

```

# Import the useful Libraries
# Ignore warnings
import warnings
warnings.filterwarnings('ignore')

#Handle table-like data and matrices
import numpy as np
import pandas as pd

#For visualisaton
import matplotlib as mpl
import matplotlib.pyplot as plt
import seaborn as sns

#Configure visualisations
%matplotlib inline
mpl.style.use( 'ggplot' )
sns.set_style( 'white' )

# Read the data set of "Titanic" in data.

Titanic= pd.read_csv("train.csv")

# printing the data
print(Titanic.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
None

```

```
#Printing shape of dataset Titanic
```

```
print ( 'The data shape of Titanic dataset is =', Titanic.shape)
```

The data shape of Titanic dataset is = (891, 12)

```
#In order to see the first 5 rows of dataset
```

```
Titanic.head()
```

	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

```
#In order to see what kind of values all columns have we will use  
describe funtion
```

```
Titanic.describe()
```

	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

#Plotting of graphs in order to have relationships and distributions

Histogram plot
Distribution
Categories
Correlation Plots

```
def plot_histograms( Titanic , variables, n_rows, n_cols ):
    fig = plt.figure( figsize = ( 16, 12 ) )
    for i, var_name in enumerate( variables ):
        ax=fig.add_subplot( n_rows , n_cols , i+1 )
        Titanic[ var_name ].hist( bins=10 , ax=ax )
        ax.set_title( 'Skew: ' + str( round ( float( Titanic[ var_name
].skew() ), ) ) ) # setting title of histogram
        ax.set_xticklabels( [] , visible=False ) # name of the x and
y axis
        ax.set_yticklabels( [] , visible=False )
    fig.tight_layout() #improves appearance
    plt.show()

def plot_distribution( Titanic , var , target , **kwargs ):
    row = kwargs.get( 'row' , None )
    col = kwargs.get( 'col' , None )
    facet = sns.FacetGrid( Titanic , hue=target , aspect=4 , row = row
, col = col )
    facet.map( sns.kdeplot , var , shade= True )
    facet.set( xlim=( 0 , Titanic[ var ].max() ) )
    facet.add_legend()

def plot_categories( Titanic , cat , target , **kwargs ):
    row = kwargs.get( 'row' , None )
    col = kwargs.get( 'col' , None )
    facet = sns.FacetGrid( Titanic , row = row , col = col )
