Case Study on Titanic dataset

May 12, 2021

0.0.1 Name:Krishna More

0.0.2 Seat no.:38

0.0.3 XIE-201903030

1 Case Study on Titanic dataset

1.1 Using Titanic dataset find out information about how many male survived who had cabin and age is less than 50. Also show graphical representation of male and female survived and dead in the tragedy. (LO6)

[1]:
```
pip install ipy_table
```

```
Requirement already satisfied: ipy_table in
c:\users\asus\anaconda3\anaconda3\lib\site-packages (1.15.1)
Note: you may need to restart the kernel to use updated packages.
```

[2]:
```
#importing of required modules
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import ipy_table as tbl
from numbers import Number
from scipy import stats
#allow plots and visualisations to be displayed in the report
%pylab inline
```

```
Populating the interactive namespace from numpy and matplotlib
```

[3]:
```
def as_percent(val, precision='0.2'):
    """Convert number to percentage string."""
    if isinstance(val, Number):
        return "{{:{}%}}".format(precision).format(val)
    else:
        raise TypeError("Numeric type required")

def calculate_percentage(val, total, format_percent = False):
```

```
    """Calculates the percentage of a value over a total"""
    percent = np.divide(val, total, dtype=float)
    if format_percent:
        percent = as_percent(percent)
    return percent
```

2 Read CSV into dataframe

[4]:
```
# Read csv into Pandas Dataframe and store in dataset variable
titanic_df = pd.read_csv('titanic_data.csv')
```

3 Data Wrangling / Cleaning

[5]:
```
# print out information about the data
titanic_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: 66.2+ KB
```

[6]:
```
titanic_df.describe()
```

[6]:

	PassengerId	Survived	Pclass	Age	SibSp	\
count	891.000000	891.000000	891.000000	714.000000	891.000000	
mean	446.000000	0.383838	2.308642	29.699118	0.523008	
std	257.353842	0.486592	0.836071	14.526497	1.102743	
min	1.000000	0.000000	1.000000	0.420000	0.000000	
25%	223.500000	0.000000	2.000000	20.125000	0.000000	
50%	446.000000	0.000000	3.000000	28.000000	0.000000	

```
75%     668.500000    1.000000    3.000000   38.000000    1.000000
max     891.000000    1.000000    3.000000   80.000000    8.000000

            Parch        Fare
count  891.000000  891.000000
mean     0.381594   32.204208
std      0.806057   49.693429
min      0.000000    0.000000
25%      0.000000    7.910400
50%      0.000000   14.454200
75%      0.000000   31.000000
max      6.000000  512.329200
```

[7]:
```
titanic_df.head()
```

[7]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       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
```

[8]:
```
titanic_df.tail()
```

[8]:
```
     PassengerId  Survived  Pclass                                      Name  \
886          887         0       2                     Montvila, Rev. Juozas
887          888         1       1              Graham, Miss. Margaret Edith
888          889         0       3  Johnston, Miss. Catherine Helen "Carrie"
889          890         1       1                     Behr, Mr. Karl Howell
890          891         0       3                       Dooley, Mr. Patrick

        Sex   Age  SibSp  Parch      Ticket   Fare Cabin Embarked
886    male  27.0      0      0      211536  13.00   NaN        S
```

```
887  female  19.0      0      0      112053  30.00   B42        S
888  female   NaN      1      2  W./C. 6607  23.45   NaN        S
889    male  26.0      0      0      111369  30.00  C148        C
890    male  32.0      0      0      370376   7.75   NaN        Q
```

[9]:
```
titanic_df.head(10).T
```

[9]:
```
                                    0  \
PassengerId                         1
Survived                            0
Pclass                              3
Name         Braund, Mr. Owen Harris
Sex                             male
Age                               22
SibSp                              1
Parch                              0
Ticket                     A/5 21171
Fare                            7.25
Cabin                            NaN
Embarked                           S

                                                        1  \
PassengerId                                             2
Survived                                                1
Pclass                                                  1
Name         Cumings, Mrs. John Bradley (Florence Briggs Th...
Sex                                                  female
Age                                                      38
SibSp                                                     1
Parch                                                     0
Ticket                                             PC 17599
Fare                                                71.2833
Cabin                                                   C85
Embarked                                                  C

                                    2  \
PassengerId                         3
Survived                            1
Pclass                              3
Name         Heikkinen, Miss. Laina
Sex                            female
Age                                26
SibSp                               0
Parch                               0
Ticket             STON/O2. 3101282
Fare                           7.925
Cabin                            NaN
```

