

Practical 1

Set A

Q.3. Create 'User' Data set having 5 columns namely: User ID, Gender, Age, EstimatedSalary and Purchased. Build a logistic regression model that can predict whether on the given parameter a person will buy a car or not.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import random
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn import metrics
import seaborn as sn
import matplotlib.pyplot as plt

ID=random.sample(range(0,100),100)

Gender = [];
gender = ["Female","Male"];

for i in range(100):
    Gender.append(random.choice(gender));
print("-----Random Generated Data of Gender-----");

Age=random.sample(range(0,100),100)

Esalary=random.sample(range(10000,70000),100)

Purchased=[];
purchase=[0,1];
for i in range(100) :
    Purchased.append(random.choice(purchase));

data=list(zip(ID,Gender,Age,Esalary,Purchased));
df = pd.DataFrame(data,columns=("ID","Gender","Age","EstimatedSalary","Purchased"));
print(df);

X = df.iloc[:,[2,3]].values;
```

```
Y = df.iloc[:,4].values;
X_train,X_test,Y_train,Y_test = train_test_split(X,Y,train_size=0.5,random
_state=100);

lobj = LogisticRegression();
lobj.fit(X_train,Y_train);
Prediction = lobj.predict(X_test);
print(Prediction);
print(X_test);
print(Prediction);
Accuracy = metrics.accuracy_score(Y_test,Prediction)*100;
print("Accuracy of Logisctics Regression is : ",Accuracy);

confusion_matrix = pd.crosstab(Y_test, Prediction, rownames=['Actual'], co
lnames=['Predicted'])
sn.heatmap(confusion_matrix, annot=True)
plt.show()
```

Output :-

[illegible]

