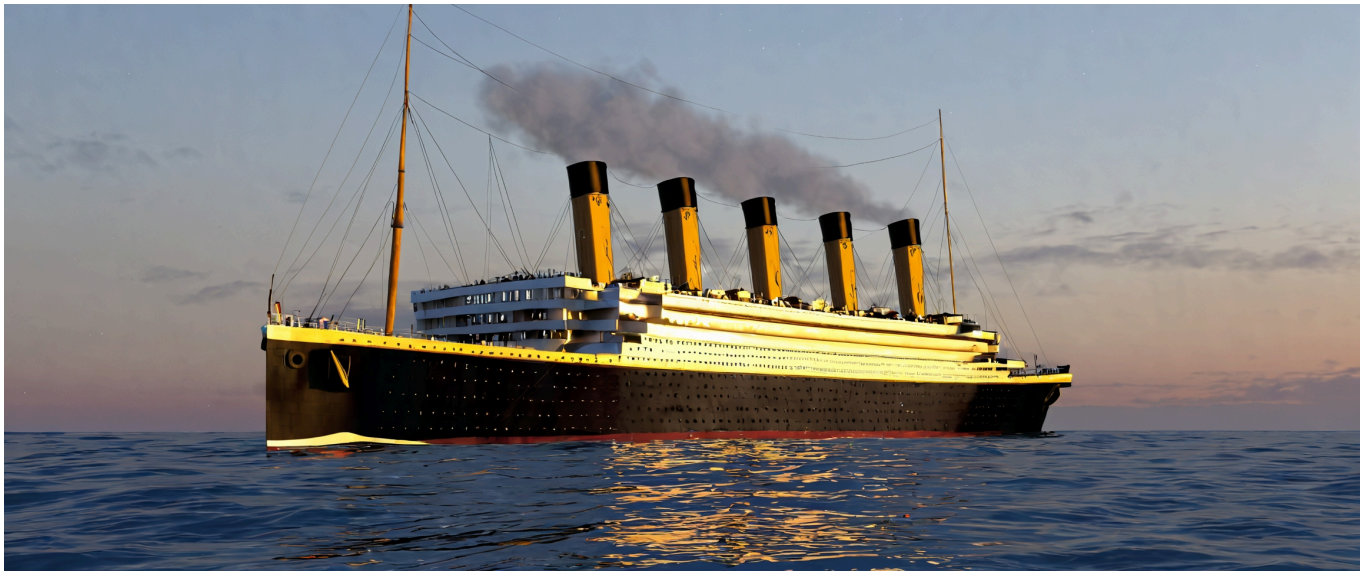
EDA On Titanic Dataset

Double-click (or enter) to edit

```
Start coding or generate with AI.
```



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

```
df= pd.read_csv('train.csv')
```

```
df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
				Futrelle, Mrs. Jacques								

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
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
```

```
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

```
(df.isnull().sum()/df.shape[0])*100
```

	0
PassengerId	0.000000
Survived	0.000000
Pclass	0.000000
Name	0.000000
Sex	0.000000
Age	19.865320
SibSp	0.000000
Parch	0.000000
Ticket	0.000000
Fare	0.000000
Cabin	77.104377
Embarked	0.224467

dtype: float64

```
df.drop('Cabin',axis=1,inplace=True)
```

```
(df.isnull().sum()/df.shape[0])*100
```

	0
PassengerId	0.000000
Survived	0.000000
Pclass	0.000000
Name	0.000000
Sex	0.000000
Age	19.865320
SibSp	0.000000
Parch	0.000000
Ticket	0.000000
Fare	0.000000
Embarked	0.224467

dtype: float64

```
df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	S

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df.drop('PassengerId',axis=1,inplace=True)
```

```
df.head()
```

	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	S
1	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C
2	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	S
3	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	S

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df['Ticket'].unique()
```

```
       'SOTON/O2 3101287', '2683', '315090', 'C.A. 5547', '349213',
       '347060', 'PC 17592', '392091', '113055', '2629', '350026',
       '28134', '17466', '233866', '236852', 'SC/PARIS 2149', 'PC 17590',
       '345777', '349248', '695', '345765', '2667', '349212', '349217',
       '349257', '7552', 'C.A./SOTON 34068', 'SOTON/OQ 392076', '211536',
       '112053', '111369', '370376'], dtype=object)
```

