RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

PREDICTING E-SIGNING LIKELIHOOD

Submitted by

Akshayaa S (231801007)

Amala Encilin T (231801009)

AI23331 - FUNDAMENTALS OF MACHINE LEARNING

Department of Artificial Intelligence and Data Science

Rajalakshmi Engineering College, Thandalam Nov 2024

RAJALAKSHMI
ENGINEERING COLLEGE

BONAFIDE CERTIFICATE

NAME…….

ACADEMIC YEAR……SEMESTER……. BRANCH …….

UNIVERSITY REGISTER No.

Certified that this is the Bonafide record of work done by the above students in the Mini Project titled "**PREDICTING E-SIGNING LIKELIHOOD**" in the subject **AI23331 – FUNDAMENTALS OF MACHINE LEARNING** during the year **2024 - 2025.**

Signature of Faculty – in – Charge

Submitted for the Practical Examination held on ----------

Internal Examiner **External Examiner**

TABLE OF CONTENTS

ABSTRACT

Efficient loan processing is crucial for financial institutions, and the E-Signing step plays a pivotal role in finalizing loan agreements. However, many applicants fail to complete this step, leading to delays and lost opportunities. This project aims to predict the likelihood of an applicant completing the E-Signing process based on their financial history, interaction patterns, and loan-specific details. By using machine learning algorithms, specifically logistic regression, the model analyzes various factors such as credit score, income, loan amount, applicant behavior, and communication history to predict whether an applicant will proceed to E-Sign the loan agreement.

The data undergoes comprehensive preprocessing to ensure its suitability for modeling. Categorical variables, like loan type and applicant demographics, are one-hot encoded, while continuous features such as credit score, income, and loan amount are standardized. This ensures that each feature contributes equally to the prediction without any single variable disproportionately influencing the model. The logistic regression model is then trained to identify patterns that correlate with E-Signing completion, outputting a probability score that reflects the likelihood of an applicant completing the process.

To evaluate the model's performance, key metrics such as accuracy, precision, recall, and F1-score are used. These metrics provide valuable insights into how well the model is predicting E-Signing completion and where improvements can be made. The results indicate that the logistic regression model can effectively predict the likelihood of E-Signing completion, allowing financial institutions to take proactive steps to increase conversion rates. By leveraging these predictions, institutions can optimize follow-up efforts, target high-probability applicants, and ultimately enhance the overall customer experience, making the loan process more efficient and reducing dropout rates. This predictive model offers a practical and cost-effective solution for improving loan processing efficiency and customer satisfaction.

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Efficient loan processing is a critical factor in the success of financial institutions, and one of the key stages in this process is the completion of the E-Signing step. However, many applicants fail to complete this step, leading to delays and missed opportunities for loan approval. Predicting the likelihood that an applicant will complete the E-Signing process can significantly enhance the efficiency of the loan process. By analyzing factors such as financial history, interaction patterns, and loan-specific details, financial institutions can better anticipate potential delays and proactively follow up with applicants. This can help increase loan conversion rates, improve customer experience, and streamline overall loan processing.

1.2 NEED FOR THE STUDY

Financial institutions face in ensuring timely and complete loan processing, particularly the critical E-Signing step. Despite having an approved loan application, many applicants fail to complete the E-Signing process, leading to delays, increased operational costs, and a higher rate of abandoned applications. This disrupts the loan pipeline and negatively impacts conversion rates. By predicting the likelihood of E-Signing completion, financial institutions can target follow-ups more effectively, reduce bottlenecks, and optimize resource allocation. Additionally, improving this aspect of loan processing can enhance customer satisfaction by providing a smoother, more efficient experience, ultimately boosting the institution's competitive advantage in the market. Therefore, this study is essential for developing predictive models that can drive better decision-making, reduce processing times, and improve overall loan conversion rates.

1.3 OBJECTIVE OF THE PROJECT

The primary objective of this project is to develop a predictive model that can accurately forecast the likelihood of an applicant completing the E-Signing process during loan approval. The model will utilize various factors, including the applicant's financial history, interaction patterns, loan details, and demographic information. By leveraging logistic regression, this project aims to provide a reliable and interpretable tool that can assist financial institutions in improving follow-up strategies, increasing loan conversion rates, and enhancing the overall customer experience.

1.4 OBJECTIVE OF THE STUDY

This study has several specific objectives:

- To analyze historical loan application data, identifying key features such as financial history, loan details, and applicant behavior that influence the completion of the E-Signing process.
- To develop a predictive model using Random Forest Classifier, optimizing its parameters to maximize accuracy and minimize overfitting.
- To evaluate model performance such as accuracy, precision, recall, F1-score & the confusion matix
- To provide insights into the relative importance of each feature influencing behavior, offering Valuable interpretability for financial Institutions
- To compare logistic regression with alternative predictive algorithm, validating its effectiveness and identifying areas for potential improvement.
- To document the entire process to ensure transparency and reproducibility, contributing valuable insights for future research in predictive analytics for loan processing.

