EX: 13
Data:

Decision tree Classification

Aim:

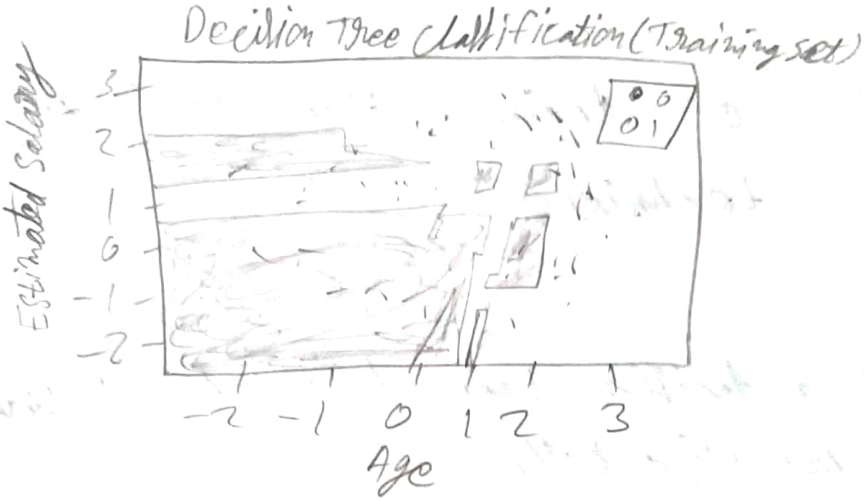
To classify the social Network dataset Using Decision tree analysis

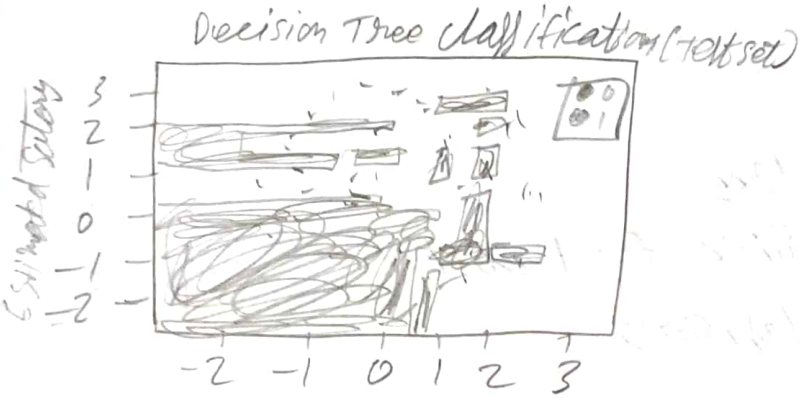
Source Code:

```
from google.colab import drive
drive.mount("/content/gdrive")

import Pandas as pd
import numpy as np
import matplotlib.Pyplot as Plt
dataset = pd.read_CSV('/content/gdrive/My Drive/Social_Network_Ads.csv')

X = dataset.iloc[:,[2,3]].values
Y = dataset.iloc[:, -1].values
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, y, test_size
= 0.25, random_state = 0)

from sklearn.preprocessing import StandardScaler
SC = StandardScaler()
X_train = SC.fit_transform(X_train)
X_test = SC.transform(X_test)
```

```
from sklearn.tree import DecisionTreeClassifier
classifier = DecisionTreeClassifier(criterion = 'entropy', random_state = 0)
classifier.fit(X_train, y_train)
y_pred = classifier.predict(X_test)

from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, y_pred)
print(cm)
from matplotlib.colors import ListedColormap
X_set, y_set = X_train, y_train

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), al
pha = 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('Decision Tree Classification (Training set)')
    plt.xlabel('Age')
    plt.ylabel('Purchase')
    plt.legend()
plt.show()
```

Output:





Result:

Thus Decision tree classification is successfully executed & output verified.