Visualizing And Analyzing The Data

Date	01 November 2022
Team ID	PNT2022TMID00521
Project Name	Smart Lender - Applicant Credibility Prediction for Loan Approval

IMPORTING THE LIBRARIES

- Import the required libraries for the model to run. The first step is usually importing the libraries that will be needed in the program.

```
[2]  import pandas as pd
     import numpy as np
     import pickle
     import matplotlib.pyplot as plt
     %matplotlib inline
     import seaborn as sns
     import sklearn
     from sklearn.tree import DecisionTreeClassifier
     from sklearn.ensemble import GradientBoostingClassifier
     from sklearn.neighbors import KNeighborsClassifier
     from sklearn.model_selection import RandomizedSearchCV
     from sklearn.model_selection import train_test_split
     from sklearn.preprocessing import StandardScaler
     from sklearn.metrics import accuracy_score,classification_report,confusion_matrix,f1_score
```

READING THE DATASET

- Our dataset format might be in .csv, excel files, .txt, .json, etc. We can read the dataset with the help of pandas.
- In pandas, we have a function called read_csv() to read the dataset. As a parameter, we have to give the directory of the CSV file.

```
df = pd.read_csv("/content/drive/MyDrive/loan_data.csv")
df
```

	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

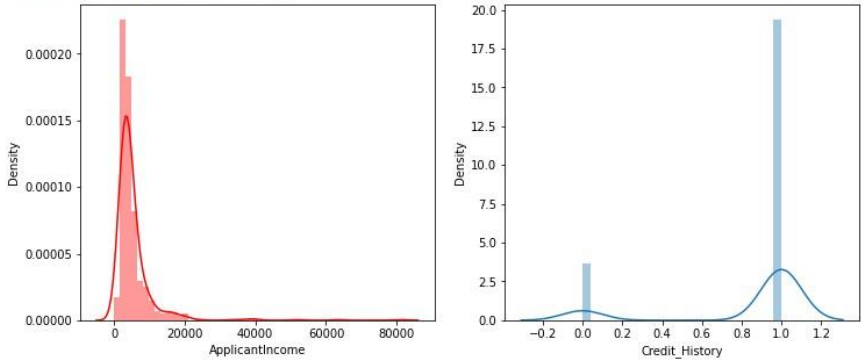
614 rows × 13 columns

UNIVARIATE ANALYSIS

- Univariate analysis is understanding the data with a single feature. Here we have displayed two different graphs such as distplot and countplot.
- The Seaborn package provides a wonderful function distplot. With the help of distplot, we can find the distribution of the feature. To make multiple graphs in a single plot, we use a subplot.

```
#plotting the using distplot
plt.figure(figsize=(12,5))
plt.subplot(121)
sns.distplot(df['ApplicantIncome'], color='r')
plt.subplot(122)
sns.distplot(df['Credit_History'])
plt.show()
```

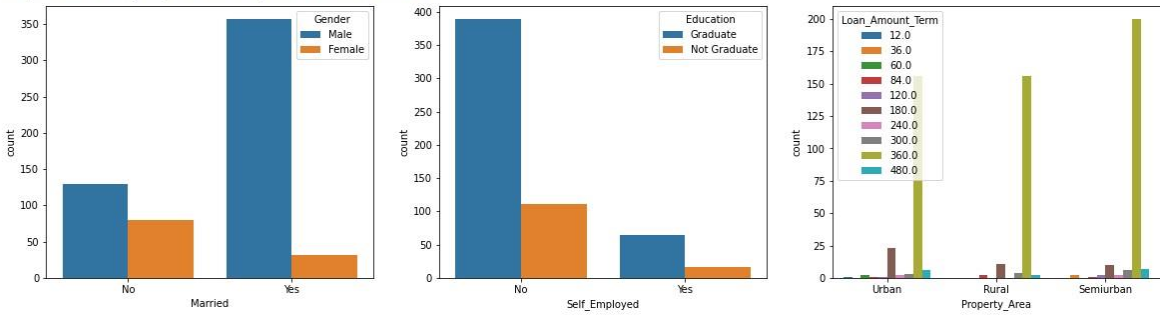
```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will
  warnings.warn(msg, FutureWarning)
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will
  warnings.warn(msg, FutureWarning)
```



BIVARIATE ANALYSIS

```
plt.figure(figsize=(20,5))
plt.subplot(131)
sns.countplot(df['Married'], hue=df['Gender'])
plt.subplot(132)
sns.countplot(df['Self_Employed'], hue=df['Education'])
plt.subplot(133)
sns.countplot(df['Property_Area'], hue=df['Loan_Amount_Term'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, th
  FutureWarning
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, th
  FutureWarning
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, th
  FutureWarning
<matplotlib.axes._subplots.AxesSubplot at 0x7f9eb3d6ff10>
```

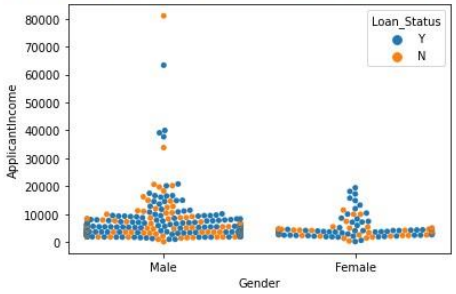


MULTIVARIATE ANALYSIS

- Multivariate analysis is to find the relation between multiple features. Here we have used a swarm plot from the seaborn package.

```
sns.swarmplot(df['Gender'], df['ApplicantIncome'], hue = df['Loan_Status'])
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/_decorators.py:43: FutureWarning: Pass the following variables as keyword args: x, y. Fr
  FutureWarning
/usr/local/lib/python3.7/dist-packages/seaborn/categorical.py:1296: UserWarning: 67.5% of the points cannot be placed; you may want to
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.7/dist-packages/seaborn/categorical.py:1296: UserWarning: 33.0% of the points cannot be placed; you may want to
  warnings.warn(msg, UserWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f9eb3bd9a90>
```



DESCRIPTIVE ANALYSIS

- Descriptive analysis is to study the basic features of data with the statistical process. Here pandas have a worthy function called describe. With this describe function we can understand the unique, top, and frequent values of categorical features. And we can find mean, std, min, max and percentile values of continuous features.

```
df.describe()
```

	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History
count	614.000000	614.000000	592.000000	600.00000	564.000000
mean	5403.459283	1621.245798	146.412162	342.00000	0.842199
std	6109.041673	2926.248369	85.587325	65.12041	0.364878
min	150.000000	0.000000	9.000000	12.00000	0.000000
25%	2877.500000	0.000000	100.000000	360.00000	1.000000
50%	3812.500000	1188.500000	128.000000	360.00000	1.000000
75%	5795.000000	2297.250000	168.000000	360.00000	1.000000
max	81000.000000	41667.000000	700.000000	480.00000	1.000000

Code is uploaded in the following drive link:

https://colab.research.google.com/drive/10YDz5VLr60QmNikdFTSWBxKUMaqxPi2w