Predicting the Survival of Titanic Passengers

Prepared by: Pranav Reddy Machannagari
Teammate: Samhitha Kasireddy
Report Date: 12/20/2021

Introduction

We will be constructing a machine learning model using the well-known Titanic dataset. It gives details on the Titanic's passengers' survival, broken down by economic standing (class), gender, age and other attributes. Because there were insufficient lifeboats for everyone, 1502 of the 2224 persons on board died, however survival or death was not random. The task now is to create a predictive model based on Kaggle's data to answer the question, "What sorts of attributes define survivors?"

Business Use Cases

There are a couple of business use cases we could think of here:

1) We are interested in understanding what characteristics of a person on board more or less defined their survival. This would be helpful for Naval forces or Ship companies all around the world to accommodate high survival score people on high-risk voyages in order to minimize the loss of life in worst case scenarios. For the same reason, the accuracy of this model becomes all the more important.
2) Besides, we are also interested in building this model for the army naval forces to use in case of disasters like Titanic sink that could happen, to know who could have survived the disaster and initiate priority tracking measures accordingly. If our model prediction is accurate and we predict that there is a possibility of survival for a particular naval officer, it would make more sense to prioritize tracking these people.

Hypothesis

If you've seen the movie Titanic, you'll recall that the wealthy were more likely to survive. Children, women, and the elderly were also given priority. So, according to our hypothesis, older wealthier women and children had the best chance of surviving, whereas poor middle-aged males had the worst chance.

Data Exploration

There are 891 samples in the training set, with 11 features plus the target variable (survived). There are two floats, five integers, and five objects among the features.

Only 38% of the training data survived the Titanic. The passenger ages range from 0.4 to 80. Furthermore, we can already recognize several features with missing data, such as the 'Age' and ‘Cabin’ features.

We could see that men have a higher chance of survival between the ages of 18 and 30, which is partially true for women but not entirely. Women have a better probability of surviving between the ages of 14 and 40. Men have an extremely low chance of surviving between the ages of 5 and 18, but this is not the case for women. It's also worth noting that infants have a slightly higher chance of surviving.

Pclass plays a role in a person's survival chances, especially if they are in class 1. We can also see that a person in Pclass 3 has a high chance of not surviving. Besides, with 1 to 3 relatives, you have a higher chance of survival, but with less than 1 or more than 3 you have a reduced chance.

Depending on the gender, being embarked also appears to be linked to survival. Women who are on ports Q and S have a better probability of surviving. If they're at port C, the opposite is true. If men are on port C, they have a good chance of surviving, but if they are on port Q or S, they have a low chance of survival.

Data Processing

First, I'll remove 'PassengerId' from the train set because it has no bearing on a person's chances of surviving.

Missing Data
Cabin (687), Embarked (2), and Age are all missing data points (177). A typical cabin number is 'C123,' with the letter referring to the deck. As a result, we'll extract these and create a new feature with a person's deck. The values that are missing will be set to zero. For age, we’ll make an array with random integers that are calculated using the mean age value. Because the Embarked feature only has two missing values, we'll just use the most common one as a replacement.

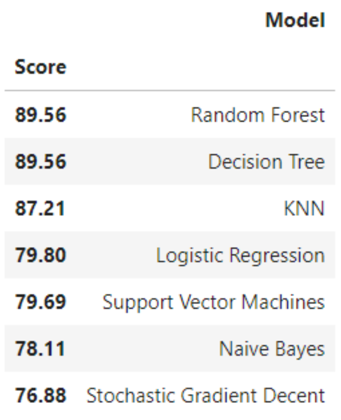
Converting Features
We used "astype()" to convert "Fare" from float to int64. We have taken the Name feature to extract the Titles from the Name and used them to create a new feature. We converted the 'Sex' feature to a numeric value. Further, it will be difficult to transform the 681 unique tickets in the Ticket attribute into useful classes. As a result, we choose to exclude it from the analysis. We also converted the 'Embarked' characteristic to a numerical value.

Creating Categories
By categorizing each age into a group, we will establish the new 'AgeGroup' variable and do the same thing with the ‘fare’ using qcut() function.

Also, we are adding a couple of relevant features in Age*Class and Fare per person.

Building and Testing several ML Models

We trained and compared the results of numerous Machine Learning models.

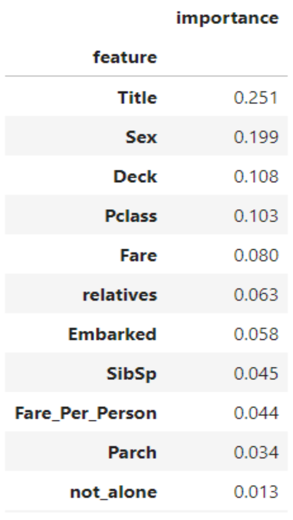


Score	Model
89.56	Random Forest
89.56	Decision Tree
87.21	KNN
79.80	Logistic Regression
79.69	Support Vector Machines
78.11	Naive Bayes
76.88	Stochastic Gradient Decent

On top of the heap is the Random Forest classifier.

Our model has an average accuracy of 82 percent and a standard deviation of 4 percent after 10-fold cross validation. This suggests that the accuracy of our model can vary by +/- 4%. We believe the accuracy is still very good.

Another excellent characteristic of random forests is that they make determining the relative value of each variable very simple. Sklearn determines the relevance of a feature by examining how much impurity is reduced on average by tree nodes that employ that feature. After training, it automatically calculates this score for each feature and scales the findings so that the sum of all importances equals 1.



