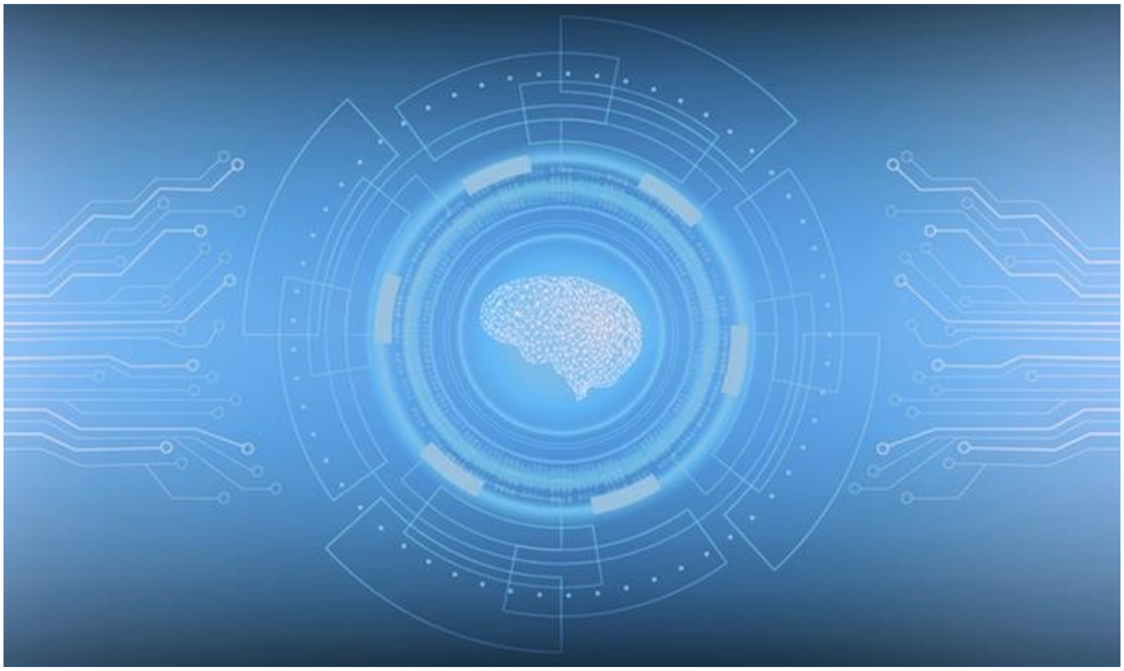
User-purchasing predicting model based on user data



By: Eshaan Mathakari

Tanya Evita George

User-Purchasing predicting model based on userdata set

Introduction:

The idea of e-commerce assists managers or those in positions who are responsible to make decisions for the progress of their companies. Undoubtedly,most of these decisions are influenced by the results derived from studying the data based on the purchasing behavior of their target audiences or the customer's data of online customers by experts in data analysis and machine learning.

The dataset we will use contains information of users from a company's database. It contains information about UserID, Gender, Age, EstimatedSalary, Purchased. We are using this dataset for predicting whether a user will purchasethe company's newly launched product or not.

Finding out the customer's needs and wants are extremely important. With the data set provided, we analyze the customer history, as well as if a person is our potential customer or not.

Using a logistic regression model helps us plot our data out and hence thedensely populated region can be noted down and we can use this for our prediction.

