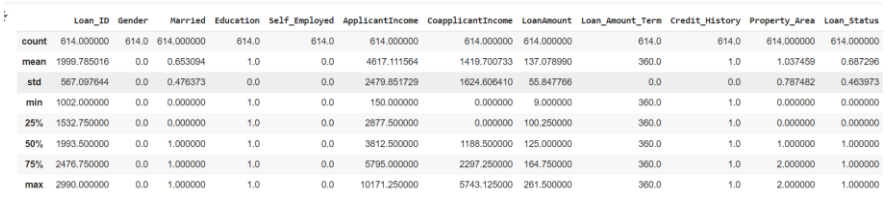
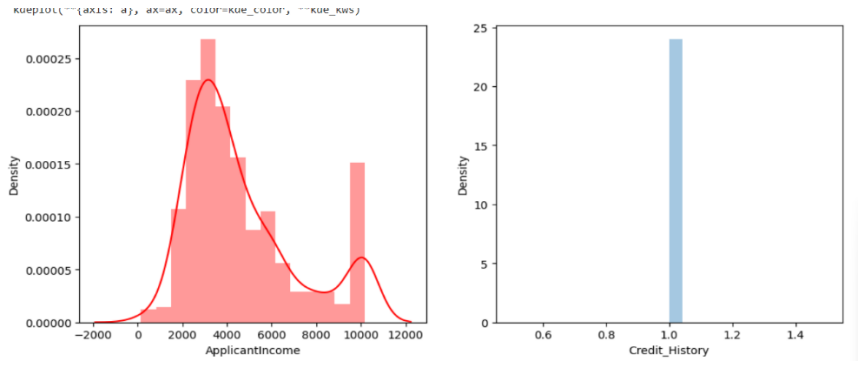
Data Collection and Preprocessing Phase

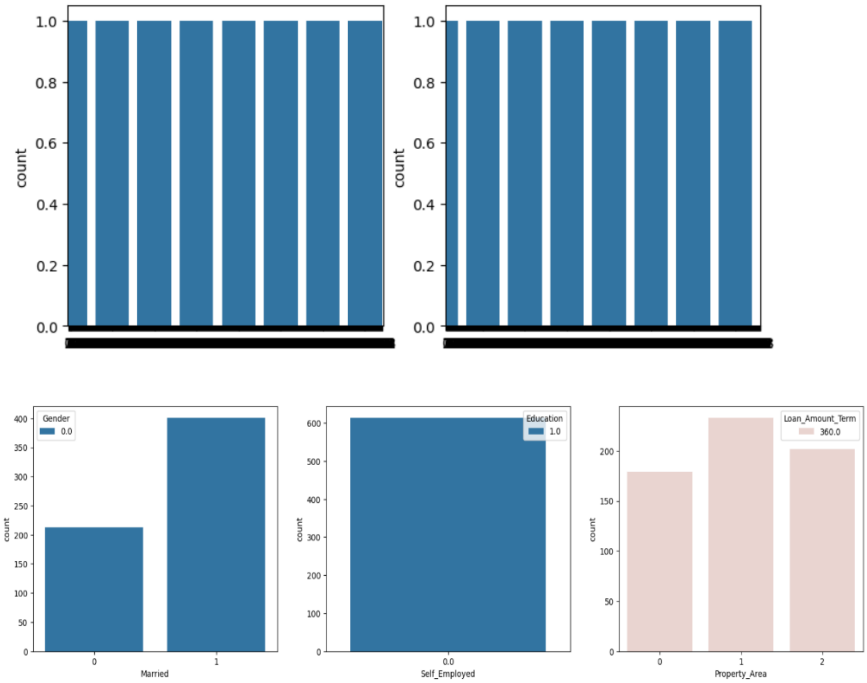
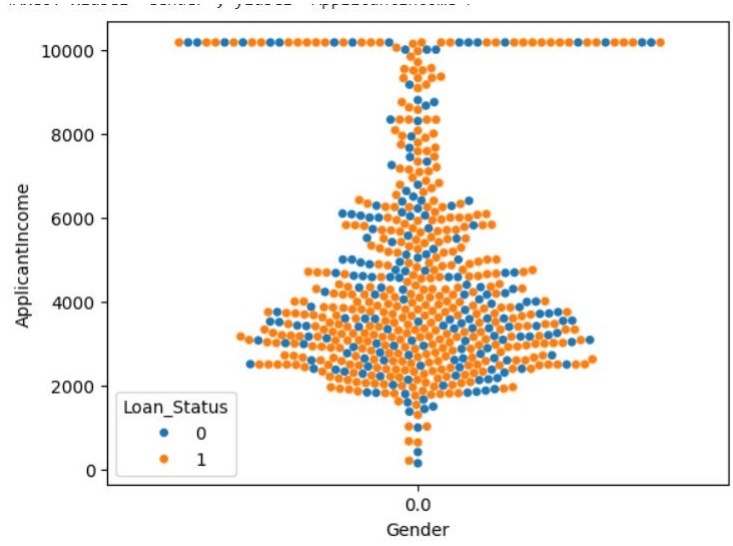
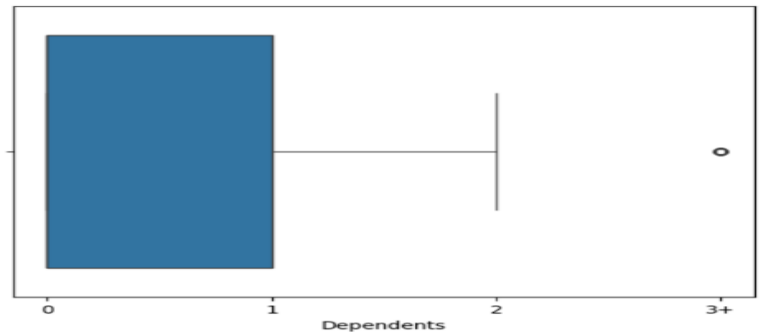
Date	11 July 2024
Team ID	740052
Project Title	Smart Lender - Applicant Credibility Prediction For Loan Approval
Maximum Marks	6 Marks

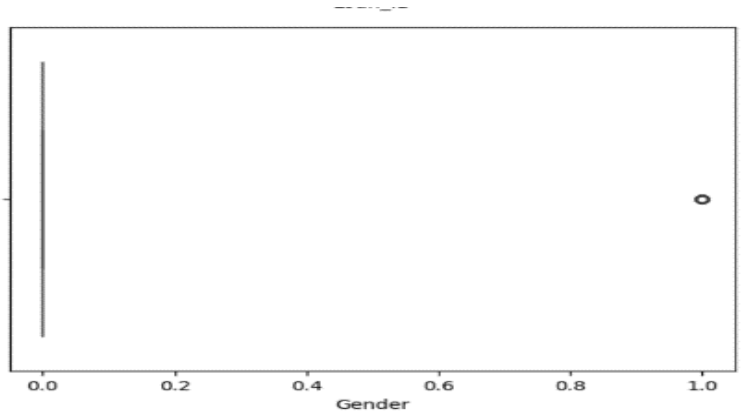
Data Exploration and Preprocessing Template

Data exploration and preprocessing for loan approval prediction begins with thorough exploratory data analysis (EDA) to understand distributions and relationships among features like credit scores, income levels, and loan amounts. Addressing missing data is critical; imputation methods such as mean, median, or predictive modeling are applied. Numerical features are scaled to ensure uniformity in influence, while categorical variables are encoded (e.g., one-hot encoding) for compatibility with machine learning models. These steps ensure the dataset is ready for training robust models that predict loan approval outcomes accurately based on processed data.

Section	Description
Data Overview	Dimensions: 614 rows × 13 columns Descriptive statistics: 
Univariate Analysis	

	Loan_ID	Gender	Married	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
count	614.000000	614.0	614.000000	614.0	614.0	614.000000	614.000000	614.000000	614.0	614.0	614.000000	614.000000
mean	1999.785016	0.0	0.653094	1.0	0.0	4617.111564	1419.700733	137.078990	360.0	1.0	1.037459	0.687296
std	567.097644	0.0	0.476373	0.0	0.0	2479.851729	1624.606410	55.847766	0.0	0.0	0.787482	0.463973
min	1002.000000	0.0	0.000000	1.0	0.0	150.000000	0.000000	9.000000	360.0	1.0	0.000000	0.000000
25%	1532.750000	0.0	0.000000	1.0	0.0	2877.500000	0.000000	100.250000	360.0	1.0	0.000000	0.000000
50%	1993.500000	0.0	1.000000	1.0	0.0	3812.500000	1188.500000	125.000000	360.0	1.0	1.000000	1.000000
75%	2476.750000	0.0	1.000000	1.0	0.0	5795.000000	2297.250000	164.750000	360.0	1.0	2.000000	1.000000
max	2990.000000	0.0	1.000000	1.0	0.0	10171.250000	5743.125000	261.500000	360.0	1.0	2.000000	1.000000

Bivariate Analysis	
Multivariate Analysis	
Outliers and Anomalies	



Data Preprocessing Code Screenshots

Loading Data

```
#importing the dataset which is in the csv file
import pandas as pd
data =pd.read_csv('/content/loan_prediction.csv')
data
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	Urban	Y
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	Rural	N
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	Urban	Y
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	Urban	Y
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	Urban	Y
...	...	...	...	...	...	...	...	...	...	...	...	...	...
609	LP002978	Female	No	0	Graduate	No	2900	0.0	71.0	360.0	1.0	Rural	Y
610	LP002979	Male	Yes	3+	Graduate	No	4106	0.0	40.0	180.0	1.0	Rural	Y
611	LP002983	Male	Yes	1	Graduate	No	8072	240.0	253.0	360.0	1.0	Urban	Y
612	LP002984	Male	Yes	2	Graduate	No	7583	0.0	187.0	360.0	1.0	Urban	Y
613	LP002990	Female	No	0	Graduate	Yes	4583	0.0	133.0	360.0	0.0	Semiurban	N

614 rows × 13 columns

Handling Missing Data

```
#Code for handlling with Null values
data['Gender'].fillna(data['Gender'].mode()[0],inplace=True)
```

```
data['Married'].fillna(data['Married'].mode()[0],inplace=True)
```

```
data['Dependents'].fillna(data['Dependents'].mode()[0],inplace=True)
```

```
data['Self_Employed'].fillna(data['Self_Employed'].mode()[0],inplace=True)
```

```
data['LoanAmount'].fillna(data['LoanAmount'].mode()[0],inplace=True)
```

```
data['Loan_Amount_Term'].fillna(data['Loan_Amount_Term'].mode()[0],inplace=True)
```

```
data['Credit_History'].fillna(data['Credit_History'].mode()[0],inplace=True)
```

Data Transformation

```
#Handling categorical values
data['Gender']=data['Gender'].map({'Female':1,'Male':0})
```

```
data['Property_Area']=data['Property_Area'].map({'Urban':2,'Semiurban':1,'Rural':0})
```

```
data['Married']=data['Married'].map({'Yes':1,'No':0})
```

```
data['Education']=data['Education'].map({'Graduate':1,'Not Graduate':0})
```

```
data['Loan_Status']=data['Loan_Status'].map({'Y':1,'N':0})
```

Feature Engineering	Attached the codes in final submission.
Save Processed Data	-