4/16/2021

ASSIGNMENT-4

Classification Algorithms

Name Rishabh Sharma

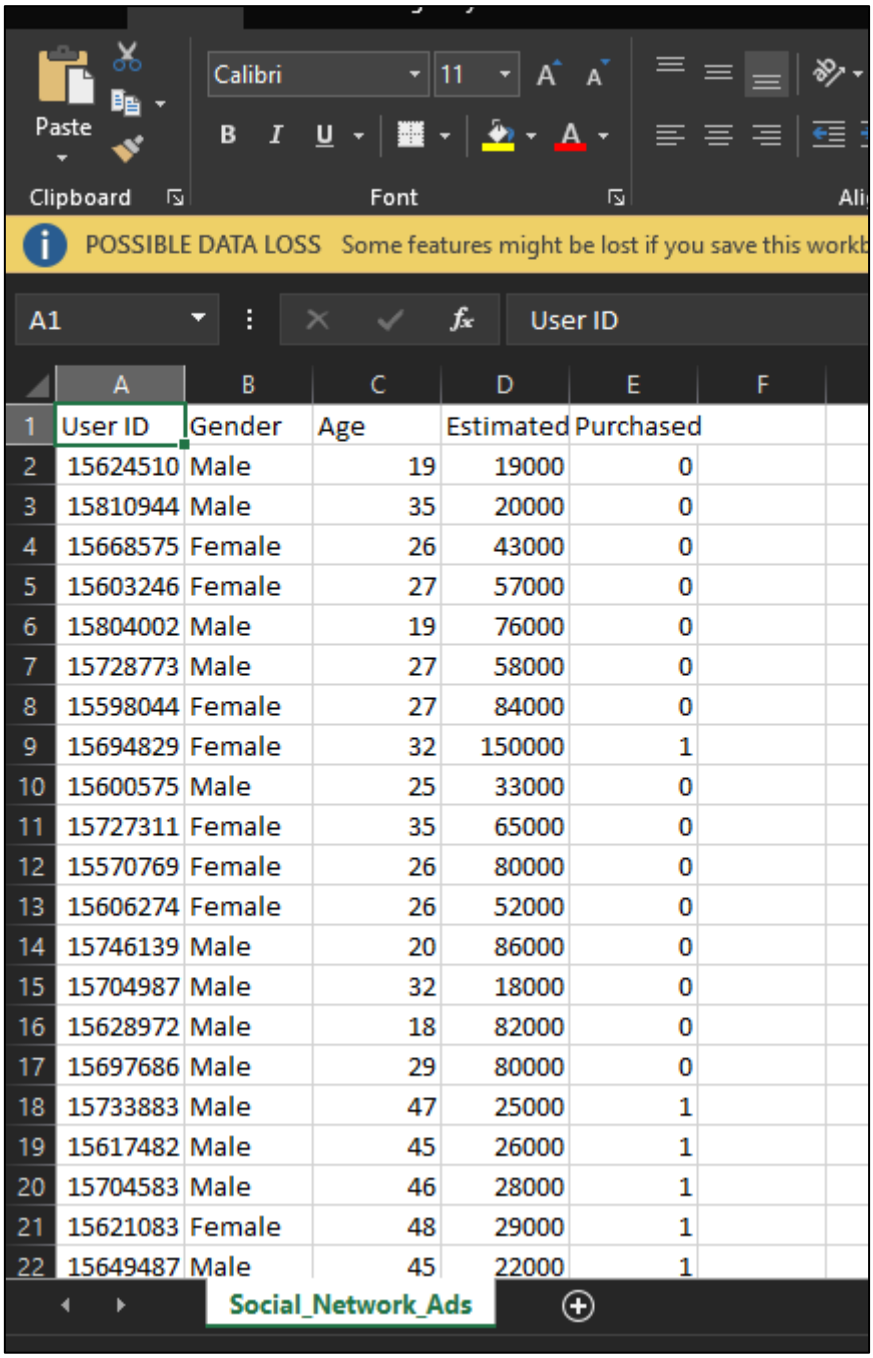
Registration Number 20MAI0082

GitHub Link :- https://github.com/rishabh5197/Data-Mining/tree/main/Assignment-4

Dataset Link :-
https://www.kaggle.com/rakeshrau/social-network-ads

Description About dataset

- There are 400 rows and 5 columns present in our dataset from which user_id is unique and the rest 3 columns are going to be independent data and the last one is going to be the dependent data which is purchased.



	A	B	C	D	E	F
1	User ID	Gender	Age	Estimated	Purchased	
2	15624510	Male	19	19000	0	
3	15810944	Male	35	20000	0	
4	15668575	Female	26	43000	0	
5	15603246	Female	27	57000	0	
6	15804002	Male	19	76000	0	
7	15728773	Male	27	58000	0	
8	15598044	Female	27	84000	0	
9	15694829	Female	32	150000	1	
10	15600575	Male	25	33000	0	
11	15727311	Female	35	65000	0	
12	15570769	Female	26	80000	0	
13	15606274	Female	26	52000	0	
14	15746139	Male	20	86000	0	
15	15704987	Male	32	18000	0	
16	15628972	Male	18	82000	0	
17	15697686	Male	29	80000	0	
18	15733883	Male	47	25000	1	
19	15617482	Male	45	26000	1	
20	15704583	Male	46	28000	1	
21	15621083	Female	48	29000	1	
22	15649487	Male	45	22000	1	

Python Program Implementation

- Importing necessary libraries

```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings("ignore")
```

- Reading dataset and selecting the features that are needed. Selecting Independent variable as x and dependent variable as y and dividing the dataset using train_test_split.

```
read=pd.read_csv("Social_Network_Ads.csv")
```

```
read.head()
```

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0

```
x=read.iloc[:,[2,3]]
y=read.iloc[:,-1]
```

```
from sklearn.model_selection import train_test_split
xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=1/4,random_state=0)
```

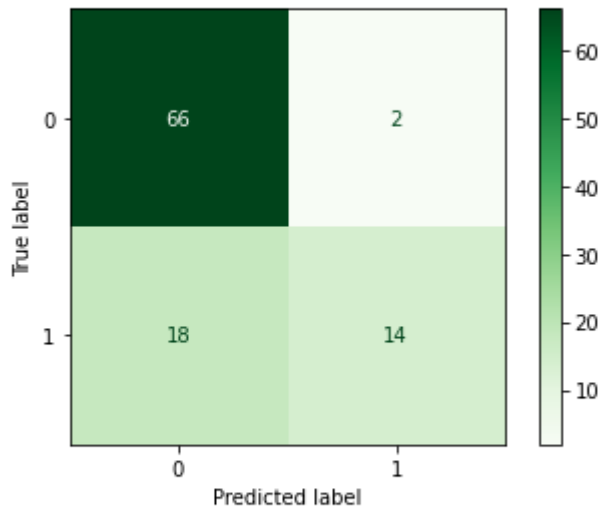
Firstly, implementing SVM

- Using Sklearn package to import SVM

```
from sklearn.svm import SVC
svc = SVC()
svc.fit(xtrain,ytrain)
ypred = svc.predict(xtest)
```

- Getting evaluation of the model using confusion matrix and accuracy score.

```
from sklearn.metrics import plot_confusion_matrix,accuracy_score
plot_confusion_matrix(svc,xtest,ytest,cmap="Greens")
plt.show()
```




```
accuracy_score(ytest,ypred)
```

0.8

Implementing Naïve Bayes

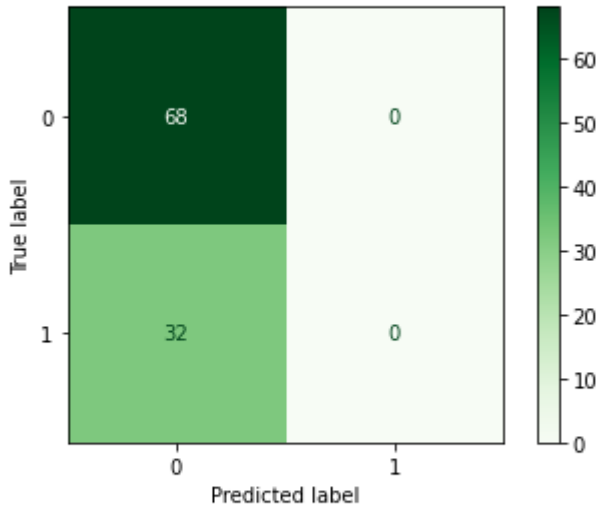
- Using sklearn package to import Naïve Bayes.

```
from sklearn.naive_bayes import BernoulliNB
nb = BernoulliNB()
nb.fit(xtrain,ytrain)
ypred = nb.predict(xtest)
```

```
ypredict = nb.predict(xtest)
```

- Getting evaluation of the model using confusion matrix and accuracy score.

```
from sklearn.metrics import plot_confusion_matrix,accuracy_score
plot_confusion_matrix(nb,xtest,ytest,cmap="Greens",)
plt.show()
```




```
accuracy_score(ytest,ypredict)
```

```
0.68
```