    facet.map( sns.barplot , cat , target )
    facet.add_legend()

def plot_correlation_map( Titanic ):
    corr = Titanic.corr()
    _ , ax = plt.subplots( figsize =( 12 , 10 ) )
    cmap = sns.diverging_palette( 220 , 10 , as_cmap = True )
    _ = sns.heatmap(
        corr,
```

```

        cmap = cmap,
        square=True,
        cbar_kws={ 'shrink' : .9 },
        ax=ax,
        annot = True,
        annot_kws = { 'fontsize' : 12 }
    )

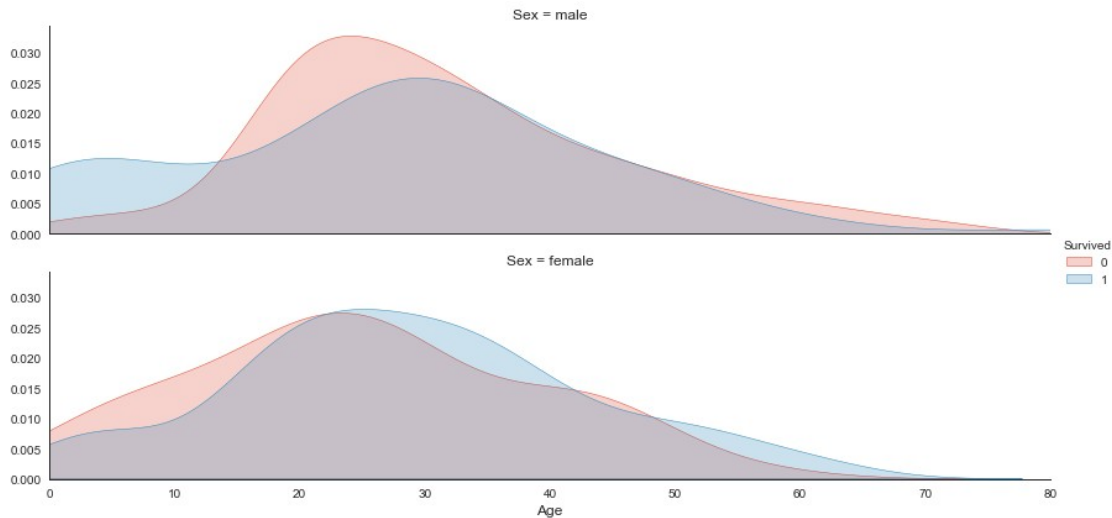
def describe_more( Titanic ):
    var = [] ; l = [] ; t = []
    for x in df:
        var.append( x )
        l.append( len( pd.value_counts( Titanic[ x ] ) ) )
        t.append( Titanic[ x ].dtypes )
    levels = pd.DataFrame( { 'Variable' : var , 'Levels' : l ,
'Datatype' : t } )
    levels.sort_values( by = 'Levels' , inplace = True )
    return levels

plot_correlation_map( Titanic ) #states the correlation between each
variables of titanic dataframe

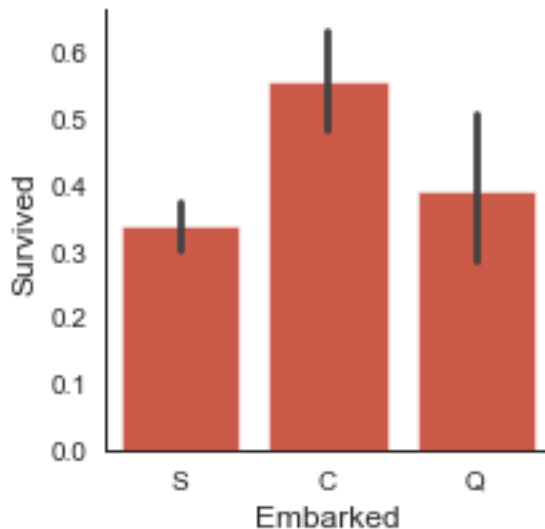
```



```
# Plot distribution of Age of passengers who survived or didn't
survive
plot_distribution( Titanic , var = 'Age' , target = 'Survived' , row =
'Sex' )
```



```
# Plot survival rate by Embarked
plot_categories( Titanic , cat = 'Embarked' , target = 'Survived' )
```



```
#categorical variables need to be transformed to numeric variables
#Transform Sex into binary values 0 and 1
```

```
sex = pd.Series( np.where( Titanic.Sex == 'male' , 1 , 0 ) , name =
'Sex' )
print(sex.head())
```

```
0    1
1    0
2    0
```

```
3      0
4      1
Name: Sex, dtype: int32
```

```
# fill missing values in variables
Titanic.Fare.isnull()
```

```
0      False
1      False
2      False
3      False
4      False
...
886     False
887     False
888     False
889     False
890     False
Name: Fare, Length: 891, dtype: bool
```

```
# Missing values for all columns into the dataframe
print("\n Missing values in dataframe are \n ",Titanic.isnull().sum())
```

```
# NA values in the Titanic's age column
print("\n\n Number of Age missings are : ",Titanic.Age.isna().sum())
```

```
Missing values in dataframe are
  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
```

```
Number of Age missings are : 177
```

```
# Creating a dataset, to store the imputed values
imputed = pd.DataFrame()
```

