20250525_01

May 26, 2025

[4]:
```
import seaborn as sns
data = sns.load_dataset("titanic")
```

[5]:
```
data.head()
```

[5]:
```
   survived  pclass     sex   age  sibsp  parch     fare embarked  class  \
0         0       3    male  22.0      1      0   7.2500        S  Third
1         1       1  female  38.0      1      0  71.2833        C  First
2         1       3  female  26.0      0      0   7.9250        S  Third
3         1       1  female  35.0      1      0  53.1000        S  First
4         0       3    male  35.0      0      0   8.0500        S  Third

     who  adult_male deck  embark_town alive  alone
0    man        True  NaN  Southampton    no  False
1  woman       False    C    Cherbourg   yes  False
2  woman       False  NaN  Southampton   yes   True
3  woman       False    C  Southampton   yes  False
4    man        True  NaN  Southampton    no   True
```

[6]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   survived     891 non-null    int64
 1   pclass       891 non-null    int64
 2   sex          891 non-null    object
 3   age          714 non-null    float64
 4   sibsp        891 non-null    int64
 5   parch        891 non-null    int64
 6   fare         891 non-null    float64
 7   embarked     889 non-null    object
 8   class        891 non-null    category
 9   who          891 non-null    object
 10  adult_male   891 non-null    bool
 11  deck         203 non-null    category
```

```
 12  embark_town  889 non-null    object
 13  alive        891 non-null    object
 14  alone        891 non-null    bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

[7]:
```
data.describe()
```

[7]:

	survived	pclass	age	sibsp	parch	fare
count	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

[8]:
```
data.isnull().sum()
```

[8]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

[9]:
```
data['age'] = data['age'].fillna(data['age'].median())
```

[10]:
```
data = data.drop(columns = ['deck'])
```

[12]:
```
data['embarked'] = data['embarked'].fillna(data['embarked'].mode()[0])
data['embark_town'] = data['embark_town'].fillna(data['embark_town'].mode()[0])
```

[13]:
```
data.isnull().sum()
```

[13]:
```
survived       0
pclass         0
```

```
sex            0
age            0
sibsp          0
parch          0
fare           0
embarked       0
class          0
who            0
adult_male     0
embark_town    0
alive          0
alone          0
dtype: int64
```

[18]:
```
data.groupby('sex')['survived'].mean()
```

[18]:
```
sex
female    0.742038
male      0.188908
Name: survived, dtype: float64
```

[19]:
```
data.groupby('pclass')['age'].mean()
```

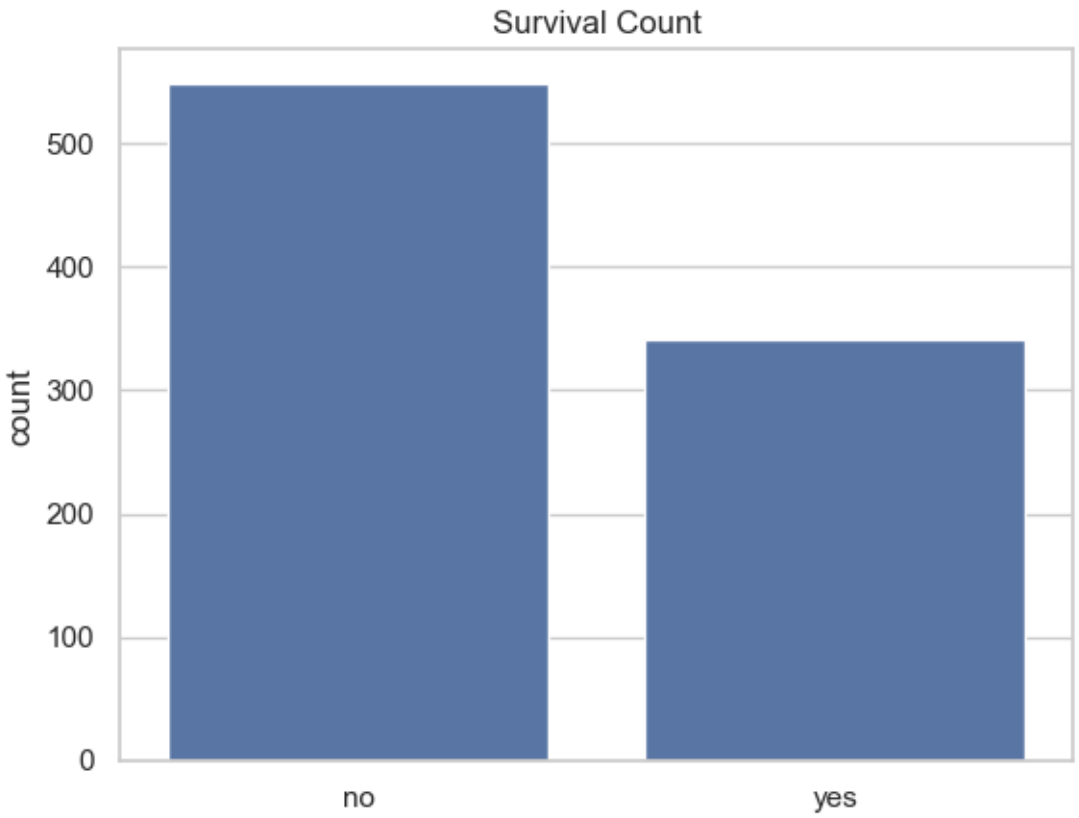
[19]:
```
pclass
1    36.812130
2    29.765380
3    25.932627
Name: age, dtype: float64
```

