HARSH KUMAR

Pardon. Submitting the file after due date as I was suffering from food poisioning earlier.

21BDS0391

harsh.kumar2021d@vitstudent.ac.in

```
import numpy as np
import pandas as pd
import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
```

```
/kaggle/input/titanic-dataset/Titanic-Dataset.csv
```

```
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler


train_data = pd.read_csv("/kaggle/input/titanic-dataset/Titanic-Dataset.csv")
test_data = pd.read_csv("/kaggle/input/titanic-dataset/Titanic-Dataset.csv")

print("Summary of train_data:")
print(train_data.info())

print("\nSummary of test_data:")
print(test_data.info())

print("\nFirst few rows of train_data:")
print(train_data.head())

print("\nFirst few rows of test_data:")
print(test_data.head())
```

```
Summary of train_data:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
None

Summary of test_data:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
None
```

```
First few rows of train_data:
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0
```

```
print("Training Data Null Values:")
print(train_data.isnull().sum())

print("\nTest Data Null Values:")
print(test_data.isnull().sum())
```

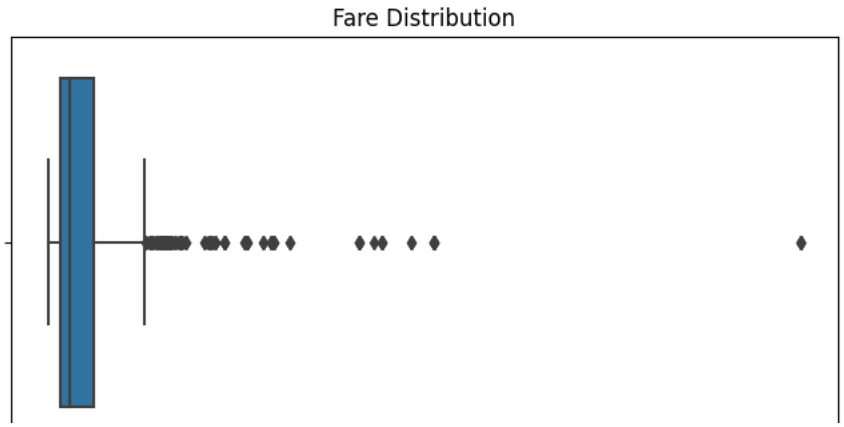
```
Training Data Null Values:
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64

Test Data Null Values:
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

DATA VISUALIZATION

```
sns.countplot(data=train_data, x='Sex', hue='Survived')
plt.title('Survival Count by Gender')
plt.show()
```

```
plt.figure(figsize=(8, 4))
sns.boxplot(data=train_data, x='Fare')
plt.title('Fare Distribution')
plt.show()
```




```
y = train_data["Survived"]
features = ["Pclass", "Sex", "SibSp", "Parch"]
X = pd.get_dummies(train_data[features])
X_test = pd.get_dummies(test_data[features])

print("Dependent Variable (y):")
print(y)

print("\nIndependent Variables (X):")
print(X)

print("\nIndependent Variables for Test Data (X_test):")
print(X_test)
```

```
Dependent Variable (y):
0      0
1      1
2      1
3      1
4      0
      ..
886    0
887    1
888    0
889    1
890    0
Name: Survived, Length: 891, dtype: int64

Independent Variables (X):
     Pclass  SibSp  Parch  Sex_female  Sex_male
0         3      1      0       False      True
1         1      1      0        True     False
2         3      0      0        True     False
3         1      1      0        True     False
4         3      0      0       False      True
..      ...    ...    ...         ...       ...
886       2      0      0       False      True
887       1      0      0        True     False
888       3      1      2        True     False
889       1      0      0       False      True
890       3      0      0       False      True

[891 rows x 5 columns]

Independent Variables for Test Data (X_test):
     Pclass  SibSp  Parch  Sex_female  Sex_male
0         3      1      0       False      True
1         1      1      0        True     False
2         3      0      0        True     False
3         1      1      0        True     False
4         3      0      0       False      True
..      ...    ...    ...         ...       ...
886       2      0      0       False      True
887       1      0      0        True     False
888       3      1      2        True     False
889       1      0      0       False      True
890       3      0      0       False      True
```

