Final Project

Warm Up: Machine Learning with a Heart

Name: Young, Brent

DrivenData Name: bdy3176

DrivenData Public Score: 0.31960

MSDS 454 Section #: 55

Quarter: Summer 2018

Introduction

Problem

The purpose of the final project is to analyze various measurements on patient health and cardiovascular statistics to predict the probability that a patient has heart disease. The dataset is from a study of heart disease that has been open to the public for many years, courtesy of the Cleveland Heart Disease Database via the UCI Machine Learning repository. However, predicting the probability that a patient has heart disease can be difficult given that the data provided is one of the smallest datasets on DrivenData. For instance, it has only 180 records in the training dataset and 90 in the test dataset.

Significance

The problem is significant/interesting because good data-driven systems for predicting heart disease can improve research and prevention processes. This can lead to healthier lifestyles and help minimize the risk of heart disease in the future. This is particularly important given that heart disease is the number one cause of death worldwide for both men and women. In fact, according to the Centers for Disease Control and Prevention, about 610,000 people die of heart disease in the U.S. every year and about 735,000 Americans have a heart attack each year *(525,000 are a first heart attack and 210,000 happen in people who have already had a heart attack)*. Additionally, heart disease is the leading cause of death in the U.S. for African Americans, Hispanics, and Whites, while for Asian Americans, heart disease is second only to cancer.

Applicability to Data Scientists

This real-world data problem is applicable to data scientists because it serves as a great resource for practicing our data science skills and testing out various machine learning algorithms. For instance, since this is a classification problem, it'll provide data scientists the opportunity to practice classification modeling techniques such as Logistic Regression, LDA, FDA, KNN, Decision Trees, Bagging, Random Forest, Neural Network, Gradient Boosting Machines, and XGBoost. Furthermore, the popularity of analytics within the healthcare industry has risen tremendously in the recent years, so I thought it would be good to gain exposure to this industry.

Data Exploration

Structure and Description of Training, Validation, and Test Datasets

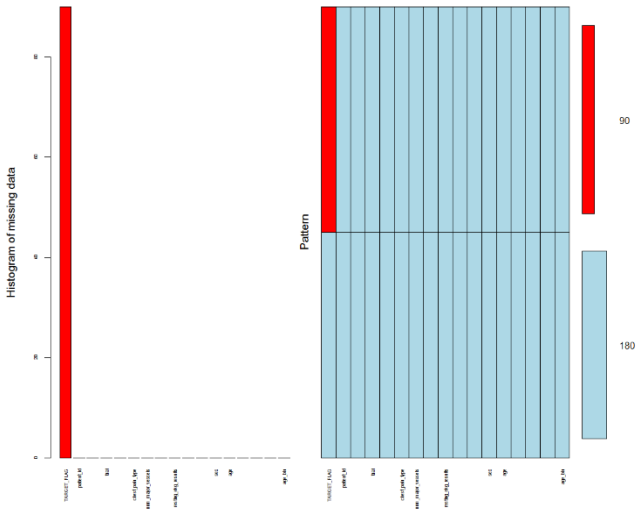
To help streamline the analysis, both the train (180 observations) and test datasets (90 observations) have been consolidated into one dataset called total. The combined dataset prior to feature engineering consists of 15 variables (includes identification and response variable) and 270 observations. The data was then split into a 33.3/33.3/33.3: train/validation/test so that the validation and test sets have the same amount of observations. For instance, there are 90 training observations (33.3%), 90 validation observations (33.3%), and 90 test observations

(33.3%). The training set will be used to fit the models, the validation set will be used to estimate prediction error for model selection, and the test set will be used for assessment of the prediction error of the final chosen model.

Data Mining/ Cleaning

Cleaning

In order to make the data more R friendly, I renamed heart_disease_present (response variable) to TARGET_FLAG and changed the following variables to factor variables: TARGET_FLAG, fasting_blood_sugar_gt_120_mg_per_dl, sex, and exercise_induced_angina, respectively. After conducting summary statistics on the dataset. The results showed that there were no missing values, except the 90 TARGET_FLAG NA's in the test set (see right). As a result, missing value imputation was not conducted.



Summary of Variables, Feature Creation, Measurement Levels, and Standardization

Patient_id, which is used for identification purposes will be ignored. TARGET_FLAG represents our classification response variable of whether or not a patient has heart disease (0 = No, 1 = Yes). There is a total of 40 people who had heart disease in the training dataset and 40 people who also had heart disease in the validation dataset. The other 13 variables are a mix of numeric and categorical variables. For instance, the numeric/integer variables consist of slope_of_peak_exercise_st_segment, resting_blood_pressure, chest_pain_type, num_major_vessels, resting_ekg_results, serum_cholesterol_mg_per_dl, oldpeak_eq_st_depression, age, and max_heart_rate_achieved. The categorical variables consist of thal (aka: thallium stress test), fasting_blood_sugar_gt_120_mg_per_dl, sex, and exercise_induced_angina (aka: exercise-induced chest pain). Through feature creation, I also created one new numeric variable, which proved to be a total "game changer" and proved to be one of the most important variables in most of my top models: Percentage of Max. Predicted Heart Rate Achieved (max_predicted_heart_rate_achieved = max_heart_rate_achieved/(220-age)). *Note: The formula was found on the Canadian Society of Echocardiography website.* I also created one qualitative variable: age_bin, which creates groups for ages 29 to 44, 45 to 64, and 65+. However, it's important to note that this variable was not beneficial and was removed from most of my models. Furthermore, SQRT & LOG transformations were also conducted on most of the variables. However, after using the transformed variables on some of my models, I did not notice any improvement, so I removed it from my code. Lastly, all variables, except, patient_id, TARGET_FLAG, and all the other

categorical variables have been standardized to have a mean of 0 and standard deviation of 1 in the training, validation, and test datasets. Note: Standardization was conducted after EDA.

Data Visualization

Descriptive Statistics

Presence of Heart Disease	Count
Yes	40
No	50

of Missing Values
0

Quantitative Variables (Averages)	
resting_blood_pressure	130.8
serum_cholesterol_mg_per_dl	250.6
oldpeak_eq_st_depression	0.9811
age	53.44
max_heart_rate_achieved	150.7
max_predicted_heart_rate_achieved (%)	0.9035

Figure 1a

Qualitative Variables (Counts)		
thal	thal (normal)	44
	thal (reversible_defect)	45
	thal_fixed_defect	1
fasting_blood sugar > 120	Yes	11
	No	79
sex	Male	66
	Female	24
exercise_induced_angina	Yes	29
	No	61
age_bin	29 to 44	21
	45 to 64	56
	65+	13

Figure 1b

Observations: Figure 1a shows a table of some of the summary statistics of the variables in the training dataset so that I can check for missing values, outliers, distributions, etc. The data shows that 40 people had heart disease, while 50 did not. Additionally, the average resting_blood_pressure is 130.8, serum_cholesterol_mg_per_dl is 250.6, oldpeak_eq_st_depression is 0.9811, age is 53.44, max_heart_rate_achieved is 150.7, and percentage of max_predicted_heart_rate_achieved is 0.9035. *The averages of the variables are also shown on the plots below (figures 2 to 4), denoted by a red dashed line.*

Univariate & Multivariate Plots of Numeric Variables

Figure 2

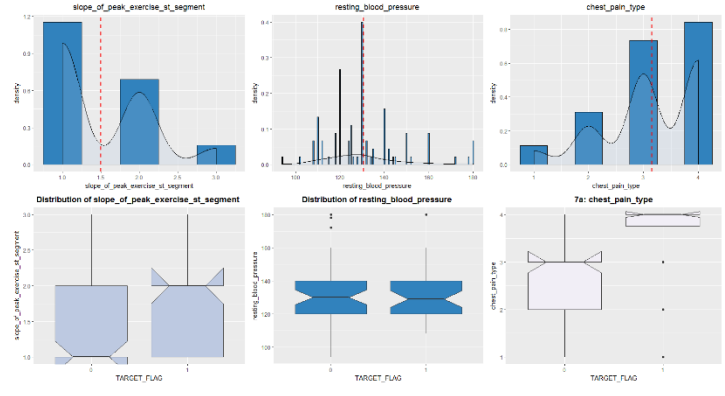


Figure 3

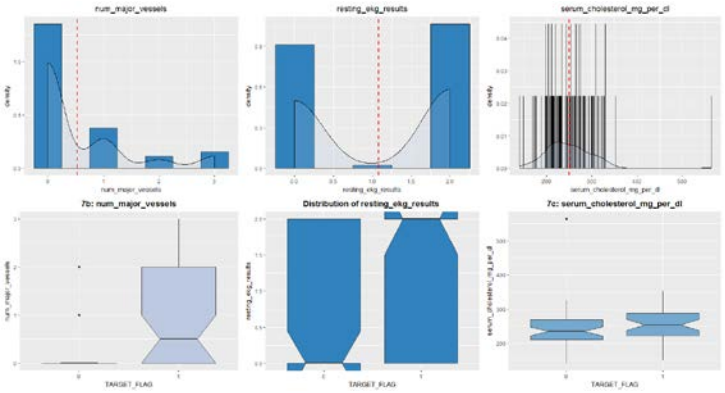
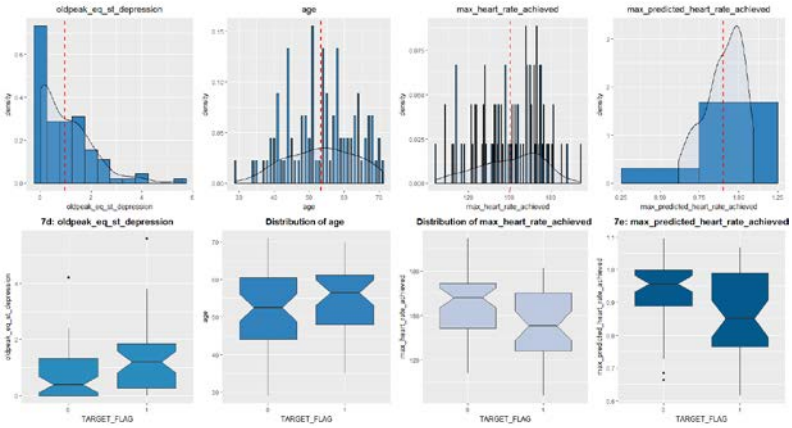