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

Ticket	count
347082	7
CA. 2343	7
1601	7
3101295	6
CA 2144	6
...	...
9234	1
19988	1
2693	1
PC 17612	1
370376	1

681 rows × 1 columns

dtype: int64

```
df.drop('Ticket',axis=1,inplace=True)
```

```
df.head()
```

	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Fare	Embarked
0	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	7.2500	S
1	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	71.2833	C
2	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	7.9250	S
3	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	53.1000	S
4	0	3	Allen, Mr. William Henry	male	35.0	0	0	8.0500	S

Next steps: Generate code with df | View recommended plots | New interactive sheet

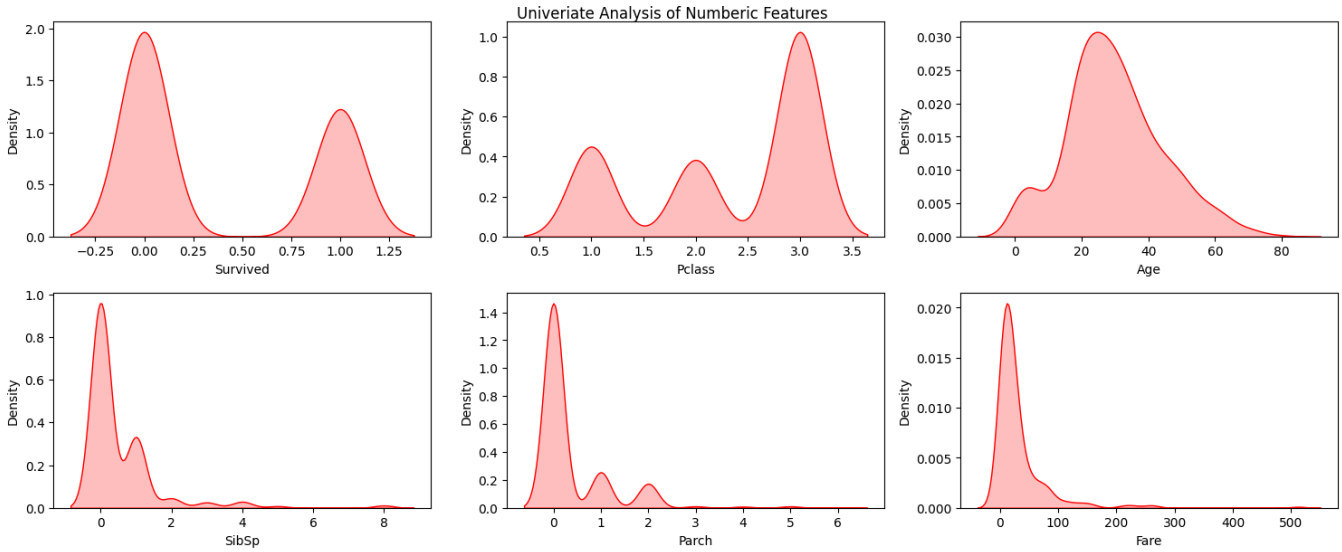
```
numeric_feature = [f for f in df.columns if df[f].dtype!='O']
categorical_feature = [f for f in df.columns if df[f].dtype=='O']
```

```
print(numeric_feature)
print(categorical_feature)
```

```
['Survived', 'Pclass', 'Age', 'SibSp', 'Parch', 'Fare']
['Name', 'Sex', 'Embarked']
```

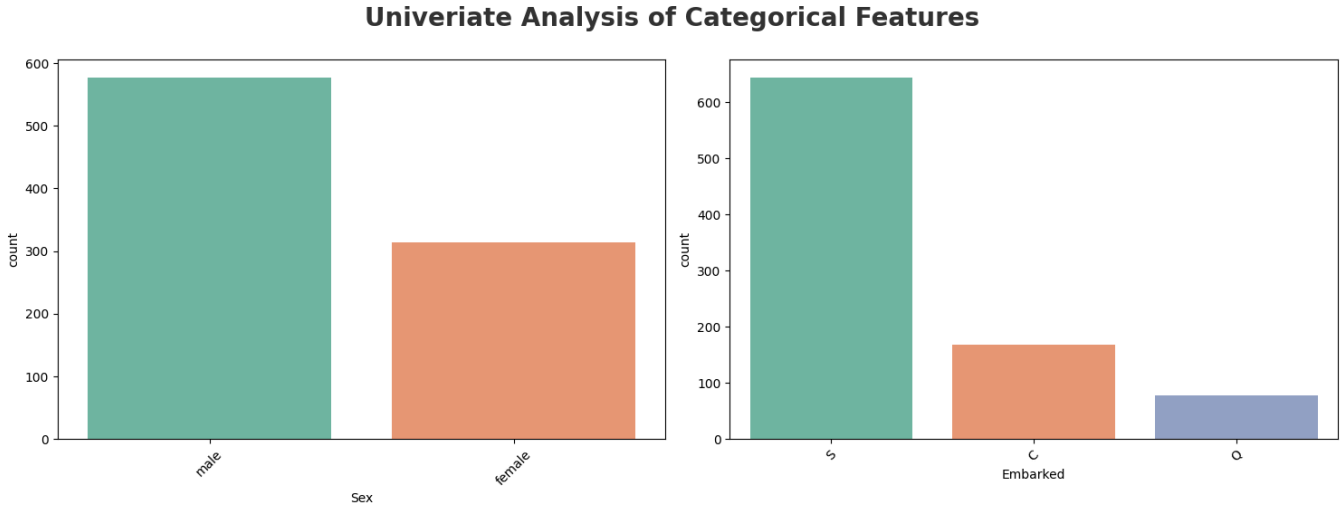
```
plt.figure(figsize=(15,15))
plt.suptitle('Univeriate Analysis of Numberic Features')
for i in range (0, len(numeric_feature)):
  plt.subplot(5,3,i+1)
  sns.kdeplot(x=df[numeric_feature[i]],shade=True,color='r')
  plt.xlabel(numeric_feature[i])
  plt.tight_layout()
```

```
<ipython-input-88-dc9797babe90>:5: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df[numeric_feature[i]],shade=True,color='r')
<ipython-input-88-dc9797babe90>:5: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df[numeric_feature[i]],shade=True,color='r')
<ipython-input-88-dc9797babe90>:5: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df[numeric_feature[i]],shade=True,color='r')
<ipython-input-88-dc9797babe90>:5: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df[numeric_feature[i]],shade=True,color='r')
<ipython-input-88-dc9797babe90>:5: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df[numeric_feature[i]],shade=True,color='r')
<ipython-input-88-dc9797babe90>:5: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df[numeric_feature[i]],shade=True,color='r')
```

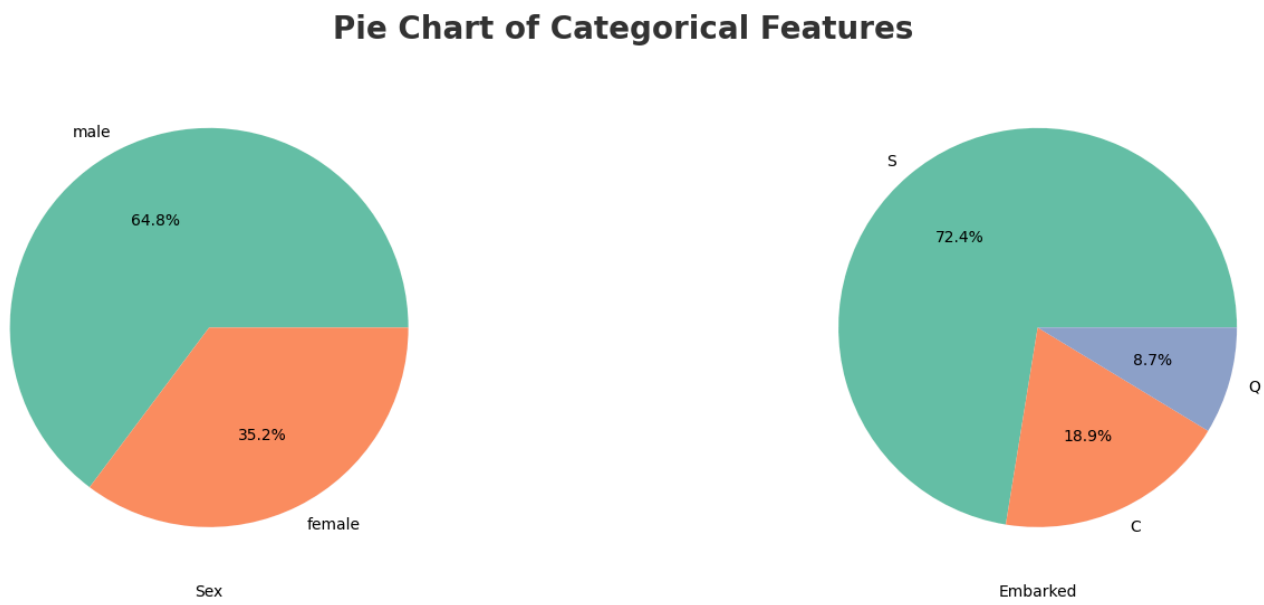


```
plt.figure(figsize=(15,10))
plt.suptitle('Univeriate Analysis of Categorical Features',fontsize=20,fontweight='bold',alpha=0.8,y=1.)
cat=['Sex', 'Embarked']
for i in range (0, len(cat)):
  plt.subplot(2,2,i+1)
  sns.countplot(x=df[cat[i]],palette="Set2")
  plt.xlabel(cat[i])
  plt.xticks(rotation=45)
  plt.tight_layout()
```

```
<ipython-input-89-36ba5c4b38d7>:6: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue`

  sns.countplot(x=df[cat[i]],palette="Set2")
<ipython-input-89-36ba5c4b38d7>:6: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue`

  sns.countplot(x=df[cat[i]],palette="Set2")
```



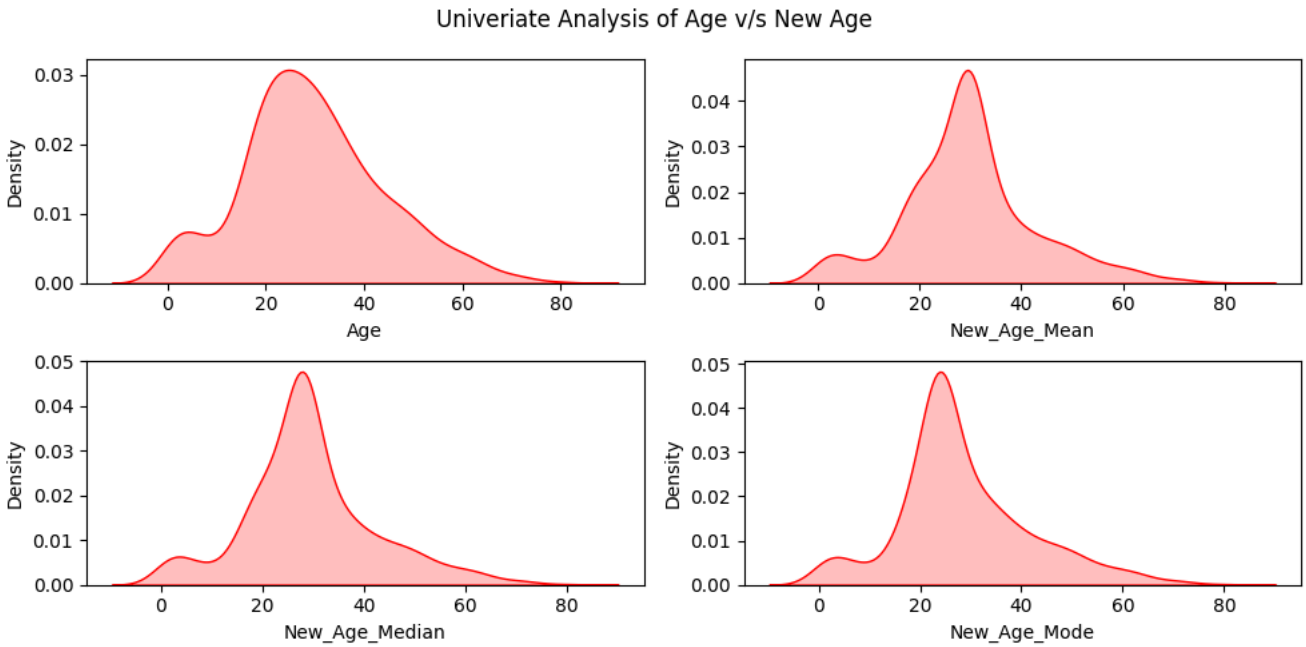

```
plt.figure(figsize=(15,10))
plt.suptitle('Pie Chart of Categorical Features',fontsize=20,fontweight='bold',alpha=0.8,y=1.)
cat=['Sex', 'Embarked']
for i in range (0, len(cat)):
  plt.subplot(2,2,i+1)
  counts = df[cat[i]].value_counts()
  plt.pie(counts, labels=counts.index, autopct='%1.1f%%', colors=sns.color_palette("Set2", len(counts)))
  plt.xlabel(cat[i])
  plt.xticks(rotation=45)
  plt.tight_layout()
```



```
df['New_Age_Mean']=df['Age'].fillna(df['Age'].mean())
df['New_Age_Median']=df['Age'].fillna(df['Age'].median())
df['New_Age_Mode']=df['Age'].fillna(df['Age'].mode()[0])
```

```
plt.figure(figsize=(10,5))
plt.suptitle('Univeriate Analysis of Age v/s New Age')

plt.subplot(2,2,1)
sns.kdeplot(x=df['Age'],shade=True,color='r')
plt.xlabel("Age")
plt.tight_layout()

plt.subplot(2,2,2)
sns.kdeplot(x=df['New_Age_Mean'],shade=True,color='r')
plt.xlabel("New_Age_Mean")
plt.tight_layout()

plt.subplot(2,2,3)
sns.kdeplot(x=df['New_Age_Median'],shade=True,color='r')
plt.xlabel("New_Age_Median")
plt.tight_layout()

plt.subplot(2,2,4)
sns.kdeplot(x=df['New_Age_Mode'],shade=True,color='r')
plt.xlabel("New_Age_Mode")
plt.tight_layout()
```

```
<ipython-input-92-29b6aba7c4fc>:5: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df['Age'],shade=True,color='r')
<ipython-input-92-29b6aba7c4fc>:10: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df['New_Age_Mean'],shade=True,color='r')
<ipython-input-92-29b6aba7c4fc>:15: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df['New_Age_Median'],shade=True,color='r')
<ipython-input-92-29b6aba7c4fc>:20: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(x=df['New_Age_Mode'],shade=True,color='r')
```




```
df['Embarked'].unique()
```

```
array(['S', 'C', 'Q', nan], dtype=object)
```