[21]:
```
data.groupby('embark_town')['survived'].mean()
```

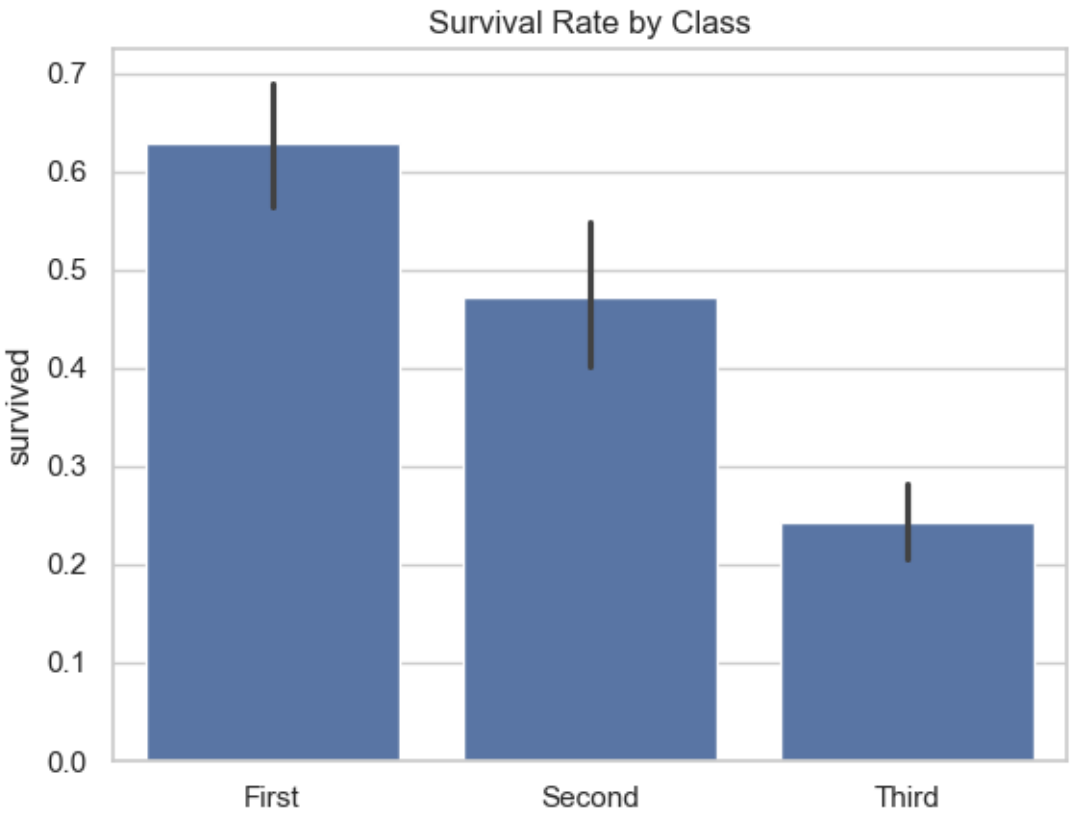
[21]:
```
embark_town
Cherbourg      0.553571
Queenstown     0.389610
Southampton    0.339009
Name: survived, dtype: float64
```

