Titanic survival rate classification report

Phase(1):Discovery

Motivation for developing this model is to offer improvement and risk assessment for ship insurance. Insurance companies provide coverage for ships and vessels, including passenger ships like the Titanic. Accurate risk assessment is crucial for determining appropriate insurance premiums and coverage for such ships. By analyzing the Titanic survival dataset, insurance companies can gain insights into the factors that influenced survival rates, which can help improve risk assessment and underwriting processes for passenger ships.

By utilizing Titanic Survival Dataset for Risk Modeling through:

1. Feature Analysis: The Titanic survival dataset contains information about passengers' characteristics, such as age, gender, passenger class, and family size. Insurance companies can analyze this data to identify which features were correlated with higher or lower survival rates. For example, they may find that fare class or gender class played a significant role in survival.

2. Risk Modeling: Based on the analysis, insurance companies can develop risk models that take into account the relevant features identified in the dataset. These models can be used to estimate the risk of future passenger ships based on similar characteristics.

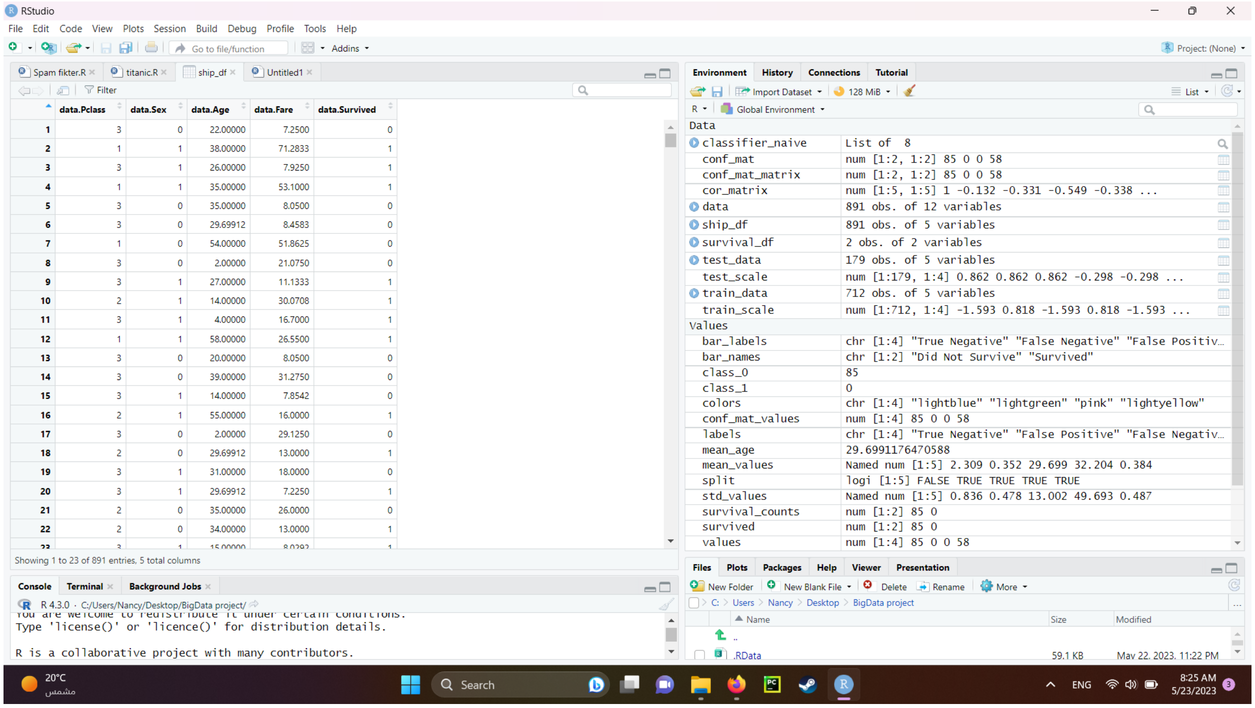
<u>Phase(2):Data Preparation</u>

- After exploring the available dataset it was identified that it consists of the following classes survival ‘0=>No// 1=>Yes’, pclass ‘Ticket class’, Sex, Age, sibsp ‘No of siblings/spouses who were abroad’, parch ‘No of parents/ children were abroad’, ticket ‘ticket number’, fare ‘passenger fare’, cabin ‘cabin number’, embarked ‘port of embarkation’.

- <u>The following transformations were done on dataset which are:</u>

Considered as a feature engineering step, It is suggested that the suitable column/ target variables for developing such a model will be focusing on (pclass, sex, age, fare, survived). These features are considered to be important features that will contribute in the model development to offer the company desired results.

