**Problem Statement**

**“For banks, there is a need for an efficient method of credit appraisal that will help determine whether or not to disburse loans from applicants; its existing system is through manual processing of applications taking much time and thus prone to errors that lead to delays and misjudgments. In the context of a credit approval prediction model for loans by machine learning using applicant demographics, financial history, and credit history information, banks would deliver quick, data-driven decisions with loan default risks minimized. Other major objectives include developing a predictive model, determining the determinants of factors that influence loan decisions, and providing insights to be optimized in the loan approval process. This model can automate loan approvals and speed up decision making; it would also help target marketing efforts towards applicants likely to be approved, and eventually help reduce default rates by identifying high-risk factors. The challenges involved in this problem are handling missing values, imbalance of classes, and feature engineering like an income-to-loan ratio. The benefits that would result from achieving the said goals include better customer satisfaction as the marketing strategies may be optimized and risk management strengthened”**

1. **Introduction**

The approval procedures for bank loans are crucial for maintaining the financial stability and operational efficiency of banks. This project aims to use machine learning models to categorize loan requests by considering various factors like applicant information, financial status, and property details. By analyzing historical loan data patterns, the predictive model helps banks make informed decisions, decrease manual errors, and lower risks related to loan defaults.

Moreover, the solution aims to simplify risk management processes and improve the customer journey. By having more precise forecasts on loan approval requirements, banks can provide quicker responses to applicants, promote transparency, and establish customer confidence. This project highlights how data-driven methods can enhance conventional processes by improving their efficiency, dependability, and scalability within current financial environments.

1. **OBJECTIVE OF THE PROJECT**
2. DESIGN ADVANCED MACHINE LEARNING MODEL, Specially intended for loan approval prediction based on the applicant profile; this model will classify applications with high degree of accuracy, thus significantly reducing the need for manual review processes and speeding up the decision-making process.

2. Identify and order key influencing factors such as applicant income, stability in employment, credit history, co-applicant details, value of property, and loan term. This will bring out patterns and trends necessary for data-driven decision-making.

3. Good optimization techniques that enhance the models based on precision and efficiency performance, as observed in terms of precision, recall, F1-score, and area under the ROC curve, are devised with a purpose to generalize well on novel data.

4. Provide actionable insights for risk management, such as identifying the characteristics of high-risk applicants; predicting the likelihood of a loan default; and suggesting appropriate intervention strategies that reduce financial risks.

5. Enabling better customer experience from easier loan application processing, which presented approval criteria clearly for the customer, thus moving on to improve trust and reduce frustration in the realization of overall satisfaction with banking services. 6. Enabling strategic growth with high-value customer segments for special marketing programs: The model will help banks in attracting and retaining good customers, thus propelling its long-term profitability and competitive edge.

1. **DATASET DESCRIPTION WITH PREPROCESSING DONE AND** FEATURES USED

The data set contains loan application information designed to identify factors that influence loan approvals. It contains a combination of attributes across demographics, financials, and loan details. Below is the detailed description of the dataset and preprocessing applied in developing the model.

Dataset Source:

The data used is taken from the UCI Machine Learning Repository and Kaggle: this diversity and realism in applicant scenarios enhance model robustness.

Key Features:

Demographic Information:

Gender: Male/Female.

Marital Status: Married/Single.

Dependents: Number of dependents (0, 1, 2, 3+).

Education Level: Graduate/Not Graduate.

Employment Status: Self-employed/Not Self-employed.

Financial Information:

Applicant Income: The main applicant's average monthly earnings.

Co-Applicant Income: The average monthly income of a co-applicant, if any.

Loan Information:

Amount borrowed: Sum required to borrow.

Amount Loan Term In terms of the number of months.

Credit History: Binary indicator (1: satisfactory credit history, 0: unsatisfactory/unknown).

Property Information:

Property Area: Type of location (Urban, Semi-Urban, Rural).

Target Variable:

Loan\_Status: Shows whether the loan was approved ('Y') or disapproved ('N').

Preprocessing Steps

1. Managing Missing Values:

Numerical Columns:

LoanAmount: The missing values were replaced by the median, which is less sensitive to outliers than the mean.

Loan\_Amount\_Term: Missing values imputed with the median based on the distribution of term durations (e.g., 360 months as the most common).

Categorical Columns:

Mode imputation was used in variables like Gender, Marital Status, and Dependents, where all of these were filled with the most frequent category.

2. Encode categorical variables:

StringIndexer: It converted the categorical labels like Property Area, Education Level, and Marital Status into numerical indices.

OneHotEncoder: The above indices were converted to binary (dummy) variables to avoid ordinal interaction and to have better interpretability of the model. For example, Property Area was transformed into three independent columns.

3. Feature Transformation and Normalization:

StandardScaler: Applied to ApplicantIncome, Co-applicantIncome, and LoanAmount columns as data distribution needs to be standardized to have zero mean and unit variance.

ApplicantIncome and LoanAmount had outliers; was taken care of with log scaling so the very high numbers wouldn't overrule the model's learning.

4. Feature Engineering:

Total\_Income: Consolidated ApplicantIncome and Co-applicantIncome to create an overall picture of their ability to repay.

Income\_Loan\_Ratio: Total\_Income divided by LoanAmount is calculated. It reflects the ability to service loan repayments that are proportionate to income.

Credit\_Score\_Weighted: Created a weighted credit score feature to give priority to a good credit history above unknown or poor records.

5. Outliers Treatment:

Identify outliers for numerical variables, such as ApplicantIncome using boxplot analysis Capped extreme values to the 95th percentile so that the integrity of the data is preserved without skewing predictions. 6. Train-Test Splitting and Balancing of the Dataset: Stratified sampling divides the set into a training subset and a test subset, such that approved and rejected loans are proportionally represented in both subsets. SMOTE-Synthetic Minority Oversampling TechniqueSMOTE has tackled the problem of class imbalance by generating synthetic examples for the minority class that is Loan\_Status 'N'. Final Features Used for Modeling: Encoded categorical features, for instance Gender, Property Area. Numeric Features standardized and scaled, for instance: ApplicantIncome, LoanAmount. Engineered features-for example, Total\_Income, Income\_Loan\_Ratio, Credit\_Score\_Weighted. Original critical attributes such as Credit History and Loan\_Amount\_Term. Such an extensive preprocessing pipeline can achieve clean, balanced, and enriched datasets for training predictive models to ensure the resulting model to be informative and accurate.

1. **Tables Hardware/Software/ Technique Used**

|  |  |
| --- | --- |
| **Criteria** | **Details** |
| Hardware Configuration | ASUS TUF Gaming F15 |
| Software Configuration | Windows 11 |
| Big Data Tools Used with Version | Apache Spark |
| Python Library Used | Data Manipulation: Pandas, NumPy |
| Visualization Tool | Matplotlib, Seaborn |
| Any other tools, libraries used | Scikit-learn, PySpark MLlib |

1. **Implementation Methodology along with Flowchart**

There is a structured methodology in implementing loan approval prediction, including stages that start with data preparation and a stage called exploratory analysis, feature engineering, model training, and finally evaluation. The following describes each of these steps in detail and also makes use of a flowchart to illustrate the methodology:

**Begin**

**|**

**▼**

**Data Gathering**

**|**

**▼**

**Data Preparation**

**├─ Handling Missing Values**

**├─ Encoding Categorical Variables**

**└─ Scaling Numeric Features**

**|**

**▼**

**Feature Engineering**

**├─ Compute Total Income**

**└─ Compute Income to Loan Ratio**

**|**

**▼**

**Exploratory Data Analysis**

**├─ Visualize Distributions**

**├─ Analyze Feature Correlation**

**└─ Examine Class Imbalance**

**|**

**▼**

**Train-Test Split**

**├─ 80% Train Data**

**└─ 20% Test Data**

**|**

**▼**

**Model Selection**

**├─ Logistic Regression**

**├─ Random Forest Classifier**

**├─ Support Vector Machine (SVM)**

**└─ Gradient Boosting**

**|**

**▼**

**Hyperparameter Tuning**

**├─ Grid Search**

**└─ Cross-Validation**

**|**

**▼**

**Model Evaluation**

**├─ Accuracy**

**├─ Precision**

**├─ Recall**

**├─ F1-Score**

**└─ ROC-AUC**

**|**

**▼**

**Visualization**

**├─ Confusion Matrix**

**├─ ROC Curves**

**└─ Feature Importance**

**|**

**▼**

**Insights and Recommendations**

**▲**

**Conclusion**

**Methodology:**

1. Data Collection:

The set of information used comprises loan applicants with access to demographic, financial, and loan-related information. Key attributes are Gender, Marital Status, Dependents, Applicant Income, Co-applicant Income, Loan Amount, Credit History, Property Area, and the target variable is (Loan\_Status).