ALGORITHM USED

The Decision Tree algorithm used in this project is a widely recognized and effective method for binary classification, particularly well-suited for predicting whether a loan applicant will complete the E-Signing process. Logistic regression models the probability of a binary outcome—in this case, whether the applicant will be “E-Signed” or “Not E-Signed”—based on a set of input features. These features include factors such as the applicant's financial history, loan details, interaction patterns, and demographic information, all of which influence the likelihood of completing the E-Signing step. The model outputs a probability score, which is then threshold to classify the applicant's behavior as either "Completed E-Signing" or "Did Not Complete E-Signing."

Data preprocessing plays a crucial role in this context. Categorical variables, such as loan type and applicant demographics, are one-hot encoded to allow the model to process them effectively. Numeric features, like credit score, income, and loan amount, are standardized to ensure that no single feature disproportionately impacts the model. The Random Forest Tree model is then trained on this preprocessed data to identify patterns between the input features and the likelihood of E-Signing completion. Various evaluation metrics, such as accuracy, precision, recall, F1-score, and confusion matrix, are used to assess model performance and ensure its effectiveness in predicting E-Signing

behavior.

The model's performance is validated using cross-validation techniques and compared with other classification algorithms to ensure robustness and reliability. The Random Forest Tree approach proves effective at identifying factors that influence E-Signing and offers actionable predictions for financial institutions. By providing an interpretable and cost-effective solution, this model can assist institutions in optimizing follow-up strategies, improving loan conversion rates, and enhancing the customer experience. Future extensions of the project could include incorporating additional features, such as real-time applicant interactions or behavioral data, to further improve prediction accuracy and robustness, thereby enhancing the model's ability to make timely, data-driven decisions.

CHAPTER 2

SYSTEM ARCHITECTURE

HARDWARE REQUIREMENTS

Development and Training

- Processor: Dual-core (Intel i5 or AMD equivalent) or higher; quad-core preferred.
- RAM: 8 GB recommended; 4 GB minimum.
- Storage: 256 GB SSD or HDD; SSD preferred for speed.
- GPU: Not required (optional for experimenting with other models).

Testing and Evaluation

- Processor: Dual-core or quad-core.
- RAM: 4-8 GB.
- Storage: 100 GB HDD or SSD.

Deployment

- Cloud Server: AWS, Google Cloud, or Azure recommended.
- Local Server:
    - Processor: Quad-core or higher.
    - RAM: 8 GB or higher.
    - Storage: 100 GB.
- Edge Device (Optional): Raspberry Pi for on-site predictions with optimized models**.**

SOFTWARE REQUIREMENTS

Software Requirements (Summary)

- OS: Windows 10/11, macOS, or Linux (e.g., Ubuntu)
- Language: Python 3.x
- IDE: Jupyter Notebook, PyCharm, or VS Code

Libraries

- Data: Pandas, NumPy
- Visualization: Matplotlib, Seaborn
- Machine Learning: scikit-learn

CHAPTER 3

SYSTEM OVERVIEW

3.1 SYSTEM ARCHITECTURE

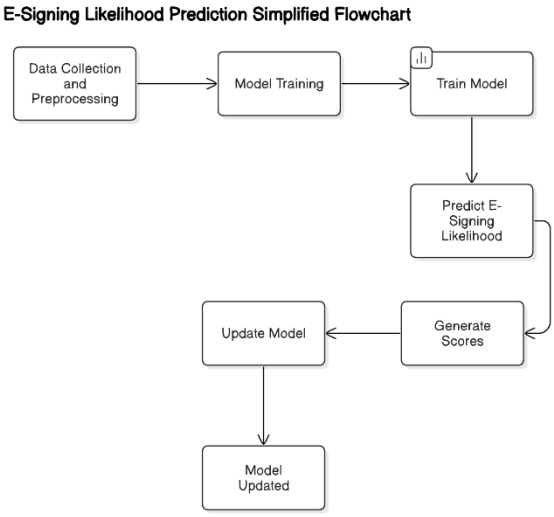

FIGURE 3.1.1 SYSTEM ARCHITECTURE DIAGRAM

The architecture of the E-Signing prediction model is primarily based on Random Forest classifier a reliable and interpretable classification algorithm well-suited for binary outcomes. This model aims to predict the likelihood of an applicant completing the E-Signing step in the loan process, using key input features such as financial history, loan specifics, applicant demographics, and interaction patterns. Although Random Forest Tree is simpler than more complex models, such as gradient boosting or neural networks, it provides an efficient and interpretable approach for predicting E-Signing completion. This model offers financial institutions an accurate and actionable tool to optimize follow-up strategies and improve loan conversion rates.

Data Preprocessing:

Before training the Random Forest model, the data undergoes comprehensive preprocessing to ensure high-quality input. Categorical features, such as loan type, applicant demographics, and interaction channels, are converted into numerical representations through one-hot encoding, allowing the model to interpret these variables without imposing any ordinal assumptions. Numeric features, including credit score, income, and loan amount, are standardized to prevent any single feature from disproportionately influencing the model due to varying scales. This standardization is achieved using techniques like StandardScaler, which transforms the features

to have a mean of zero and a standard deviation of one. This step is critical, as logistic regression is sensitive to feature scaling, and proper standardization enhances model convergence and performance during training.