```
Embarked                          S

                                                      3  \
PassengerId                                           4
Survived                                              1
Pclass                                                1
Name         Futrelle, Mrs. Jacques Heath (Lily May Peel)
Sex                                                female
Age                                                   35
SibSp                                                  1
Parch                                                  0
Ticket                                            113803
Fare                                                53.1
Cabin                                               C123
Embarked                                               S

                                  4                5  \
PassengerId                       5                6
Survived                          0                0
Pclass                            3                3
Name         Allen, Mr. William Henry  Moran, Mr. James
Sex                            male              male
Age                              35               NaN
SibSp                             0                0
Parch                             0                0
Ticket                       373450           330877
Fare                           8.05           8.4583
Cabin                           NaN              NaN
Embarked                          S                Q

                                  6                                7  \
PassengerId                       7                                8
Survived                          0                                0
Pclass                            1                                3
Name         McCarthy, Mr. Timothy J  Palsson, Master. Gosta Leonard
Sex                            male                             male
Age                              54                                2
SibSp                             0                                3
Parch                             0                                1
Ticket                        17463                           349909
Fare                        51.8625                          21.075
Cabin                           E46                              NaN
Embarked                          S                                S

                                                             8  \
PassengerId                                                  9
Survived                                                     1
```

```
Pclass                                                      3
Name         Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)
Sex                                                    female
Age                                                        27
SibSp                                                       0
Parch                                                       2
Ticket                                                 347742
Fare                                                  11.1333
Cabin                                                     NaN
Embarked                                                    S

                                                            9
PassengerId                                                10
Survived                                                    1
Pclass                                                      2
Name         Nasser, Mrs. Nicholas (Adele Achem)
Sex                                                    female
Age                                                        14
SibSp                                                       1
Parch                                                       0
Ticket                                                 237736
Fare                                                  30.0708
Cabin                                                     NaN
Embarked                                                    C
```

3.1 Missing Ages

In order to populate the missing ages I will use the mean age based on the Sex and Pclass

[10]:
```
missing_ages = titanic_df[titanic_df['Age'].isnull()]
# determine mean age based on Sex and Pclass
mean_ages = titanic_df.groupby(['Sex','Pclass'])['Age'].mean()

def remove_na_ages(row):
    '''
    function to check if the age is null and replace wth the mean from
    the mean ages dataframe
    '''
    if pd.isnull(row['Age']):
        return mean_ages[row['Sex'],row['Pclass']]
    else:
        return row['Age']

titanic_df['Age'] =titanic_df.apply(remove_na_ages, axis=1)
```

3.2 Missing embarkation ports

In order to populate the missing embarked ports I need to first determine if the people with the missing information may have been travelling with others.

[11]:
```
missing_ports = titanic_df[titanic_df['Embarked'].isnull()]
missing_ports
```

[11]:
```
     PassengerId  Survived  Pclass                                      Name  \
61            62         1       1                       Icard, Miss. Amelie
829          830         1       1  Stone, Mrs. George Nelson (Martha Evelyn)

        Sex   Age  SibSp  Parch  Ticket  Fare Cabin Embarked
61   female  38.0      0      0  113572  80.0   B28      NaN
829  female  62.0      0      0  113572  80.0   B28      NaN
```

[12]:
```
# search by ticket number and cabin
titanic_df[(titanic_df['Embarked'].notnull()) & ((titanic_df['Ticket'] ==␣
 ↪'113572') | (titanic_df['Cabin'] == 'B28'))]
```

[12]:
```
Empty DataFrame
Columns: [PassengerId, Survived, Pclass, Name, Sex, Age, SibSp, Parch, Ticket,
Fare, Cabin, Embarked]
Index: []
```

Since searching for similar records did not return any results and it appears that both were travelling in the same cabin and with the same ticket number and the bulk of passengers were travelling from Southhampton, I have choosen to use Southhampton as the missing value.

[13]:
```
titanic_df['Embarked'].fillna('S',inplace=True)
```

3.3 Remove un-wanted columns

Since the Cabin, Name and Ticket numbers are not required in this analysis I will remove them to improve the speed of processing the dataframe.

[14]:
```
# dropping columns Cabin, Name and Ticket
titanic_df = titanic_df.drop(['Cabin','Name','Ticket'], axis=1)
titanic_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 9 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   Sex          891 non-null    object
```

```
 4   Age          891 non-null    float64
 5   SibSp        891 non-null    int64
 6   Parch        891 non-null    int64
 7   Fare         891 non-null    float64
 8   Embarked     891 non-null    object
dtypes: float64(2), int64(5), object(2)
memory usage: 55.8+ KB
```

3.4 Mapping data (values to descriptions)

I will also add a Family Size column so that I can compare the size of families with the number of survivors.

[15]:
```
def map_data(df):
    '''
    Function which takes the original dataframe and returns a
    clean / updated dataframe
    '''
    # survived map
    survived_map = {0: False, 1: True}
    df['Survived'] = df['Survived'].map(survived_map)

    # PClass map
    pclass_map = {1: 'Upper Class', 2: 'Middle Class', 3: 'Lower Class'}
    df['Pclass'] = df['Pclass'].map(pclass_map)

    # Embarkation port map
    port_map = {'S': 'Southampton', 'C': 'Cherbourg','Q':'Queenstown'}
    df['Embarked'] = df['Embarked'].map(port_map)

    # add new column (FamilySize) to dataframe - sum of SibSp and Parch
    df['FamilySize'] = df['SibSp'] + df['Parch']

    return df

titanic_df = map_data(titanic_df)
titanic_df.head(3)
```

[15]:

	PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	FamilySize
0	1	False	Lower Class	male	22.0	1	0	7.2500	Southampton	1
1	2	True	Upper Class	female	38.0	1	0	71.2833	Cherbourg	1
2	3	True	Lower Class	female	26.0	0	0	7.9250	Southampton	0

3.5 Grouping / Binning Ages

To make the ages easier to analyse I thought it would be a good idea to group / bin the ages. This way we can compare groups of ages instead of individual ages.

[16]:
```
age_labels = ['0-9', '10-19', '20-29', '30-39', '40-49', '50-59', '60-69',␣
 ↪'70-79']
titanic_df['age_group'] = pd.cut(titanic_df.Age, range(0, 81, 10), right=False,␣
 ↪labels=age_labels)
```

4 Analysis of data

4.1 Number of Survivors

Before trying to determine the characteristics of a passenger that would make them more likely to survive, the number of survivors in the sample should be compared to the actual number of survivors. Based on the information provided by the source of the dataset (Kaggle) there were 2224 passengers and 722 survivors.

[17]:
```
# passengers and number of survivors based on Kaggle results
kaggle_passengers = 2224
kaggle_nonsurvivors = 1502
kaggle_survivors = kaggle_passengers - kaggle_nonsurvivors

# Count number of passengers and number of survivors in sample data
sample_passengers = len(titanic_df)
sample_survivors = len(titanic_df[titanic_df.Survived==True])
sample_nonsurvivors = sample_passengers - sample_survivors


survivors_data = titanic_df[titanic_df.Survived==True]
non_survivors_data = titanic_df[titanic_df.Survived==False]

survivors = [
    ['Item','Kaggle (Count)','Kaggle (%)' ,'Sample Dataset (Count)', 'Sample␣
 ↪Dataset (%)'],
    ['Total Passengers',kaggle_passengers,'-', sample_passengers,'-'],
    ['Survivors',
         kaggle_survivors,
         calculate_percentage(kaggle_survivors,kaggle_passengers, True),
         sample_survivors,
         calculate_percentage(sample_survivors,sample_passengers, True)
    ],
    ['Non-survivors',
         kaggle_nonsurvivors,
         calculate_percentage(kaggle_nonsurvivors,kaggle_passengers, True),
         sample_nonsurvivors,
```

```
        calculate_percentage(sample_nonsurvivors,sample_passengers, True)
    ]
]
tbl.make_table(survivors)
```

[17]: <ipy_table.ipy_table.IpyTable at 0x928fbb0>

When comparing the number of survivors from the sample dataset to the actual number of survivors we can see that the percentage of survivors is realitively close to each other.

4.2 Which gender had a better chance of survival?

In order to answer this question we need to look at how many males and females were on board and which gender had the highest survival rate. ### Hypothesis The hypothesis for this question is that the gender does impact the chances of survival

H0 = Gender has no impact on survivability

HA = Gender does impact the chances of survivabily

4.2.1 Count of Survivors by Gender

[18]:
```
table = pd.crosstab(titanic_df['Survived'],titanic_df['Sex'])
print(table)
```

```
Sex       female  male
Survived
False         81   468
True         233   109
```

4.2.2 Proportion of survivors by Gender

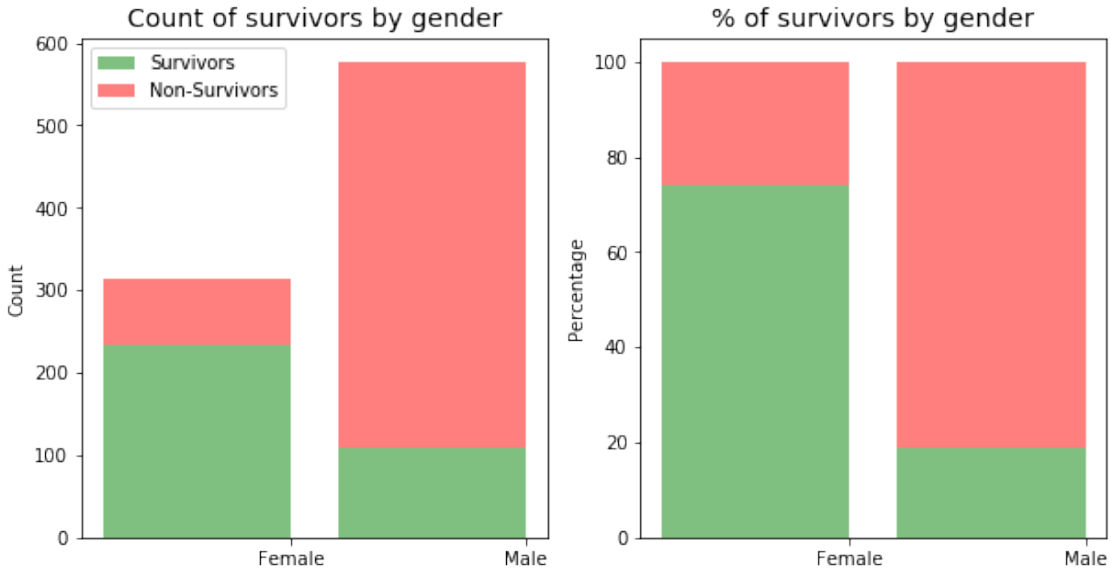
[19]:
```
print (titanic_df.groupby('Sex').Survived.mean())
```

```
Sex
female    0.742038
male      0.188908
Name: Survived, dtype: float64
```

[20]:
```
# calculate values for each survival status
survivors_gender = survivors_data.groupby(['Sex']).size().values
non_survivors_gender = non_survivors_data.groupby(['Sex']).size().values

# calculate totals for percentates
totals = survivors_gender + non_survivors_gender

# use calculate_percentage_function to calculate percentage of the total
data1_percentages = calculate_percentage(survivors_gender, totals)*100
```

```
data2_percentages = calculate_percentage(non_survivors_gender, totals)*100

gender_categories = ['Female', 'Male']

f, (ax1, ax2) = plt.subplots(1, 2, figsize=(10,5))
# plot chart for count of survivors by class
ax1.bar(range(len(survivors_gender)), survivors_gender, label='Survivors',␣
 ↪alpha=0.5, color='g')
ax1.bar(range(len(non_survivors_gender)), non_survivors_gender,␣
 ↪bottom=survivors_gender, label='Non-Survivors', alpha=0.5, color='r')
plt.sca(ax1)
plt.xticks([0.4, 1.4], gender_categories )
ax1.set_ylabel("Count")
ax1.set_xlabel("")
ax1.set_title("Count of survivors by gender",fontsize=14)
plt.legend(loc='upper left')

# plot chart for percentage of survivors by class
ax2.bar(range(len(data1_percentages)), data1_percentages, alpha=0.5, color='g')
ax2.bar(range(len(data2_percentages)), data2_percentages,␣
 ↪bottom=data1_percentages, alpha=0.5, color='r')
plt.sca(ax2)
plt.xticks([0.4, 1.4],  gender_categories)
ax2.set_ylabel("Percentage")
ax2.set_xlabel("")
ax2.set_title("% of survivors by gender",fontsize=14)
```

[20]: Text(0.5, 1.0, '% of survivors by gender')



The plots and proportions above show that there were a significant more males on board the Titanic compared to the number of females. Whilst the second plot (% of survivors by gender) shows that Females had a higher proportion (74.2%) of survivors compared to the proportion of males (18.9%). This shows that females had a greater rate of survival.

As the P-Value is less than 0.05 the probability of that the age group will impact the chances of survival is high. Therefore I believe that we can reject the null hypothesis.

[21]:
```
table = pd.crosstab([titanic_df['Survived']], titanic_df['Sex'])
chi2, p, dof, expected = stats.chi2_contingency(table.values)
results = [
    ['Item','Value'],
    ['Chi-Square Test',chi2],
    ['P-Value', p]
]
tbl.make_table(results)
```

[21]:
```
<ipy_table.ipy_table.IpyTable at 0xa4dd550>
```

As the P-Value is less than 0.05 the probability of that the gender will impact the chances of survival is high. Therefore I believe that we can reject the null hypothesis. I also believe that the plots above confirm this result.

4.3 Which social class had a better chance of survival?

4.3.1 Hypothesis

The hypothesis for this question is that the social class does impact the chances of survival

H0 = Social Class has no impact on survivability

HA = Social Class does impact the chances of survivabily

4.3.2 Count of survivors by class

[22]:
```
table = pd.crosstab(titanic_df['Survived'],titanic_df['Pclass'])
print (table)
```

```
Pclass    Lower Class  Middle Class  Upper Class
Survived
False             372            97           80
True              119            87          136
```

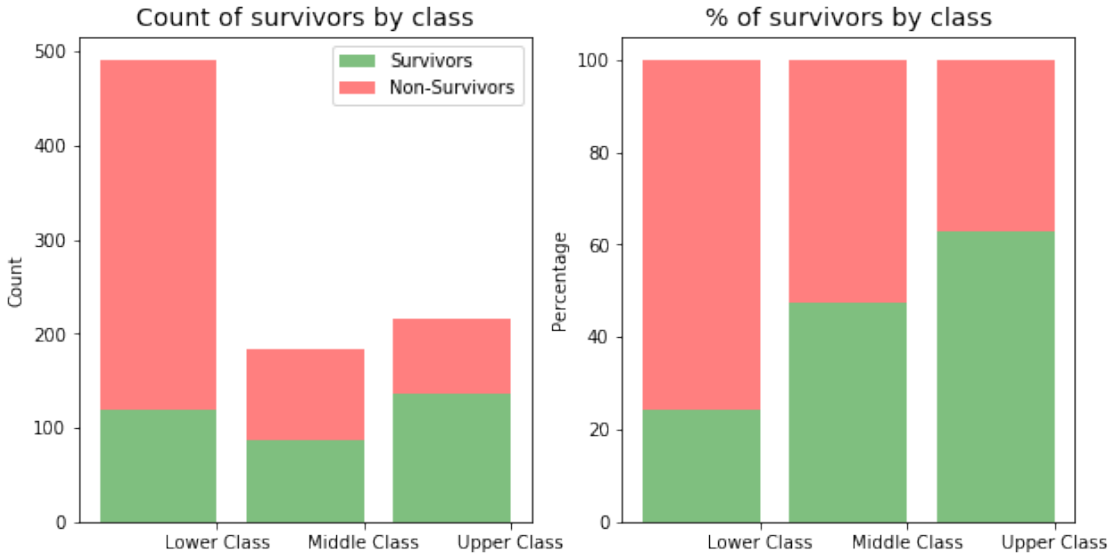
4.3.3 Proportion of survivors by class

[23]:
```
print (titanic_df.groupby('Pclass').Survived.mean())
```

```
Pclass
Lower Class     0.242363
Middle Class    0.472826
Upper Class     0.629630
Name: Survived, dtype: float64
```

[24]:
```
# calculate values for each survival status
survivors_class = survivors_data.groupby(['Pclass']).size().values
non_survivors_class = non_survivors_data.groupby(['Pclass']).size().values

# calculate totals for percentates
totals = survivors_class + non_survivors_class

# use calculate_percentage_function to calculate percentage of the total
data1_percentages = calculate_percentage(survivors_class, totals)*100
data2_percentages = calculate_percentage(non_survivors_class, totals)*100


class_categories = ['Lower Class', 'Middle Class', 'Upper Class']


f, (ax1, ax2) = plt.subplots(1, 2, figsize=(10,5))
# plot chart for count of survivors by class
ax1.bar(range(len(survivors_class)), survivors_class, label='Survivors', alpha=0.
 ↪5, color='g')
ax1.bar(range(len(non_survivors_class)), non_survivors_class,␣
 ↪bottom=survivors_class, label='Non-Survivors', alpha=0.5, color='r')
plt.sca(ax1)
plt.xticks([0.4, 1.4, 2.4], class_categories )
ax1.set_ylabel("Count")
ax1.set_xlabel("")
ax1.set_title("Count of survivors by class",fontsize=14)
plt.legend(loc='upper right')

