```
import pandas as pd

df = pd.read_csv('Churn_Modelling.csv')
df.head()
```

```
   RowNumber  CustomerId   Surname  CreditScore Geography  Gender  Age
\
0          1    15634602  Hargrave          619    France  Female   42

1          2    15647311      Hill          608     Spain  Female   41

2          3    15619304      Onio          502    France  Female   42

3          4    15701354      Boni          699    France  Female   39

4          5    15737888  Mitchell          850     Spain  Female   43


   Tenure    Balance  NumOfProducts  HasCrCard  IsActiveMember  \
0       2       0.00              1          1               1
1       1   83807.86              1          0               1
2       8  159660.80              3          1               0
3       1       0.00              2          0               0
4       2  125510.82              1          1               1

   EstimatedSalary  Exited
0        101348.88       1
1        112542.58       0
2        113931.57       1
3         93826.63       0
4         79084.10       0
```

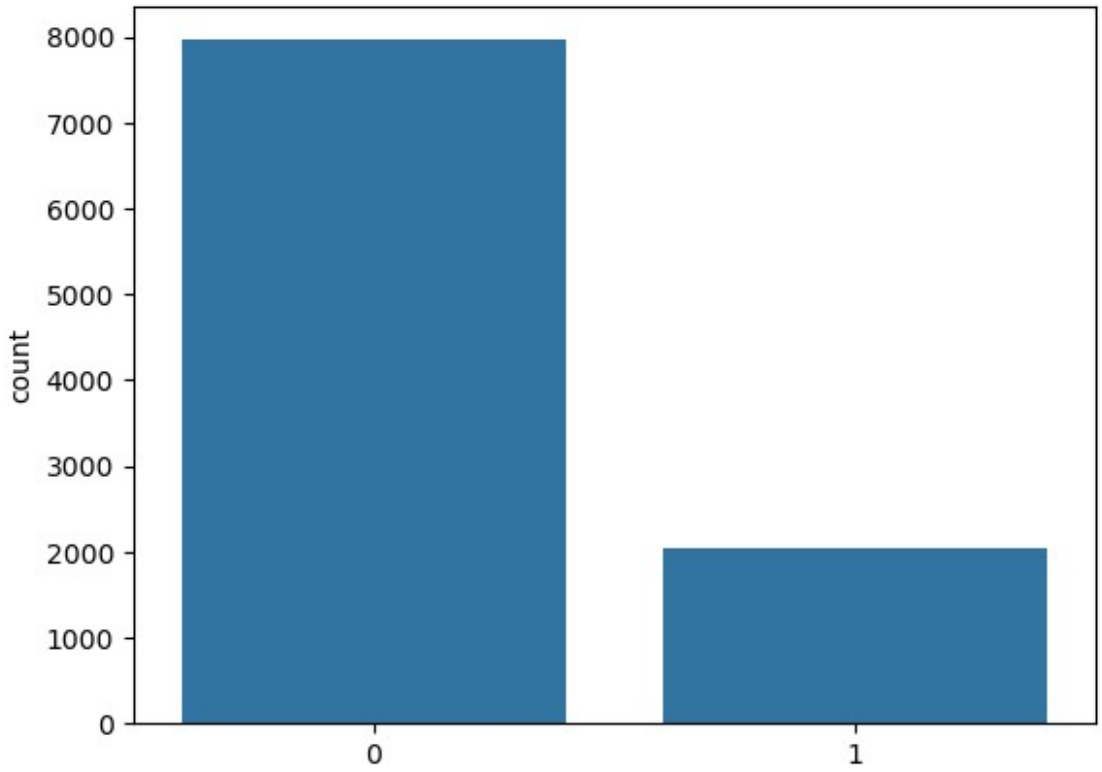
```
# Separate the feature and target sets
df.drop(columns=['RowNumber','CustomerId','Surname','Geography','Gende
r'],inplace=True)

y = df['Exited'].values
x = df.loc[:,df.columns != 'Exited'].values

# Split the data into training and test sets
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=42)

import seaborn as sns
sns.countplot(x=y)
```

```
<Axes: ylabel='count'>
```




```
from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()
x_train = scaler.fit_transform(x_train)
x_test = scaler.transform(x_test)

!pip install tensorflow
```

```
Collecting tensorflow
  Using cached tensorflow-2.18.0-cp312-cp312-win_amd64.whl.metadata
(3.3 kB)
Collecting tensorflow-intel==2.18.0 (from tensorflow)
  Using cached tensorflow_intel-2.18.0-cp312-cp312-
win_amd64.whl.metadata (4.9 kB)
Requirement already satisfied: absl-py>=1.0.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (2.1.0)
Requirement already satisfied: astunparse>=1.6.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (1.6.3)
Requirement already satisfied: flatbuffers>=24.3.25 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (24.3.25)
Requirement already satisfied: gast!=0.5.0,!=0.5.1,!=0.5.2,>=0.2.1 in
c:\users\shubham\anaconda3\lib\site-packages (from tensorflow-
intel==2.18.0->tensorflow) (0.6.0)
```

```
Requirement already satisfied: google-pasta>=0.1.1 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (0.2.0)
Requirement already satisfied: libclang>=13.0.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (18.1.1)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (3.4.0)
Requirement already satisfied: packaging in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (23.2)
Requirement already satisfied: protobuf!=4.21.0,!=4.21.1,!=4.21.2,!
=4.21.3,!=4.21.4,!=4.21.5,<6.0.0dev,>=3.20.3 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (3.20.3)
Requirement already satisfied: requests<3,>=2.21.0 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (2.32.2)
Requirement already satisfied: setuptools in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (69.5.1)
Requirement already satisfied: six>=1.12.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (1.16.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (2.5.0)
Requirement already satisfied: typing-extensions>=3.6.6 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (4.11.0)
Requirement already satisfied: wrapt>=1.11.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (1.14.1)
Requirement already satisfied: grpcio<2.0,>=1.24.3 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (1.67.1)
Requirement already satisfied: tensorboard<2.19,>=2.18 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (2.18.0)
Requirement already satisfied: keras>=3.5.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (3.6.0)
Requirement already satisfied: numpy<2.1.0,>=1.26.0 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (1.26.4)
Requirement already satisfied: h5py>=3.11.0 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (3.11.0)
```

```
Requirement already satisfied: ml-dtypes<0.5.0,>=0.4.0 in c:\users\
shubham\anaconda3\lib\site-packages (from tensorflow-intel==2.18.0-
>tensorflow) (0.4.1)
Requirement already satisfied: wheel<1.0,>=0.23.0 in c:\users\shubham\
anaconda3\lib\site-packages (from astunparse>=1.6.0->tensorflow-
intel==2.18.0->tensorflow) (0.43.0)
Requirement already satisfied: rich in c:\users\shubham\anaconda3\lib\
site-packages (from keras>=3.5.0->tensorflow-intel==2.18.0-
>tensorflow) (13.3.5)
Requirement already satisfied: namex in c:\users\shubham\anaconda3\
lib\site-packages (from keras>=3.5.0->tensorflow-intel==2.18.0-
>tensorflow) (0.0.8)
Requirement already satisfied: optree in c:\users\shubham\anaconda3\
lib\site-packages (from keras>=3.5.0->tensorflow-intel==2.18.0-
>tensorflow) (0.13.0)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\
shubham\anaconda3\lib\site-packages (from requests<3,>=2.21.0-
>tensorflow-intel==2.18.0->tensorflow) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\shubham\
anaconda3\lib\site-packages (from requests<3,>=2.21.0->tensorflow-
intel==2.18.0->tensorflow) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\shubham\
anaconda3\lib\site-packages (from requests<3,>=2.21.0->tensorflow-
intel==2.18.0->tensorflow) (2.2.2)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\shubham\
anaconda3\lib\site-packages (from requests<3,>=2.21.0->tensorflow-
intel==2.18.0->tensorflow) (2024.8.30)
Requirement already satisfied: markdown>=2.6.8 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorboard<2.19,>=2.18->tensorflow-
intel==2.18.0->tensorflow) (3.4.1)
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0
in c:\users\shubham\anaconda3\lib\site-packages (from
tensorboard<2.19,>=2.18->tensorflow-intel==2.18.0->tensorflow) (0.7.2)
Requirement already satisfied: werkzeug>=1.0.1 in c:\users\shubham\
anaconda3\lib\site-packages (from tensorboard<2.19,>=2.18->tensorflow-
intel==2.18.0->tensorflow) (3.0.3)
Requirement already satisfied: MarkupSafe>=2.1.1 in c:\users\shubham\
anaconda3\lib\site-packages (from werkzeug>=1.0.1-
>tensorboard<2.19,>=2.18->tensorflow-intel==2.18.0->tensorflow)
(2.1.3)
Requirement already satisfied: markdown-it-py<3.0.0,>=2.2.0 in c:\
users\shubham\anaconda3\lib\site-packages (from rich->keras>=3.5.0-
>tensorflow-intel==2.18.0->tensorflow) (2.2.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in c:\users\
shubham\anaconda3\lib\site-packages (from rich->keras>=3.5.0-
>tensorflow-intel==2.18.0->tensorflow) (2.15.1)
Requirement already satisfied: mdurl~=0.1 in c:\users\shubham\
anaconda3\lib\site-packages (from markdown-it-py<3.0.0,>=2.2.0->rich-
>keras>=3.5.0->tensorflow-intel==2.18.0->tensorflow) (0.1.0)
```

```
Using cached tensorflow-2.18.0-cp312-cp312-win_amd64.whl (7.5 kB)
Using cached tensorflow_intel-2.18.0-cp312-cp312-win_amd64.whl (390.3
MB)
Installing collected packages: tensorflow-intel, tensorflow
Successfully installed tensorflow-2.18.0 tensorflow-intel-2.18.0
```

```
import tensorflow as tf
from tensorflow import keras

# Define the model architecture
model = keras.Sequential([
    keras.layers.Dense(64, input_dim=x_train.shape[1],
activation='relu'),  # Input layer
    keras.layers.Dense(32, activation='relu'),
# Hidden layer
    keras.layers.Dense(1, activation='sigmoid')
# Output layer for binary classification
])

# Compile the model
model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])

# Train the model
model.fit(x_train, y_train, epochs=10, batch_size=32)
```

```
C:\Users\Shubham\anaconda3\Lib\site-packages\keras\src\layers\core\
dense.py:87: UserWarning: Do not pass an `input_shape`/`input_dim`
argument to a layer. When using Sequential models, prefer using an
`Input(shape)` object as the first layer in the model instead.
