Identification of Suitable Predictive Models for Classifying Terminal Patients

Roberto Cancel, Kyle Esteban Dalope, and Nicholas Lee

University of San Diego

Master of Science, Applied Data Science

ADS-503 Applied Predictive Modeling

Section 01

June 27, 2022

Introduction

One of the most daunting tasks faced in the medical field is the decision of the order in which patients are admitted. The severity of the same illness can vary from person to person; thus, mortality also varies between individuals. Early models that attempted to predict severity and mortality found limited success. For example, Green et al. (1990) the Health Care Financing Administration (HCFA) constructed a model that could not accurately predict patient mortality across multiple hospitals. Since the release of the study by Green et al., with a deeper understanding of machine learning and medicine, it has been shown that machine learning models can accurately define severity and a patient's probability of mortality. One study by Steele et al. (2018) utilized "Cox models, random forests, and elastic net regression" models to predict patient mortality with regards to coronary artery disease. Another study utilized "Support Vector Machine (SVM), Artificial Neural Networks, Random Forest, Decision Tree, Logistic Regression, and K-Nearest Neighbor (KNN)" algorithms to accurately identify 89.98% of COVID-19-related mortality rates (Pourhomayoun & Shakibi, 2021). While the models above could accurately predict mortality and severity, those models were limited to a single illness. In cases where illnesses are diverse, such as in the emergency room or intensive care unit, there is a need for a model to rank severity and predict mortality across multiple illnesses at once. To do so, the Acute Physiology and Chronic Health Assessment Evaluation scoring system (APACHE) is commonly used. This study aims to utilize APACHE scores, along with other descriptive features of patients, to create a model that can accurately identify the risk of death in the presence of multiple illnesses.

Data Description

The dataset provided by Kaggle, "dataset.csv", was initially created as a data frame with the read.csv() function to undergo pre-processing and convert any blank observations with NA. It consisted of 84 predictors and one binary response variable from 91,713 patient observations. Predictor features consisted of patient and hospital identifiers, patient demographic features, patient lab results, patient vitals, APACHE comorbidities, APACHE groupings, and APACHE predictions (see Data Dictionary in Appendix). Due to PHI compliance concerns, the four ID columns were dropped (encounter_id, hospital_id, patient_id, and icu_id). Since this is a classification prediction problem, the response variable (hospital_death) was converted into a factor ("death" and "no_death") and the "gender" variable into a binary factored feature. Next, the table() function was used to identify the proportion of each class in our response variable - the classes suffered from a severe imbalance with only 8.63% of observations resulting in hospital death (0-No death N = 83,781 and 1-Death N = 7,907). This class imbalance is addressed when the data is prepared for modeling.

Data Splitting, Wrangling, and Preprocessing

To avoid data leakage concerns, the dataset was split before any wrangling or preprocessing steps were taken using a stratified random sampling by class methodology with an 80% training and 20% test set. By splitting the data before wrangling, the test set remains "unseen" since all preprocessing techniques use only the information in the training set. When working with large datasets that include missing values, outliers, and correlated features, there are multiple strategies to prepare the data for modeling and analysis. For example, datasets centered around Healthcare, Biomedical, and/or Biology topics often contain many observations and features; thus, data preprocessing is an important factor in the modeling process. Depending

on the data being analyzed and modeled, users have a variety of methods to choose from. This can include data transformations, centering and scaling, Box-Cox transformations, and data reduction and feature extraction such as Principal Component Analysis (PCA).

Upon observing the data set with the summary() function, there were quite a large number of missing (NA) observations (288,046). Schafer (1999) asserted that a missing rate of 5% or less is inconsequential and vice versa; therefore, it was decided to remove any missing data features within each column above a 5% threshold of missingness. This resulted in the removal of 25 features. It was also observed that 11 APACHE features had the same number of missing observations. After subsetting the observations with the missing APACHE features, several of the features included binary diagnoses (e.g., AIDS and cirrhosis). Since imputing diagnoses can introduce severe bias concerns and these observations represented only 0.76% of all observations, the observations with missing diagnoses were dropped. Then, observations with missing ethnicity values were re-coded to the already available “Other/Unknown" value. Finally, the remaining missingness in the data set was imputed by the median of the training set since missingness was now constrained to vital sign data. Near-zero variance features and highly correlated features with a cutoff of 80% were then dropped before conducting Exploratory Data Analysis (EDA).

Exploratory Data Analysis (EDA)

Given the objective, the target feature is hospital_death, a binary feature where zero represents a patient who has not died, and one represents a patient who will pass or has passed. The upper bar chart of Figure 1 illustrates the proportion of instances by the target feature. In addition, the plot shows how imbalanced the data set is, where instances associated with death make up less than 10% of the data set. The subsequent plots help promote insight into the

relationship between mortality rate and patient characteristics by plotting the target features against some predictors. For example, the bottom bar chart in Figure 1 shows the total proportion of death in each ICU Stay Type category. While the proportion of death may not be different by category, this plot indicates that there is not much of a difference in mortality risk whether the patient is newly admitted to the ICU, released and readmitted, or transferred to the ICU from another department.

As mentioned previously, the Apache scoring system measures the risk for mortality in the ICU. The Apache system has several categories in that patients receive a score. The scores from each category are summed into a cumulative score, and the overall risk is determined from the sum. In the data set, different versions of the Apache system were taken into consideration, as the system by hospitals globally varies. Apache II versus Apache III differ in that Apache III has been updated; and, according to Hsu et al. (2001), Apache III has slightly more discriminative power, performing better in terms of its ability to predict high-risk patients than Apache II does. In Figure 2, the bar charts show the proportion of death and non-death patients for each category of the two Apache systems. From these bar charts, it can be deduced that patients that receive a score for sepsis with the Apache III system have a higher risk for mortality. Similarly, with the Apache II system, the three categories with the highest proportion of mortality are respiratory, hematologic, and cardiovascular. A plot, such as Fig. 2, can assist in determining whether a patient is at high risk for death if they receive a score for a certain category.

Since physical characteristics play a role in health-related risks, it is important to examine the relationship between the features related to physical traits and the target variable. For example, age can significantly affect recovery time, illness severity, and general health quality. Figure 3 shows the histogram for age in the data set, colored by the response variable for each

bin and a boxplot. Both the histogram and boxplot show that most of the data lie around the age range of 50 to 75. The data is skewed left; thus, the results from this study may be slightly more applicable to those in that age range, as those outside the age range are not as highly represented. Although from a logical perspective, this distribution of data makes sense; it is not commonly expected that those in a younger age range would be hospitalized. The boxplot reveals four outliers on the lower end of the distribution, where the histogram shows that the data is skewed left. While uncommon, there may be instances where a patient falls in the same age range as the outlier points. Therefore, the outliers were not removed from the data set.

Similar to age, height and weight are physical traits used to measure a patient's general health. While there are some exceptions based on muscle mass and other factors, in general, an individual's weight can indicate their health status based on their height. Figure 4 shows a scatterplot, colored by the target variable, of the relationship between height and weight in the data set. Typically, an individual determined to be unhealthy is either heavier set and shorter or taller and lighter in weight. If this relationship had a confounding effect on the data, it is expected that the points representing death follow a diagonal line with a negative slope. However, the plot shows that instances where the patient died are randomly scattered throughout the plot. Thus, the relationship between height and weight, described above, does not influence the data set. Figures in the appendix show the individual distributions of height and weight, with their relationship to the target variable.

One aspect of health is the patient’s ethnic background. It is known that ethnicity, and therefore genetics, can lead to an inherent predisposition for illnesses and immunity. Thus, when examining the demographics of patients in the data set, it is key to also study the relationship between ethnicity and the target feature. Figure 5 shows two sets of bar charts. On the left, there

is a bar chart of frequency for each category in the variable, ethnicity. On the right, the normalized shows the proportion of the target variable by each category in the predictor. The plots reveal that the majority of data is represented by Caucasians. While the differences in total count of each ethnicity may lead to some selective biases, the proportion of death and non-death instances are roughly the same across ethnic classes. Thus, there is limited influence by ethnicity on mortality rate, as the probability of death is approximately equal by each ethic class.

Preparing the Data for Modeling

The severe class imbalance was resolved using down sampling since all models had difficulty predicting the minority class with the class imbalance. Before resampling, our training set consisted of 71,195 observations. Upsampling would limit our modeling capabilities due to computing and memory constraints related to working with a training set with more than 120,000 observations. After downsampling to balance our classes, our training set consisted of 12,332 observations - resulting in significant and proportional model performance and computational time improvement.

Based on the aim to explore multiple modeling approaches, training and test sets were developed with respect to varying model assumptions. Some algorithms require all features to be numeric; therefore, non-numeric categorical features were encoded using the One-Hot encoding methodology. The first encoding for each feature was dropped to mitigate collinearity concerns and assist with feature reduction. Next, linear dependence was evaluated to mitigate multi-collinearity concerns. Twelve features were identified as linear dependent - these features represent duplicate observations of APACHE II and III body system values. Since APACHE III represented the most updated versions, they were retained and the APACHE II versions were

dropped. This resulted in a second training and test set named training_with_dummies_r and testing_with_dummies_r.

To summarize our training and testing sets, X_train, X_test, y_train, and y_test sets consisted of the non-encoded categorical features for use in tree-based classifiers after converting categorical features to factors. The training_with_dummies_r and testing_with_dummies_r with their corresponding y_train and y_test sets consisted of the one-hot encoded, non-linear dependent features for use in the non-tree-based classifiers.

Model Strategies

The main objective of this study was to accurately classify patients into a binary category, “no_death” or “death”. Given this objective, a logistic regression model seemed reasonable. Thus, logistic regression was the first model constructed, and acted as the baseline model. Discriminant analysis and logistic regression are similar in that they both produce probabilities for classification into each outcome class. However, discriminant analysis assumes a normal distribution in the predictors. In addition, linear discriminant analysis (LDA) and partial least squares discriminant analysis (PLSDA) estimate the coefficients differently than logistic regression (Antonogeorgos et al., 2009). Mixture Discriminant Analysis (MDA) was also explored as a possible solution since it is a non-linear form of discriminant analysis. It was also thought that penalizing the logistic regression model may enhance its performance.

To evaluate a full range of classification models, decision trees, a random forest, a boosted tree, a K-Nearest Neighbors (KNN), and a neural network model were also constructed and evaluated. Unlike logistic regression, LDA, and PLSDA, decision trees were chosen based on their ability to make informed decisions through evaluations of the most significant features at each node. However, because a simple decision tree model can be prone to overfitting, a random

forest model and boosted tree model were also utilized. While the two models suffer from a limited interpretability, it was necessary to evaluate models that would reduce the weaknesses of a standard decision tree, such as the decision tree's high variance.

KNN was utilized to determine if the data had any underlying groupings that could predict the target variable. Where logistic regression was implemented based on the nature of the target variable, a KNN model was used based on the nature of the situation. In healthcare, illnesses are often diagnosed by symptoms. The more symptoms a patient has to a known disease, the more likely that patient has that disease. Thus, it is reasonable to say that the more similar a patient is to another, whose target level is known, the more likely the patient is to fall in the same target level. Because KNN is a similarity based model, it was deemed appropriate for the purposes of this study.

Finally a Neural Network (NN) approach was explored. NN's were developed to loosely mimic the human brain and consist of a set of algorithms designed to recognize patterns. Since labeled training data exists, the NN can be used for our classification problem. NN's help to cluster and classify by recognizing patterns and finding the correct manner of transforming predictors into response variables to optimize the classification. When additional layers (or nodes) are introduced its findings are summed and passed through a node's activation function to measure progress in classification performance. In this way, NN's identify the most significant features with regard to how the NN classifies and clusters input. It should be noted that NN's are computationally exhaustive and not very interpretable; however, their predictive capabilities are often exemplary, so it was explored.

All models were developed with consistent training control parameters which included 10-fold cross validation, two class summaries, and calculation and saving of class probabilities.

Also, given the objective of correctly classifying patients likely to die in order to provide life-saving treatment, models were trained to optimize for ROC. Sensitivity is used for final model selection since it is the proportion of true positives tests out of all patients with a condition. In other words, it is the ability of the model to yield a positive result for a subject that is likely to die.

Validation and Testing

To combat the phenomena of underfitting and overfitting our models, the trade-off between bias and variance was highly considered to minimize the risk when validating the classification models on the test data sets. Given the large data set and down sampling due to the risk of overfitting and constraint of computational time, the chosen models were selected that would best predict the response variable of "death" and "no death". Whereas, for model tuning all the models had hyperparameters that allowed adjustment for prediction accuracy and computational time as the dataset included many features and observations. These adjustments were initially applied to the training data set to identify optimal performance metrics to run the testing data set for evaluation of performance on classification.

With the ten chosen models, the hyperparameters varied between each to decrease computational time, increase the performance of accuracy, and chosen metric of ROC. These hyperparameters include setting the training control (ctrl) with cross-validation, two class summary, and centering and scaling for each model. The Logistic Regression model does not have hyperparameters to adjust. The Penalized Logistic Regression included had tuneGrid to optimize the alpha (.2, .4, .6) and lambda (length = 2) parameters for regularization. LDA only varied with the preproc() function with Box-Cox transformations, PLSDA was adjusted with the number of components (ncomp = 1:10), and MDA had the parameter of subclasses with tuneGrid

(subclasses=1:3) to set the number of distributions per class. For the Decision Tree (CART), the tuneLength was set to a value of 30. The Random Forest model had the mtry parameter which was set to 7 (mtry = sqrt(p)). The Boosted Tree included the tuneGrid() function, where the the maximum number of nodes per tree, number of trees, shrinkage parameter, and minimum number of observations in trees' terminal nodes can be set (interaction depth = c(5), n.trees = (1:20)*5, shrinkage = c(.01, .1), and minobsinnode = 5). As for KNN, the hyperparameter of the number of neighbors (k) was set between 111 to 151 with increments of 2. Finally, the Neural Net model with hyperparameters of size (number of units in hidden layer) and decay (regularization parameter) are set within the tuneGrid() function (size=1:2 and decay=c(0,0.1,1). To evaluate the models, it can be observed in Table 1 of the *Results and Final Model Selection* where the ROC, Specificity, and Sensitivity metrics were compared between the training and testing data sets.

Results and Final Model Selection

After training each model, an ROC curve was produced. Figure 6 shows all the ROC curves plotted on the same graph. Ideally, a model would have an ROC curve that follows the y-axis vertically, then moves horizontally once sensitivity equals one, thus maximizing the area under the curve (AUC). However, the form of the curve described previously is not always achievable. Fig. 6 shows that the ROC curves do take on a positive, concave shape, an expected shape for non-ideal situations. Based on these curves, the best models would be those with the largest AUC. Without exact metrics however, the curves look nearly identical. Table 1 shows that the best three models, based on the ROC curves, are the boosted tree model (AUC = 0.8746), the random forest model (AUC = 0.8634), and the logistic regression mode (AUC = 0.8579).

In addition to AUC, sensitivity is another metric that can be used to measure a model's predictive ability. As described previously, sensitivity is a measure of the ability to correctly predict the positive class. In this study, sensitivity is a measure of how well the model can predict a patient whose target level is "death", given that the patient does belong to the class "death." Of the top three models, based on ROC, the random forest and boosted tree models had the highest sensitivity on the training set (sensitivity = 0.7955). Of the two models, based on the test data, the random forest model had the highest sensitivity (0.7988), whereas the sensitivity for the boosted tree model was equal to 0.7917.

Though the random forest model was a higher sensitivity metric on the test data, the value was not that different from the sensitivity metric of the boosted tree model. Even though the random forest model outperformed the boosted tree model in one metric, the boosted tree model performed as good or better than the random forest model for every other metric, presented in Table 1. Due to computational constraints, a wider array of hyperparameters could not be examined for each model. A wider exploration of hyperparameters could potentially help distinguish the better model between the two. However, with the constraints, the boosted tree model performed better on all but one metric. Therefore, for that reason and the fact that its weakest metric was not far off from the best value, the boosted tree model was selected as the best model.

Discussion and Conclusions

The objective of this study was to identify suitable model(s) that could correctly identify high risk patients. Of the 10 models tested, nine of which had sensitivity values above 75% percent; the KNN model had a sensitivity value of 0.6030 on the test data. Thus, this study found multiple models capable of achieving the objective. Given greater computational power, the

study could have expanded the number of hyperparameters tested for each model, thus bettering each model's performance. The work done here contributes to the ongoing discussion of machine learning algorithms in the healthcare industry. With over 75% probability of correctly classifying high risk cases, the models in this work show that machine learning is capable of assessing and identifying patients that have a high risk for mortality. Where previous models, as discussed earlier, were highly successful at identifying patients on the basis of a common illness, these models indicate that, given the appropriate features and computational strength, such models can be expanded to apply to patients with varying conditions. Having such a technology could greatly help healthcare workers decide where to allocate resources. Through this technology, common features that are shared among high risk patients can be identified, thus highlighting areas of research that would benefit from increased funding. Continuing this study and improving upon the work done here will have numerous benefits for many industries and populations.

References

Antonogeorgos, G., Panagiotakos, D. B., Priftis, K. N., & Tzonou, A. (2009). Logistic regression and linear discriminant analyses in evaluating factors associated with asthma prevalence among 10-to 12-years-old children: divergence and similarity of the two statistical methods. *International journal of pediatrics, 2009*.

Green, J., Wintfeld, N., Sharkey, P., & Passman, L. J. (1990). The importance of severity of illness in assessing hospital mortality. *JAMA, 263*(2), 241-246.

Hsu, C. W., Wann, S. R., Chiang, H. T., Lin, C. H., Kung, M. H., & Lin, S. L. (2001). Comparison of the APACHE II and APACHE III scoring systems in patients with respiratory failure in a medical intensive care unit. *Journal of the Formosan Medical Association, 100*(7), 437-442.

Kuhn, M., & Johnson, K. (2013). *Applied Predictive Modeling*. New York: Springer.

Pourhomayoun, M., & Shakibi, M. (2021). Predicting mortality risk in patients with COVID-19 using machine learning to help medical decision-making. *Smart Health, 20*, 100178.

Schafer JL. Multiple imputation: a primer. *Stat Methods in Med*. 1999;8(1):3–15. doi: 10.1191/096228099671525676.

Steele, A. J., Denaxas, S. C., Shah, A. D., Hemingway, H., & Luscombe, N. M. (2018). Machine learning models in electronic health records can outperform conventional survival models for predicting patient mortality in coronary artery disease. *PloS one, 13*(8), e0202344.

Appendix

Figure 1

Bar Charts of the Target Feature Isolated and Associated with ICU Stay Type

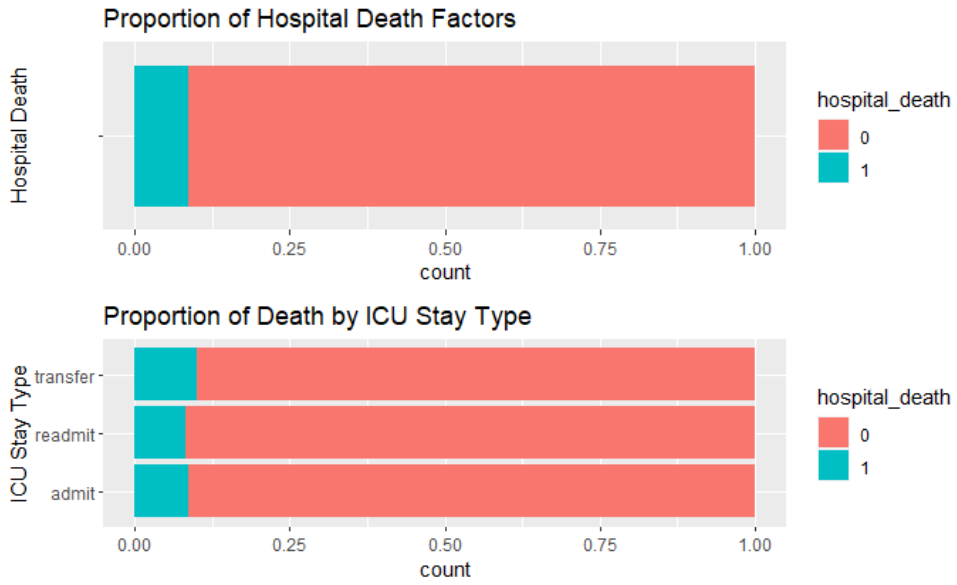

Figure 2

Stack Bar Charts of Categories for Each Apache System

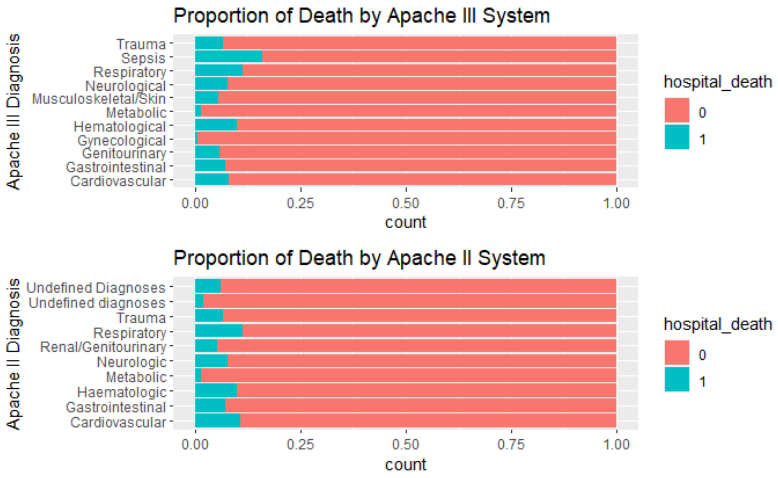

Figure 3

Distribution of Age

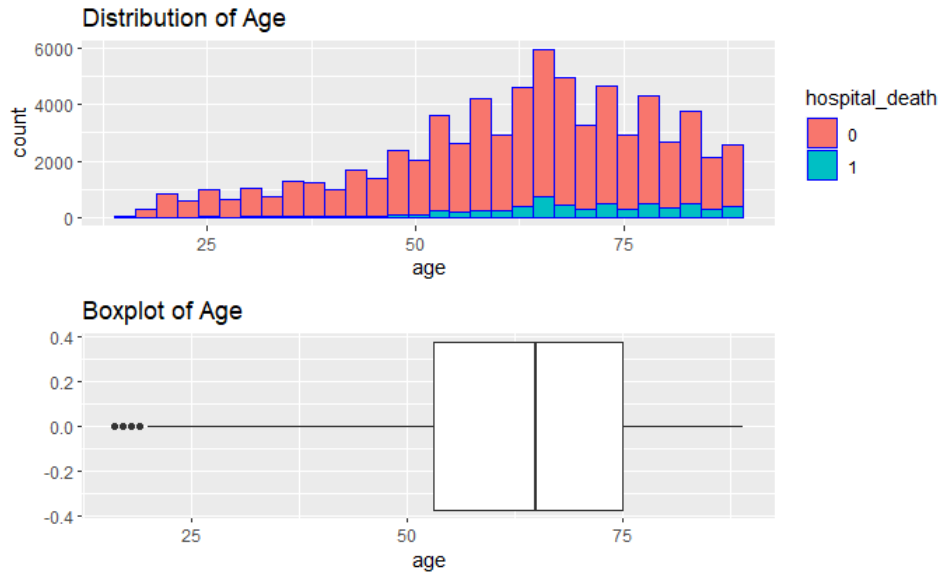

Figure 4

Scatterplot of Height and Weight By the Target Feature

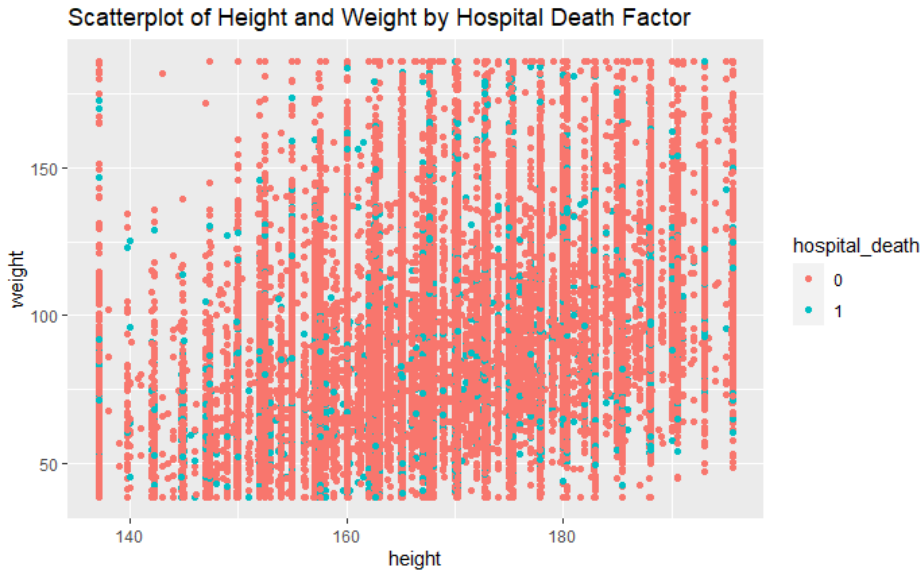

Figure 5

Bar Charts of Ethnicity

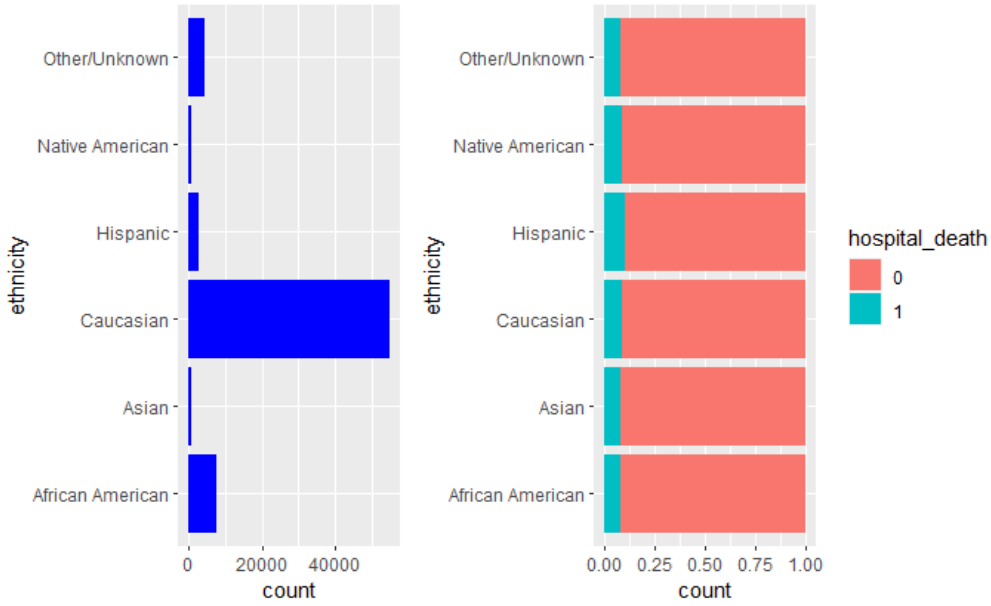

Figure 6

ROC Curves for Each Model Based on Training Data

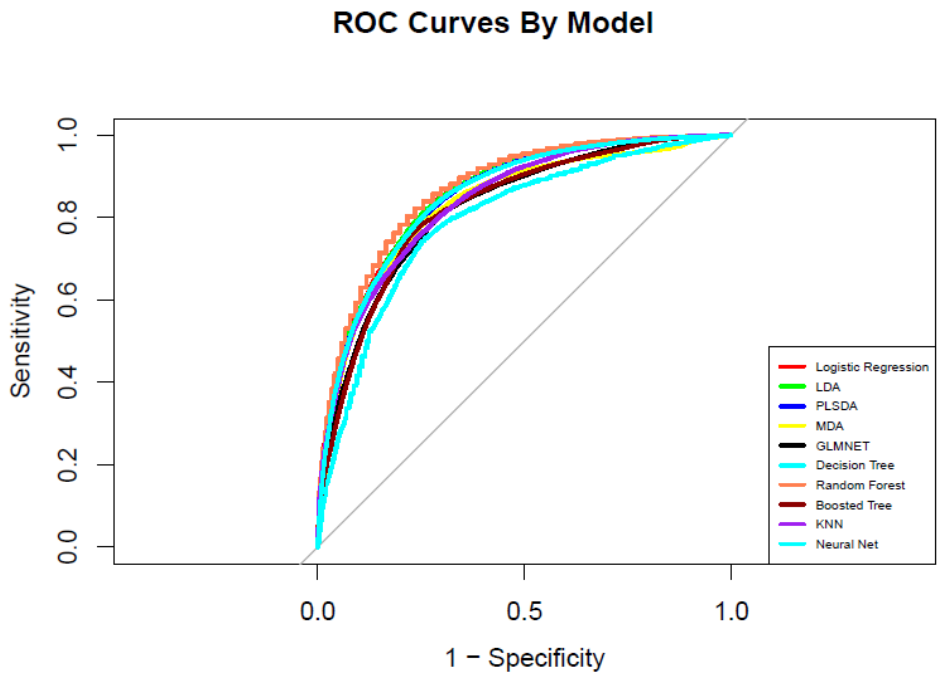

Table 1

Performance Metrics of Each Model

Models	ROC..Train	Sens. . . .Train	Spec. . . .Train	Sens. . . .Test	Spec. . . .Test
Logistic Regression	0.8579	0.7683	0.7830	0.7611	0.7972
Penalized Logistic Regression	0.8576	0.7613	0.7835	0.7578	0.7982
LDA	0.8574	0.7624	0.7848	0.7533	0.7991
PLSDA	0.8565	0.7606	0.7843	0.7507	0.7985
MDA	0.8559	0.7588	0.7836	0.7513	0.7989
Decision Tree	0.8167	0.7634	0.7499	0.7533	0.7547
Random Forest	0.8634	0.7955	0.7699	0.7988	0.7777
Boosted Tree	0.8746	0.7955	0.7851	0.7917	0.8000
KNN	0.8401	0.5981	0.8766	0.6030	0.8795
Neural Net	0.8571	0.7833	0.7681	0.7910	0.7710

Note. Sens. is an abbreviation for sensitivity and Spec. is an abbreviation for specificity. Columns ending in "Train" indicate that the values are based on the training data, whereas columns ending in "Test" are based on the test data set.