

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
df = pd.read_csv("Social_Network_Ads.csv")
df
```

```
Out[1]:
```

	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
...
395	15691863	Female	46	41000	1
396	15706071	Male	51	23000	1
397	15654296	Female	50	20000	1
398	15755018	Male	36	33000	0
399	15594041	Female	49	36000	1

400 rows × 5 columns

```
In [2]: df_cleaned = df.drop(["User ID", "Gender"], axis=1)
df_cleaned
```

```
Out[2]:
```

	Age	EstimatedSalary	Purchased
0	19	19000	0
1	35	20000	0
2	26	43000	0
3	27	57000	0

	Age	EstimatedSalary	Purchased
4	19	76000	0
...
395	46	41000	1
396	51	23000	1
397	50	20000	1
398	36	33000	0
399	49	36000	1

400 rows × 3 columns

```
In [3]: x = df_cleaned.drop(["Purchased"],axis=1)
        y = df_cleaned["Purchased"]
        x
```

```
Out[3]:
```

	Age	EstimatedSalary
0	19	19000
1	35	20000
2	26	43000
3	27	57000
4	19	76000
...
395	46	41000
396	51	23000
397	50	20000
398	36	33000
399	49	36000

400 rows × 2 columns

```
In [4]: from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x,y)
print(x_train.shape, x_test.shape)
```

```
(300, 2) (100, 2)
```

```
In [5]: from sklearn.preprocessing import StandardScaler
scalar = StandardScaler()
x_train = scalar.fit_transform(x_train)
x_test = scalar.fit_transform(x_test)
x_train[0:5]
```

```
Out[5]: array([[ 0.902812, -1.28659058],
               [-0.69314683,  0.26814367],
               [-0.03598731,  0.21056092],
               [ 1.74773139, -0.27889245],
               [-1.06866655,  0.29693505]])
```

```
In [6]: from sklearn.svm import SVC
model = SVC(kernel="rbf", random_state=0)
model.fit(x_train,y_train)
y_pred = model.predict(x_test)
```

```
In [7]: model.score(x_train,y_train)*100
```

```
Out[7]: 91.66666666666666
```

```
In [8]: from sklearn.metrics import confusion_matrix
confusion_matrix(y_test,y_pred)
```

```
Out[8]: array([[57,  6],
               [ 5, 32]], dtype=int64)
```

```
In [11]: from sklearn.metrics import classification_report
print(classification_report(y_test,y_pred))
```

	precision	recall	f1-score	support
0	0.92	0.90	0.91	63
1	0.84	0.86	0.85	37

accuracy			0.89	100
macro avg	0.88	0.88	0.88	100
weighted avg	0.89	0.89	0.89	100

In []: