# Prediction on the Survival of Titanic Passengers

Rikky Aguila
Joshua White
Michael Thomson

May 20, 2022

## Executive Summary

This is a second attempt at the titanic data set, with the goal of improving the accuracy achieved on the last attempt. For the first attempt, the highest score was a random forest model with 77.51% accuracy. Many different feature engineering techniques were utilized to improve the model, including clustering and the addition of a full title set. Now the best model that we were able to produce was a lasso logistic model that had an accuracy of 79.186% which is a modest improvement on the previous attempt. The table below shows how many passengers survived and did not survive the titanic based on the model's predictions, as well as the true accuracy of the predictions.

| Survived | Total | Percent Survived | Kaggle Accuracy |
|---|---|---|---|
| 145 | 418 | 34.69% | 79.186% |

# Introduction

For this project, data for 891 passengers aboard the Titanic was analyzed. The goal was to use this data to create a model that would predict the survival of an additional 418 passengers. Some of the variables included: `Survived, Name, Sex, Age, Fare, Cabin,` and `Embarked`.

## Data Imputation

Analyzing this data, the first thing of note was that there were many `NA` values for the `Age` category, denoting missing values. A package called `mice` was utilized to impute the missing entries in `Age`. The `mice` package provides many methods for imputing data, and the one used was the predictive mean matching (pmm) method. A single observation for fare and two for `Embarked` were also imputed in this way. Additionally, a new categorical variable, `AgeCat`, was created to indicate which Age variables had been imputed. There was data missing for `Cabin` as well, so a new category was added within `Cabin`, named "O", that signified a missing `Cabin` value. Another categorical variable, `CabinLet`, was added, containing the first letter of cabin, since different lettered cabins could potentially determine position on the ship, and the number following each cabin value was determined to be unimportant for prediction.

| Passenger ID | Age | Predicted Age |
|---|---|---|
| 1 | 22 | 22 |
| 2 | 38 | 38 |
| 3 | 26 | 26 |
| 4 | 35 | 35 |
| 5 | 35 | 35 |
| 6 | NA | 41 |
| 7 | 54 | 54 |
| 8 | 2 | 2 |
| 9 | 27 | 27 |
| 10 | 14 | 14 |

Table 1: Comparison of age and predicted age

## Feature Engineering

One more categorical variable that was added is `CabinMult` that signaled if an individual had multiple cabins. For the original attempt, two new variables were extracted from the `Name` category, `HasMaster` and `HasRev`, that separated names that had the titles Master or Rev. For this attempt, unique titles were created for all passenger strings, resulting in 18 unique titles: `Mrs, Mr, Don, Miss, Master, Mme, Lady, Sir, Dr, Mlle, Col, Rev, Ms, Major, Capt, the Countess, Jonkheer,` and `Dona.` The titles `Dona, Lady, Madame, Mme,` and `the Countess,` denoting feminine nobility, were combined to `Lady`. The titles `Don, Jonkheer,` and `Sir,` denoting masculine nobility, were combined to `Sir`. `Miss, Ms,` and `Mlle` were combined to `Miss` and `Major` and `Col` were combined to `Officer`, resulting in 10 unique titles. One final category was added, a 929-level factor called TicketGroup indicating if passengers shared the same ticket and therefore traveled together.

Something new that we tried to look at for this data set was to use clusters as a form of feature engineering. The idea behind this was that the clusters may be able to separate distinct groups not included in the original data. To decide on the amount of clusters that we want to use we created a scree plot and based the number of clusters off of that.

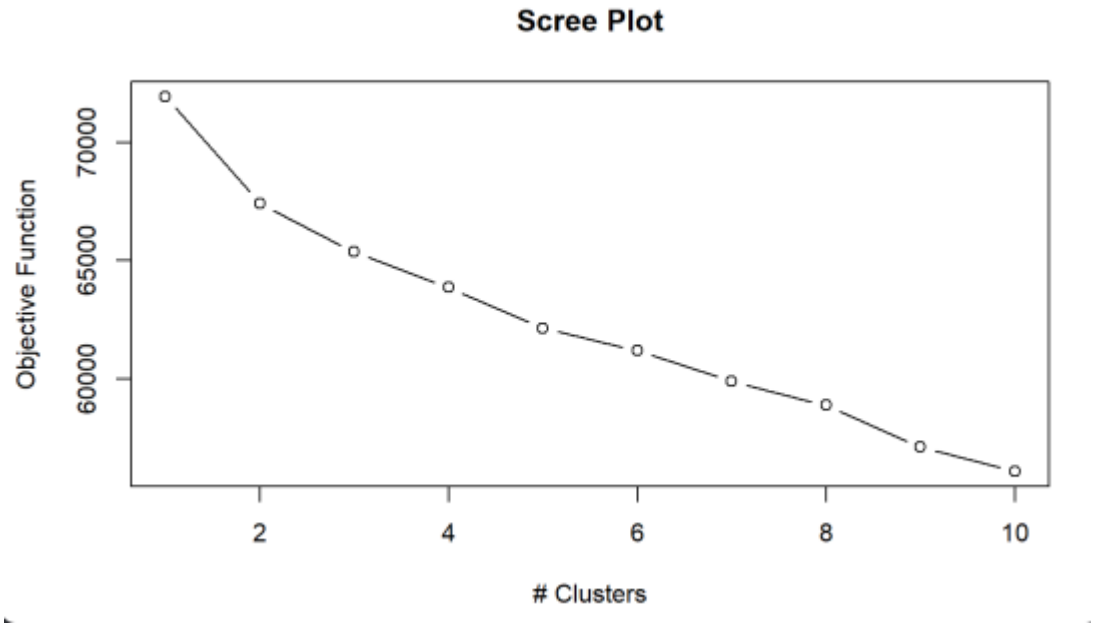



Figure 1: This is a scree plot of the clusters

From the scree plot we can see that the there is a noticeable drop from the the $8th$ cluster to the $9th$, so we decided to use 9 clusters. After trying to use these clusters in our model we found that none of them were significant. This may be do to the fact that this testing set wasn't big enough (418) to give us meaningful cluster groups.

With feature engineering complete, the 891 passengers were split into a training set containing 75% and a test set containing 25% . The goal was to accurately predict the final 25% of the data using the initial 75%. After the best model was determined using the split, a final model was made using the full 891 data points to predict on the 418 for which predictions were desired.

A lasso model was created for feature selection using the full set of squared interactions. The model selected 107 variables, which is too many to list, but most of these variables were specific ticket groupings. Most ticket groupings that contained more than one observation were significant, as groups traveling together tended to have the same fate.

## Results

The first three results below are from our original attempt at the titanic dataset. The three models with the top AUC values are shown in Table 2 below. The three models are General Linear Model (GLM), Random Forest, and a Support Vector Machine (SVM). The new Lasso model had far superior AUC but lower accuracy. It turned out that all of the original models were overfit compared to the New Lasso Model.

| | GLM | Random Forest | SVM | New Lasso |
|---|---|---|---|---|
| Overall Accuracy | 0.8558559 | 0.8648649 | 0.8423423 | 0.8153153 |
| AUC | 0.8877073 | 0.8924838 | 0.8678803 | 0.9354619 |

Table 2: Accuracy performance on three classification methods

Below are plots demonstrating the performance of the orignal three models on the training and test sets measured by AUC. Note that GLM performs very well on the training set but the Random Forest overtakes it on the test set.

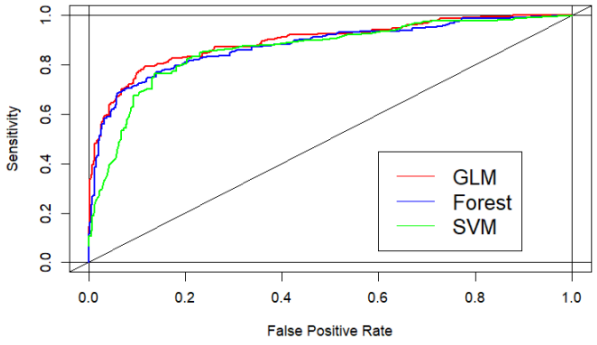



Figure 2: ROC comparison on training set

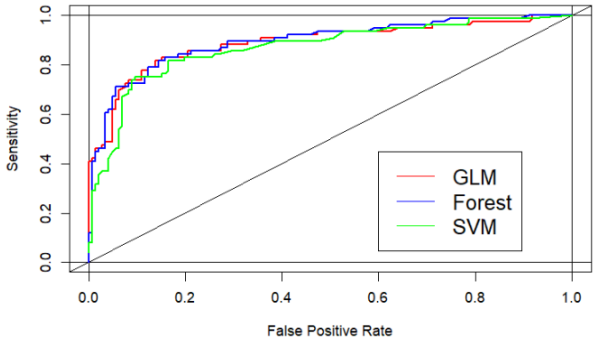
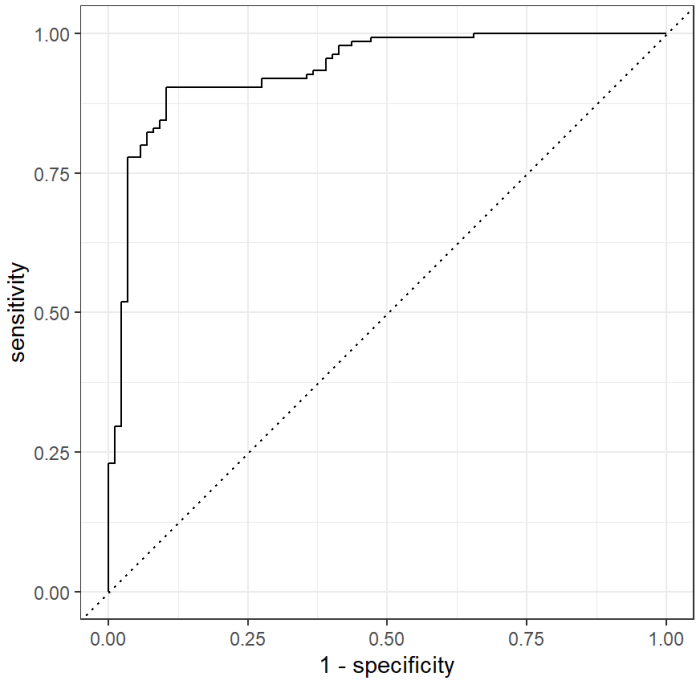



Figure 3: ROC comparison on test set

Here is the ROC curve for the New Lasso model used to generate the final predicitons for this attempt.



## Conclusion

The final Lasso model created was demonstrably better than any of the previous models from the first attempt. The new feature engineering including a full title set and unique identifiers for each ticket proved useful in obtaining modestly better results. It is uncertain what more can be done to further improve predictions beyond what we have achieved here. We suspect that more accurately predicting `Age` or utilizing ensemble models with different weaknesses may be some of the last steps to achieving a more optimal model.

| Survived | Total | Percent Survived | Kaggle Accuracy |
|---|---|---|---|
| 144 | 418 | 34.45% | 77.511% |

Table 3: Original Survival Predictions and Kaggle Accuracy

| Survived | Total | Percent Survived | Kaggle Accuracy |
|---|---|---|---|
| 145 | 418 | 34.69% | 79.186% |

Table 4: New Survival Predictions and Kaggle Accuracy