GB656 Final Project

Kaggle – Titanic Competition

Yuzhe Zhao
yuzhe.zhao@wisc.edu

CONTENT

Introduction

Problem Statement

At 2:20 a.m. on April 15, 1912, the luxury cruise liner Titanic sank in the North Atlantic Ocean after running into an iceberg at around 11:40 p.m. on April 14, 1912. More than 1500 of the estimated 2,224 passengers and crew aboard died in this disaster, making the sinking at the time the deadliest of a single ship in the West.

In the process of compiling the data of the passengers on the Titanic, people found that the probability of survival was different for people with different features. In another word, there are some people have a better chance of survival than others.

Therefore, in this article, the data of passengers will be used to build machine learning models to answer the question that "what sorts of people were more likely to survive". Data of passengers were provided by Kaggle, here is the link of the competition and data: https://www.kaggle.com/c/titanic.

Data Preparation

Load data & Overview

There are two datasets: train.csv and test.csv. There are 12 features of 891 passengers in the train dataset and 11 features of 418 passengers in the test dataset. The train.csv dataset has one more feature about whether the passenger survived.

Here is the illustration of the means of all features:

1. PassengerId is the unique id of passengers.
2. Survived is whether the passenger survived:

   1 = Survived

   0 = Not Survived
3. Pclass is the passenger class (1st, 2nd, 3rd), which is a proxy of socio-economic class:

   1 = Upper Class

   2 = Middle Class

   3 = Lower Class
4. Name is the passengers' name with titles.
5. Sex:

   female = female

   male = male
6. Age is passenger’s age
7. SibSp is the number of the passengers' siblings and spouse that were on board
8. Parch is the number of the passengers' parents and children that were on board

9. Ticket is the ticket Id

10. Fare is the ticket prices

11. Cabin is the cabin number of the passenger

12. Embarked is the place that the passenger embarked:

    C = Cherbourg

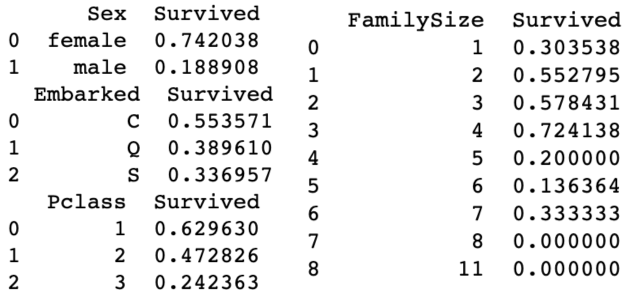
    Q = Queenstown

    S = Southampton

Exploratory data analysis (EDA)

Table analysis

In this stage, tables were used to explore whether discrete/categorical variables have effects on survival rate. The following results and insights were obtained:



```
      Sex  Survived
0  female  0.742038
1    male  0.188908
  Embarked  Survived
0        C  0.553571
1        Q  0.389610
2        S  0.336957
   Pclass  Survived
0       1  0.629630
1       2  0.472826
2       3  0.242363
```

```
   FamilySize  Survived
0           1  0.303538
1           2  0.552795
2           3  0.578431
3           4  0.724138
4           5  0.200000
5           6  0.136364
6           7  0.333333
7           8  0.000000
8          11  0.000000
```

Table1: Survival rate by Sex, Embarked. Pclass, and FamilySize

1. The survival rate for women is much higher than for men. The survival rates for women and man are 0.74 and 0.19.

2. The survival rate for people embarked from Cherbourg is higher than from Queenstown and Southampton. The survival rates for people embarked from Cherbourg, Queenstown and Southampton are 0.55, 0.39. 0.34.

3. The survival rate for people in the first class is higher than for people in the second class; the survival rate for people in the second class is higher than for people in the third class; The survival rate for people in the first, second, and third class are 0.63, 0.47, 0.24.

4. The survival rate for people alone is lower than people that go out in a group. For example, the survival rate for people alone and people go out with 1-3 people are: 0.3, 0.55, 0.58, 0.72.

Chart analysis

In this stage, charts were used to explore whether discrete/categorical variables have effects on survival rate. The following results and insights were obtained:

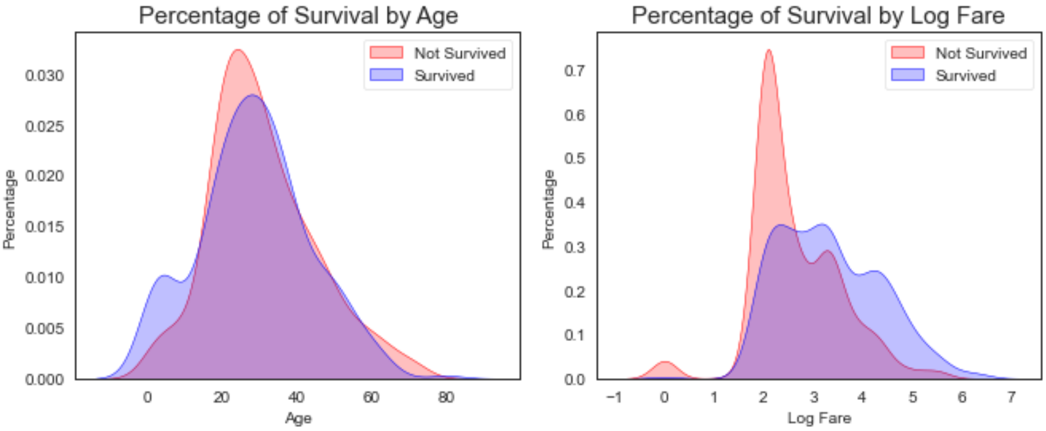


Chart 1: Percentage of Survival by Age and Log Fare

1. The number of survived infants is larger than died ones. The number of survived people in their 20-30s are smaller than died ones.