Figure 4



Observations: The histograms, density plots, and notched boxplots above illustrates that there are a few minor outliers in some of the numeric variables such as resting_blood_pressure, chest_pain_type (figure2), num_major_vessels, serum_cholesterol_mg_per_dl (figure 3), and oldpeak_eq_st_depression (figure 4). Furthermore, some of the numeric variables have a right skew (e.g., slope_of_peak_exercise_st_segment, num_major_vessels, serum_cholesterol_mg_per_dl, oldpeak_eq_st_depression), while other variables have a left skew (e.g., chest_pain_type, max_predicted_heart_rate_achieved), bimodal distribution (e.g., resting_ekg_results), or normal distribution (e.g., resting_blood_pressure, age, max_heart_rate_achieved). These distributions and outliers were validated using quantiles as well. However, I decided not to handle/truncate the outliers because it did not improve my models when I tested them on the validation dataset. In regards to the qualitative variables (figure 1b, table), most of the patients in the dataset have a thal of normal or reversible_defect, do not have a fasting blood sugar > 120 mg/dl, are male, do not have exercise-induced chest pain, and are between the ages of 45 to 64 years old.

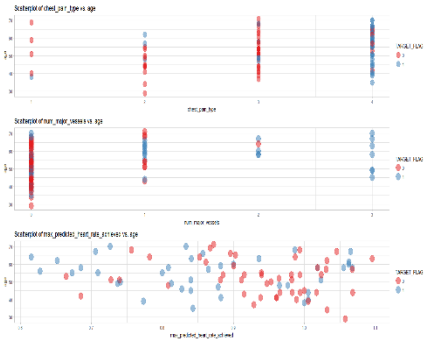


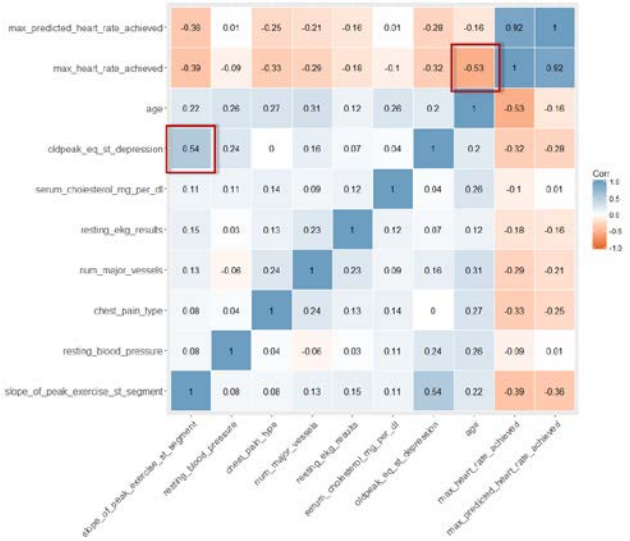
Figure 5

Additional Observations: Figure 2 on the bottom right-hand corner shows a notched boxplot of chest_pain_type vs. TARGET_FLAG, so that I can compare the median differences and variability between the numeric variable and TARGET_FLAG. The results show that the higher the chest pain value (e.g., level 4), the more likely the patient had heart disease (vice versa, *see figure 5 graph, top level, for an additional scatter plot of chest_pain_type vs. age, colored by TARGET_FLAG, red=no, blue=yes, which validates this*). Furthermore, the notch displays a confidence interval around the median which is normally based on the median +/- 1.58*IQR/sqrt(n), which allows us to visually compare if the medians differ. Figure 3 on the bottom left-hand corner shows a notched boxplot of num_major_vessels vs. TARGET_FLAG. The results show that as the number of major vessels colored by flourosopy increases, the more likely the patient had heart disease (vice versa, *see figure 5, middle level, for an additional scatter plot of* num_major_vessels *vs. age, colored by TARGET_FLAG, which validates this*). Figure 3 on the bottom right-hand corner shows a boxplot of serum_cholesterol_mg_per_dl vs. TARGET_FLAG. The results show that as

serum_cholesterol_mg_per_dl increases, the more likely the patient had heart disease (vice versa). Figure 4 on the bottom left-hand corner shows a notched boxplot of oldpeak_eq_st_depression vs. TARGET_FLAG. The results show that as oldpeak_eq_st_depression, which is oldpeak = ST depression induced by exercise relative to rest, a measure of abnormality in electrocardiograms increases, the more likely the patient had heart disease (vice versa). Lastly, figure 4 on the bottom right-hand corner shows a notched boxplot of max_predicted_heart_rate_achieved vs. TARGET_FLAG. The results show that the higher the percentage of max_predicted_heart_rate_achieved, the less likely the patient had heart disease (vice versa, *see figure 5 graph, bottom level, shows an additional scatter plot of* max_predicted_heart_rate_achieved *vs. age, colored by TARGET_FLAG, which validates this*).

Overall, this provides evidence that these variables are strong predictors to include in our models since the median difference between whether or not a patient had heart disease (0 = No, 1 = Yes) is wide. The notched boxplots confirmed this as well. For instance, since the notches of the two boxes do not overlap, there is strong evidence that the medians differ. I also conducted additional notched boxplots of TARGET_FLAG (x-axis) vs. the other numeric variables (y-axis) and found that age and max_heart_rate_achieved (figure 4) have median differences between TARGET_FLAG as well. On the other hand, slope_of_peak_exercise_st_segment (figure 2), resting_blood_pressure (figure 2), and resting_ekg_results (figure 3) did not have median differences between whether or not a patient had heart disease (0 = No, 1 = Yes) and were often removed from my models. Overall, all these variables were also validated using variable importance in the Caret package.

Figure 5: Correlation Matrix *(see appendix for additional GGally::ggpairs scatterplot matrix)*



Observations: Figure 5 shows a correlation matrix of all the numeric variables. This allows us to see which variables may be correlated with each other so that I can gleam interesting insights. The plot shows that there is a strong negative correlation between age and max_heart_rate_achieved (-0.53). The higher the age, the lower the max_heart_rate_achieved. As a result, this provides strong evidence that they are interconnected, which is one of the primary reasons I created max_predicted_heart_rate_achieved during feature creation. This also provides evidence to remove age and max_heart_rate_achieved when using max_predicted_heart_rate_achieved in some of our models. Furthermore, the plots revealed strong positive correlations between slope_of_peak_exercise_st_segment vs. oldpeak_eq_st_depression (0.54), which makes sense given that both metrics involve electrocardiography read outs. As a result, this provides evidence to only include one and not both in some of our models.

Multivariate Plots for Qualitative Variables

Observations: Figure 6 shows bar plots of thal (top left), sex (bottom left), exercise_induced_angina (top right), and age_bin (bottom right) vs. patients who had heart disease and those who did not. The data shows that patients who had a thal of reversible defect (69%) had a higher presence of heart disease compared to those who had a thal of normal (20%) or fixed_defect (0%). Furthermore, the data shows that those who were male (56%) had a higher presence of heart disease compared to those who were female (12.5%). The data also shows that those who had exercise-induced chest pain (69%) had a higher presence of heart disease compared to those who did not (33%). Lastly, patients who fell between the ages of 45 to 64 (50%) and 65+ (46%), had a higher presence of heart disease compared to 29 to 44 year olds (29%). This shows evidence that the differences between the various bins within the variable (e.g., 45 to 65 and 65+) were minimal, which provides evidence to remove it from most of my models. This was also similarly seen in the fasting_blood_sugar_gt_120_mg_per_dl variable. Overall, all these variables were also validated using the variable importance feature in the Caret package.

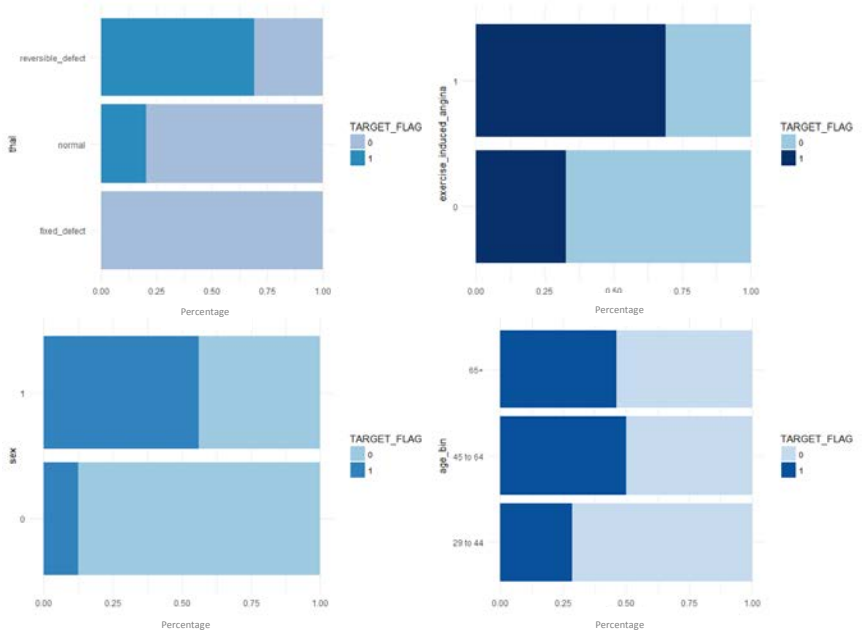


Figure 6: Bar plots of Qualitative Variables vs. TARGET_FLAG

Reviews of Literature & Formulation of Models

Reviews of Literature *(see last page for references)*

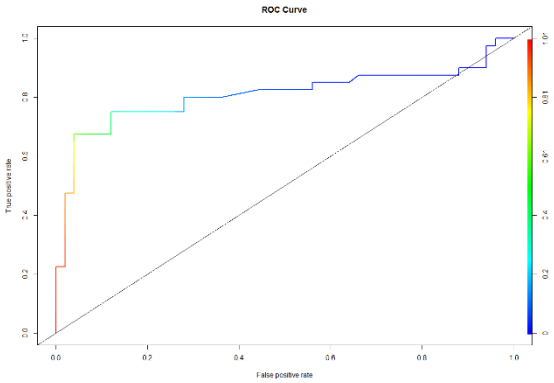
There were many peer reviewed journals in the NU library database that used logistic regression, bagging, random forests, boosting, etc. to predict the probability that a patient has heart disease. For instance, in *The American Journal of Cardiology* (2014), Shmilovich, Cheng, Nakazato, Smith, Thomas, Otaki, Nakanishi, Paz, Pimentel, Berman, and Rajani used logistic regression to estimate the probability of significant coronary artery disease determined by computed tomographic angiography in patients. Second, in the *Journal of Clinical Epidemiology* (2013), Austin, Tu, Ho, Li, Levy, and Lee, use machine learning techniques such as bagging, random forests, and boosting in comparison to logistic regression to predict the probability of the presence of heart failure (HF) with preserved ejection fraction (HFPEF) and HF with reduced ejection fraction. Interestingly, they found that conventional logistic regression had better performance for predicting the probability of the presence of HFPEF compared to the machine

learning methods mentioned above. Lastly, in *Expert Systems* (2017), Gupta, Ahuja, Malhotra, Bala, and Kaur used machine learning techniques such as boosted trees to help predict heart disease based on the historical data of patients to help aid medical experts in their decision-making. They found that using ensemble approaches by combining other models led to increased overall accuracy.

Modeling Strategy

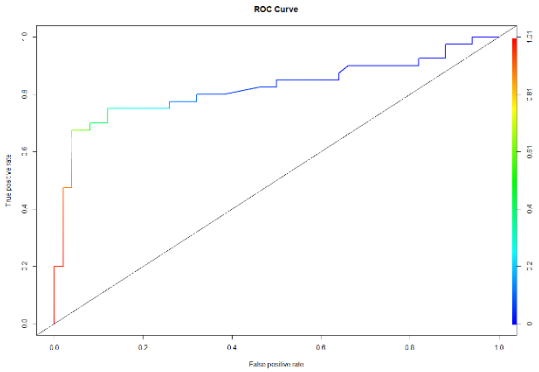
Given that the goal of the DrivenData competition Warm Up: Machine Learning with a Heart is to predict the probability that that a patient has heart disease, I began my analysis using linear classification techniques such as logistic regression with stepAIC and LDA to serve as initial baselines prior to conducting more sophisticated modeling techniques. After building logistic regression and LDA models, I then moved to FDA, KNN, Decision Trees and then ensemble methods (bagging, random forest, gbm, XGboost, and neural network). The next section provides a summary of my results for each modeling technique.

Application of Tools

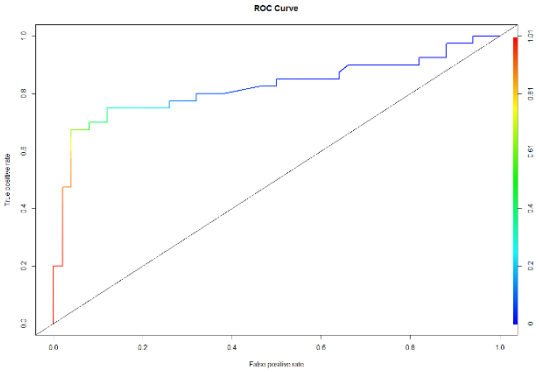


Logistic Regression: Logistic regression models the probability that the response variable belongs to a specific category and assumes a linear decision boundary (James, Witten, Hastie, & Tibshirani, 2013). For instance, it models the probabilities of the K classes using linear functions in x, while also ensuring that they sum to 1 and remain in-between 0 and 1 (Hastie, Tibshirani, & Friedman, 2009). This is accomplished using the logistic function and maximum likelihood, which is used to fit the model (James, et al., 2013). As a result, using the glm function, I produced a logistic regression model of TARGET_FLAG ~ chest_pain_type + thal + num_major_vessels + oldpeak_eq_st_depression + sex. These variables were chosen using stepwise regression (stepAIC). The Analysis of Deviance table and varimp showed that chest_pain_type, impacted the model the most. In regards to the coefficients of the model, variables such as chest_pain_type, which has a positive coefficient make intuitive sense and were statistically significant. For instance, as the chest_pain_type number increases, the more likely a patient has heart disease (vice versa). The model produced the following performance metrics on the training dataset: AIC: 76.28839, BIC: 93.78706 and the following accuracy metrics and evaluation criteria on the validation dataset: accuracy: 0.8222, AUC: 0.80725 (see ROC curve on the right), and LogLoss: N/A. Note: I also tried logistic regression GAM, but the performance was exactly the same as a standard logistic regression model (e.g., model.gam1 <- glm(TARGET_FLAG ~ s(chest_pain_type,5)+ thal + num_major_vessels + oldpeak_eq_st_depression + sex)).

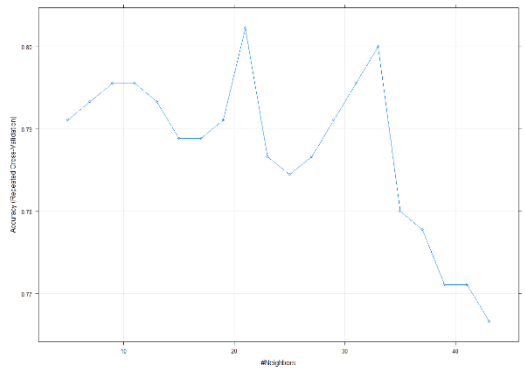
Linear Discriminant Analysis: LDA is very similar in form to logistic regression (distributions are assumed to be normal), except it models the distribution of the predictors separately in each of the response classes and then applies Bayes theorem (James, et al., 2013). This model also uses Gaussian densities, which arises when we assume that the classes have a common covariance matrix, and assumes a linear decision boundary (Hastie, et al., 2009). Using the lda function, I then produced a linear discriminant analysis model of TARGET_FLAG ~ chest_pain_type + thal + num_major_vessels + oldpeak_eq_st_depression + sex. These variables were chosen using the same variables from the logistic regression model (with stepAIC). The model also produced the following accuracy metrics and evaluation criteria: accuracy: 0.8222, AUC: 0.82075 (see ROC curve on the right) and LogLoss: 0.5954447. LDA performed similarly to logistic regression.



Flexible Discriminant Analysis: FDA is an extension of LDA that leads to a more nonparametric and flexible classifier compared to LDA. It is based on a mixture of linear regression models and uses optimal scoring to transform the response variable so that the data are in a better form for linear separation and multiple MARS to generate the discriminant surface (Hastie, et al., 2009). Using the mda function, I then produced a flexible discriminant analysis model of TARGET_FLAG ~ chest_pain_type + thal + num_major_vessels + oldpeak_eq_st_depression + sex. These variables were chosen using the same variables from the logistic regression model (with stepAIC). The model also produced the following accuracy metrics and evaluation criteria: accuracy: 0.8222, AUC: 0.82075 (see ROC curve on the right), and LogLoss: 0.603699. FDA performed similarly to logistic regression and LDA.



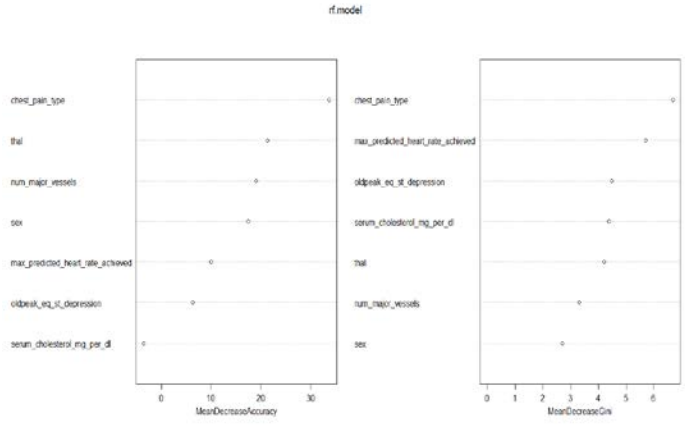
K-Nearest Neighbors: KNN is a non-parametric method that applies Bayes rule by classifying a given observation to the class with the highest estimated probability based on a similarity measure (e.g., distance functions) (James, et al., 2013). Using the function trainControl in the caret package (grid search), K=21 was advised (see graph on the right). The model produced the following accuracy metrics and evaluation criteria: accuracy: 0.8444, AUC: 0.8905 (see ROC curve on the right), and LogLoss: 0.3882528. KNN performed better than logistic regression, LDA, and FDA.



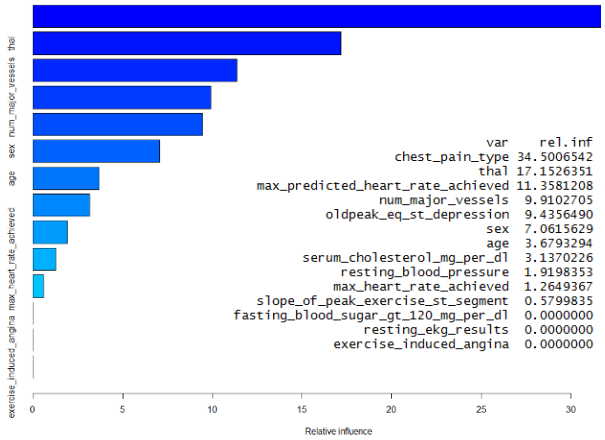
Decision Tree: A decision tree is a tree-based method that involves stratifying or segmenting the predictor space into a number of simple regions (James, et al., 2013). For instance, predictions are made by assigning an observation in a given region to the most common occurring class of training observations in that region (James, et al., 2013). Using the tree function, I produced a decision tree model with 1 predictor variable, after cross-validation helped eliminate all predictor variables, except chest_pain_type, which shows the importance of this variable. The model produced the following accuracy metrics and evaluation criteria: accuracy: 0.7111, AUC: 0.7125, and LogLoss: 0.5583778 on the validation dataset. The performance was worse than logistic regression, LDA, FDA, and KNN.

Bagging: Bagging is a technique for reducing the variance of an estimation prediction function. For classification, a committee of trees each cast a vote for the predicted class (aka: majority vote) (Hastie, et al., 2009). As a result, using the randomForest function (mtry=14, ntree= 1050), a bagged decision tree model was produced using all the predictor variables, except age_bin. The model produced the following accuracy metrics and evaluation criteria: accuracy: 0.7444, AUC: 0.8455, and LogLoss: 0.4310598 on the validation dataset. Bagging was an improvement over a single decision tree and performed better than all the models above, except KNN.

Random Forest: Random forest provides an improvement over bagged trees by incorporating a small tweak that decorrelates the trees and then averages them (e.g., forces each split to only consider a subset of predictors and will not consider strong predictors so that other predictors will have more of a chance (James, et al., 2013)). As a result, using the randomForest function (ntree=2000 and mtry=1, which was found using a grid search), a random forest model was produced using chest_pain_type, thal, num_major_vessels, sex, oldpeak_eq_st_depression, max_predicted_heart_rate_achieved, and serum_cholesterol_mg_per_dl. These variables were chosen using EDA and the variable importance function (which measures prediction strength). The plot on the right shows that chest_pain_type, thal, and num_major_vessels are the most important variables. The model produced the following accuracy metrics and evaluation criteria: accuracy: 0.8333, AUC: 0.8975, and LogLoss: 0.3655469 on the validation dataset. Random Forest performed better than all the models above and had similar performance to KNN.

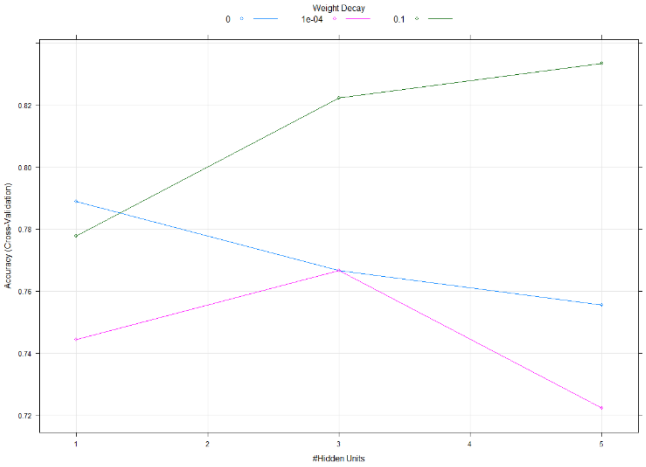


Gradient Boosting Machines: Boosting provides another approach for improving the predictions resulting from a decision tree by fitting each tree on an altered version of the original dataset (James, et al., 2013). In other words, trees are grown sequentially (e.g., each tree is grown using information from previously grown trees). As a result, using the gbm function, a boosted decision tree model was produced using all the predictor variables, except age_bin. I also incorporated n.trees =50, shrinkage=0.1, and depth=1, which was determined using a grid search. Relative importance showed that chest_pain_type, thal, and max_predicted_heart_rate_achieved are the most important variables, similar to what was seen in randomForest (see plot on the right). The model produced the following accuracy metrics and evaluation criteria: accuracy: 0.8556, AUC: 0.8475, and LogLoss: 0.3698292 on the validation dataset. GBM produced the highest accuracy out of all the models above and had similar performance to KNN and Random Forest.



var	rel.inf
chest_pain_type	34.5006542
thal	17.1526351
max_predicted_heart_rate_achieved	11.3581208
num_major_vessels	9.9102705
oldpeak_eq_st_depression	9.4356490
sex	7.0615629
age	3.6793294
serum_cholesterol_mg_per_dl	3.1370226
resting_blood_pressure	1.9198353
max_heart_rate_achieved	1.2649367
slope_of_peak_exercise_st_segment	0.5799835
fasting_blood_sugar_gt_120_mg_per_dl	0.0000000
resting_ekg_results	0.0000000
exercise_induced_angina	0.0000000

Neural Network: Neural network (aka: single hidden layer back-propagation network) is a nonlinear statistical model that is basically a nonlinear generalization of a linear model (Hastie, et al., 2009). It contains inputs, a hidden layer, and outputs that are typically represented by a network diagram (Hastie, et al., 2009). Additionally, neural network has unknown parameters called weights that introduce nonlinearities where needed (Hastie, et al., 2009). Using the nnet function, I then produced a neural network model of TARGET_FLAG ~chest_pain_type + thal + num_major_vessels + oldpeak_eq_st_depression + sex. These variables were chosen using the same variables from the logistic regression model (with stepAIC). Additionally, using the variables from the logistic regression model makes sense given that neural network is essentially a bunch of logistic regressions, fed into a multinomial logit model. I also incorporated 5 hidden layers into the model with a decay= 0.1 and maxit=1000, which was determined using a grid search (see plot on the right). The model produced the following accuracy metrics and evaluation criteria: accuracy: 0.8333, AUC: 0.87425, and LogLoss: 0.4112035 on the validation dataset. Neural network performed worse than KNN, Random Forest, and GBM, but better than the other models.



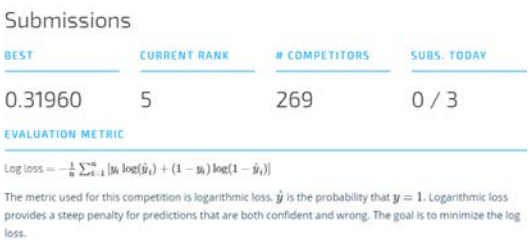
xgbTree variable importance

	Overall
max_predicted_heart_rate_achieved	0.21170
chest_pain_type	0.19837
serum_cholesterol_mg_per_dl	0.17415
thal	0.13709
sex	0.11773
oldpeak_eq_st_depression	0.11139
num_major_vessels	0.04956

XGBoost: XGBoost is similar to GBM, in the sense that it follows the same principle of gradient boosting. However, XGBoost uses different modeling parameters, incorporates parallel processing, and uses a more regularized model formalization to control over-fitting, which often results in better performance. In fact, XGBoost's ability to control over-fitting was very useful given how much Random Forest seemed to over-fit whenever I made submissions to DrivenData. As a result, using the xgbTree in the caret package, an XGBoost model was fit using TARGET_FLAG~chest_pain_type + thal + num_major_vessels + oldpeak_eq_st_depression + sex + serum_cholesterol_mg_per_dl + max_predicted_heart_rate_achieved. These variables were chosen using a combination of EDA and the variable importance function in Random Forest and GBM. Interestingly, max_predicted_heart_rate_achieved was the most important variable, followed by chest_pain_type and serum_cholesterol_mg_per_dl. I also incorporated nrounds = 50, max_depth = 1, eta = 0.3, gamma = 0, colsample_bytree = 0.6, min_child_weight = 1 and subsample = 0.5., which was determined using a grid search. The model produced the following accuracy metrics and evaluation criteria: accuracy: 0.8556, AUC: 0.89125, and LogLoss: 0.370117 on the validation dataset. XGBoost performed the best out of all the models above.

Performance/Accuracy of Classification Models on Validation Set & DrivenData

Model Name	Accuracy	AUC	LogLoss
Logistic Regression	0.8222	0.80725	N/A
Linear Discriminant Analysis	0.8222	0.82075	0.5954
Flexible Discriminant Analysis	0.8222	0.82075	0.6037
KNN	0.8444	0.8905	0.3883
Decision Tree	**0.7111**	**0.7125**	**0.5584**
Bagging	0.7444	0.8455	0.4311
Random Forest	0.8333	0.8975	0.3655
Gradient Boosting Machines	0.8556	0.8475	0.3698
Neural Network	0.8333	0.87425	0.4112
XGBoost	**0.8556**	**0.89125**	**0.3701**



Observations: The heat map above shows the performance/accuracy and evaluation criteria for all the models that were fit in the validation dataset *(blue = best, white = average, and red = worst)*. The results show that XGBoost and GBM had the highest accuracy, while decision tree had the lowest accuracy out of all the models. Additionally, XGBoost, KNN, and Random Forest had similar AUC around 0.88 to 0.89, while the decision tree had the worst AUC at 0.7125. Furthermore, XGBoost, GBM, KNN, and Random Forest had similar LogLoss around 0.36 to 0.38. Given that XGBoost had the highest accuracy, a very high AUC, and a very low LogLoss, I applied the model to the test dataset, and then submitted it to DrivenData. The LogLoss score was 0.31960, which currently places me in the top 2% (5th place) out of 269 competitors (as of 8/25/18) (see visual on top right).

Conclusion

Future Work

In regards to future work, there are four primary areas that could help improve my models. First, it would be beneficial to obtain additional data and variables such as ethnicity, smoker (Yes/No), weight, diabetes (Yes/No), average # of exercise hours per week, and average # of alcoholic drinks per day. Second, it could be helpful creating additional predictor variables through feature engineering and applying them to my models. Third, it could be helpful exploring different mixing of models using a stacking or ensemble approach or model averaging since model diversity can help increase accuracy and performance. Lastly, it could be helpful to explore other boosting packages in R such as CatBoost and Light GBM and other modeling techniques.

Learnings

In the end, I learned three primary things from building these models. First, I learned how to build different classification models using various methods. Second, I learned that it's really important to conduct a thorough EDA and that a lot can be learned from it. For instance, it can inform a modeler which direction he/she should take in regards to feature creation. Third, I learned that trying different modeling approaches can result in better performance and to never settle on a model due to gut instinct. Lastly, I found that feature creation is imperative and can be a huge "game changer", especially when the newly created variable lands within the top 3 in regards to variable importance.

References

1. Austin, Tu, Ho, Levy, & Lee. (2013). Using methods from the data-mining and machine-learning literature for disease classification and prediction: A case study examining classification of heart failure subtypes. *Journal of Clinical Epidemiology*,66(4), 398-407.

2. Shmilovich, Cheng, Nakazato, Smith, Otaki, Nakanishi, . . . Rajani. (2014). Incremental Value of Diagonal Earlobe Crease to the Diamond-Forrester Classification in Estimating the Probability of Significant Coronary Artery Disease Determined by Computed Tomographic Angiography. *The American Journal of Cardiology*, 114(11), 1670-1675.

3. Gupta, N., Ahuja, N., Malhotra, S., Bala, A., & Kaur, G. (2017). Intelligent heart disease prediction in cloud environment through ensembling. *Expert Systems*, 34(3), N/A.

4. James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). An Introduction to Statistical Learning with Applications in R. New York: Springer Science + Business Media.

5. Hastie, T., Tibshirani, R., and Friedman, J. (2009). The Elements of Statistical Learning. New York: Springer Science + Business Media.

6. Canadian Society of Echocardiography. (2016). Maximum Predicted Heart Rate by Age. Retrieved from http://csecho.ca/mdmath/?tag=maxphrage.

Appendix

Scatterplot Matrix of Numeric Variables using GGally::ggpairs