

```

import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split

from sklearn.neural_network import MLPClassifier
from sklearn.metrics import ConfusionMatrixDisplay, confusion_matrix
from sklearn.metrics import accuracy_score, classification_report

df = pd.read_csv("D:/Training datasets/Churn_Modelling.csv")
df

df.columns

df['Geography'].nunique()

df = pd.concat([df, pd.get_dummies(df['Gender'])], axis = 1)
df

df.drop(columns = ['RowNumber', 'CustomerId'])

#df = df.drop(columns = ['Gender'])
df

def num(column):
    for i in range(len(column)):
        if column[i]==True:
            column[i]=1
        else:
            column[i]=0

num(df['Geo_France'])
num(df['Geo_Germany'])
num(df['Geo_Spain'])
num(df['Female'])
num(df['Male'])

df

df.columns

x = df[['CreditScore', 'Age', 'Tenure', 'Balance', 'NumOfProducts',
'HasCrCard', 'IsActiveMember',

```

```
'EstimatedSalary','Geo_France','Geo_Germany', 'Geo_Spain', 'Female',  
'Male']]
```

```
y = df['Exited']
```

```
sns.countplot(x=y)
```

```
sc = StandardScaler()
```

```
sc_x = sc.fit_transform(x)
```

```
x_train, x_test, y_train, y_test = train_test_split(sc_x, y,  
test_size=0.2)
```

```
mlp = MLPClassifier()
```

```
# In[24]:
```

```
mlp.fit(x_train, y_train)
```

```
predict = mlp.predict(x_test)
```

```
print(accuracy_score(predict,y_test))
```

```
ConfusionMatrixDisplay.from_predictions(predict,y_test)
```

```
cm = confusion_matrix(predict,y_test)  
cm
```

```
TP = cm[0,[0]]  
TN = cm[1,[1]]  
FP = cm[0,[1]]  
FN = cm[1,[0]]
```

```
TP
```

```
acc = (TP+TN) / (TP+TN+FP+FN)
acc
```

```
# In[32]:
```

```
cr = classification_report(predict,y_test)
cr
```

```
# In[16]:
```

```
import requests
print(requests.get("https://raw.githubusercontent.com/Sahil030303/LP-3/main/ML/ML5.py").text)
```

```
# In[ ]:
```