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

Embarked	count
S	644
C	168
Q	77

dtype: int64

```
Start coding or generate with AI.
```

```
df['Embarked']=df['Embarked'].fillna(df['Embarked'].mode()[0])
```

```
df['Embarked'].unique()
```

```
array(['S', 'C', 'Q'], dtype=object)
```

```
df.drop('Age',axis=1,inplace=True)
```

```
df.isnull().sum()
```

	0
Survived	0
Pclass	0
Name	0
Sex	0
SibSp	0
Parch	0
Fare	0
Embarked	0
New_Age_Mean	0
New_Age_Median	0
New_Age_Mode	0

dtype: int64

```
df.drop('Name',axis=1,inplace=True)
```

```
df.head()
```

	Survived	Pclass	Sex	SibSp	Parch	Fare	Embarked	New_Age_Mean	New_Age_Median	New_Age_Mode
0	0	3	male	1	0	7.2500	S	22.0	22.0	22.0
1	1	1	female	1	0	71.2833	C	38.0	38.0	38.0
2	1	3	female	0	0	7.9250	S	26.0	26.0	26.0
3	1	1	female	1	0	53.1000	S	35.0	35.0	35.0
4	0	3	male	0	0	8.0500	S	35.0	35.0	35.0

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
df['family_Size']=df['SibSp']+df['Parch']
```

```
df.drop(['SibSp','Parch'],axis=1,inplace =True)
```

```
df['family_Size']=df['family_Size'].astype(int)
```

```
df.head()
```

	Survived	Pclass	Sex	Fare	Embarked	New_Age_Mean	New_Age_Median	New_Age_Mode	family_Size
0	0	3	male	7.2500	S	22.0	22.0	22.0	1
1	1	1	female	71.2833	C	38.0	38.0	38.0	1
2	1	3	female	7.9250	S	26.0	26.0	26.0	0
3	1	1	female	53.1000	S	35.0	35.0	35.0	1
4	0	3	male	8.0500	S	35.0	35.0	35.0	0

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
from sklearn.preprocessing import OneHotEncoder
```

```
encoder=OneHotEncoder()
```

```
df1=pd.DataFrame(encoder.fit_transform(df[['Sex','Embarked']]).toarray(),columns=encoder.get_feature_names_out())
```

```
df.head()
```

