Machine Learning Project

Andrew Aziz, Aftab Khan

December 2023

1 Overview of the problem:

The sinking of the Titanic is one of the most infamous shipwrecks in history. On April 15, 1912, during her maiden voyage, the widely considered "unsinkable" RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren't enough lifeboats for everyone onboard, resulting in the death of 1502 out of 2224 passengers and crew. While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others.

In this challenge, we are asked to build a predictive model that answers the question: "What sorts of people were more likely to survive?" using passenger data (i.e., name, age, gender, socio-economic class, etc.).

In this competition, we have access to two similar datasets that include passenger information like name, age, gender, socio-economic class, etc. One dataset is titled `train.csv` and the other is titled `test.csv`.

`Train.csv` contains the details of a subset of the passengers on board (891 to be exact) and, importantly, reveals whether they survived or not.

The `test.csv` dataset contains similar information but does not reveal whether each passenger survived or not. Our job is to predict these outcomes.

Using the patterns in the `train.csv` data, we have to predict whether the other 418 passengers on board (found in `test.csv`) survived.

We were so motivated to solve this problem because the sinking of the Titanic was a tragic event that resulted in the loss of many lives. Developing a predictive model to determine the factors that influenced survival can contribute to a better understanding of historical events and, in a broader sense, help prevent similar disasters in the future. Moreover, the skills developed in solving this problem can have practical applications in various industries. Predictive modeling is widely used in fields such as finance, healthcare, marketing, and more. The Titanic dataset serves as a hands-on example to practice and apply these skills in a real-world context.

2 Approach for the problem:

In this problem, we have to predict whether the people in Titanic survived or not. Therefore, it is clear that it is a binary classification problem. We find the random forest classifier will be the best approach to solve this problem in an efficient way. When we compared the results of the logistic regression model and the random forest model, we found that the random forest model made better predictions than the logistic regression model. The prediction score for the logistic regression model was about 0.77. However, for the random forest model, it was about 0.784. Furthermore, according to Fernandez-Delgado et al. (2014), the random forest model was among the best machine learning techniques in terms of their accuracy.

We first start by loading the training data and the testing data using the pandas library. Then, we start our data preprocessing, where we start to look carefully at our data set to see whether there are missing values in our training data set and testing data set or not. We count the number of missing values in each column to see how we can deal with these missing values. We find that the best way to deal with these missing values is to use Forward fill Imputation. According to Shadbahr et al. (2023), imputation is the best way to deal with the missing values in the dataset. Forward fill imputation is a method used to handle missing data by replacing the missing value with the most recent known (non-missing) value that occurs before it in the same column. We found this imputation provides better results than using other imputations such as mean or median. Moreover, we can use this imputation for both categorical data and numeric data.

Then, we start to look carefully at the data to see whether we will take all the features in the training data set or not. We decided not to take Name, PassengerId, Ticket, and Cabin because these features differ from one passenger to another, and they do not have a significant impact on predicting whether a passenger will survive or not. Furthermore, the Cabin feature includes a lot of missing values in it. Thus, we found that the prediction will be more accurate if we do not take these features into consideration. However, there is a problem with the other features, which is that they include categorical variables, and machine learning models typically require numerical input. Thus, we convert categorical variables into a one-hot encoded format by using **pd.get_dummies** function. One-hot encoding is a process of representing categorical variables as binary vectors (0s and 1s) to make them suitable for machine learning algorithms.

When we plotted our output, which is the people who survived in the training data set, We found that there was a bias towards the people who died. Therefore, we decided to use one of the augmentation techniques to reduce this bias, which is Smote. According to Maharana et al. (2022), augmentation is one of the best techniques that can help decrease the dependency on training data and improve the performance of the machine learning model. SMOTE, which stands for Synthetic Minority Over-sampling Technique, is an augmentation technique commonly used to address the class imbalance problem. Class imbalance occurs

when one class in a classification problem (which are the people who survived in our data set) has significantly fewer samples than another, leading to biased models that might perform poorly on the minority class.
The goal of SMOTE is to generate synthetic examples of the minority class to balance the class distribution. SMOTE works by creating synthetic samples in feature space, effectively "oversampling" the minority class to make it more comparable in size to the majority class.
We did not use the undersampling augmentation technique because undersampling involves removing instances from the majority class, which resulted in information loss. This reduction in the majority class led to the model not learning the complete representation of the majority class, which impacted its performance negatively. In addition, oversampling can contribute to better generalization of the model, especially when the minority class is under-represented. By introducing synthetic instances, the model may learn a more robust decision boundary that generalizes well to unseen data.
We also used a pipeline because it allowed us to encapsulate and organize multiple steps of our machine learning process, including data preprocessing (such as SMOTE) and model training (Random Forest), into a single object. This makes it easier to manage and understand the workflow.
Finally, we use cross-validation to assess how well a model generalizes to an independent dataset and to estimate the performance of our random forest tree model on an unseen dataset. Cross-validation is particularly useful here in this problem because we have a limited amount of data and want to make the most out of it for training and evaluating your model. Moreover, we chose our hyperparameters based on the best performance observed during cross-validation.

3 Results and Conclusions:

The random forest model achieved a higher prediction score (approximately 0.784) compared to the logistic regression model (0.77).
Forward Fill Imputation was found to be more effective than mean or median imputations for handling missing values.
Features like Name, PassengerId, Ticket, and Cabin were excluded due to their limited impact on survival prediction.
The use of SMOTE for addressing class imbalance contributed to a more balanced and robust model.
The pipeline organization facilitated a systematic and understandable workflow for data preprocessing and model training.
Cross-validation provided a reliable estimate of the model's generalization performance on unseen data.

4 Project Impact:

The developed predictive model contributes to understanding the factors influencing survival on the Titanic.
The choice of a random forest classifier and the use of SMOTE for addressing class imbalance can be valuable insights for similar predictive modeling tasks.
The project demonstrates practical applications of machine learning skills in real-world contexts, emphasizing their relevance in various industries.

5 Acknowledgments:

We got help from Mickeal, TA in COMP 562, about using imputation and augmentation techniques to improve our prediction.

6 References:

Fernandez-Delgado, M., Cernadas, E., Barro, S., & Amorim, D. (2014). Do we need hundreds of classifiers to solve real-world classification problems? *The Journal of Machine Learning Research*, 15(1), 3133–3181.

Kaggle. (2023). *Titanic: Machine Learning from Disaster*. https://www.kaggle.com/competitions/titanic

Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings*, 3(1), 91-99.

Shadbahr, T., Roberts, M., Stanczuk, J., Gilbey, J., Teare, P., Dittmer, S., ... & Schönlieb, C. B. (2023). The impact of imputation quality on machine learning classifiers for datasets with missing values. *Communications Medicine*, 3(1), 139.