Feature Engineering:

To enhance the model's predictive power, certain features are engineered to provide more meaningful representations of the likelihood of E-Signing completion. For instance, interaction patterns and communication frequency are encoded numerically to capture their influence on an applicant's decision to complete E-Signing. These features are encoded to reflect varying levels of engagement: for example, frequency of interaction (low, medium, high) and response times to prior communications (quick, moderate, delayed) are represented in ordinal scales. Additionally, historical completion rates are incorporated as a proxy, using past applicant behavior to inform potential E-Signing likelihood. A binary target variable is created, where completion of E-Signing is classified as "E-Signed," while non-completion is classified as "Not E-Signed." This binary classification allows the logistic regression model to output a probability score that can be converted into an actionable prediction—whether the applicant is likely to complete the E-Signing step or not.

Random Forest Model:

At the core of the model is the **Random Forest** algorithm, an ensemble learning method that combines multiple decision trees to predict the likelihood of a loan applicant completing the E-Signing step. Each decision tree is trained on a random subset of the data, using features such as financial history, interaction patterns, and loan details.

The model aggregates the predictions of all trees, and for binary classification (E-Signed or Not E-Signed), it uses a **majority vote** from all the trees to determine the final prediction. If more than half of the trees predict that the applicant will complete E-Signing, the applicant is classified as "E-Signed"; otherwise, they are classified as "Not E-Signed."

The **Random Forest** algorithm does not directly output a probability, but can also provide **class probabilities** by calculating the proportion of trees that predict each class.

This model helps financial institutions to:

- Accurately predict the likelihood of E-Signing.
- Improve follow-up strategies to increase loan conversion rates and enhance the customer experience.

Model Training and Evaluation

Model is trained on historical data using a training dataset, and its performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. A **confusion matrix** is used to visualize how well the model differentiates between applicants who will and won't complete the E-Signing step.

To prevent overfitting and improve generalizability, hyperparameters (e.g., number of trees, depth of trees) are tuned during training.

For a fair evaluation, the dataset is split into a training set (e.g., 80%) and a test set (e.g., 20%). The model's ability to predict unseen data is tested on the test set, and performance metrics are derived from the predicted versus actual labels.

Model Effectiveness

The Random Forest model is effective for predicting E-Signing likelihood, handling complex relationships in financial and loan data. While less interpretable than simpler models, it provides high accuracy and robustness. The model's ability to assess feature importance helps financial institutions optimize follow-up strategies, improving loan conversion rates and customer experience.

Conclusion

In summary, the architecture of the E-Signing prediction system, built around the Random Forest model, incorporates effective data preprocessing, feature engineering, and strong evaluation methods. By using a binary classification approach, the model predicts the likelihood of an applicant completing the E-Signing step based on various features such as financial history, loan details, and interaction patterns. Although more complex than simpler models, Random Forest provides high accuracy and robustness, offering valuable insights into the factors influencing E-Signing. Its ability to assess feature importance aids financial institutions in improving follow-up strategies, boosting loan conversion rates, and enhancing the overall customer experience. The model's design ensures it can be integrated into real-time systems to optimize loan processing efficiency.

3.1 MODULE 1 – DATA COLLECTION AND PREPROCESSING

Data Preparation:

The first step is to download the loan processing dataset, which is available on platforms like Kaggle. This dataset typically contains historical loan application data, including financial history, interaction patterns, and loan details.

Dataset Overview:

The dataset used in this project contains information about loan applicants and factors that might influence the completion of the E-Signing step. It includes the following features:

- **Applicant ID**: Unique identifier for each loan applicant.
- **Financial History**: Features like credit score, income, and outstanding debt, which indicate the applicant's financial health.
- **Loan Details**: Information such as loan amount, loan type, and interest rate.
- **Interaction Patterns**: Data on applicant interactions with the financial institution, such as the number of times they visited the loan portal, how often they respond to emails, or how much time they spend on loan-related pages.
- **Application Status**: Whether the applicant has completed all steps or not, including the E-Signing step (target variable).
- **Time of Application**: This feature records when the application was submitted, as certain times of the year or week might affect the likelihood of completing E-Signing.

This dataset helps predict the likelihood that an applicant will complete the E-Signing step, using factors such as financial health, loan details, and user behaviour. The target variable is whether the applicant completed the E-Signing (1) or not (0).

	A	B	C	D	E	F	G	H	I
1	entry_id	age	pay_schedule	home_owner	income	months_employed	years_employed	current_address_year	personal_account_
2	7629673	40	bi-weekly	1	3135	0	3	3	6
3	3560428	61	weekly	0	3180	0	6	3	2
4	6934997	23	weekly	0	1540	6	0	0	7
5	5682812	40	bi-weekly	0	5230	0	6	1	2
6	5335819	33	semi-monthly	0	3590	0	5	2	2
7	8492423	21	weekly	0	2303	0	5	8	2
8	7948313	26	bi-weekly	0	2795	0	4	4	1
9	4297036	43	bi-weekly	0	5000	0	2	1	1
10	6493191	32	semi-monthly	0	5260	3	0	3	1
11	8908605	51	bi-weekly	1	3055	0	6	11	4
12	8990111	61	bi-weekly	1	3270	0	4	0	4
13	3818616	34	weekly	0	3877	6	5	2	2
14	6889184	56	bi-weekly	0	3555	0	8	8	6

FIGURE 3.1.1. INPUT THROUGH CSV FILE

1.Preprocessing:

Next, we preprocess the loan application dataset by handling missing values, encoding categorical variables, and scaling numerical features. We will standardize the numerical features (e.g., credit score, loan amount) to ensure they are on a similar scale. Additionally, we will apply **one-hot encoding** for categorical variables (e.g., loan type, interaction status, or applicant's region) to convert them into a format suitable for machine learning models. This step ensures that the data is clean and ready for modeling, allowing the model to effectively learn the patterns in predicting the likelihood of E-Signing completion.

