Bank Loan Classification

By: Bhuwan Khatiwada.

Problem Statement:

The task requires to analyze a dataset of bank loans to extract hidden meanings and information from the dataset and to create a machine learning algorithm capable of making predictions of bank loans from the dataset provided.

Dataset Description

The dataset was provided in a spreadsheet format i.e. an Excel file. The dataset consisted of 16 columns including the target variable. The figure below shows the snapshot of the dataset.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
ID	Age	Gender	Experience	Income	ZIP Code	Family	CCAvg	Education	Mortgage	Home Ownership	Personal Loan	Securities Account	CD Account	Online	CreditCard
1	25		1	49	91107	4	1.60	1	0		0	1	0	0	0
2	45		19	34	90089	3	1.50	1	0		0	1	0	0	0
3	39		15	11	94720	1	1.00	1	0		0	0	0	0	0
4	35		9	100	94112	1	2.70	2	0		0	0	0	0	0
5	35		8	45	91330	4	1.00	2	0		0	0	0	0	1
6	37		13	29	92121	4	0.40	2	155		0	0	0	1	0
7	53		27	72	91711	2	1.50	2	0		0	0	0	1	0
8	50		24	22	93943	1	0.30	3	0		0	0	0	0	1
9	35		10	81	90089	3	0.60	2	104		0	0	0	1	0
10	34	M	9	180	93023	1	8.90	3	0	Home Owner	1	0	0	0	0
11	65		39	105	94710	4	2.40	3	0		0	0	0	0	0
12	29		5	45	90277	3	0.10	2	0		0	0	0	1	0
13	48		23	114	93106	2	3.80	3	0		0	1	0	0	0
14	59		32	40	94920	4	2.50	2	0		0	0	0	1	0
15	67		41	112	91741	1	2.00	1	0		0	1	0	0	0
16	60		30	22	95054	1	1.50	3	0		0	0	0	1	1
17	38	M	14	130	95010	4	4.70	3	134	Rent	1	0	0	0	0
18	42		18	81	94305	4	2.40	1	0		0	0	0	0	0
19	46	M	21	193	91604	2	8.10	3	0	Rent	1	0	0	0	0

Figure 1: Bank Loan Dataset

The columns of the dataset are as follows:

- ID: ID of the customer
- Age: Age of the customer
- Gender: M for Male, F for Female and O for Others
- Experience: Amount of work experience in years
- Income: Amount of annual income (in thousands)
- Home Ownership: Home Owner, Rent and Home Mortgage.
- Zipcode: Postal code in which the client lives
- Family: Number of family members
- CCAvg: Average monthly spending with the credit card (in thousands)
- Education: Education level (1: bachelor's degree, 2: master's degree, 3: advanced/professional degree)
- Mortgage: Value of home mortgage, if any (in thousands)

- Securities Account: Does the customer have a securities account with the bank?
- CD Account: Does the customer have a certificate of deposit account (CD) with the bank?
- Online: Does the customer use the internet banking facilities?
- CreditCard: Does the customer use a credit card issued by the bank?
- Personal Loan: Did this customer accept the personal loan offered in the last campaign? (Target Variable)

Data Exploration and Preprocessing:

The dataset was loaded into a pandas dataframe using the pandas module. Data exploration was performed using the several functions provided by the pandas library such as df.head(), df.tail(), df.info() etc.

ID	Age	Gender	Experience	Income	ZIP Code	Family	CCAvg	Education	Mortgage	Home Ownership	Personal Loan	Securities Account	CD Account	Online	CreditCard
1	25	NaN	1	49.0	91107	4	1.6	1	0	NaN	0	1	0	0.0	0
2	45	NaN	19	34.0	90089	3	1.5	1	0	NaN	0	1	0	0.0	0
3	39	NaN	15	11.0	94720	1	1.0	1	0	NaN	0	0	0	0.0	0
4	35	NaN	9	100.0	94112	1	2.7	2	0	NaN	0	0	0	0.0	0
5	35	NaN	8	45.0	91330	4	1.0	2	0	NaN	0	0	0	0.0	1

