MATH 4432 Final Project: Analysis and prediction on survival on the Titanic

Yanbang Wang[1] and Zizheng Lin[2]

[1]Department of Computer Science and Engineering, HKUST
[2]Department of Mathematics, HKUST

1. Introduction

On April 15, 1912, the Titanic sank after colliding with an iceberg, and 1502 out of 2224 passengers on board died.[] This report reflects on our journey of doing the Kaggle Competition on Titanic survival prediction. Based on the public survival dataset on Kaggle, we conducted analysis to figure out the potential factors that affect the survival rate of passengers, and adopted different machine learning models to make prediction on each specific person's survival. Empirical studies led us to the finding that Kaggle competitions highly stress on the importance of data preprocessing and hyper-parameter tuning(This makes sense because those modeling algorithms are publicly available to all). Our report would therefore pay special attention to introducing our effort in this realm. We develop our own, original kernel on the numerous difficulties and uncertainties of the competition, despite the numerous kernels that are publicly shared and copied on the platform.

2. Titanic Dataset

The dataset contains 891 and 418 observations in training set and test set respectively. The available 10 features include the social status of a person('PClass'), a person's name, gender, age, ticket fee, and so on. It comes to our realization at the first sight that there are multiple issues with this dataset that needs to be addressed, including but not limited to frequent missing values and badly formatted strings like names and ticket numbers, useless/noisy information like Ticket, and potentially biased distribution of test set. We approached all these problems by analyzing the features one by one, which turned out to be crucial to our final result.

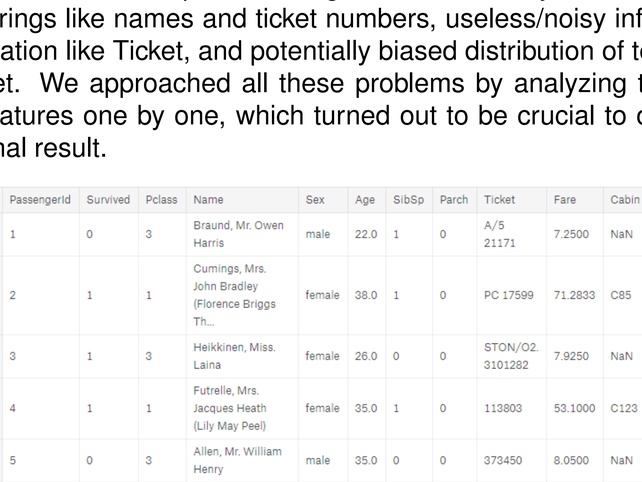
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

Figure 1: *A snapshot on original dataset*

3. Preprocessing and Feature Engineering

3.1 Verify Identical Train-Test Distribution

The first to care about is whether the training set and test set approximately follows the same distribution (since this is what we base most our model's assumption on). To do this, we pick up several features and draw their distributions in training and test set respectively.



Remember that "training set size" : "test set size" is 68%:32%, so those in the test set should have a count that is slightly smaller than half of the corresponding count in the test set, which proves to be the case. Can a conclusion be reached that the training set and test set are identically distributed? We might have a better confidence towards a "yes" answer, but we reserve our opinion before further analysis.
We then look at the features one by one, analyze them with necessary visualization, and curate them if necessary.

3.2 Analyze the Features

3.2.1 Feature Analysis: Survival

This is the dependent variable and our predicted target. Since it is already binary valued, and contains no missing values, we simply visualize it and plan to leave it untouched.

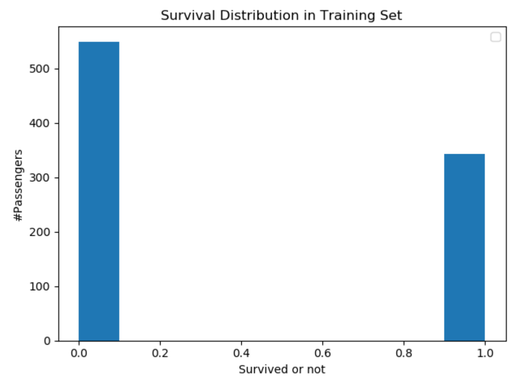


Figure 3: *Training set's positive observations follows a biased distribution*

However, it turned out that this visualization is quite important: according to the the official introduction of this competition, (2224-1502)/2224=32.5% of passengers finally survived the accident, whereas it becomes evident here in training set, that the survival rate is high above half of the death rate. This leads to our final decision that the training set is actually slightly biased, thus the variance of our modeling needs to be carefully restricted (i.e. use simpler model, reduce the fold number of cross-validation etc.). It also helps to answer why later when we the cross-validated error on our training set is always around 5% lower than that on the test set.

3.2.2 Feature Analysis: Pclass

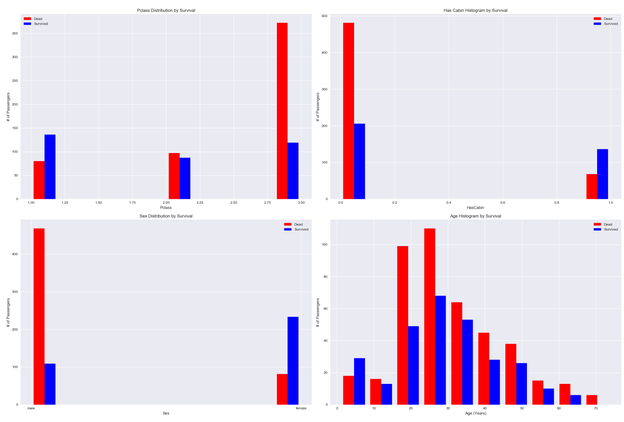


Figure 4: *Use histogram to reveal the correlation between features and prediction target*

This is a numerical feature with no missing values, and according to figure4's first subplot, it became quite evident that those in the first and second class seat possess a much higher survival rate than the third class, which makes a lot of sense in real life. Therefore, we readily leave it untouched at this step.

3.2.3 Feature analysis: Name

This is a very mixed-formatted feature, recording both the title and the name of the passengers. While it might look useless at the first sight, we can actually utilize people's titles (i.e. Mr, Mrs, etc.) . Note that apart from those titles that we are familiar with, multiple less frequently used titles, like "Mile" and "Don", also appear. Work is done to transform/unify them into the several titles that we frequently use. This "title" feature turns out to be quite useful as it reflects a person's age, gender, and marital/family status(some of which are missing in our dataset).

3.2.4 Feature analysis: Sex

The bottom-left plot reveals with great clarity that sex is a highly determinant feature in our whole analysis. The original data was in male-female form, so we simply binarize it into 0-1's (in face, using this feature alone helps achieve 66%).

3.2.5 Feature analysis: Age

While age could potentially be a highly illuminating feature (see bottom-right subplot of figure 4), it suffers great portion of missing values (nearly 45%). An somewhat "irresponsible" approach could be either discarding it or filling it with some straightforward statistics like mean or median. Here, instead, we decide after careful analysis to complete this feature by taking it as a prediction target first, and then utilizing the "Pclass" and "title" of those data that contains age information to predict the missing ages. (It actually still remains arguable since one generally hopes the features that they use to be independent each other, because otherwise a secondary feature derived from others would only add to computation burden). Empirical study has proved this feature very useful, as opposed to either filling it casually or discretizing it into value intervals(aka. making bins)

3.2.6 Feature analysis: SibSp & Parch

SibSp describes the sibling/spouse number of a specific passenger, and Parch describes the parent/children number. Since both variables focus on counting the relatives, we add them together to create a new feature "family", and drop the originals. This makes sense because either alone could not give us a complete picture of a passenger's family members that we ultimately care about.

3.2.7 Feature analysis: Ticket

This is a rather noisy feature, with terribly formatted strings and no description/interpretation of the ticket numbers. Therefore, we simply choose to discard it

3.2.8 Feature analysis: Fare

This feature represents the ticket price of each passenger. We first notice that there is one missing value in this feature. Using the same technique as that of "Age" feature, we fill in the single empty slot(this actually does not make much difference with respect to the hundreds of data, but we do it simply for perfection). We have plotted quite a few schemes to deal with this feature: 1)leave it untouched, since it is already numeric, and has very concrete meaning; 2)Log transform it.Reflecting on what could decide a person's destiny in face of the Titanic disaster, what we ultimately want to get from this feature is the passenger's social-economic status, and location on board at the time of the disaster. Does a passenger holding a 150-pound ticket reflect 20 times more important than a passenger with 7.5-pound ticket, in terms of his socio-economic status? Probably not. Log transformation, therefore, is trying to address this discrepancy by shrinking the distance among people. 3) categorize this feature into different price intervals, either using percentile cut, or value cut, based on the same rationale as 2), but with loss of the original data information.
Empirical studies have shown that 1 and 2 outperforms the others in different algorithms.

3.2.9 Feature analysis: Cabin

This is a string-type feature, marking people's cabin index, if they have any. There are different types of cabins as well. Several possible approaches includes: 1)Discard this features, because it could be questioned, that the majority values of this features are missed, and we simply could not decide whether those are really missed values (there is no "no-cabin" value in this feature, but either an index, or empty) 2)categorize the cabins into their own types, and interpreting all missing values as "no-cabin" 3) only record whether one has cabin or not, and discard the original.
Empirical studies have also shown that 1 and 2 outperforms the others in different algorithms.

3.3 Most Important:Standardization

Obviously, those features are not on the same scale. Standardization became highly crucial at this point, since a highly skewed dataset would not only lead to the slowness of gradient decent convergence, but also result in the invalidity of many distance-related algorithms, including KNN and SVM. In face, standardization along helps boost our cross-validation accuracy by around 5% from 78% to 83%.

3.4 A little Wrap-up: Tedious but Important Work

Both me and my teammate finally (after numerous disappointment) realize that the data preprocessing step are of great significance. An ultimate goal of the work here is to remove the noise as best as we can, and explore the information hidden in those values that could hardly be ”learned” by the current algorithms (like the titles in names). A very strongly but adverse fact in this specific task, is that to many features are mutually dependent, and thus not leading to much information gain – they ultimately comes down to depicting a person's 1) socio-economic status, which might earn him privilege to survive, 2) sex and age, which decides whether one could board a life boat first (woman and child are privileged) 3) relative position on board, which decides whether one could get quick access to the life boat.
Even if all these 3 factors are accurately determined, one still could not reach a model anywhere close to perfect, because there is so many random factors involved at the time of the disaster that could determine one's life and death (which by no means can be reflected on this dateset)– that's why the best HONEST score could only achieve at most 83% accuracy (there are people achieving 88% using genetic programming, which I have not learned so I have no comment. but it is true that anyone using the traditional ML algorithms could only go that far without external data)

4. Modeling & Prediction

The modeling part is quite standard. An important experience that we come to realize is about Occam's Razor: it is evident from the foregoing analysis that the dataset does not provide too much insight about one's specific situation at the time of the accident. The only three factors (maybe there could be one or two more, but are quite limited) have been very clearly identified. Therefore, one should not expect a very complicated model to provide too much help. Lets see what it turns out:

4.1 Model Introduction

	model	param	scores
5	Random Forest	{'max_features': 15, 'min_samples_leaf': 4, 'n...	0.851852
6	XGBoost	{'learning_rate': 0.105}	0.847363
3	SVM (linear)	{'C': 7}	0.836139
0	Ridge Regression	{'alpha': 0.05}	0.835017
2	SVM (non-linear)	{'C': 0.05, 'degree': 5, 'coef0': 2}	0.833895
4	KNN	{'weights': 'uniform', 'n_neighbors': 10}	0.832772
1	Logistic Regression	{'penalty': 'l2', 'C': 30}	0.831650

Figure 5: *Accuracies obtained on training set using 5-fold cross-validation*

In this task, we have attempted quite a few models, varying from simplest one, like KNN, ridge regression, and logistic regression, to finely structured one, like SVM(with different kernels), to ensemble methods, like random forest and gradient boosting (actually XGBOOST).
Hyper-parameter tuning are done using gridsearch. For flexible models like XGBoost, this are done step by step from n_estimators, to tree-related parameters, regularization parameters, and finally learning rate.
”Accuracy” is chosen as the metric, instead of ROC_AUC, since the competition evaluates our result based on accuracy. 5-fold cross-validation is chosen, in consideration of the fact that training set is slightly biased (as mentioned below). In fact, it turns out that this training set is ”positively biased”: as we increase the fold number from 2 to 10, the accuracy first decreases (the bias gets reduced), and then steadily increased.

4.2 Result Analysis

Ensemble methods still appears to outperform others on our training set, with a win of around 1-2 % compared with others. However, this is not the case when we submit the result to the leatherboard. In fact, three submissions that have achieved highest generalized accuracy are ridge regression, SVM with linear kernel, and KNN, while ensemble methods encounter a sharp drop on unseen dataset.
From another perspective, the results achieved, despite small discrepancies, are quite similar, and seem be bounded. Indeed, looking at the leatherboard, we realized there were 4000+ people crowding in 78%-83% interval. Given the randomness inherited in the algorithms, we identify that there were not much difference among these top teams. Our best submission achieves 79.4% accuracy on public test set, and ranked 25% at the time of our submission.

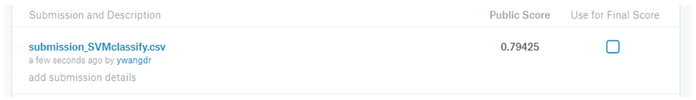


Figure 6: *Best Kaggle Rank*

5. Conclusion

In this final project, we attempted the Titanic Survival Prediction Challenge. We have done a lot of data curation work, and have thought very carefully about difference scenarios and strategies. After tuning the hyperparameters, we finally came to the result of top 25% of the competition, which was satisfactory, in consideration that we didn't really exhaust all the possibilities of deal with the data. We are hoping to make further improvement of our result with genetic programming in the future.