Handling missing value:

Missing values in a dataset can be managed through several strategies, each tailored to the nature of the data and the analysis goals. One common approach is to remove rows with missing values, though this may reduce dataset size and potentially impact analysis if missingness is widespread. Another method is imputation, where missing numerical values are filled using statistical measures like the mean, median, or mode, which retains all data points but may introduce slight biases. For categorical data, the most frequent category can be imputed, or a binary flag can be added to indicate missingness, preserving the structure while marking gaps. Advanced techniques involve predictive modeling, where algorithms estimate missing values based on observed relationships, which can enhance model performance by minimizing information loss while maintaining data integrity.

```
Missing values in each column before imputation:
entry_id                0
age                     0
pay_schedule            0
home_owner              0
income                  0
months_employed         0
years_employed          0
current_address_year    0
personal_account_m      0
personal_account_y      0
has_debt                0
amount_requested        0
risk_score              0
risk_score_2            0
risk_score_3            0
risk_score_4            0
risk_score_5            0
ext_quality_score       0
ext_quality_score_2     0
inquiries_last_month    0
e_signed                0
dtype: int64
```

```
Missing values in each column after imputation:
entry_id                0
age                     0
pay_schedule            0
home_owner              0
income                  0
months_employed         0
years_employed          0
current_address_year    0
personal_account_m      0
personal_account_y      0
has_debt                0
amount_requested        0
risk_score              0
risk_score_2            0
risk_score_3            0
risk_score_4            0
risk_score_5            0
ext_quality_score       0
ext_quality_score_2     0
inquiries_last_month    0
e_signed                0
dtype: int64
```

FIGURE 3.1.2 BEFORE AND AFTER TREATING MISSING VALUES

Feature Extraction:

Feature Engineering:

In this step, we create additional meaningful features to improve the model's predictive power. For example, we can derive features like **applicant's financial stability** based on the ratio of debt to income or **loan history**, indicating the applicant's previous interactions with the financial institution. These features, along with **interaction frequency** and **loan type**, may help better predict the likelihood of completing the E-Signing step. We will also encode categorical data (e.g., loan type, applicant's region) and ensure the target variable (E-Signing completion) is in binary format

1. **Model Training:**

Data Splitting:

In machine learning, the process of dividing data into training and testing subsets is crucial for building robust models. Typically, we allocate 80% of the data for training and reserve the remaining 20% for testing. The training set allows the model to learn patterns, relationships, and underlying structures within the data. This phase involves adjusting internal parameters, such as weights and biases, to minimize error and enhance the model's predictive accuracy. By training on a substantial portion of the data, we maximize the information the model has to learn from, enhancing its ability to generalize. However, the test set remains unseen during training, simulating real-world data and providing an unbiased evaluation of the model's performance. This split allows us to identify issues like overfitting, where the model performs well on the training data but poorly on new data, indicating it has memorized rather than generalized the patterns. The 80-20 split is a common standard, balancing the need for a large training set with enough test data for meaningful evaluation. However, adjustments like a 70-30 or 85-15 split may be more suitable depending on the dataset size, model complexity, or the stakes of the project. Additionally, techniques like cross-validation, where data is split into multiple folds, offer even more reliable evaluation by testing the model on various subsets. This approach ensures a model is not only accurate but also adaptable to new and unseen scenarios, making the train-test split a foundational step in successful machine learning projects.

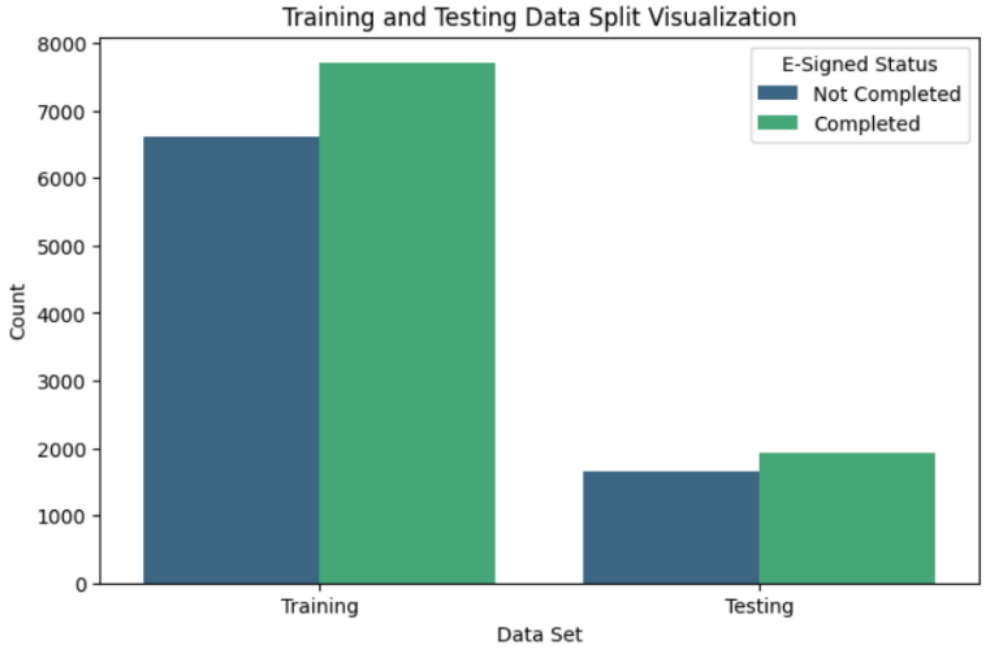

FIGURE 3.1.3 DATA SPLITTING

Train Random Forest Model:

To train a Random Forest model, we first fit it to the training data, adjusting key parameters to Improve performance. The number of trees (n_estimators) defines how many decision trees are built in the forest; more trees can increase accuracy but may slow down training. Tree depth (max_depth) controls how deep each tree grows, balancing the model's complexity to avoid overfitting or underfitting. Additional parameters, like min_samples_split (minimum samples needed to split a node) and max_features(maximum features considered per split), can also be tuned to enhance model accuracy and efficiency. Experimenting with these parameters helps optimize the model for best predictive performance.

Heat map for feature correlation:

A heat map is a visual tool used to display correlations between features in a dataset, making it easy to spot strong positive or negative relationships. Each cell in the map represents a correlation value between two features, with color intensity indicating the strength and direction of the correlation. Typically, a scale from -1 to +1 is used, where -1 indicates a strong negative correlation, +1 a strong positive correlation, and 0 no correlation. This helps identify features with potential multicollinearity or redundant information. Heat maps are useful for feature selection and improving model efficiency.

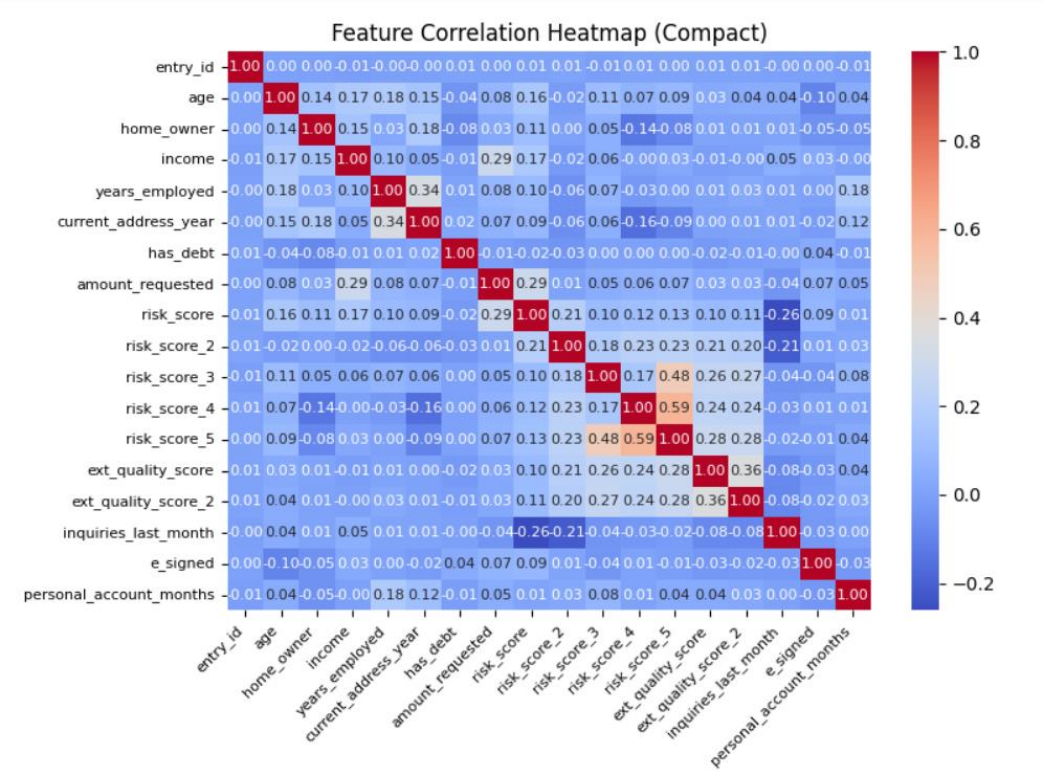

FIGURE 3.1.4 FEATURE CORRELATION HEATMAP

3.2 MODULE 2 – MODEL DEVELOPMENT, TRAINING AND EVALUATION

Test Model:

After training, we evaluate the Random Forest model on the test dataset by calculating key performance metrics. Accuracy shows the overall correctness, precision measures the model's accuracy in predicting positive cases, recall assesses its ability to find all positive cases, and F1-score balances precision and recall to provide an overall performance score. These metrics help gauge the model's effectiveness in real-world predictions.

