

Exp No: 12

Sub:

Decision Tree

Classification.

Aim:

To classify the social network dataset

Program:

```
from google.colab import drive
drive.mount('') # mount (gdrive)
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

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

```
x_train, x_test, y_train, y_test = train_test_split(
    x, y, test_size=0.2)
```

```
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()
```

for sh-kun miter opant efuim

an-ayam- miter (y-lyt-8-mit)  
pvt (am)

listhys = x-set [: 1] .max (1) + 1, shp. ay  
reshys (2, max (1), x2. min (1))

Listed color map (x1. min (1), x1. max (1))  
for i, j in enumerate (y. max (1) - 1)  
plt.scatter (x-set [y-set == j, 0],  
x-set [y-set == j, 1], c=  
Listed color map ("red", "green"))  
(5) .set (j).

plt.xlabel ('Age')  
plt.ylabel ('predicted')  
plt.legend ()  
plt.show ()

Labels type (Categorical, numerical, etc.)

6. Listed color map (x1. min (1), x1. max (1))  
(Listed color map) (x1. min (1), x1. max (1))  
(Listed color map) (x1. min (1), x1. max (1))

plt.xlabel ('Age')  
plt.ylabel ('predicted')  
plt.legend ()  
plt.show ()



*[Faint, mostly illegible handwritten notes at the top of the page, possibly related to a project or assignment.]*

$(10^8 \times 10^7) / (10^8 \times 10^7) = x$   
 $(10^8 \times 10^7) / (10^8 \times 10^7)$   
 $(10^8 \times 10^7) / (10^8 \times 10^7)$   
 $(10^8 \times 10^7) / (10^8 \times 10^7)$   
 $(10^8 \times 10^7) / (10^8 \times 10^7)$   
 $(10^8 \times 10^7) / (10^8 \times 10^7)$

Result: *[Faint handwritten text]*  
 The page is successfully created.

(10^8 x 10^7) / (10^8 x 10^7) = x  
 (10^8 x 10^7) / (10^8 x 10^7)  
 (10^8 x 10^7) / (10^8 x 10^7)  
 (10^8 x 10^7) / (10^8 x 10^7)  
 (10^8 x 10^7) / (10^8 x 10^7)