Therefore, all columns were dropped from the data frame and previously mentioned columns were kept.



	data.Pclass	data.Sex	data.Age	data.Fare	data.Survived
1	3	0	22.00000	7.2500	0
2	1	1	38.00000	71.2833	1
3	3	1	26.00000	7.9250	1
4	1	1	35.00000	53.1000	1
5	3	0	35.00000	8.0500	0
6	3	0	29.69912	8.4583	0
7	1	0	54.00000	51.8625	0
8	3	0	2.00000	21.0750	0
9	3	1	27.00000	11.1333	1
10	2	1	14.00000	30.0708	1
11	3	1	4.00000	16.7000	1
12	1	1	58.00000	26.5500	1
13	3	0	20.00000	8.0500	0
14	3	0	39.00000	31.2750	0
15	3	1	14.00000	7.8542	0
16	2	1	55.00000	16.0000	1
17	3	0	2.00000	29.1250	0
18	2	0	29.69912	13.0000	1
19	3	1	31.00000	18.0000	0
20	3	1	29.69912	7.2250	1
21	2	0	35.00000	26.0000	0
22	2	0	34.00000	13.0000	1
23	3	1	15.00000	8.0292	1

Showing 1 to 23 of 891 entries, 5 total columns

```
R 4.3.0 · C:/Users/Nancy/Desktop/BigData project/
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
```

```
Data
classifier_naive    List of  8
conf_mat            num [1:2, 1:2] 85 0 0 58
conf_mat_matrix     num [1:2, 1:2] 85 0 0 58
cor_matrix          num [1:5, 1:5] 1 -0.132 -0.331 -0.549 -0.338 ...
data                891 obs. of 12 variables
ship_df             891 obs. of 5 variables
survival_df         2 obs. of 2 variables
test_data           179 obs. of 5 variables
test_scale          num [1:179, 1:4] 0.862 0.862 0.862 -0.298 -0.298 ...
train_data          712 obs. of 5 variables
train_scale         num [1:712, 1:4] -1.593 0.818 -1.593 0.818 -1.593 ...
Values
bar_labels          chr [1:4] "True Negative" "False Negative" "False Positiv…
bar_names           chr [1:2] "Did Not Survive" "Survived"
class_0             85
class_1             0
colors              chr [1:4] "lightblue" "lightgreen" "pink" "lightyellow"
conf_mat_values     num [1:4] 85 0 0 58
labels              chr [1:4] "True Negative" "False Positive" "False Negativ…
mean_age            29.6991176470588
mean_values         Named num [1:5] 2.309 0.352 29.699 32.204 0.384
split               logi [1:5] FALSE TRUE TRUE TRUE TRUE
std_values          Named num [1:5] 0.836 0.478 13.002 49.693 0.487
survival_counts     num [1:2] 85 0
survived            num [1:2] 85 0
values              num [1:4] 85 0 0 58
```

C: > Users > Nancy > Desktop > BigData project

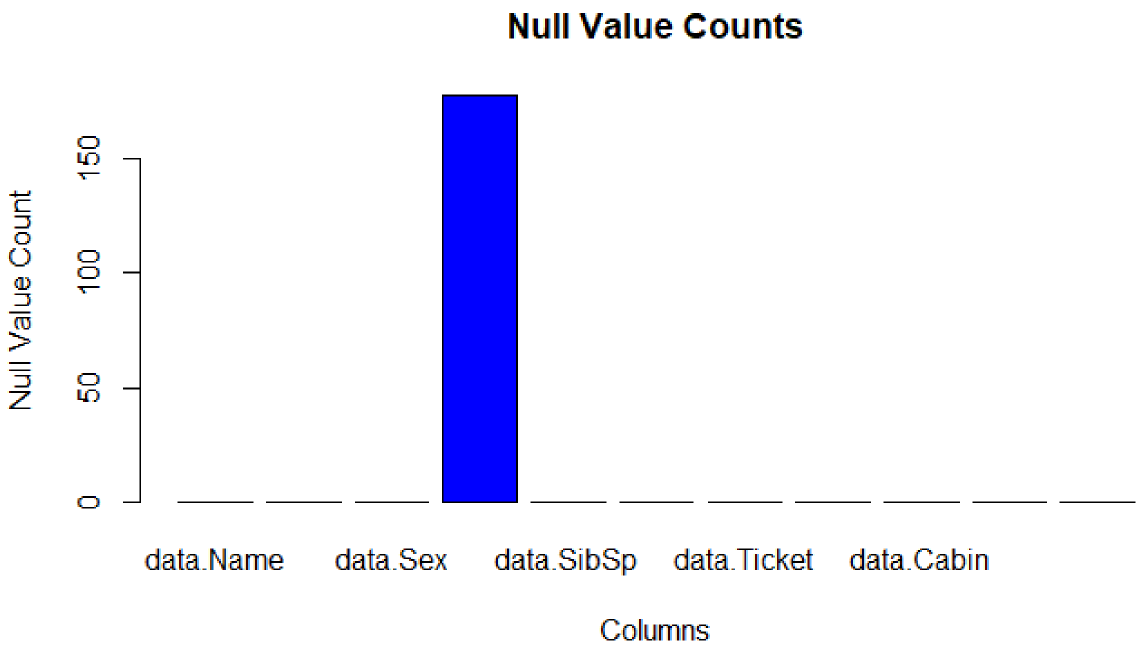
	Name	Size	Modified
	..		
□	.RData	59.1 KB	May 22, 2023, 11:22 PM

20°C مشمس

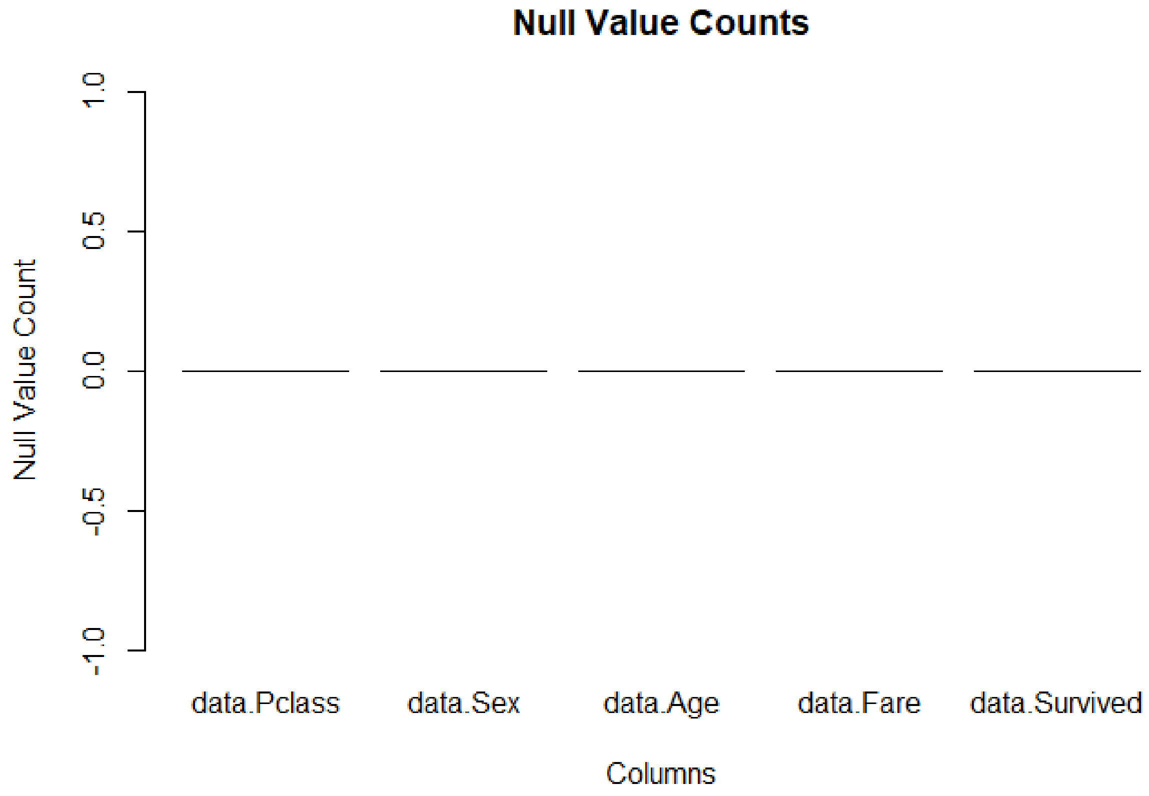
8:25 AM 5/23/2023

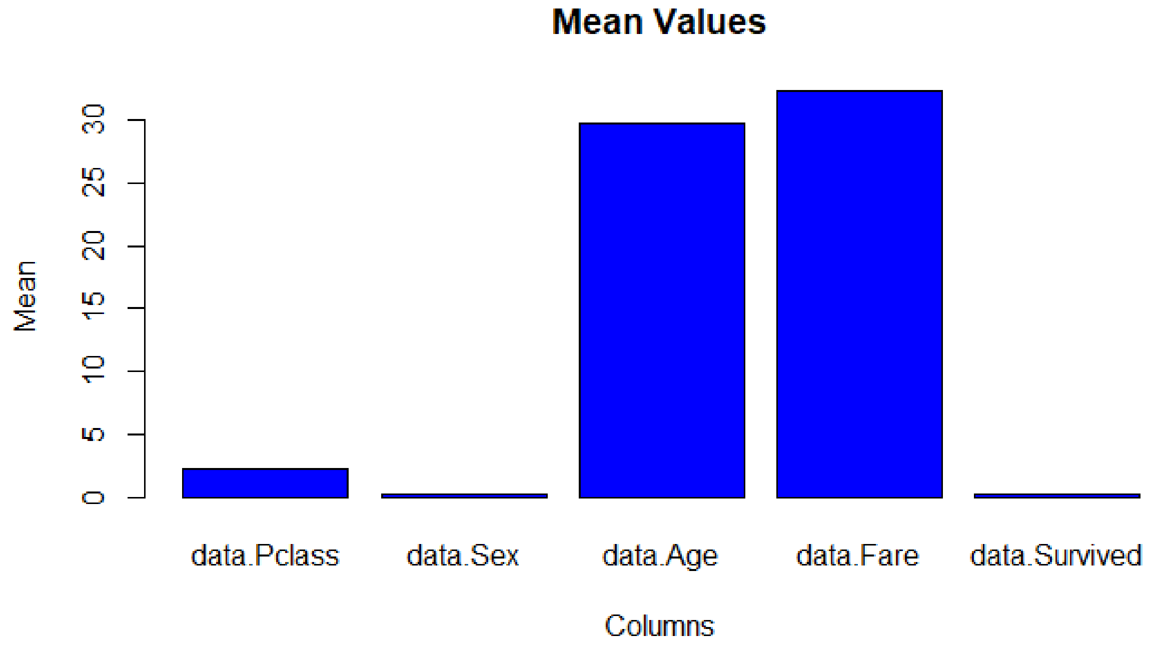
- Since it’s hard to deal with categorical data as shown in sex column, therefore male and female classes were changed into binary classes of 0=> Male and 1=>Female.

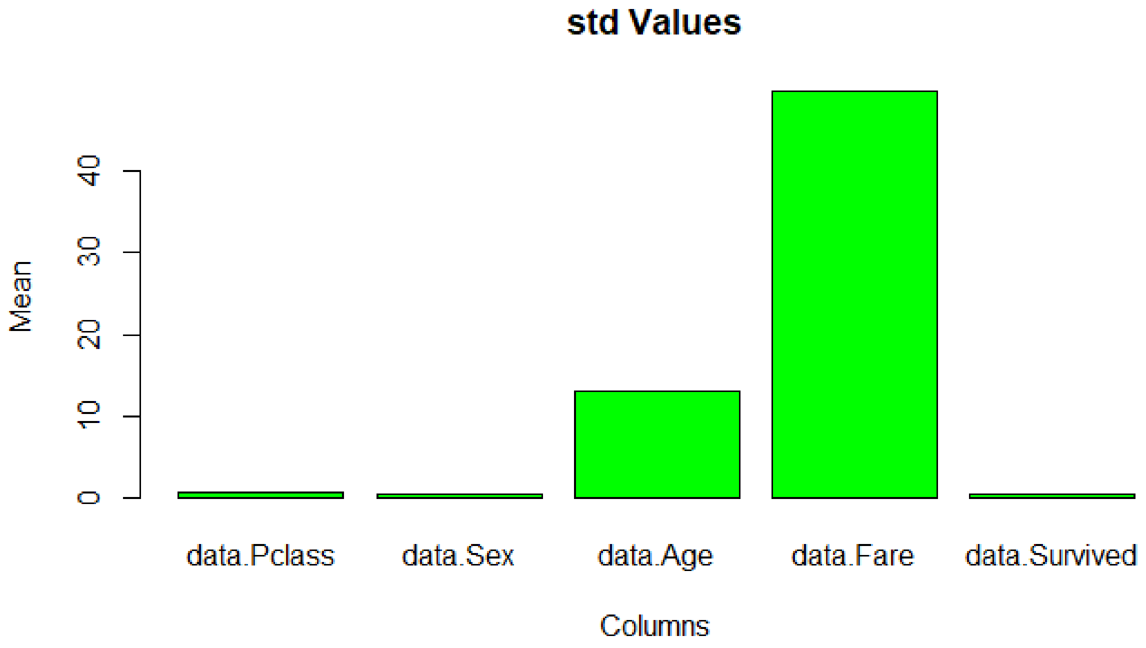
- On viewing the whole data it found nulls, especially in the Age column, we had two options. Either to omit these records or fill the empty records with the mean. The later solution was much better in order not to lose any of our data and to provide a good model.



The following bar chart is a representation for nulls in the Age column. After applying the previously explained step.

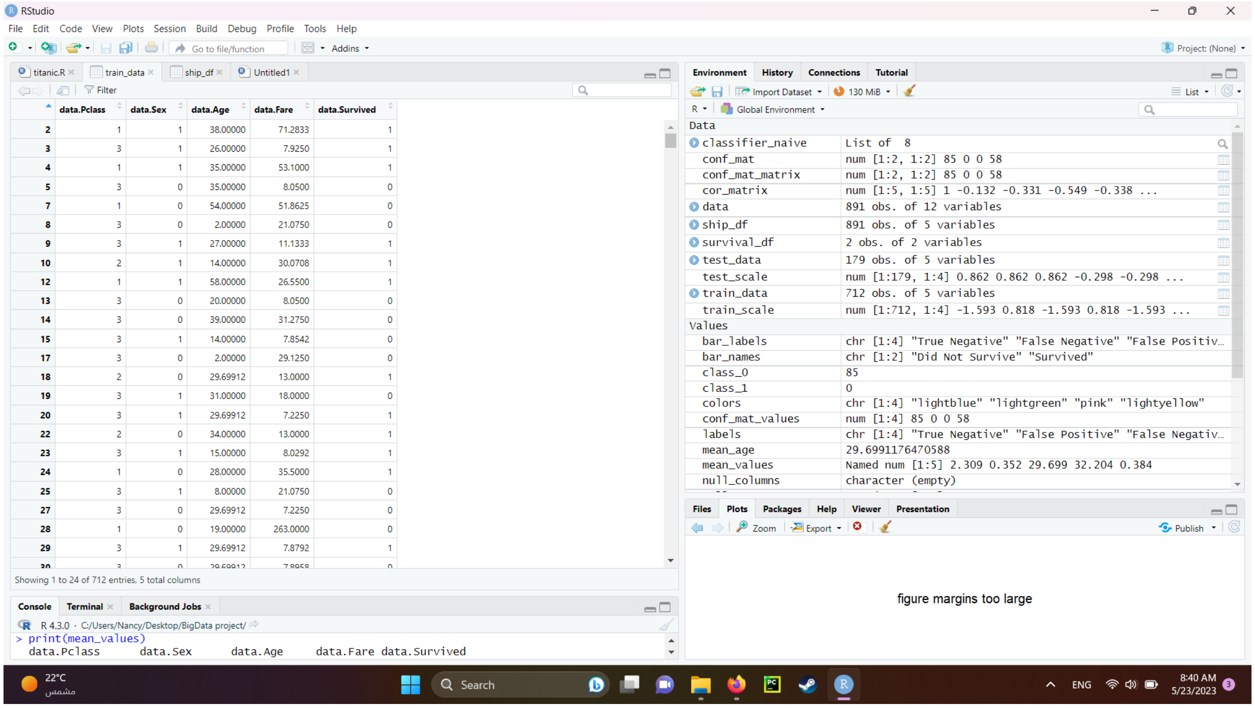

Null Value Counts
Null Value Count
1.0
0.5
0.0
-0.5
-1.0
data.Pclass
data.Sex
data.Age
data.Fare
data.Survived
Columns


Mean Values
Mean
30
25
20
15
10
5
0
data.Pclass
data.Sex
data.Age
data.Fare
data.Survived
Columns



Phase(3):Model Planning

Suggested techniques to be applied for such a problem is, since we need the model to predict based on inserted data if it’s most likely to survive or not from the titanic based on inserted data. We have applied the Naive Bayes model. By splitting our dataset into test and training sets.



	data.Pclass	data.Sex	data.Age	data.Fare	data.Survived
2	1	1	38.00000	71.2833	1
3	3	1	26.00000	7.9250	1
4	1	1	35.00000	53.1000	1
5	3	0	35.00000	8.0500	0
7	1	0	54.00000	51.8625	0
8	3	0	2.00000	21.0750	0
9	3	1	27.00000	11.1333	1
10	2	1	14.00000	30.0708	1
12	1	1	58.00000	26.5500	1
13	3	0	20.00000	8.0500	0
14	3	0	39.00000	31.2750	0
15	3	1	14.00000	7.8542	0
17	3	0	2.00000	29.1250	0
18	2	0	29.69912	13.0000	1
19	3	1	31.00000	18.0000	0
20	3	1	29.69912	7.2250	1
22	2	0	34.00000	13.0000	1
23	3	1	15.00000	8.0292	1
24	1	0	28.00000	35.5000	1
25	3	1	8.00000	21.0750	0
27	3	0	29.69912	7.2250	0
28	1	0	19.00000	263.0000	0
29	3	1	29.69912	7.8792	1
30	3	0	29.69912	7.8958	0

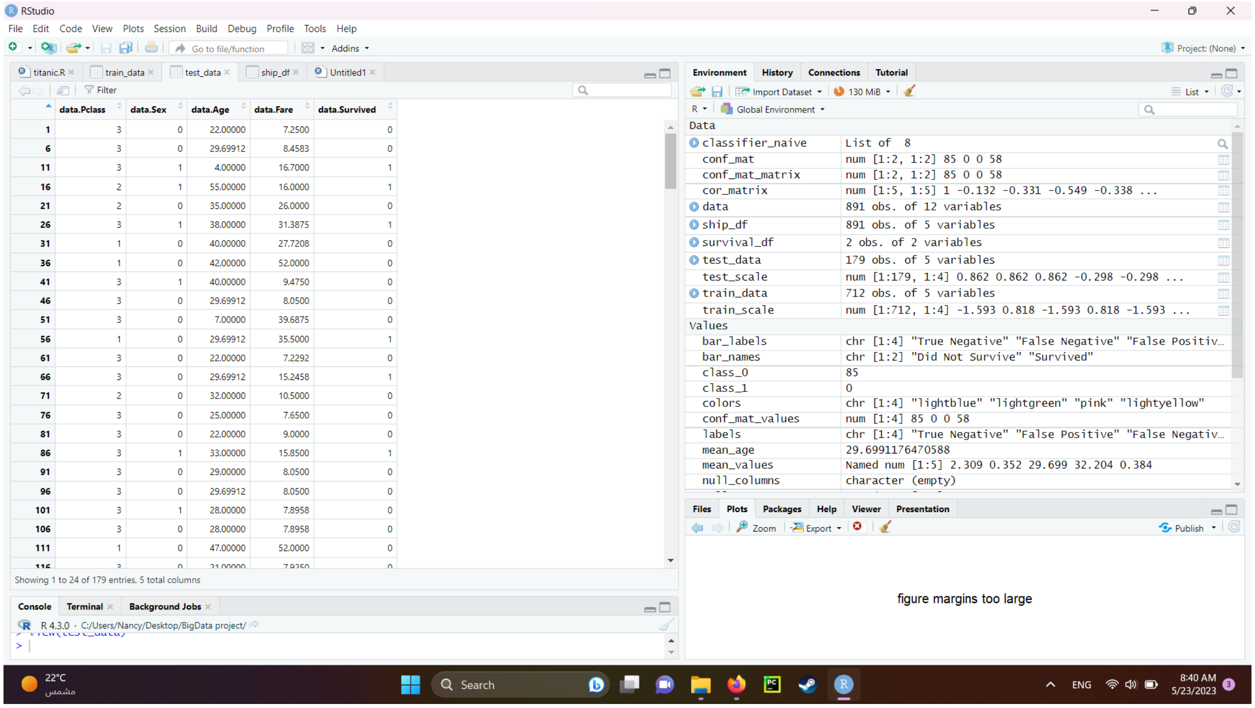
Showing 1 to 24 of 712 entries, 5 total columns

```
Data
classifier_naive    List of  8
conf_mat            num [1:2, 1:2] 85 0 0 58
conf_mat_matrix     num [1:2, 1:2] 85 0 0 58
cor_matrix          num [1:5, 1:5] 1 -0.132 -0.331 -0.549 -0.338 ...
data                891 obs. of 12 variables
ship_df             891 obs. of 5 variables
survival_df         2 obs. of 2 variables
test_data           179 obs. of 5 variables
test_scale          num [1:179, 1:4] 0.862 0.862 0.862 -0.298 -0.298 ...
train_data          712 obs. of 5 variables
train_scale         num [1:712, 1:4] -1.593 0.818 -1.593 0.818 -1.593 ...
Values
bar_labels          chr [1:4] "True Negative" "False Negative" "False Positiv…
bar_names           chr [1:2] "Did Not Survive" "Survived"
class_0             85
class_1             0
colors              chr [1:4] "lightblue" "lightgreen" "pink" "lightyellow"
conf_mat_values     num [1:4] 85 0 0 58
labels              chr [1:4] "True Negative" "False Positive" "False Negativ…
mean_age            29.6991176470588
mean_values         Named num [1:5] 2.309 0.352 29.699 32.204 0.384
null_columns        character (empty)
```

figure margins too large

```
R 4.3.0 · C:/Users/Nancy/Desktop/BigData project/
> print(mean_values)
  data.Pclass       data.Sex       data.Age      data.Fare data.Survived
```

Our training set data.



	data.Pclass	data.Sex	data.Age	data.Fare	data.Survived
1	3	0	22.00000	7.2500	0
6	3	0	29.69912	8.4583	0
11	3	1	4.00000	16.7000	1
16	2	1	55.00000	16.0000	1
21	2	0	35.00000	26.0000	0
26	3	1	38.00000	31.3875	1
31	1	0	40.00000	27.7208	0
36	1	0	42.00000	52.0000	0
41	3	1	40.00000	9.4750	0
46	3	0	29.69912	8.0500	0
51	3	0	7.00000	39.6875	0
56	1	0	29.69912	35.5000	1
61	3	0	22.00000	7.2292	0
66	3	0	29.69912	15.2458	1
71	2	0	32.00000	10.5000	0
76	3	0	25.00000	7.6500	0
81	3	0	22.00000	9.0000	0
86	3	1	33.00000	15.8500	1
91	3	0	29.00000	8.0500	0
96	3	0	29.69912	8.0500	0
101	3	1	28.00000	7.8958	0
106	3	0	28.00000	7.8958	0
111	1	0	47.00000	52.0000	0
116	3	0	21.00000	7.9250	0

Showing 1 to 24 of 179 entries, 5 total columns

```
Data
classifier_naive    List of  8
conf_mat            num [1:2, 1:2] 85 0 0 58
conf_mat_matrix     num [1:2, 1:2] 85 0 0 58
cor_matrix          num [1:5, 1:5] 1 -0.132 -0.331 -0.549 -0.338 ...
data                891 obs. of 12 variables
ship_df             891 obs. of 5 variables
survival_df         2 obs. of 2 variables
test_data           179 obs. of 5 variables
test_scale          num [1:179, 1:4] 0.862 0.862 0.862 -0.298 -0.298 ...
train_data          712 obs. of 5 variables
train_scale         num [1:712, 1:4] -1.593 0.818 -1.593 0.818 -1.593 ...
Values
bar_labels          chr [1:4] "True Negative" "False Negative" "False Positiv…
bar_names           chr [1:2] "Did Not Survive" "Survived"
class_0             85
class_1             0
colors              chr [1:4] "lightblue" "lightgreen" "pink" "lightyellow"
conf_mat_values     num [1:4] 85 0 0 58
labels              chr [1:4] "True Negative" "False Positive" "False Negativ…
mean_age            29.6991176470588
mean_values         Named num [1:5] 2.309 0.352 29.699 32.204 0.384
null_columns        character (empty)
```

figure margins too large

```
R 4.3.0 · C:/Users/Nancy/Desktop/BigData project/
> View(test_data)
> 
```

Our testing set data.

Phase(4):Model Building

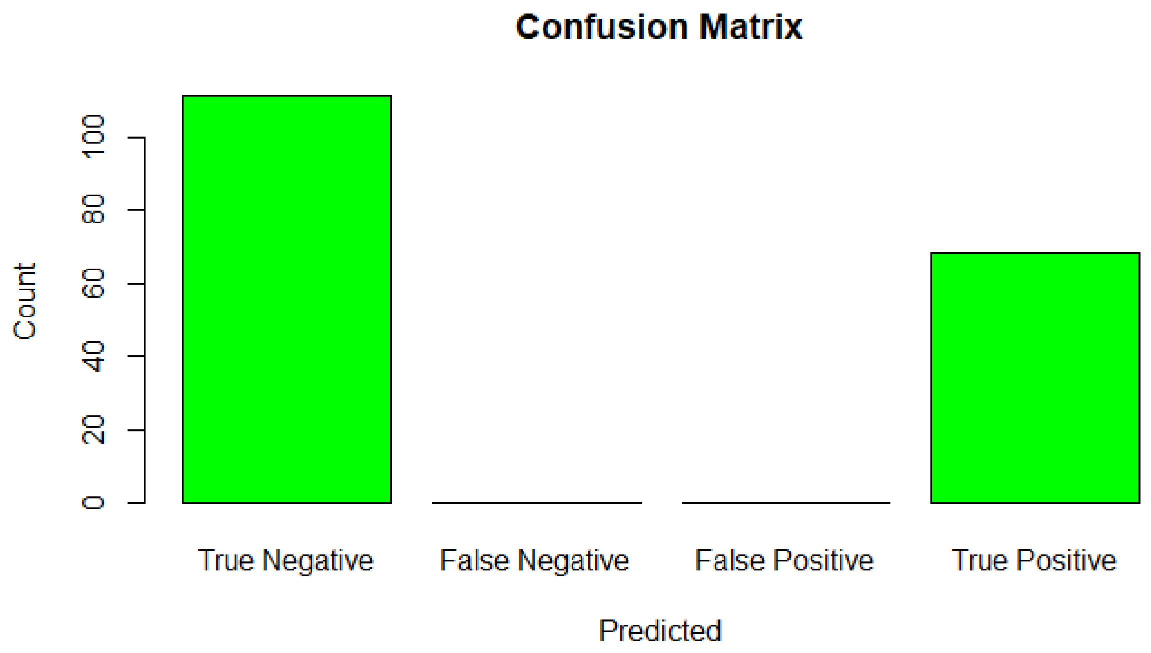
- The suitable models to use for such problem type will be the following:

    1) Naive Bayes model which provides probabilistic results
    2) Logistic regression algorithm
    3) Decision

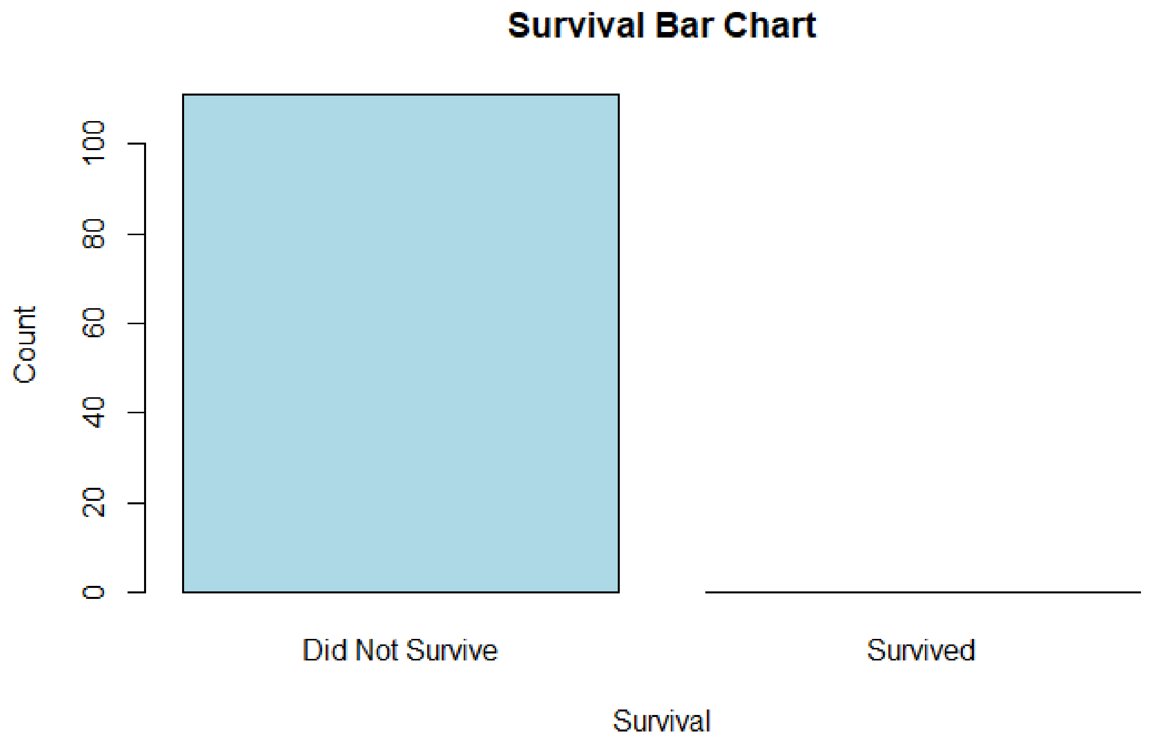
- Training and testing data have been splitted of ratio 80% train and 20% test.
- All columns are set as X variable and Survival column is set as Y variable (Target variable)..

Phase(5):Communicate results

```
Confusion Matrix and Statistics

      Predicted
Actual  0  1
     0 85  0
     1  0 58

               Accuracy : 1
                 95% CI : (0.9745, 1)
    No Information Rate : 0.5944
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 1

 Mcnemar's Test P-Value : NA

            Sensitivity : 1.0000
            Specificity : 1.0000
         Pos Pred Value : 1.0000
         Neg Pred Value : 1.0000
             Prevalence : 0.5944
         Detection Rate : 0.5944
   Detection Prevalence : 0.5944
      Balanced Accuracy : 1.0000

       'Positive' Class : 0
```



From the following bar chart it’s clear that our model has predicted all entries true as per True negative(Meaning that it has predicted this much who won’t survive) and true positive (Meaning that it predicted this much that are most likely to survive).



Phase(6):Operationalize