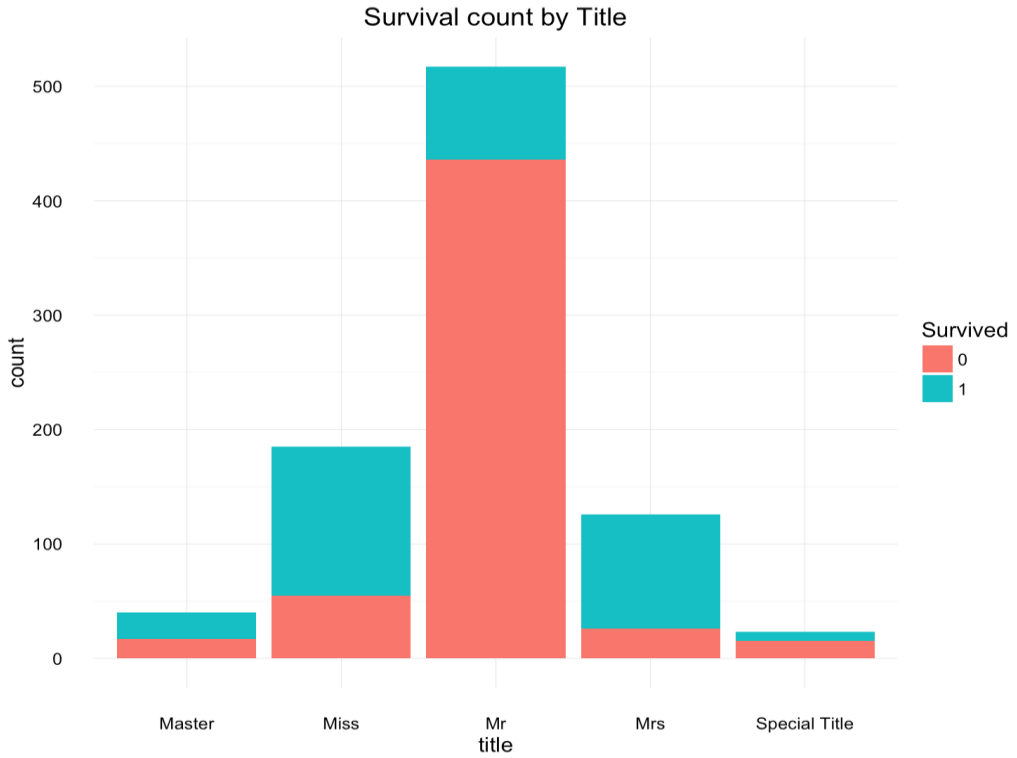
Who lives – who dies – Is this a story?

The given train data set contained variables that could influence the survival. I used following independent variables for my analysis:

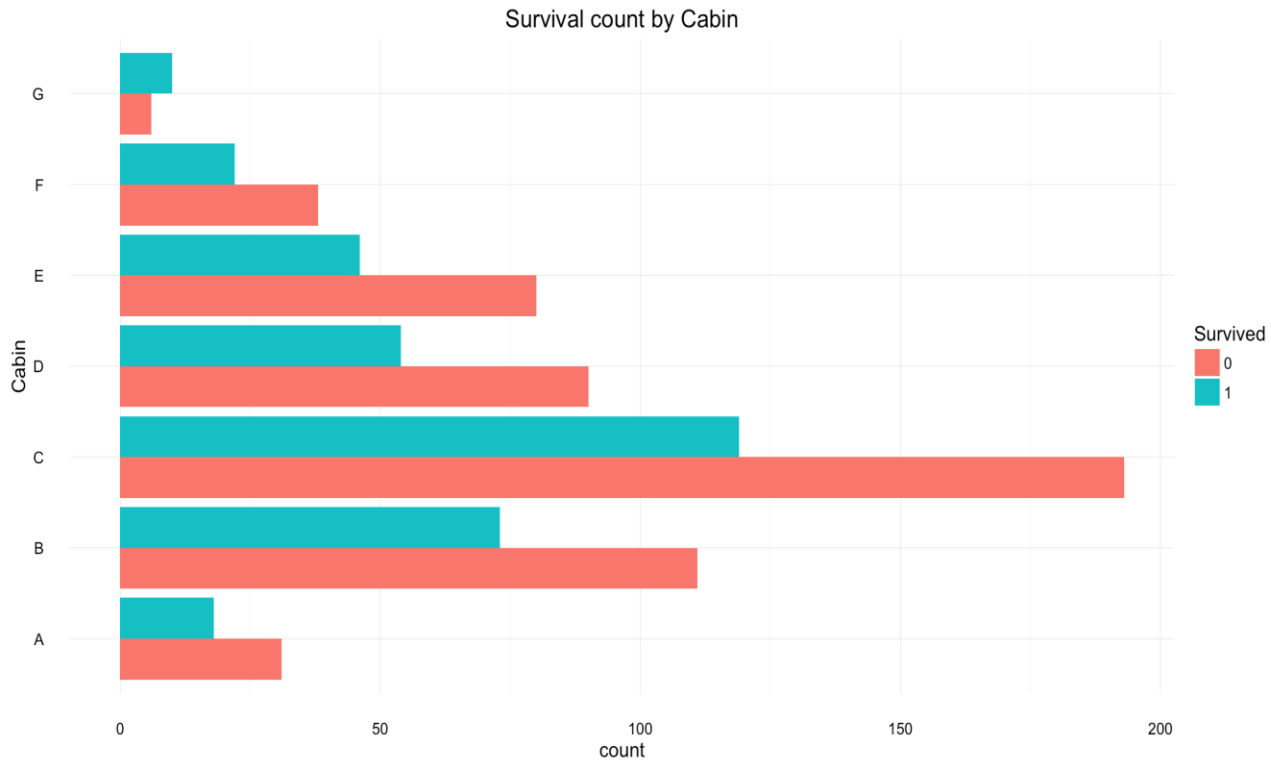
1. Pclass: No missing value, used as it is.
2. Title: New variable formed out of the variable 'Name' by partially using the title. It includes factors: Mr, Mrs, Miss, Master, Rev, Sol, Dr. etc. But there are very few entries in the special titles like Dr, Sol, Rev, so combining those titles into 'Special Title'
3. Sex: Used as it is. Male, Female
4. Age: used as it is, imputed missing values using linear regression
5. family_size: made by adding SibSp and Parch and 1 for person himself.
6. Fare: used as it is, imputed missing value
7. Cabin: used the first alphabet of cabin for the values present. Absent values were replaced with 'unknown'
8. Embarked: used as it is, imputed missing value

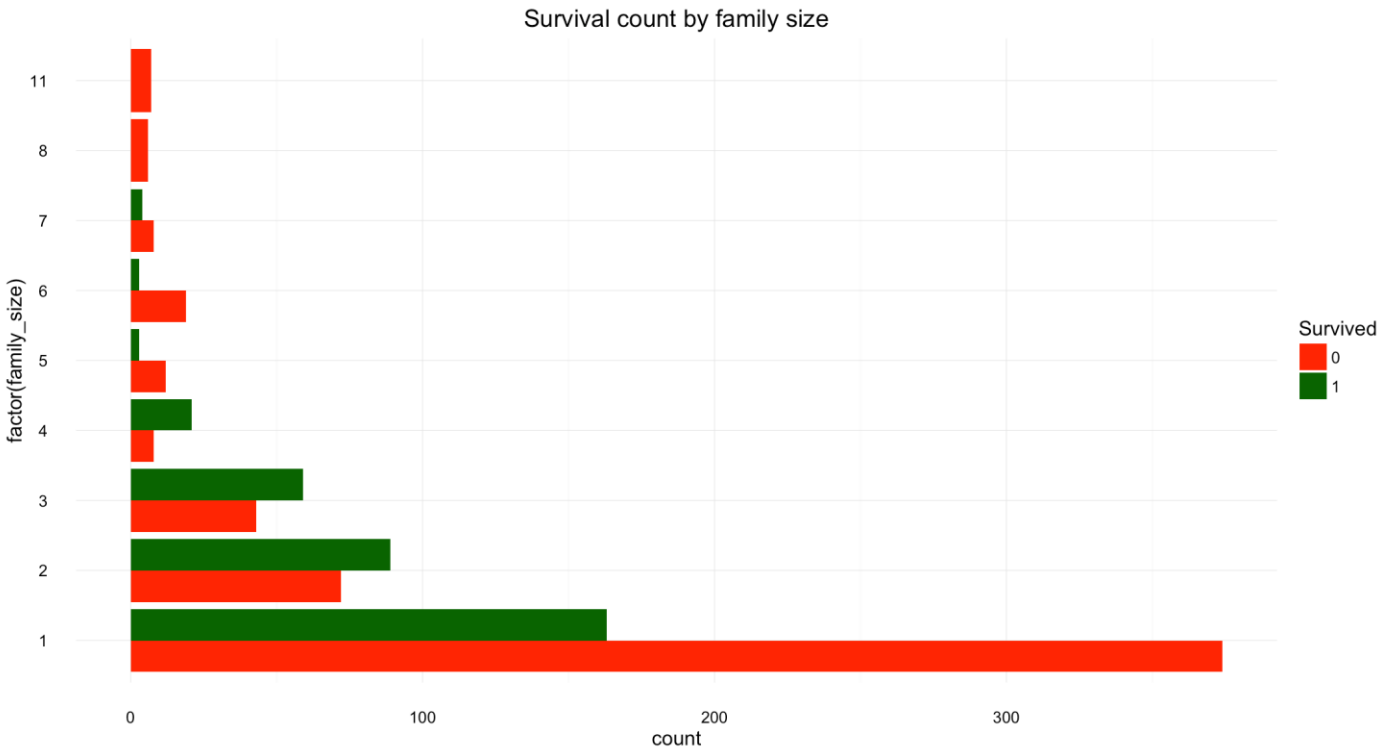
The details on imputation can be found below.

Now let's see how the Survival rate varies by different variables. As can be seen in the graph below, the highest deaths were for people with title, 'Mr.'



This graph shows that the survival was also dependent upon the Cabin people were in, but was not directly related.

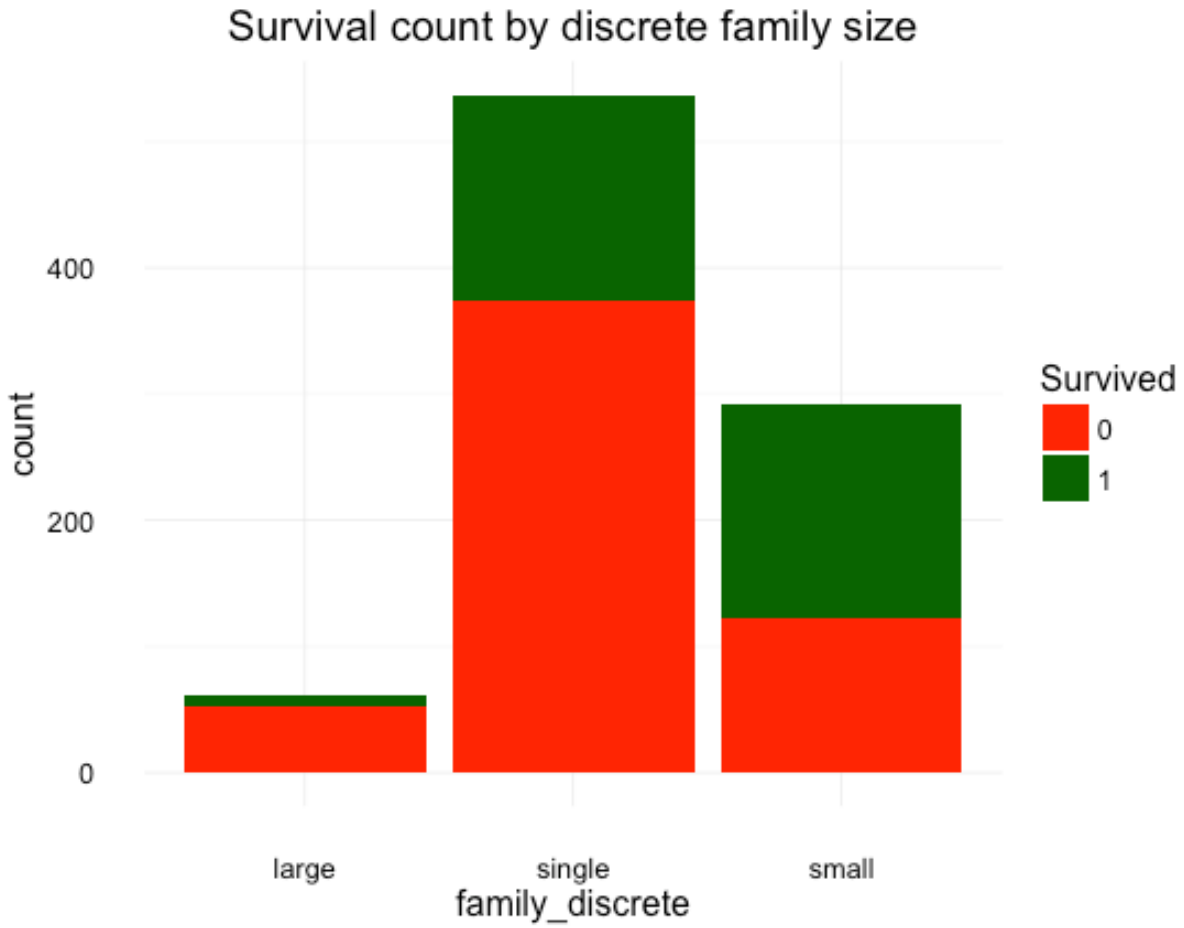




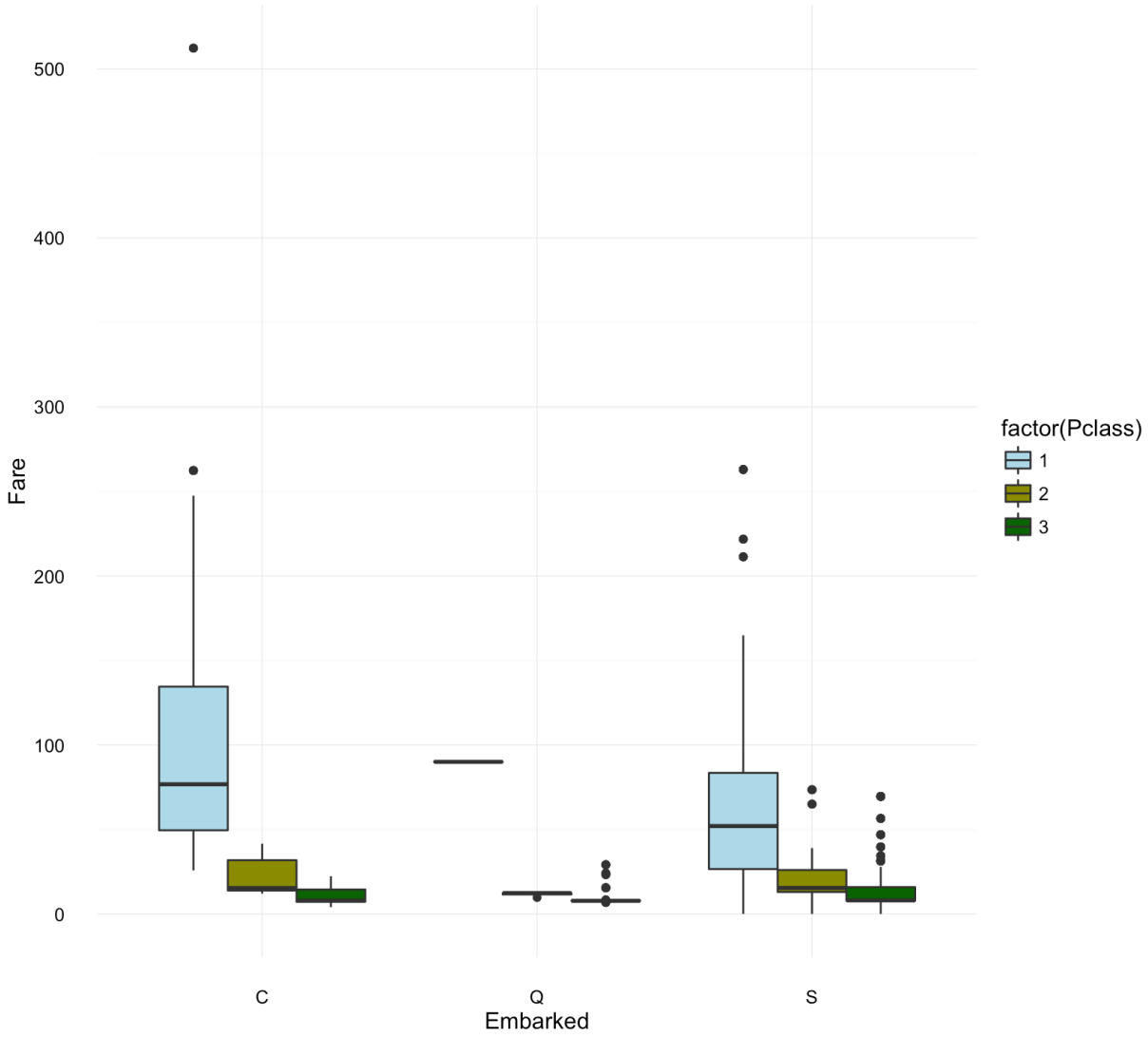
The graph above shows the variation in survival based on family size.