# plot chart for percentage of survivors by class
ax2.bar(range(len(data1_percentages)), data1_percentages, alpha=0.5, color='g')
ax2.bar(range(len(data2_percentages)), data2_percentages,␣
 ↪bottom=data1_percentages, alpha=0.5, color='r')
plt.sca(ax2)
plt.xticks([0.4, 1.4, 2.4],  class_categories)
ax2.set_ylabel("Percentage")
ax2.set_xlabel("")
ax2.set_title("% of survivors by class",fontsize=14)
```

[24]:
```
Text(0.5, 1.0, '% of survivors by class')
```



The graphs above so that whilst the lower class had more passengers, than all classes, and more survivors than the middle class, the lower class had the lowest survival rate. The Upper Class passengers had the highest survival rate

4.3.4 Hypothesis Test

For this test I will be using the chi-sqaure test for independence

[25]:
```
table = pd.crosstab([titanic_df['Survived']], titanic_df['Pclass'])
chi2, p, dof, expected = stats.chi2_contingency(table.values)
results = [
    ['Item','Value'],
    ['Chi-Square Test',chi2],
    ['P-Value', p]
]
tbl.make_table(results)
```

[25]:
```
<ipy_table.ipy_table.IpyTable at 0xabd87d0>
```

As the P-Value is less than 0.05 the probability of that the social class will impact the chances of survival is high. Therefore I believe that we can reject the null hypothesis. I also believe that the plots above confirm this result.

4.4 Which age group had a better chance of survival?

4.4.1 Hypothesis

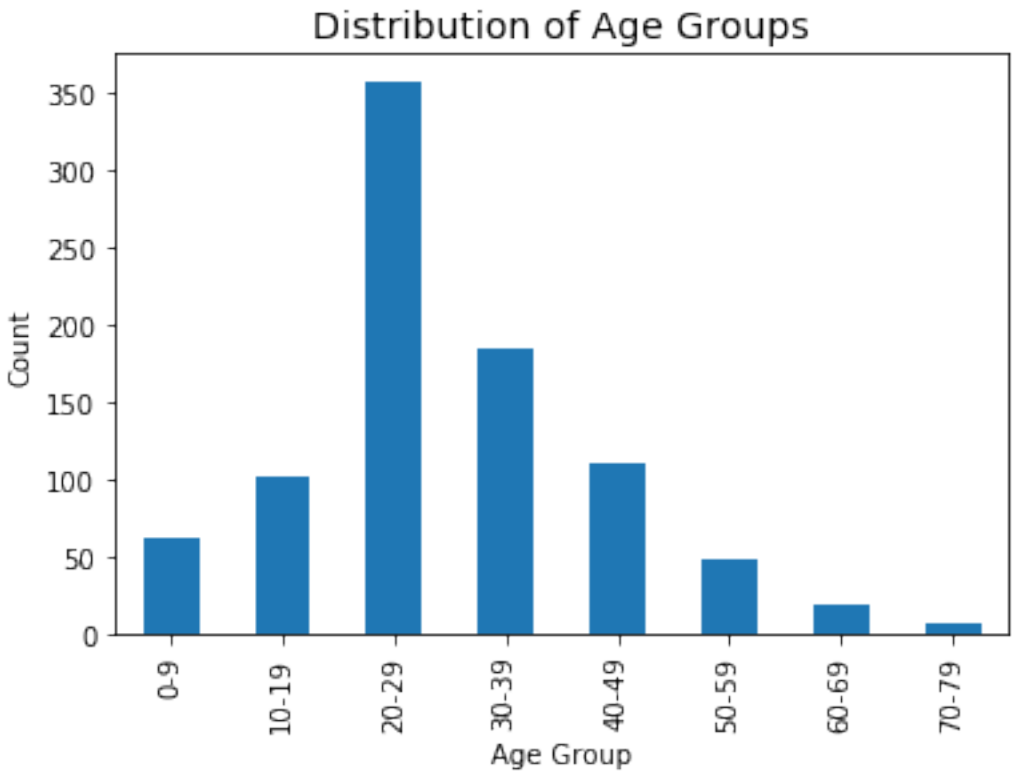
The hypothesis for this question is that the age group does impact the chances of survival

H0 = Age Group has no impact on survivability

HA = Age Group does impact the chances of survivabily

4.4.2 Distribution of Age Groups

[26]:
```
titanic_df.groupby(['age_group']).size().plot(kind='bar',stacked=True)
plt.title("Distribution of Age Groups",fontsize=14)
plt.ylabel('Count')
plt.xlabel('Age Group');
```



From the plot above we can see that the majority of passengers were aged between 20-29

4.4.3 Proportion of survivors by age group

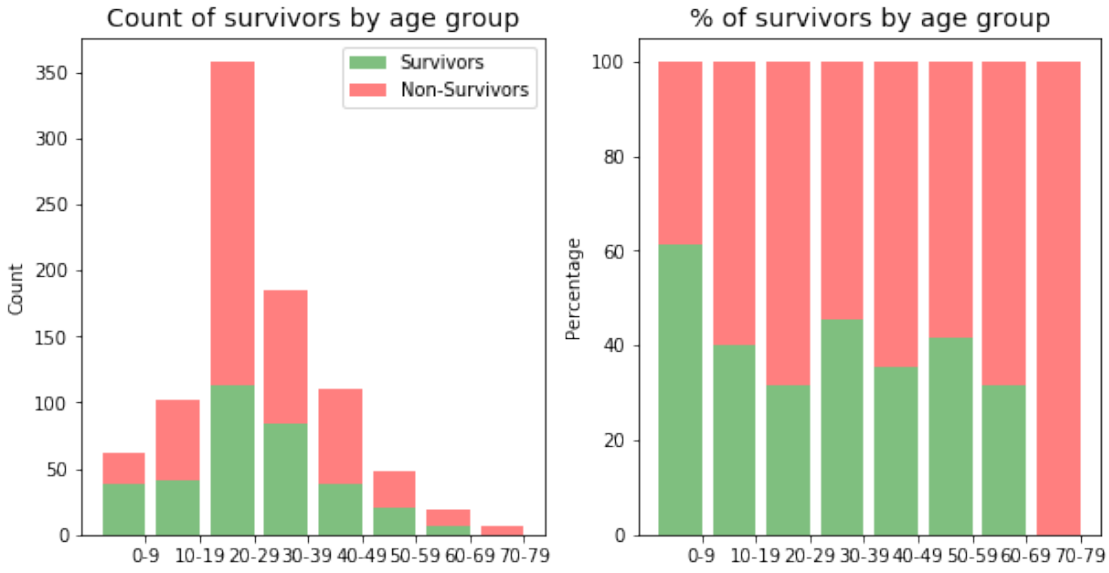
[27]:
```
print (titanic_df.groupby(['age_group']).Survived.mean())
```