```
[891 rows x 5 columns]
```

FEATURE SCALING

```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
X_test_scaled = scaler.transform(X_test)
print("X_scaled (Training Data):")
print(X_scaled)

print("\nX_test_scaled (Test Data):")
print(X_test_scaled)
```

```
X_scaled (Training Data):
[[ 0.82737724  0.43279337 -0.47367361 -0.73769513  0.73769513]
 [-1.56610693  0.43279337 -0.47367361  1.35557354 -1.35557354]
 [ 0.82737724 -0.4745452  -0.47367361  1.35557354 -1.35557354]
 ...
 [ 0.82737724  0.43279337  2.00893337  1.35557354 -1.35557354]
 [-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]]

X_test_scaled (Test Data):
[[ 0.82737724  0.43279337 -0.47367361 -0.73769513  0.73769513]
 [-1.56610693  0.43279337 -0.47367361  1.35557354 -1.35557354]
 [ 0.82737724 -0.4745452  -0.47367361  1.35557354 -1.35557354]
 ...
 [ 0.82737724  0.43279337  2.00893337  1.35557354 -1.35557354]
 [-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]]
```

SPLITTING THE DATASET

```
X_train, X_val, y_train, y_val = train_test_split(X_scaled, y, test_size=0.2, random_state=1)
print("X_train:")
print(X_train)

print("\nX_val:")
print(X_val)

print("\ny_train:")
print(y_train)

print("\ny_val:")
print(y_val)
```

```
 [ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [-0.36936484 -0.4745452  -0.47367361  1.35557354 -1.35557354]
 [ 0.82737724 -0.4745452  -0.47367361  1.35557354 -1.35557354]
 [-0.36936484 -0.4745452   2.00893337  1.35557354 -1.35557354]
 [ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [-0.36936484 -0.4745452   2.00893337 -0.73769513  0.73769513]
 [-1.56610693  0.43279337 -0.47367361  1.35557354 -1.35557354]
 [-1.56610693  0.43279337 -0.47367361 -0.73769513  0.73769513]
 [ 0.82737724  0.43279337 -0.47367361  1.35557354 -1.35557354]
 [ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [ 0.82737724 -0.4745452  -0.47367361  1.35557354 -1.35557354]
 [-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [-0.36936484 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [ 0.82737724 -0.4745452  -0.47367361  1.35557354 -1.35557354]
 [-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [ 0.82737724  2.24747049  2.00893337 -0.73769513  0.73769513]
 [ 0.82737724  0.43279337 -0.47367361 -0.73769513  0.73769513]
 [ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
 [-0.36936484  1.34013193  3.25023685  1.35557354 -1.35557354]
 [ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
```

```
[ 0.82737724  0.43279337 -0.47367361  1.35557354 -1.35557354]
[ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[ 0.82737724  3.15480905  2.00893337  1.35557354 -1.35557354]
[-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-1.56610693  0.43279337 -0.47367361 -0.73769513  0.73769513]
[ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[ 0.82737724 -0.4745452  -0.47367361  1.35557354 -1.35557354]
[ 0.82737724 -0.4745452   2.00893337  1.35557354 -1.35557354]
[-1.56610693  0.43279337  0.76762988  1.35557354 -1.35557354]
[-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-0.36936484 -0.4745452   2.00893337 -0.73769513  0.73769513]
[ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[ 0.82737724  4.06214761  2.00893337 -0.73769513  0.73769513]
[-0.36936484 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-0.36936484 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-0.36936484 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-0.36936484 -0.4745452  -0.47367361  1.35557354 -1.35557354]
[-0.36936484 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-0.36936484 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[ 0.82737724 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[ 0.82737724  0.43279337 -0.47367361  1.35557354 -1.35557354]
[-0.36936484 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[-1.56610693 -0.4745452  -0.47367361 -0.73769513  0.73769513]
[ 0.82737724  0.43279337  5.73284383 -0.73769513  0.73769513]
```