Project Description

Titanic was a passenger liner that sank on 15 April 1912 during its maiden voyage from the UK to New York City after colliding with an iceberg. Only 722 out of 2224 passengers survived in this disaster. Although there was some element of luck involved in surviving the sinking, some groups of people were more likely to survive than others, such as women, children, and the upper-class.

The goal of this project is to predict what sorts of people were likely to survive. For more information see Titanic – Machine Learning from Disaster presented by Kaggle.

Data

As with most Kaggle competitions, the Titanic data consists of a training set and a testing set, both of which are .csv files:

- The training set contains data for a subset of passengers including the outcomes or "ground truth" (i.e. the "Survived" response variable). We see that the training set has 891 observations (rows) and 12 variables. This set will be used to build the machine learning models.

- The testing set contains data for a subset of passengers without providing the ground truth, since the project's objective is to predict these outcomes. The testing set has 418 observations and 11 variables, since the "Survived" variable is missing. This set will be used to see how well the developed model performs on unseen data.

A description of the variables that are encountered in the Titanic dataset is given in Table 1. A few additional notes were made by Kaggle regarding specific details for some of the variables included in the Titanic data. It is first noted that passengers of 1st class (Pclass=1) belong to the upper socio-economic class, passengers of 2nd class (Pclass=2) belong to the middle socio-economic class and passengers of the 3rd class (Pclass=3) belong to the lower socio-economic class. As regards the Age variable, it can also appear as fractional if the passenger is less than 1 year old. Moreover, one will be able to tell if Age was estimated, that is if the Age value is in the form of xx.5, e.g. 40.5. There was also further explanation given by Kaggle for the family relation variables, i.e. SibSp and Parch. SibSp is an abbreviation for siblings and spouse. As siblings are considered the brothers, sisters, stepbrothers and stepsisters and as spouses are considered only husbands or wives. Thus, any mistresses and fiancés were not included in the SibSp variable. The Parch variable identifies both parents and children for each passenger aboard. Parents are considered to be either a mother or father and children can be a daughter, son, stepdaughter or stepson. Based on these definitions, we can conclude that some children, who travelled only with a nanny, will have the Parch variable equal to 0.

Variable Name	Variable Description	Possible Values	Categorical/Numerical
PassengerId	Observation Number	1, 2, …, 1309	Numerical
Survived	Survival	1 = Yes, 0 = No	Categorical
Pclass	Passenger Class	1 = 1st, 2 = 2nd, 3 = 3rd	Categorical
Name	Passenger Name	Braund, Mr. Owen Harris, etc.	Categorical
Sex	Sex of Passenger	male, female	Categorical
Age	Age of Passenger	0.17 – 80	Numerical
SibSp	No. of Siblings/Spouses Aboard	0 – 8	Numerical
Parch	No. of Parents/Children Aboard	0 – 9	Numerical
Ticket	Ticket Number	A/5 21171, 3101282, etc.	Categorical
Fare	Passenger Fare	0 – 512.3292	Numerical
Cabin	Passenger Cabin	C85, E46, etc.	Categorical
Embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton	Categorical

Table 1: Data description.

Descriptive Statistics and Exploratory Analysis

First, we are going to examine what number and what percentage of passengers survived from the training set.

```
  Amount Percentage    # (0=Perished,1=Survived)
0    549  0.6161616
1    342  0.3838384
```

We see that only 342 passengers survived, while 549 died, which means that the rate of survival is only 38%.

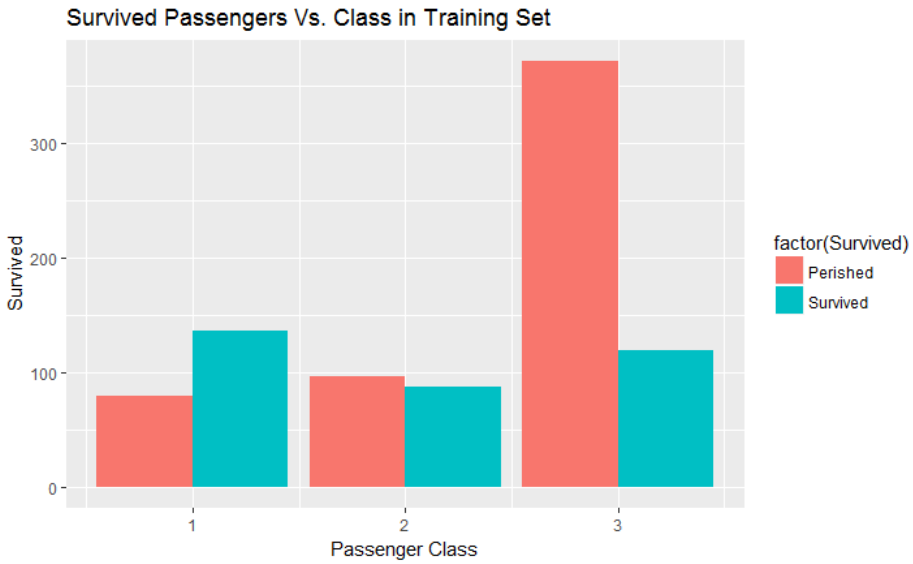
Next, we will try to find if the passenger class has an impact on survival. Let's first see the distribution of people across classes.

```
  1   2   3
216 184 491
```

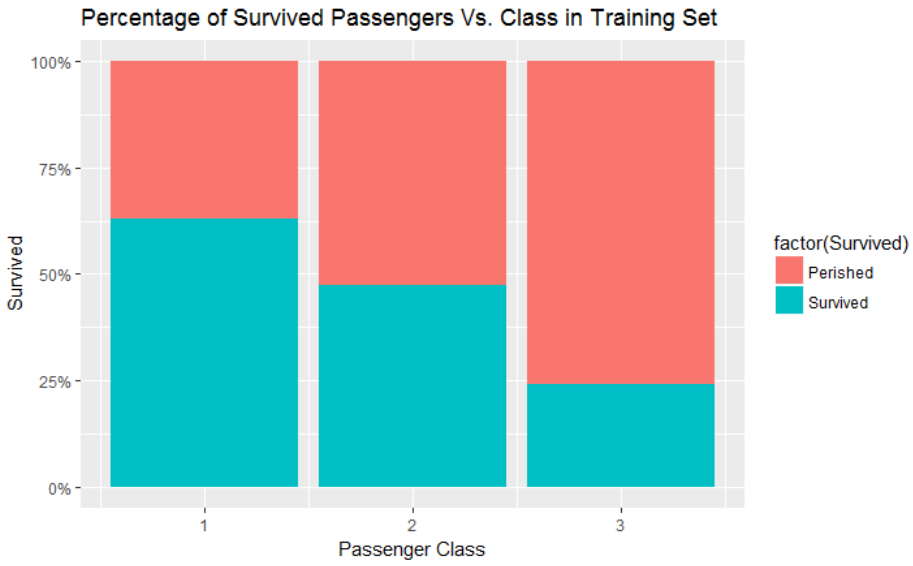
We notice that the majority of passengers were in the 3rd class. Let's now check if the passenger class has an impact on survival.

```
          0         1      # (0=Perished,1=Survived)
1 0.3703704 0.6296296
2 0.5271739 0.4728261
3 0.7576375 0.2423625
```

We can also use some visualizations in order to better understand the relationships between variables. The next figures show the number and percentage of survived passengers in relation to their class. We observe that ~ 75% of people in the 3rd class perished, compared to ~ 52% of people in the 2nd class and only ~ 37% of people in the 1st class. Thus, it is obvious that people in the upper classes have a significantly higher rate of survival.



(a)



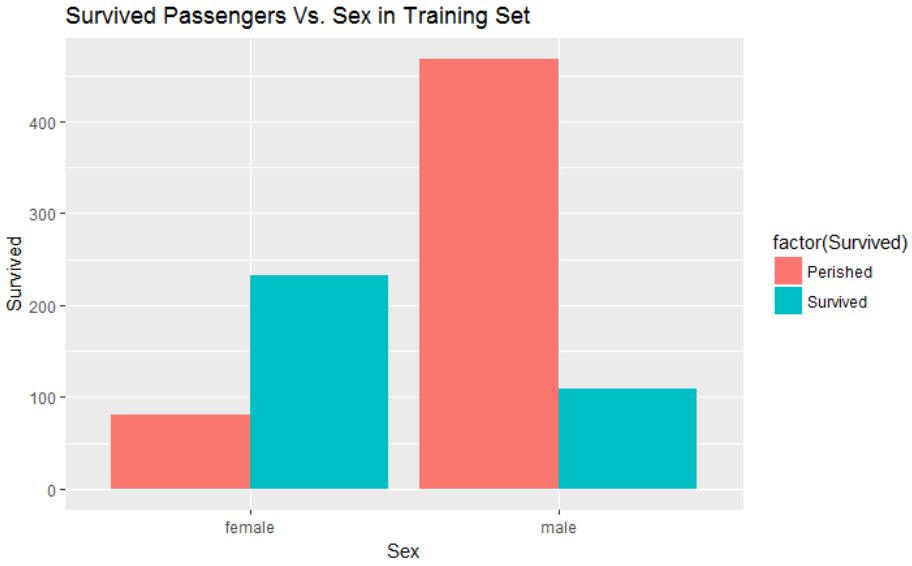
(b)

Figure 1: Figure (a) shows the number and Figure (b) the percentage of survived passengers in relation to their class.