Figure 2: Output using the head() function

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 5000 entries, 1 to 5000
Data columns (total 15 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Age                 5000 non-null   int64
 1   Gender              480 non-null    object
 2   Experience          5000 non-null   int64
 3   Income              4933 non-null   float64
 4   ZIP Code            5000 non-null   int64
 5   Family              5000 non-null   int64
 6   CCAvg               5000 non-null   float64
 7   Education           5000 non-null   int64
 8   Mortgage            5000 non-null   int64
 9   Home Ownership      480 non-null    object
 10  Personal Loan       5000 non-null   object
 11  Securities Account  5000 non-null   int64
 12  CD Account          5000 non-null   int64
 13  Online              4960 non-null   float64
 14  CreditCard          5000 non-null   int64
dtypes: float64(3), int64(9), object(3)
memory usage: 625.0+ KB
```

Figure 3:Output of info() function

The dataset has 5000 entries as seen from the info() function (also can be checked using df.shape()). Info function also returned the data types of the variables which helps us in better understanding the dataset.

After performing basics data exploration, it is necessary to preprocess the data for further processing. For the preprocessing phase following tasks were performed.

- ## Dropping Columns

The column gender has only 480 non-null values out of 5000 entries. This could be because people are not comfortable sharing their gender and chose not to give the information during data collection.

Normally, for missing values imputation could be applied but since the actual data is in very scare quantity imputation may cause the dataset to lose its original characteristics. Therefore, it makes more sense to completely drop the column instead.

Similar is the case with the case with the column "Home-ownership". Although, this variable may have a noteworthy impact on the target variable, due to the risk of hampering the characteristics of data this column will also be dropped.

- ## Renaming Columns

For the given dataset, there is no need for checking duplicate entries because the nature of variables allows for duplicates.

Here, the columns "Income", "CCAVG" and "Home Mortgage" have their values described in thousands. So, this information is added in respective column names for more clarity.

- ## Changing Data Types

Here, the data type of "ZIP Code" is "int64" but since we cannot perform any meaningful mathematical arithmetic computation with this variable, the data type of this variable is changed to "object" type.

Also, the "Education" variable is shown as of type "int64" while it is a categorical variable so converting the type of "Education" as well to “Category”

Similarly, the variables "Securities Account", "CD Account", "Online" and "CreditCard" are also going to be changed to categorical type.

Finally, the target variable "Personal Loan" is also changed to categorial type as there can only be two possible values for this variable.

- ### Handling Empty Values

The columns “Income(in thousands)” and “Online” has several empty values which needed to be filled. The empty values in “Online” column was replaced or filled using backfill and the empty values of the “Income(in thousands)” was replaced with the mean values of the income values.

- ### Handling Outliers

Finally, for the data preprocessing phase the outliers were removed from the columns in which they were detecting along with the whole entry relating to the outliers.

Descriptive Statistics