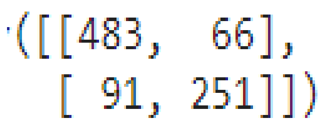
feature	importance
Title	0.251
Sex	0.199
Deck	0.108
Pclass	0.103
Fare	0.080
relatives	0.063
Embarked	0.058
SibSp	0.045
Fare_Per_Person	0.044
Parch	0.034
not_alone	0.013

In our random forest classifiers prediction procedure, neither not_alone nor Parch play a significant role. As a result, I'll remove them from the dataset and retrain the classifier, we get model accuracy 92.82%.

Model Evaluation

Confusion Matrix:

The first row is about not-survived-predictions: 483 passengers were accurately classified as not survived (called true negatives) and 66 passengers were incorrectly classified as not survived (called false negatives) (false positives).



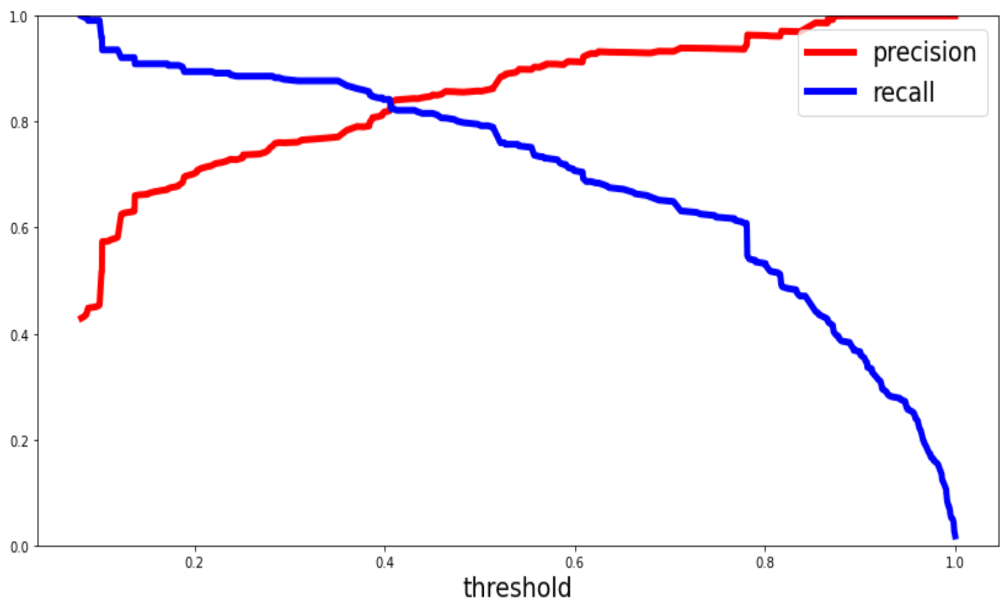
```
([[483,  66],
  [ 91, 251]])
```

The second-row concerns survived-predictions: 91 passengers were incorrectly categorized as survivors (false negatives), while 251 were accurately labeled as survivors (true positives).

Our algorithm correctly predicts a passenger's survival 79% of the time (precision). According to the recall, it correctly predicted the survival of 74% of those who survived.

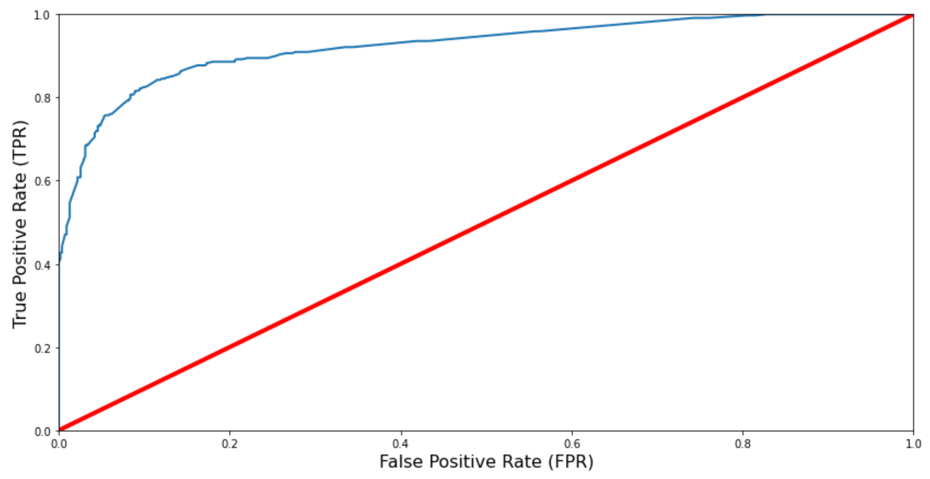
Precision Recall Curve:

The Random Forest algorithm computes a probability based on a function for each person it has to classify and then classifies the person as having survived (when the score is higher than the threshold) or not having survived (when the score is lower than the threshold) . As a result, the threshold plays a crucial role.



As you can see in the graph above, the recall is steadily decreasing at a precision of roughly 85%. As a result, you might wish to choose the precision/recall tradeoff before that, perhaps at around 75%.

Area under the ROC curve:



ROC-AUC-Score: 0.9243

Summary

We began with data exploration, which allowed us to get a sense of the dataset, check for missing data, and determine which aspects are most essential. We computed missing values, transformed features to numeric ones, sorted values into categories, and created a few additional features during the data processing phase. Following that, we began training eight different machine learning models before selecting random forest and performing cross validation on it. Finally, we calculated the model's precision and recall using its confusion matrix and evaluated it using Area under the ROC curve and Precision Recall Curve.

The work has been submitted to the Kaggle: Titanic – Machine Learning from Disaster Data Science Competition; Link: https://www.kaggle.com/c/titanic

The relevant output files have been attached with this report.