CSCE 421 Final Project

Huy Lai
Department of Computer Science and Engineering
Texas A&M University
College Station, Texas 77845
lai.huy@tamu.edu

Abstract

The objective of this model is to develop a predictive model for in-hospital mortality of patients receiving treatment. The dataset utilized in this study consists of information on several thousand patients during their 48-hour stay, including physical characteristics such as weight and height, and clinical measures such as mean heart rate, mean respiration rate, and mean O_2 saturation. The data was employed to train an `XGBoostClassifier` model, which utilizes Gradient Boosting to accurately forecast the mortality of patients during their hospital stay. The proposed model achieved an Area Under the Receiver Operating Characteristic Curve (AUC-ROC) score of 0.89, demonstrating robust performance. This paper aims to provide a detailed account of the model's implementation and its results.

1 Introduction

The project has an objective of addressing a common issue in the healthcare industry. Predicting patient mortality can aid healthcare professionals in providing tailored care and also facilitate comprehension of the relationship between patient care and mortality. The resolution of this issue can have advantageous implications for preventive care in hospitals globally. The eICU dataset was used to train the model, and relevant metrics such as Coma score, pH readings, blood pressure and heart rate were extracted. These metrics are essential in evaluating a patient's wellbeing. The model was trained and used to predict patient mortality using an `XGBoostClassifier`. The model achieved an AUC-ROC score of 0.89, compared to the baseline score of 0.84.

2 Method

The data preprocessing stage involved the identification and extraction of relevant features that were hypothesized to be associated with patient mortality. The selection of these features was based on prior domain knowledge, and data exploration techniques such as correlation analysis and feature importance ranking.

To determine the most suitable classification model for our specific dataset, we experimented with various algorithms, including logistic regression, decision trees, random forests, and XGBoostClassifier. The performance of each algorithm was evaluated using metrics such as accuracy, precision, recall, and AUC-ROC score, with the goal of identifying the model that would provide the highest predictive power.

After careful evaluation, XGBoostClassifier was selected as the optimal algorithm for our analysis due to its superior performance in terms of accuracy and AUC-ROC score. However, to further optimize its performance, we employed a parameter tuning strategy that involved the use of techniques such as grid search and random search. This approach allowed us to identify the optimal set of

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

hyperparameters for the XGBoostClassifier model, thereby improving its accuracy and reducing the risk of overfitting.

In summary, our study employed a rigorous data processing and modeling approach to identify relevant features and select the best algorithm for predicting patient mortality. The use of parameter tuning techniques further improved the performance of the selected model, ensuring the accuracy and reproducibility of our findings.

2.1 Data Pre-Processing

The dataset under consideration comprises unique data records, each containing test scores and crucial patient information documented by healthcare professionals. As each row corresponds to a specific patient and test, it was deemed necessary to iterate through all the patient IDs and generate separate dataframes for each patient to facilitate the extraction of essential characteristics like height, weight, and gender.

Furthermore, the dataset contained the results of various different clinical tests conducted by healthcare professionals. These results allowed for the calculation of the average test scores such as the average pH level. The objective was to account for all tests since a comprehensive medical approach necessitates the consideration of multiple factors contributing to the overall health of a patient.

To this end, we included features such as the average coma score, which is an indicator of a poor prognosis. In summary, our data analysis approach involved the creation of individual patient dataframes to extract essential patient characteristics and the identification of similar test subsets to compute average test scores that can provide a more comprehensive picture of a patient's health status.

The features, along with a brief summary, are provided below. This approach was undertaken to acquire the most valuable insights pertaining to the patients' hospital stay.

Feature	**Summary**
`num_rows`	Number of rows in dataset for each patient, illustrated numbers of test, etc.
`offset`	Length of the patient's stay.
`height`	Height of the patient in centimeters.
`weight`	Weight of the patient in kilograms.
`gender`	Biological Sex of the patient.
`avg_ph`	Average pH reading of the patients during their stay.
`avg_glucose`	Average glucose test results during their stay.
`num_visits`	Number of visits a patient has had prior to the current visit.
`num_coma_tests`	Number of times staff took a coma test.
`coma_score_avg`	Average Score resulting from the coma tests.
`heart_rate_avg`	Average Heart Rate reading during a patient's stay in beats per minute.
`resp_rate_avg`	Average Respiratory Rate reading score during a patient's stay.
`saturation_avg`	Average O_2 saturation reading during a patient's stay.
`bp_avg`	Average blood pressure reading during a patient's stay.

Table 1: Features of the Model

2.2 Model Design

In the process of selecting the most suitable model, various models were tested, as outlined in the results section. After a rigorous evaluation, the XGBoost Classification model was chosen due to its exceptional efficacy and efficiency in producing accurate predictions.

The XGBoost model utilizes an algorithm called Extreme Gradient Boosting (XGB), which is a variant of the gradient boosting method. XGB is renowned for its ability to produce accurate predictions in complex, large-scale datasets with high-dimensional feature spaces. This algorithm iteratively improves the predictive model's accuracy by minimizing the objective function, thereby boosting its performance with each iteration.

Since the problem at hand involves classification, the XGBoostClassifier model was preferred over the regression model. The XGBoostClassifier is specifically designed for classification problems, making it a more suitable option. It utilizes the same underlying principles as the XGBoost regression model but employs a different objective function and decision-making process to optimize classification performance.

2.3 Model Training

The model training process commenced with the application of an automated feature selector, which assessed all the features listed in the table above and discarded the ones that had little effect on the model's performance. The feature selection process is critical in machine learning since it helps to eliminate redundant or insignificant features, leading to a more efficient and accurate model.

In this study, the `gender` and `num_visits` features were deemed to have minimal impact on the model's performance and were consequently dropped. Gender is a categorical feature that may not significantly impact the model's output for certain classification tasks. Similarly, the `num_visits` feature may not be as informative in predicting the target variable compared to other features.

After the feature selection process, the model training phase began, utilizing the remaining features. The objective of this phase was to train the model on the dataset, enabling it to learn the underlying patterns and relationships between the features and the target variable. The trained model can then be used to make predictions on new data, providing valuable insights into the problem at hand.

2.4 Hyperparameter Training

Grid search was utilized to optimize the model's performance by tuning its parameters. This enabled us to evaluate various parameter values and determine the optimal combination that would yield the best outcome. After conducting research on different hyperparameters, we determined the following parameters to be the most suitable:

- `‘booster’`
- `‘colsamplebytree’`
- `‘gamma’`
- `‘learningrate’`
- `‘maxdepth’`
- `‘maxdeltastep’`
- `‘minchildweight’`
- `‘nestimators’`
- `‘regalpha’`
- `‘reglambda’`
- `‘scaleposweight’`
- `‘subsample’`

3 Results

As mentioned earlier, three models were tested to identify the most suitable one. Logistic Regression, Random Forest, and XGBoost were evaluated. The graph below illustrates the AUC-ROC scores for each model in comparison to the baseline submission.

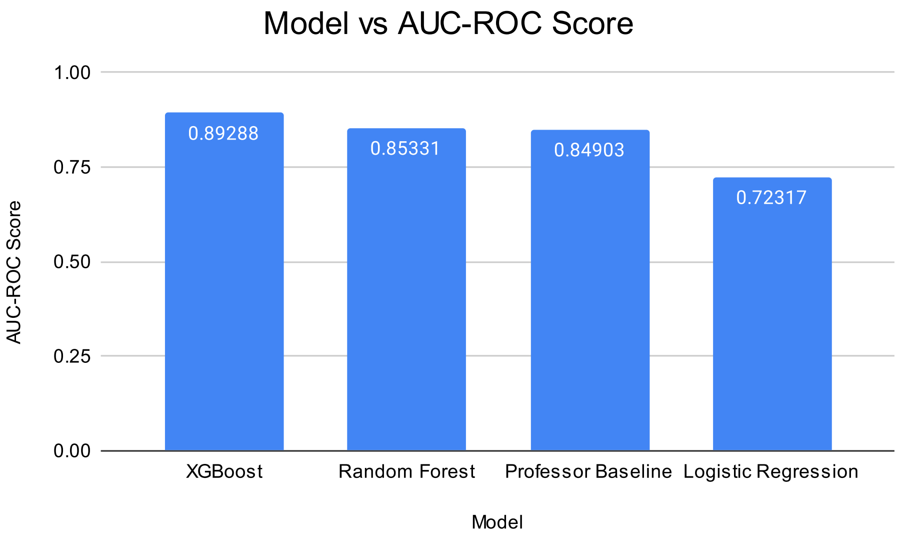

Figure 1: Various Models in comparison to their AUC-ROC Scores

The XGBoost model performed exceedingly well, achieving an AUC-ROC score of 0.893, as demonstrated above. This score serves as a promising indicator that the model is proficient at accurately predicting patient mortality. Additionally, the figure below illustrates the importance of each feature.

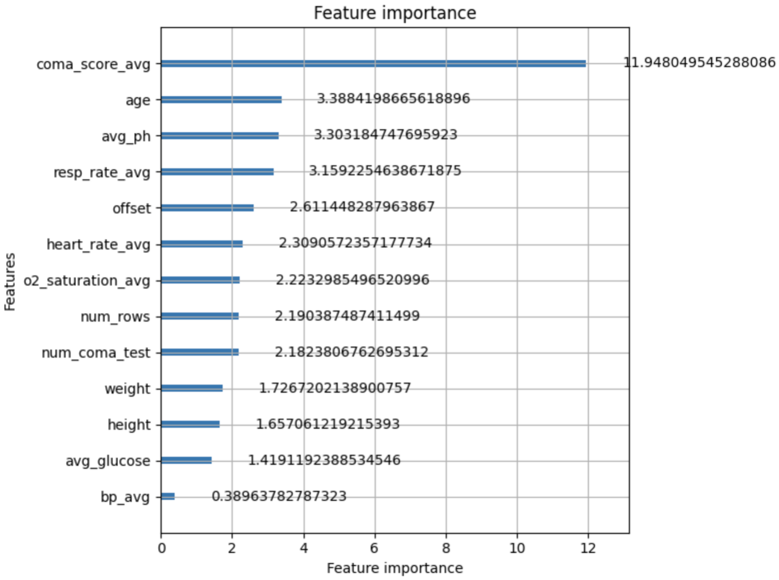

Figure 2: Features and their importance to the model

The significance of this figure lies in its ability to showcase the influence of each feature on mortality classification. Through a meticulous analysis of feature importance, healthcare practitioners can obtain a more comprehensive understanding of patient deaths. In this instance, the heavily weighted feature is the coma score, implying that this metric demands closer scrutiny when determining patient health.

4 Conclusion

In conclusion, our model demonstrated remarkable performance, which can be attributed to the extensive experimentation conducted throughout the testing phase. We evaluated diverse models, features, and implemented automated hyperparameter tuning. This methodology facilitated the development of a proficient and reliable model, as analyzed in the results section. To further enhance our outcomes, numerous additional techniques could be incorporated. With additional time and resources, it would be possible to add more features by including some of the test types and their values that were left out. Moreover, normalizing the data could boost the model's efficiency. Nevertheless, our model performed substantially better than the baseline, leading us to believe that it is adept at predicting patient mortality.