UPES
UNIVERSITY WITH A PURPOSE

Pattern & Anomaly Detection Lab

Experiment: 4

Submitted By:
Dhruv Singhal
500075346
R177219074
AIML B3(SEM 5)

Submitted To:
Dr. Gopal Phartiyal
Professor
SOCS
UPES

AIM:

To perform Logistic Regression

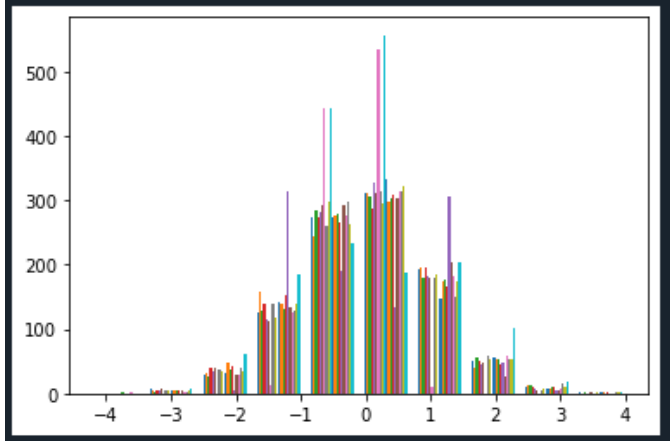
CODE:

```
"""
@author: Dhruv Singhal
"""

#%%
from sklearn.datasets import make_classification
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
import seaborn as sns
import pandas as pd
from sklearn.metrics import confusion_matrix
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
#%%

#generate dataset
X,y = make_classification(n_samples=1000,n_classes=2)
X=pd.DataFrame(X)
y=pd.Series(y,name=20)
print("X values are:",X.head())
print("Y values are:",y.head())

#%% Visualization

plt.hist(X)
plt.show()

#%%
sns.distplot(y)
plt.show()

#%%
sns.distplot(X)
plt.show()

#%% feature extractionpreprocessing, correlation matrix
_,graph=plt.subplots(figsize=(15,10))
sns.heatmap(X.corr(),annot=True,ax=graph,square=True)
plt.show()

#%%
df=pd.merge(X,y,right_index=True,left_index=True)
print(df.head())
print(df.corr()[[20]].abs().sort_values(by=20,ascending=False))
```

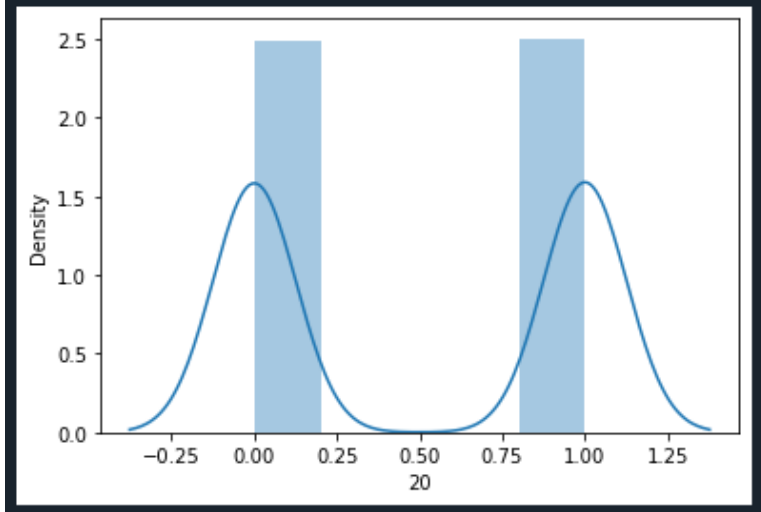
```
#%% columns with high correlation will be dropped
# creating useful data
datasetX=df[[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19]]
datasetY=df[20]
print(datasetY.head())

#%%
X_train,X_test,Y_train,Y_test=train_test_split(X,y,test_size=0.15,random_state=42)

model=LogisticRegression()
model.fit(X_train,Y_train)
print(model.classes_)

Y_pred=model.predict(X_test)
print(Y_pred)

print("train Accuracy:",model.score(X_train,Y_train))
print("test Accuracy:",model.score(X_test,Y_test))

print(confusion_matrix(Y_test,Y_pred))
```

OUTPUT:

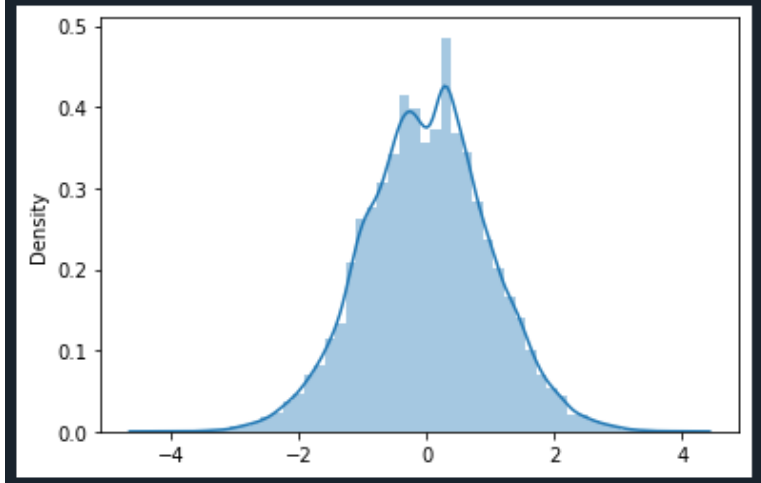
```
Python 3.7.6 (default, Jan  8 2020, 20:23:39) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 7.26.0 -- An enhanced Interactive Python.

In [1]: runcell(0, 'C:/Users/Dhruv Singhal/untitled0.py')

In [2]: runcell(1, 'C:/Users/Dhruv Singhal/untitled0.py')

In [3]: runcell(2, 'C:/Users/Dhruv Singhal/untitled0.py')
X values are:          0         1         2   ...        17        18        19
0 -1.259436 -0.372492  0.054760  ... -0.035846 -1.462309 -1.140185
1 -0.800033  0.478840 -0.629627  ...  0.170002  0.445090 -1.299911
2 -0.770212 -0.741631  1.075089  ...  0.075282 -1.240537 -0.914384
3  1.583047 -0.705226  0.710976  ... -1.260828  0.947186 -0.973737
4  0.431166  0.902277  2.205464  ...  0.333836  0.417141  1.348560

[5 rows x 20 columns]
Y values are: 0    0
1    0
2    1
3    0
4    1
Name: 20, dtype: int32

In [4]: runcell('Visualization', 'C:/Users/Dhruv Singhal/untitled0.py')
```