  super().__init__(activity_regularizer=activity_regularizer,
**kwargs)

Epoch 1/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 4s 2ms/step - accuracy: 0.7544 - loss:
0.5144
Epoch 2/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8470 - loss:
0.3931
Epoch 3/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8455 - loss:
0.3696
Epoch 4/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8489 - loss:
0.3662
Epoch 5/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8633 - loss:
0.3384
Epoch 6/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8536 - loss:
```

```
0.3468
Epoch 7/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8564 - loss:
0.3439
Epoch 8/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 1s 2ms/step - accuracy: 0.8566 - loss:
0.3465
Epoch 9/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8543 - loss:
0.3506
Epoch 10/10
235/235 ━━━━━━━━━━━━━━━━━━━━ 0s 2ms/step - accuracy: 0.8592 - loss:
0.3400

<keras.src.callbacks.history.History at 0x156c795c1a0>
```

```
from sklearn.metrics import accuracy_score, confusion_matrix

y_pred = model.predict(x_test)
y_pred = (y_pred > 0.5).astype(int)  # Convert probabilities to binary
predictions (0 or 1)

accuracy = accuracy_score(y_test, y_pred)
confusion_mat = confusion_matrix(y_test, y_pred)

print("Accuracy:", accuracy)
print("Confusion Matrix:")
print(confusion_mat)
```

```
79/79 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step
Accuracy: 0.8588
Confusion Matrix:
[[1932   71]
 [ 282  215]]
```

To implement the improvement

```
model = keras.Sequential([
    keras.layers.Input(shape=(x_train.shape[1],)),  # Input layer with
the same number of features
    keras.layers.Dense(128, activation='relu'),    # Increased the
number of neurons in the first hidden layer
    keras.layers.Dropout(0.3),                    # Added dropout to
prevent overfitting
    keras.layers.Dense(64, activation='relu'),     # Added another
hidden layer
    keras.layers.BatchNormalization(),             # Batch
normalization layer
```

```
    keras.layers.Dense(1, activation='sigmoid')
])

model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])

# Train the model with early stopping
early_stopping = keras.callbacks.EarlyStopping(patience=5,
restore_best_weights=True)

history = model.fit(x_train, y_train, epochs=50, batch_size=32,
validation_split=0.1, callbacks=[early_stopping])

# Step 5: Print the accuracy score and confusion matrix
y_pred = model.predict(x_test)
y_pred = (y_pred > 0.5).astype(int)

accuracy = accuracy_score(y_test, y_pred)
confusion_mat = confusion_matrix(y_test, y_pred)

print("Accuracy:", accuracy)
print("Confusion Matrix:")
print(confusion_mat)
```

```
Epoch 1/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 6s 5ms/step - accuracy: 0.6494 - loss:
0.6437 - val_accuracy: 0.8320 - val_loss: 0.4148
Epoch 2/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8190 - loss:
0.4245 - val_accuracy: 0.8480 - val_loss: 0.3670
Epoch 3/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8366 - loss:
0.3897 - val_accuracy: 0.8467 - val_loss: 0.3553
Epoch 4/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8362 - loss:
0.3819 - val_accuracy: 0.8453 - val_loss: 0.3647
Epoch 5/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8420 - loss:
0.3842 - val_accuracy: 0.8480 - val_loss: 0.3485
Epoch 6/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8430 - loss:
0.3743 - val_accuracy: 0.8467 - val_loss: 0.3503
Epoch 7/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 4ms/step - accuracy: 0.8425 - loss:
0.3771 - val_accuracy: 0.8400 - val_loss: 0.3570
Epoch 8/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8498 - loss:
0.3688 - val_accuracy: 0.8467 - val_loss: 0.3465
Epoch 9/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8428 - loss:
```

```
0.3792 - val_accuracy: 0.8440 - val_loss: 0.3469
Epoch 10/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8452 - loss:
0.3656 - val_accuracy: 0.8413 - val_loss: 0.3542
Epoch 11/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8465 - loss:
0.3706 - val_accuracy: 0.8440 - val_loss: 0.3493
Epoch 12/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8530 - loss:
0.3573 - val_accuracy: 0.8480 - val_loss: 0.3478
Epoch 13/50
211/211 ━━━━━━━━━━━━━━━━━━━━ 1s 3ms/step - accuracy: 0.8585 - loss:
0.3473 - val_accuracy: 0.8493 - val_loss: 0.3494
79/79 ━━━━━━━━━━━━━━━━━━━━ 0s 3ms/step
Accuracy: 0.8596
Confusion Matrix:
[[1945   58]
 [ 293  204]]
```