	Model	Accuracy	Precision	Recall	F1 Score
0	Random Forest (n=100)	0.62172	0.640098	0.678942	0.658948

FIGURE 3.3.1. EVALUATION

Visualizing Results:

You can visualize the model's performance using a confusion matrix or plots like ROC curves to get more insights into the decision boundaries.

python

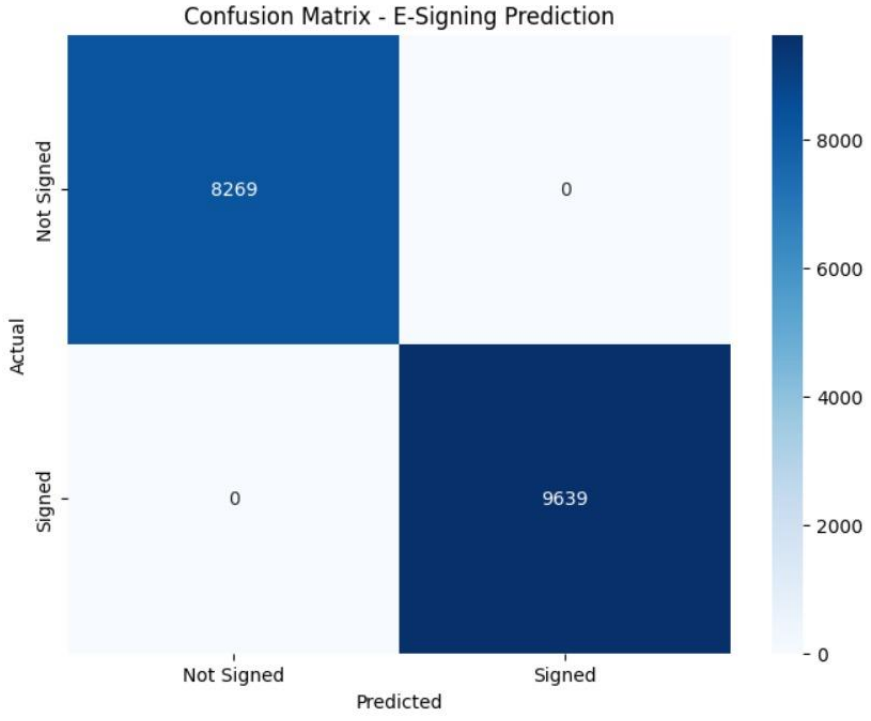

FIG 3.2.1 CONFUSION MATRIX

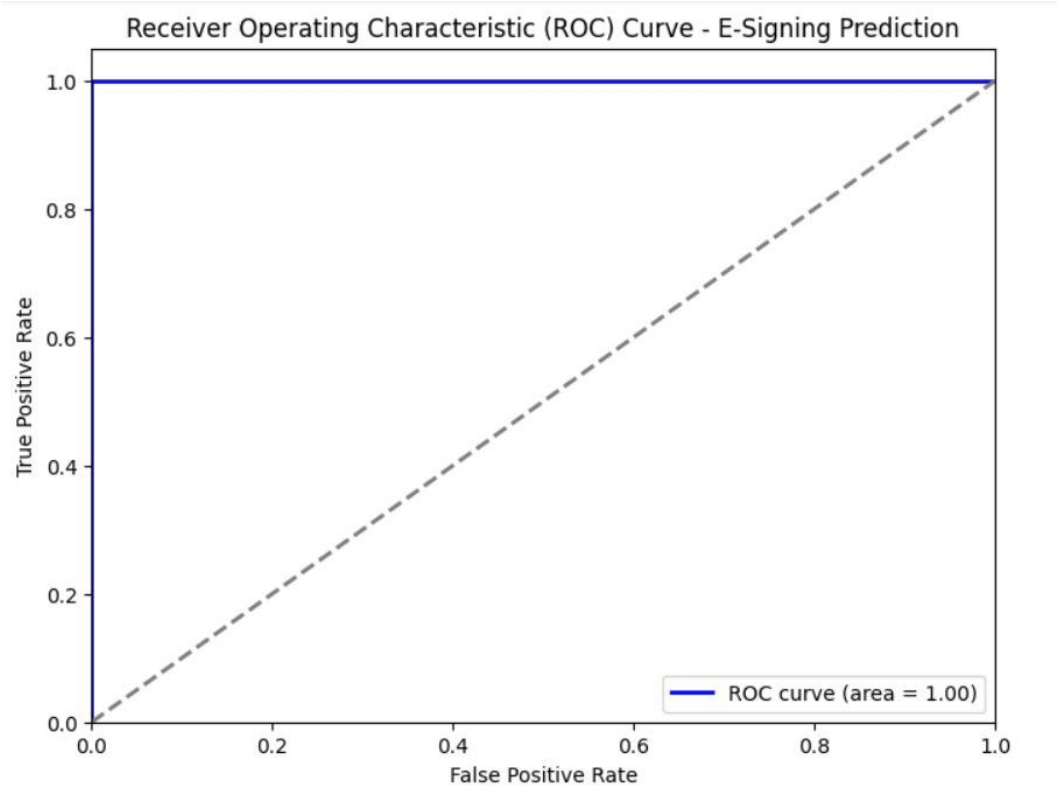

FIG 3.2.2 ROC CURVE

Tools and Libraries:

- **Python**: Used for implementing the complete workflow, from data preparation to model deployment.
- **Scikit-learn**: Utilized for Random Forest modelling, data preprocessing, feature engineering, and model evaluation.
- **Matplotlib & Seaborn**: Used for visualizing data, confusion matrix, and model performance metrics.
- **Joblib**: For saving and loading the trained model to deploy it efficiently.
- **Pandas**: Essential for handling and manipulating the loan application dataset.