```
age_group
0-9       0.612903
10-19     0.401961
20-29     0.315642
30-39     0.454054
40-49     0.354545
50-59     0.416667
```

```
60-69    0.315789
70-79    0.000000
Name: Survived, dtype: float64
```

[28]:
```
# calculate values for each survival status
survivors_age_group = survivors_data.groupby(['age_group']).size().values
non_survivors_age_group = non_survivors_data.groupby(['age_group']).size().values

# calculate totals for percentates
totals = survivors_age_group + non_survivors_age_group

# use calculate_percentage_function to calculate percentage of the total
data1_percentages = calculate_percentage(survivors_age_group, totals)*100
data2_percentages = calculate_percentage(non_survivors_age_group, totals)*100


tick_spacing = np.array(range(len(age_labels)))+0.4


f, (ax1, ax2) = plt.subplots(1, 2, figsize=(10,5))
# plot chart for count of survivors by class
ax1.bar(range(len(survivors_age_group)), survivors_age_group, label='Survivors',␣
 ↪alpha=0.5, color='g')
ax1.bar(range(len(non_survivors_age_group)), non_survivors_age_group,␣
 ↪bottom=survivors_age_group, label='Non-Survivors', alpha=0.5, color='r')
plt.sca(ax1)
plt.xticks(tick_spacing, age_labels )
ax1.set_ylabel("Count")
ax1.set_xlabel("")
ax1.set_title("Count of survivors by age group",fontsize=14)
plt.legend(loc='upper right')

