Machine Learning

Now, we are done with both exploratory data analysis and preprocessing. Now we will go ahead with performing machine learning.

Importing all packages

In [74]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import *
from sklearn.linear_model import *
from math import *
from sklearn.ensemble import *
from sklearn.feature_selection import *
from sklearn.feature_extraction import *
from sklearn.naive_bayes import *
from sklearn.discriminant_analysis import *
from sklearn.preprocessing import *
from sklearn.metrics import *
from sklearn.neighbors import *
from sklearn.cluster import *
from sklearn.svm import *
import warnings
warnings.filterwarnings("ignore")
```

Importing all datasets

In [75]:
```
df_train = pd.read_csv("train_preprocessed.csv")
df_test = pd.read_csv("test_preprocessed.csv")
```

Displaying first 5 elements of training dataset

In [76]:
```
df_train.head()
```

Out[76]:

	PassengerId	HomePlanet_Europa	HomePlanet_Mars	Cabin Desk B	Cabin Desk C	Cabin Desk D	Cabin Desk E	Cabin Desk F	Ca D
0	0001_01	1.0	0.0	1.0	0.0	0.0	0.0	0.0	
1	0002_01	0.0	0.0	0.0	0.0	0.0	0.0	1.0	
2	0003_01	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	0003_02	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	0004_01	0.0	0.0	0.0	0.0	0.0	0.0	1.0	

5 rows × 23 columns

Displaying first 5 elements of testing dataset

In [77]:
```
df_test.head()
```

Out[77]:

	PassengerId	HomePlanet_Europa	HomePlanet_Mars	Cabin Desk B	Cabin Desk C	Cabin Desk D	Cabin Desk E	Cabin Desk F	Ca D
0	0013_01	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0018_01	0.0	0.0	0.0	0.0	0.0	0.0	1.0	
2	0019_01	1.0	0.0	0.0	1.0	0.0	0.0	0.0	
3	0021_01	1.0	0.0	0.0	1.0	0.0	0.0	0.0	
4	0023_01	0.0	0.0	0.0	0.0	0.0	0.0	1.0	

5 rows × 22 columns

Length of training and testing datasets

In [78]:
```
print("Training length : ",len(df_train))
print("Testing length  : ",len(df_test))
```

```
Training length :  8122
Testing length  :  4017
```

In [79]:
```
print("Training percentage : ",round((len(df_train)*100/(len(df_train)+len(df_t
print("Testing percentage  : ",round((len(df_test)*100/(len(df_train)+len(df_te
```

```
Training percentage :  66.91
Testing percentage  :  33.09
```

Splitting the training data into input and output

In [80]:
```
X_train = df_train.drop("Transported",axis=1,inplace=False)
y_train = df_train["Transported"]