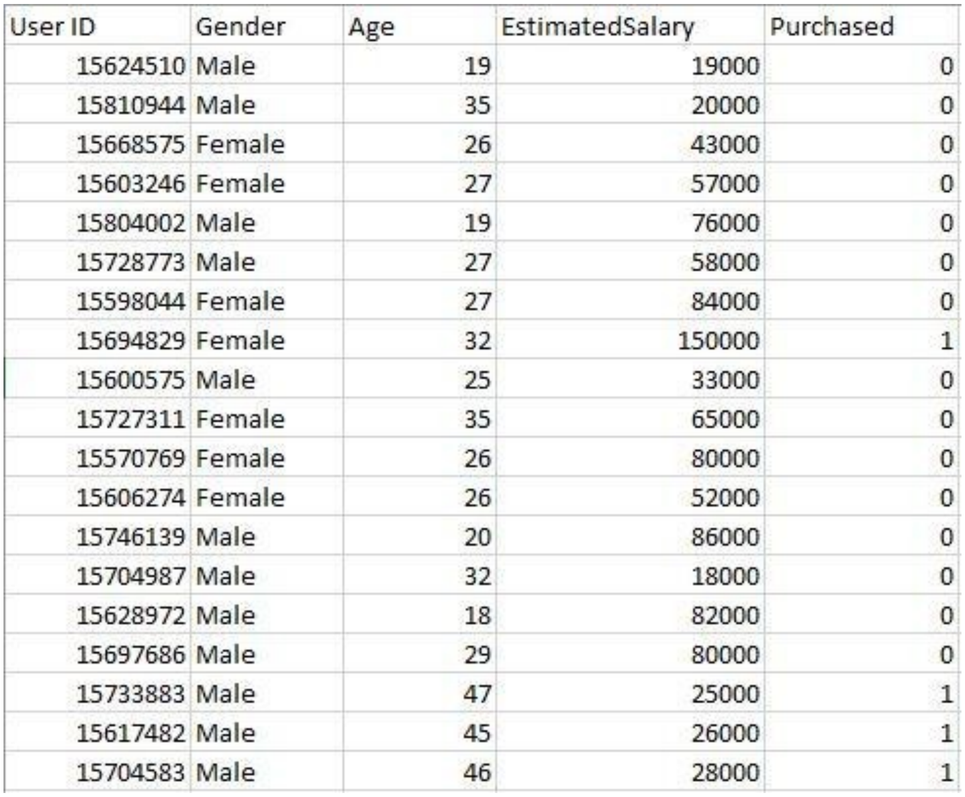
Data Description:

The aim of putting together this dataset is to predict or understand at least if the customers will buy or purchase the new product the company will release in a few days. Surveys are conducted by asking the target audience their feedback about previous products they have purchased and used and their thoughts on the quality of the company's brand. The customer's answers to their inquiry will help us identify which customers the marketing team need to have a focus on with regard to the next promotional offers they will be putting together for the launch of new products.

Data Preparation:

It is important to keep attributes of the data set ready before hand so that it is easy to conduct surveys, take feedbacks from customers and arrange the collected information in a proper order.

We use the given below data set to compute prediction and to make our regression model.



User ID	Gender	Age	EstimatedSalary	Purchased
15624510	Male	19	19000	0
15810944	Male	35	20000	0
15668575	Female	26	43000	0
15603246	Female	27	57000	0
15804002	Male	19	76000	0
15728773	Male	27	58000	0
15598044	Female	27	84000	0
15694829	Female	32	150000	1
15600575	Male	25	33000	0
15727311	Female	35	65000	0
15570769	Female	26	80000	0
15606274	Female	26	52000	0
15746139	Male	20	86000	0
15704987	Male	32	18000	0
15628972	Male	18	82000	0
15697686	Male	29	80000	0
15733883	Male	47	25000	1
15617482	Male	45	26000	1
15704583	Male	46	28000	1

Attributes used-

- User ID
- Gender of the customer
- Age of the customer
- Estimated Salary of the customer
- Purchased Products by the customer

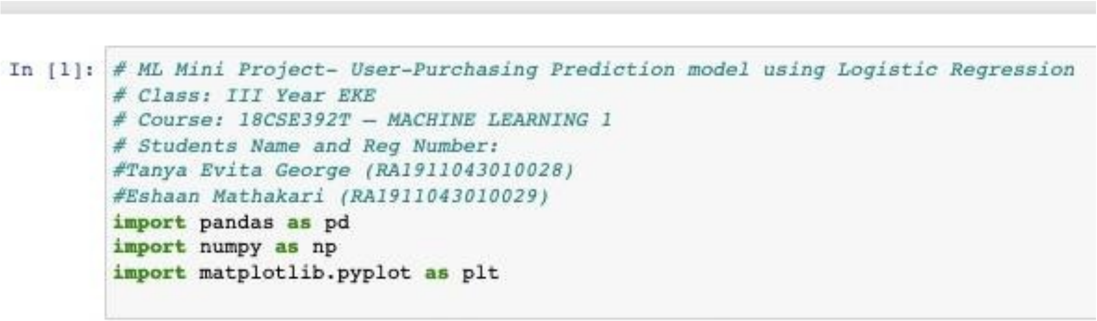
Objective:

Using the given dataset, a machine learning model using logistic regression is made that predicts whether an online customer of a company will make their nextpurchase of the newly launched product in the market of that same company's from the day they made their last purchase.

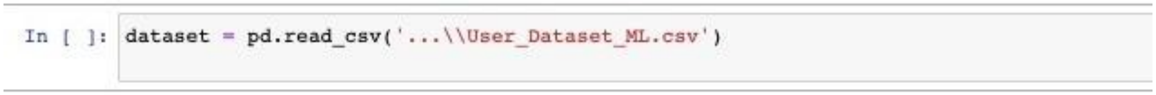
Coding of the prediction model:

Visualization of data is an imperative aspect of data science. It helps to understand data and also to explain the data to another person. Python has several interesting visualization libraries such as Matplotlib, Seaborn etc.

Loading dataset – User_Dataset_ML

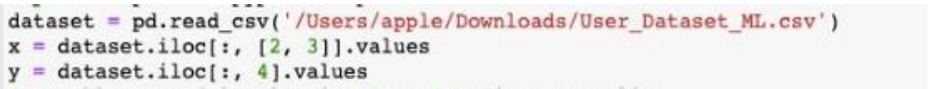


```
In [1]: # ML Mini Project- User-Purchasing Prediction model using Logistic Regression
        # Class: III Year EKE
        # Course: 18CSE392T – MACHINE LEARNING 1
        # Students Name and Reg Number:
        #Tanya Evita George (RA1911043010028)
        #Eshaan Mathakari (RA1911043010029)
        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
```



```
In [ ]: dataset = pd.read_csv('...\\User_Dataset_ML.csv')
```

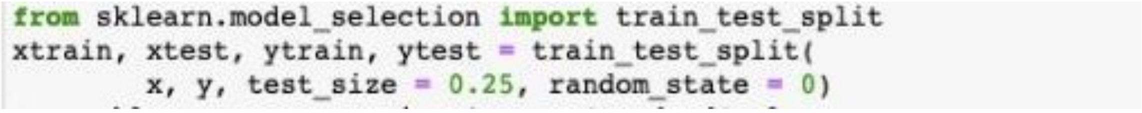
Now, to predict whether a user will purchase the product or not, one needs to find out the relationship between Age and Estimated Salary. Here User ID and Gender are not important factors for finding out this.



```
dataset = pd.read_csv('/Users/apple/Downloads/User_Dataset_ML.csv')
x = dataset.iloc[:, [2, 3]].values
y = dataset.iloc[:, 4].values
```

Splitting the dataset:

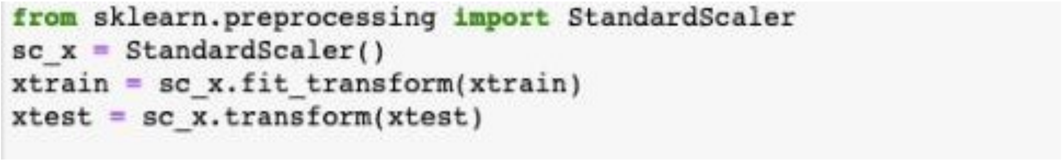
Splitting the dataset to train and test. 75% of data is used for training the model and 25% of it is used to test the performance of our model.



```
from sklearn.model_selection import train_test_split
xtrain, xtest, ytrain, ytest = train_test_split(
        x, y, test_size = 0.25, random_state = 0)
```

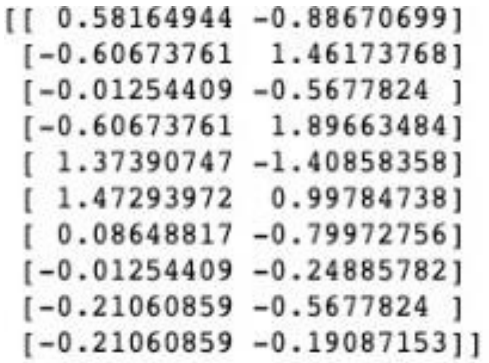
Feature Scaling:

Feature scaling is important here because Age and Estimated Salary values lie in different ranges. If we don’t scale the features then Estimated Salary feature will dominate Age feature when the model finds the nearest neighbor to a data point in data space.



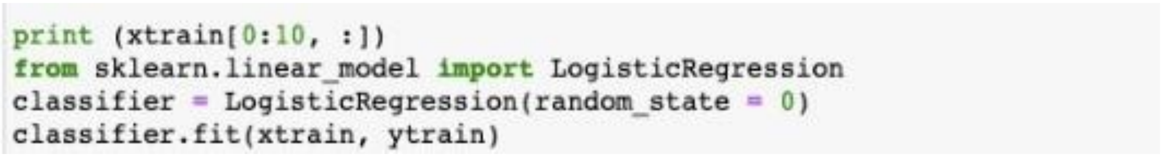
```
from sklearn.preprocessing import StandardScaler
sc_x = StandardScaler()
xtrain = sc_x.fit_transform(xtrain)
xtest = sc_x.transform(xtest)
```

Output:



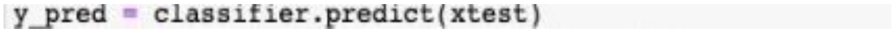
```
[[ 0.58164944 -0.88670699]
 [-0.60673761  1.46173768]
 [-0.01254409 -0.5677824 ]
 [-0.60673761  1.89663484]
 [ 1.37390747 -1.40858358]
 [ 1.47293972  0.99784738]
 [ 0.08648817 -0.79972756]
 [-0.01254409 -0.24885782]
 [-0.21060859 -0.5677824 ]
 [-0.21060859 -0.19087153]]
```

Here once see that Age and Estimated salary features values are scaled and now there in the -1 to 1. Hence, each feature will contribute equally in decision making i.e. finalizing the hypothesis.
Finally, we are training our Logistic Regression model.



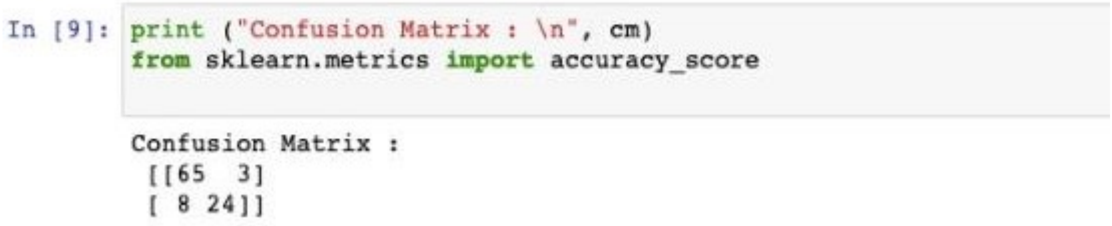
```
print (xtrain[0:10, :])
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression(random_state = 0)
classifier.fit(xtrain, ytrain)
```

After training the model, it time to use it to do prediction on testing data.



```
y_pred = classifier.predict(xtest)
```

Now, we use a confusion matrix to test the performance of our Logistic Regression model.



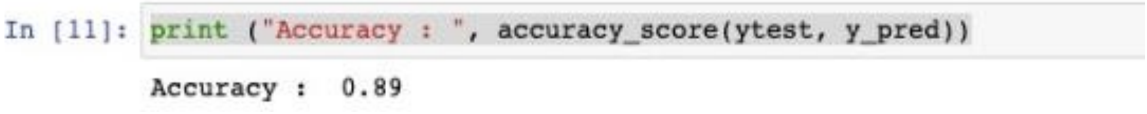
```
In [9]: print ("Confusion Matrix : \n", cm)
        from sklearn.metrics import accuracy_score

        Confusion Matrix :
         [[65  3]
         [ 8 24]]
```

