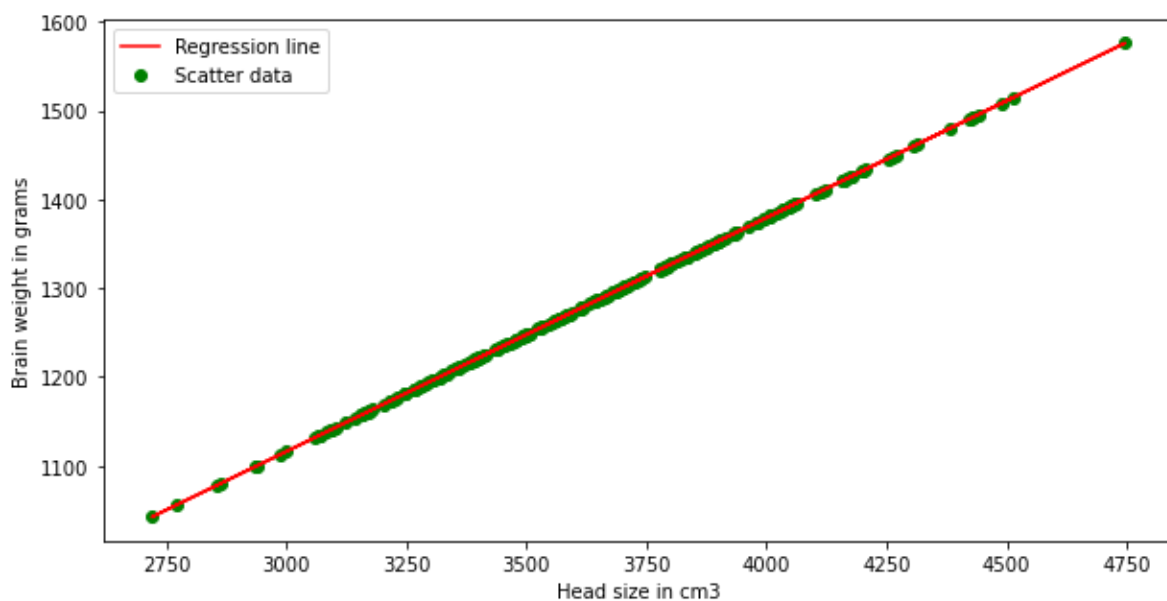


In [20]:

```
#Plotting values and regression line
%matplotlib inline
plt.rcParams['figure.figsize'] = (10.0, 5.0)
max_x = np.max(x) + 100
min_x = np.min(x) - 100
y = b0 + b1 * x
#Plotting line
plt.plot(x,y, color='red', label='Regression line')
#Plotting scatter points
plt.scatter(x,y, c = 'green', label='Scatter data')
plt.xlabel('Head size in cm3')
plt.ylabel('Brain weight in grams')
plt.legend()
plt.show()
```



14.05.2021

Logistic Regression

In [33]:

```
#IMPORTING IMPORTANT LIBRARIES
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import math
%matplotlib inline
```

1. Importing the data

In [34]:

```
#1 IMPORTING THE CSV
titanic_data = pd.read_csv("C:/Users/lenovo/Desktop/titanic.csv")
```

In [35]:

```
titanic_data.head(10)
```

Out[35]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708

In [36]:

```
#COUNTING THE NO OF PASSENGERS IN THE DATA  
print('no of passenger:' +str(len(titanic_data.index)))
```

no of passenger:891

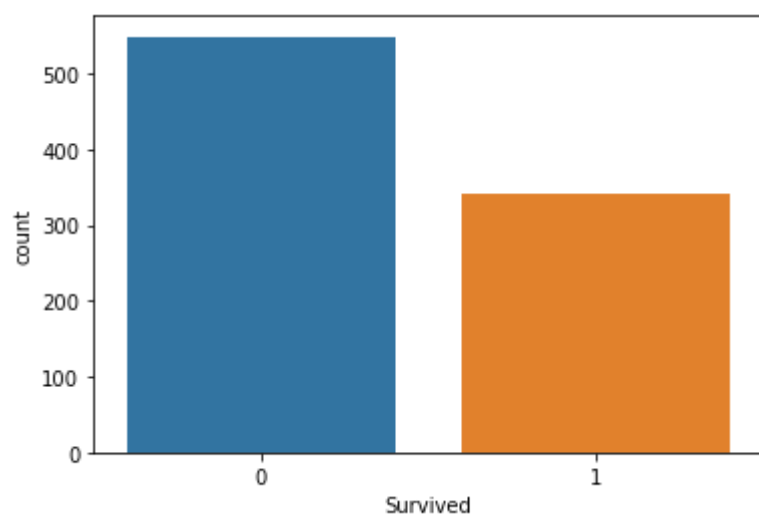
2. Analysing the data

In [37]:

```
sns.countplot(x='Survived',data=titanic_data)
```

Out[37]:

<AxesSubplot:xlabel='Survived', ylabel='count'>

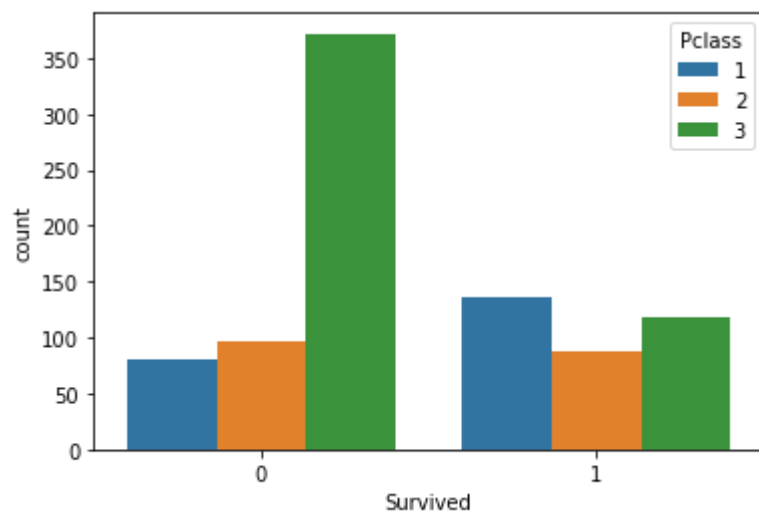


In [38]:

```
sns.countplot(x='Survived',hue='Pclass',data=titanic_data)
```

Out[38]:

<AxesSubplot:xlabel='Survived', ylabel='count'>

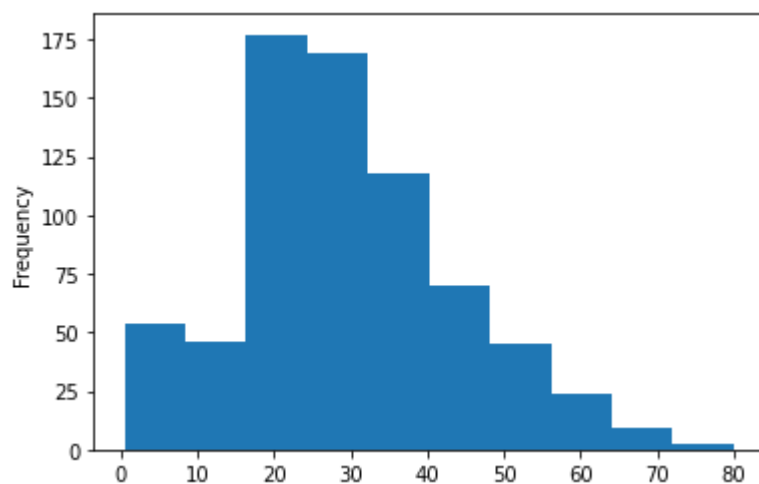


In [39]:

```
titanic_data['Age'].plot.hist()
```

Out[39]:

<AxesSubplot:ylabel='Frequency'>



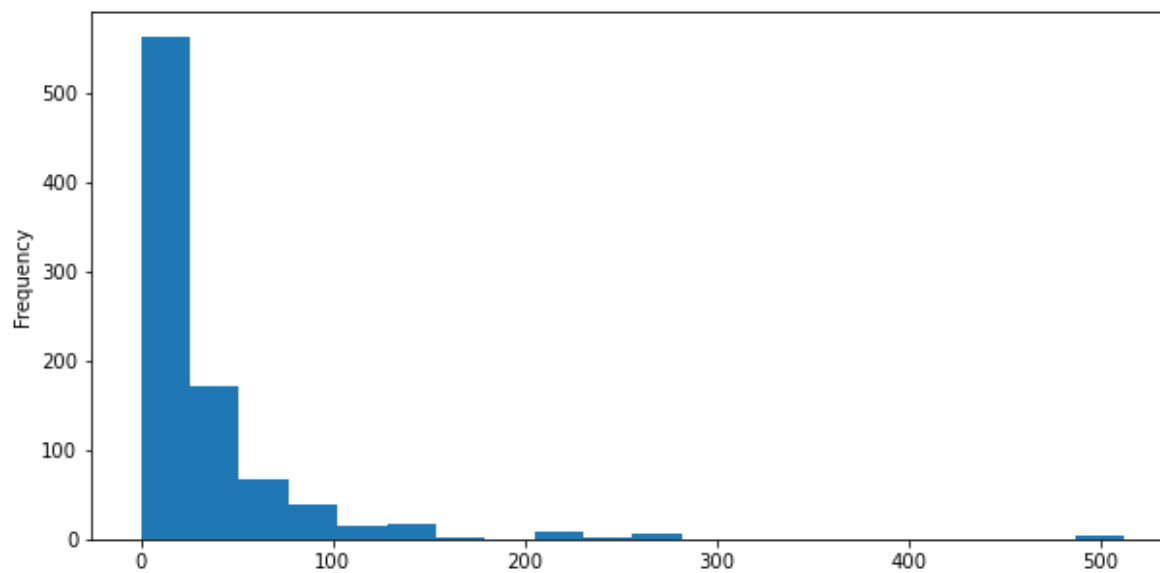
In [40]:



```
titanic_data['Fare'].plot.hist(bins=20,figsize=(10,5))
```

Out[40]:

<AxesSubplot:ylabel='Frequency'>



In [41]:



```
titanic_data.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
```

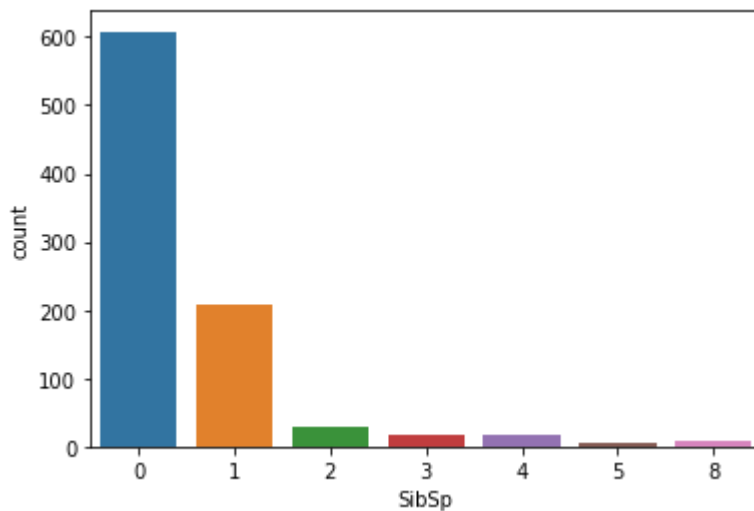
In [42]:



```
sns.countplot(x='SibSp',data=titanic_data)
```

Out[42]:

```
<AxesSubplot:xlabel='SibSp', ylabel='count'>
```



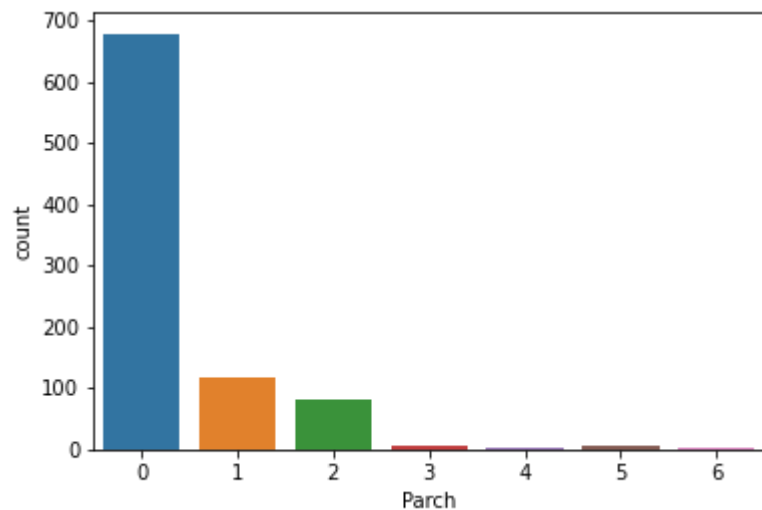
In [43]:



```
sns.countplot(x='Parch',data=titanic_data)
```

Out[43]:

```
<AxesSubplot:xlabel='Parch', ylabel='count'>
```



3. Data Wrangling

In [44]:

```
#LOOKING FOR THE NULL VALUES
titanic_data.isnull()
```

Out[44]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	False	False	False	False	False	False	False	False	False	False	True	
1	False	False	False	False	False	False	False	False	False	False	False	
2	False	False	False	False	False	False	False	False	False	False	True	
3	False	False	False	False	False	False	False	False	False	False	False	
4	False	False	False	False	False	False	False	False	False	False	True	
...
886	False	False	False	False	False	False	False	False	False	False	True	
887	False	False	False	False	False	False	False	False	False	False	False	
888	False	False	False	False	False	True	False	False	False	False	True	
889	False	False	False	False	False	False	False	False	False	False	False	
890	False	False	False	False	False	False	False	False	False	False	True	

891 rows × 12 columns

In [45]:

```
#COUNT OF NULLS
titanic_data.isnull().sum()
```

Out[45]:

```
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
```

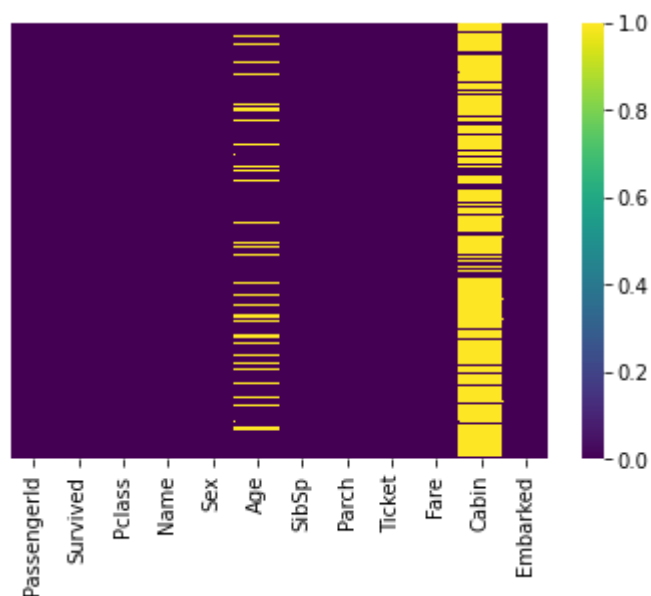

In [46]:

#HEATMAP FOR NULLS

```
sns.heatmap(titanic_data.isnull(),yticklabels=False,cmap='viridis')
```

Out[46]:

<AxesSubplot:>



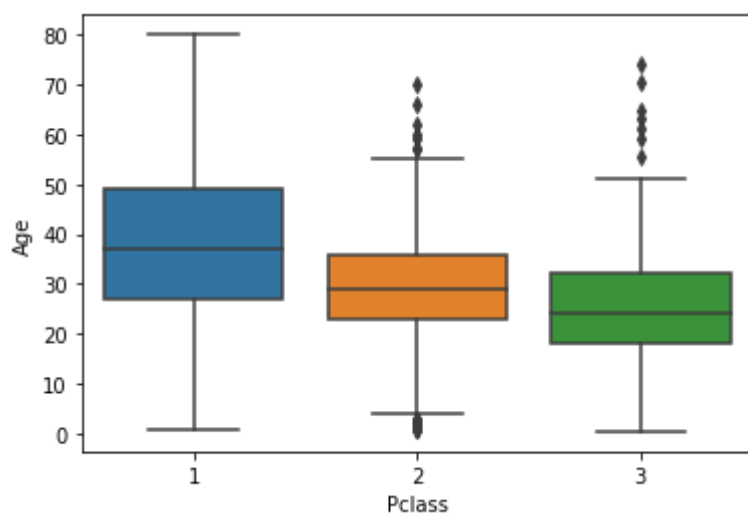
In [47]:

#BOX PLOT

```
sns.boxplot(x='Pclass',y='Age',data=titanic_data)
```

Out[47]:

<AxesSubplot:xlabel='Pclass', ylabel='Age'>



In [48]:

```
titanic_data.head(5)
```

Out[48]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500

In [49]:

```
titanic_data.drop('Cabin',axis=1,inplace=True)
```

In [50]:

```
titanic_data.head(5)
```

Out[50]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	I
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	
3	4	1	1	Futelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	

In [51]:

```
#DROPPING THE NULL VALUES FROM THE DATA FRAME  
titanic_data.dropna(inplace=True)
```

In [52]:



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

Out[52]:

```
PassengerId    0
Survived        0
Pclass         0
Name           0
Sex            0
Age           0
SibSp          0
Parch          0
Ticket         0
Fare           0
Embarked       0
dtype: int64
```

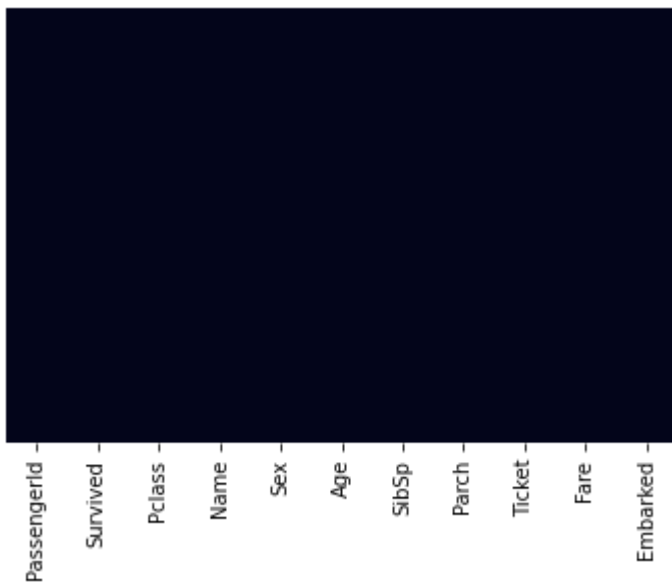
In [53]:



```
sns.heatmap(titanic_data.isnull(),yticklabels=False,cbar=False)
```

Out[53]:

<AxesSubplot:>



In [54]:

```
titanic_data.head(2)
```

Out[54]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	1	0	3Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	
1	2	1	1Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	

In [55]:

```
pd.get_dummies(titanic_data['Sex'])
```

Out[55]:

	female	male
0	0	1
1	1	0
2	1	0
3	1	0
4	0	1
...
885	1	0
886	0	1
887	1	0
889	0	1
890	0	1

712 rows × 2 columns

In [56]:



```
sex=pd.get_dummies(titanic_data['Sex'],drop_first=True)  
sex.head(5)
```

