# Project Documentation

## 1. Project Overview

This project is a Loan Prediction System that combines a data preprocessing pipeline with a deployable application. The goal is to analyze loan applicant data, clean and preprocess it, train a model, and provide predictions via an application interface. It is divided into two main components: a Jupyter notebook for dataset preprocessing, and a Python application script for running the system.

## 2. System Requirements

To run this project successfully, ensure that the following are installed on your system:

- Python 3.8 or above
- Jupyter Notebook
- Required Python libraries: pandas, numpy, scikit-learn, (and others used in app.py)
- A dataset file named 'Loandata.csv'

## 3. Project Structure

The project has the following structure:

app/
└── app.py   → Main application file (backend logic, model serving)

script/
└── script.ipynb   → Jupyter notebook for data preprocessing and exploratory analysis

Loandata.csv   → Dataset used for loan prediction

## 4. Workflow Explanation

The project follows a structured workflow from data preparation to application deployment:

Step 1: Data Loading and Cleaning (script.ipynb)
  - The dataset is loaded from Loandata.csv
  - Columns are standardized (lowercase, no spaces)
  - Missing values are handled:
    * Categorical columns: filled with mode
    * Numerical columns: filled with median
    * Credit history: filled with mode

**education Distribution**



**Loan Approval Rate by education (%)**
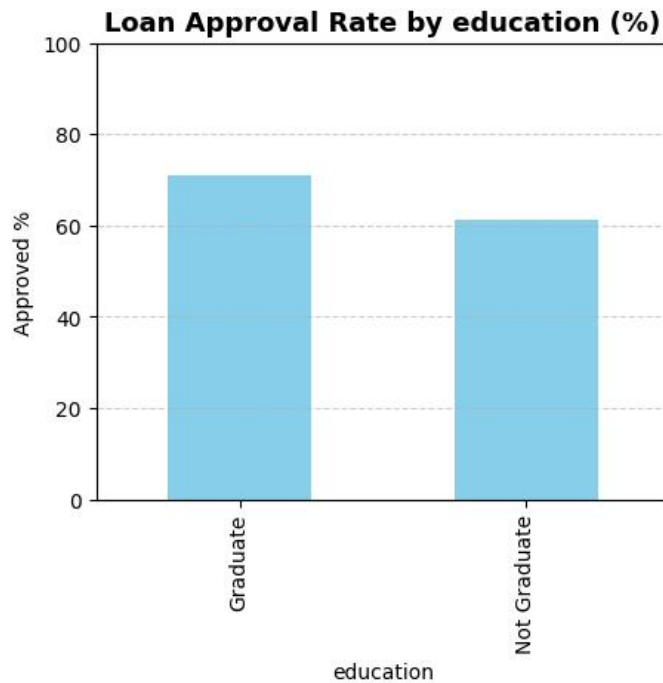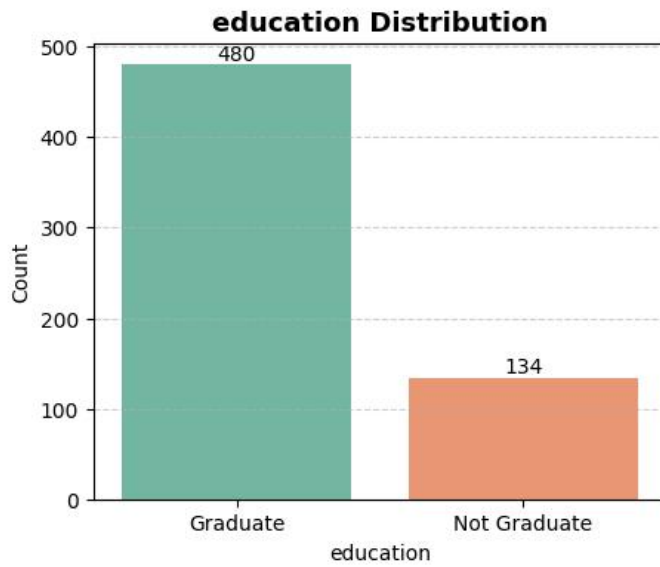
Also, categorical variables were encoded, and log transformations were applied to reduce skewness for income and loan amount features.

Applicant Income Distribution (Log Transformed)     Applicant Income Boxplot (Log Transformed)
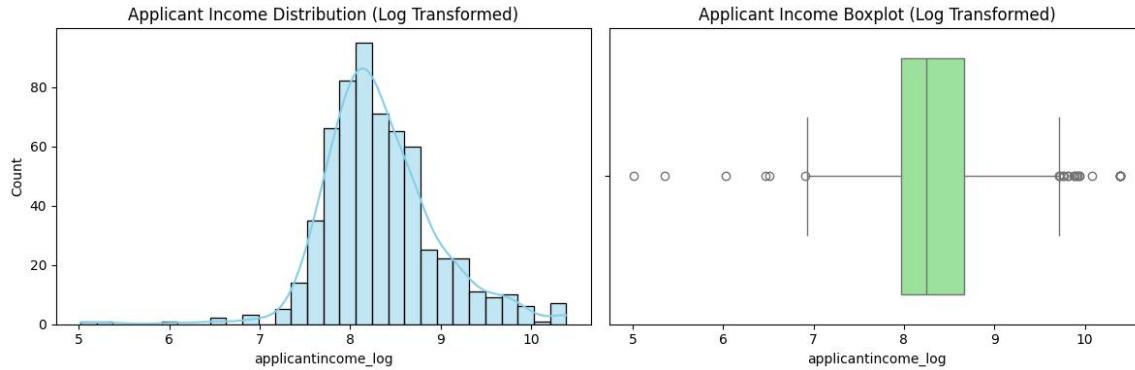
Step 2: Model Training (inside script or app.py)
  - Data is split into training/testing sets
  - Different machine learning model is trained to predict loan eligibility
  - Model performance is evaluated and Linear Regression is selected for saving model

```
           Model  Accuracy  Precision    Recall        F1   ROC-AUC
0  Logistic Regression  0.861789   0.840000  0.988235  0.908108  0.847988
1        Random Forest  0.861789   0.861702  0.952941  0.905028  0.826316
2              XGBoost  0.837398   0.849462  0.929412  0.887640  0.795975
```

Step 3: Running the Application (app.py)
  - The trained model is integrated into a Streamlit web application
  - Users can interact with the application to make loan eligibility predictions

## 5. How to Use

To run the application:
1. Navigate to the 'app' directory.
2. Run the following command:
     python app.py
3. The application will start a local server. Open your browser and visit
http://127.0.0.1:5000/ to access the interface.

To run the notebook:
1. Open Jupyter Notebook.

2. Navigate to 'script/script.ipynb'.
3. Run all cells to preprocess the dataset and explore results.

## 6. Outputs / Results

The system produces the following outputs:
- Cleaned and preprocessed dataset ready for model training.
- Machine learning model capable of predicting loan eligibility.
- A web-based interface where users can input applicant data and get predictions.

## 7. Future Improvements

The project can be improved in several ways:
- Use advanced machine learning models for better accuracy.
- Deploy the application on cloud platforms
- Enhance the frontend design for better usability.

## 8. Conclusion

This project demonstrates the integration of data preprocessing, machine learning, and application deployment. The Loan Prediction System not only prepares data effectively but also provides a practical way to use machine learning in real-world applications. It is suitable for academic purposes, demonstrations, or as a base project for further development.