From the Output:

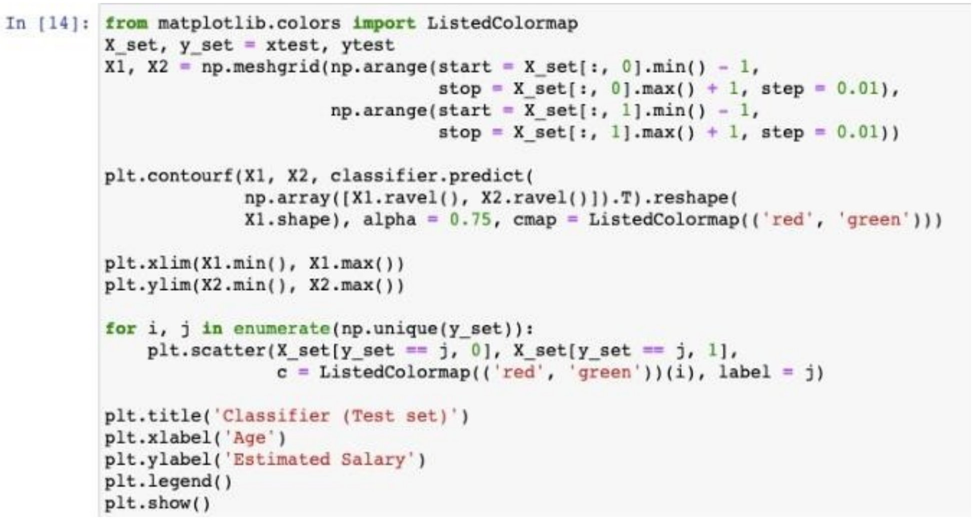
Out of 100 :
TruePostive + TrueNegative = 65 + 24
FalsePositive + FalseNegative = 3 + 8
Performance measure – Accuracy

Model accuracy:

Visualizing the performance of our model.



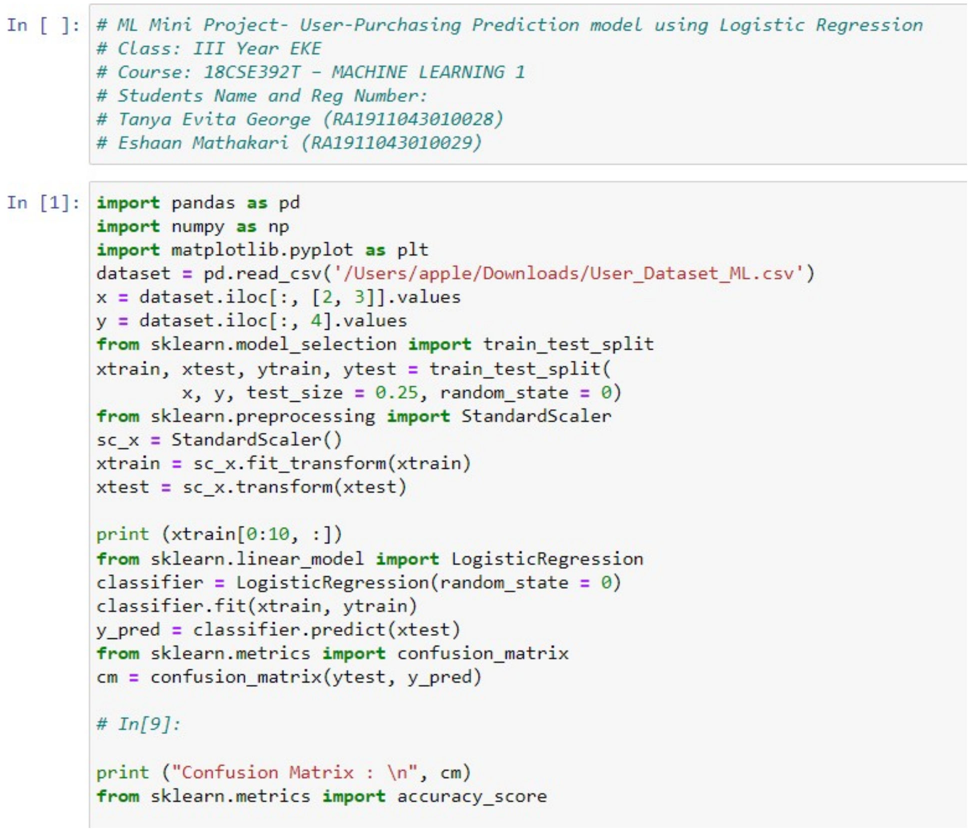
```
In [11]: print ("Accuracy : ", accuracy_score(ytest, y_pred))

         Accuracy :  0.89
```



```
In [14]: from matplotlib.colors import ListedColormap
         X_set, y_set = xtest, ytest
         X1, X2 = np.meshgrid(np.arange(start = X_set[:, 0].min() - 1,
                                        stop = X_set[:, 0].max() + 1, step = 0.01),
                              np.arange(start = X_set[:, 1].min() - 1,
                                        stop = X_set[:, 1].max() + 1, step = 0.01))

         plt.contourf(X1, X2, classifier.predict(
                      np.array([X1.ravel(), X2.ravel()]).T).reshape(
                      X1.shape), alpha = 0.75, cmap = ListedColormap(('red', 'green')))

         plt.xlim(X1.min(), X1.max())
         plt.ylim(X2.min(), X2.max())

         for i, j in enumerate(np.unique(y_set)):
             plt.scatter(X_set[y_set == j, 0], X_set[y_set == j, 1],
                         c = ListedColormap(('red', 'green'))(i), label = j)

         plt.title('Classifier (Test set)')
         plt.xlabel('Age')
         plt.ylabel('Estimated Salary')
         plt.legend()
         plt.show()
```

