Research and Analysis of Kaggle Titanic Project

Jonathan Mi

Summer Research Project
Hong Kong
mijonathan314@gmail.com

1. Background Information:

The sinking of the RMS Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the Titanic sank after colliding with an iceberg, killing 1502 out of 2224 passengers and crew. This sensational tragedy shocked the international community and led to better safety regulations for ships.

One of the reasons that the shipwreck led to such loss of life was that there were not enough lifeboats for the passengers and crew. 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.

In this project, the objective was to complete the analysis of what sorts of people were likely to survive. Also, in particular, there was an emphasis on demonstrating aptitude in data cleaning and classification with the programming language R.

2. Dataset Description

For this project, there were two datasets that were provided. First, there was the training data, which included 819 observations, one for each passenger. The training data was used to build a logistic model using machine learning tools in R. The information included in this dataset contained data such as the passenger's age, SibSp, which indicated the number of siblings and spouses the passenger had on board, and Pclass, which showed the passenger's ticket class at the time, ranked by the levels 1, 2, and 3, with 1 being first class, all the way down to 3 being third class. In addition, the training data had information regarding the passenger's sex, ticket number, cabin room number, the passenger fare amount, the point of embarkment, which was denoted as "Embarked", and finally Parch, which defined family relations, which displayed the number of parents or children aboard the Titanic. Most importantly, the training data set recorded if the passenger survived or not in the "Survived" column, showing 1 if the passenger lived, and 0 if he or she died.

The second dataset, called "Test", was used to see how well the model built from the training set performed on unseen data. The test had 418 observations, one for each passenger, and included all the information in the training dataset such as sex, age, and Pclass. However, the one difference is that in the test set, the dataframe did not include a column to indicate whether or not the passenger survived, because the objective of this project is to predict which passenger lived through, and test the model.

3. Data Cleaning

One of the most important tasks in this project was data cleaning. Many of the rows in the dataset had missing values for factors such as Embarked, Fare, and Age. In this model, one of the most important factors to fix was Age, because there were in total 263 combined missing values in the training and test dataset, as seen from the table on the right. *Age* had a major impact on the model, as shown from the chart

```
> table(is.na(titanic.full$Age))

FALSE  TRUE
 1046   263
```

below, which lists the Mean Decrease in Gini, or the average (mean) of a variable's total decrease in node impurity, weighted by the proportion of samples reaching that node in each individual decision tree in the random forest. In short, a higher mean decrease in gini indicates higher variable importance. So, instead of discrediting the feature, my model at first replaced the missing age values with the median age value of all the passengers in the training data set. However, this method was slightly inaccurate, and resulted in a decreased accuracy in the first model compared to my second predictive model, which utilized linear regression.

	MeanDecreaseGini
Pclass	35.67935
Sex	110.50877
Age	67.86136
SibSp	16.17461
Parch	11.34592
Fare	76.51207
Embarked	11.51875

Other variables that required data cleaning were Fare and Embarked. However, these factors, which are also displayed on the Mean Decrease Gini chart, clearly are not as important as Age. In addition, “Embarked” only had two missing values out of the 1309 total observations. As a result, for simplicity, in my model, I assigned the two missing values to be both S, which is the port with the most embarked passengers. Thus, for the Fare, which had only one missing value out of all the total observations, I just assigned the missing value to equal the mean passenger fare of the entire dataset.

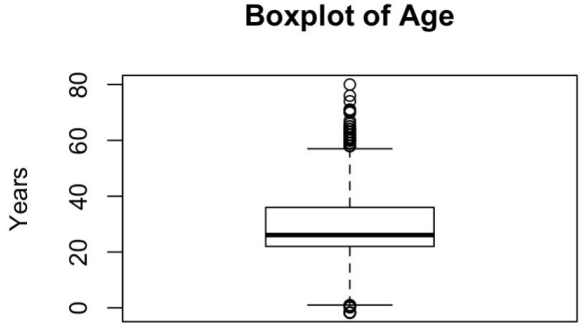
```
> table(titanic.full$Embarked)

  C   Q   S
270 123 916
```

Finally, the last variable that had missing values in the two datasets was the Cabin Number. For this variable, because there were lots of missing data observations, this feature was withdrawn from my predictive model in RandomForest.

4. Linear Model for Age

Before I created the linear model for age, I filtered out all the rows that are outliers in age. To do this, I created a box and whisker plot, and excluded out all the rows which had extreme age values above the 3rd quartile. Thus, my linear model can more accurately predict the unknown age values given the known variables.



Due to the large number of missing observations for age and the impact it has on the RandomForest model used to predict survival, for this analysis project, I decided to employ a linear model to fill in the missing age values. At first, the independent variables I formulated in this model were Pclass, SibSp, Embarked, and Parch. However, as seen on the summary of the linear model on the upper right, the variables with the lowest Pr(>|t|) values, which are indicated with three stars next to it on the chart, are Pclass and SibSp. This indicates that Pclass and SibSp are very strong predictors of Age and are also highly correlated with Age. In the RandomForest model, including the other two variables, Embarked and Parch, actually decreased the accuracy for survival. Thus, in my final model, my independent variables only included Pclass and SibSp to predict age, shown on the bottom right. This turned out to produce the most accurate results in predicting the survivability of the passengers.

