Titanic Mini Project

-SYB Tech COMP C

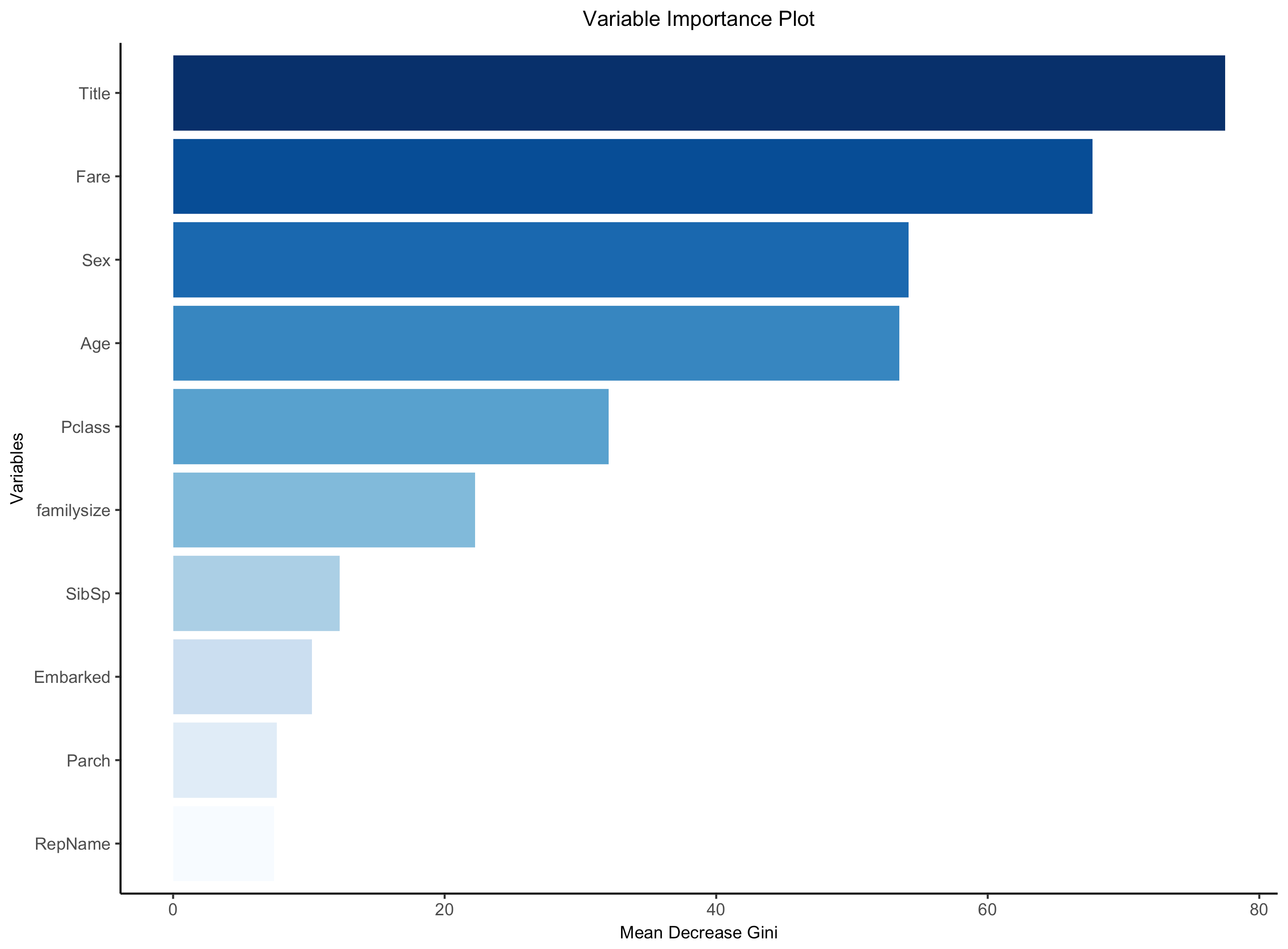
The following project was about predicting whether a person on board will sink or survive. This prediction was to be done on the basis of a data set including list of all passengers on board and their following information like seat number, class they belonged to, gender, family members etc. We have used language – PYTHON for writing the code for predicting this. Random Forest Classifier Algorithm is used on the data set.

Training dataset and Test dataset:

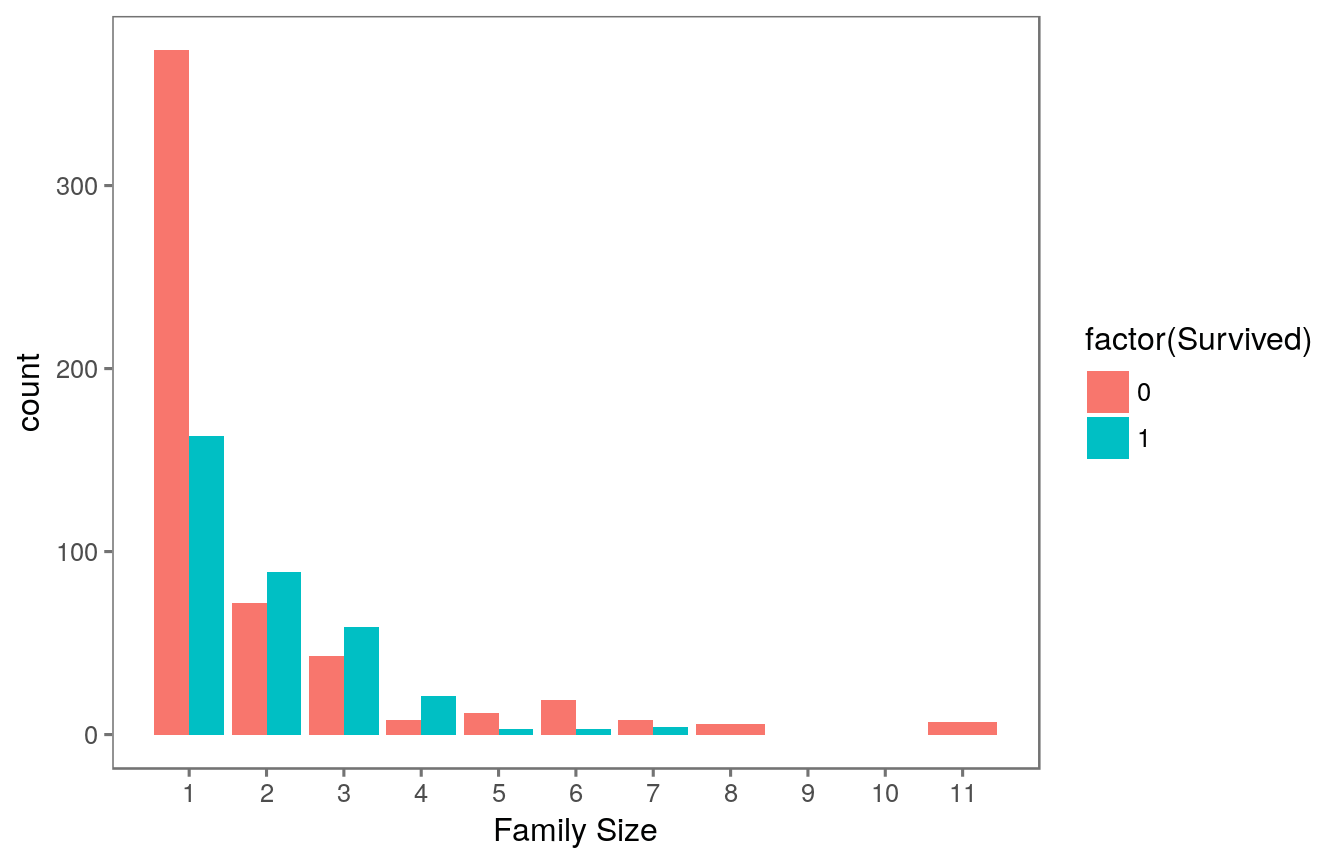
Data used to build the model comes from dataset. Usually we need two dataset to create model. The model is initially trained on training data set using supervised learning method. Training dataset is dataset of examples used for learning. These training data consist of pairs of input vector corresponding to answer vector known as label. Model is then run with training dataset and produce result, which is compared with target. Based on result and algorithm used, parameters of model are adjusted to achieve high accuracy. Test set is dataset independent of training dataset, and doesn't contain answer vector i.e. label as training dataset, but follows same probability distribution as the training dataset. Finally test dataset used to provide an unbiased evaluation of a final model fit on training dataset. Minimal over fitting indicates that model fits well to test dataset.

**The input features used are as follows:**

* PassengerID – This is an Unique ID given to every passenger.
* Survived – 0 for deceased and 1 for survived
* Pclass – Passenger class ranging from 1 to 3
* Name – Passenger full name with honorifics
* Sex – Male or female
* Age – Age of the passenger
* SibSp – number of siblings or spouses on board related to the specific passenger
* Parch – number of parents or children on board related to the specific passenger
* Ticket – Ticket number
* Fare – Ticket fare
* Cabin – Cabin alloted to the passenger
* Embarked – Port of embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)



The first variable to the attention is **passenger name** because we can break it down into additional meaningful variables which can feed predictions or be used in the creation of additional new variables. For instance, **passenger title** is contained within the passenger name variable and we can use **surname** to represent families. Further we can see that almost more than 65% passengers are with families, so we now need to find whether family members affect the survival of the passengers or not.



Following chart shows us that we can see that there’s a survival penalty to singletons and those with family sizes above 4. We can collapse this variable into three levels which will be helpful since there are comparatively fewer large families. So **discrete family size** variable is created.