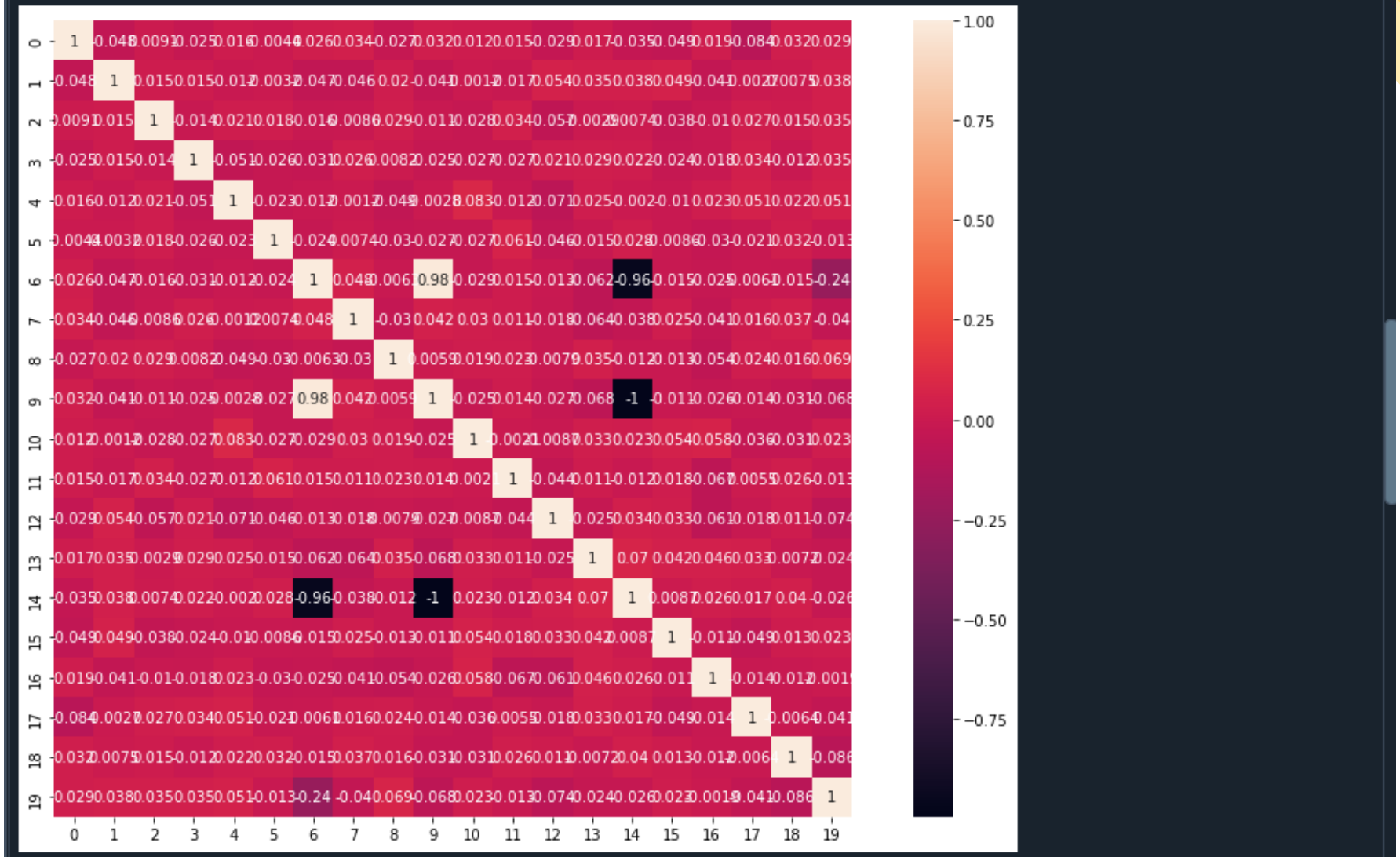
```
In [5]: runcell(4, 'C:/Users/Dhruv Singhal/untitled0.py')
```




```
In [6]: runcell(5, 'C:/Users/Dhruv Singhal/untitled0.py')
```



```
In [7]: runcell('feature extractionpreprocessing, correlation matrix', 'C:/Users/Dhruv Singhal/untitled0.py')
```



```
In [8]: runcell(7, 'C:/Users/Dhruv Singhal/untitled0.py')
          0         1         2         3   ...        17        18        19  20
0 -1.259436 -0.372492  0.054760 -2.045250  ... -0.035846 -1.462309 -1.140185   0
1 -0.800033  0.478840 -0.629627 -0.773224  ...  0.170002  0.445090 -1.299911   0
2 -0.770212 -0.741631  1.075089 -1.506821  ...  0.075282 -1.240537 -0.914384   1
3  1.583047 -0.705226  0.710976 -1.241509  ... -1.260828  0.947186 -0.973737   0
4  0.431166  0.902277  2.205464 -0.400889  ...  0.333836  0.417141  1.348560   1

[5 rows x 21 columns]
          20
20  1.000000
14  0.907235
9   0.906191
6   0.882873
13  0.067238
12  0.050728
1   0.046343
15  0.036825
3   0.032795
7   0.029817
0   0.029335
11  0.028842
16  0.026296
18  0.018547
5   0.018465
4   0.015409
10  0.011313
19  0.007832
17  0.005624
8   0.001292
2   0.000811

In [9]: runcell('columns with high correlation will be dropped', 'C:/Users/Dhruv Singhal/untitled0.py')
0    0
1    0
2    1
3    0
4    1
Name: 20, dtype: int32
```

```
In [10]: runcell(9, 'C:/Users/Dhruv Singhal/untitled0.py')
[0 1]
[1 0 1 0 0 1 0 0 0 1 0 0 1 0 1 0 0 0 1 0 0 0 0 1 0 1 1 1 0 0 0 0 1 1 1 1 0
 0 1 1 0 1 1 0 1 0 0 1 1 0 0 1 1 1 1 0 0 0 0 0 1 1 1 1 1 0 1 0 0 0 1 0 0 0
 0 0 1 0 1 0 0 0 0 1 1 0 0 0 1 0 1 1 1 0 1 1 1 0 0 0 0 1 1 1 0 0 0 1 0 1 0
 0 0 0 0 1 1 0 0 1 0 0 0 1 0 1 1 0 1 1 0 1 0 1 0 1 0 0 0 0 1 0 1 0 0 0 1 1
 0 0]
train Accuracy: 0.9776470588235294
test Accuracy: 0.9733333333333334
[[82  0]
 [ 4 64]]

In [11]:
```