2. Clean Data:

-Missing Value Handling The missing values in numerical columns- LoanAmount, Loan\_Amount\_Term-all were filled using the median, and in categorical columns-Gender, Dependents-it was done with mode.

Convert Categorical Feature:- Several categorical features are converted to the numerical format using OneHotEncoder.

Scaling Numeric Features: Continuous variables such as ApplicantIncome, Co-applicantIncome, and LoanAmount were scaled in unison using StandardScaler.

3. Feature Engineering:

- Total Income: ApplicantIncome + co-applicant's Income into a new column to mean that total earning capability.

Income to Loan Ratio: Built a derived feature by performing the computation of Total Income divided by LoanAmount which represents the repayability.

4. Exploratory Data Analysis (EDA):

Distribution Pictures: Employed histograms and box plots to generate distribution and identify outliers.

- Feature Correlation: Developed a correlation heatmap to identify relationships between variables.

Class Imbalance: Target variable is class imbalanced. The classes of Loan\_Status are imbalanced; SMOTE has been applied rightly at other times.

5. Train-Test Split:

Split the dataset into training (80%) and testing (20%) subsets to ensure reliable model evaluation.

6. Model Selection:

Train the data using multiple machine learning algorithms including:

- Logistic Regression

- Random Forest Classifier

- Support Vectors Machine (SVM)

- Gradient Boosting

7. Hyperparameter Tuning:

Performed grid search and cross-validation to optimize hyperparameters for each model, enhancing predictive performance.

**8. Model Evaluation:**

Models were assessed against the following criteria:

- Accuracy: Percentage of correctly predicted instances.

The ratio of actual positive predictions Accuracy. Recall Sensitivity Detect for positive cases.

- F1-Score: Harmonic Mean of precision and recall.

- ROC-AUC: Area under the ROC curve used to assess the classification performance.

9. Confusion Matrix : It featured the breakdown of true positives, false positives, true negatives and false negatives. - ROC Curves: Plot the true positive rates for several different thresholds against the false positive rates with a trade-off. Feature Importance Ranked Features By the contribution to the predictor's power.

10. Conclusion and Recommendations: Recommendations have thus been provided to drive further improvement in the loan approval process with actionable insights on decision improvement, customer satisfaction improvement, and risk mitigation.

11. Conclusion: We successfully automated loan approval predictions using machine learning, allowing for minimal manual effort and also reducing human errors while making this data-driven decision-making process.

1. **Paste your Running Source Code at the end of the report.  
   CODE : ”https://colab.research.google.com/drive/1UAnipEoxR5uGZ4yYuZP-pXTSIauB\_Ihl?usp=sharing”**
2. **Comparative Analysis of PYSPARK ML Models**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model Name** | **Accuracy** | **Precision** | **Recall** | **F1-Score** | **ROC** |
| **Logistic Regression** | **0.78** | **0.75** | **0.80** | **0.77** | **0.82** |
| **Decision Tree Classifier** | **0.76** | **0.74** | **0.78** | **0.76** | **0.76** |
| **Random Forest Classifier** | **0.88** | **0.87** | **0.89** | **0.88** | **0.90** |
| **Support Vector Machine (SVM1)** | **0.82** | **0.80** | **0.81** | **0.80** | **0.83** |
| **K-Nearest Neighbors (KNN)** | **0.74** | **0.72** | **0.75** | **0.73** | **0.73** |

From the comparative analysis:

Analyzing the applied models for machine learning in the prediction of bank loan approvals points out to their strengths and weaknesses. Using the tested model, it appears that the Random Forest model had a good score, with an accuracy rate of 88%, while the ROC-AUC resulted in 0.90 because of its power and efficiency in handling non-linear relationships. Finally, the ensemble learning approach by Random Forest-aggregated results from multiple decision trees-was ultimately associated with high recall (89%) and precision (87%), especially in minimizing false negatives-a factor that rather plays a drastically concerning role in loan approval decisions.

An assessment of feature importance reveals that Credit History, Income to Loan Ratio, and Property Area are by far the most significant predictors of an application being approved. Such results are by no means unexpected in the domain. Here, strong credit history and favorable income-to-loan dynamics tend typically to give rise to good applications.

Both the SVM and Logistic Regression models followed at respective accuracies of 82% and 78%. Although SVM is known to work rather well in high-dimensional feature spaces, it provided balanced precision at 80% and a recall percentage at 81%, hence is a reliable alternative. Logistic Regression was also a good baseline in its simplicity while providing balanced performance.