X_test = df_test.copy()
```

In [81]:
```
lsvc = SVC()
model = lsvc.fit(X_train,y_train)
```

In [82]:
```
y_test = pd.Series(model.predict(X_test))
```

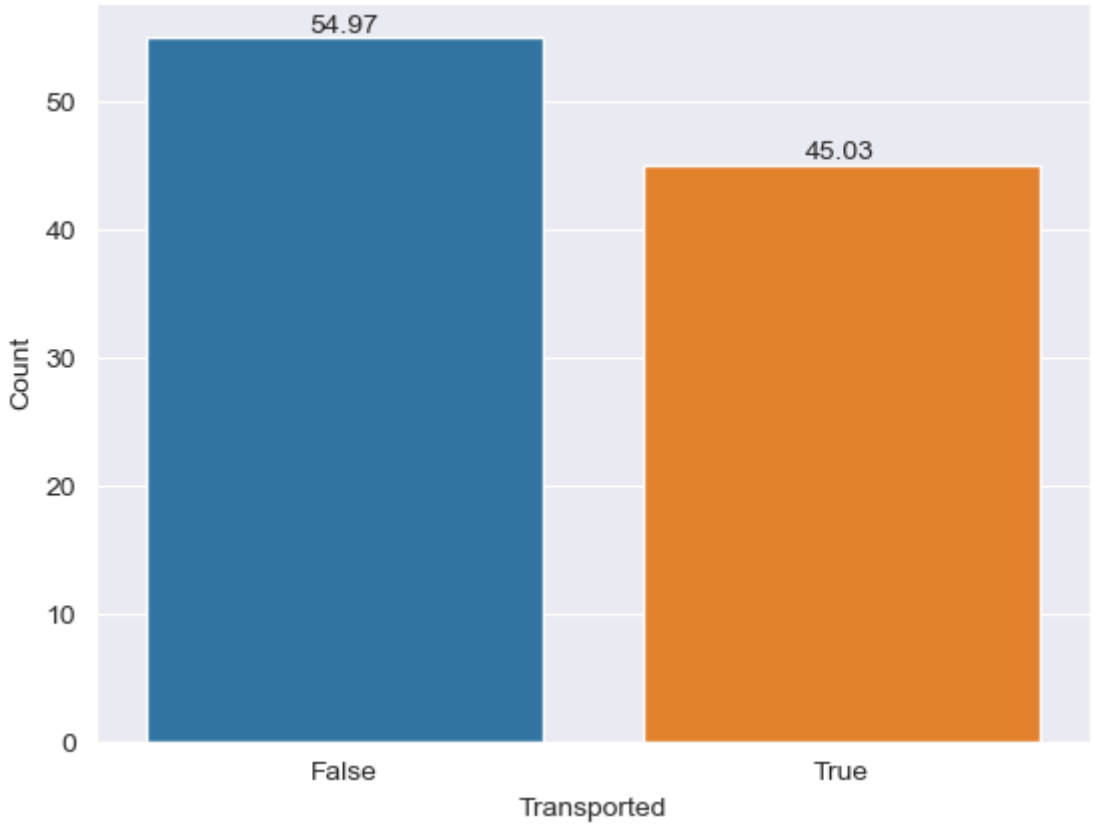
In [83]:
```
y_test = y_test.map({0:False,1:True})
```

In [84]:
```
v = round((y_test.value_counts()*100/y_test.value_counts().sum()),2)
v = v.reset_index()
v
```

Out[84]:

	index	0
0	False	54.97
1	True	45.03

In [85]:
```
plot = sns.barplot(x=v["index"],y=v[0])
plot.set(ylabel="Count",xlabel="Transported")
for i in plot.containers:
    plot.bar_label(i,)
```



In [86]:
```
sub_df = pd.DataFrame(columns=["PassengerId","Transported"])
sub_df["PassengerId"] = df_test["PassengerId"]
sub_df["Transported"] = y_test
```

In [87]:
```
sub_df.head()
```

Out[87]:

	PassengerId	Transported
0	0013_01	False
1	0018_01	False
2	0019_01	False
3	0021_01	False
4	0023_01	False

In [88]:
```
main_df1 = pd.read_csv("test.csv")
main_df = main_df1["PassengerId"]
```

In [89]:
```
merge_df = pd.merge(left=main_df,right=sub_df,how="left",on="PassengerId")
merge_df.head()
```

Out[89]:

	PassengerId	Transported
0	0013_01	False
1	0018_01	False
2	0019_01	False
3	0021_01	False
4	0023_01	False

In [90]:
```
merge_df.isna().sum()
```

Out[90]:
```
PassengerId       0
Transported     260
dtype: int64
```

In [91]:
```
final_df = merge_df.fillna(value=False,inplace=False)
final_df.head()
```

Out[91]:

	PassengerId	Transported
0	0013_01	False
1	0018_01	False
2	0019_01	False
3	0021_01	False
4	0023_01	False

In [92]:
```
final_df.isna().sum()
```

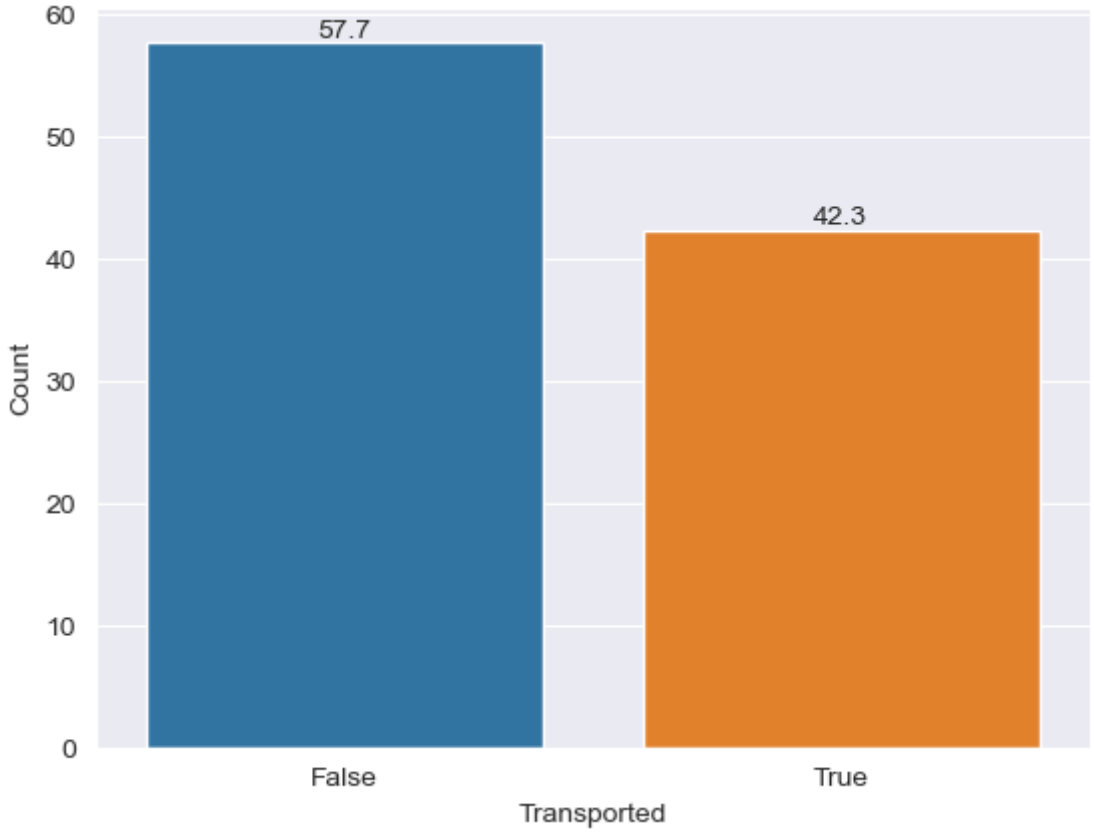
Out[92]:
```
PassengerId     0
Transported     0
dtype: int64
```

In [93]:
```
cou = round((final_df["Transported"].value_counts()*100/final_df["Transported"]
cou = cou.reset_index()
cou
```

Out[93]:

	index	Transported
0	False	57.7
1	True	42.3

In [94]:
```
plot1 = sns.barplot(x=cou["index"],y=cou["Transported"])
plot1.set(ylabel="Count",xlabel="Transported")
for i in plot1.containers:
    plot1.bar_label(i,)
```



In [95]:
```
final_df.to_csv("amith_submission.csv",index=False)
```

In [ ]:
```

```