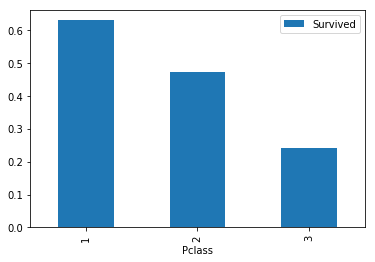
The first problem we face is that of missing values. Missing data in the training data set can reduce the power / fit of a model or can lead to a biased model because we have not analyzed the behavior and relationship with other variables correctly. It can lead to wrong prediction or classification. A lot of columns and rows from the imported data set are missing. These missing values can be predicted by analyzing the available data set in various patterns. For e.g. we can take into account the fare paid by a passenger to predict his/her boarding location and destination. The fare can also help us conclude the class of the passenger. Such predictions are done with the help of following techniques-

1. Mean, mode, median calculations
2. Mice package importing-multiple imputation using chained equations
3. R-part-recursive partitioning for regression

So basically the missing data is rectified using imputation.

After all the missing data is rectified we perform the reduction of domain. It includes the visualization of relatively important variables. This helps in increasing the accuracy and precision. The entire output depends on the quality of data set provided. Thus providing an accurate data set is extremely necessary.

In order for this to be useful to our machine learning model, we can separate this continuous feature into a categorical feature by dividing it into ranges. We have use the pandas.cut().

The next important variable was Passenger class that is the class in which the passenger belonged. From actual facts it is known that the higher class passengers were more likely to survive. This can also be proved by plotting a bar graph based on the data available

We have to then check for the importance of the ‘Sex’ variable.

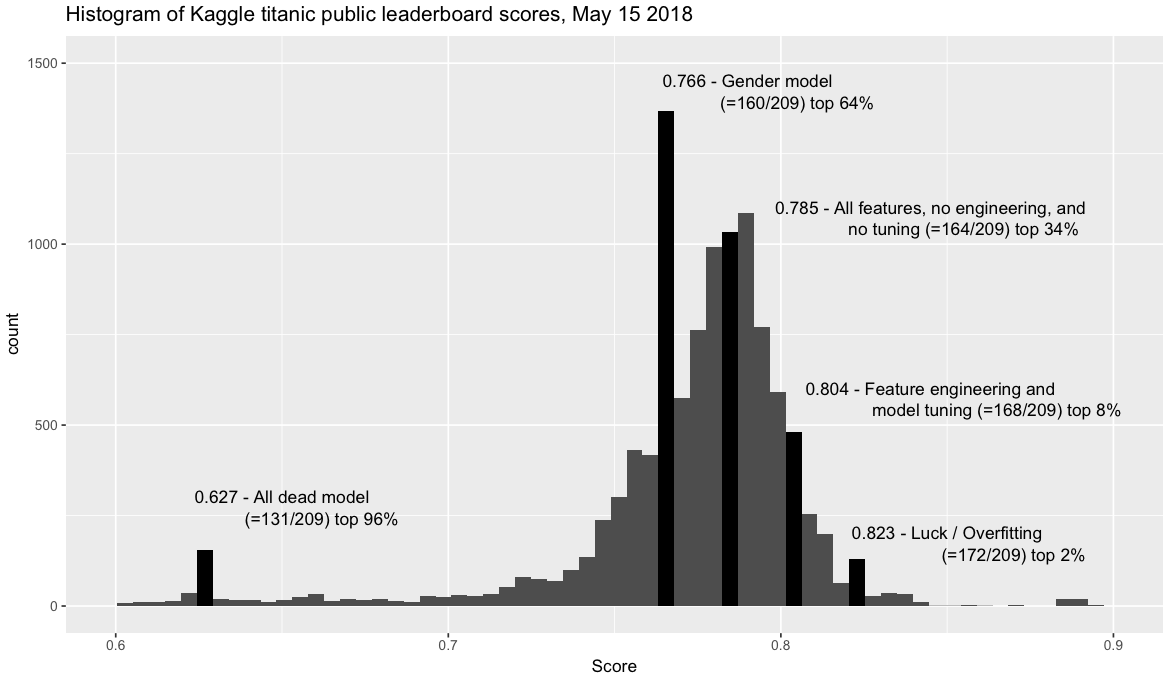
For this we use the pivot\_table() function to see who is more likely to survive.

The outcome was as follows.

|  |  |
| --- | --- |
| Sex | Survived |
| female | 0.742038 |
| male | 0.188908 |

Thus it can be seen that females survived more than the males, hence it can be a deciding factor between the chances of survival.

COMPARISONS BETWEEN DIFFERENT MODELS:



From the above chart it is clear how the accuracy depends on the importance we give to a variable. If we see the chart the accuracy goes on increasing with appropriate data set and feature engineering. On the top we can see highest accuracy is being given to the gender model as females have survived more than the males so it has the highest accuracy.

The main take away from the entire study of titanic code is to have maximum accuracy of prediction from our model, our data sets should be apt, and algorithms should be used appropriately. Minimizing garbage, importing all necessary features (Python language), etc is to be done properly. The base for creating a model of our own, and having a clear sight as to how the things work out for predicting any particular result, uses of algorithms, variable importance, creating an apt data set etc, are the concepts we clearly understood through this project.

This entire code is studied by the group. As we are beginners in Python language, we were unable to recreate the code/write our own. Although we have studied and understood this code thoroughly, and are confident that we would be able to code our own model of prediction using Random Forest or any other algorithm.