```python
# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv(r'C:\Users\Hanshu\Downloads\logit classification.csv')

x = dataset.iloc[: , [2,3]].values
y = dataset.iloc[: , -1].values


# split the data into x-train.....
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.20, random_state=0)


# feature scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
#we mentioned feature scaling only to independent variable not dependent variable at all

#datapreprocessing done


#now going to build the KNN model
from sklearn.neighbors import KNeighborsClassifier
classifier = KNeighborsClassifier()
classifier.fit(x_train, y_train)
#we have to fit the logistic regression model to our training set

# Predicting the Test set results
y_pred = classifier.predict(x_test)
#now you compare X_test with y_pred, x-test we have age and salary ,
#if u look at the first observation this user is not be able to buy the car but if you look at c
#in this case logistic regression model classify the which users are going to buy the car or not

#we build our logistic model and fit it to the training set & we predict our test set result


#now we will use the confusion matrix to evalute


# making the confusion matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print(cm)
#we can say that 65 + 24 = 89 correct prediction we found & 8+3 = 11 incorrect prediction made


#this is to get model accuracy
from sklearn.metrics import accuracy_score
ac = accuracy_score(y_test, y_pred)
print(ac)

# This is to get the Classification Report
# classifier report
from sklearn.metrics import classification_report
cr  = classification_report(y_test, y_pred)
```

```python
print(cr)

   # trainig score
bias = classifier.score(x_train, y_train)
print(bias)

# testing scire
variance = classifier.score(x_test, y_test)
print(variance)


# 4-09-2025

dataset1 = pd.read_csv(r"C:\Users\Hanshu\Downloads\2.LOGISTIC REGRESSION CODE\2.LOGISTIC REGRESS

d2 = dataset1.copy()

dataset1 = dataset1.iloc[:, [3,4]].values

from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
M = sc.fit_transform(dataset1)

y_pred1 = pd.DataFrame()


d2['y_pred1'] = classifier.predict(M)

d2.to_csv('final1.csv')
```