[24]:
```
import matplotlib.pyplot as plt
```

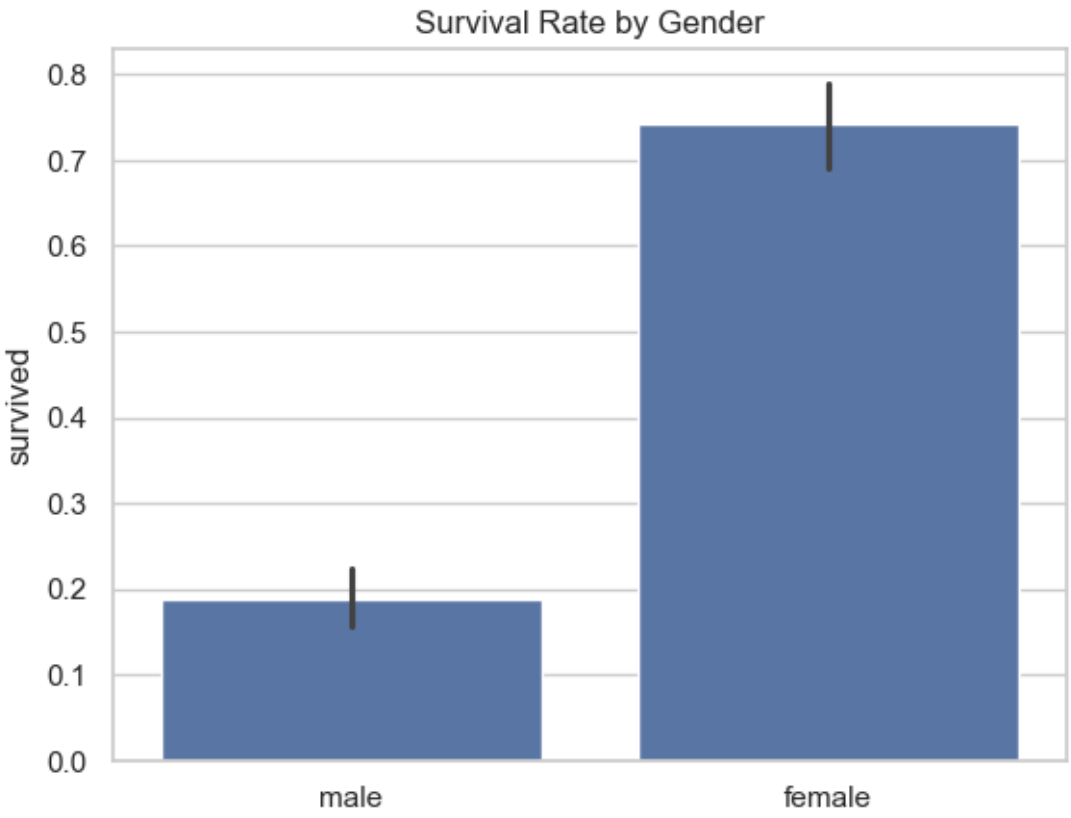
[30]:
```
sns.set(style = "whitegrid")

sns.countplot(x = 'alive', data = data)

plt.title("Survival Count")
plt.xlabel("")
plt.show()
```



[34]:
```
sns.barplot(x = 'class', y = 'survived', data = data)

plt.title("Survival Rate by Class")
plt.xlabel("")
plt.show()
```



[37]:
```
sns.barplot(x = 'sex', y = 'survived', data = data)

plt.title("Survival Rate by Gender")
plt.xlabel("")
plt.show()
```



0.1 Conclusion

Review how to check the data, deal with missing values and visualization.