Task 5

Exploratary Data Analysis

- Initially downloaded the dataset from Kaggle and extracted the files from zip
- Later uploaded the files into google colab

- **Creating a data frame :**
  - Initially import pandas library (which helps for data visualization) and then create a data frame
  - After creating the data frame upload the file path in it and run the cell
  - **Code** : import pandas as pd
    df=pd.read_csv("train.csv")

- **Performing operations after creating data frame :**
  - To know the information of the dataset u can use info() function
  - **Code :** df.info( )
  - **Output :**

```
df.info()
```

```
<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
```

- **To describe about the dataset** we use the function called describe ( )
- **Code :** df.describe( )
- **Output :**

```
df.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

- **To describe about the values of the column**
- **Code :** df['survived’].value_counts()
- **[** Here ‘survived’ is a column name **]**
- **Output :**

```
df['Survived'].value_counts()
```

Survived	count
0	549
1	342

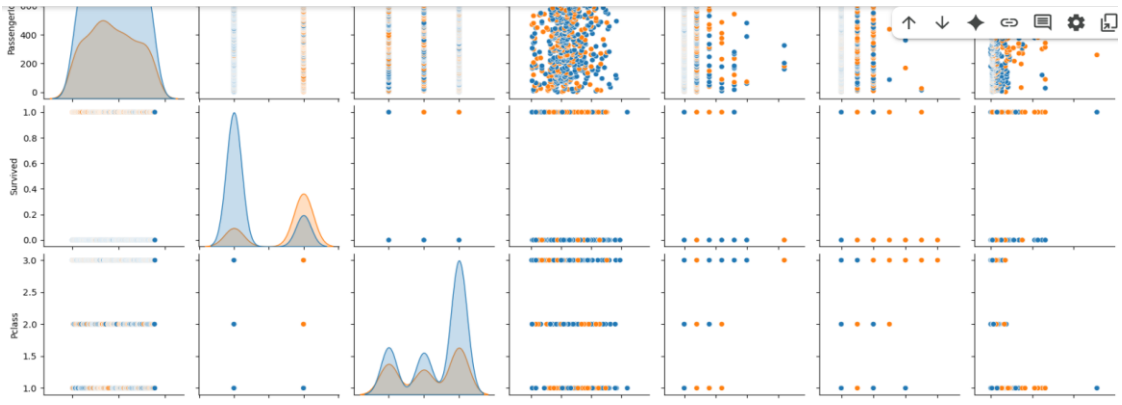
dtype: int64

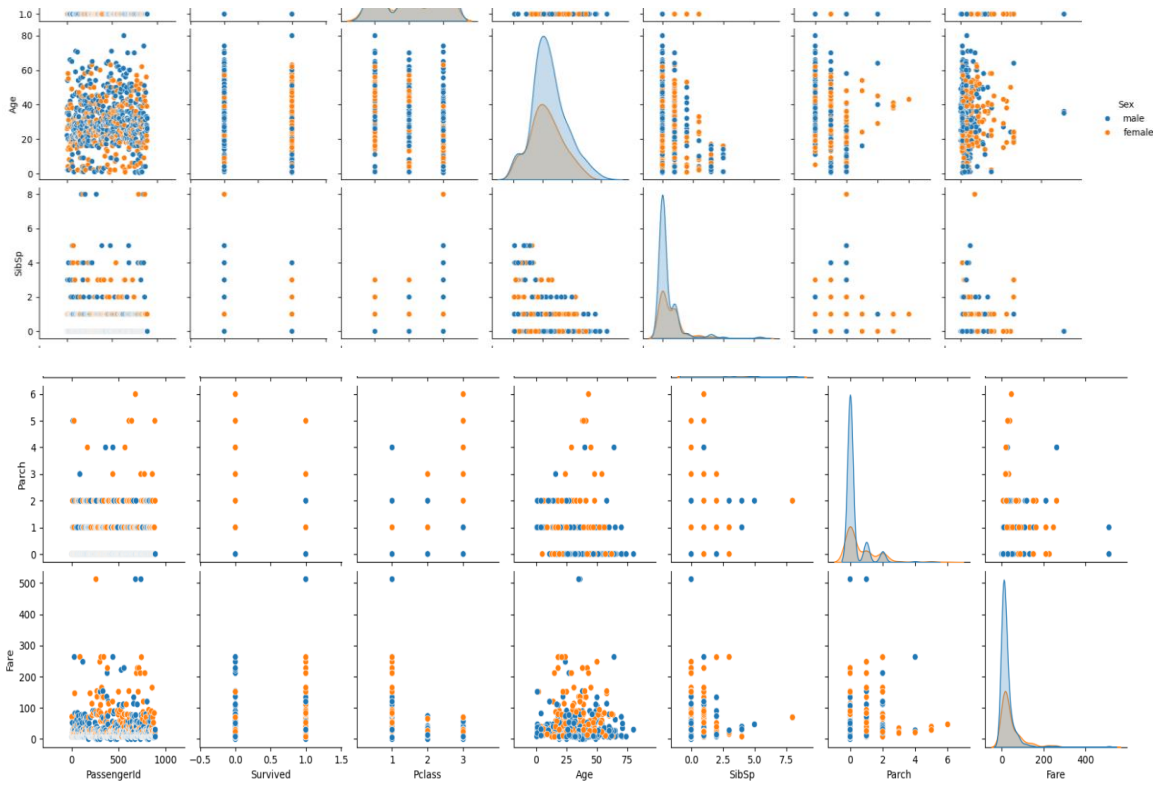
➢ Different plots used for visualization

- **sns.pairplot( ) :**
- **code :**