Better as well as moderately performing at 76%, the Decision Tree model overfitted, and the K-Nearest Neighbors (KNN) algorithm struggled with the high dimensional and possibly noisy data, only yielding an accuracy of 74%.

Overall, the best model seemed to be Random Forest for loan approval, precise and yet simultaneously interpretable and robust. These results not only verify the power of the model but also provide actionable insights with regard to which features are essential for decision-making, thereby providing considerable value to banks while streamlining and optimizing their loan approval processes.

**Write your analytics findings in 2-3 paragraphs justifying why one model performs better compared to other models.**

**Plot all relevant results with description and supporting screenshot (As per project)**

1. **Data Visualization.**

Through the process of data visualization in the project, insights both related to the dataset and toward various machine learning models' performance are derived. Below are detailed summaries about all of the visualizations created, categorized by their purpose and the insights they offer.

7.1 Measuring Performance:

In this work, it has visualized the performance of the models using different performance metrics-like accuracy, precision, recall, and F1-score-collected in the table along with corresponding plots as grouped bar charts for comparison of models.

A graph of different colored bars

Description automatically generated

Key Takeaways:

Random Forest exhibited the highest accuracy and F1-score, making it the most reliable model for predictions.

SVM and Logistic Regression were equally as accurate and with recall, although outcompeted by Decision Tree and KNN.

7.2 Confusion matrix

Confusion matrices were plotted for each model to visualize true positives, true negatives, false positives, and false negatives. This helped understand where models made incorrect predictions and identify patterns in errors.

A diagram of a decision tree

Description automatically generatedA red and blue squares with numbers

Description automatically generated

A red and blue squares with numbers

Description automatically generated

Key Takeaways

It had the fewest false negatives, therefore correctly predicting most of the approved loans.

Decision Tree showed higher false positives, suggesting susceptibility to overfitting.

7.3 ROC Curves

Apart from the models developed, ROC curves were developed for the Decision Tree model, Random Forest model, and SVM model, with both TPR and FPR comparison shown.

A green line graph with a pointy line

Description automatically generated with medium confidence

A graph of a curve

Description automatically generated

A graph of a curve

Description automatically generated

Bottom Line:

Largest area under the curve (AUC = 0.90) was observed in Random Forest, thus testing its higher discrimination between approved and rejected loans. SVM also depicted strong AUC performance, 0.83, while Decision Tree showed moderate performances.

7.4 Feature Importance

It visualized the feature importance of the Decision Tree model as a bar chart to

highlight the contribution of each feature in predicting loan approval.

A graph with text and numbers

Description automatically generated

Key Findings:

Credit history was the most influencing feature. Income to Loan Ratio and Property Area also are highly contributing factors.

7.5 Exploratory Data Analysis

1. Loan Status Distribution: A countplot visualized the proportion of approved (Y) vs. rejected (N) loans.

A graph of a loan status

Description automatically generated with medium confidence

2. Income Distribution by Loan Status: A boxplot was created that showed that the higher combined income tends to have approved loans.

A graph of income distribution by loan status

Description automatically generated

3. Credit Score and Approval Rates of Loans: A bar graph brought out the importance of good credit score as approval rates of loans.

A graph of a credit history

Description automatically generated

1. **Conclusion**

This project effectively utilized machine learning methods to forecast approvals for bank loans, utilizing the Bank Loan Approval dataset to create and assess multiple classification models. The Random Forest model surpassed other models, with the highest accuracy (88%) and ROC-AUC (0.90), showcasing its capacity to manage intricate data relationships and accurately distinguish between accepted and declined loan requests. Credit History, Income to Loan Ratio, and Property Area were found to be the most crucial factors in determining loan approval, which is in line with standard banking procedures according to feature importance analysis. Although Support Vector Machine (SVM) and Logistic Regression models performed decently, they were surpassed by Random Forest in terms of overall performance and reliability. The examination and presentations offered practical data for enhancing the loan approval procedure, such as pinpointing essential factors for quicker decision-making. This project showcases how machine learning can improve risk management, optimize loan approval systems, and offer data-**driven recommendations for financial institutions to streamline operations.**

1. **Relevant References**

**Bank Loan.csv(Kaggle or similar data source)**

**PySpark MLlib Documentation**

**Books and articles on machine learning with PySpark**

**CODE :<https://colab.research.google.com/drive/1UAnipEoxR5uGZ4yYuZP-pXTSIauB_Ihl?usp=sharing>**