Demo

April 28, 2025

[7]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[9]:
```
# Load the dataset directly from URL
url = 'https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic.
↪csv'
df = pd.read_csv(url)
df.head()
```

[9]:

	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
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

[11]:
```
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
```

[13]:
```
df.describe()
```

[13]:

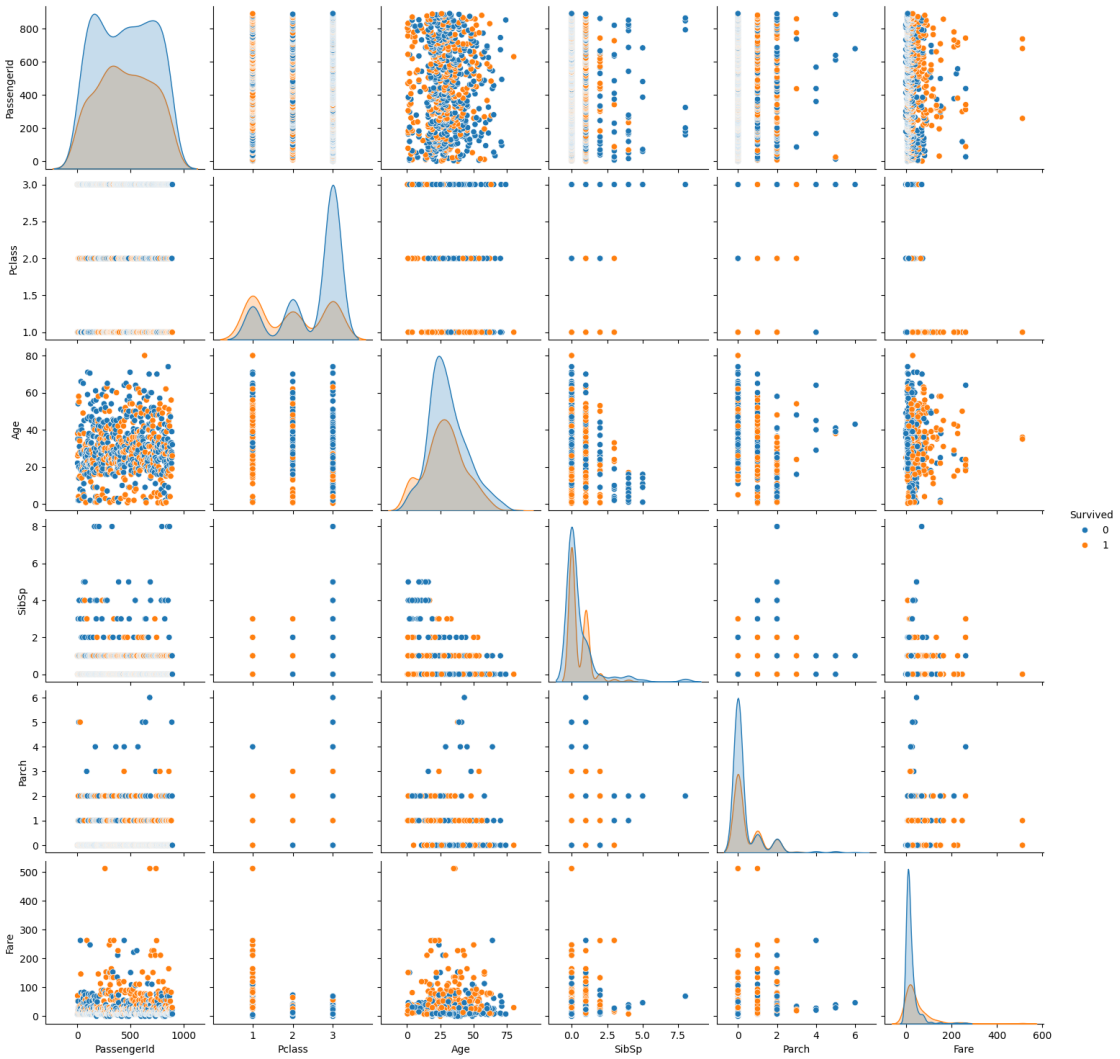
	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	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

[17]:
```
df['Age'].value_counts()
```

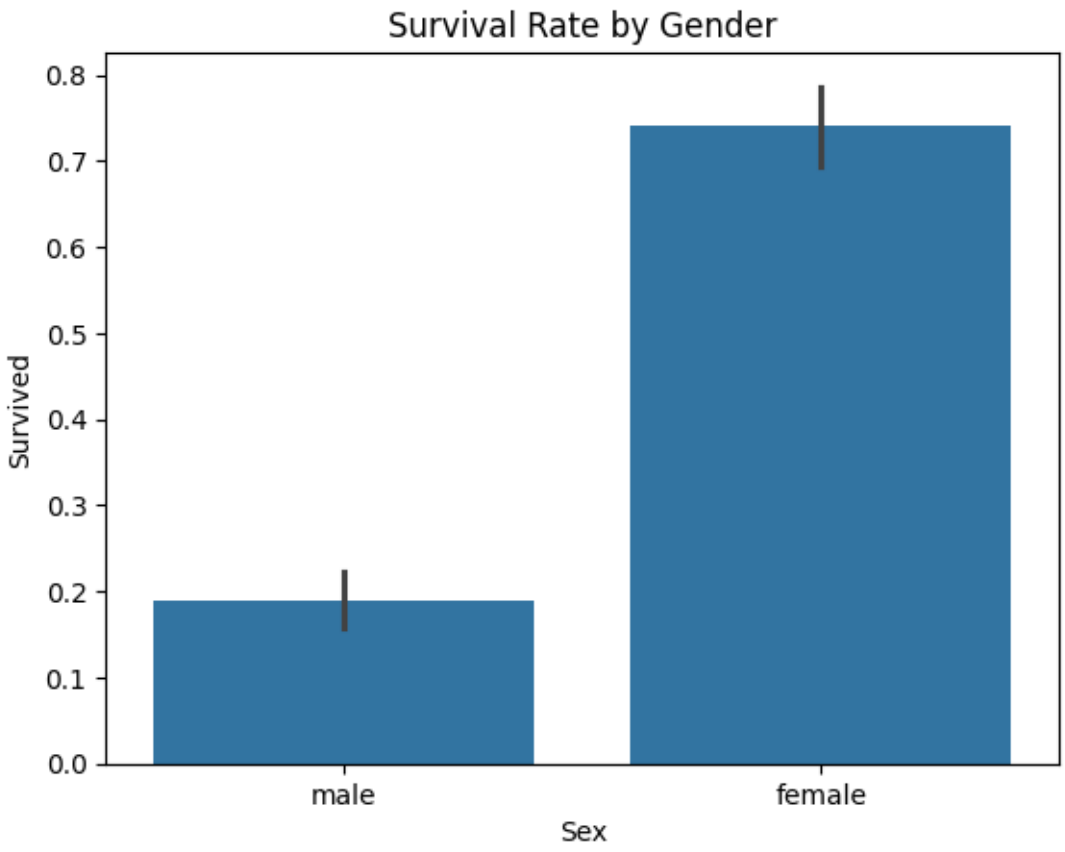
[17]:
```
Age
24.00    30
22.00    27
18.00    26
28.00    25
30.00    25
         ..
24.50     1
```

```
0.67      1
0.42      1
34.50     1
74.00     1
Name: count, Length: 88, dtype: int64
```