2. The survival rate for people who bought tickets with high prices is larger than people who bought tickets with low prices.

Data Cleansing

Correct ticket price values

There are two types of tickets: individual tickets and group tickets. In the dataset, some people's ticket prices are abnormally high because the Fare data shows the group ticket price, rather than the individual ticket price. These group ticket prices are replaced with ticket price per person.

Ticket price per person = Group ticket price/Number of individuals that share the same ticket ID

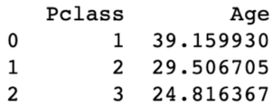
Impute missing values

There are 263, 1, 1014, and 2 missing values in the Age, Fare, Cabin, and Embarked features for the total dataset that contains data of 1309 passengers. Because these features have different datatypes and number of missing values, different methods were used for imputing missing values.

1. Fill the missing values in Embarked and Fare with modes and median

Because there are only 2 and 1 missing values in Embarked and Fare, I just use the fastest method to fill them with modes of embarked and median of fare.

2. Fill the missing values in Age based on Pclass



```
   Pclass        Age
0       1  39.159930
1       2  29.506705
2       3  24.816367
```

Table 2: Average age by Passenger class

Age and Passenger classes are highly correlated. Table 2 shows that the average age for people in the 1st class, 2nd class and 3rd class are 39.16, 29.51, and 24.82. Therefore, the missing age values were filled in by the average age of the passenger's passenger class.

3. Delete the Cabin feature

The Cabin feature were deleted for two reasons. First, there are too many missing values. Second, we cannot detect the Cabin from available data because these cabins are scatted in Titanic and do not highly correlated with other features. (Although some notebooks found cabin related to passenger class through data analysis, it is not reasonable when we see the blueprint of Titanic.)

Feature Engineering

1. Ticket Letter

The alphabets in ticket numbers may contain some information. Therefore, a feature of the alphabets in ticket numbers were created. There are some alphabets only exist in less than 10 tickets, these alphabets were stored to the miscellaneous (Misc) category.

2. FamilySize and IsAlone

FamilySize = the number of SibSp + the number of Parch + 1.
IsAlone = 1 when FamilySize = 1, IsAlone = 0 when FamilySize >1.

3. Title of Name

The titles of passengers' name contain information related to age, sex, and marriage. Therefore, a feature of the titles of passengers' name were created. There are some titles only show less than 10 times, these titles were stored to the miscellaneous (Misc) category.

4. Bins for Age and Fare

Age and Fare are two continuous variables. They were transformed to categories and stored in AgeBin and FareBin features. 10 bins were created for fare, the number of passengers in each bin are equal. 9 bins were created for age, the distances between two bins are equal.

Correlation Analysis

Variables are correlated but also different. Therefore, it would be nice to use these variables to build models. See Chart 2.

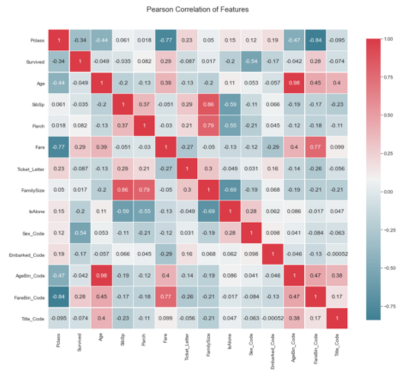


Chart 2: Correlation Analysis for Variables

Modeling

Choose the best models for ensemble

Because this problem is a classification and regression problem, some available models are: Logistic regression, Linear Discriminant Analysis, Quadratic Discriminant Analysis, KNN, Support Vector Machine, Decision Tree, Random Forest, AdaBoost, Bagging, and Gradient Boosting.

Here some models will be selected for ensemble. Therefore, cross validation with default parameters for each model were conducted to make a rough model selection. Chart 3 shows the result.

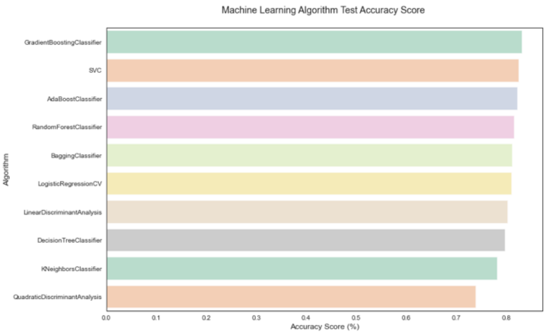


Chart 3: Machine Learning Algorithm Test Accuracy Score

The top 5 score models: Gradient Boosting, Support Vector Mapping Convergence, AdaBoost, Random Forest, and Bagging were selected for ensemble.

