LOAN ELIGIBILITY PREDICTION ANALYSIS

-110122091

Problem :-

This project aims to develop a machine learning solution for Dream Housing Finance company to automate the process of evaluating loan applications. The company wants to improve its loan approval process by quickly determining which applicants are eligible for a loan, based on features such as their income, credit history, and loan requirements. The core objective is to create a model that accurately predicts loan eligibility, thereby reducing manual effort and improving operational efficiency.

Objectives:

The objectives of this project are as follows:

1. **Eligibility Classification:** Develop a machine learning model to predict whether a loan applicant is eligible for a loan.
2. **Recommendations for Ineligible Customers:** Build models to predict the maximum loan amount and minimum loan duration for applicants who are not eligible for their requested loan terms.
3. **Model Evaluation and Tuning**: Evaluate and fine-tune the models using relevant metrics and hyperparameter tuning methods for optimal performance.

Dataset Description:

The dataset contains 614 loan applications with 13 features, including applicant demographics, financial information, and loan details. The 'loan_status' variable indicates if a loan application was approved ("y") or rejected ("n"). The dataset contains missing values and requires pre-processing.

Exploratory Data Analysis (EDA) :-

This section provides a detailed account of the exploratory data analysis (EDA) process applied to the loan application dataset. The goal of EDA is to understand the data's characteristics, identify patterns, and gain valuable insights that can inform model building.

Data Loading and Initial Inspection The dataset, named `training_set.csv`, was loaded into a Pandas DataFrame using the `pd.read_csv()` function. The DataFrame was then examined to understand its structure and characteristics using the following Pandas functions: * `df.head()`: to display the first five rows of the DataFrame to provide a general idea of data format * `df.info()`: to inspect data types and identify missing values in each column. * `df.describe()`: to obtain descriptive statistics (count, mean, standard deviation, min, max, percentiles) for the numerical columns. The results of these initial inspections provided key information regarding the data types, missing values, and distribution of numerical features in the dataset. The dataset has the following column names: 'loan_id', 'gender', 'married', 'dependents', 'education', 'self_employed', 'applicantincome', 'coapplicantincome', 'loanamount', 'loan_amount_term', 'credit_history', 'property_area', and 'loan_status'.

Missing Value Analysis and Handling The initial inspection revealed that several features contained missing values. These columns, along with the number of missing values are as follows: * 'gender' has 13 missing values. * 'married' has 3 missing values. * 'dependents' has 15 missing values * 'self_employed' has 32 missing values * 'loanamount' has 22 missing values * 'loan_amount_term' has 14 missing values * 'credit_history' has 50 missing values. To handle the missing data, the following strategies were employed: * **Categorical Features:** For categorical features such as 'gender', 'married', 'self_employed', 'property_Area' and 'education' missing values were imputed with the most frequent value in the respective column using the `mode()` function. * **Numerical Features:** For numerical features such as 'LoanAmount', 'Loan_Amount_Term' and 'Credit_History', the missing values were imputed with the median value using the `median()` function. This approach is more robust to outliers as compared to mean imputation. * **Dependents:** The 'Dependents' column was pre-processed by replacing '3+' with '3' and filling missing values using fillna with the default value of 0 and converting the dtype to `int`. This approach ensured that all missing data were handled using appropriate techniques and the dataset was complete before analysis.

Data Distribution Analysis * **Numerical Features:** To analyze the distribution of numerical features, I used the `sns.histplot()` function from the seaborn library for plotting the histograms of numerical features and identify distribution patterns such as skewness and outliers. * `applicantincome`: The histogram shows that the applicant income data is right-skewed, indicating that the majority of applicants have income values that are clustered towards the lower end and some outliers with very high income * `coapplicantincome`: The coapplicant income data also shows right-skewness as the majority of applicants have low coapplicant income with some applicants with high coapplicant income * `loanamount`: The loan amount distribution is also right-skewed with most loan amount values clustered towards the lower end with the presence of some outliers with very high loan amount* `loan_amount_term`: The loan term is not very continuous and the majority of the loan terms have value of * `credit_history`: The credit history is discrete with two values, 1.0 and 0.0, and we can see in the graph that the majority of the applicant's have credit history of 1.0 * **Categorical Features:** The distribution of the categorical features was analyzed using seaborn's `countplot` function. The following distributions were observed: * `gender`: Most of the applicants are male and only few are females. * `married`: Most of the applicants are married. * `dependents`: Most of the applicants have 0 dependents. A small number of applicants had 1,2 or 3 dependents. * `education`: There are more graduate applicants as compared to not graduate) * `self_employed`: Majority of the

applicants are not self employed. * `property_area`: There is a more or less balanced distribution of applicants across Urban, Semiurban and Rural property areas.

Relationship Analysis * **Correlation Heatmap:** A correlation matrix was generated for the numerical features to find the degree of correlation using seaborn's `heatmap()` function. The analysis showed a strong positive correlation between: * `applicantincome` and `loanamount` (0.57). * `coapplicantincome` and `loanamount` (0.38). * `applicantincome` and `total_income` (0.9) * `loanamount` and `total_income` (0.62) * **Scatter Plots/Pair Plots:** The relationships between the feature columns and the loan status was analysed through scatter plots using `sns.pairplot()`. There were no clear linear correlations found through this analysis.

Feature Importance (Based on EDA) Based on the EDA, the following features seemed to have the most potential for predicting loan eligibility: * `credit_history` : The distribution of the credit history and it's high correlation with the loan status suggests its importance. * `applicantincome_log` & `total_income_log`: These features were shown to be skewed and after applying log transformation, can play an important role. * `loanamount_log` : The loan amount was also highly correlated with income, and therefore might be an important feature

Insights from EDA The EDA revealed the following key insights: * The dataset has missing values, which needs to be handled before the modelling stage. * The income-related features showed skewed distribution. * `applicantincome`, `coapplicantincome`, `loanamount` are positively correlated. * Features like 'credit_history', 'applicantincome', 'loanamount', and 'total_income' are likely to have a strong impact on the loan approval process.

Limitations The dataset had some missing values and outliers, which were handled through imputation techniques and log transformations respectively. However, further investigations could be done to explore other options such as more sophisticated feature engineering techniques. The current dataset had limited set of features that could limit the prediction power of the developed models.

Model Building

Model Building

This section details the process of building regression models to recommend maximum loan amounts and minimum duration to ineligible applicants.

Data Preparation

A new dataset was created for the regression models by filtering out the loan applications where loan status was eligible from the training set, resulting in data for ineligible customers. The new set of features was used for the regression task. Data was also separated based on the prediction target. LoanAmount was the target variable for maximum loan amount prediction, and Loan_Amount_Term was used for the loan duration prediction.

Model Selection and Evaluation

To build the regression model, two algorithms were considered: Linear Regression and Random Forest Regressor. GridSearchCV was used for hyperparameter tuning of all models with the `cv=3` parameter and the

performance were evaluated using R^2, MAE, MSE, and RMSE. The residual plot and predicted vs. actual values were also plotted for each model.

* **Linear Regression:**

  * Model Explanation: Linear Regression is a linear approach to model the relationship between a dependent variable and one or more independent variables.

  * GridSearchCV & Results:

    * GridSearchCV was used to tune the hyperparameters of the model. The following hyperparameter was considered: `'fit_intercept':[True, False]`. The best parameters selected by GridSearchCV are `{'fit_intercept': True}`

  * Performance:

    * R^2: `-0.03`

    * MAE: `77.40`

    * MSE: `9983.04`

    * RMSE: `99.92`

* **Random Forest Regressor:**

  * Model Explanation: Random Forest Regressor is an ensemble model that uses many decision trees to learn the relation between features and a continous target variable.

  * GridSearchCV & Results:

    * GridSearchCV was used to tune the hyperparameters of the model. The following hyper parameters were considered: `'n_estimators': [100, 200, 300],'max_depth': [5, 8, 10],'min_samples_leaf': [1, 3, 5], 'min_samples_split':[2, 4, 6]`.

    * The best parameters selected by GridSearchCV are `{'max_depth': 5, 'min_samples_leaf': 3, 'min_samples_split': 6, 'n_estimators': 100}` for Loan Amount prediction and `{'max_depth': 5, 'min_samples_leaf': 5, 'min_samples_split': 2, 'n_estimators': 100}` for loan duration prediction.

  * Performance for loan amount prediction:

    * R^2: `0.55`

    * MAE: `58.93`

    * MSE: `4528.68`

    * RMSE: `67.30`

  * Performance for loan duration prediction:

    * R^2: `-0.11`

    * MAE: `65.93`

    * MSE: `5096.48`

    * RMSE: `71.38`

Result

The best performing classification model was the `Random Forest Classifier`, which produced a cross-validation accuracy score of `80.46`. The best performing regression model was the `Random Forest Regressor`, which produced a R2 score of `0.55` for loan amount and `-0.11` for loan duration prediction.

The feature `credit_history` and `total_income_log` were observed to be most impactful for the prediction of loan eligibility and maximum loan amount. `loanamount_log` and `loan_amount_term_log` also played an important role in model performance.

The models were evaluated using different metrics and were optimized using GridSearchCV. However, further optimization and testing different models might further increase performance.

Recommendations :-

Based on the analysis, the following recommendations are made:

Loan Eligibility: The key factor for loan eligibility is the credit history, as observed during the EDA. Additionally, an applicant's total income and the loan amount requested have an important impact on approval.

Maximum Loan Amount: Maximum loan amount is heavily influenced by the applicant's income, as observed in the modeling and through simple multiplication. The models suggest to only allow up to 50% of the applicant's income.

Minimum Loan Duration: The minimum loan duration requirement varies and is higher for applicants with low total income and bad credit history. Also, note that the model's accuracy is not very high on loan duration and can be improved further with more data.

Conclusion :-

This project delivered a working machine learning application that classifies loan applicants and predicts the maximum loan amount and minimum duration for ineligible customers, utilizing a range of data analysis techniques and algorithms. The results suggest that credit history, income, and loan amount have a significant impact on the loan approval process.