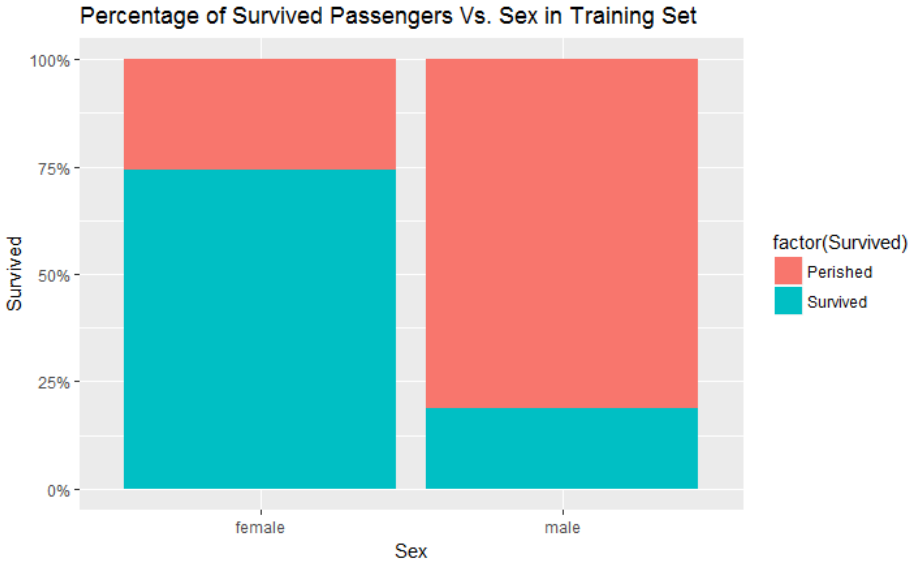
In Titanic the captain gave order for women and children to be saved first. So, we will investigate if sex and age have indeed an impact on the rate of survival. The next table shows what percentage of male and female passengers survived.

```
                0         1       # (0=Perished,1=Survived)
female 0.2579618 0.7420382
male   0.8110919 0.1889081
```

Figure 2 shows the number and percentage of survived passengers in relation to their sex. We notice that the majority of passengers were male. Furthermore, it is evident that females have a significantly better survival rate (~ 74%) compared to male passengers (~ 18%).



(a)



(b)

Figure 2: Figure (a) shows the number and Figure (b) the percentage of survived passengers in relation to their sex.

By examining Age variable we observe that it has 177 missing values. But since Age is a very important variable for prediction, we have to continue the analysis in order to get a better insight from other variables that will help us to tackle the missing values from Age.

Data Processing

First, we have to find how many missing values we have and in which variables and replace them with sensible values. We see that we have missing values in Age, Cabin and Embarked variables in the training set and Age, Fare and Cabin variables in the testing set. To tackle this problem, we are going to predict the missing values with the full data set, which means that we need to combine the training and testing sets together. The missing values in the full set can be seen in the next table.

```
     [,1]       [,2]
[1,] "Age"      "263"
[2,] "Fare"     "1"
[3,] "Cabin"    "1014"
[4,] "Embarked" "2"
```

Variable "Cabin"

We see that Cabin is missing most of its values. We will create a new variable “Deck” with values A – F by separating and pulling off the deck letter contained in the Cabin and replace the missing values with U (for Unknown). However, given the sparseness of this column, we will not further process it.

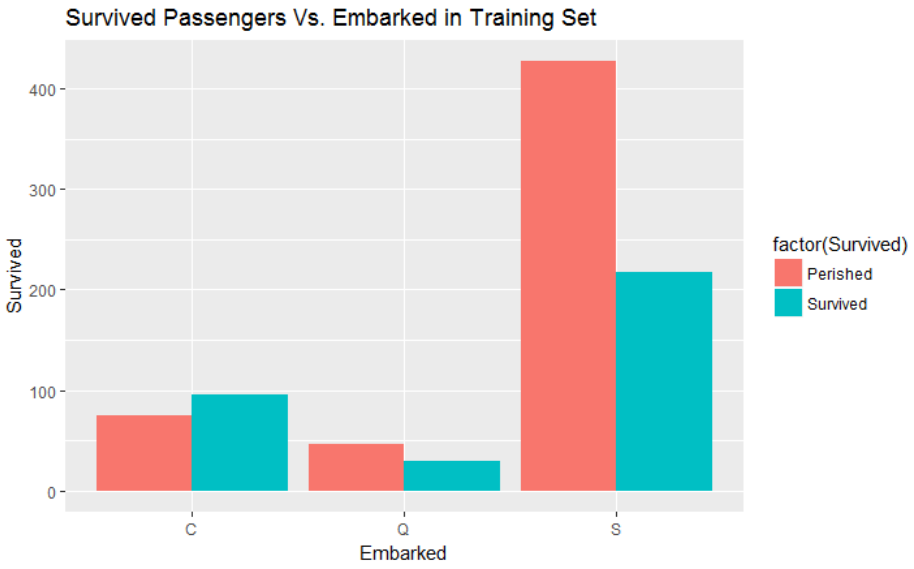
Variable "Embarked"

We know that two passengers have missing Embarked values. We will infer these values based on present data that seem relevant: passenger class (Pclass) and fare (Fare). We see that the passengers who have missing Embarked values paid $80 and their class is 1. The next table shows the median fare for 1st class passengers in relation to port of embarkation.

```
  Embarked    Fare
1        C 76.7292
2        Q 90.0000
3        S 52.0000
```

Since, passengers who have missing Embarked values paid $80, we can replace their NA values with 'C' (Charbourg).

The next figures show the number and percentage of survived passengers in relation to their port of embarkation. It is obvious that the majority of passengers embarked from Southampton (S). Also, we observe that people who embarked from Charbourg (C) have the highest rate of survival, while people that embarked from Queenstown (Q) and Southampton (S) have considerably lower survival rates.



(a)

(b)

Figure 3: Figure (a) shows the number and Figure (b) the percentage of survived passengers in relation to their port of embarkation.

Variable "Fare"

The next table shows a summary of the Fare variable, which includes the mean, the median, the min and max values, the 1st and 3rd quartiles and the number of missing values.

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  0.000   7.896  14.454  33.295  31.275 512.329       1
```

We see that there is zero fare, which corresponds to these passengers:

```
     Age Fare
180   36    0
264   40    0
272   25    0
278   NA    0
303   19    0
414   NA    0
467   NA    0
482   NA    0
598   49    0
634   NA    0
675   NA    0
733   NA    0
807   39    0
816   NA    0
823   38    0
1158  NA    0
1264  49    0
```

There might be some error, since zero fares do not correspond to infants, that possibly were allowed to travel free of cost. We will replace the zero fares with the median values of the fares corresponding to each passenger class (Pclass) and embarkment (Embarked). The next table shows the median fare for passengers in relation to class and port of embarkation.

```
  Pclass Embarked    Fare
1      1        C 78.2667
2      2        C 15.3146
3      3        C  7.8958
4      1        Q 90.0000
5      2        Q 12.3500
6      3        Q  7.7500
7      1        S 52.0000
8      2        S 15.3750
9      3        S  8.0500
```

After substitution is done, the summary of the Fare variable becomes:

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  3.171   7.925  14.500  33.669  31.387 512.329       1
```

Fare still has one missing value. We see that the passenger with the missing fare is a 3rd class passenger, who embarked from Southampton ('S'). We will replace the NA Fare value with the median fare value for 3rd class passengers who departed from Southampton, which is $8.05.

Figure 4 shows the number of survived passengers in relation to their fare.

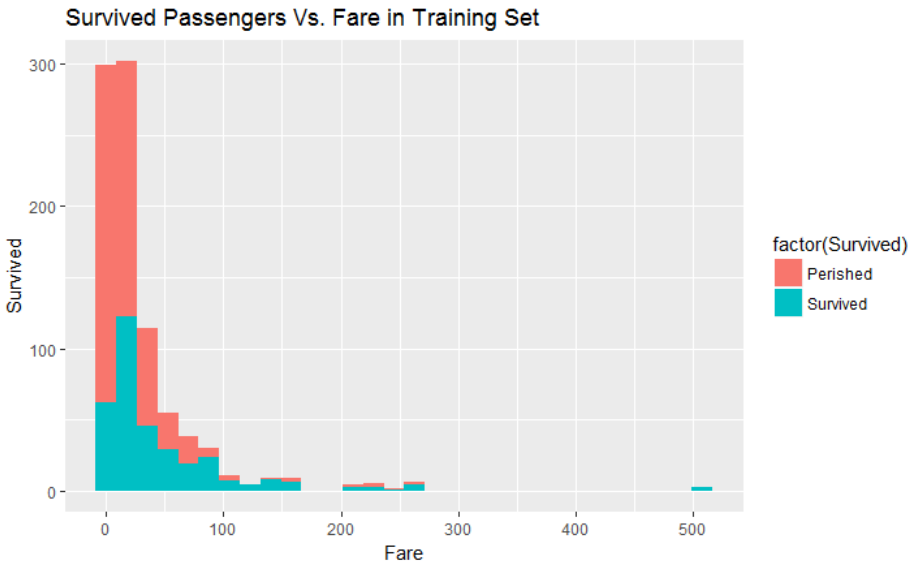
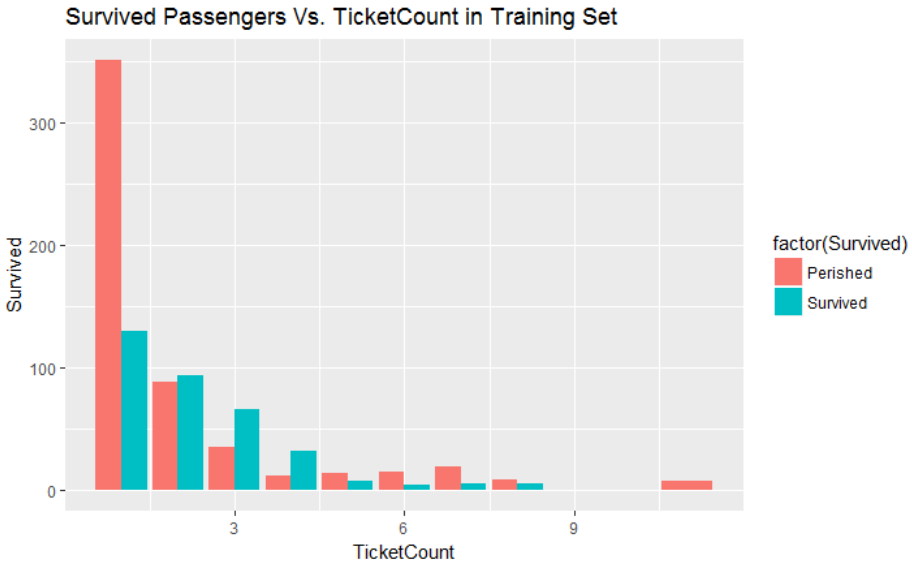


Figure 4: Number of survived passengers in relation to their fare.

Variable "Ticket"

By exploring the Ticket variable, we see that some passengers have the same ticket number. So, we will create a new variable "TicketCount" which counts the number of passengers that have the same ticket number. Figure 5 shows the number and percentage of survived passengers in relation to their ticket number. We see that there's a survival penalty to those who have a unique ticket number and those with same ticket numbers above 4, but an advantage for those with small same ticket numbers.



(a)

(b)

Figure 5: Figure (a) shows the number and Figure (b) the percentage of survived passengers who have the same ticket number.

Variable "Title"

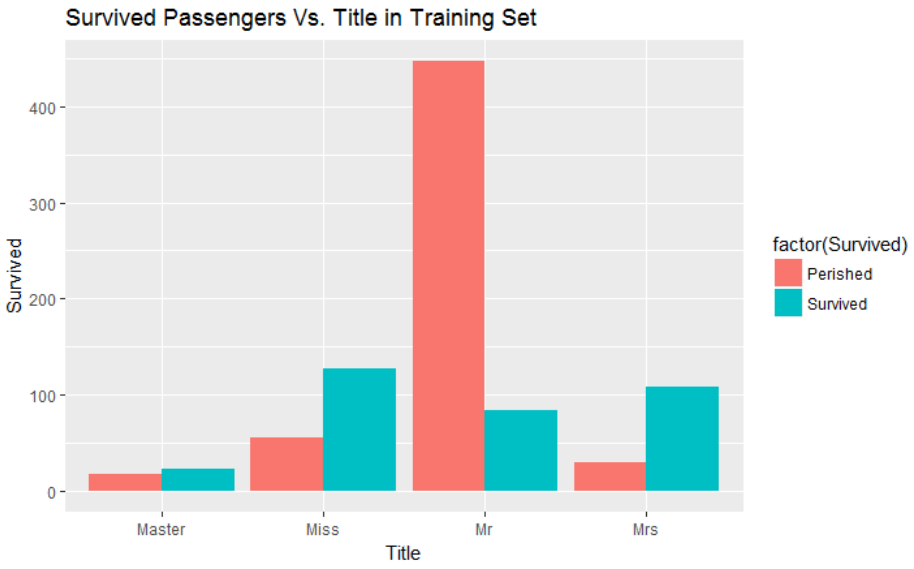
By exploring the Name variable, we see that it is a combination of first name, last name and title (e.g., Mr, Mrs etc.). We are going to use this variable to extract another feature called “Title”, which might be helpful to predict the missing Age values. The next table shows a summary of the Title variable.

```
  Capt         Col         Don        Dona          Dr    Jonkheer        Lady
     1           4           1           1           8           1           1
 Major      Master        Miss        Mlle         Mme          Mr         Mrs
     2          61         260           2           1         757         197
    Ms         Rev         Sir the Countess
     2           8           1           1
```

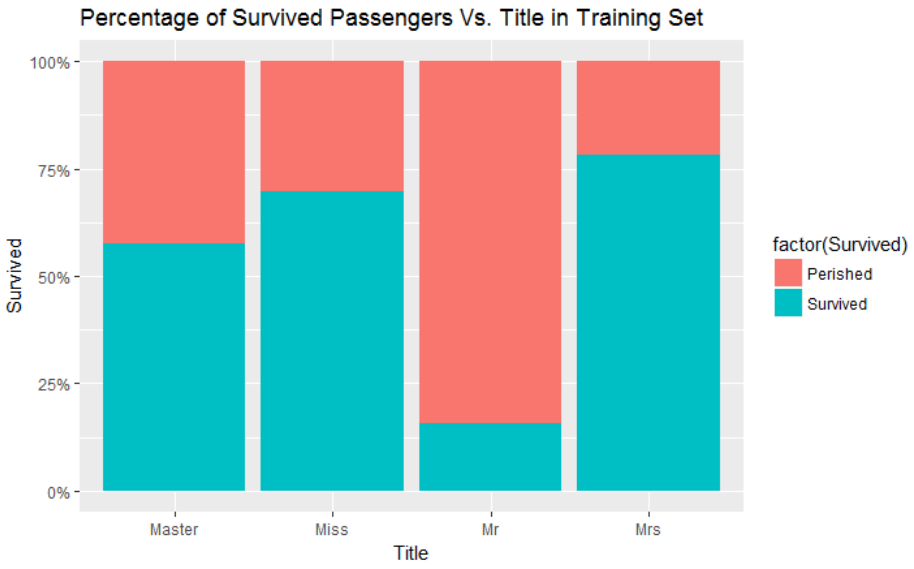
We see that there are many different title groups. We will merge them to the most common 4 groups: Mr and Master for male and Miss and Mrs for female. The next table shows the new Title counts by sex.

```
        Master Miss  Mr Mrs
female      0  260   0 206
male       61    0 775   7
```

The next figures show the number and percentage of survived passengers in relation to their title. From Figure (a) it is evident that the majority of passengers have the title ‘Mr’. Also, we observe that passengers who have the title ‘Mr’ have the lowest survival rate (only ~ 0.15), while passengers that have the title ‘Mrs’ have the highest survival rate (~0.78).



(a)



(b)

Figure 6: Figure (a) shows the number and Figure (b) the percentage of survived passengers in relation to their title.

Finally, we can extract surname from passenger name and create a new variable “Surname” to represent families. We discover that there are 875 unique surnames in a total of 1309 passengers.

Variable "FamilySize"

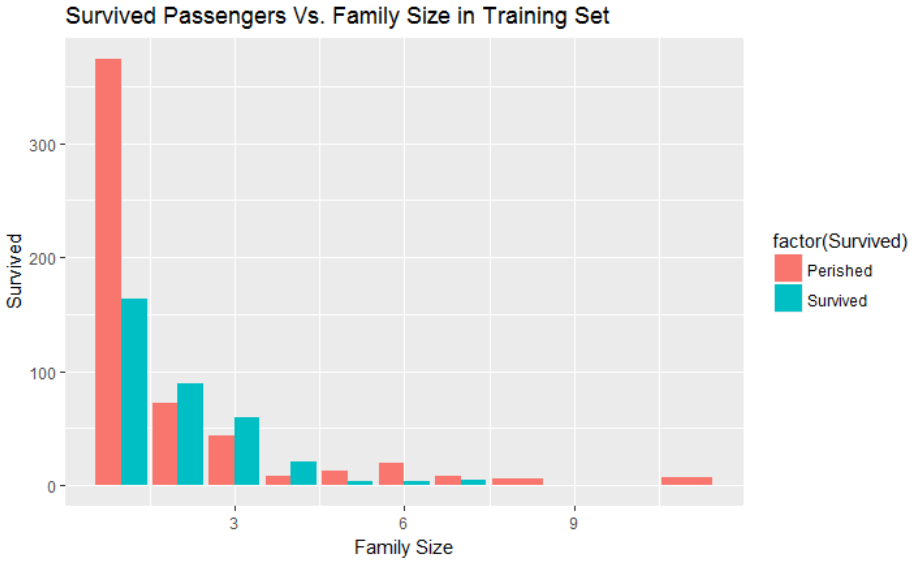
Based on number of siblings/spouse (SibSp) and number of children/parents (Parch) we create the new variable “FamilySize”. In order to better understand how FamilySize relates to survival we will compute the number of passengers survived in relation to their family size.

```
     0   1      # (0=Perished,1=Survived)
1  374 163
2   72  89
3   43  59
4    8  21
5   12   3
6   19   3
7    8   4
8    6   0
11   7   0
```

The next table shows the corresponding percentage of passengers survived in relation to their family size.

```
           0         1      # (0=Perished,1=Survived)
1  0.6964618 0.3035382
2  0.4472050 0.5527950
3  0.4215686 0.5784314
4  0.2758621 0.7241379
5  0.8000000 0.2000000
6  0.8636364 0.1363636
7  0.6666667 0.3333333
8  1.0000000 0.0000000
11 1.0000000 0.0000000
```

Figure 7 shows the number and percentage of survived passengers in relation to their family size. We notice that there's a survival penalty to singletons and those with family sizes above 4, but an advantage for those with small families.

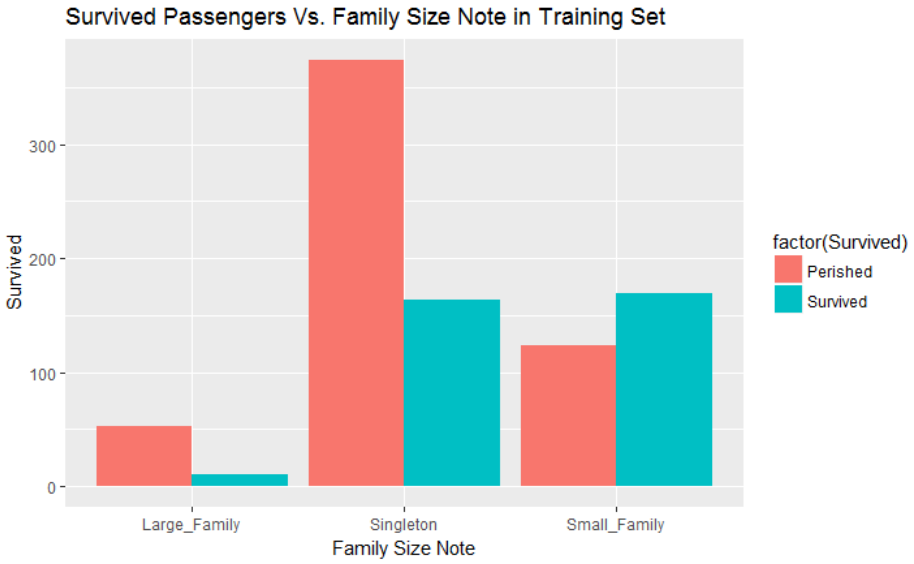


(a)

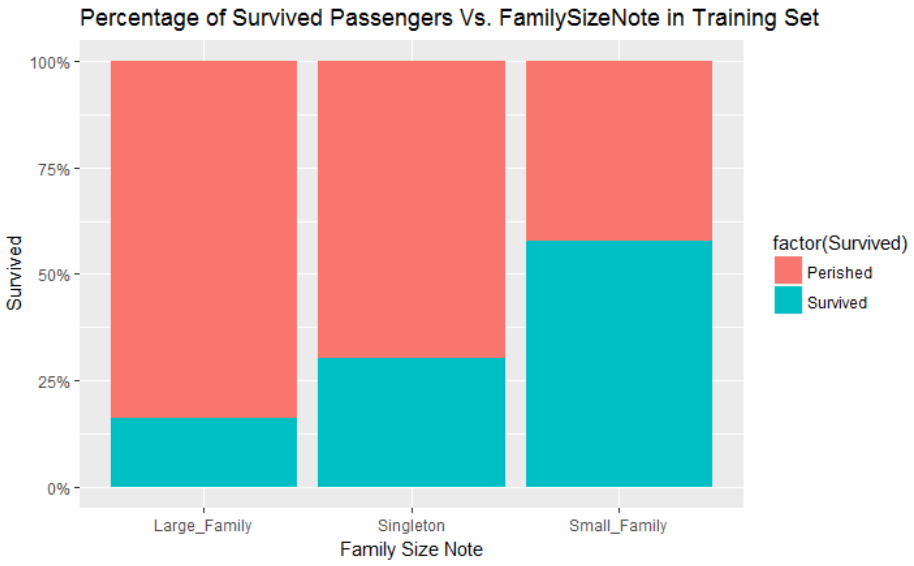
(b)

Figure 7: Figure (a) shows the number and Figure (b) the percentage of survived passengers in relation to their family size.

We can also create a discretized family size variable with 3 levels (Singleton, Small Family, Large Family), since there are comparatively fewer large families. Figure 8 shows the number and percentage of survived passengers in relation to their family size (using the discretized variable). From these figures, it is more obvious than before, that there's a survival penalty to singletons and those with large families, but an advantage for those with small families.



(a)



(b)

Figure 8: Figure (a) shows the number and Figure (b) the percentage of survived passengers in relation to their family size (using a discretized variable).

Finally, we will create a "FamilyID" variable using the surname and the number of family members, e.g. Palsson_5. We see that there are 928 unique FamilyIDs and that some passengers have the same surname but are not in the same family, e.g. two people with surname Andersson that travelled alone and are not in the same family, have the same FamilyID, i.e. : Andersson_1. So let's change the FamilyIDs that have family size of two or less and call it a "Small" family. After this substitution, the FamilyID variable has only 78 unique levels, compared to the original 928.

Variable "Age"

Age has many missing values. However, replacing its missing values with the median age might not be the best idea, since age may differ by groups and categories of the passengers. To see an example, we will group the dataset by Sex, Title and Class, and for each subset we will compute the median age.

```
      Sex  Title Pclass  Age
2  female   Miss      1 30.0
7  female   Miss      2 20.0
12 female   Miss      3 18.0
4  female    Mrs      1 45.0
9  female    Mrs      2 30.0
14 female    Mrs      3 31.0
1    male Master      1  6.0
6    male Master      2  2.0
11   male Master      3  6.0
3    male     Mr      1 42.0
8    male     Mr      2 30.0
13   male     Mr      3 26.0
5    male    Mrs      1 47.0
10   male    Mrs      2 38.5
```

We see that the median age depends a lot on the Sex, Title and Pclass values.

In order to predict the missing Age values, we will create a model predicting ages based on other variables. For this purpose, we will use rpart (recursive partitioning for regression). Once we get the predicted values, we can compare the original distribution of passenger ages with the predicted distribution of passenger ages, to ensure that they look similar. The distributions can be seen in Figure 9.

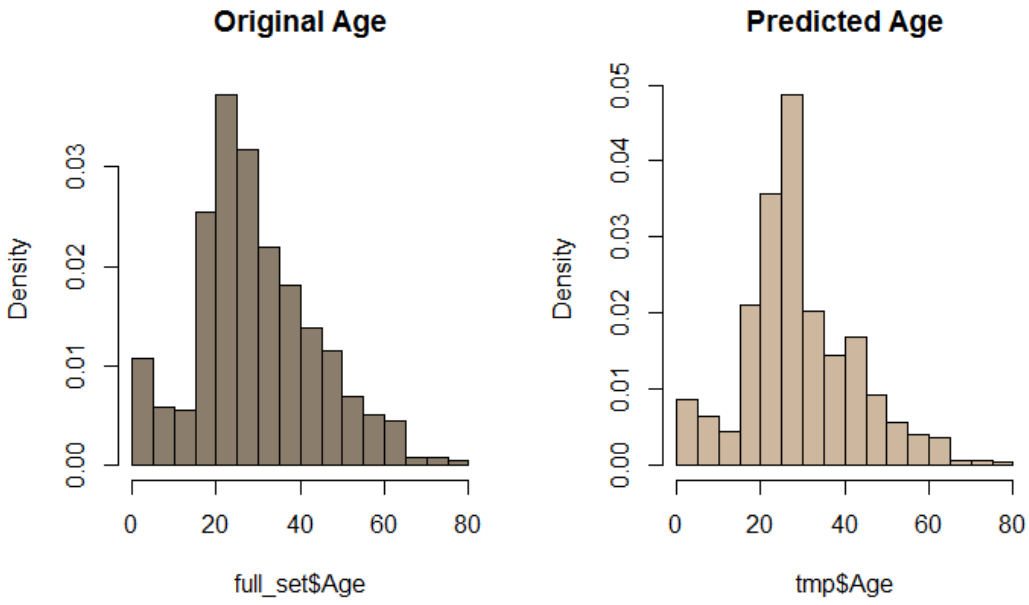


Figure 9: Comparison of the original distribution of passenger ages with the predicted distribution of passenger ages.

As we can see, they look quite similar, so we can replace the missing Age values with the predicted ones. The next figure shows the number of survived passengers in relation to their age.

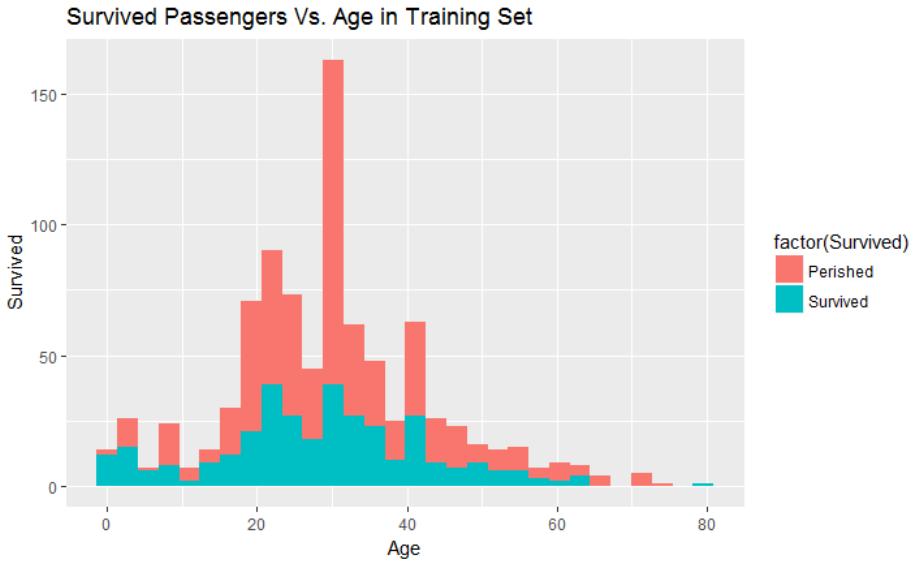


Figure 10: Number of survived passengers in relation to their age.

We can also include the variable “Sex” in the previous figure, since we know that it is a significant predictor. Figure 11 shows the number of survived passengers in relation to their age and sex. It is evident that females have a significantly better survival rate irrespective of their age.

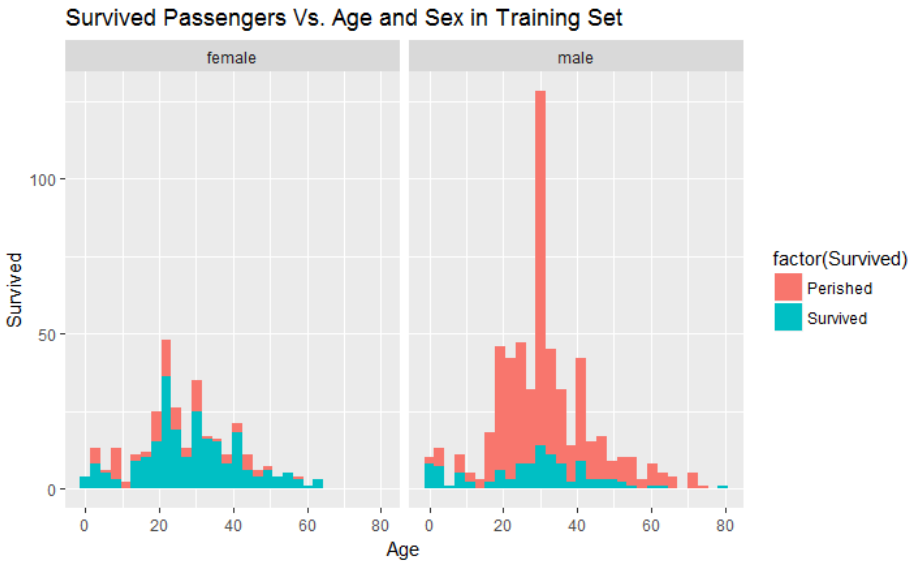


Figure 11: Number of survived passengers in relation to their age and sex.

Note that Age is a continuous variable and thus drawing proportion tables is almost useless, as there may only be one or two passengers for each age. Therefore, now that we have Age values for all passengers, we can create a few more age-dependent variables.

Variable “Child”

We can create a variable that indicates if a passenger is child or adult. Anyone who is less than 18 years is considered to be child.

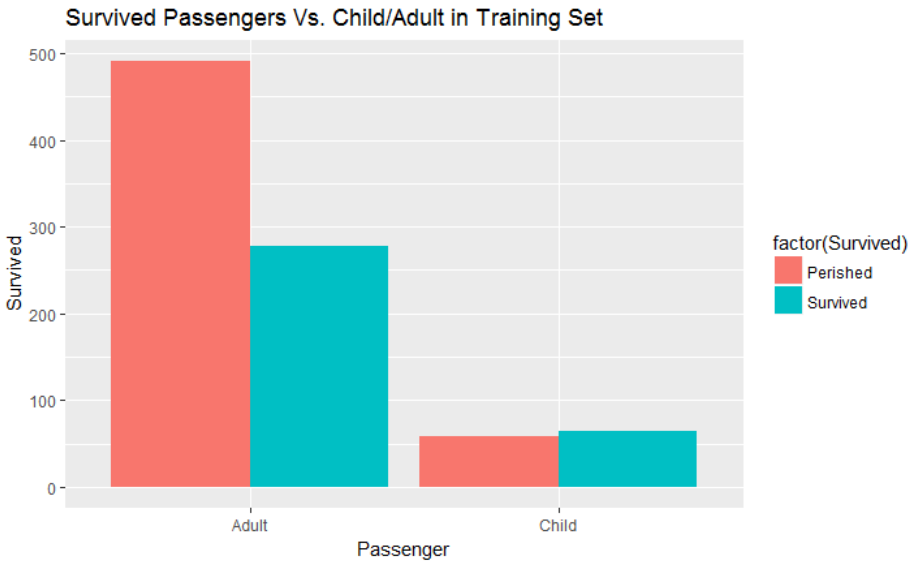
Now, that we have created this variable, we can examine if children are mostly likely to be rescued first.

```
          0   1    # (0=Perished,1=Survived)
Adult 491 278
Child  58  64
```

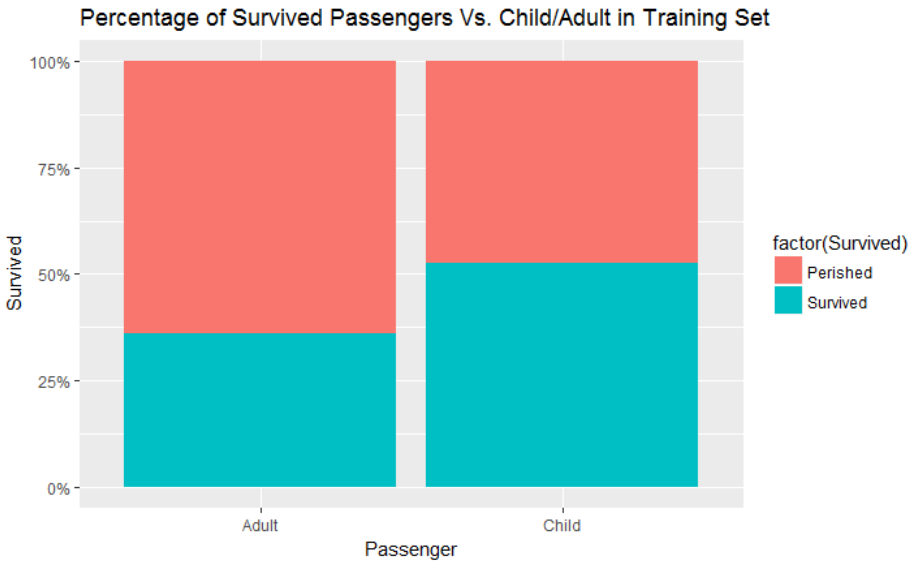
We can also compute the corresponding probabilities.

```
  Child  Survived
1 Adult 0.3615085
2 Child 0.5245902
```

Figure 12 shows the number and percentage of survived underage and adult passengers. As we can see, there is only a ~50% chance that you will survive if you are a child.



(a)



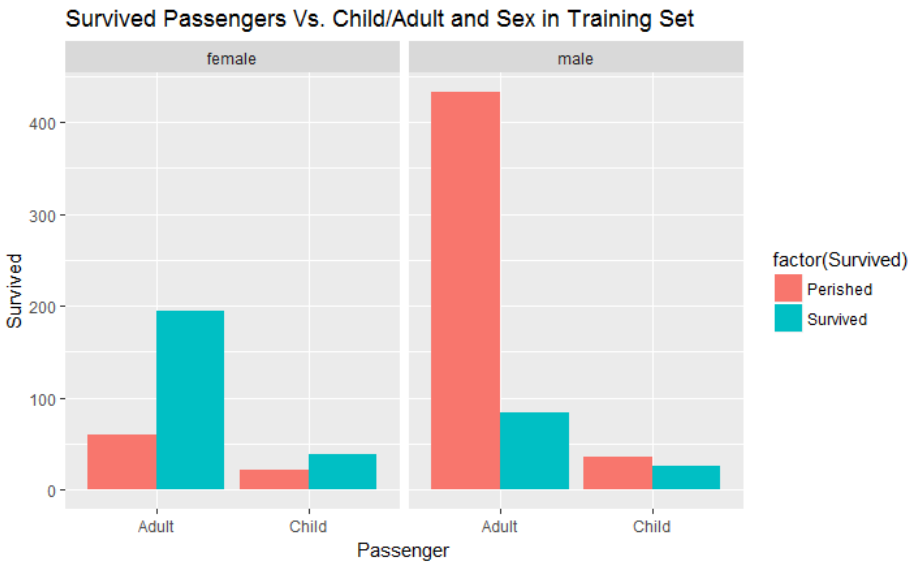
(b)

Figure 12: Figure (a) shows the number and Figure (b) the percentage of survived underage and adult passengers.

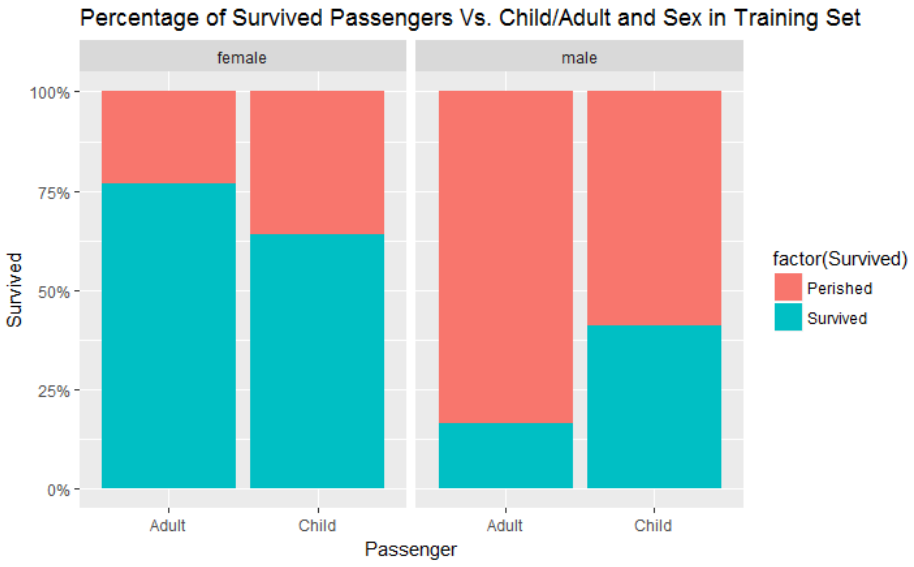
It would be interesting to investigate if female children had a higher chance to survive compared to male children. The percentages of survived children based on their sex are given in the next table.

```
  Child    Sex  Survived
1 Adult female 0.7667984
2 Child female 0.6393443
3 Adult   male 0.1627907
4 Child   male 0.4098361
```

Figure 13 shows the number and the percentage of survived female and male underage and adult passengers. It is obvious that female children are more likely to survive (0.6393443) compared to male children (0.4098361).



(a)



(b)

Figure 13: Figure (a) shows the number and Figure (b) the percentage of survived female and male underage and adult passengers.

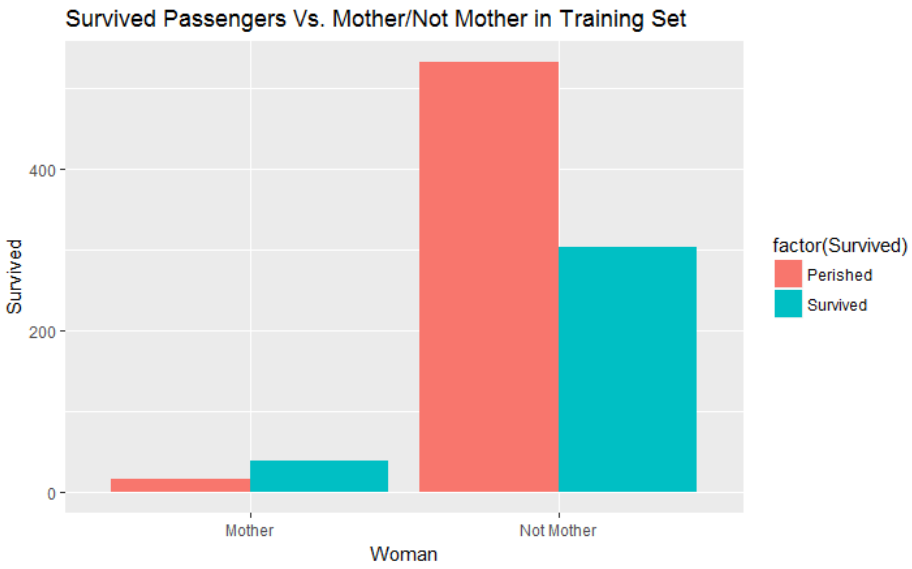
Variable “Mother”

Finally, we can create a "Mother" variable to indicate whether the passenger is mother or not. As mother is considered a passenger who is female, over 18 years old, has 1 child or more and has the Title "Mrs".

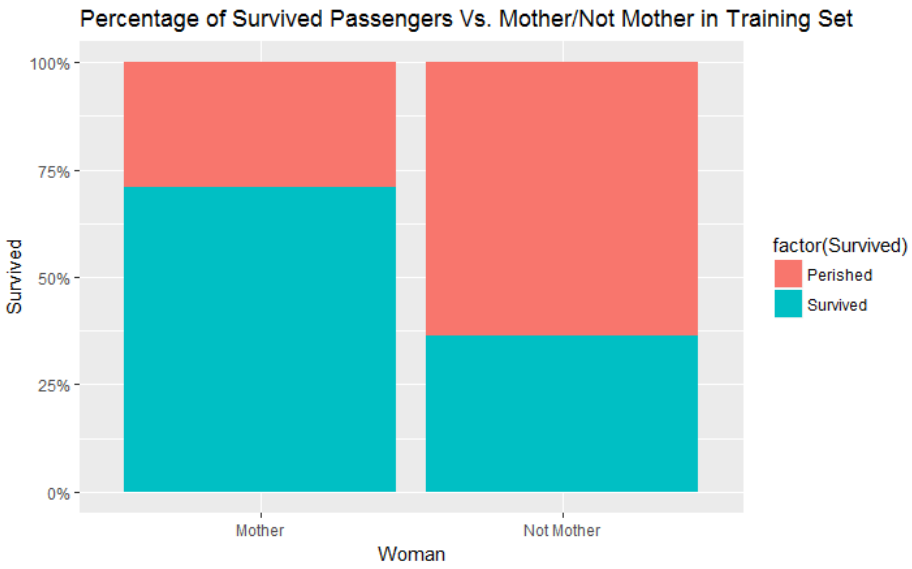
The next table shows the percentage of survived mothers compared to those that were not mothers.

```
      Mother  Survived
1     Mother 0.7090909
2 Not Mother 0.3624402
```

Figure 14 shows the number and percentage of survived mothers compared to those that were not mothers. From Figure (a) we notice that the majority of women do not have children. Also, it is obvious that “Mothers” are more likely to survive (0.7090909) compared to “Not Mothers” (0.3624402).



(a)



(b)

Figure 14: Figure (a) shows the number and Figure (b) the percentage of survived mothers compared to those that were not mothers.

Prediction

In the begging of data processing we merged training and testing sets. Therefore, our first step now is to split the data back into the training and testing sets.

Model Building

Now, we will build and fit some models to the training set and we will compute their accuracy on the testing set. For this purpose, we will build seven different models: Decision Tree (CART), Random Forest (RF), Forest of conditional inference trees, Generalized Linear Model (GLM), Gradient Boosting Machine (GBM), Neural Network (NN) and Support Vector Machine (SVM) with Radial Basis Function Kernel. We reset the random number seed before each run to ensure that each algorithm will be evaluated using the same data partitions. This means that the results will be directly comparable.

Decision Trees

Decision Trees classify observations by sorting them down the tree from the root node to some leaf node which provides the classification of the observation. Each node in the tree specifies a test on a particular attribute (explanatory variable) of the observation, and each branch descending from that node corresponds to one of the possible values for that test [1]. Notice that decision trees are prone to overfitting which means that we should be very careful with how deep we grow them.

In this project, we will use the rpart (which stands for recursive partitioning) algorithm to build the tree. First, we will build a decision tree to predict Survived using only the variables Age and Sex. The next figure illustrates this decision tree.

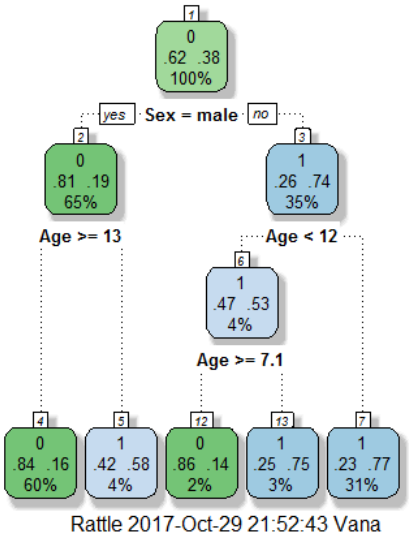


Figure 15: Visualization of the decision tree.

From the top we can see that the node is voting 0, so at this level everyone would die. Below we see that 62% of passengers die, while 38% survive (the most will die here that's why the node is voting that everyone dies). If we go down to the male/female 81%/26% will die and 19%/74% will survive as the proportions exactly match those we found earlier in the data analysis. Then decisions have been made based on the Age variable. 84% of male passengers older than 13 years old will die, while only 16% of them will survive and 42% of male passengers younger than 13 years old will die while 58% of them will survive. 47% of female passengers younger than 12 years old will die, while 53% of them will survive and more specifically 86% of female passengers older than 7.1 years old will die, while only 14% of them will survive and 25% of female passengers younger than 7.1 years old will die, while 75% of them will survive. Finally, 23% of female passengers older than 12 years old will die, while 77% of them will survive.

We can now evaluate this model by computing its accuracy on the training set and the testing set (by submitting a .csv file to Kaggle). The accuracy on the training set is 0.80471 and on the testing set 0.76555.

Using different variables as predictors we can build new decision trees. Table 2 provides a summary of the model evaluation results. The best prediction for survival with decision trees was achieved using the formula "Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount + FamilyID". The accuracy of this model on the training and testing set is 0.84960 and 0.80382, respectively. Figure 16 visualizes the decision tree that was built with these predictors. We observe that our new variables are governing the tree. It is well known that decision trees are biased to favour factors with many levels. This is evident here since the 78-level FamilyID factor is so prominent here. In most cases though, the sex or title variables govern the first decision due to the greedy nature of decision trees.

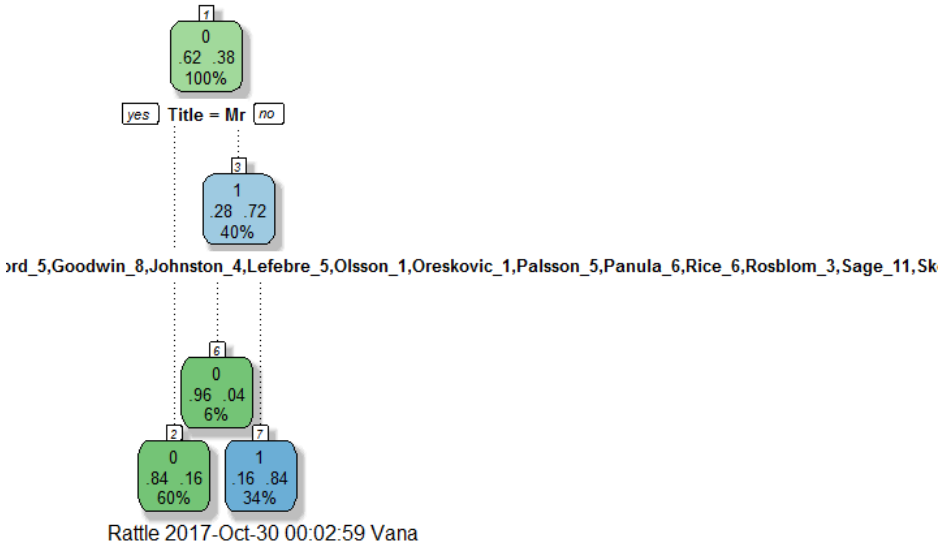


Figure 16: Visualization of the decision tree that achieved the highest accuracy on the testing set.

Formula	Accuracy (Training Set)	Accuracy (Testing Set) – Kaggle Score
1. Survived ~ Age + Sex	0.80471	0.76555
2. Survived ~ Age + Sex + Pclass	0.82828	0.74641
3. Survived ~ Age + Sex + Fare	0.79910	0.77033
4. Survived ~ Age + Sex + Pclass + Fare	0.84848	0.79425
5. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch	0.84848	0.79904
6. Survived ~ Age + Sex + Pclass + SibSp + Parch + Fare + Embarked	0.83950	0.78947
7. Survived ~ Age + Sex + Pclass + FamilySize	0.83950	0.77990
8. Survived ~ Age + Sex + FamilySize	0.82716	0.78947
9. Survived ~ Age + Sex + FamilySizeNote	0.82828	0.78947
10. Survived ~ Age + Sex + Pclass + FamilySizeNote	0.84062	0.77990
11. Survived ~ Age + Sex + Pclass + FamilySize + Embarked	0.83950	0.77990
12. Survived ~ Age + Sex + Pclass + Fare + Title	0.83726	0.77990
13. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Title	0.83726	0.77990
14. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother	0.84848	0.79904
15. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote	0.83277	0.78947
16. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Embarked + Title + FamilySizeNote + TicketCount	0.83389	0.79425
17. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother +Embarked + Title + FamilySizeNote + TicketCount	0.83389	0.79425
18. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote + TicketCount + Deck	0.83389	0.79425
19. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount	0.83389	0.79425
20. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount + FamilyID	0.84960	0.80382

Table 2: Performance summary of decision tree models.

Random Forest

Random Forest (RF) is an algorithm that fits many classification (or regression) tree models to random subsets of the input data and averages the predictions from such simple trees for prediction [2]. Random forests correct for decision trees' habit of overfitting to their training set [3].

Note that random forests in R can only digest factors with up to 32 levels. This means that our FamilyID variable, which has 78 levels, cannot be used as predictor. There are two possible solutions to this problem: either change these levels to their underlying integers (using the unclass() function) and having the tree treat them as continuous variables, or manually reduce the number of levels to keep it under the threshold. In this project we used the second approach, and the new FamilyID variable has only 29 levels.

It should also be mentioned that in this project we can grow a large number of trees and not worry about their complexity, because the dataset is small and the model will run pretty fast.

First, we will build a random forest model to predict Survived using only the variables Age and Sex. It is well-known that random forests do not waste the out-of-bag (OOB) observations, since they use them to examine how well each tree performs on unseen data. The mean decrease in Gini measures how pure the nodes are at the end of the tree. It actually tests how worse the model would perform if each variable was taken out, and a high score means that the variable was important. The next figure shows the relative variable importance by plotting the mean decrease in Gini calculated across all trees. In this simple first model, we see that the most important predictor variable is sex.

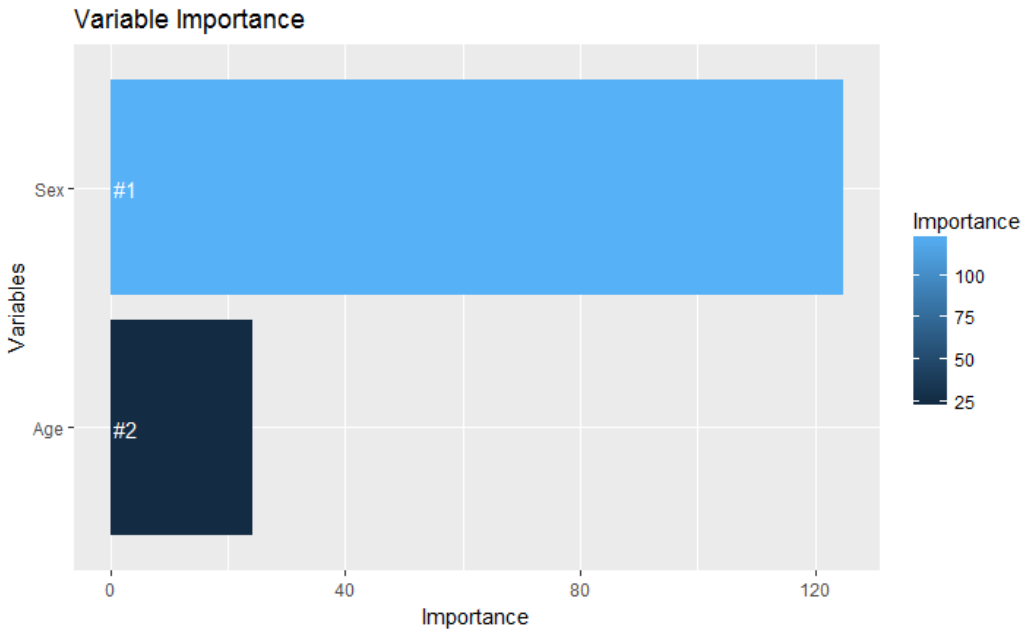


Figure 17: Variable importance for the developed random forest model.

We can now evaluate this model by computing its accuracy on the training set and the testing set (by submitting a .csv file to Kaggle). The accuracy on the training set is 0.79797 and on the testing set 0.76555.

Using different variables as predictors we can build new random forests models. Table 3 provides a summary of the model evaluation results. The last model of Table 3 was tuned to find the optimal parameter values, i.e., the optimal value of mtry (number of variables used at each split of the tree) and the optimal value of ntree (number of trees). The best prediction for survival with random forests was achieved using the formula "Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote + TicketCount". The accuracy of this model on the training and testing set is 0.91245 and 0.80382, respectively. Figure 18 illustrates the relative variable importance of these predictors. Unsurprisingly, our Title variable has the highest relative importance out of all predictor variables.

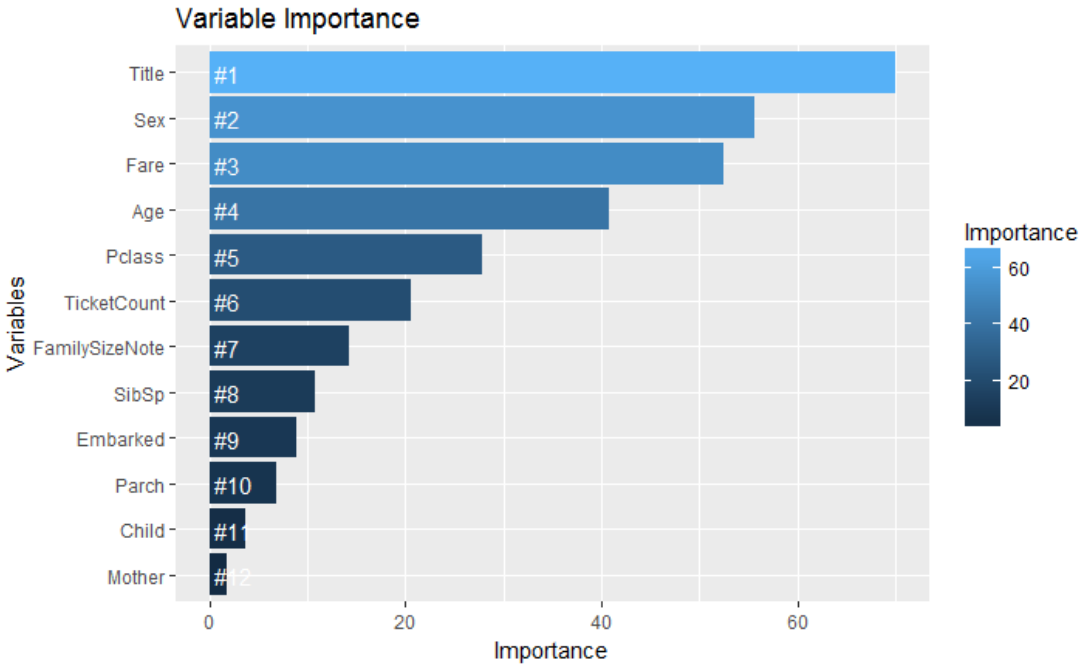


Figure 18: Variable importance for the random forest model that achieved the highest accuracy on the testing set.

Formula	Accuracy (Training Set)	Accuracy (Testing Set) – Kaggle Score
1. Survived ~ Age + Sex	0.79797	0.76555
2. Survived ~ Age + Sex + Pclass	0.83052	0.75119
3. Survived ~ Age + Sex + Fare	0.80022	0.76076
4. Survived ~ Age + Sex + Pclass + Fare	0.90460	0.77033
5. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch	0.91021	0.77990
6. Survived ~ Age + Sex + Pclass + SibSp + Parch + Fare + Embarked	0.90796	0.77511
7. Survived ~ Age + Sex + Pclass + FamilySize	0.85297	0.77033
8. Survived ~ Age + Sex + FamilySize	0.82603	0.77033
9. Survived ~ Age + Sex + FamilySizeNote	0.83164	0.78947
10. Survived ~ Age + Sex + Pclass + FamilySizeNote	0.84399	0.77990
11. Survived ~ Age + Sex + Pclass + FamilySize + Embarked	0.87542	0.77990
12. Survived ~ Age + Sex + Pclass + Fare + Title	0.88327	0.77511
13. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Title	0.88664	0.78947
14. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother	0.87205	0.77990
15. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote	0.90796	0.79904
16. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Embarked + Title + FamilySizeNote + TicketCount	0.92704	0.79425
17. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother +Embarked + Title + FamilySizeNote + TicketCount	0.91245	0.80382
18. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote + TicketCount + Deck	0.92143	0.78947
19. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount	0.89674	0.78947
20. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID2 + SibSp + Parch	0.92817	0.77511
21. Survived ~ Age + Sex + Pclass + Fare + Title + FamilySizeNote + TicketCount	0.89113	0.79904

Table 3: Performance summary of random forest models.

Forest of conditional inference trees

Conditional inference trees are a popular tree-based classification method. Similar to traditional decision trees, conditional inference trees also recursively partition the data by performing a univariate split on the dependent variable. The difference between conditional inference trees and traditional decision trees is that conditional inference trees adapt the significance test procedures to select variables rather than selecting variables by maximizing information measures [4]. Since conditional inference trees are able to handle factors with more levels than random forests, we can again use the 78-level variable FamilyID.

First, we will build a forest of conditional inference trees to predict Survived using only the variables Age and Sex. We can now evaluate this model by computing its accuracy on the training set and the testing set (by submitting a .csv file to Kaggle). The accuracy on the training set is 0.80695 and on the testing set 0.74641.

Using different variables as predictors we can build new forests of conditional inference trees. Table 4 provides a summary of the model evaluation results. The best prediction for survival with forests of conditional inference trees was achieved using the formula “Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch”. The accuracy of this model on the training and testing set is 0.85409 and 0.82296, respectively.

Formula	Accuracy (Training Set)	Accuracy (Testing Set) – Kaggle Score
1. Survived ~ Age + Sex	0.80695	0.74641
2. Survived ~ Age + Sex + Pclass	0.82603	0.72248
3. Survived ~ Age + Sex + Fare	0.84511	0.75598
4. Survived ~ Age + Sex + Pclass + Fare	0.83277	0.76076
5. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch	0.84736	0.78468
6. Survived ~ Age + Sex + Pclass + SibSp + Parch + Fare + Embarked	0.82940	0.78947
7. Survived ~ Age + Sex + Pclass + FamilySize	0.82828	0.75598
8. Survived ~ Age + Sex + FamilySize	0.82603	0.78947
9. Survived ~ Age + Sex + FamilySizeNote	0.82379	0.78947
10. Survived ~ Age + Sex + Pclass + FamilySizeNote	0.82379	0.77990
11. Survived ~ Age + Sex + Pclass + FamilySize + Embarked	0.82379	0.78468
12. Survived ~ Age + Sex + Pclass + Fare + Title	0.83277	0.77990
13. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Title	0.84062	0.79425
14. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother	0.81705	0.78947
15. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote	0.83838	0.80382
16. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Embarked + Title + FamilySizeNote + TicketCount	0.84062	0.80382
17. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother +Embarked + Title + FamilySizeNote + TicketCount	0.83950	0.79904
18. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote + TicketCount + Deck	0.83613	0.79904
19. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount	0.84848	0.80382
20. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount + FamilyID	0.85409	0.80382
21. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch	0.85409	0.82296

Table 4: Performance summary of forests of conditional inference trees.

Generalized Linear Models (GLMs)

Generalized Linear Models estimate regression models for outcomes following exponential distributions. In addition to the Gaussian (i.e. normal) distribution, these include Poisson, binomial and gamma distributions [5]. In this project we do a binomial regression (classification).

First, we will build a generalized linear model to predict Survived using only the variables Age and Sex. We can now evaluate this model by computing its accuracy on the training set and the testing set (by submitting a .csv file to Kaggle). The accuracy on the training set is 0.78675 and on the testing set 0.76555.

Using different variables as predictors we can build new generalized linear models. Table 5 provides a summary of the model evaluation results. The best prediction for survival with generalized linear models was achieved using the formula "Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch". The accuracy of this model on the training and testing set is 0.85409 and 0.80382, respectively.

Formula	Accuracy (Training Set)	Accuracy (Testing Set) – Kaggle Score
1. Survived ~ Age + Sex	0.78675	0.76555
2. Survived ~ Age + Sex + Pclass	0.79461	0.75598
3. Survived ~ Age + Sex + Fare	0.78338	0.76076
4. Survived ~ Age + Sex + Pclass + Fare	0.79461	0.75598
5. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch	0.80583	0.75119
6. Survived ~ Age + Sex + Pclass + SibSp + Parch + Fare + Embarked	0.80583	0.76555
7. Survived ~ Age + Sex + Pclass + FamilySize	0.80583	0.75598
8. Survived ~ Age + Sex + FamilySize	0.79124	0.77033
9. Survived ~ Age + Sex + FamilySizeNote	0.80359	0.77033
10. Survived ~ Age + Sex + Pclass + FamilySizeNote	0.82154	0.75598
11. Survived ~ Age + Sex + Pclass + FamilySize + Embarked	0.80134	0.76076
12. Survived ~ Age + Sex + Pclass + Fare + Title	0.80246	0.77990
13. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Title	0.83277	0.78947
14. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother	0.80471	0.75598
15. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote	0.83389	0.77990
16. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Embarked + Title + FamilySizeNote + TicketCount	0.83389	0.78468
17. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother +Embarked + Title + FamilySizeNote + TicketCount	0.83501	0.78468
18. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount	0.83277	0.77990
19. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch	0.85409	0.80382

Table 5: Performance summary of Generalized Linear Models.

Gradient Boosting Machine (GBM)

The way the Gradient Boosting Machine (GBM) works is very similar to the Random Forest, since both approaches combine weaker models (typically decision trees) to predict the class. Their difference is that Random Forest trains the trees from different random subsets of the input data, while Gradient Boosting takes the error from the previous tree and uses it to improve the next one [6].

First, we will build a gradient boosting machine to predict Survived using the variables Age, Sex, Pclass, Fare, SibSp, Parch, Child, Mother, Embarked, Title, FamilySizeNote and TicketCount. We can now evaluate this model by computing its accuracy on the training set and the testing set (by submitting a .csv file to Kaggle). The accuracy on the training set is 0.83277 and on the testing set 0.74641.

Using different variables as predictors we can build new gradient boosting machines. Table 6 provides a summary of the model evaluation results. The best prediction for survival with gradient boosting machines was achieved using the formula "Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch". The accuracy of this model on the training and testing set is 0.85409 and 0.80382, respectively.

Formula	Accuracy (Training Set)	Accuracy (Testing Set) – Kaggle Score
1. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother +Embarked + Title + FamilySizeNote + TicketCount	0.83277	0.74641
2. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount	0.83164	0.78947
3. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch	0.85409	0.79904

Table 6: Performance summary of Gradient Boosting Machines.

Neural Network (NN)

A Neural Network (NN) is a computational model inspired by the way biological neural networks in the human brain process information. In other words, we can say that a neural network is a system composed of many simple processing elements which can acquire, store, and utilize experiential knowledge [7]. It consists of units (neurons), arranged in layers, which convert an input into some output. Each unit takes an input, applies a (often nonlinear) function to it and then passes the output on to the next layer [8].

In this section we will investigate if a more complicated model gives better results. First, we will build a neural network model to predict Survived using the variables Age, Sex, Pclass, Fare, SibSp, Parch, Child, Mother, Embarked, Title, FamilySizeNote and TicketCount. We can now evaluate this model by computing its accuracy on the training set and the testing set (by submitting a .csv file to Kaggle). The accuracy on the training set is 0.85521 and on the testing set 0.74641.

By using the predictor variables Age, Sex, Pclass, Fare, Title, Embarked, FamilySize, FamilyID , SibSp and Parch we achieve an accuracy of 0.87878 on the training set and 0.78947 on the testing set. It seems that it is very easy to overfit NN models.

Support Vector Machines (SVM)

We also consider Support Vector Machines (SVM) with Radial Basis Function (RBF) Kernel. Basically, the SVM classifier maps the input space into a new space by a kernel transformation, and then finds the optimal separating hyperplane and the margin of separations in that space. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new observations [9].

We are going to tune the SVM models to find the optimal parameter values, i.e., the gamma and cost values. The gamma parameter defines how far the influence of a single training example reaches, with low values meaning 'far' and high values meaning 'close', while the cost parameter is a regularization term that controls the complexity of the model. A high cost value will force the SVM model to create a complex enough prediction function to missclassify as few training points as possible, while a lower cost parameter will lead to a simpler prediction function [10].

First, we will build a SVM model to predict Survived using only the variables Age and Sex. We can now evaluate this model by computing its accuracy on the training set and the testing set (by submitting a .csv file to Kaggle). The accuracy on the training set is 0.80695 and on the testing set 0.74641.

Using different variables as predictors we can build new SVM models. Table 7 provides a summary of the model evaluation results. The best prediction for survival with SVM was achieved using the formula “Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch”. The accuracy of this model on the training and testing set is 0.85970 and 0.80382, respectively.

Formula	Accuracy (Training Set)	Accuracy (Testing Set) – Kaggle Score
1. Survived ~ Age + Sex	0.80695	0.74641
2. Survived ~ Age + Sex + Pclass	0.88888	0.69856
3. Survived ~ Age + Sex + Fare	0.80583	0.77033
4. Survived ~ Age + Sex + Pclass + Fare	0.83501	0.77033
5. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch	0.83164	0.77990
6. Survived ~ Age + Sex + Pclass + SibSp + Parch + Fare + Embarked	0.83501	0.79904
7. Survived ~ Age + Sex + Pclass + FamilySize	0.84175	0.77990
8. Survived ~ Age + Sex + FamilySize	0.82940	0.79425
9. Survived ~ Age + Sex + FamilySizeNote	0.82716	0.78947
10. Survived ~ Age + Sex + Pclass + FamilySizeNote	0.83613	0.79425
11. Survived ~ Age + Sex + Pclass + FamilySize + Embarked	0.83726	0.78468
12. Survived ~ Age + Sex + Pclass + Fare + Title	0.84287	0.76076
13. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Title	0.83277	0.78947
14. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother	0.84287	0.78468
15. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother + Embarked + Title + FamilySizeNote	0.83277	0.78947
16. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Embarked + Title + FamilySizeNote + TicketCount	0.83501	0.78947
17. Survived ~ Age + Sex + Pclass + Fare + SibSp + Parch + Child + Mother +Embarked + Title + FamilySizeNote + TicketCount	0.84287	0.78947
18. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySizeNote + TicketCount	0.85858	0.77990
19. Survived ~ Age + Sex + Pclass + Fare + Title + Embarked + FamilySize + FamilyID + SibSp + Parch	0.85970	0.80382

Table 7: Performance summary of Support Vector Machines.

Conclusion

From the experimental section, we see that the most accurate model is the forest of conditional inference trees when using the predictor variables Age, Sex, Pclass, Fare, Title, Embarked, FamilySize, FamilyID, SibSp and Parch. The accuracy of this model on the training set is 0.85409 and on the testing set 0.82296. At the time of writing, this model is on the top 3% of this Kaggle competition.

References

[1] T. Mitchell, *Machine Learning*, McGraw Hill, 1997

[2] L. Breiman, “Random forests”, Machine Learning, Vol. 45, No. 1, pp 5–32, 2001

[3] “Random forest”, [Online]. Available: https://en.wikipedia.org/wiki/Random_forest, [Accessed: 02- Nov- 2017]

[4] W. Yu and D. Chiu, Machine Learning with R Cookbook, 1st Edition, Chapter 5. Classification (I) – Tree, Lazy, and Probabilistic, p 166, Packt Publishing, 2015

[5] “Generalized Linear Model (GLM)”, [Online]. Available: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/glm.html, [Accessed: 02- Nov- 2017]

[6] T. Drabas, Practical Data Analysis Cookbook, Chapter 3. *Classification Techniques*, p 91, Packt Publishing, 2011

[7] A. Pandya and R. Macy, *Pattern Recognition with Neural Networks in C++*, p 43, CRC Press and IEEE Press, 1996

[8] J. Brownlee, “Non-Linear Regression in R”, 2014 [Online]. Available: https://machinelearningmastery.com/non-linear-regression-in-r/, [Accessed: 02- Nov- 2017]

[9] S. Wan and M.W. Mak, *Machine Learning for Protein Subcellular Localization Prediction*, p 167, De Gruyter, 2015

[10] A. Karatzoglou, D. Meyer and K. Hornik, Support Vector Machines in R, Journal of Statistical Software, Vol. 15, No. 9, 2006