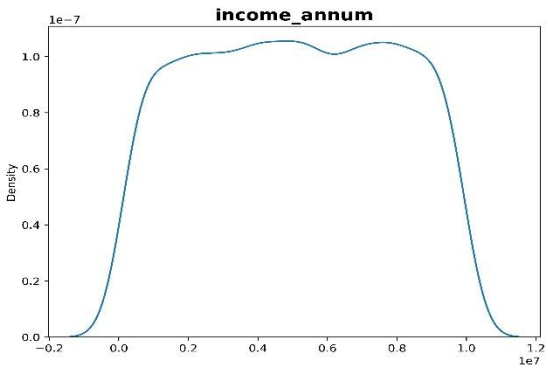


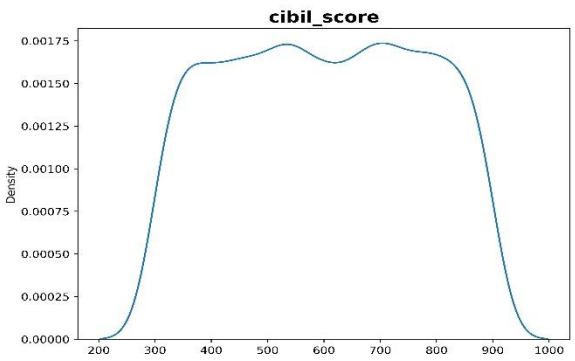












Conclusions drawn from EDA

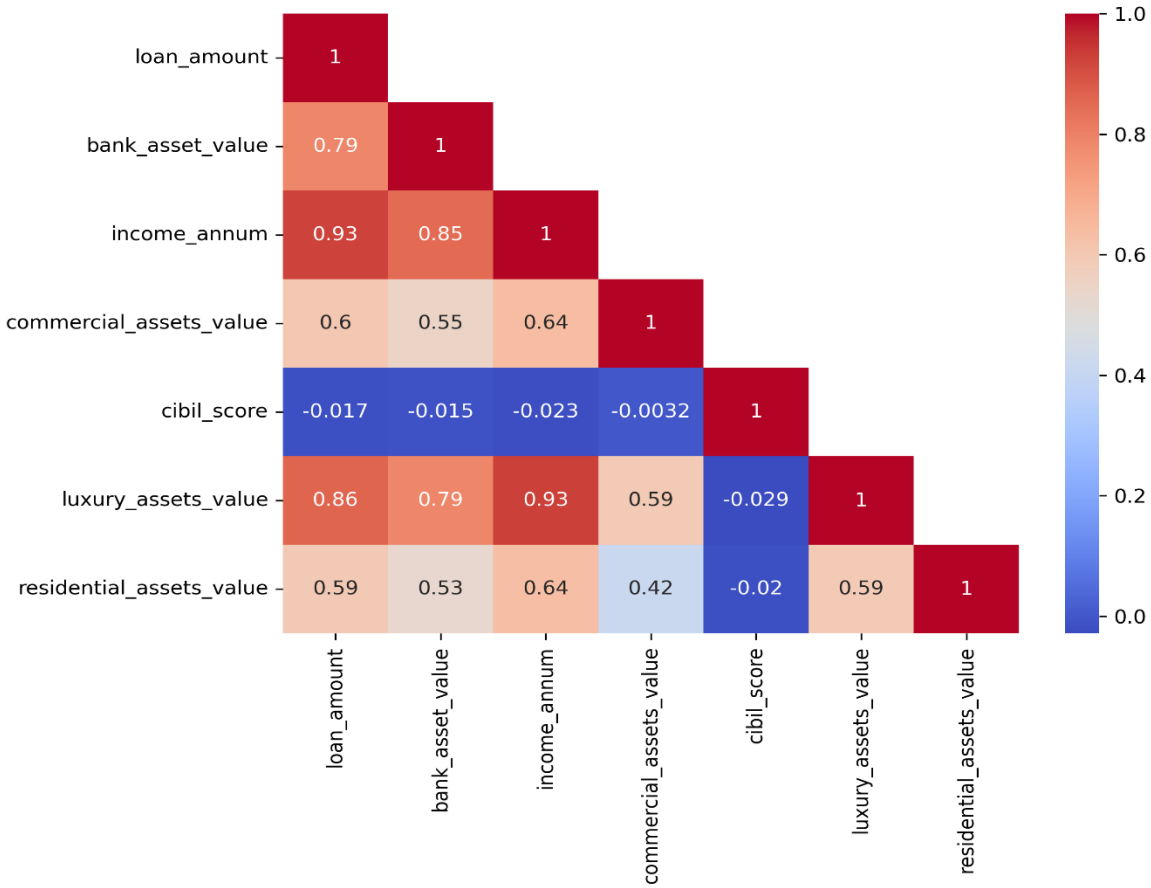
- An increase in your Bank Assets typically leads to a higher loan amount requirement.
- A rise in Bank Assets often indicates increased wealth, which is usually accompanied by a rise in Luxury Assets.
- Credit Score is the most crucial metric in determining whether you will be approved for a loan.
- An increase in Income Amount generally corresponds to a higher loan amount request.
- Individuals with high Luxury Asset values are more likely to request larger loan amounts.
- The number of people who are not graduates and take loans greater than 13.5M is higher than the number of graduates who take such loans.
- Credit score doesn't have effect on loan amount

Correlation

- Loan amount is positively correlated with bank asset value
- Loan amount is positively correlated with annual Income
- Loan amount is positively correlated with luxury asset value
- Annual Income is positively correlated with luxury asset value
- Annual Income is positively correlated with residential asset value
- Annual Income is positively correlated with bank asset value
- Bank asset value is positively correlated with luxury asset value
- Credit score has nearly no correlation with any one of them
- Bank asset value is positively correlated with commercial asset value
- Commercial asset value is positively correlated with luxury asset value
- Luxury asset value is positively correlated with Residential asset value

Conclusion

Since many features are dependent, resulting in multicollinearity, we will need to address this issue by Identifying and removing highly correlated features, applying dimensionality reduction techniques like PCA, using regularization methods to mitigate the impact of multicollinearity.



Note

I Continued cleaning data using power query in excel

Feature Engineering

- I converted the loan status feature (target) into a binary integer format using Label Encoder.
- I transformed the other categorical features into numerical values using Ordinal Encoder.
- I Standard Scaled the numerical features while preserving their original distribution.
- I split the data into training and testing sets.
- I selected the most important features using Select K Best.

Model Training & Selection

- **Initial Model**: I started with a logistic regression model trained on the entire dataset, achieving an accuracy of nearly 89% on the test data.
- **Feature Selection**: I then refined the model by selecting the best features, which resulted in an improved accuracy of 94% using logistic regression.
- **Data Transformation**: To address right skewness, I applied a square root transformation after handling zero and negative values, which helped in normalizing the data distribution.
- **Hyperparameter Tuning**: I performed hyperparameter tuning on the logistic regression model using Grid Search CV, discovering that a value of C=1 yielded the highest accuracy.
- **Decision Tree Model**: My final model was a decision tree. I optimized its parameters using Randomized Search CV, identifying the best parameters: max leaf nodes = 20, criterion = Gini and max depth = 16. This model achieved an impressive accuracy of 99%.
- **Final Selection**: Ultimately, I selected the decision tree model due to its superior performance. I saved both the scaler and the model using pickle for future use.

Model Deployment

The deployment of the loan approval prediction model is carried out using Flask, a lightweight web framework in Python. Flask serves as the API layer, allowing the trained machine learning model to be easily integrated into a web-based or application environment.