The Pandas df.describe() was used to generate a table that gave results of certain statistics like mean, median, quartiles , min and max values etc.

	Age	Experience	Income(in thousands)	ZIP Code	Family	CCAvg(in thousands)	Education	Mortgage(in thousands)	Personal Loan	Securities Account	CD Account	Online	CreditCard
count	4934.000000	4934.000000	4934.000000	4934.0	4934.000000	4934.000000	4934.0	4934.000000	4934.0	4934.0	4934.0	4934.0	4934.0
unique	NaN	NaN	NaN	467.0	NaN	NaN	3.0	NaN	2.0	2.0	2.0	2.0	2.0
top	NaN	NaN	NaN	94720.0	NaN	NaN	1.0	NaN	0.0	0.0	0.0	1.0	0.0
freq	NaN	NaN	NaN	164.0	NaN	NaN	2078.0	NaN	4468.0	4419.0	4636.0	2940.0	3483.0
mean	45.571544	20.333401	72.549097	NaN	2.389947	1.929597	NaN	56.413660	NaN	NaN	NaN	NaN	NaN
std	11.345845	11.312065	44.988309	NaN	1.148576	1.741221	NaN	101.373342	NaN	NaN	NaN	NaN	NaN
min	24.000000	0.000000	8.000000	NaN	1.000000	0.000000	NaN	0.000000	NaN	NaN	NaN	NaN	NaN
25%	36.000000	11.000000	39.000000	NaN	1.000000	0.700000	NaN	0.000000	NaN	NaN	NaN	NaN	NaN
50%	46.000000	20.000000	64.000000	NaN	2.000000	1.500000	NaN	0.000000	NaN	NaN	NaN	NaN	NaN
75%	55.000000	30.000000	93.750000	NaN	3.000000	2.500000	NaN	101.000000	NaN	NaN	NaN	NaN	NaN
max	97.000000	43.000000	224.000000	NaN	4.000000	10.000000	NaN	635.000000	NaN	NaN	NaN	NaN	NaN

Figure 4 Descriptive Statistics of all columns

Data Visualization and Exploratory Data Analysis:

Several plots were generated using different visualization libraries like Matplotlib and Seaborn. The visulizations were instrumental in showing the truths hidden inside the data.

Here are some of the findings,

- People with higher income were more likely to receive loan as shown by the plot.
- There is no relation between having an experience and getting a loan.
- Age also didn’t seem to have an impact in the chances of getting a loan.
- Its more likely to get a loan with higher number of family members.
- There is no relation with a Zip code and a persons chances of getting a loan.
- It is more likely to get a loan with Higher CCAvg value.

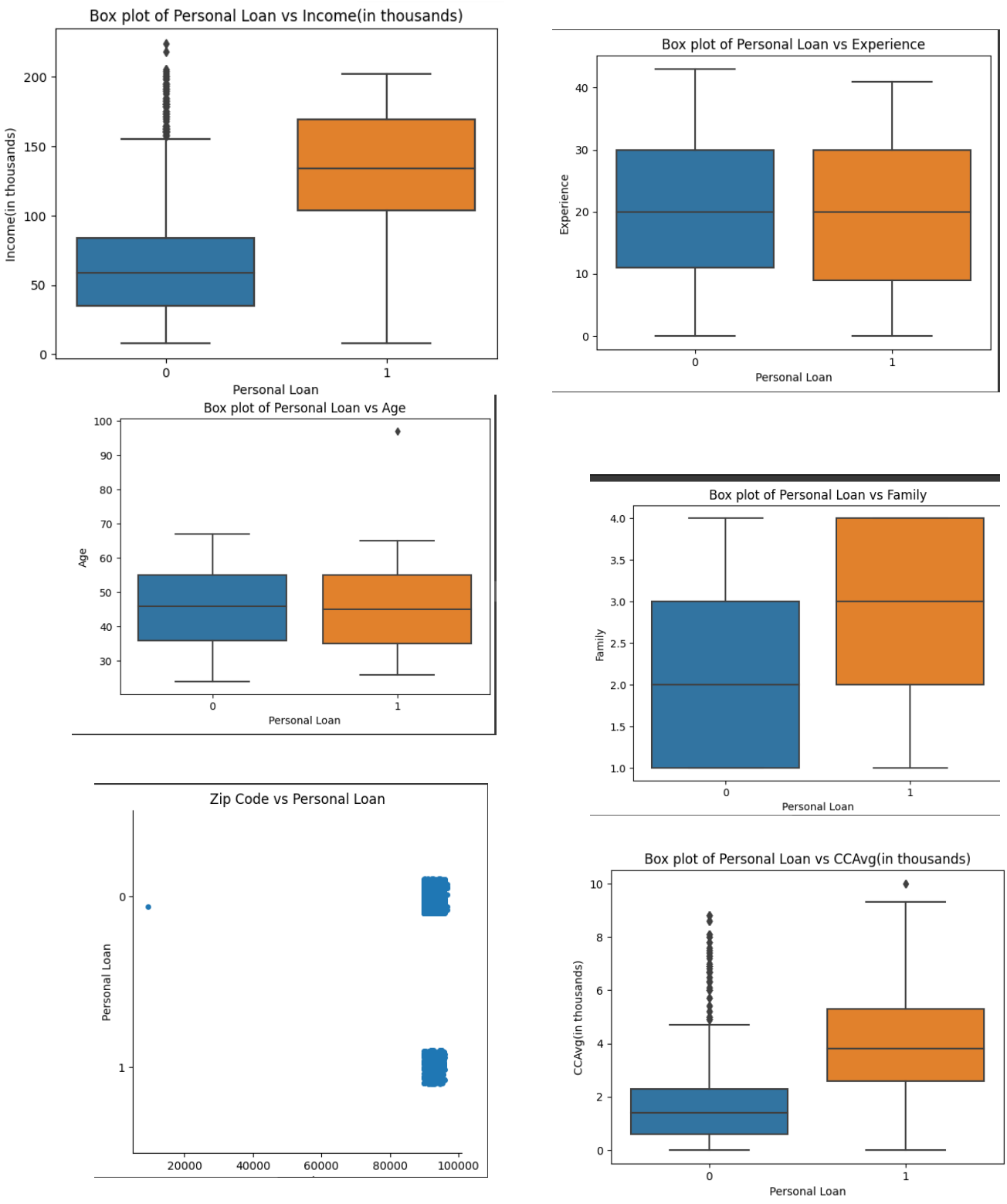

Figure51 Data Visualization Plots

-

Model Development

For this task three models were used for experimental approach. The tree algorithms are DecisionTreeClassifier, RandomForestClassifier and HistGradientBoostingClassifier .

Using sklearn, the dataset was split into training and validation splits in ratio of 0.8 to 0.2.

For training, the target variable was removed from the dataset and other columns were used as features.

For experimentation, three different sets of features were used.:

- **Features-1:** All the columns of the **preprocessed** dataset.
- **Features-2:** The features which were assumed to have more impact based on the visualizations. i.e. "Income(in thousands)","CCAvg(in thousands)","Family", "Mortgage(in thousands)", "Securities Account","Online", "CD Account"
- **Features-3:** Only numerical variables from the above feature list.

For measuring the performance of the models accuracy was used as metric. The accuracy score of different models with different features are given below:

- **HistGradientBoostingClassifier**
  - **Feature-1:** 0.9837
  - **Feature-2:** 0.9452
  - **Feature-3:** 0.9442
- **DecisionTreeClassifier**
  - **Feature-1:** 0.9807
  - **Feature-2:** 0.9351
  - **Feature-3:** 0.9513
- **RandomForestClassifier**
  - **Feature-1:** 0.9767
  - **Feature-2:** 0.9503
  - **Feature-3:** 0.9493

Conclusion

Here, the HistGradientBoostingClassiefier seemed to perform best among the used models and in all the models using the full processed datset as features produced the best results.