	Survived	Pclass	Sex	Fare	Embarked	New_Age_Mean	New_Age_Median	New_Age_Mode	family_Size
0	0	3	male	7.2500	S	22.0	22.0	22.0	1
1	1	1	female	71.2833	C	38.0	38.0	38.0	1
2	1	3	female	7.9250	S	26.0	26.0	26.0	0
3	1	1	female	53.1000	S	35.0	35.0	35.0	1
4	0	3	male	8.0500	S	35.0	35.0	35.0	0

Next steps: Generate code with df | View recommended plots | New interactive sheet

```
new_df=pd.concat([df1, df], axis=1)
```

```
new_df.head()
```

	Sex_female	Sex_male	Embarked_C	Embarked_Q	Embarked_S	Survived	Pclass	Sex	Fare	Embarked	New_Age_Mean	New_A
0	0.0	1.0	0.0	0.0	1.0	0	3	male	7.2500	S	22.0	
1	1.0	0.0	1.0	0.0	0.0	1	1	female	71.2833	C	38.0	
2	1.0	0.0	0.0	0.0	1.0	1	3	female	7.9250	S	26.0	
3	1.0	0.0	0.0	0.0	1.0	1	1	female	53.1000	S	35.0	
4	0.0	1.0	0.0	0.0	1.0	0	3	male	8.0500	S	35.0	

Next steps: Generate code with new_df | View recommended plots | New interactive sheet

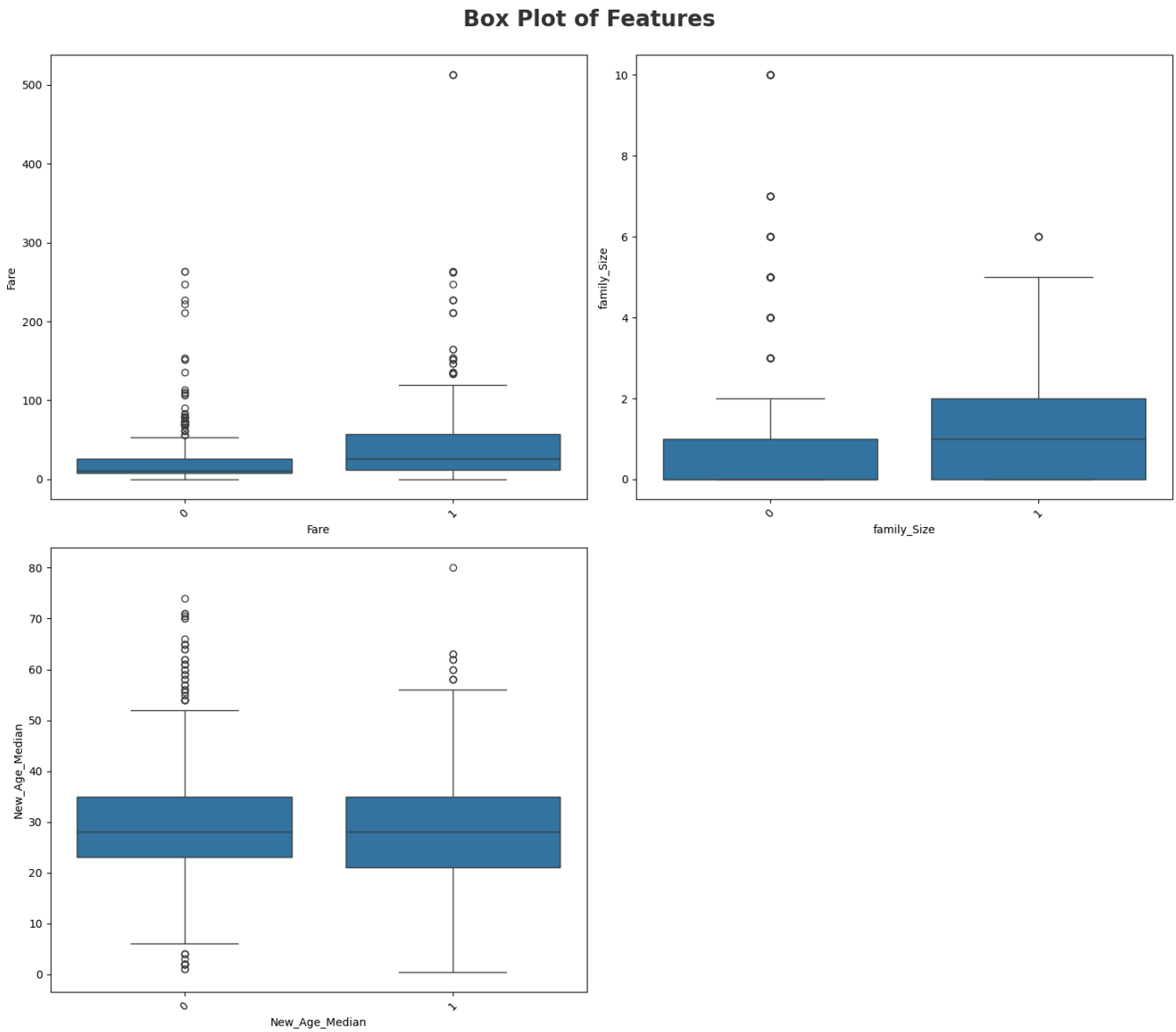
```
new_df.drop(['Sex','Embarked','New_Age_Mean','New_Age_Mode'],axis=1,inplace=True)
```

```
new_df.head()
```