```
import seaborn as sns
import matplotlib.pyplot as plt
sns.pairplot(df, hue='Sex')
plt.show()
```

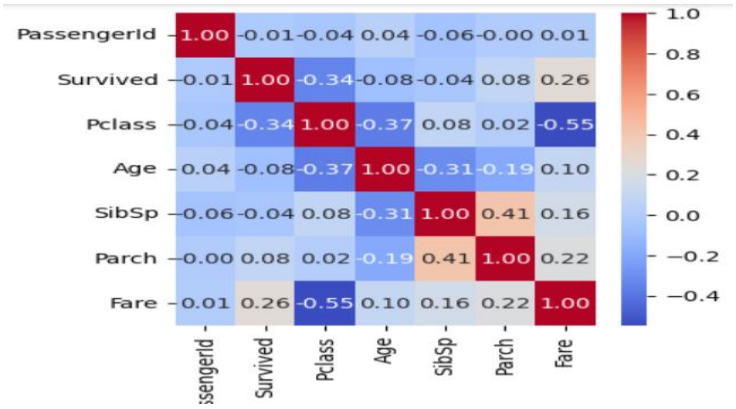
- **Output :**





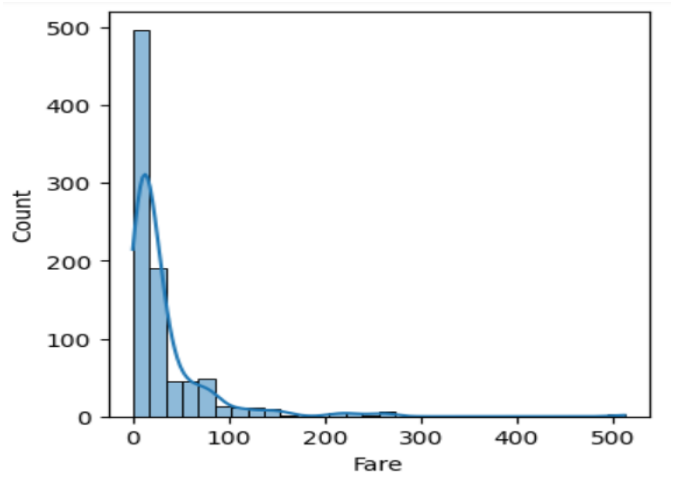
- **Observation :** For each column in the dataset there is a separate plot

- **Inorder to create a heatmap first u need to find corelation of the dataframe**
- **Code :** corr = df.corr(numeric_only=True)
- **sns.heatmap():**
- **Code :** import matplotlib.pyplot as plt
  import seaborn as sns
  plt.figure(figsize=(8,6))
  sns.heatmap(corr, annot=True, cmap='coolwarm', fmt=".2f")
  plt.show()
- **Output :**

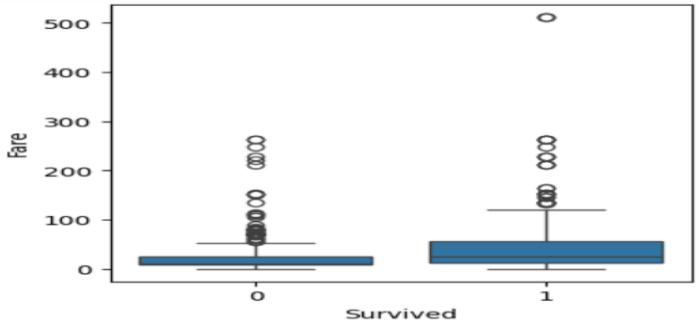


- **Observation :** survival has positive corelation with fare and negative corelation with passenger class

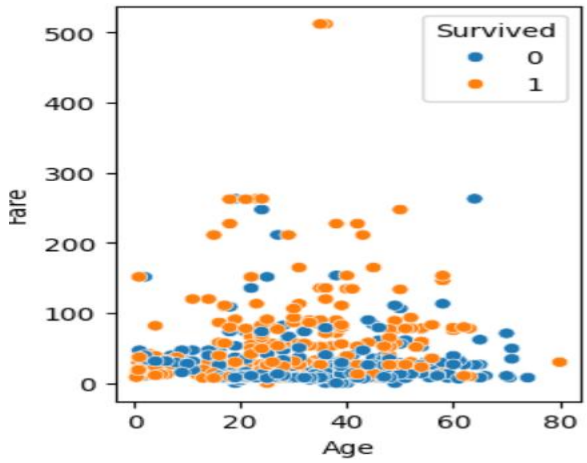
- **Histogram :**
- **Code :**
```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(8,5))
sns.histplot(df['Fare'], bins=30, kde=True)
plt.title('Fare Distribution')
plt.show()
```
- **Output :**



- **Boxplot :**
- **Code :**
```
plt.figure(figsize=(8,5))
sns.boxplot(x='Survived', y='Fare', data=df)
plt.title('Fare vs Survival')
plt.show()
```
- **Output :**



- **Scatterplot :**
- **Code :** plt.figure(figsize=(3,4))
  sns.scatterplot(x='Age', y='Fare', hue='Survived', data=df)
  plt.title('Age vs Fare by Survival')
  plt.show()
- **Output :**



- **Observation :**
- The death ratio is less between 60 – 80 age group and more between 20-60 age group
- The people who paid more fare have less no of deaths