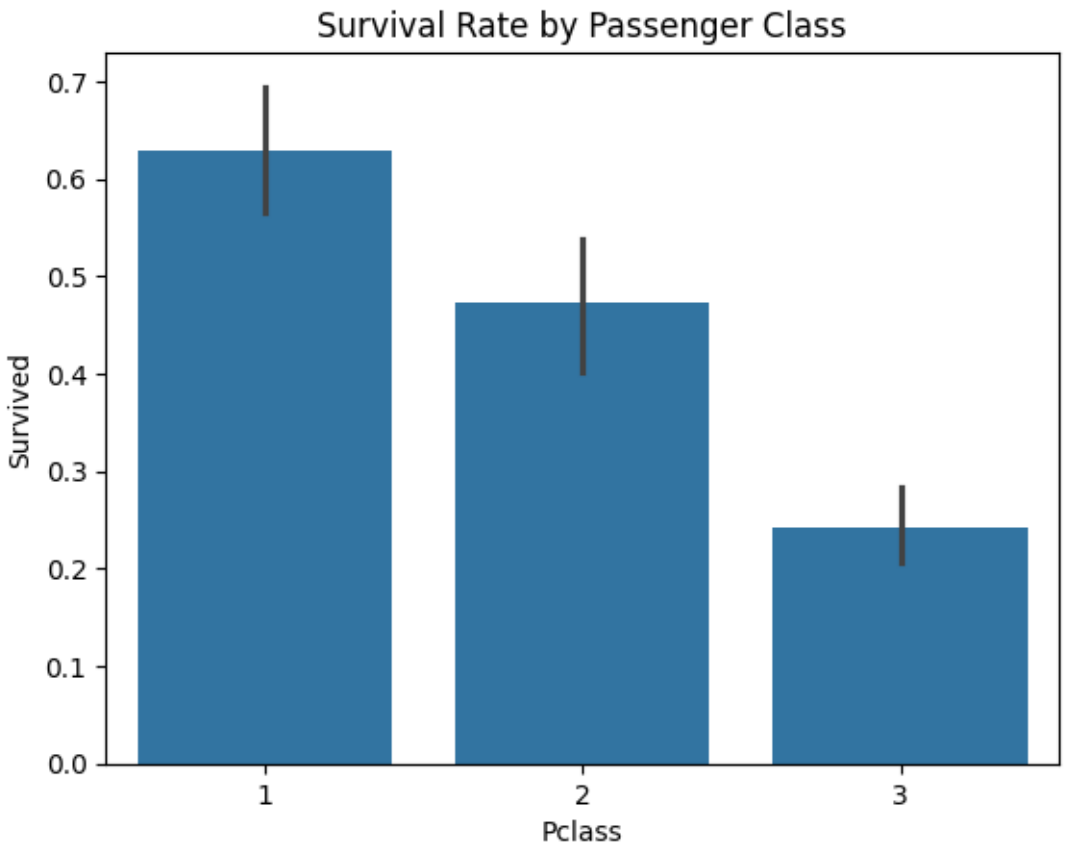
[23]:
```
# Pairplot
sns.pairplot(df,hue='Survived')
plt.show()
```



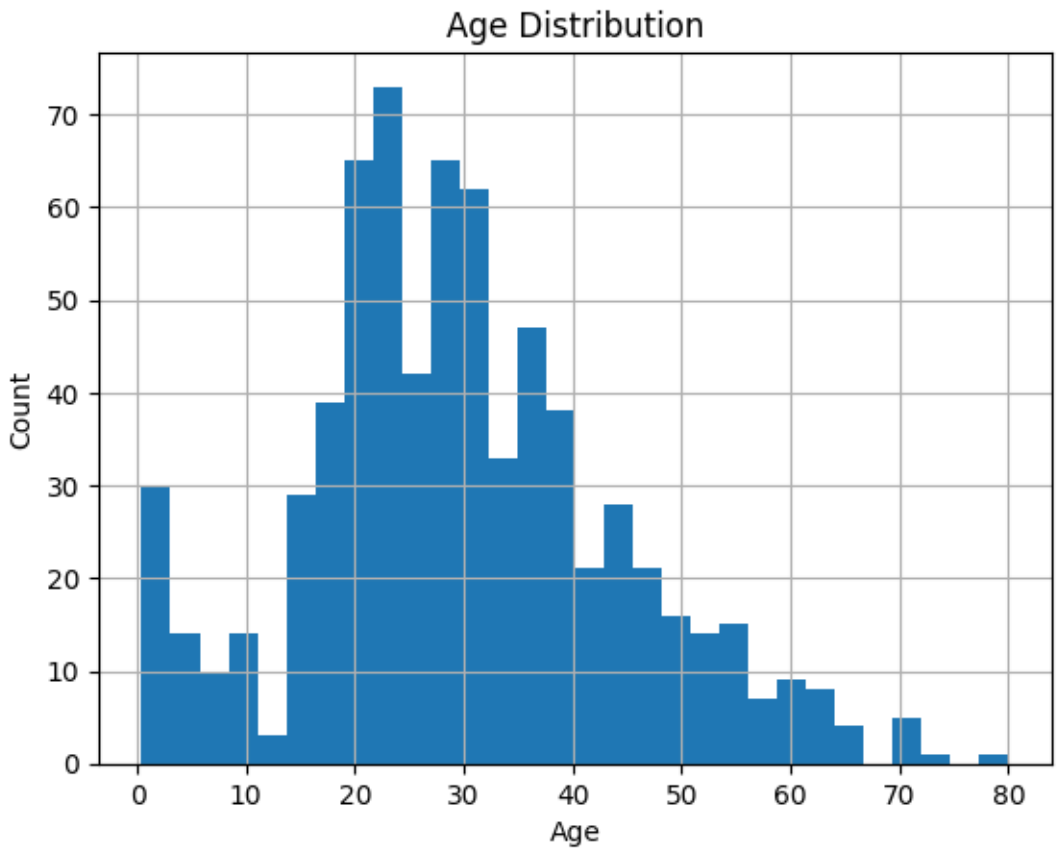
[31]:
```
# Survival Rate by gender
sns.barplot(x='Sex' ,y='Survived',data=df)
plt.title('Survival Rate by Gender')
plt.show()
```



