MATH4432: Project 2

Kaggle contest classification: Predict survival on the Titanic

Sarah Catherine James 20501098
Alexander Moellers 20501050
Nora Eliza Norden 20500343

Thursday 12th April, 2018

1 Introduction

The RMS Titanic sank in the early hours of 15th April, 1912 during its voyage to New York City from Southampton, having been badly damaged during a collision with an iceberg. This led to the death of 1502 out of an estimated 2,224 passengers and crew aboard.

The overwhelming number of deaths has predominantly been put down to unsafe regulations, specifically regarding number of lifeboats- there simply were not enough lifeboats onboard to save many of the passengers. As we have an explanation for why the death toll was so high in general, we will now aim to find an explanation for the distribution of these deaths across variables such as class, gender, and age. In this project we will carry out our own analysis of the available data to identify the most significant passenger characteristics in regards to likelihood of survival, and ultimately create a highly accurate algorithm to predict who survived, and who did not.

This project is based on the Kaggle competition, Titanic: Machine Learning from Disaster and was implemented in R. For our full code, visit the project repository.

2 Feature engineering

The Titanic dataset consists of 10 predictors as seen in table 1. Furthermore, each passenger is given a passenger ID and a survival indicator (0/1). The survival is only given for the training set. Two predictors in particular seem to influence the survival rate. These are `Sex` and `Pclass`, visualized in figs. 1a and 1b.

Firstly, consider `Pclass`. From a first glance at fig. 1b, it is apparent that a significantly higher number of people from third class died compared to first and second, indicating that `Pclass` is an important predictor of survival. Intuitively one may explain this with the prioritization of people of a higher class- they would be deemed more important and so more effort would be put into ensuring their survival. Furthermore, they would be in a cabin higher up in the ship, and so would logistically be in a better position to get to the lifeboats first, and hence claim a seat on one. `Pclass` is therefore concluded significant for our algorithm.

Table 1: List of predictors.

Variable	Definition
`Pclass`	Passenger class. 1, 2, or 3 for 1st, 2nd and 3rd class respectively.
`Name`	Surname, followed by title and given name.
`Sex`	Male or female.
`Age`	Age.
`SibSp`	Number of siblings and spouses aboard.
`Parch`	Number of parents and children aboard.
`Ticket`	Ticket number.
`Fare`	Fare for the whole ticket.
`Cabin`	Cabin number, starting with deck (A-G, T).
`Embarked`	Port of embarkation. C/Q/S for Cherbourg, Queenstown or Southampton.

Next, the variable `Sex` is considered. From fig. 1, it is observable that a much larger proportion of men died when compared to women, suggesting that `Sex` should also be deemed a variable with strong predictive power. Again, this fits in with intuition as the idea that the lives of women and children take priority over that of men's fits in with the context of the time of the shipwreck, and this idea would have been acted on when selecting who to put into a lifeboat.

A third variable, `SibSp`, seemed worth investigating. In order to test the predictive power of `SibSp`, it is converted from a continuous variable into a factor. It is clear from fig. 1c, that a passenger without a spouse or less siblings is more likely to survive. This may be because as a spouse you have another person to look after and so are less likely to survive than if you are alone, and the more siblings you have, the more difficult it is for your parents to save you.

The variables `Pclass`, `Sex` and `SibSp` all have a clear link to a passengers title within their name. `Pclass` and `Sex` are both captured by title, and `SibSp` implies a passengers title- if you are travelling with a larger number of siblings, you are most likely a child travelling with family, and so would have the title 'Miss' or 'Master'. Clearly a passengers title has strong predictive power, and this will be considered when creating our algorithm. The actions taken as a result of our data exploration are outlined in the subsection below.

Before fitting a model to the training set, the data requires cleaning. The first issue is that the data set is incomplete - cabin number is only given for 295 of 1309 data points in the training and test sets combined. The data set is also missing 2 embarkation ports, 1 fare and 263 ages. Another issue is the prevalence of some seemingly uninformative predictors. `Cabin`, `Ticket` and `Name` fall into that category, since the taken values are largely unique or shared by a small number of passengers. Thus, a model fitted to these variables will lack in generality. They may still hold valuable information and shouldn't be discarded immediately. Instead, we will attempt to extract features of interest from them.

2.1 Feature Extraction

We start with the `Name` variable, firstly ensuring there were no double entries leading to inaccurate analysis. Intuitively, a person's title captures more information than their given name and surname, at least in regards to social status, which in turn could make title a good indicator for survival rate. It's usefulness is also supported by our findings on the predictive power of `Pclass` and `Sex` outlined above,

which title also implies. Therefore, the titles are extracted from the names, and a separate variable `Title` is created. Again, most of the values for `Title` are very infrequent, some appearing only once. To address this issue and maintain the predictive power of `Title`, the number of values are reduced according to table 2. The 'Other' category consists of professional, military and honorary titles. This grouping is appropriate based on the exploration of `Pclass`. Later exploration implies that the titles included in 'Other' may need to be refined as we see their title did not help them survive, for example, 'Dr' and 'Rev'. Furthermore, we grouped together any titles that imply the same characteristics but are country-specific. For example, "Mademoiselle" equates to "Miss".

The survival count of the different groups is seen in fig. 2. Much of the title-based survival can be explained by the sex- 'Master' and 'Mr' are male titles whereas 'Miss' and 'Mrs' are female. However, the 'Other' class is mainly male but has a higher survival rate than males in general (compare to fig. 1a), again supporting the idea that nobility will lead to prioritization.

The cabin number variable suffers from the same problem as the passenger name, having too many, too infrequent values. A simple solution is to copy the first letter - the deck - to a `Deck` variable. For now, the missing data will be assigned deck 'U' for unknown.

Moving on to the next predictor, we note that the fare sometimes takes strange values. More specifically, the fare is equal for all passengers with the same ticket number, and generally much higher than the corresponding fares of single-passenger tickets. For example, 3 passengers have the ticket number 110152

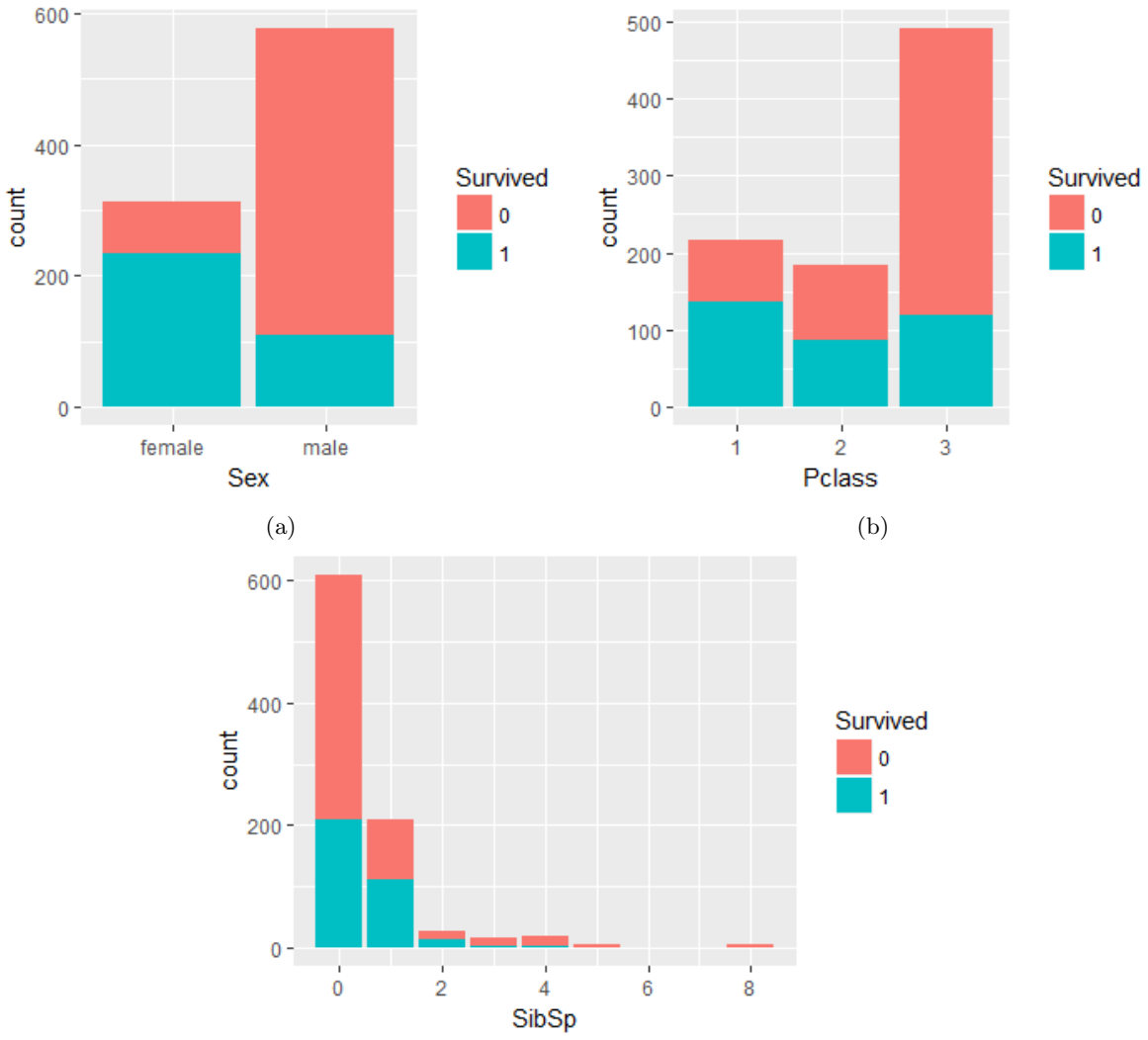

Figure 1: Survival based on (a) sex, (b) passenger class and (c) siblings and spouses.

Table 2: List of titles.

New title	Old title(s)	Count
Master	Master	61
Miss	Miss, Ms, Mademoiselle	264
Mr	Mr	757
Mrs	Mrs, Madame	198
Other	Captain, Colonel, Don, Dona, Dr, Jonkheer, Lady, Major, Reverend, Sir, Countess	29

and each passenger has a fare of 86.5, whereas the 1 passenger with ticket number 110564 has a fare of 26.55, which is more than three times 86.5. It appears that all passengers with the same ticket have been given the total ticket fare of all tickets bought together. A new variable, `FareAdj`, is created, which values correspond to the fare per passenger. This is calculated by simply dividing the total fare in `Fare` by the number of passengers with the same ticket number, as the travel companions stay in the same passenger class and often share cabins. The new variable is easier to interpret for humans and, hopefully, also for a machine. It is, however, flawed by design as it doesn't consider the possibility that e.g. a child fare is cheaper than an adult fare.

The size of a family can be calculated as the sum of `Parch` and `SibSp`, plus 1 for the passenger himself. This new variable, `FamSize` might illustrate how solo-travellers fare versus small to large family groups. Similarly, a `TravelSize` variable is created by looking at the number of people sharing the same ticket, which implies that they bought the trip together. Unlike `FamSize`, TravelSize captures other travel companions like fiancées and friends.

The new set of variables and possible predictors are summarized in table 3.

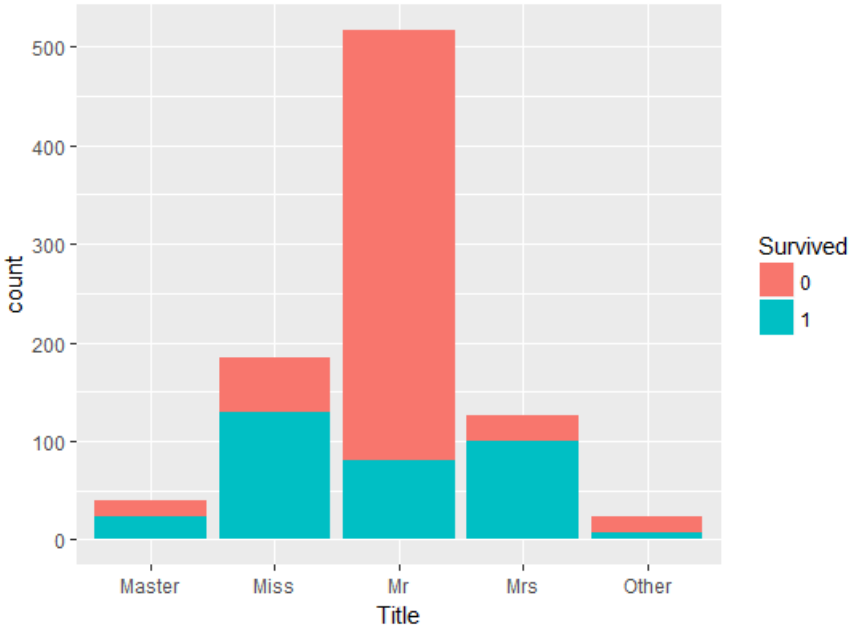

Figure 2: Survival based on title.

Table 3: List of predictors after feature extraction.

Variable	Definition
`Pclass`	Passenger class. 1, 2, or 3.
`Title`	Master, Miss, Mr, Mrs or Other.
`Sex`	Male or female.
`Age`	Age.
`FamSize`	Family size.
`TravelSize`	Travel company size.
`FareAdj`	Fare adjusted for a single passenger.
`Deck`	Deck A-G, T, or U for unknown.
`Embarked`	Port of embarkation, C, Q or S.

2.2 Imputation

To deal with the missing values, data imputation is used. Very basic approaches include filling in non-values with the variable mean, median or mode. We will however use more sophisticated machine learning methods in order to try to reduce the noise added.

Starting with the `FareAdj` variable with one omitted value, we find that the passenger fare ranges from 0 to 128, with a median of 8 and mean of 15. The missing value is predicted using a regression tree model and the predictors `Sex`, `Pclass`, `Title`, `Deck`, `Embarked`, `TravelSize` and `FamSize`. The predicted value is 12.68, which is close to the mean. Therefore, the model seems valid.

Also, the two missing embarkation values are estimated using a classification tree, predicting "S" (Southampton) for both points. This is also the mode of `Embarked`, and so the predictions are not unexpected.

The missing `Age` values are more carefully predicted using a random forest of 500 trees because of the sheer quantity of the NA's. Passenger fare, title and family size seem suggestive of age and are thus used as predictors. The prediction median and mean are close to those of the known data. The predictions are also more conservative, having a much smaller standard deviation.

3 Algorithm and Feature Selection

For this project the team chose to use the random forest algorithm as it comes with several advantages. Firstly, it gives accurate predictions and does not overfit easily. Secondly, the algorithm outputs an interpretable model and facilitates feature selection by automatically estimating variable importance. Although different R Random Forest packages were tested, the team ended up using the standard *randomForest* package. As an initial step, the algorithm was implemented using all available features and the importance estimate was used to confirm the insights gained into the predictive power of the predictors. After this, a group of of variables could be isolated to be considered in further exploratory modelling. These features were `FareAdj`, `FamSize`, `TravelSize`, `Title`, `Sex`, `Parch`, `Deck` and `Pclass`. Hereby, the most powerful predictors were `Title` and `Pclass`, which thus were included in nearly all future models. Furthermore, the selection of these two variavles also helped to exclude other features which contained similar information. An example for this is `Sex`, which can be discarded when `Title` is used. Moreover, `Deck` was removed from the modelling, as it is already included in `Pclass`. Now, the exploratory modelling could start.

Table 4: Predicted test accuracy by cross validation.

Number of predictors	Accuracy
2	0.834
3	0.832
4	0.827

To precisely estimate the changes in accuracy for different predictors the cross validation method was used. The original training set was partitioned into several folds. For every round of cross validation, one of the folds was used for training and the rest for testing. In the end, this method was very helpful for quickly and easily predicting how the models would perform in practice. This facilitates the process of exploratory modelling and soon the combination of `FareAdj`, `TravelSize`, `Pclass` and `Title` turned out to produce the best model. A thousand trees were fitted and the the tunelength was set to 3. This parameter was helpful for identifying the important predictors, but it was no longer necessary once the features had been found.