	Sex_female	Sex_male	Embarked_C	Embarked_Q	Embarked_S	Survived	Pclass	Fare	New_Age_Median	family_Size
0	0.0	1.0	0.0	0.0	1.0	0	3	7.2500	22.0	1
1	1.0	0.0	1.0	0.0	0.0	1	1	71.2833	38.0	1
2	1.0	0.0	0.0	0.0	1.0	1	3	7.9250	26.0	0
3	1.0	0.0	0.0	0.0	1.0	1	1	53.1000	35.0	1
4	0.0	1.0	0.0	0.0	1.0	0	3	8.0500	35.0	0

Next steps: Generate code with new_df | View recommended plots | New interactive sheet

```
plt.figure(figsize=(15,13))
plt.suptitle('Box Plot of Features',fontsize=20,fontweight='bold',alpha=0.8,y=1.)
cat=['Fare','family_Size','New_Age_Median']
for i in range (0, len(cat)):
  plt.subplot(2,2,i+1)
  sns.boxplot(x='Survived',y=cat[i],data=new_df)
  plt.xlabel(cat[i])
  plt.xticks(rotation=45)
  plt.tight_layout()
```



```
corr=new_df[new_df.columns].corr()
corr
```

	Sex_female	Sex_male	Embarked_C	Embarked_Q	Embarked_S	Survived	Pclass	Fare	New_Age_Median	fam:
Sex_female	1.000000	-1.000000	0.082853	0.074115	-0.119224	0.543351	-0.131900	0.182333	-0.081163	
Sex_male	-1.000000	1.000000	-0.082853	-0.074115	0.119224	-0.543351	0.131900	-0.182333	0.081163	
Embarked_C	0.082853	-0.082853	1.000000	-0.148258	-0.782742	0.168240	-0.243292	0.269335	0.030248	
Embarked_Q	0.074115	-0.074115	-0.148258	1.000000	-0.499421	0.003650	0.221009	-0.117216	-0.031415	
Embarked_S	-0.119224	0.119224	-0.782742	-0.499421	1.000000	-0.149683	0.074053	-0.162184	-0.006729	
Survived	0.543351	-0.543351	0.168240	0.003650	-0.149683	1.000000	-0.338481	0.257307	-0.064910	
Pclass	-0.131900	0.131900	-0.243292	0.221009	0.074053	-0.338481	1.000000	-0.549500	-0.339898	
Fare	0.182333	-0.182333	0.269335	-0.117216	-0.162184	0.257307	-0.549500	1.000000	0.096688	
New_Age_Median	-0.081163	0.081163	0.030248	-0.031415	-0.006729	-0.064910	-0.339898	0.096688	1.000000	
family_Size	0.200988	-0.200988	-0.046215	-0.058592	0.077359	0.016639	0.065997	0.217138	-0.245619	

Next steps: Generate code with corr | View recommended plots | New interactive sheet

```
plt.figure(figsize=(15,10))
sns.heatmap(corr,annot=True,cmap='coolwarm')
```

```
<Axes: >
```