# plot chart for percentage of survivors by class
ax2.bar(range(len(data1_percentages)), data1_percentages, alpha=0.5, color='g')
ax2.bar(range(len(data2_percentages)), data2_percentages,␣
 ↪bottom=data1_percentages, alpha=0.5, color='r')
plt.sca(ax2)
plt.xticks(tick_spacing,  age_labels)
ax2.set_ylabel("Percentage")
ax2.set_xlabel("")
ax2.set_title("% of survivors by age group",fontsize=14)
```

[28]:
```
Text(0.5, 1.0, '% of survivors by age group')
```



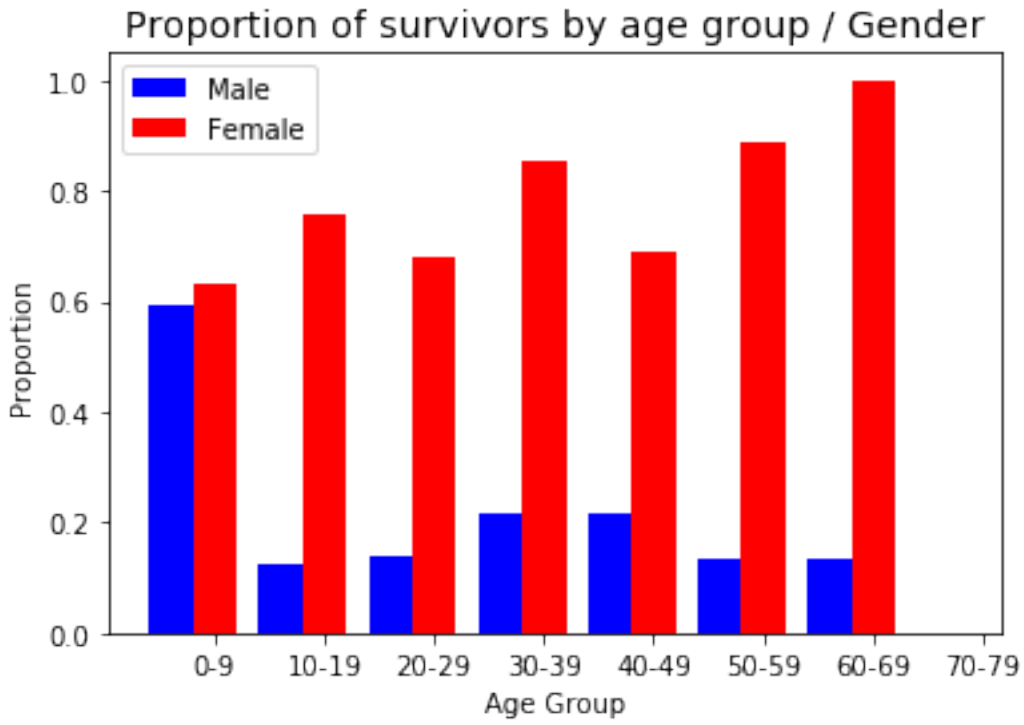
When looking at proportions and percentages of survivors per age group, initially I was suprised by the results, until I thought that this analysis shoudl take into consideration the gender / sex of the passengers as well.

[29]:
```
print (titanic_df.groupby(['Sex','age_group']).Survived.mean())
```

```
Sex     age_group
female  0-9          0.633333
        10-19        0.755556
        20-29        0.681034
        30-39        0.855072
        40-49        0.687500
        50-59        0.888889
        60-69        1.000000
        70-79             NaN
male    0-9          0.593750
        10-19        0.122807
        20-29        0.140496
        30-39        0.215517
        40-49        0.217949
        50-59        0.133333
        60-69        0.133333
        70-79        0.000000
Name: Survived, dtype: float64
```

[30]:
```
male_data = titanic_df[titanic_df.Sex == "male"].groupby('age_group').Survived.
 ↪mean().values
```

```
female_data = titanic_df[titanic_df.Sex == "female"].groupby('age_group').
 ↪Survived.mean().values
ax = plt.subplot()
male_plt_position = np.array(range(len(age_labels)))
female_plt_position = np.array(range(len(age_labels)))+0.4
ax.bar(male_plt_position, male_data,width=0.4,label='Male',color='b')
ax.bar(female_plt_position, female_data,width=0.4,label='Female',color='r')
plt.xticks(tick_spacing,  age_labels)
ax.set_ylabel("Proportion")
ax.set_xlabel("Age Group")
ax.set_title("Proportion of survivors by age group / Gender",fontsize=14)
plt.legend(loc='best')
plt.show()
```



After relooking at the proportion of survivors by age group and gender, the data supports notion of women and children to be given preferential treatment over men. The plot "Proportion of survivors by age group / gender", shows that children (0-9 years old, male and female) and women (all ages) had a much higher proportion of survivors. This supports the notion of the seats in the lifeboats been given to Women and Children first.

4.4.4 Hypothesis Test

For this test I will be using the chi-sqaure test for independence

[31]:
```
table = pd.crosstab([titanic_df['Survived']], titanic_df['age_group'])
chi2, p, dof, expected = stats.chi2_contingency(table.values)
results = [
    ['Item','Value'],
    ['Chi-Square Test',chi2],
    ['P-Value', p]
]
tbl.make_table(results)
```

[31]:
```
<ipy_table.ipy_table.IpyTable at 0xac57890>
```