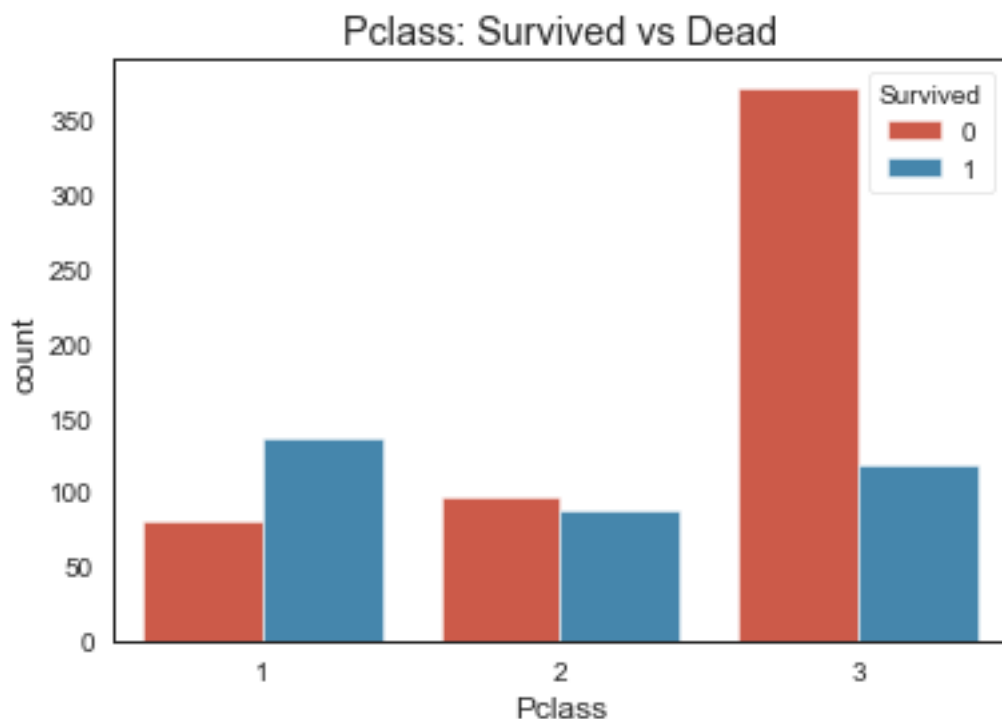
```
#Fill missing values of Age with the average of Age (mean)
imputed[ 'Age' ] = Titanic.Age.fillna( Titanic.Age.mean() )
```

```
imputed.head()
```

```
   Age
0  22.0
1  38.0
2  26.0
3  35.0
4  35.0
```

```
#plotting of the Data
```

```
sns.countplot('Pclass', hue='Survived', data=Titanic)
plt.title('Pclass: Survived vs Dead')
plt.show()
```

