M6 Project

Kaggle: Titanic - Machine Learning from Disaster

Johnathan Wang, Nicholas Charles Vitellaro, Makenna Shae Owens, and Marc Mahanna

COSC523: Artificial Intelligence, November 10, 2023

***Abstract* - This project focuses on use machine learning approaches to implement models predicting which passengers survive the Titanic shipwreck. The format of this project is that of a Kaggle competition** [1]**. Goals of this project are to 1) implement a baseline score and 2) try to improve on the baseline score.**

DESIGN CHOICES AND IMPLEMENTATION

The approach selected was to implement a baseline and an improved model intended to improve score beyond the baseline. Random Forrest Classifier is the default implementation and is provided in the Kaggle competition tutorial [2]. Gradient Boosting Classifier is improved model providing the best performance after much team experimentation. For each experiment and subsequent baseline and improved implementation, data was loaded, data was cleaned, features were explored and selected, models were trained, and test results were produced.

The baseline model followed the Kaggle tutorial. Data cleaning and preparation was minimal as the selected features 'Pclass', 'Sex', 'SibSp', and 'Parch' do not have missing values. Feature exploration in the baseline model produced interesting results of 74% of women surviving and 19% of men. The Random Forrest Classifier model was initialized per the tutorial with n_estimators=100, max_depth=5, random state=1. Resulting performance when submitted to Kaggle was 77.511%.

The improved model was the result of much team experimentation. The improved model uses features that have missing values, wide ranges (Age, Fare), and mixed types (Cabin). Data cleaning and preparation included filling missing values with reasonable values derived from that feature (Age, Embarked, Fare) informed by other features (Cabin). A "FamilySize" feature was created by combining "SibSp" and "Parch". The Cabin feature was engineered to capture the Level or floor of the passenger's cabin as "CabinLtr". N/A's were filled by matching the distribution of CabinLtr versus Pclass. Feature exploration includes correlation matrix (Survived vs each feature, also each feature vs each feature), and evaluation of inclusion / exclusion of each feature. Both default initialization and results of GridSearchCV parametric searches were used to explore more favorable hyperparameters. Multiple models were explored including extending/improving Random Forrest Classifier, Gaussian Nieve Bayes, and Gradient Boosting Classifier. After experimentation, Gradient Boosting Classifier with default hyperparameter values was used. Resulting performance when submitted to Kaggle was 79.186%.

CHALLENGES AND OBSTACLES

The project group members are familiar with machine learning concepts and have prior experience with all three selected models. Use of standard practices of data exploration, cleaning, and normalization were employed making exploration of model performance achievable.

The provided train data set has a lot of noise that most hyperparameter tuning chased causing models to overfit. Relaxing models back to default hyperparameter settings provided greater overall performance avoiding overfitting.

The provided train data set is not sizable resulting in underrepresentation of categories. Model CV or GridSearchCV best score was found to not be a good indicator of model performance on test dataset; a model with CV performance of >80% would score less than 77% on Kaggle, only submission to Kaggle provided overall performance ground truth.

DISCUSSION AND FUTURE WORK

The team believes that the features used in the default implementation are sufficiently explored and by themselves, cannot be improved upon. The team did find through feature exploration that the Cabin feature holds merit. The improved implementation included feature engineering of a "CabinLtr" feature indicating the level or floor of the passenger's cabin. There is a pattern as show in in Figure 1 (Pclass, CabinLtr, Survived) that certain floors had association with Pclass allowing backfilling missing values. This feature is implemented in the improved model and resulted in increased performance of about 2% over the baseline model.

1-A-0	8
1-A-1	7
1-B-0	12
1-B-1	35
1-C-0	24
1-C-1	35
1-D-0	7
1-D-1	22
1-E-0	7
1-E-1	18
1-T-0	1
2-D-0	1
2-D-1	3
2-E-0	1
2-E-1	3
2-F-0	1
2-F-1	7
3-E-1	3
3-F-0	4
3-F-1	1
3-G-0	2
3-G-1	2

Figure 1 - CabinLtr Feature

Additionally, the team started development of a CabinRmNum feature that provides greater promise that the CabinLtr feature as shown in Figure 2. Room numbers (regardless of CabinLtr) have greater survivability below room number 50. Couple this survivability distribution with Pclass and backfill of N/A's for the Cabin feature may be done in a reasonable manner that better follows the dataset. The team believes the CabinRmNum shows greater promise when combined CabinLtr and proposes this as follow on.

	Room Number Bins						
	1	***25***	***50***	***75***	***100***	***125***	***150***
Survive	42	36	17	18	11	3	0
Not Survive	17	18	11	11	7	1	0

Figure 2 - CabinRmNum Feature

SUMMARY

This project provided the opportunity to compare the performance of different machine learning models. A consumable data set and a challenge issued in the form of a competition provided an enriching and rewarding experience for the project group.

REFERENCES

[1] "Titanic - Machine Learning from Disaster," [Online]. Available: https://www.kaggle.com/competitions/titanic/overview.

[2] "Titanic Tutorial," [Online]. Available: https://www.kaggle.com/code/alexisbcook/titanic-tutorial.