Titanic: Machine Learning from Disaster

Tianyang Chen
Department of ECE
University of Arizona
tianyangchen@email.arizona.edu

Zili Rong
Department of ECE
University of Arizona
zilirong@email.arizona.edu

Abstract

The sinking of the RMS Titanic is one of the most infamous shipwrecks in history. We cannot do anything to save the victims anymore but we can do something to prevent the tragedy happen again. An efficient way is to analysis what kind of people are more likely to survive in a sudden disaster. Machine learning techniques can do us a favor for its highly developed algorithms, as we discussed in class. The algorithm conducted in this paper is based on feature selection, logistic regression, support vector machine and random forest. The work has been done by the group and include data analysis, model training as well as prediction testing.

keywords: feature selection, random forest, support vector machine, logistic regression, ensemble, disaster prediction

1 Introduction

This project comes from Kaggle Data Science Competition Website. On April 15, 1912, during the maiden voyage of RMS Titanic, she 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 [1].

In this paper, we predict which kind of passengers are more likely to survive the tragedy. Once we build this model with high prediction accuracy, it can be used to estimate about how many lifeboats should be prepared in the ship or etc., since one of the reasons for this shipwreck was not having enough lifeboats. Machine learning techniques offer the possibility of developing reliability of the model by increasing prediction accuracy. Our prediction model is developed among several algorithms, and we choose the one with the most accuracy, which is a combination of feature selection and support vector machine.

2 Related work

2.1 Prediction using machine learning

Among the world, prediction is one efficient and crucial way to explore the future. As Machine Learning developed so mature within these years, it generate more and more possibilities for us to achieve better performance in all field. Prediction based on Machine Learning has become the mainstream of prediction methods. In Computer Architecture, neural networks are been used for dynamic branch prediction [2], which can give better performance as pipelines deepen and the

number of instructions issued per cycle increases. Also in casinos, bandit algorithm is able to find the arm with the most potential profit [3]. So, in our project, based on the specific dataset we have and the high accuracy we pursue, combination of feature selection and support vector machine is the model we propose.

2.2 Dataset

We get passengers' information data from Kaggle database, where they have already split the historical data into two groups, a 'training set' and a 'test set'. Based on the 'training set' we generate a model and use the model to predict who will survive in the 'test set', 891 samples are in 'training set' and 418 samples are in 'testing set'.

We have information on different aspects of passengers, which will serve as features during the training and testing process. list of features and notation of them are shown in the *Table1* below.

Table 1: Features and notation

Variable	Definition	Key
survival	Survival	0 = No, 1 = Yes
pclass	Ticket class	1 = 1st, 2 = 2nd, 3 = 3rd
name	Name	
sex	Sex	
Age	Age in years	
sibsp	# of siblings / spouses aboard	
parch	# of parents / children aboard	
ticket	Ticket number	
fare	Passenger fare	
cabin	Cabin number	
embarked	Port of Embarkation	C = Cherbourg, Q = Queenstown, S = Southampton

We find that not all the features are meaningful and contributing to the training of the prediction model, so we drop several features to make the predictor less affected by the noise. Features in *Table 1* with grey background are the features we don't use, which we choose based on common sense and feature selection algorithm. So the final dataset include features of {*survival, pclass, sex, Age, sibsp, parch, fare, cabin, embarked*}.

To manage the sample data, the first thing we need to convert the form of some data, since some features are in text, and what we can use is only numerical data. Then, effort must be done to eliminate the samples without full information or fill the blanks based other information.

3 Methodology

3.1 Idea

Based on the well-managed sample data, we use different classifiers from *Python's Scikit-learn*, such as *logistic regression, support vector machine, random forests*, and combinations of more than one classifiers trying to make the prediction more accurate. Then, we implement our model on the testing data set to generate the prediction and upload the results to Kaggle website, which will return the accuracy of our prediction.

According to the simulation performance (shown in the simulation section below), we finally find using *Support Vector Machine* to train our model can achieve the highest accuracy and reliability.

3.2 Feature Engineering

We can say that the most crucial part of this project is based on the feature engineering. Vast of data carries plenty of information, but which of them are useful and may have contribution to our training model is the main question. An promising feature engineering will lead to a better prediction performance with required high accuracy. Effort we did on manage every feature is explained below.

pclass: The original data include 3 classes, we need to separate them into 3 columns with each indicating one class. In column 1, we assign 1 if the sample belongs to class 1 and 0 if not. Same cases for the rest 2 columns. We complete this task by using ***pandas.get_dummies***[4], whose function is to convert categorical variable into dummy/indicator variables. We doing so because we cannot use the original format of data in logistic regression.

sex: Same as *pclass*, we define 'female' as class 1 and 'male' as class 2, using ***pandas.get_dummies*** to generate useable dataset.

Age: This feature, unfortunately, only 714 samples have data. In this case, we need some algorithm to fill those blanks. After reading some literature, we found several ways to handle this. For example, We could set those samples will NaN as a new class, or use the mean value to fill those blanks, or try to predict those missing values based on the data we have. In our project, we train a random forest to predict those missing values. Furthermore, based on the movie we suppose children and old people are more likely to be rescued, so we distribute this feature into 8 groups, since the highest age value we find is 80 and each group include 10 age values. For example, group one include samples from age 0 to age 10. Again, we apply the ***pandas.get_dummies*** to manage the data.

fare: For this feature, we normalize the data with zero mean and unit variance using ***sklearn.preprocessing.StandardScaler***[5].

cabin: We simply take this feature by two groups, with or without cabin number.

embarked: Intuitively, this feature seems to do nothing with the survival result. However, after making a histogram about the survived situation and port of embarkation, it seems that passengers boarded on port S are more likely to suffer. Same as *pclass,* we claim 3 embarkation ports, each of them is divided as a individual feature by using ***pandas.get_dummies***, as shown in *figure 1*below..

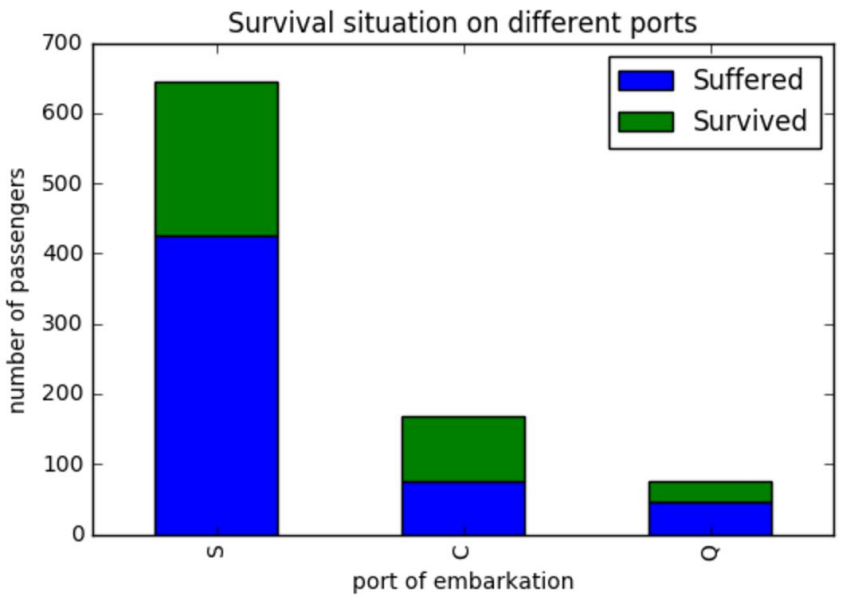

Figure 1: Survival benchmarks on different ports

3.3 Logistic Regression

We first apply the most common algorithm to train the model. However, it is known that logistic regression can only be applied on a 2-classes problem[6], and this is the reason why we did so much effort to convert the format of the original data. Since we are not sure about the importance of each feature, we also add a L2 regularization term.

3.4 Support Vector Machine

We use the *Support Vector Machine* with gaussian kernel, we set the kernel coefficient to be 0.1. If the coefficient becomes too large, it will lead to overfitting; and if the coefficient is too small, the performance will be low.

3.5 Random Forest

The random forest (Breiman, 2001) is an ensemble approach that can also be thought of as a form of nearest neighbor predictor, it yields generalization error rate that compares favorably to Adaboost, yet is more robust to noise[7].

3.6 Ensembles

We combine those three predictors above and ensemble them by voting. The idea is that we predict each sample by three predictors, each predictor give us a predicted class and we take the class with the most votes.

4 Benchmark

4.1 Benchmark Environment

We implement our project based on Python in Linux environment, code is written and debugged under *Jupyter notebook*. Among each benchmark, we use 5-fold cross validation to try to find the classifier's best parameters, after this, we submit the prediction to Kaggle website, which will return us the prediction accuracy.

4.2 Benchmark Result

Table 2: Accuracy for different Model

Algorithm	Accuracy
Logistic Regression	0.76077
SVM	0.79426
Random Forest	0.76555
Ensemble	0.76077

4.4 Data Analysis

Table 3: Weighting coefficient in Logistic Regression

	Coefficient	Columns
0	[-0.39111354654]	SibSp
1	[-0.18856397464]	Parch
2	[-0.331280272636]	Cabin_No
3	[0.547574503395]	Cabin_Yes
4	[0.215560222848]	Embarked_C
5	[0.112566728063]	Embarked_Q
6	[-0.262619000736]	Embarked_S
7	[1.4418443793]	Sex_female
8	[-1.22555014854]	Sex_male
9	[0.55372129822]	Pclass_1
10	[0.376234175102]	Pclass_2
11	[-0.713661242562]	Pclass_3
12	[0.141258093372]	Fare_scaled
13	[1.63913837135]	Age_10

14	[0.298348804017]	Age_20
15	[0.0787148055601]	Age_30
16	[0.117037886102]	Age_40
17	[-0.265593795685]	Age_50
18	[-0.585170908362]	Age_60
19	[-0.741167727693]	Age_70
20	[-0.325013204529]	Age_80

Let's take a look at the weighting coefficient in Logistic Regression. The coefficients of "Sex_femal" and "Age_10" are really large, which means they are highly positive correlated with the survival result. And this result seems consistent with the movie we watched, the captain asked women and children first. Another thing we found that the coefficient of "Pclass_1" is higher than the other two classes, it seems even in such disaster, money still talks.

According to the four models we implemented, the accuracy of Random Forest and Logistic Regression are pretty close, and SVM is better than the others. And we combine those three classifiers as ensembles, but the accuracy doesn't improve much. Based that we only extracted limited number of features, some features might have some connections but we can not get intuitively. Since SVM can map our feature space to infinite dimensions, it's reasonable it works better.

5 Conclusion and Further Improvement

In this paper, our main aim is to find what kind of people are more likely to survive in disasters. First, we pay a lot attention on the dataset itself, we filter the unwanted features and manage the useful features to make them meaningful and usable to us. Then, we illustrate several prediction methods in machine learning, each has different advantages and disadvantages when apply to the Titanic problem. Furthermore, we compare those predictors with respect to prediction accuracy, and it comes out that *Support Vector Machine* has the best performance. To answer the original question, we find out that children and females as well as people with higher status and wealth are more likely to survive in disasters.

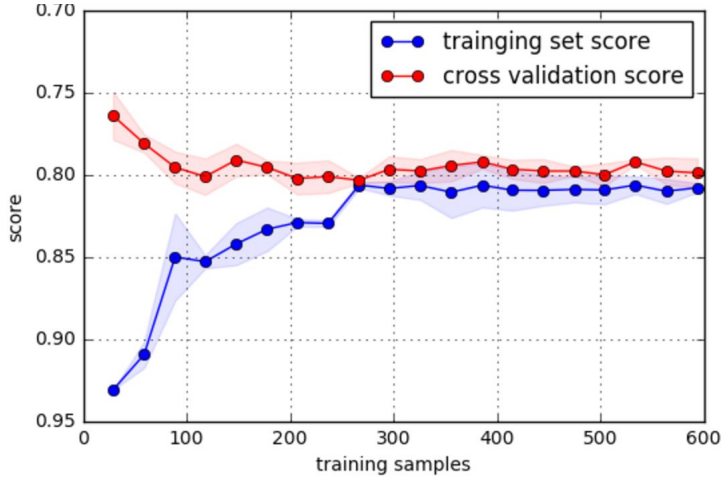

Figure 2: Learning curve about Logistic Regression

Our best performance now is 0.79426 with *Support Vector Machine*, we are ranked about 1500 on the leaderboard. There are still a lot of ways we can try to improve our accuracy.

1. We can use different way of ensembles, instead of choosing the most majority vote, weighted vote may give a better result.
2. Looking at the learning curve, the gap between training score and cross validation score is still small, which means we are not in a overfitting status, we can extract more information from the features we dropped, for example, the *Name* feature can either group the family by examining the family name or tell the status by cluster titles.
3. Since we clustered the age feature by 10, maybe we should not try to predict the missing age, instead we should set the missing age by mean of their title such as "Miss", "Mrs", etc.
4. We could engineer deeply into the Cabin feature. We guess the first letter denotes the floor in the ship, the subsequent number denotes room number. There's an interesting thing that passenger with larger number are more likely to survive.
5. We should add a new feature named "Mother", who has the name with title "Mrs", and the number of "*Parch*" greater than 1.

References

[1] (n.d.). Retrieved May 03, 2017, from https://www.kaggle.com/c/titanic

[2] Daniel A. Jimenez, *Dynamic Branch Prediction with Perceptrons*, Department of Computer Sciences, The University of Texas at Austin, Austin, TX 78712

[3] Katehakis, M. N., & Veinott, A. F. (1987). *The Multi-Armed Bandit Problem: Decomposition and Computation*. Mathematics of Operations Research, 12(2), 262-268. doi:10.1287/moor.12.2.262

[4] *Pandas.get_dummies*. Retrieved from http://pandas.pydata.org/pandas.get_dummies.html

[5]*Sklearn.preprocessing.scale* Retrieved from http://scikit-learn.org/sklearn.preprocessing.scale.html

[6] *Introduction to the Logistic Regression Model. (2005)*. Applied Logistic Regression, 1-30. doi:10.1002/0471722146.ch1

[7] Su, C., Ju, S., Liu, Y., & Yu, Z. (2015). *Improving Random Forest and Rotation Forest for highly imbalanced datasets. Intelligent Data Analysis*, 19(6), 1409-1432. doi:10.3233/ida-150789