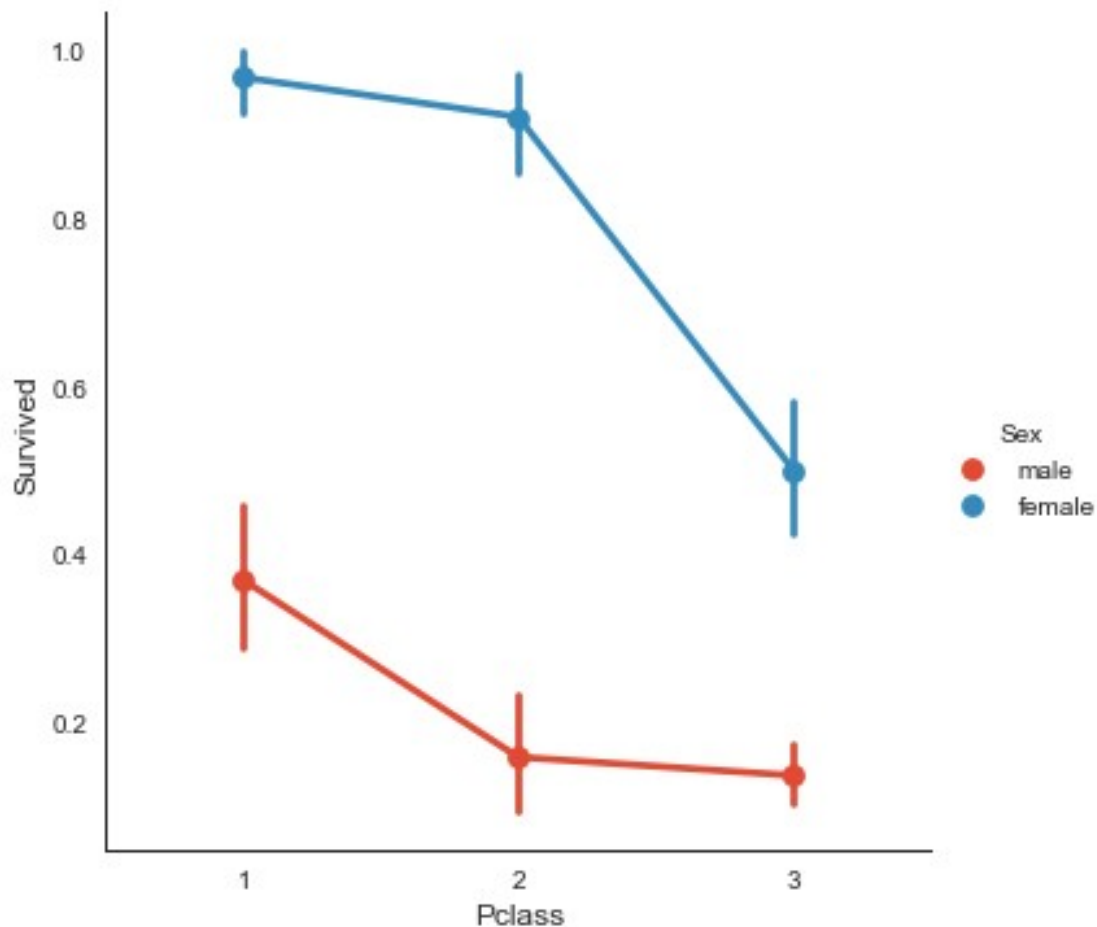


```
#Above Chart depicts that passengers of Class 1  
#have got higher preference during rescue compared to passengers of  
class 3  
#barely 25% of Class 3 passengers survived  
#class 3 has higher rate of deaths compared to other class  
#class 1 has higher survivals
```

```
#plot the crosstab of the data Survived and the Sex, with the Pclass  
pd.crosstab([Titanic.Sex,Titanic.Survived],Titanic.Pclass,margins=True  
)  
.style.background_gradient(cmap='summer_r')
```

```
<pandas.io.formats.style.Styler at 0x2bc71239ee0>
```

```
#percentage plot
sns.factorplot('Pclass', 'Survived', hue='Sex', data=Titanic)
plt.show()
```



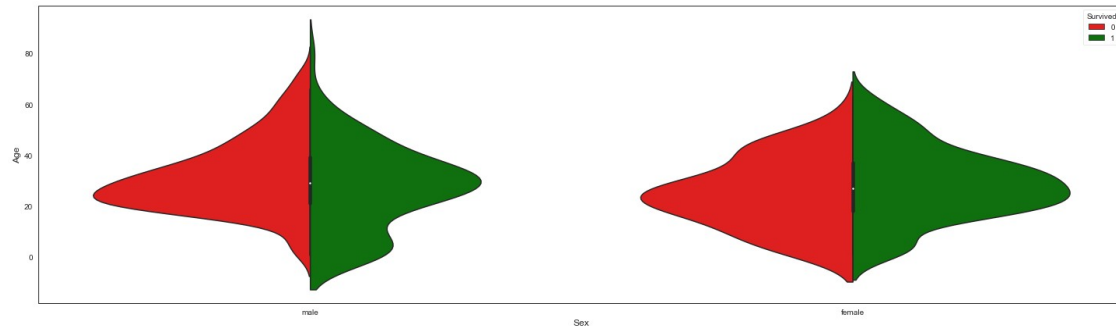
#The plots shows that women in class 1 has highest survival rate(nearly 95%)

#becuase only 3 out of 94 women died, 91 survived

#survival rate for men is low as compred to women in class1

#This depicts that instead of class women and children were given higher priority over any one else

```
fig = plt.figure(figsize=(25, 7))
sns.violinplot(x='Sex' , y='Age',
               hue='Survived', data=Titanic,
               split=True,
               palette={0: "r", 1: "g"}
               );
```

*#this graph shows that in case of emergency women and children are always given the preference for rescue of lifeboats.
#From movie we can also say that women and children were given the preference for Survival*