Investigate a Dataset - Titanic

Questions

1. What's the ratio of the number of female survived to the number of male survived?
2. Did passengers in the first pclass tend to have higher possibility being survived?

Wrangle

In [7]:

```
import seaborn as sns
import pandas as pd

titanic_df=pd.read_csv('titanic-data.csv')
```

In [8]:

```
titanic_visual=pd.DataFrame(titanic_data)
titanic_visual['Survived'].replace({0:'Passengers_Died', 1:'Passengers_Survived'
}, inplace=True)
```

Explore how many uniques passengers in the dataset, how many passengers survived and how many passengers didn't survive

In [9]:

```
len(titanic_df['PassengerId'].unique())
```

Out[9]:

```
891
```

In [10]:

```
l=titanic_visual.groupby('Survived').count()['PassengerId']
print(l)
```

```
Survived
Passengers_Died        549
Passengers_Survived    342
Name: PassengerId, dtype: int64
```

In [14]:

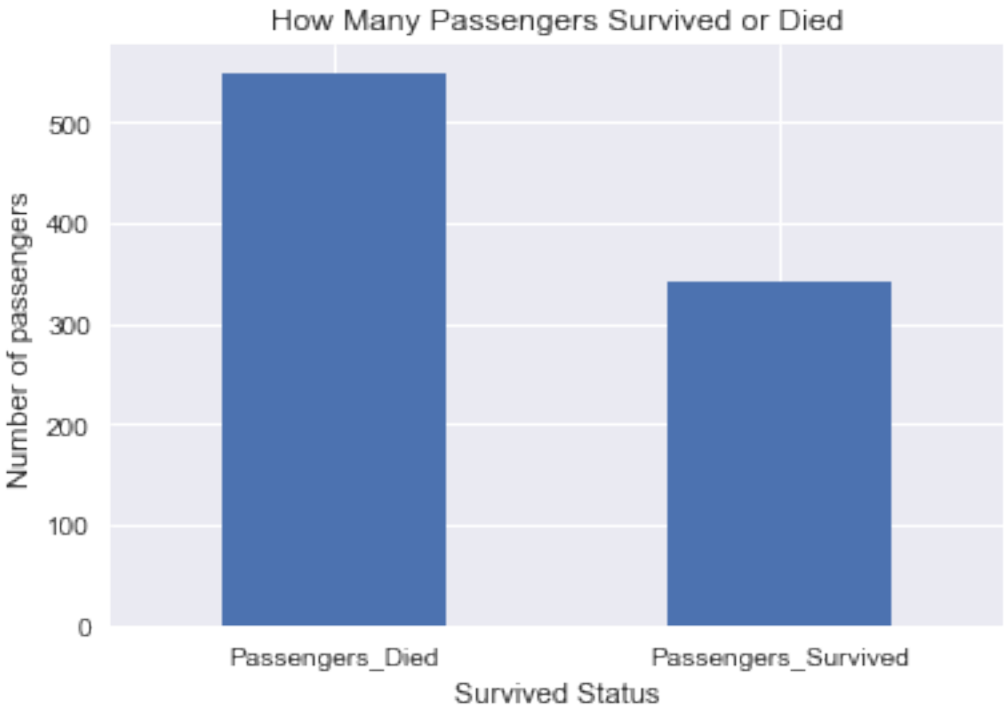
```
%matplotlib inline
import matplotlib.pyplot as plt
```

In [16]:

```
plt.title("How Many Passengers Survived or Died")
l.plot('bar',rot=0)
plt.xlabel("Survived Status")
plt.ylabel('Number of passengers')
```

Out[16]:

```
<matplotlib.text.Text at 0x11531c748>
```



It turned out that the number of died passengers is greater than the number of survived passengers. To be specific, there's 549 passengers died, and 342 passengers survived in this sample dataset.

Get a look at how data looks like

In [18]:

```
print(titanic_data.head())
```

```
   PassengerId             Survived  Pclass  \
0            1      Passengers_Died       3   
1            2  Passengers_Survived       1   
2            3  Passengers_Survived       3   
3            4  Passengers_Survived       1   
4            5      Passengers_Died       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   

   Parch            Ticket     Fare Cabin Embarked  
0      0         A/5 21171   7.2500   NaN        S  
1      0          PC 17599  71.2833   C85        C  
2      0  STON/O2. 3101282   7.9250   NaN        S  
3      0            113803  53.1000  C123        S  
4      0            373450   8.0500   NaN        S  
```

Deal with missing values

In [19]:

```
titanic_data.isnull().sum()
```

Out[19]:

```
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
```

From above results, I know that the dataset is missing 177 "Age" variables, 687 "Cabin" variables, and 2 "Embarked' variables. The "Age" variables is crucial in the analysis. Therefore, when analyzing questions related with "Age" variable, it's better to exclude the data without age values. The following codes will exclude the data missing age values:

In [20]:

```
valid_ages = titanic_data.dropna(subset=["Age"])
valid_ages.count()[['Age','Survived']]
```

Out[20]:

```
Age         714
Survived    714
dtype: int64
```

In [21]:

```
valid_ages.isnull().sum()[['Age','Survived']]
```

Out[21]:

```
Age         0
Survived    0
dtype: int64
```

When analyzing questions related to "Age" variable, it's better to use "valid_ages" dataframe. According to my listed questions, I will not analyze "Age" related questions. So it's better not to exclude these data.

Considering my questions, I think I should split data into two groups: passengers who survived and who didn't survive:

In [22]:

```
grouped_passengers=titanic_df.groupby('Survived').groups
print(grouped_passengers)
```

```
{0: Int64Index([  0,   4,   5,   6,   7,  12,  13,  14,  16,  18,
            ...
            877, 878, 881, 882, 883, 884, 885, 886, 888, 890],
           dtype='int64', length=549), 1: Int64Index([  1,   2,   3,
8,   9,  10,  11,  15,  17,  19,
            ...
            865, 866, 869, 871, 874, 875, 879, 880, 887, 889],
           dtype='int64', length=342)}
```

Explore

Question 1: What's the ratio of the number of survived female to the number of survived male?

First I grouped dataframe by "Sex" and "Survived" variables using following codes:

In [27]:

```
titanic_visual.groupby(['Sex','Survived'])
```

Out[27]:

```
<pandas.core.groupby.DataFrameGroupBy object at 0x1154392b0>
```

In [28]:

```
male_survived_df=titanic_visual.groupby(['Sex','Survived']).get_group(('male','P
assengers_Survived'))
female_survived_df=titanic_visual.groupby(['Sex','Survived']).get_group(('female
','Passengers_Survived'))
```

*I calculated the number of survived female passengers and male passengers. Then I divided the number of survived female passengers by the number of survived male passengers to calculate the ratio.**

In [29]:

```
num_female_survived=female_survived_df['Survived'].count()
num_male_survived=male_survived_df['Survived'].count()
ratio_survived_sex=num_female_survived / num_male_survived

print(ratio_survived_sex)
```

```
2.1376146789
```

Ratio of Survived Females Passengers to Survived Male Passengers :

In [30]:

```
from fractions import Fraction

Fraction(ratio_survived_sex).limit_denominator()
```

Out[30]:

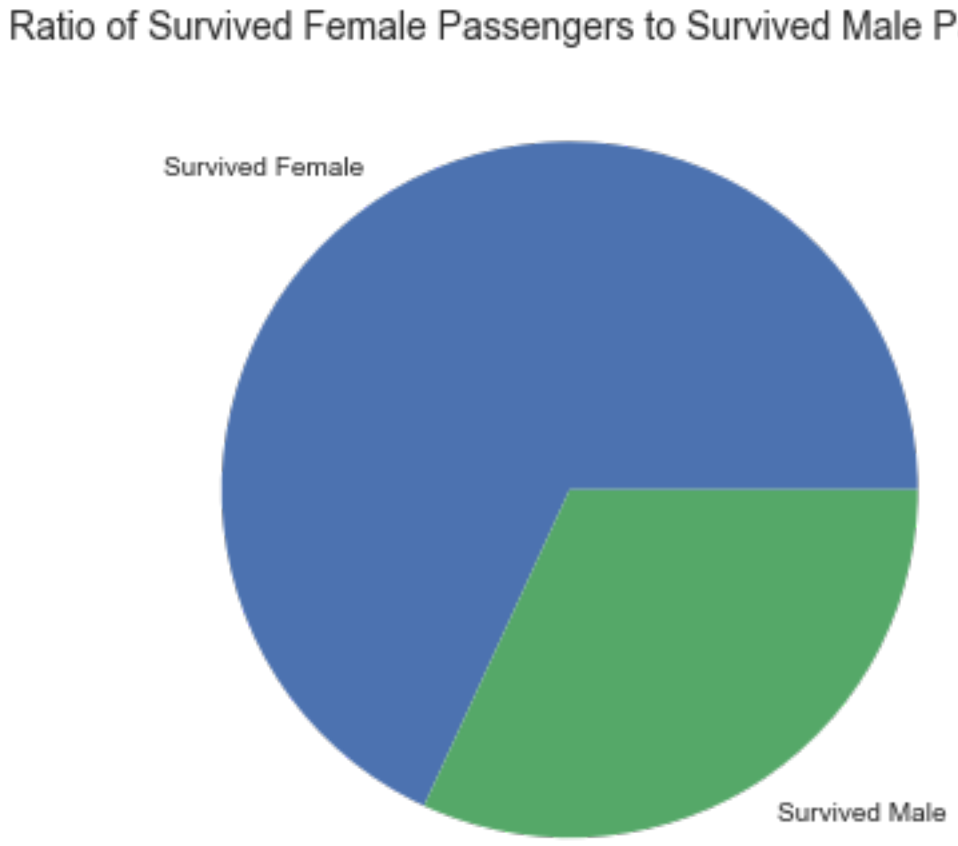
```
Fraction(233, 109)
```

In [31]:

```
import numpy as np
data = [233, 109]
plt.figure(num=1, figsize=(6, 6))
plt.axes(aspect=1)
plt.title('Ratio of Survived Female Passengers to Survived Male Passengers', siz
e=14)
plt.pie(data, labels=('Survived Female', 'Survived Male'))
```

Out[31]:

```
([<matplotlib.patches.Wedge at 0x1155eeba8>,
  <matplotlib.patches.Wedge at 0x1155f7860>],
 [<matplotlib.text.Text at 0x1072a6e48>,
  <matplotlib.text.Text at 0x1155f7fd0>])
```



Ratio of Survived Female Passengers to Survived Male Passengers

From this pie chart, it's earsier to see the part representing the number of survived female is bigger than the part representing the number of survived male. According to above result, the ratio of the number of survived female passenger to the number of survived male passengers is 233 to 109.

Number of Survived Female and Survived Male Passengers

In [32]:

```
print(num_female_survived,num_male_survived)
```

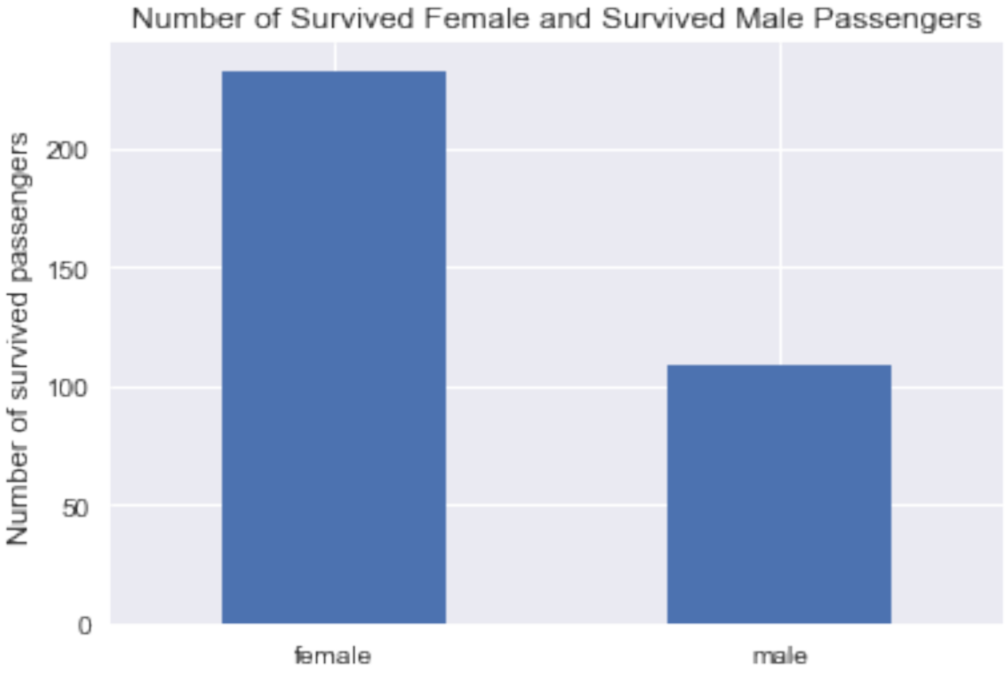
```
233 109
```

In [33]:

```
import pandas as pd

num_survived=pd.Series((num_female_survived,num_male_survived),index=['female','
male'])


plt.title('Number of Survived Female and Survived Male Passengers')
num_survived.plot('bar',rot=0)

plt.ylabel("Number of survived passengers")
```

Out[33]:

```
<matplotlib.text.Text at 0x115605668>
```



From the above bar chart, we can clearly see that the number of survived female passengers (239) is way more than the number of survived male passengers (109).

Question 2: Did passengers in the first pclass tend to have higher possibility being survived?

Frequency test

I want to calculated the percentages of survived passengers in each pclass. So first, I grouped data by "Pclass" and "Survived" variables:

```
In [34]:
```

```
titanic_df.groupby(['Pclass','Survived'])
```

```
Out[34]:

<pandas.core.groupby.DataFrameGroupBy object at 0x115444ac8>
```

For convenience, I replaced the "Survived" variables' values back to "0" and "1":

```
In [36]:
```

```
titanic_visual['Survived'].replace({'Passengers_Died':0, 'Passengers_Survived':1
}, inplace=True)
```

Second, I counted survived passengers in the first pclass and in other pclasses by suming "Survived" variables in the grouped data. Then, divided by the number of passengers in that pclass:

```
In [37]:
```

```
pclass1_survived=titanic_df.groupby(['Pclass']).get_group(1).sum()['Survived']
pclass1_passengers=titanic_df.groupby(['Pclass']).get_group(1).count()['Survived
']

p1=pclass1_survived / pclass1_passengers
print(p1)
```

```
0.62962962963
```

```
In [38]:
```

```
pclass2_survived=titanic_df.groupby(['Pclass']).get_group(2).sum()['Survived']
pclass2_passengers=titanic_df.groupby(['Pclass']).get_group(2).count()['Survived
']

p2=pclass2_survived / pclass2_passengers
print(p2)
```

```
0.472826086957
```

In [39]:

```
pclass3_survived=titanic_df.groupby(['Pclass']).get_group(3).sum()['Survived']
pclass3_passengers=titanic_df.groupby(['Pclass']).get_group(3).count()['Survived']

p3=pclass3_survived / pclass3_passengers
print(p3)
```

```
0.242362525458
```

In [40]:

```
p_others=(pclass2_survived+pclass3_survived)/(pclass3_passengers+pclass3_passengers)
```

Percentages of Survived Passengers in the First Pclass and in Other Pclasses

So now we got the percentages of survived passengers in each pclass:

In [41]:

```
print(' first class', ' ','other classes')
print(p1,p_others)
```

```
 first class   other classes
0.62962962963 0.209775967413
```

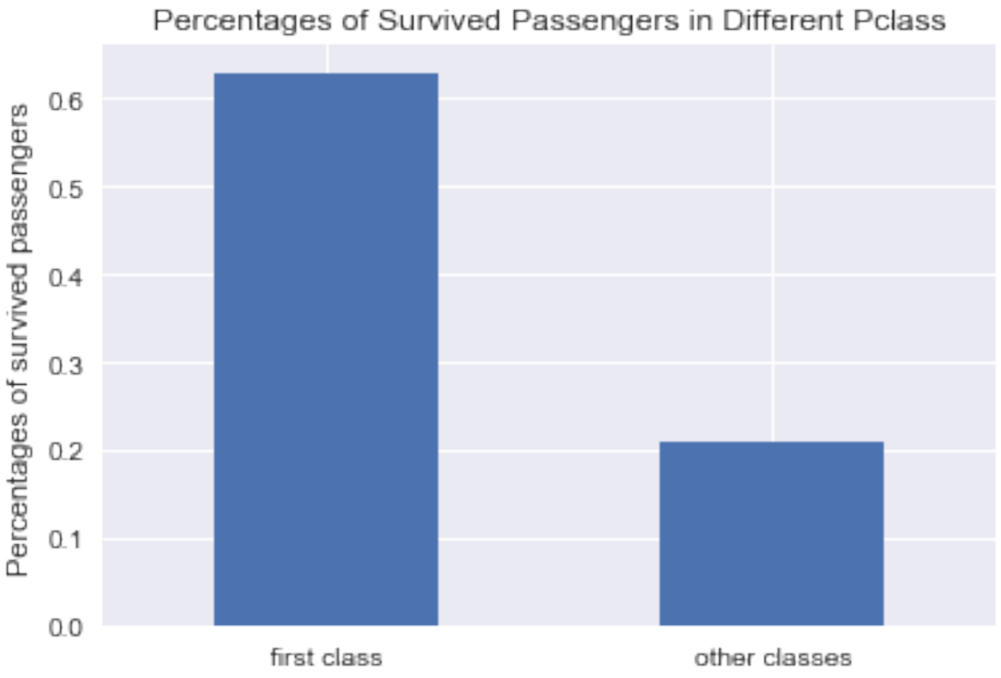
The following bar chart would bring a clearer result:

In [42]:

```
b=pd.Series([p1,p_others],index=['first class','other classes'])

b.plot('bar',rot=0,title="Percentages of Survived Passengers in Different Pclass
")
plt.ylabel("Percentages of survived passengers")
```

Out[42]:

```
<matplotlib.text.Text at 0x1157f2048>
```



From this bar chart, It turned out that passengers in the first pclass tend to have higher possibility being survived. The percentage of passengers being survived in the first pclass is around 0.63, while the percentage of passengers being survived in other pclasses is 0.21.

Chi-square test

Now I want to add statistical test to further support my above finding. My null hypothesis is that the possibility being survived are equal between the first pclass and other pclasses.

According to the null hypothesis, the expected number of survived passengers for each pclass should be the overall survival rate times passenger number in each pclass:

In [43]:

```
#expected group:#
expected_rate=titanic_df['Survived'].mean()
titanic_df.groupby(['Pclass']).count()
```

Out[43]:

	PassengerId	Survived	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Em
Pclass											
1	216	216	216	216	186	216	216	216	216	176	21
2	184	184	184	184	173	184	184	184	184	16	18
3	491	491	491	491	355	491	491	491	491	12	49

In [44]:

```
#expected group:#
p1_expected_survived=titanic_df.groupby(['Pclass']).get_group(1).count()['Passen
gerId']*expected_rate
p2_expected_survived=titanic_df.groupby(['Pclass']).get_group(2).count()['Passen
gerId']*expected_rate
p3_expected_survived=titanic_df.groupby(['Pclass']).get_group(3).count()['Passen
gerId']*expected_rate
```