Seems like there are three categories of family size here. We might be able to "categorize" the data by family size as:

single – family_size =1
small- family_size =2 or 3 or 4
large- family_size>4



The fare is related to the class of the passenger and embark point. We can look at how fares vary with class and embark in the following graph (removing any absent value for either of them).



Now looking into the Embark variable, there are 2 values that are absent in data. Those are:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	title	family_name	family_size	family_discrete
62	62	1	1	Icard, Miss. Amelie	female	38	0	0	113572	80	G		Miss	Icard	1	single
830	830	1	1	Stone, Mrs. George Nelson (Martha Evelyn)	female	62	0	0	113572	80	C		Mrs	Stone	1	single

Both of them has Pclass 1 ticket and fare is $80. Looking at the graph above, the class 1 passengers with median fare $80 belongs to Embark C. So Imputing these to Embark ‘C’.

Looking into the data to find other missing values:

```
> summary(is.na(full))
 PassengerId      Survived         Pclass           Name            Sex
 Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical
 FALSE:1309      FALSE:891       FALSE:1309      FALSE:1309      FALSE:1309
 NA's :0         TRUE :418       NA's :0         NA's :0         NA's :0
                 NA's :0
    Age             SibSp           Parch           Ticket          Fare
 Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical
 FALSE:1046      FALSE:1309      FALSE:1309      FALSE:1309      FALSE:1308
 TRUE :263       NA's :0         NA's :0         NA's :0         TRUE :1
 NA's :0                                                         NA's :0
   Cabin           Embarked        title           family_name     family_size
 Mode :logical   Mode :logical   Mode :logical   Mode :logical   Mode :logical
 FALSE:1309      FALSE:1309      FALSE:1309      FALSE:1309      FALSE:1309
 NA's :0         NA's :0         NA's :0         NA's :0         NA's :0

 family_discrete
 Mode :logical
 FALSE:1309
 NA's :0
```

As we can see from the summary, Fare has only one missing value. Looking into that row:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	title	family_name	family_size	family_discrete
1044	1044	*NA*	3	Storey, Mr. Thomas	male	60.5	0	0	3701	*NA*	D	S	Mr	Storey	1	single

Again given the Embark, and Pclass, we can impute this values taking help of boxplot on previous page and finding median for this category – which for fare is $8.05.

Finally, for imputing age, we will use linear regression model and predict approximate age. Using this model, we predict the approximate age for imputation.

Finally, the variables that makes sense to be used in our model are:
Pclass, Sex, Age, SibSp, Parch, Fare, Embarked, title, Cabin, family_discrete

MODELS

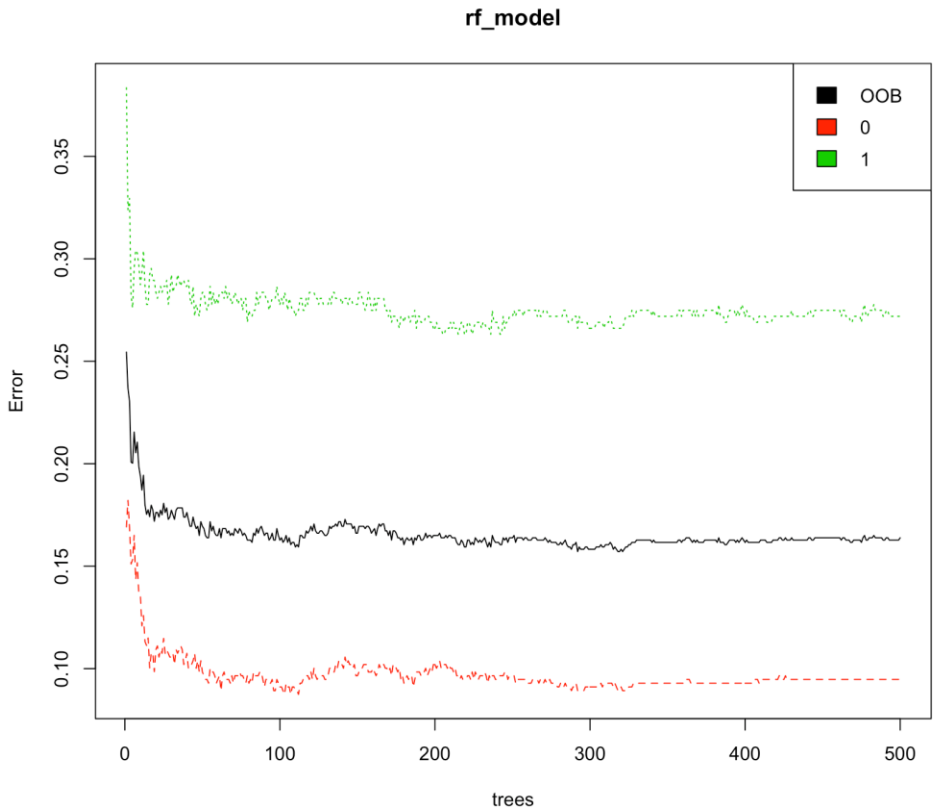
Random Forest:

We have a total of 891 rows in our train data set. To validate the test error, I'll be holding off 200 rows, and will be building the model on 691 rows.

Secondly, we will be using all set of variables to build the initial model.

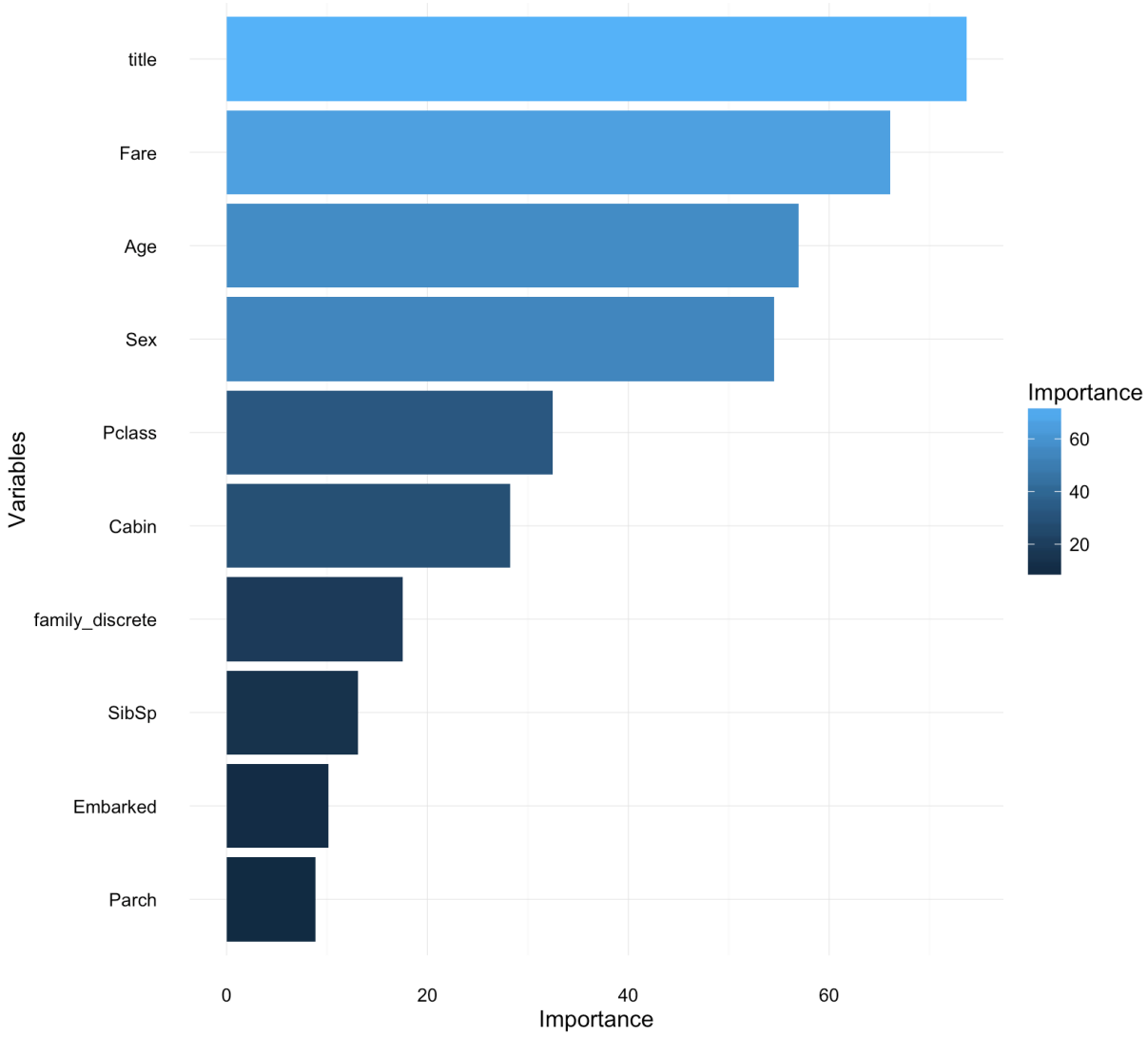
```
rf_model <- randomForest(Survived ~ Pclass + Sex + Age + SibSp + Parch +
                         Fare + Embarked + title + Cabin +
                         family_discrete, data = train)
```

The OOB error is estimated around 17.08%. Also, the error made in predicting survival(Survival=1) is higher than predicting death(Survival=0).