Hyper Parameter Tuning

The hyper parameter tuning was conducted to select the best parameters for each model. There are 30% train data were used for testing, 60% train data were used for training, 10% train data were ignored in the 10-fold cross validation. The best scores for the five tuned models are 0.8116, 0.8313, 0.8037, 0.8399, 0.8332.

Ensemble Models by hard and soft voting

Chart 4 shows the five models' predictive results are correlated but also different. Therefore, it would be nice to ensemble them.

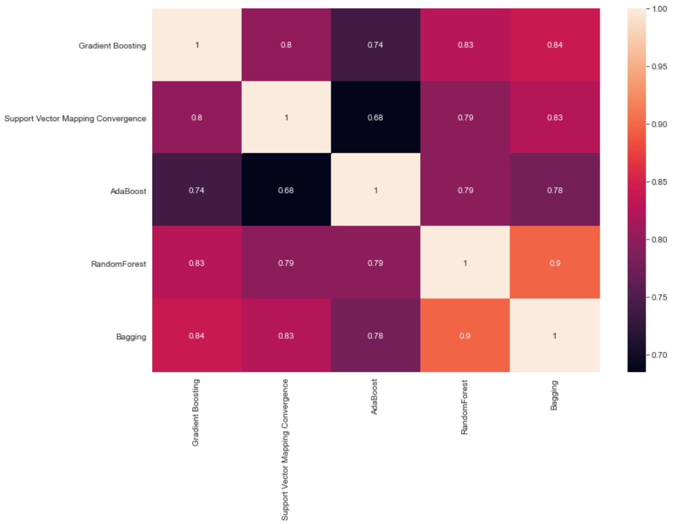


Chart 4: Correlation Analysis for Prediction Results

The 5 models were ensembled. First, the 5 models were ensembled by both hard and soft voting. Then, the accuracy scores were calculated for two models though cross validation. The accuracy score for soft voting is 0.8328, which is higher than the score of hard voting that is 0.8325. Therefore, the soft voting was chosen to predict the test dataset.

Prediction and Submission

The soft voting was used to predict the test dataset. The result was submitted to Kaggle and got a score of 0.76794.

References:

1. Kaggledotcom. (2019, September 30). How to Get Started with Kaggle's Titanic Competition | Kaggle. Retrieved December 15, 2020, from https://www.youtube.com/watch?v=8yZMXCaFshs
2. Alexisbcook. (2020, April 01). Titanic Tutorial. Retrieved December 15, 2020, from https://www.kaggle.com/alexisbcook/titanic-tutorial
3. Ldfreeman3. (2017, December 31). A Data Science Framework: To Achieve 99% Accuracy. Retrieved December 16, 2020, from https://www.kaggle.com/ldfreeman3/a-data-science-framework-to-achieve-99-accuracy/
4. Brendan45774. (2020, December 18). Titanic How I become the top 1%. Retrieved December 16, 2020, from https://www.kaggle.com/brendan45774/titanic-how-i-become-the-top-1
5. Blackhurt. (2020, November 26). My approach to be in top 2%. Retrieved December 16, 2020, from https://www.kaggle.com/blackhurt/my-approach-to-be-in-top-2
6. Startupsci. (2019, February 11). Titanic Data Science Solutions. Retrieved December 16, 2020, from https://www.kaggle.com/startupsci/titanic-data-science-solutions
7. Anonym. (n.d.). Python - python pandas dataframe: Using Conditional Means to Fill in NaNs. Retrieved December 17, 2020, from https://www.coder.work/article/377854
8. Yassineghouzam. (2017, August 09). Titanic Top 4% with ensemble modeling. Retrieved December 18, 2020, from https://www.kaggle.com/yassineghouzam/titanic-top-4-with-ensemble-modeling
9. Gunesevitan. (2020, January 20). Titanic - Advanced Feature Engineering Tutorial. Retrieved December 18, 2020, from https://www.kaggle.com/gunesevitan/titanic-advanced-feature-engineering-tutorial/comments
10. Wf592523813. (2010, January 16). Summary: Using Scikit Learn to Tuning. Retrieved December 18, 2020, from https://blog.csdn.net/wf592523813/article/details/86382037
11. JAIN, A. (2016, February 21). Gradient Boosting: Hyperparameter Tuning Python. Retrieved December 18, 2020, from https://www.analyticsvidhya.com/blog/2016/02/complete-guide-parameter-tuning-gradient-boosting-gbm-python/
12. Ankushkuwar05. (2019, November 25). ML: Voting Classifier using Sklearn. Retrieved December 18, 2020, from https://www.geeksforgeeks.org/ml-voting-classifier-using-sklearn/
13. Anisotropic. (2018, June 16). Introduction to Ensembling/Stacking in Python. Retrieved December 18, 2020, from https://www.kaggle.com/arthurtok/introduction-to-ensembling-stacking-in-python