DSBDAL

Assignment No:-05

Title:- Data Analytics II

objective:- students should be able to data analysis using logstics regression using python for any open source dataset.

Aim:- Data Analytics II
1) Implement logistic regression using python to perform classification on social-network Ads. csv dataset.
2) compute confusion matrix to find TP, FP, TN, FN, Accuracy, Error rate, precision.

Requirements:- 1) Basic of python programming.
2) concept of Regression.

Theory:-
logsitic Regression:- social network Ads.
This project will be a walk through of a simple logstic Regression model in an attempt to strategies a basic ad-targetting campigon for a social media network.
one of four sponsors advertisments seems to be particularly successful among

our older Neither users but seemingly less- so with your younger.
- We like to implement an appropriate so that we know who our target audience is this specific advertisment, thus the maximizes click-through rate.
- our dataset contains some information about all of our Users in social network.

- If We Wanted to determine the effect more independent variables on the outcome We Would have to implement a dimensions reduction aspect to the model because only describe so many dimensions visually.

- Worried about how the user's Age & estimate salary effect their - decision on click or not clients on adv.
extracting relevant vectors Independent variables (x), dependent variables (y)
- Split data into two set:- train set to learn m/c from & test set for m/c to execute on.
- This process is refeered to a across validation & We will be the implemented Scikit learn's appropriately.
Industry standard usually calls for a training set size of 70-80%.
So Well split the two.

When We look at both the model together. We can actually see that there is a shape to this data that's becoming the increasingly apparent as the no. of observation increases.
- The best x-intercept, is probably, closer to the than is to be 2, and the y-intercept is likely between 2 & 3.
- This function will compair the calculate result in our y-pred vector to the actu observation results in y-test to determine how similar they are.
- The more values that match, the higher the accuracy of the classifier.

* Conclusion:-
The confusion matrix tells us that there Were correct predictions and 11 incorre ones, meaning the model overall accomplis an 89% accuracy rating.

* csv file / dataset - Social_network_Ads.csv.
Required libraries.
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import Seaborn as sns.

from sklearn.preprocessing import standardscaler.
from sklearn.model_selection import train_test_split.
from sklearn.metrics import confusion_matrix. classification report, accuracy score, precision, score, recall_score, fc-score.
import warnings.

* function used -

```
df = pd.read_csv("Social_NetWork_Ads.csv")
df.head()
df.shape
df.info()
df.describe()
df.isnull().sum()
histplot = sns.histplot()
plt.show()
Draw histogram for each column
df["column name"].value_counts()
countplot = sns.countplot()
sns.heatmap()
```

- Data preparation
- model building
- Evaluation.

Q.1) Explain confusion matrix with Accuracy error rate, precision & Recall.

→ -It contain Actual value & predicted value.

- Terms included in confusion matrix are -
i) True Negative (TN)
ii) True positive
iii) false positive
iv) false Negative

e.g. patient have disease with sample 165 confusion matrix predicted

| | 165 | No | Yes | |
|---|---|---|---|---|
| Actual | No | 50 [TN] | 10 [FP] | 60 |
| | Yes | 5 [FN] | 100 [TP] | 105 |
| | | 55 | 110 | |

Actual value - which are already true, its reality.
predicted values - After some experiment

$$\text{Accuracy} = \frac{TP+TN}{\text{Total}} = \frac{100+50}{165} = 0.91$$

$$\text{Error rate} = 1 - \text{Accuracy} \text{ or } \frac{FP+FN}{\text{Total}} = 0.09$$

$$\text{precision} = \frac{TP}{\text{predicted Yes}} = \frac{100}{110} = 0.64$$

$$\text{Recall} = \frac{TP}{\text{actual Yes}} = \frac{100}{105} = 0.95$$