Out[56]:

	male
0	1
1	0
2	0
3	0
4	1

In [57]:



```
embark = pd.get_dummies(titanic_data['Embarked'])  
embark.head(5)
```

Out[57]:

	C	Q	S
0	0	0	1
1	1	0	0
2	0	0	1
3	0	0	1
4	0	0	1

In [58]:



```
embark = pd.get_dummies(titanic_data['Embarked'],drop_first=True)  
embark.head(5)
```

Out[58]:

	Q	S
0	0	1
1	0	0
2	0	1
3	0	1
4	0	1

In [59]:



```
pcl = pd.get_dummies(titanic_data['Pclass'])  
pcl.head(5)
```

Out[59]:

	1	2	3
0	0	0	1
1	1	0	0
2	0	0	1
3	1	0	0
4	0	0	1

In [60]:



```
pcl = pd.get_dummies(titanic_data['Pclass'],drop_first=True)  
pcl.head(5)
```

Out[60]:

	2	3
0	0	1
1	0	0
2	0	1
3	0	0
4	0	1

In [61]:



```
titanic_data = pd.concat([titanic_data,sex,embark,pcl],axis=1)
```

In [62]:



```
titanic_data.head(5)
```

Out[62]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	I
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	

In [63]:

```
titanic_data.head(10)
```

Out[63]:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	1	0	3Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500
1	2	1	1Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833
2	3	1	3Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250
3	4	1	1Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000
4	5	0	3Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500
6	7	0	1McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625
7	8	0	3Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750
8	9	1	3Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333
9	10	1	2Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708
10	11	1	3Sandstrom, Miss. Marguerite Rut	female	4.0	1	1	PP 9549	16.7000

In [64]:

```
titanic_data.drop(['Sex', 'Embarked', 'PassengerId', 'Name', 'Ticket'], axis=1, inplace=True)
```

In [65]:

```
titanic_data.head()
```

Out[65]:

	Survived	Pclass	Age	SibSp	Parch	Fare	male	Q	S	2	3
0	0	3	22.0	1	0	7.2500	1	0	1	0	1
1	1	1	38.0	1	0	71.2833	0	0	0	0	0
2	1	3	26.0	0	0	7.9250	0	0	1	0	1
3	1	1	35.0	1	0	53.1000	0	0	1	0	0
4	0	3	35.0	0	0	8.0500	1	0	1	0	1

4. Train and test

Build the model on the train data and predict on the test data

In [66]:

```
X = titanic_data.drop('Survived', axis=1)
Y = titanic_data['Survived']
```

In [69]:

```
from sklearn.model_selection import train_test_split
```

In [72]:

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.3, random_state=1)
```

In [74]:

```
from sklearn.linear_model import LogisticRegression
```

In [81]:

```
logmodel = LogisticRegression(solver='lbfgs', max_iter=10000)
```

In [82]:

```
logmodel.fit(X_train,Y_train)
```

Out[82]:

```
LogisticRegression(max_iter=10000)
```

In [84]:

```
prediction=logmodel.predict(X_test)
```

In [85]:

```
from sklearn.metrics import classification_report
```

In [86]:

```
classification_report(Y_test,prediction)
```

Out[86]:

```
'
      precision    recall  f1-score   support\n\n
0.81      0.83      0.82      126\n
0.74      88\n\n
macro avg      0.78      0.78      0.78      214\n
0.79      0.79      214\n
accuracy      0.79      214\n
weighted avg      0.79      214\n'
```

In [89]:

```
from sklearn.metrics import confusion_matrix
confusion_matrix(Y_test,prediction)
```

Out[89]:

```
array([[105,  21],
       [ 24,  64]], dtype=int64)
```

In [90]:

```
from sklearn.metrics import accuracy_score
```

In [91]:

```
accuracy_score(Y_test, prediction)
```

Out[91]:

```
0.7897196261682243
```

In [92]:



```
(105+64)/(105+21+24+64)
```

Out[92]:

```
0.7897196261682243
```

In []:

