**Machine Learning Semester Project Status Report:**

**The Titanic Disaster**

Group-#

Members:

Ian Laurain(ICL140030)

Devendra Lad(DVL160030)

Saptharsh R Heggade (SRH140130)

Raghunandan Ramesha(RXN150230)

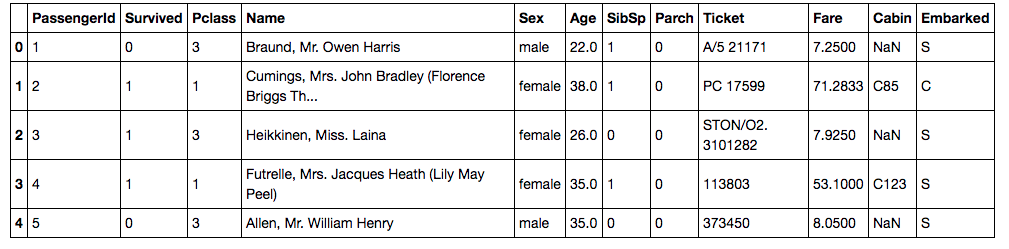
1. **The Dataset Details**

Source: <https://www.kaggle.com/c/titanic>

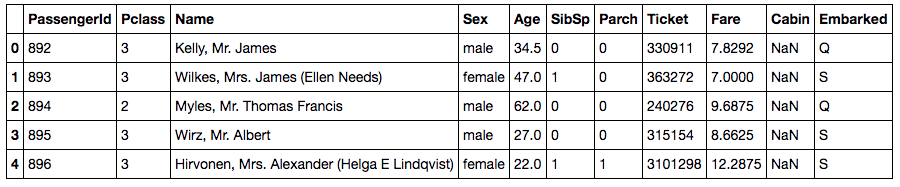
Training dataset dimensions: 891 rows & 11 features (excluding passenger ID)

Testing dataset dimensions: 418 rows & 11 features (excluding passenger ID)

Screenshot of training dataset:



Screenshot of testing dataset:



1. **Dataset description:**

VARIABLE DESCRIPTIONS:

survival Survival

(0 = No; 1 = Yes)

pclass Passenger Class

(1 = 1st; 2 = 2nd; 3 = 3rd)

name Name

sex Sex

age Age

sibsp Number of Siblings/Spouses Aboard

parch Number of Parents/Children Aboard

ticket Ticket Number

fare Passenger Fare

cabin Cabin

embarked Port of Embarkation

(C = Cherbourg; Q = Queenstown; S = Southampton)

Points to be noted:

**Pclass**: is a proxy for socio-economic status (SES)

1st ~ Upper; 2nd ~ Middle; 3rd ~ Lower

**Age is in Years:** Fractional if Age less than One (1)

Estimated age, should be of the form xx.5

With respect to the family relation variables (i.e. sibsp and parch)

some relations were ignored. The following are the definitions used

for sibsp and parch.

Sibling: Brother, Sister, Stepbrother, or Stepsister of Passenger Aboard Titanic

Spouse: Husband or Wife of Passenger Aboard Titanic (Mistresses and Fiances Ignored)

Parent: Mother or Father of Passenger Aboard Titanic

Child: Son, Daughter, Stepson, or Stepdaughter of Passenger Aboard Titanic

Other family relatives excluded from this study include cousins,

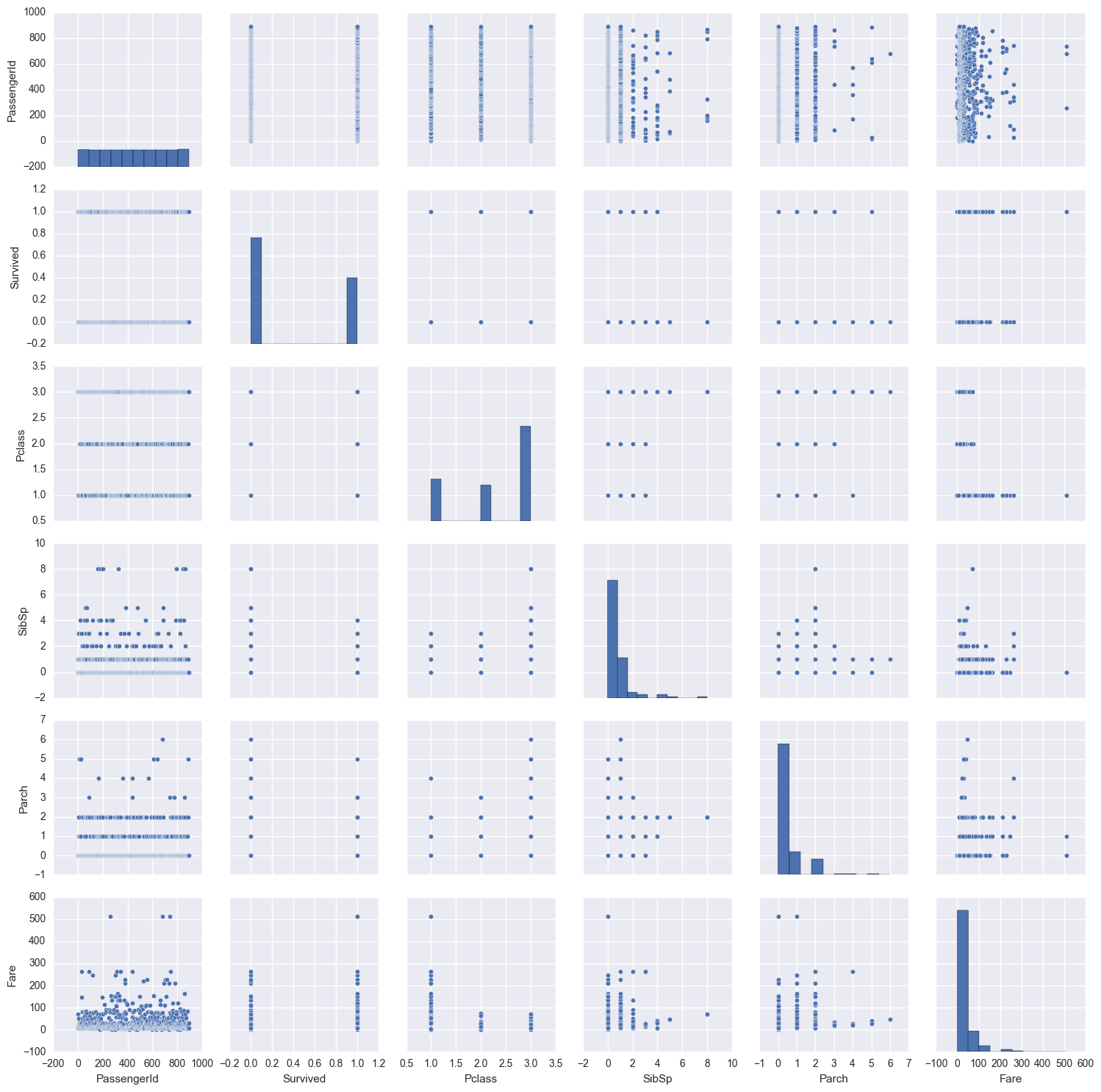
nephews/nieces, aunts/uncles, and in-laws. Some children travelled

only with a nanny, therefore parch=0 for them. As well, some

travelled with very close friends or neighbors in a village, however,

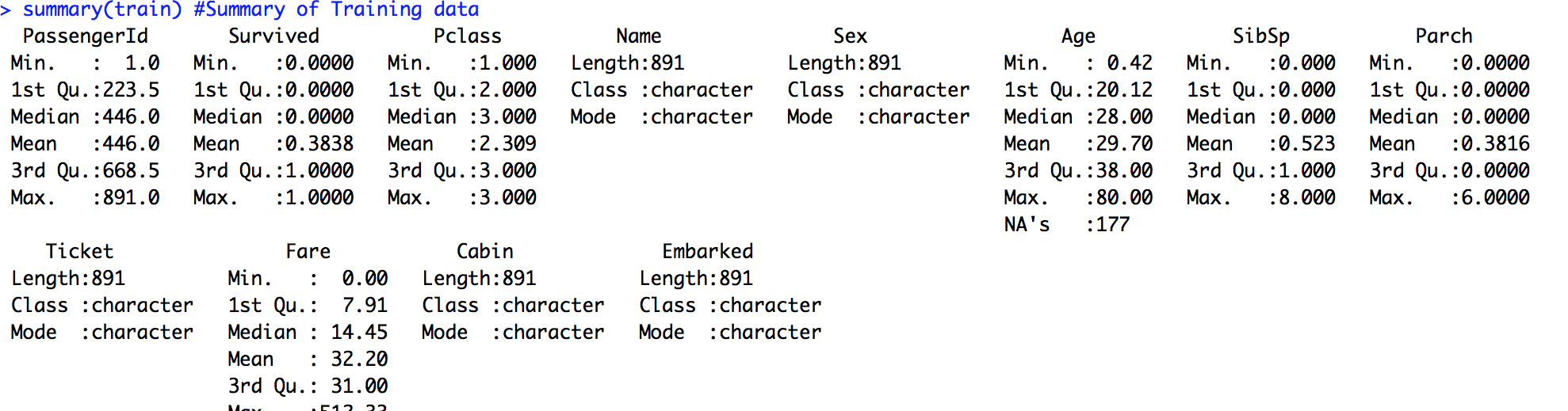
the definitions do not support such relations.

Dataset Distribution



The following pictographic representation above (made from using <http://seaborn.pydata.org>) gives an overview of the dataset description.

**Summary of Training data**



**Histogram of PClass and Age**

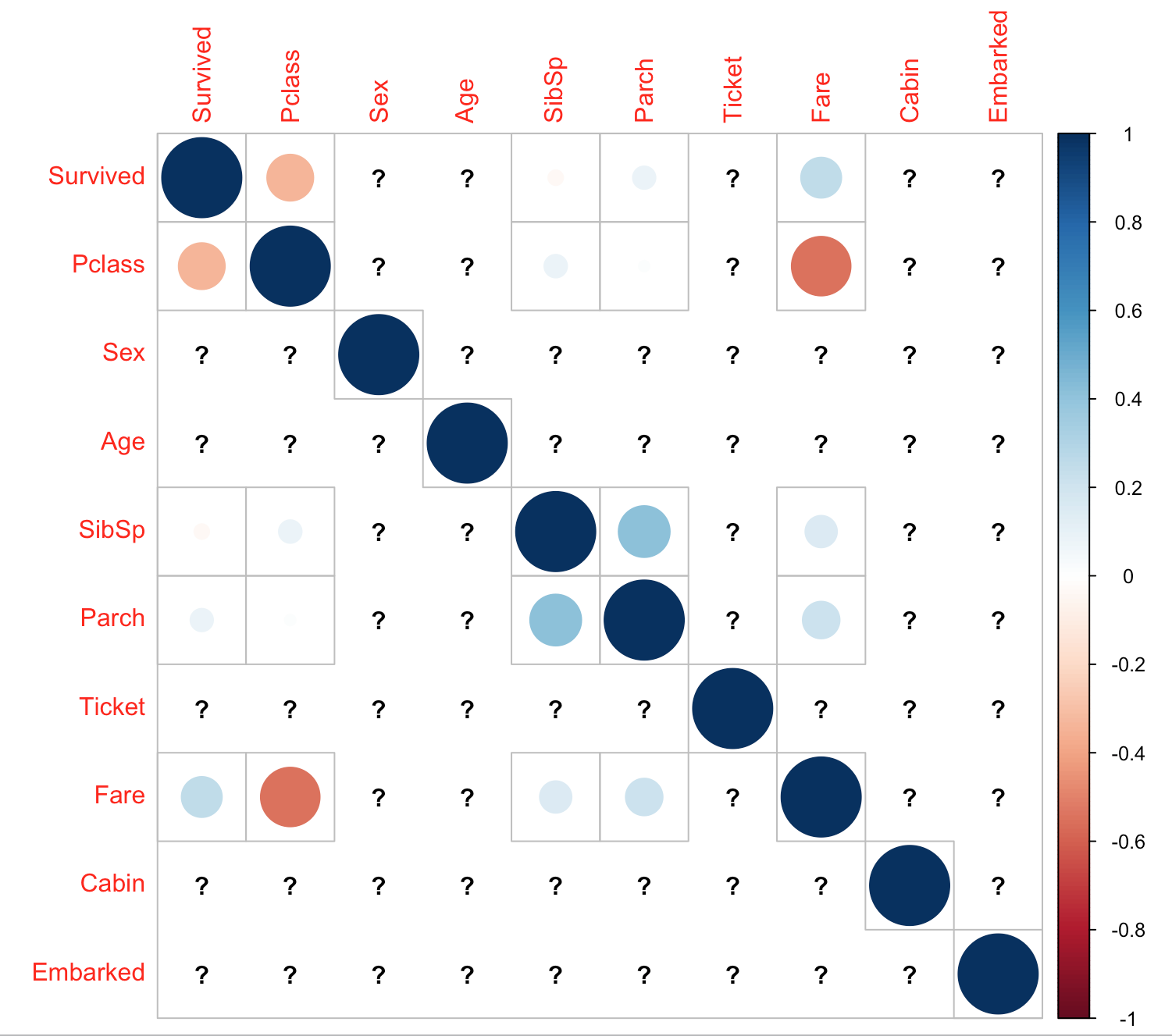




**\*\*\*Note\*\*\***

Data from cabin, age, and embarked columns were dropped because they all contain NaN values (cabin=687, age=177, embarked=2). An imputation strategy has not yet been developed, although it is planned to ignore the cabin column since such a large proportion of the instances in that column are NaN. For age, it is planned (most likely) to use the most frequent age as an imputation strategy for that column. Since the embarked column is only missing 2 values, either random values will be chosen or these rows could be ignored altogether.

**Co-Relation plot of some attributes and Survived class label**



1. Techniques planned to be implemented:

* Pre-Processing of data – The initially obtained data comprises of lot of NaN values which needs to be eliminated and the dataset should be cleaned.

The **sklearn. preprocessing** package provides several common utility functions and transformer classes to change raw feature vectors into a representation that is more suitable for the downstream estimators.

* To avoid overfitting of data. We plan to perform N-fold cross validation which ensures that a part of data is held out as test data.

Note: Using scikit-learn a random split into training and test sets can be quickly computed with the [**train\_test\_split**](http://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html#sklearn.model_selection.train_test_split) helper function.

* Training a Random Forest model with the data provided by using a suitable scripting language like R or Python.
* Use linear regression methods available in Python ML library/modules like Numpy, Pandas, & Sklearn to see if any features in the dataset allow us to predict if a passenger will survive or not.

1. Experimental methodology:

* Analyze the data

The provided data is initially analyzed to get the understanding of the data distribution and the number/type of variables involved which determine the results. Also, make clear assumptions on any hidden or unclear parameters in the dataset.

* Cleanup-Preprocessing the data

After the initial analysis of the dataset we check for any NULL/NaN values for the variables. If present, use appropriate pre-processing techniques (SkLearn library or any suitable alternative) to eliminate the junk values.

* Steps to avoid Overfitting

Divide the dataset into test-train sets appropriately to perform N-Fold Cross validation at a later stage to ensure we don’t over-fit the data to test maximum training dataset accuracy.

* Initial prediction/Intuition

Based on the initial understanding of the data and the variables and possible impact of the variables on the outcome of the result we try to predict the outcome and the extent (possible) of impact of each variable to determine the results.

* Choose Classifier /model

Training a Random Forest model with the data provided by using a suitable scripting language like R or Python

* Validation/Predicting Accuracy of results

Using the test dataset we obtained initially by making sure we didn’t over-fit the data, we perform the N-fold cross validation to predict the accuracy of the test-dataset and compare it to the training dataset to calculate the overall accuracy of the data models we used.