4 Result and Analysis

With the above mentioned random forest algorithm and set of predictors, survival predictions on the test set yielded an accuracy of 79.99 % upon submission to Kaggle. As of Thursday 12$^{\text{th}}$ April, 2018, this score led to position 1683 of 10,812 on Kaggle's public leaderboard. While not ranked among the top 10 %, our model has been kept fairly simple and general so as to not overfit the small data set. With regards to the Kaggle competition, this is particularly important since we are only given 50 % of the test data, and the final results will be determined by the accuracy on the other half when the competition ends. It should also be noted that the top rankers on the public leaderboard have scored a perfect 100 % accuracy. This is most certainly due to overfitting or manipulation of the submitted predictions, and not due to a 'perfect' model, since the latter doesn't exist.

Cross validation showed that it was beneficial to keep the number of predictors sampled in each split small. Of four given predictors, it was found that $m = 2$ randomly sampled predictors gave the highest predicted test accuracy, see table 4. The confusion matrix of the final model is seen in table 5. The out-of-bag error rate of 16.27% approximately equals the predicted error by cross validation. While the model accurately predicts casualties, it is weaker at predicting which passengers survived. This is not completely unexpected, since the majority of passengers in the training set are deceased. However, it still implies that the model does not have a strong understanding of what separates the individuals who survived from the others.

Based on experiments, this flaw is more likely to be attributed to the predictor classes rather than the choice of learning model. For instance, the `Title` variable proved to be significant for the accuracy. The different values the variable can take has varied over the course of the project. The first adaptation consisted of the groups 'Lady', 'Sir', 'Dr', 'Reverend' and 'Colonel', all of which have been combined into the 'Other' category in the final edition, see table 2. Having a larger quantity of higher-detail groups led to a significantly worse classification accuracy. The probable explanation for this was the low data frequency, with the groups holding less than 10 data points each, leading to overfitting. It is possible that another grouping of the titles would increase the accuracy further, and it remains to be investigated.

Analogously, the significant predictor `FareAdj` is in fact an oversimplification and might not accurately depict the real fare for each passenger. As mentioned in section 2.1, the total ticket fare is split equally among the number of passengers on the ticket, without considering price disparities e.g. between age groups. Hence, a non-negligible quantity of noise may have been introduced to the model. Noise has also been added through data imputation, in particular in the age data. The level of noise is hard to predict and the `Age` variable was deemed to be unusable. Consequently, it was removed from the predictor set used to train the model, which prevented the noise from reaching the model and affecting the test accuracy. It is worth noting that a passenger's age can be partially implied from their title, and so the

Table 5: Confusion matrix.

	0	1	Classification error
0	504	45	0.082
1	100	242	0.292

predictive power of age is still present in our algorithm to some extent. Nevertheless, the possibility exists that a better age prediction model that yields different values would turn age into a significant variable, making the inclusion of age as a predictor beneficial.

To conclude, using random forest to predict survival on the Titanic data set has been largely successful, despite - or because of- the simplicity of the model. We also found that feature extraction played a key part in fitting an accurate model, and the accuracy can probably be boosted by further feature exploration. Many of the features are somewhat related, for example title and age. So are passenger class, fare, cabin and title, as all relate to the wealth and social status of the passengers. Creating new variables based on the combination of these may therefore make the underlying patterns more apparent and easier for the model to pick up.

A Individual contributions

Alexander and Eliza were together responsible for feature engineering. The code is primarily written by them. During this task Alexander also created many of the plots in the report. These were later refined and interpreted in the report by Sarah, who was responsible for section 1 and the first part of section 2, as well as part section 2.1. A big part of section 2 and the majority of section 4 were written by Eliza. Alexander was mainly responsible for section 3, a lot of his time went into exploratory modelling to find the best features for the model and improve the accuracy.