The black line shows the Out of Bag error rate which falls below 20%. The red and green lines show the error rate for 'died' and 'survived' respectively. We can see that right now we're much more successful predicting death than we are survival.

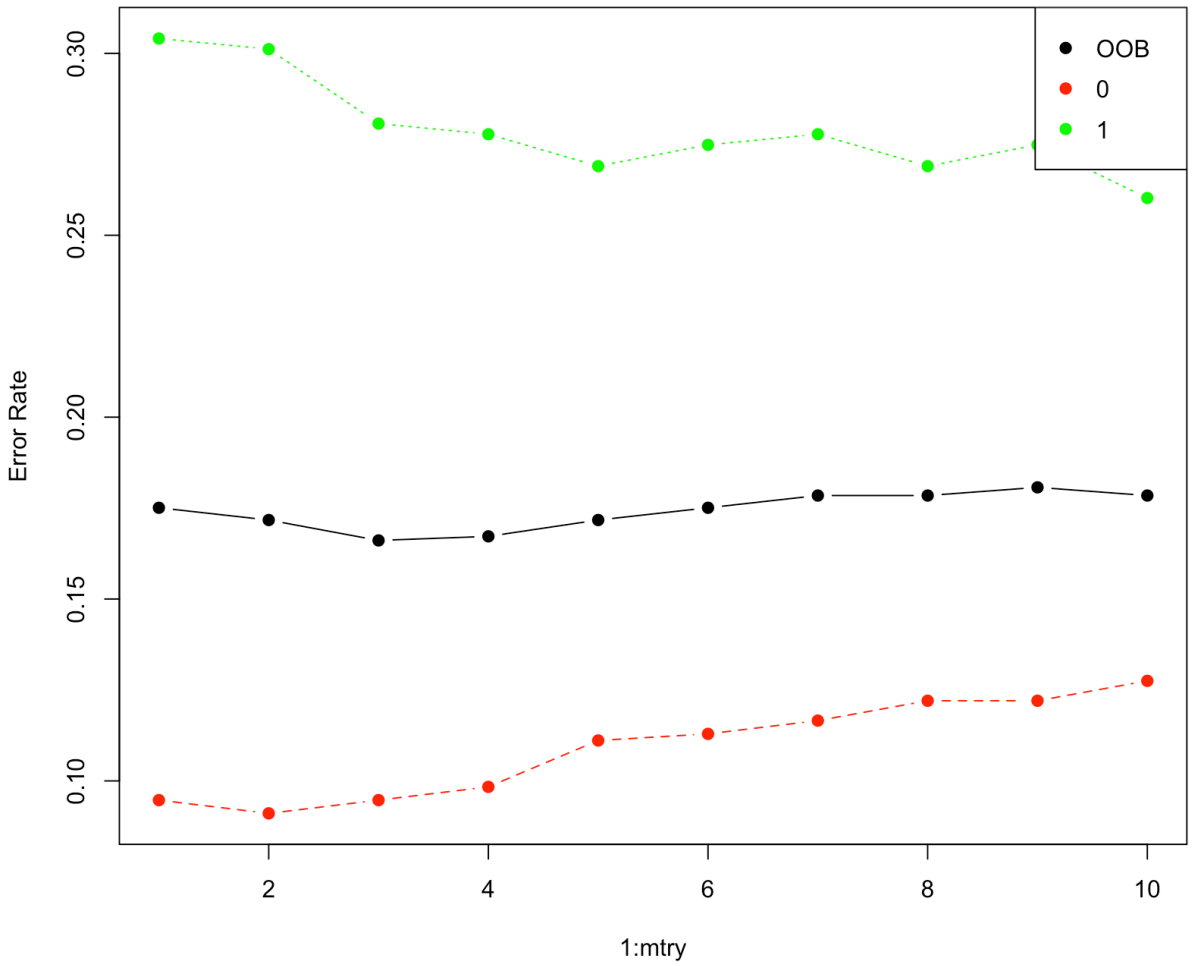
Also, the error rate almost flat out after number of trees are greater than 300. Let's fix the number of trees to 400.



As can be seen from the figure above, title seems to be the most important variable and Parch is the least. It was initially assumed that Pclass will be a very important variable for Survival but to my surprise it's 5th most important.

Model Tuning:

Now we will try to prune the tree. We built separate trees by fixing the **mtry** from 10 variables to 1 variable. The error rate varied as:



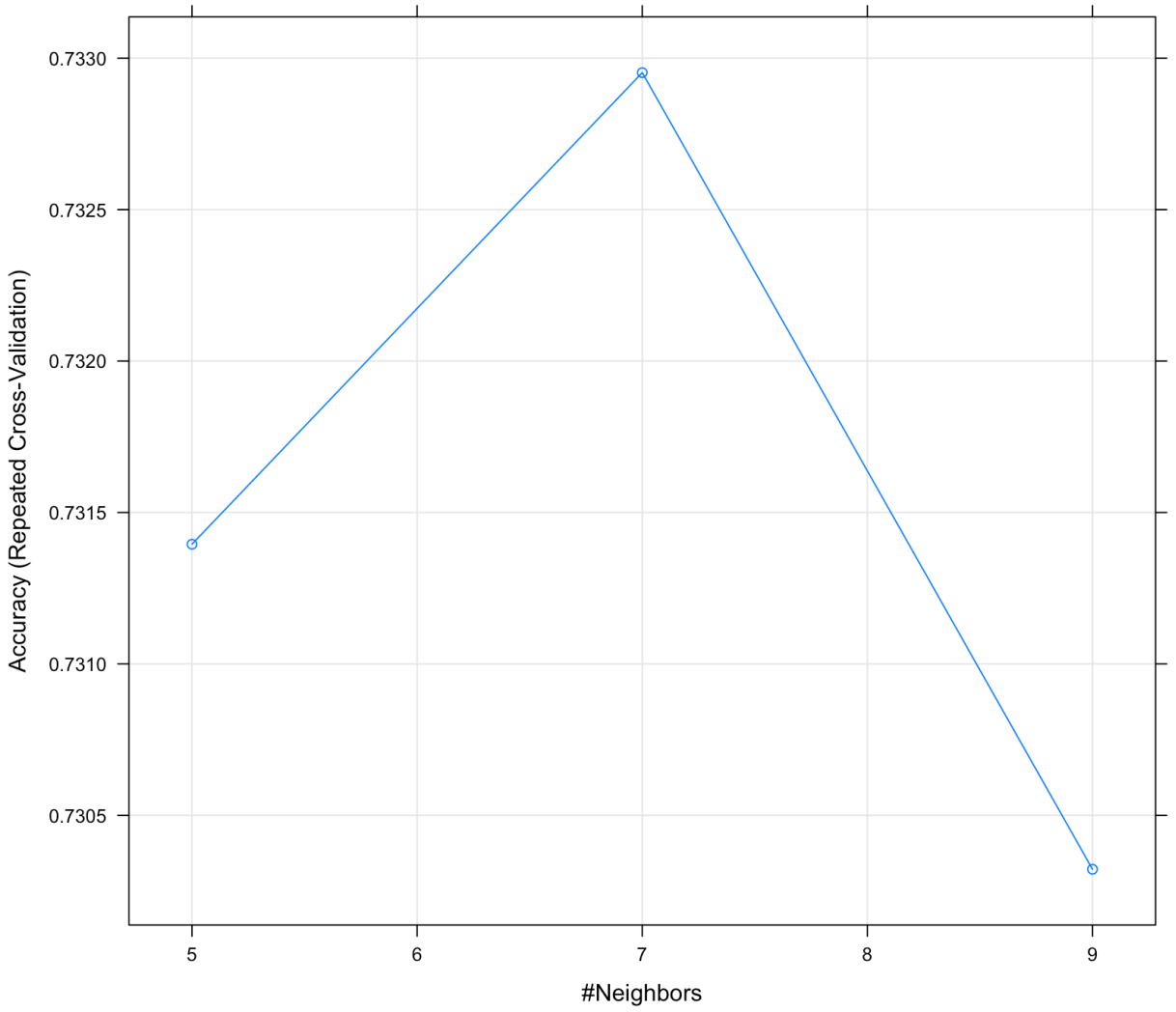
As can be seen, the **best size seems to be 5** for low error rate for Survival = 1 and little tradeoff with the Survival = 0 prediction and to have low bias. Based on the value, we tune the model to mtry=5.

Prediction on the test file resulted in 0.79904. This was lower than our training set accuracy of about 0.84.

K Nearest-Neighbour:

Again we used all the variables initially and fitted the model and validated the results using 10-fold cross validation. After iterations, we chose k=5 as the best tuning parameter.

The details of the model are:

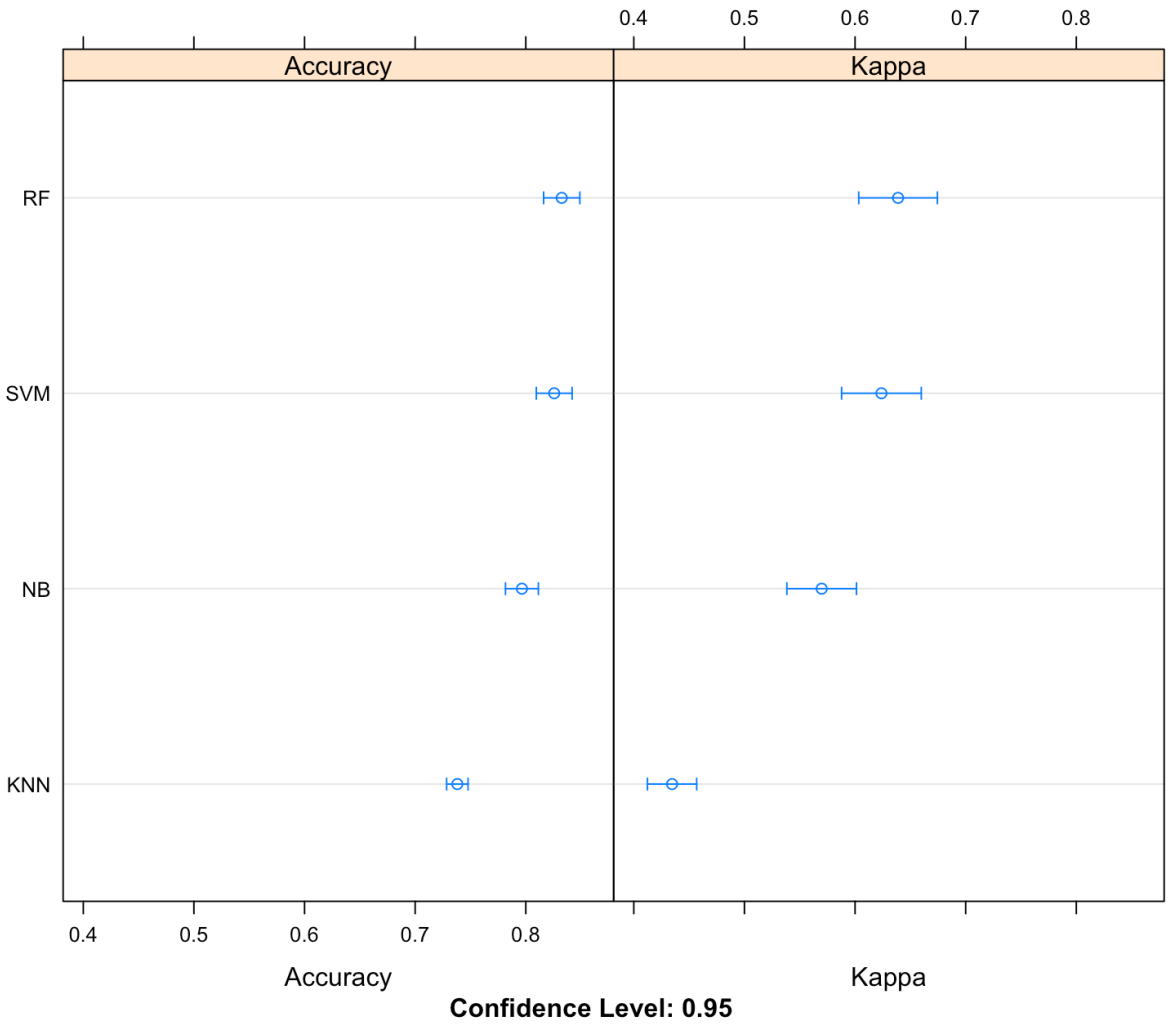


As can be seen from the figure above, the y axis represents the accuracy obtained by the models with different values of k that is represented on the x-axis. The best accuracy is obtained for k=7.

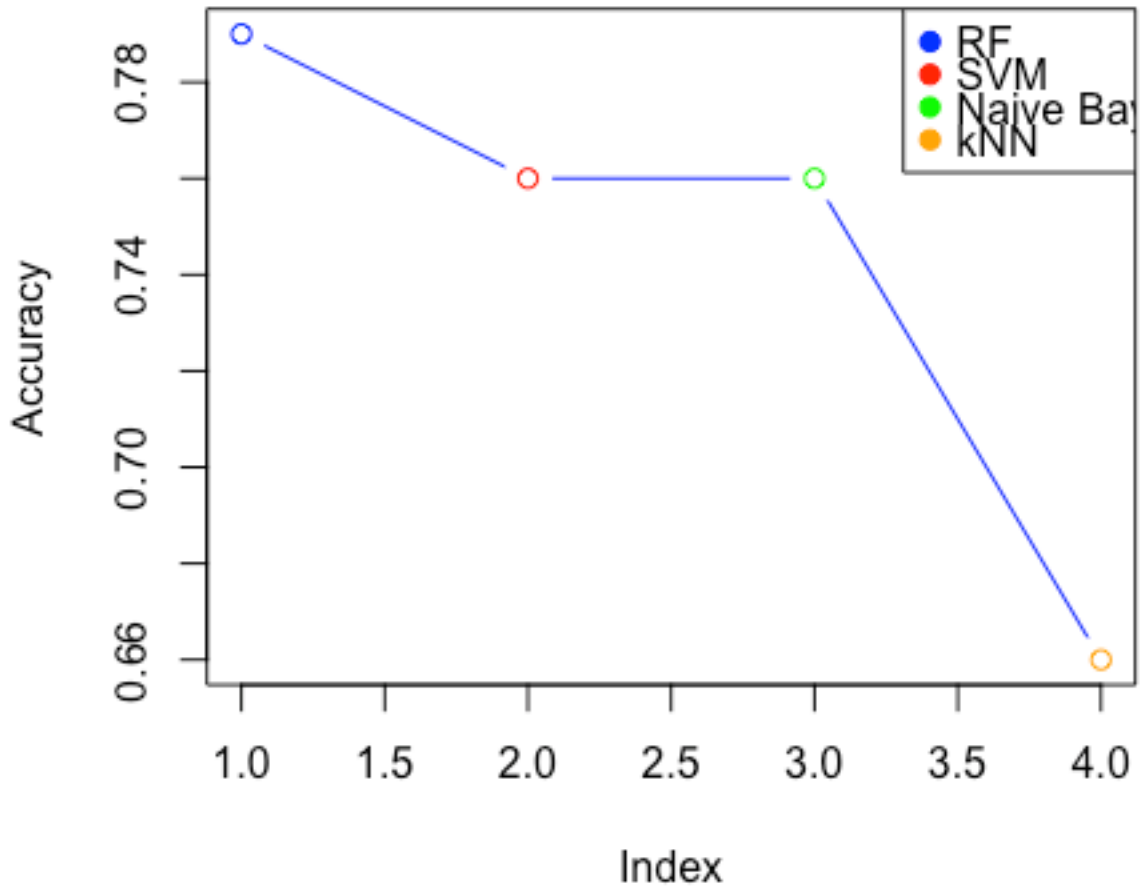
Now when the model was applied to test set, we obtained a score of 0.76077 which is lower than our other model but better than our training score of about 0.73.

Comparison:

On comparing the accuracy on the train data, the performance of various models were as follows:



The above graph represents the Accuracy of models on train set.



The best performance was for Random Forest, for both train and test accuracy. The y axis of the graph represents Accuracy on test data, the x-axis represents index, 1=RF, 2=SVM, 3=Naïve Bayes and 4=kNN