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  39.0536     0.9449  41.329  < 2e-16 ***
Pclass2     -10.3120     1.1156  -9.243  < 2e-16 ***
Pclass3     -14.3996     0.9788 -14.712  < 2e-16 ***
SibSp        -3.2242     0.4478  -7.201 1.16e-12 ***
EmbarkedQ     3.7796     2.0089   1.881   0.0602 .
EmbarkedS     2.4182     1.0224   2.365   0.0182 *
Parch        -1.0046     0.4856  -2.069   0.0388 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.19 on 1029 degrees of freedom
  (263 observations deleted due to missingness)
Multiple R-squared:  0.2332,    Adjusted R-squared:  0.2288
F-statistic: 52.16 on 6 and 1029 DF,  p-value: < 2.2e-16
```

```
> summary(age.model)

Call:
lm(formula = age.equation, data = titanic.full[age.filter, ])

Residuals:
    Min      1Q  Median      3Q     Max
-35.177  -7.628  -1.080   7.377  38.920

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  45.7249     1.0886   42.00  < 2e-16 ***
Pclass       -6.5483     0.4553  -14.38  < 2e-16 ***
SibSp        -3.4713     0.4177   -8.31 2.99e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 12.29 on 1033 degrees of freedom
  (263 observations deleted due to missingness)
Multiple R-squared:  0.2181,    Adjusted R-squared:  0.2166
F-statistic: 144.1 on 2 and 1033 DF,  p-value: < 2.2e-16
```

5. RandomForest

For my predictive model, I decided to use RandomForest, which consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest spits out a class prediction. For the factors that affect survival,

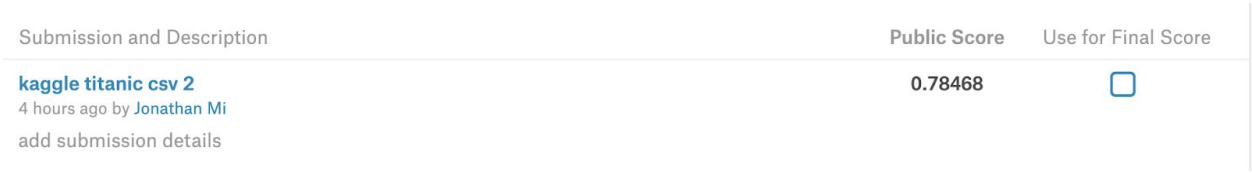
```
survived.equation<-"Survived~Pclass+Sex+Age+SibSp+Parch+Fare+Embarked"
survived.formula<-as.formula(survived.equation)
install.packages("randomForest")
library(randomForest)
titanic.model<-randomForest(formula = survived.formula, data = titanic.train, ntree = 500, mtry = 3
#featured.equation<-"Pclass+Sex+Age+SibSp+Parch+Fare+Embarked"
Survived<-predict(titanic.model, newdata=titanic.test)
```

I decided to include Pclass, Sex, Age, SibSp, Parch, Fare, and Embarked, as shown above in the code. The variables that affected survival the most were Sex, Age, Fare, Pclass, etc, in descending order, which can be inferred by the Mean Decrease Gini ranking on the right.

	MeanDecreaseGini
Pclass	35.55196
Sex	109.68381
Age	66.83579
SibSp	16.35085
Parch	11.19310
Fare	77.39679
Embarked	11.57464

6. Final Result and Error Analysis

For my kaggle titanic accuracy score, I received roughly 78.5% accuracy with my linear regression and RandomForest model. This model, while simplistic, proved to be fairly effective, considering that there were only roughly 819 observations, and a perfect model would require more observations. Also, survivability can occur due to luck, which cannot be inferred or predicted from the given variables. However, there can be room for future improvement, such as incorporating cabin number in analyzing survivability, because certain locations on the ship, which may be closer to exits, may lead to higher chances of surviving. In conclusion, though the current model could already decently predict and analyze what sorts of people were most likely to survive, future developments can still be implemented to create a more efficient model.



Acknowledgements

This report was done by Jonathan Mi

References

[1] Kaggle.com. “Titanic: Machine Learning from Disaster.” *Kaggle*, 15 July. 2019, www.kaggle.com/c/titanic/data.

[2] James, Gareth. *An Introduction to Statistical Learning: with Applications in R*. Springer, 2017.

[3] “Tutorials Library.” *Boon, MuleSoft, Nagios, Matplotlib, Java NIO, PyTorch, SLF4J, Parallax Scrolling, Java Cryptography, YAML, Python Data Science, Java i18n, GitLab, TestRail, VersionOne, DBUtils, Common CLI, Seaborn, Ansible, LOLCODE, Current Affairs 2018, Apache Commons Collections*, www.tutorialspoint.com/r/.

[4] “Understanding Random Forest - Towards Data Science.” *Medium*, Towards Data Science, 17 July 2019, towardsdatascience.com/understanding-random-forest-58381e0602d2.

[5] “Titanic: Machine Learning from Disaster.” *Kaggle*, www.kaggle.com/c/titanic.