After analyzing the performance measures – accuracy and confusion matrix and the graph, we can clearly say that our model is performing really well.

Code:

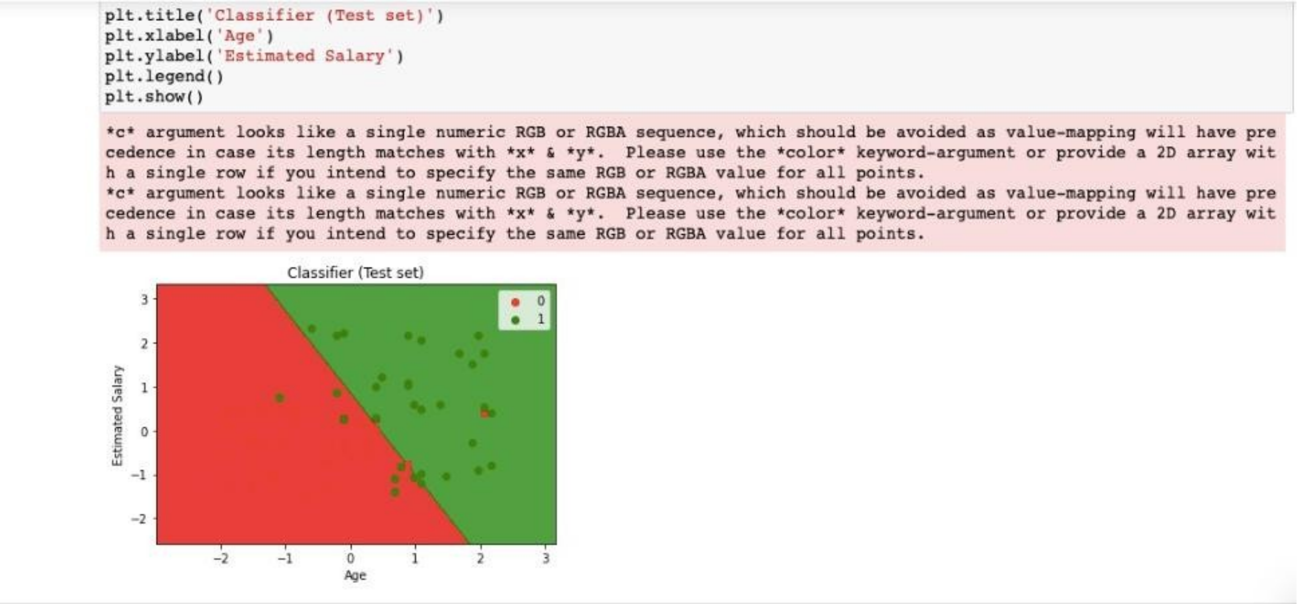


```
In [ ]: # ML Mini Project- User-Purchasing Prediction model using Logistic Regression
        # Class: III Year EKE
        # Course: 18CSE392T - MACHINE LEARNING 1
        # Students Name and Reg Number:
        # Tanya Evita George (RA1911043010028)
        # Eshaan Mathakari (RA1911043010029)
```

```
In [1]: import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
        dataset = pd.read_csv('/Users/apple/Downloads/User_Dataset_ML.csv')
        x = dataset.iloc[:, [2, 3]].values
        y = dataset.iloc[:, 4].values
        from sklearn.model_selection import train_test_split
        xtrain, xtest, ytrain, ytest = train_test_split(
                x, y, test_size = 0.25, random_state = 0)
        from sklearn.preprocessing import StandardScaler
        sc_x = StandardScaler()
        xtrain = sc_x.fit_transform(xtrain)
        xtest = sc_x.transform(xtest)

        print (xtrain[0:10, :])
        from sklearn.linear_model import LogisticRegression
        classifier = LogisticRegression(random_state = 0)
        classifier.fit(xtrain, ytrain)
        y_pred = classifier.predict(xtest)
        from sklearn.metrics import confusion_matrix
        cm = confusion_matrix(ytest, y_pred)

        # In[9]:

        print ("Confusion Matrix : \n", cm)
        from sklearn.metrics import accuracy_score
```

```
# In[11]:

print ("Accuracy : ", accuracy_score(ytest, y_pred))


# In[14]:

from matplotlib.colors import ListedColormap
X_set, y_set = xtest, ytest
X1, X2 = np.meshgrid(np.arange(start = X_set[:, 0].min() - 1,
                               stop = X_set[:, 0].max() + 1, step = 0.01),
                     np.arange(start = X_set[:, 1].min() - 1,
                               stop = X_set[:, 1].max() + 1, step = 0.01))

plt.contourf(X1, X2, classifier.predict(
             np.array([X1.ravel(), X2.ravel()]).T).reshape(
             X1.shape), alpha = 0.75, cmap = ListedColormap(('red', 'green')))

plt.xlim(X1.min(), X1.max())
plt.ylim(X2.min(), X2.max())

for i, j in enumerate(np.unique(y_set)):
    plt.scatter(X_set[y_set == j, 0], X_set[y_set == j, 1],
                c = ListedColormap(('red', 'green'))(i), label = j)

plt.title('Classifier (Test set)')
plt.xlabel('Age')
plt.ylabel('Estimated Salary')
plt.legend()
plt.show()
```

Output:



```
plt.title('Classifier (Test set)')
plt.xlabel('Age')
plt.ylabel('Estimated Salary')
plt.legend()
plt.show()
```

```
*c* argument looks like a single numeric RGB or RGBA sequence, which should be avoided as value-mapping will have pre
cedence in case its length matches with *x* & *y*.  Please use the *color* keyword-argument or provide a 2D array wit
h a single row if you intend to specify the same RGB or RGBA value for all points.
*c* argument looks like a single numeric RGB or RGBA sequence, which should be avoided as value-mapping will have pre
cedence in case its length matches with *x* & *y*.  Please use the *color* keyword-argument or provide a 2D array wit
h a single row if you intend to specify the same RGB or RGBA value for all points.
```

Result and Conclusion:

After building our Logistic Regression model and we can see that the prediction model gives the best results for our dataset with an accuracy of 89%. This helps us predict how many of the customers will positively buy the new product or not. The accuracy of prediction can however vary if thepreprocessing of the data is done differently. To increase the accuracy different sampling techniques can be implemented.

And hence, a logistic regression model was made based on the data set obtained based on whether the target customers of the company will purchase the company's products.