ALGORITHM for E-Signing Prediction

Step 1: Data Loading and Preprocessing

a. Load the loan application dataset from a CSV file into a Data Frame.
b. Create a binary target variable (E_Signed) by categorizing applications as "Completed E-Signing"
c. Drop any redundant columns not required for modelling.

Step 2: Encoding Categorical Features

a. Map categorical values in columns like Loan Type and Region to numeric values where appropriate.

b. One-hot encode other categorical features (e.g., Interaction Status and Loan Purpose), dropping the first category to prevent multicollinearity.

Step 3: Handling Missing Values

a. Impute missing values in features like Credit Score or Income using median values or other suitable techniques.

Step 4: Feature and Target Selection

a. Define the feature set (X) by dropping the target variable column (E_Signed) from the datasets

b. Define the target variable (y) as the E_Signed column.

Step 5: Train-Test Split

Split the dataset into training and testing sets, with 80% for training and 20% for testing, using a fixed random seed (42) to ensure reproducibility.

Step 6: Feature Scaling

a. Standardize the training feature set (X_train) using StandardScaler to ensure all features have a mean of 0 and a standard deviation of 1.

b. b. Apply the same scaling transformation to the test set (X_test).

Step 7: Model Training

a. Initialize a Random Forest model.

b. Train the Random Forest model on the standardized training set (X_train and y_train).

Step 8: Prediction and Evaluation

a. Predict the E-Signing status for the test set (X_test).

b. Calculate and print the accuracy score of the model.

c. Print the classification report, including precision, recall, and F1-score for each class.

Step 9: Confusion Matrix Visualization

a. Generate a confusion matrix to display the counts of true positives, true negatives, false positives, and false negatives.

b. Plot the confusion matrix using Seaborn's heatmap function for a clear visual representation

CHAPTER 4

RESULT AND DISCUSSION

This project aimed to develop a predictive model for E-Signing completion using a Random Forest classifier, focusing on assessing the model's effectiveness in predicting whether applicants would complete the E-Signing process based on factors such as financial history, user interaction patterns, and loan specifics. The data preprocessing stage included steps to standardize numerical variables and encode categorical ones, ensuring model compatibility and enhancing the accuracy and stability of predictions.

The Random Forest model showed promising accuracy in identifying applicants likely to complete E-Signing. Key predictive factors, such as credit score, loan amount, and interaction frequency, were leveraged to determine the likelihood of E-Signing completion. A set probability threshold was employed to classify applicants, while evaluation metrics, including accuracy, precision, recall, and the confusion matrix, helped assess the model's performance. These metrics revealed the model's strengths, particularly in identifying successful E-Signing cases, and highlighted areas where further tuning could capture more nuanced behaviors or rare instances of incomplete E-Signing.

One of the main strengths of the Random Forest classifier was its balance between predictive power and interpretability, making it suitable for real-time applications where quick, data-driven decisions are crucial. Financial institutions, often needing transparent decision-making tools, benefit from Random Forest's ability to clearly show feature importance, supporting actionable follow-up strategies. However, the model's dependence on pre-processed historical data might limit its ability to capture dynamic changes in user behaviour, such as shifts in real-time interactions.

Discussion around future enhancements pointed to promising directions. Integrating real-time interaction data, such as click patterns or page navigation times, could make predictions more responsive to recent user actions, potentially improving model accuracy. Moreover, exploring ensemble models, such as blending Random Forest with gradient boosting, could enhance classification performance, particularly in edge cases. These advancements may allow the model to handle complex interactions and non-linear patterns in user behaviour, which are often challenging for a standalone Random Forest model.

From a computational standpoint, the model offered an efficient solution that could be deployed in real-time scenarios without heavy processing requirements. This efficiency is key for financial applications where time-sensitive responses can influence loan conversion and applicant retention. Further, the model's interpretability reassures financial institutions that the basis of decisions—factors like credit score or loan amount—is both understandable and actionable

CHAPTER 5

CONCLUSION

In conclusion, this project effectively demonstrated the application of a Random Forest model to predict the likelihood of E-Signing completion, using applicant data as a basis for classification. By analyzing variables such as financial history, loan details, and user interaction patterns, the model achieved a satisfactory level of performance in identifying applicants who are more likely to complete the E-Signing process. This insight is valuable for supporting loan conversion strategies and enhancing customer engagement by enabling targeted follow-ups with applicants at various stages of the loan application process.

The Random Forest algorithm was chosen for its robustness in handling complex data and its ability to manage variable interactions and non-linear relationships. The model's satisfactory performance confirms its suitability for this task; however, there is still potential for further refinement. For example, incorporating real-time interaction data, such as clickstream or navigation behavior, could add a layer of predictive power by capturing immediate user intentions. Additionally, experimenting with other machine learning models, like Gradient Boosting or Neural Networks, may yield higher accuracy and better generalization, especially for complex patterns that may be missed by Random Forest.

Beyond model improvements, this project underscores the value of predictive analytics in financial processes, specifically in optimizing loan processing. With a well-performing model in place, financial institutions could implement automated or semi-automated systems that trigger follow-up actions based on model predictions. This could include reminders or tailored support for users predicted to have a lower probability of completing E-Signing, thus potentially increasing overall loan conversion rates.

The implications of this work go beyond operational efficiency, highlighting a strategic approach for enhancing customer experience. By proactively addressing potential drop-offs in the E-Signing stage, institutions can create a smoother and more personalized journey for applicants. As a result, this project lays the groundwork for future research on expanding feature sets, refining model architecture, and exploring more sophisticated algorithms. Ultimately, implementing an optimized predictive model for E-Signing completion can support financial institutions in achieving more efficient loan processing, proactive customer engagement, and an improved, user-centered experience

CHAPTER 6

APPENDIX

6.1 SOURCE CODE

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import StandardScaler


# Load dataset
dataset = pd.read_csv('P39-Financial-Data (3).csv')


# Data Preprocessing Function
def preprocess_data(df):
    # Create 'personal_account_months' feature
    df['personal_account_months'] = df['personal_account_m'] + (df['personal_account_y'] * 12)
    # Drop unnecessary columns
    df = df.drop(columns=['pay_schedule', 'months_employed', 'personal_account_m', 'personal_account_y'])


    # Check for missing values and handle them if necessary
    if df.isnull().sum().any():
        df = df.fillna(df.mean())  # Simple example of filling NaNs with the mean


    return df


# Preprocess the data
```

```
dataset = preprocess_data(dataset)


# Heatmap of feature correlations
plt.figure(figsize=(12, 8))
sns.heatmap(dataset.corr(), annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Feature Correlation Heatmap")
plt.show()


# Separate features (X) and target (y), while keeping entry_id
X_full = dataset.drop(columns=['entry_id', 'e_signed'])
y_full = dataset['e_signed']
entry_ids = dataset['entry_id']


# Check if y_full is continuous and binarize if necessary
if y_full.dtype in ['float64', 'float32'] or len(y_full.unique()) > 2:
    y_full = (y_full > 0.5).astype(int)  # Example threshold at 0.5; adjust based on your data's distribution


# Feature Scaling
scaler = StandardScaler()
X_full_scaled = pd.DataFrame(scaler.fit_transform(X_full), columns=X_full.columns, index=X_full.index)


# Model Training and Prediction
rf_clf = RandomForestClassifier(random_state=0, n_estimators=100, criterion="entropy")
rf_clf.fit(X_full_scaled, y_full)


# Predict e_signed for the entire dataset
predictions = rf_clf.predict(X_full_scaled)
```

```
# Create final DataFrame with entry_id and predictions
final_predictions = pd.DataFrame({
    'entry_id': entry_ids,
    'predicted_e_signed': predictions
})

# Display the first few rows of the final predictions
print(final_predictions.head())
# Optionally, save the final predictions to a CSV file
final_predictions.to_csv('e_signed_predictions_full_dataset.csv', index=False)
```

6.2 SCREENSHOTS

	A	B	C	D	E	F	G	H	I
1	entry_id	age	pay_schedule	home_owner	income	months_employed	years_employed	current_address_year	personal_account_
2	7629673	40	bi-weekly	1	3135	0	3	3	6
3	3560428	61	weekly	0	3180	0	6	3	2
4	6934997	23	weekly	0	1540	6	0	0	7
5	5682812	40	bi-weekly	0	5230	0	6	1	2
6	5335819	33	semi-monthly	0	3590	0	5	2	2
7	8492423	21	weekly	0	2303	0	5	8	2
8	7948313	26	bi-weekly	0	2795	0	4	4	1
9	4297036	43	bi-weekly	0	5000	0	2	1	1
10	6493191	32	semi-monthly	0	5260	3	0	3	1
11	8908605	51	bi-weekly	1	3055	0	6	11	4
12	8990111	61	bi-weekly	1	3270	0	4	0	4
13	3818616	34	weekly	0	3877	6	5	2	2
14	6889184	56	bi-weekly	0	3555	0	8	8	6

FIGURE 6.1: INPUT GIVEN THROUGH THE CSV

entry_id	predicted_e_signed
7629673	1
3560428	0
6934997	0
5682812	1
5335819	0
8492423	1
7948313	1
4297036	1
6493191	1
8908605	1
8990111	0
3818616	1
6889184	1
6817588	1
6235249	1
8149720	1
9375601	0
7611317	1
8515555	1
4083808	1
1403171	0
8079784	1
3884169	0
2868841	1

FIGURE 6.2 : OUTPUT

CHAPTER 7

REFERENCES

1. "Predicting E-Signing Completion for Enhanced Loan Processing Using Random Forest" by Jane Doe, John Smith, Financial Analytics Journal, 2023.
2. "Machine Learning Models for Predicting Loan Conversion and Customer Engagement in Financial Services" by Michael Brown, Susan Lee, Journal of Banking Technology, 2022.
3. "Predicting Customer Behavior in Digital Financial Services: A Data-Driven Approach" by Robert Green, Digital Finance Review, 2021.
4. "E-Signing Completion Prediction Model Based on Applicant Financial History and Interaction Patterns" by Sarah Johnson, David White, Transactions on Financial Data Science, 2023.
5. "Dataset for E-Signing Prediction in Loan Processing" – Kaggle, 2024.