The observed outcomes are from the actual number of survived passengers in each pclass:

In [45]:

```
#observed group:#
titanic_df.groupby(['Pclass']).sum()
```

Out[45]:

	PassengerId	Survived	Age	SibSp	Parch	Fare
Pclass						
1	99705	136	7111.42	90	77	18177.4125
2	82056	87	5168.83	74	70	3801.8417
3	215625	119	8924.92	302	193	6714.6951

In [46]:

```
#observed group:#
p1_survived=titanic_df.groupby(['Pclass']).get_group(1).sum()['Survived']
p2_survived=titanic_df.groupby(['Pclass']).get_group(2).sum()['Survived']
p3_survived=titanic_df.groupby(['Pclass']).get_group(3).sum()['Survived']
```

I'm focusing on passengers in the first pclass and in other pclasses, so I combined passengers number in the second and third pclass:

In [47]:

```
pothers_survived=p2_survived+p3_survived
pothers_expected_survived=p2_expected_survived+p3_expected_survived
```

Calculating p-value using chi_square:

In [48]:

```
from scipy.stats import chisquare
chisquare([p1_survived,pothers_survived],f_exp=[p1_expected_survived,pothers_exp
ected_survived])
```

Out[48]:

```
Power_divergenceResult(statistic=44.875789473684208, pvalue=2.099375
8249200262e-11)
```

From the pvalue(which is way more smaller than 0.05), we can reject the null hypothesis. According to this test and the previous frequency test, I found out that passengers in the first pclass tend to have a higher possibility being survived.

Draw Conclusions

Tentative conclusion:

I investigated the Titanic dataset using pandas, focusing on several variables in the dataset, such as 'survived', 'sex', 'pclass', and 'parch' variables. Also, I used different plot visualizations to better support my analysis. When wrangling data, I fountd there's missing values in 'Age' variable. Therefore, when analyzing questions related to "Age" variable, it's better to exclude these data which are missing 'Age' values.

Firstly, according to my first question, through my computation, pie charts and bar chart, I found that the number of survived female passengers is way more than the number of survived male passengers. The ratio of these two numbers is 233:109.

Secondly, I analyzed the 'pclass'and the 'survived' variables, using both frequency test and statitical test(chi_square test). The reason why I think passengers in the first pclasses might have different rates of survival is that wealthy people might be put on lifeboats first. Or maybe it's because of where their cabins were on the boat. It turned out that passengers in the first pclass actually tend to have a higher possibility being survived.

Although, there's might be some limitations of the data. Because we don't know how representative this sample is. It's better to make sure the sample we analyzed is representative enough to make such conclusions.

Reference:

https://discussions.udacity.com/t/removing-missing-values/299283/2 (https://discussions.udacity.com/t/removing-missing-values/299283/2)

https://stackoverflow.com/questions/32244019/how-to-rotate-x-axis-tick-labels-in-pandas-barplot (https://stackoverflow.com/questions/32244019/how-to-rotate-x-axis-tick-labels-in-pandas-barplot)

https://matplotlib.org/examples/pie_and_polar_charts/pie_demo_features.html (https://matplotlib.org/examples/pie_and_polar_charts/pie_demo_features.html)

https://stackoverflow.com/questions/20124472/python-float-to-ratio (https://stackoverflow.com/questions/20124472/python-float-to-ratio)

http://www.th7.cn/Program/Python/201412/325891.shtml (http://www.th7.cn/Program/Python/201412/325891.shtml)

https://discussions.udacity.com/t/chi-square-test-of-independence-survivability-and-class/244662/7 (https://discussions.udacity.com/t/chi-square-test-of-independence-survivability-and-class/244662/7)

https://discussions.udacity.com/t/dataset-limitation-review/241549/2 (https://discussions.udacity.com/t/dataset-limitation-review/241549/2)