Universal_Bank

August 30, 2020

```
[1]: import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
```

```
[2]: df_bank=pd.read_csv("UniversalBank.csv")
```

```
[3]: df_bank.shape
```

```
[3]: (5000, 14)
```

```
[4]: df_bank.head()
```

```
[4]:    ID  Age  Experience  Income  ZIP Code  Family  CCAvg  Education  Mortgage  \
    0   1   25           1      49     91107       4    1.6          1         0
    1   2   45          19      34     90089       3    1.5          1         0
    2   3   39          15      11     94720       1    1.0          1         0
    3   4   35           9     100     94112       1    2.7          2         0
    4   5   35           8      45     91330       4    1.0          2         0

       Personal Loan  Securities Account  CD Account  Online  CreditCard
    0              0                   1           0       0           0
    1              0                   1           0       0           0
    2              0                   0           0       0           0
    3              0                   0           0       0           0
    4              0                   0           0       0           1
```

```
[5]: df_bank.isnull().sum()
```

```
[5]: ID                  0
     Age                 0
     Experience          0
     Income              0
     ZIP Code            0
     Family              0
     CCAvg               0
     Education           0
     Mortgage            0
```

```
Personal Loan        0
Securities Account   0
CD Account           0
Online               0
CreditCard           0
dtype: int64
```

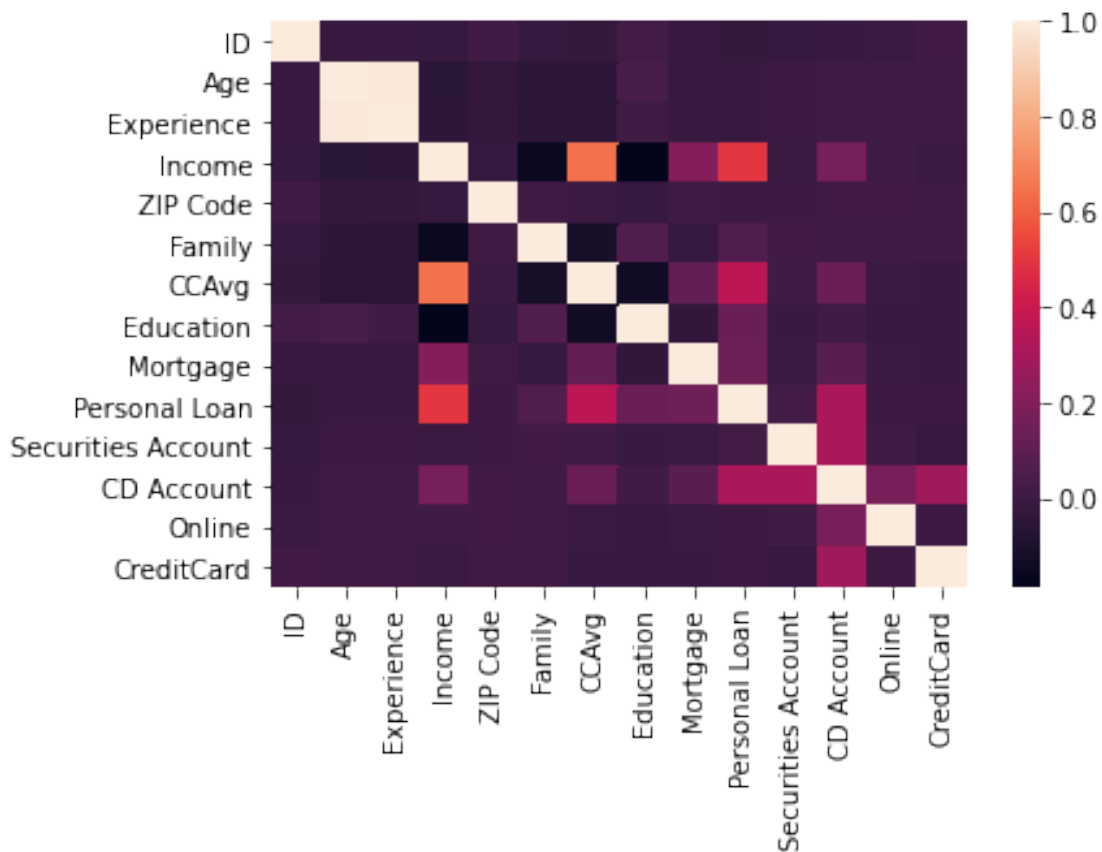[12]: `df_bank['Personal Loan'].value_counts()`

[12]:
```
0    4520
1     480
Name: Personal Loan, dtype: int64
```

[7]: `import seaborn as sns`

[8]: `sns.heatmap(df_bank.corr())`

[8]: `<AxesSubplot:>`

[9]: `df1=df_bank.drop('ZIP Code',axis=True)`

```
[10]: df1.shape
```

```
[10]: (5000, 13)
```

```
[13]: X=df1.drop('Personal Loan',axis=True)
```

```
[14]: y=df1['Personal Loan']
```

```
[15]: from sklearn.model_selection import cross_val_score
```

```
[16]: from sklearn.linear_model import LogisticRegression
```

```
[17]: lrg=LogisticRegression()
```

```
[19]: cross_val_score(lrg,X,y,cv=10,scoring='accuracy').mean()
```

```
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
```

```
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
  extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
```

```
    extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
    extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
    extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
```

[19]: 0.9418000000000001

[21]:
```python
lrg.fit(X,y)
```

```
/usr/local/lib/python3.7/site-packages/sklearn/linear_model/_logistic.py:940:
ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-
regression
    extra_warning_msg=_LOGISTIC_SOLVER_CONVERGENCE_MSG)
```

[21]:
```
LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, l1_ratio=None, max_iter=100,
                   multi_class='auto', n_jobs=None, penalty='l2',
                   random_state=None, solver='lbfgs', tol=0.0001, verbose=0,
                   warm_start=False)
```

[22]:
```python
y_pred=lrg.predict(X)
y_pred
```

[22]: array([0, 0, 0, …, 0, 0, 0])

```python
[23]: from sklearn.metrics import␣
      ↪accuracy_score,classification_report,confusion_matrix
```

```python
[24]: print(accuracy_score(y_pred,y))
```

```
0.9442
```

```python
[25]: print(classification_report(y,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.96      0.98      0.97      4520
           1       0.78      0.58      0.67       480

    accuracy                           0.94      5000
   macro avg       0.87      0.78      0.82      5000
weighted avg       0.94      0.94      0.94      5000
```

```python
[26]: confusion_matrix(y,y_pred)
```

```python
[26]: array([[4441,   79],
             [ 200,  280]])
```

```python
[27]: from sklearn.ensemble import AdaBoostClassifier
```

```python
[28]: boost=AdaBoostClassifier(n_estimators=50,learning_rate=1)
```

```python
[29]: boost.fit(X,y)
```

```python
[29]: AdaBoostClassifier(algorithm='SAMME.R', base_estimator=None, learning_rate=1,
                         n_estimators=50, random_state=None)
```

```python
[32]: y_pred_b=boost.predict(X)
      y_pred_b
```

```python
[32]: array([0, 0, 0, …, 0, 0, 0])
```

```python
[33]: accuracy_score(y_pred_b,y)
```

```python
[33]: 0.9726
```

```python
[35]: confusion_matrix(y,y_pred_b)
```

```python
[35]: array([[4483,   37],
             [ 100,  380]])
```

[ ]: