```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import confusion_matrix, accuracy_score
from google.colab import drive
drive.mount('/content/drive',force_remount=True)
```

```
Mounted at /content/drive
```

```python
file_path='/content/drive/My Drive/MACHINE LEARNING/Colab Notebooks/suv_data.c
df = pd.read_csv(file_path)
display(df.head())
```

|   | 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 |

```python
x=df[["Age","EstimatedSalary"]]
y=df["Purchased"]
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=
```

```python
log_reg=LogisticRegression()
log_reg.fit(x_train,y_train)
y_pred=log_reg.predict(x_test)
```

```python
tp,fn,fp,tn=confusion_matrix(y_test,y_pred).ravel()
print(f"True Positive: {tp}     False Negative: {fn}")
print(f"False Positive: {fp}    True Negative: {tn}")
```

```
True Positive: 74     False Negative: 5
False Positive: 10    True Negative: 31
```

```python
test_acc=accuracy_score(y_test,y_pred)
train_acc=log_reg.score(x_train,y_train)
print(f"Training Accuracy: {train_acc:.2f}")
print(f"Test Accuracy: {test_acc:.2f}")
```

```
Training Accuracy: 0.84
Test Accuracy: 0.88
```