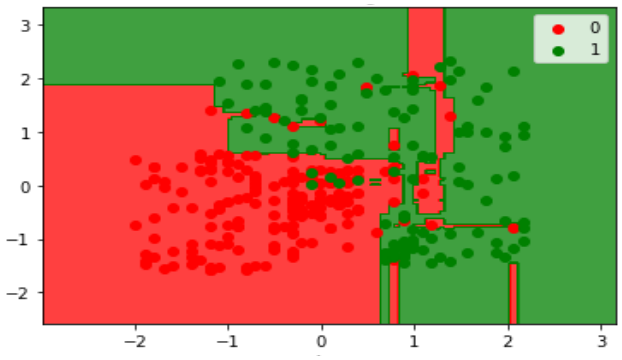
Curneu MedTech Innovation Private Limited

Name	**:**	**Hari haran B**
College register no	**:**	**1632011**
Task number	**:**	**SD03Q02**
Repository link	**:**	**https://github.com/Hariharanb47/Curneu-medtech-tasks.git**

Task - 2

Try to understand the dataset of Social_Network_Ads.csv and try to find the best suitable ML algorithm and write the code in python for algorithm from scratch and try to achieve the below output plot.



AIM

To understand the given social network ads dataset and to implementa suitable Machine learning model.

DATASET

Name	**:**	Social network ads dataset
Record size	**:**	401 records
Attributes	**:**	User id, Gender, age, estimated salary, purchased

Sample dataset

	User ID	Gender	Age	Estimated	Purchased
2	15624510	Male	19	19000	0
3	15810944	Male	35	20000	0
4	15668575	Female	26	43000	0
5	15603246	Female	27	57000	0
6	15804002	Male	19	76000	0
7	15728773	Male	27	58000	0
8	15598044	Female	27	84000	0
9	15694829	Female	32	150000	1
10	15600575	Male	25	33000	0
11	15727311	Female	35	65000	0
12	15570769	Female	26	80000	0
13	15606274	Female	26	52000	0
14	15746139	Male	20	86000	0
15	15704987	Male	32	18000	0

METHODOLOGY

Tool used : Google colaboratory (Python 3.6)

The given social network ads consists of attributes such as gender, estimated salary, age and purchased out which purchased is the resultant variable and rest of the others are features**.**

Step 1 : Dataset is uploaded

Step 2 : Dataset is splitted into train and test data for training and testing the model.

Code snippet

```
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)
```

Step 3 : Feature scaling is done for x and y.

Code snippet

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

Step 4 : Random forest classifier is implemented by using SKlearn package.

Random forest algorithm

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean/average prediction of the individual trees.

Code snippet

```
from sklearn.ensemble import RandomForestClassifier
classifier = RandomForestClassifier(n_estimators = 10,
criterion = 'entropy', random_state = 0)
classifier.fit(X_train, y_train)
```

Step 5 : Test results had been predicted

Output

```
[0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 1 1 0 1 0 0 1 0 1]
```

Step 6 : Visualization had been done and following are the visualizations.