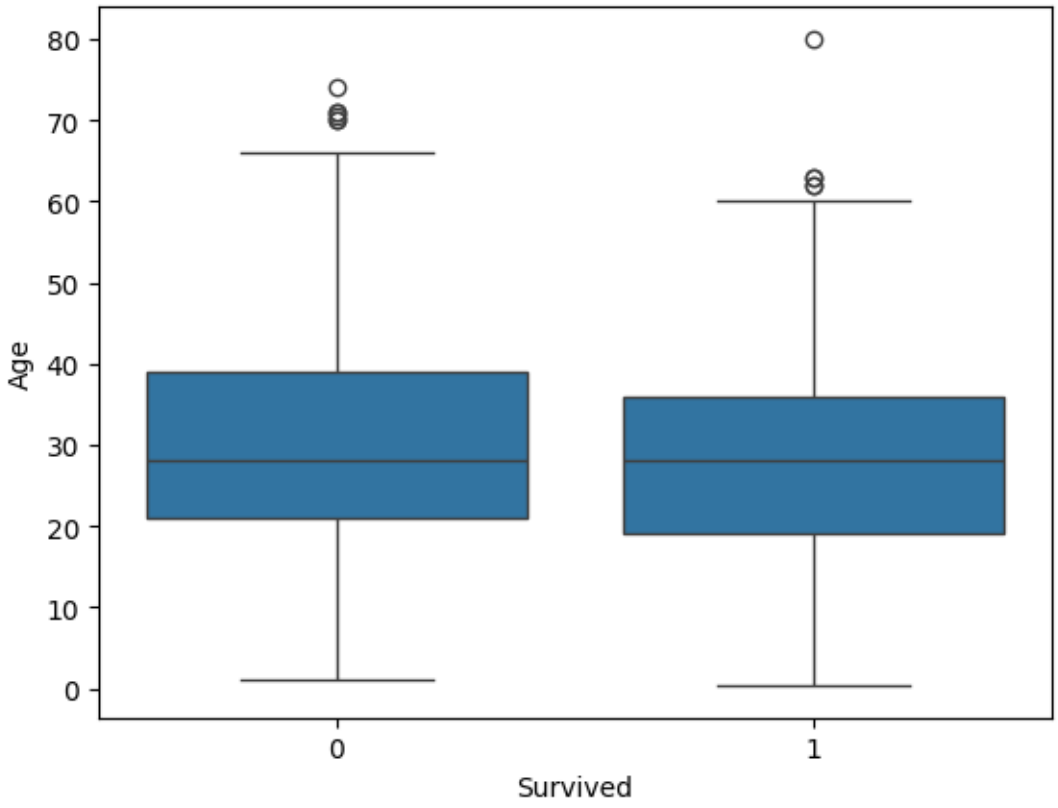
[33]:
```
# Survival Rate by Pclass
sns.barplot(x='Pclass' ,y='Survived',data=df)
plt.title('Survival Rate by Passenger Class')
plt.show()
```



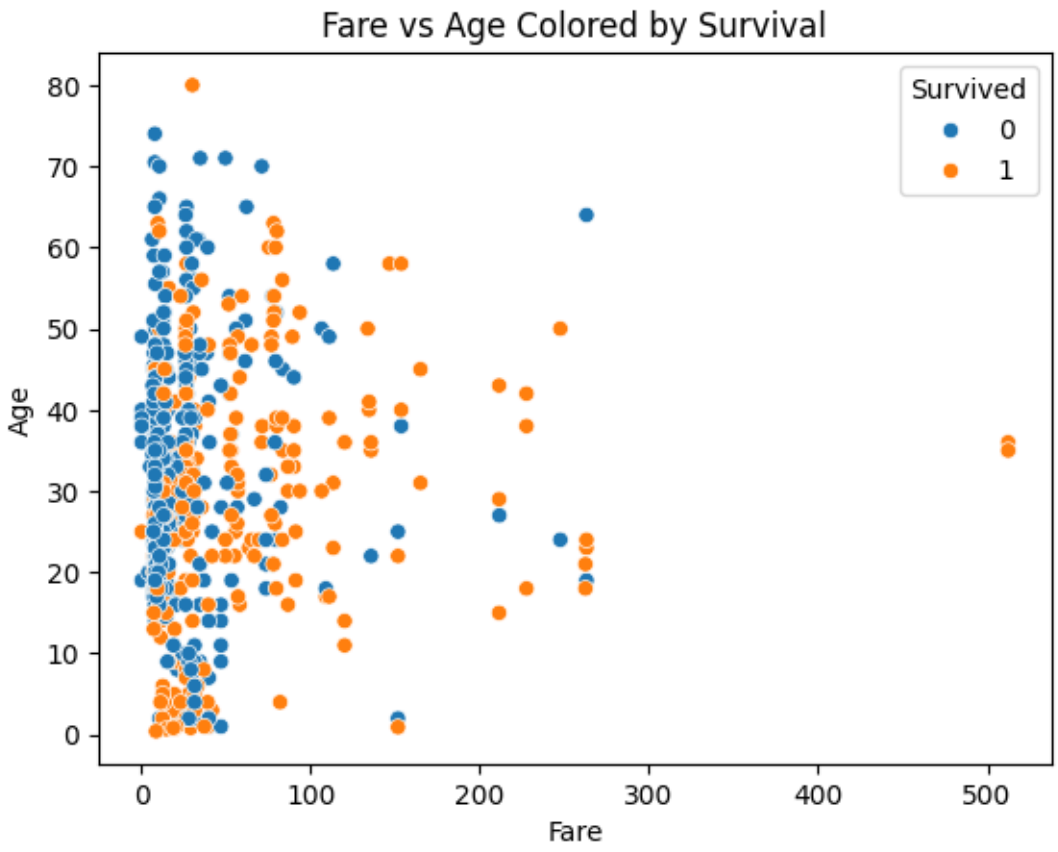
[35]:
```
# Histogram :Age Distribution
df['Age'].hist(bins=30)
plt.title('Age Distribution')
plt.xlabel('Age')
plt.ylabel('Count')
plt.show()
```



[45]:
```
# Box Plot : Age vs Survived
sns.boxplot(x='Survived',y='Age',data=df)
plt.show()
```



[43]:
```
# Scatter Plot: Fare vs Age
sns.scatterplot(x='Fare',y='Age',hue='Survived',data=df)
plt.title('Fare vs Age Colored by Survival')
plt.show()
```


Fare vs Age Colored by Survival
Survived
0
1
Age
Fare
0
10
20
30
40
50
60
70
80
0
100
200
300
400
500

[ ]: