# P8106 Final Project: Predicting COVID-19 Recovery Time and Identifying Significant Risk Factors

Runze Cui (rc3521), Yuchen Hua (yh3555), Hongpu Min (hm2946)

2023-05-01

## Contents

## Data Introduction and Preprocessing

To gain a better understanding of the factors that predict recovery time from COVID-19 illness, this study was designed to combine three existing cohort studies that have been tracking participants for several years. The ultimate goal is to develop two models (the regression model in **Primary analysis** and classification model in **Secondary analysis**) to predict COVID-19 recovery time of patients and identify important risk factors. The dataset (recovery.RData) contains basic demographic characteristics, multiple subject's information about COVID-19 such as severity of infection and COVID-19 recovery time. Also, the dataset provides several biomarkers, vital measurements, and disease status such as height, weight, BMI, hypertension, diabetes, systolic blood pressure and LDL cholesterol. In Primary analysis dataset dat, the study's response is the subject's COVID-19 recovery time, and 14 predictor variables are included (check **Table 1**). Particularly, in Secondary analysis, our continuous response COVID-19 recovery time is converted into binary response (greater than 30 days encoded as "great" and less than or equal to 30 days encoded as "less" in dataset dat_2). Specifically, there are 8 categorical predictors, 6 continuous predictors and 1 binary response variable (check Table 2). The recovery.RData original dataset consists of data on 10,000 participants. A random sample of 3,587 participants was used for data analysis, and the sample size is randomly partitioned into two parts (training data: 70%, test data: 30%).

## Exploratory Analysis and Data Visualization

The primary and secondary analysis shared the same continuous predictors, which contained age, height, weight, low-density lipoprotein (LDL), BMI and systolic blood pressure (SBP). The distributions were visualized as scatterplots and boxplots, respectively (Check **Figure 1.1** and **Figure 1.4**). Also, the violin plots and the bar plots were created to check the categorical predictors' distribution in datasets dat and dat_2 (Check **Figure 1.2** and **Figure 1.5**). A correlation matrix containing only continuous variables helped us to check the linear correlation and prevents the occurrence of multicollinearity bias (Check **Figure 1.3**). For instance, there was a relatively strong positive correlation between BMI and body weight, which might be mirrored in our further model interpretation section. Multiple findings associated with the relationships and hidden patterns are listed as follows:

- There is no obvious association between variables LDL, SBP, age, and COVID-19 recovery time.
- For variables height, there is a slight negative association with COVID-19 recovery time.
- Subjects with relatively higher and lower value on BMI and wight tend to have a longer COVID-19 recovery time. In other words, moderate body shape and weight promote COVID-19 recovery.
- Male patients appear to have a shorter recovery time than females.
- White patients appear to have a shorter recovery time compared to all other races.
- Never-smoked patients seem to have shorter recovery time then patients who is former smoker and current smoker.
- Patients with hypertension appear to have a longer recovery time.
- Patients with diabetes appear to have a longer recovery time.
- Not surprisingly, the vaccinated patients have a shorter recovery time.
- The severely infected COVID-19 patients appear to have a longer recovery time.
- For study (A, B, C where patients belong to), Study B's patients have a shorter recovery time than study A and study C.

Since the only difference between primary and secondary analysis is the type of response variable. Some figures like scatterplot are not suitable for binary outcome anymore. That is why the box plots and bar plots were used for secondary analysis in this section. However, the associations and hidden patterns between predictor variables and outcome are expected to be similar.

## Model Training

Multiple different models (for regression and classification) would be fit to predict the recovery time of COVID-19 using all predictors. During each cross-validation in the primary analysis, 10-fold cross-validation and 5 replications of the training dataset were used as trControl. The twoClassSummary was added in the secondary analysis to compare cross-validation ROCs in the model comparison section. Cross-validation errors, cross-validation ROCs, and misclassification error rates would all be reported for performance evaluation and model comparison. Significantly, the final model would be determined by the cross-validated RMSE and ROC.

### Primary Analysis (Linear and non-linear methods):

**Linear model**: Assumptions: (I). Constant variance (homoscedasticity); (II). Normally distributed error; (III). Observations are independent of each other. (IV). No or little multicollinearity and no autocorrelation. Linear model assumes a basic linear relationship between the predictors and response variables (Set method = "lm"). The variable importance plot (Check

**Figure 2**) was created to visualize the statistical significance of estimated coefficients in the model. The cross-validation RMSE is 23.81.

**LASSO & Ridge models**: The LASSO (Least Absolute Shrinkage and Selection Operator) model is a linear regression model used for feature selection and regularization (set method = "glmnet" and alpha = 1). It adds a penalty term to the ordinary least squares regression model, which helps to shrink the coefficients of less important features to zero. It can also effectively reduce the model's complexity. The expand.grid() function is used to generate all possible combinations of alpha and λ in the grid for tuning. **Figure 3** is the cross-validation procedure for selecting a better tuning parameter λ for lowest CV RMSE. The best tuning parameter λ is about 0.0089 and cross-validation RMSE is 23.81. The Ridge model is similar to LASSO model by adding a penalty term to OLS model, but L2 regularization is used instead of L1 (set alpha = 0 for Ridge fit). The L2 penalty reduces the coefficient's magnitude without setting them to 0. After checking the **Figure 4**, the best tuning parameter λ is about 0.70 and the cross-validation RMSE is 24.95.

**Elastic Net model**: The Elastic Net model is a linear regression model that combines the L1 (LASSO) and L2 (Ridge) regularization methods. It is effective in situations where there are many features, some of which may be highly correlated. It can help to select a subset of features by shrinking the coefficients of less important features towards zero (like the LASSO model), while also handling correlated features by grouping them together and assigning similar coefficients to them (like the Ridge model). Specific alpha is chosen by cross validation. In this case, the best tuning parameter λ is selected as 0.0034 with α equals to 0.20 (check **Figure 5**). The cross-validation RMSE is 23.81.

**PLS model**: Partial least squares (PLS) is a statistical method that is used for modeling the relationship between two data matrices, where one matrix contains the predictor variables and the other matrix contains the response variable. (set method = "glmnet"). PLS is commonly used in situations where there are many predictor variables, some of which may be highly correlated, and where the number of observations is smaller than that of predictor variables. There are 11 components (Check **Figure 6**) in PLS model and the cross-validation RMSE is 23.81.

**GAM model**: GAM model's response variable is modeled as a sum of smooth functions of the predictor variables, where the smooth functions can be nonlinear and non-monotonic. The smooth functions are estimated using nonparametric techniques, such as spline smoothing or kernel smoothing, and the estimation is carried out by optimizing a penalized likelihood function (set method = "gam"). **Figure 7** shows the GAM without feature selection is preferable in this study. The cross-validation RMSE is 22.37.

**MARS model**: The MARS model uses a combination of linear regression and piecewise regression to construct a model that is both flexible and interpretable. It works by recursively partitioning the data based on the independent variables and fitting a linear regression model to each partition. The MARS model is particularly useful when the relationship between the dependent variable and independent variables is complex and nonlinear (set method = "earth"). The degree parameter specifies the maximum degree of interactions between the independent variables that are allowed in the model, and nprune parameter specifies the minimum number of samples that must be present in each terminal node to prevent further splitting. According to the results of cross validation (See **Figure 8**), the best degree parameter is 3 and nprune parameter is 9. The cross-validation RMSE is 22.17.

**Regression Tree:** Regression tree is a decision tree algorithm to build model by recursively partitioning data into smaller regions by choosing nodes to minimize the sum of predictor's squared error (Check **Figure 9.1**). Tuning parameter cost complexity pruning will be applied to obtain the best tree by controlling the trade-off between tree's complexity and its fit to training data (Check **Figure 9.2).** The best tuning parameter Cp is 0.0061 and the cross-validation error is 23.38.

**Random Forests**: The Random Forest algorithm works similarly in regression problems as in classification problems. It can be considered as a kind of 'Blackbox' model, which means it is difficult to describe what the model looks like and interpret it. A set of decision trees is grown using random subsets of the features and data, and the final prediction is made by averaging the predictions of all the individual trees. The cross-validation RMSE performance is visualized as **Figure 10**. Obviously, when mtry equals to 7 and min.node.size equals to 6, the random forests model has the lowest cross-validation error which is 21.67.

**Boosting**: Boosting is an ensemble machine learning algorithm to form an accurate model by combining weaker models into powerful models by correcting the errors of previous models, repeatedly. The models are grown sequentially based on the previous model. The number of trees is controlled to prevent overfitting, the shrinkage parameter will control the boosting learning rate and the number of splits, as the interaction depth, controls the model complexity and model interactions order. Based on **Figure 11**, the minobsinnode is set as 1 (learn slowly for better performance), and the model has the lowest cross-validation error (21.49), when n.trees equals to 4000, interaction.depth equals to 3, and shrinkage equals to 0.003.

## Secondary Analysis:

**Logistic regression & penalized logistic models**: Logistic model (glm) is a type of statistical model for binary classification by modeling the relationship between a binary response and predictors. The response, probability, ranges from 0 to 1, avoiding the problem of linear model, which induces probability to be less than 0 or larger than 1. The relationship between predictor and response tends to be linear on the logit scale. The misclassification error rate is 0.279 and the CV ROC is 0.7095. Besides, penalized logistic model (glmn) is a regularized logistic regression modeling by adding penalty terms to prevent overfitting. Tuning parameters are used to control the strength of the penalty to shrink the coefficient estimates (Check **Figure I**). In this study, $\lambda$ is $2.22\times10^{-5}$ with $\alpha$ equals to 0.47. The misclassification error rate is 0.278 and the CV ROC is 0.7091.

**GAM & MARS models (for classification)**: For both GAM and MARs models for classification response are similar to those for continuous one but different in the function used to transform predictors to final output. Instead of the linear link function used by regression model, non-linear link function is used for transformation. GAM has the best tuning to be choosing GCV.cp method with model selection option. The misclassification error rate is 0.268 and the CV ROC is 0.725. MARS has the best tuning parameter nprune to be 11 with 1 degree. The misclassification error rate is 0.263 and the CV ROC is 0.724. Further details about cross-validation performance can be found on **Figure II** and **Figure III**, respectively.

**LDA & QDA models**: The Discriminant Analysis is method to model distribution of predictors of class separately and flip things based on Bayes theorem. Predictors are assumed to have a multivariate normal distribution within each class. Linear Discriminant Analysis (LDA) further assumes all the classes have a common covariance matrix and the decision boundary between classes is a straight line. The misclassification error rate is 0.279 and the cross-validation ROC is 0.710. Quadratic Discriminant Analysis (QDA), compared with LDA, allows different covariance matrix between classes and non-linear boundary to maximized class separation and minimize class variance, which is a more flexible model than LDA. The misclassification error rate is 0.299 and the cross-validation ROC is approximately 0.696.

**Naïve Bayes (NB)**: Naïve Bayes is a simple and computationally efficient method to classify data. It assumes that the features between classes are independent. It provides a simple and effective way to make classification by making a probabilistic model. "Laplace correction" and parameter fL are applied to increase the count of value to adjust the kernel density estimates and prevent overall probability to be zero. The best tuning parameter adjust is 2.2 by setting fL to be 1. According to **Figure IV** for CV procedure, the non-parametric type performs better than Gaussian. The misclassification error rate is 0.277 and the CV ROC is 0.713.

**Classification trees (rpart & ctree)**: Classification tree is a decision tree algorithm for classification in machine learning. The data will be partitioned based on terminal nodes and stopped when each node has fewer than minimum observations. Gini index or cross-entropy is used to measure mode purity indicating predominant observation from classes. rpart method (Check **Figure V(I)**) and ctree (Check **Figure VI(I)**) were used in the study to split the data into binary partition in a greedy way and a non-binary partition based on statistical test. The rpart 's misclassification error rate is 0.265 and the cross-validation ROC 0.686 under best tuning parameter of Cp to be 0.0092. The ctree 's misclassification error rate is 0.261 and the cross-validation ROC is 0.691 under best tuning parameter of minimum criterion to be 0.75. Further details about cross-validation performance can be found on **Figure V(II)** and **Figure VI(II)**, respectively.

**Random Forests (for classification)**: Random Forests for classification is different from that for regression in the way handling the output variable. The regression model predicts a continuous value while the classification one predicts a categorical value. The best tuning parameters are mtry = 1, splitrule = gini, and the min.node.size = 2 (Check **Figure VII**). The misclassification error rate is 0.293 and the CV ROC is 0.703.

**Adaboost**: Adaptive Boosting is similar to Boosting to improve a more robust model but different in the way adjust weights of data. Each sample in training data will be assigned a weight and the misclassified samples will be corrected by increasing their weights. The exponential loss over the distribution is expected to be minimized after better classifying the misclassified observations. The tuning parameter controls the tree number, shrinkage parameter and interaction depth. 3000 trees are contained with 3 interaction depth and the shrinkage parameter is 0.001 and minimum number of nodes is 1 (Check **Figure VIII**). The misclassification error rate is 0.275 and the CV ROC is 0.716.

**Support Vector Machine (linear and radical in kernlab package)**: SVM is a type of supervised learning algorithm used for classification. To separate data into different classes, the hyperplane maximizing the classes' margin will be detected. The tuning parameter cost is used to control the trade-off between low training error and low testing error. Linear SVM, as a method, is used to separate data with straight line or hyperplane with linear kernel function. The best cost is 0.135. The misclassification error rate is 0.295 and the CV ROC is 0.65 (Check **Figure IX**). The radial kernel will use nonlinear kernel to separate the data by hyperplane. Parameter gamma is chosen to control the smoothness of the nonlinear decision boundary. The best cost is 0.0312. The misclassification error rate is 0.263 and the CV ROC is 0.69 (Check **Figure X**).

## Results

**Model Comparison:** According to **Figure 12** and **Figure XI** below, the models with the lowest median CV RMSE (**Random Forests**) and the highest median CV ROC (**GAM**) values are chosen as the final models in primary and secondary analysis, respectively. Lower CV RMSE, higher CV ROC and lower misclassification error rate indicate better performance during cross-validation procedure. It should be the priority for final model selection. However, the test error could also be checked for the model prediction performance. But it should not be used in determining final models.

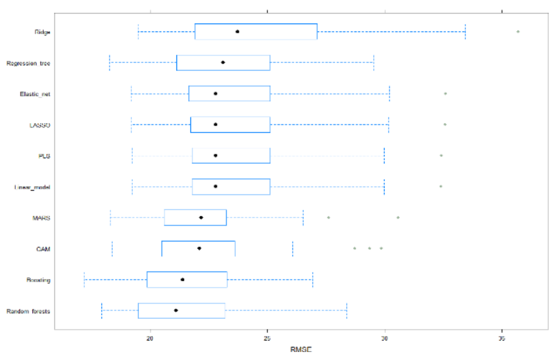


Figure 12. Primary Analysis: Model Comparison Plot Using RMSE

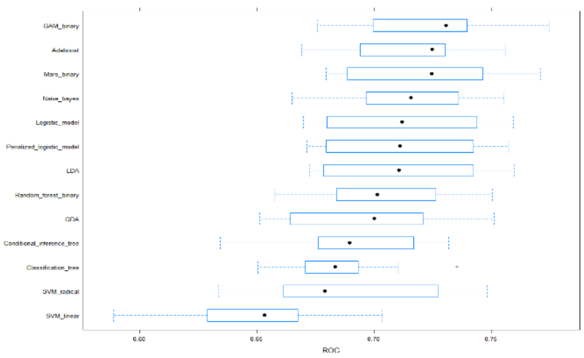
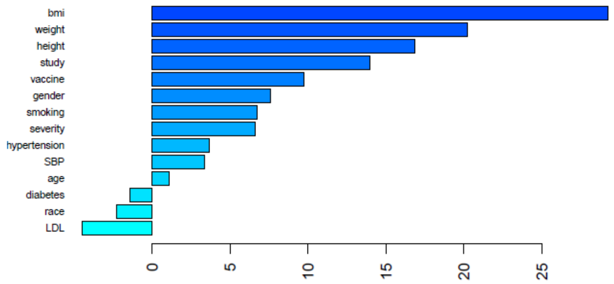


Figure XI. Secondary Analysis: Model Comparison Plot Using ROC

**Final Model Interpretation:** In primary analysis, the final model is Random Forests, which is a black box model. The only way to globally interpret the black box model is checking variable importance, partial dependence plots (PDPs) and individual conditional expectation curves (ICE). The study measure variable importance from the fitted models by permuting OOB data (**Figure 10.1**). In other words, the permutation-based method compares the model performance before and after permuting the predictors. Another way using total decrease in node impurity (**Figure 10.2 in appendix**) is also feasible. From the two variable importance plots, three important variables were selected as they showed to be more significant than others in both plots: BMI, weight, and height.

Figure 10.1. Variable Importance by OOB permutation



PDPs (**Figure 10.3**) were created to find the direction and magnitudes of relationship between predictor and variable, given other conditions constant, which helps us explore the marginal effects of certain important variables in the model. Based on the PDPs, the recovery time is expected to be lowest when the BMI and weight are in a proper value. It increases when BMI and weight either decrease or increase from the proper value. In addition, overweight showed to have much more significant on recovery time increase. The recovery time decreases significantly when height increases and changes

Figure 10.3. Random Forests PDPs for Important Variables

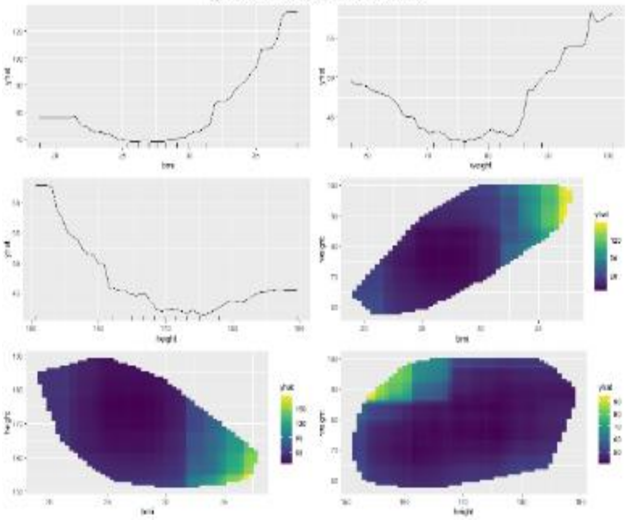
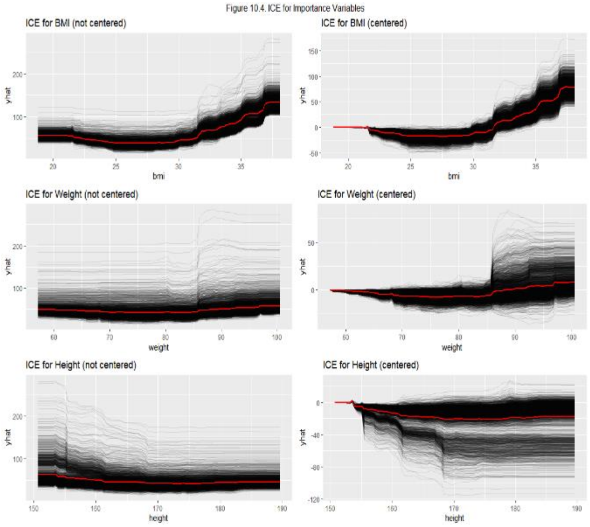


Figure 10.4. ICE for Importance Variables



little when the height reaches 161cm.

However, PDPs are unable to show the heterogeneous effects which can be improved by ICE (**Figure 10.4**). The ICE, similar to PDPs, would visualize the prediction dependence for each instance separately and performs better under interactions. BMI's ICEs show to be steep for both centered and uncentered, indicating a strong effect of BMI on recovery time. Weight's ICEs are not too steep while heights are stable, either centered or non-centered, indicating there are weak effects of weight and height to recovery time. BMI has correlation with weight and height as $BMI = \frac{Weight}{Height^2}$. Height and weight showed to be independent. Due to the existence of interaction, ICE plot would provide a better view than the PDPs.

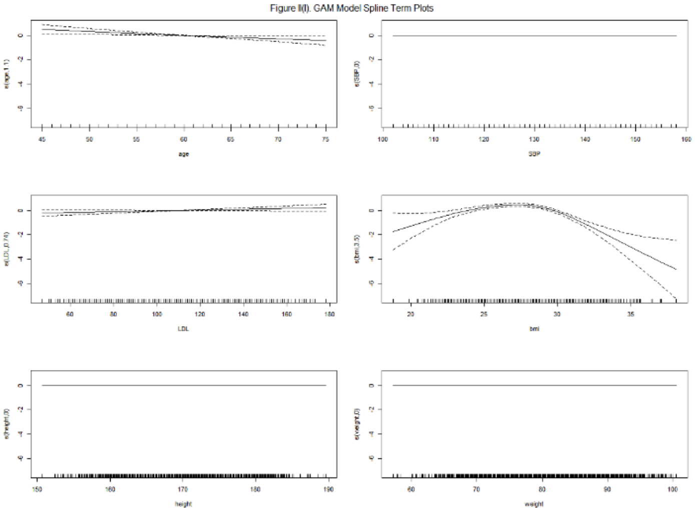


Figure II(I). GAM Model Spline Term Plots

In secondary analysis, the GAM is selected as the final classification model. There are 17 predictor variables in total and 8 of it are statistically significant, which is gender1, smoking1, smoking2, vaccine1, severity1, studyB, age, and bmi. The intercept represents the estimated value of the recovery time when all the predictor variables are equal to 0. The other coefficients represent the estimated change in the recovery time for one unit increase in the corresponding predictor variables, holding other predictor variables constant, and so on. The UBRE is 0.070885, which is the unbiased risk estimate of the model. The scale estimate is 1, which is the estimate of the dispersion parameter of the binomial distribution. The total number of observations in the model is 2513.

Particularly, there are 6 smooth terms which capture the non-linear relationships between the continuous variables and binary response variables. Based on **Figure II(I)** above, for example, the smooth term for BMI has an effective degree of freedom of 3.4962, which indicates that the relationship between BMI and our response is complex and non-linear. The F-statistics for BMI is 86.941 with p value less than $2\times10^{-16}$, indicating this smooth term is a significant predictor of the binary response. In general, since Random Forests regression model belongs to the black box model, it is difficult to show what the final model looks like and interpret by texts only. However, the final classification model GAM can be interpreted as:

- Under given condition, the male has the odds ratio 1.33 over the female to have a short recovery time.
- Under given condition, the former smoker has the odds ratio of 0.646 over the non-smokers to have a short recovery time. The current smoker has an odds ratio of 0.620 over the non-smokers to have a short recovery time.
- Under given condition, vaccinated patients have an odds ratio of 1.71 over the non-vaccinated patients to have a short recovery time.
- Under given condition, patients with severe symptoms have an odds ratio of 0.49 over the patients without severe symptom to have a short recovery time.
- Under given condition, patients in study B have an odds ratio of 3.94 over the patients in study A to have a short recovery time.

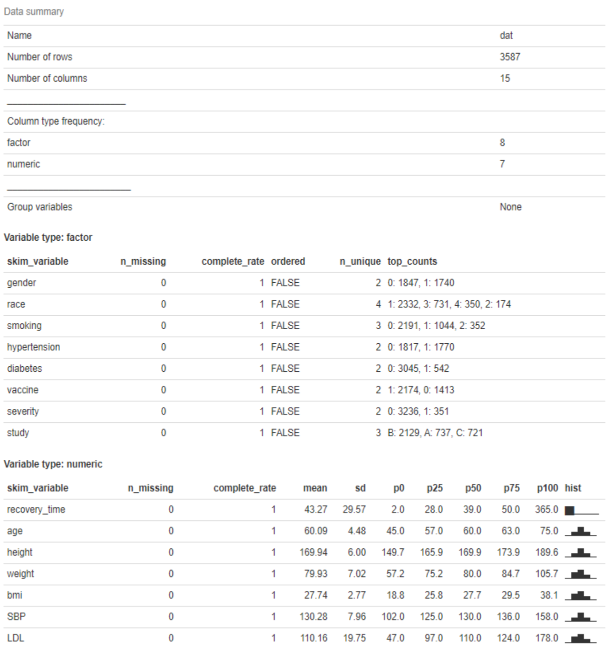
**Model Prediction Performance**: After checking the test error of both final regression and classification models, the test RMSE is 25.76 and test error rate is 0.286. Both models perform well enough in the test datasets.

## Conclusions:

In this study, Random Forest model is selected as the final regression model and GAM is chosen as the final classification model. Variable BMI has a significant effect in both models, indicating it might greatly influence the recovery time after COVID-19. In the Random Forest regression model, through analyzing the PDPs and ICE plots, patients with higher BMI, larger weight and shorter height will be expected to have a significantly longer COVID-19 recovery time. In the GAM classification model, on the other hand, unvaccinated female patients in study B, with smoking history and experiencing severe symptom, will be more likely to have a greater than 30 days of COVID-19 recovery time.

# Appendix

## Table 1: Data Exploration in Primary analysis



Data summary

| | |
|---|---|
| Name | dat |
| Number of rows | 3587 |
| Number of columns | 15 |
| _______________________ | |
| Column type frequency: | |
| factor | 8 |
| numeric | 7 |
| ________________________ | |
| Group variables | None |

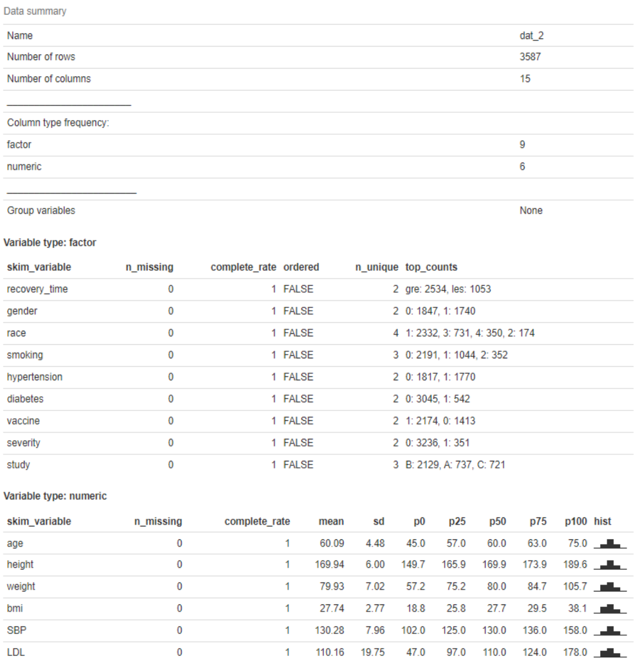
**Variable type: factor**

| skim_variable | n_missing | complete_rate | ordered | n_unique | top_counts |
|---|---|---|---|---|---|
| gender | 0 | 1 | FALSE | 2 | 0: 1847, 1: 1740 |
| race | 0 | 1 | FALSE | 4 | 1: 2332, 3: 731, 4: 350, 2: 174 |
| smoking | 0 | 1 | FALSE | 3 | 0: 2191, 1: 1044, 2: 352 |
| hypertension | 0 | 1 | FALSE | 2 | 0: 1817, 1: 1770 |
| diabetes | 0 | 1 | FALSE | 2 | 0: 3045, 1: 542 |
| vaccine | 0 | 1 | FALSE | 2 | 1: 2174, 0: 1413 |
| severity | 0 | 1 | FALSE | 2 | 0: 3236, 1: 351 |
| study | 0 | 1 | FALSE | 3 | B: 2129, A: 737, C: 721 |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| recovery_time | 0 | 1 | 43.27 | 29.57 | 2.0 | 28.0 | 39.0 | 50.0 | 365.0 | ▇▁▁▁▁ |
| age | 0 | 1 | 60.09 | 4.48 | 45.0 | 57.0 | 60.0 | 63.0 | 75.0 | ▁▃▇▃▁ |
| height | 0 | 1 | 169.94 | 6.00 | 149.7 | 165.9 | 169.9 | 173.9 | 189.6 | ▁▃▇▃▁ |
| weight | 0 | 1 | 79.93 | 7.02 | 57.2 | 75.2 | 80.0 | 84.7 | 105.7 | ▁▅▇▃▁ |
| bmi | 0 | 1 | 27.74 | 2.77 | 18.8 | 25.8 | 27.7 | 29.5 | 38.1 | ▁▅▇▃▁ |
| SBP | 0 | 1 | 130.28 | 7.96 | 102.0 | 125.0 | 130.0 | 136.0 | 158.0 | ▁▃▇▃▁ |
| LDL | 0 | 1 | 110.16 | 19.75 | 47.0 | 97.0 | 110.0 | 124.0 | 178.0 | ▁▅▇▃▁ |

## Table 2: Data Exploration in Secondary analysis



Data summary

| | |
|---|---|
| Name | dat_2 |
| Number of rows | 3587 |
| Number of columns | 15 |
| _______________________ | |
| Column type frequency: | |
| factor | 9 |
| numeric | 6 |
| ________________________ | |
| Group variables | None |

**Variable type: factor**

| skim_variable | n_missing | complete_rate | ordered | n_unique | top_counts |
|---|---|---|---|---|---|
| recovery_time | 0 | 1 | FALSE | 2 | gre: 2534, les: 1053 |
| gender | 0 | 1 | FALSE | 2 | 0: 1847, 1: 1740 |
| race | 0 | 1 | FALSE | 4 | 1: 2332, 3: 731, 4: 350, 2: 174 |
| smoking | 0 | 1 | FALSE | 3 | 0: 2191, 1: 1044, 2: 352 |
| hypertension | 0 | 1 | FALSE | 2 | 0: 1817, 1: 1770 |
| diabetes | 0 | 1 | FALSE | 2 | 0: 3045, 1: 542 |
| vaccine | 0 | 1 | FALSE | 2 | 1: 2174, 0: 1413 |
| severity | 0 | 1 | FALSE | 2 | 0: 3236, 1: 351 |
| study | 0 | 1 | FALSE | 3 | B: 2129, A: 737, C: 721 |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---|---|---|---|---|---|---|---|---|---|---|
| age | 0 | 1 | 60.09 | 4.48 | 45.0 | 57.0 | 60.0 | 63.0 | 75.0 | ▁▃▇▃▁ |
| height | 0 | 1 | 169.94 | 6.00 | 149.7 | 165.9 | 169.9 | 173.9 | 189.6 | ▁▃▇▃▁ |
| weight | 0 | 1 | 79.93 | 7.02 | 57.2 | 75.2 | 80.0 | 84.7 | 105.7 | ▁▅▇▃▁ |
| bmi | 0 | 1 | 27.74 | 2.77 | 18.8 | 25.8 | 27.7 | 29.5 | 38.1 | ▁▅▇▃▁ |
| SBP | 0 | 1 | 130.28 | 7.96 | 102.0 | 125.0 | 130.0 | 136.0 | 158.0 | ▁▃▇▃▁ |
| LDL | 0 | 1 | 110.16 | 19.75 | 47.0 | 97.0 | 110.0 | 124.0 | 178.0 | ▁▅▇▃▁ |

## Figure 1.1

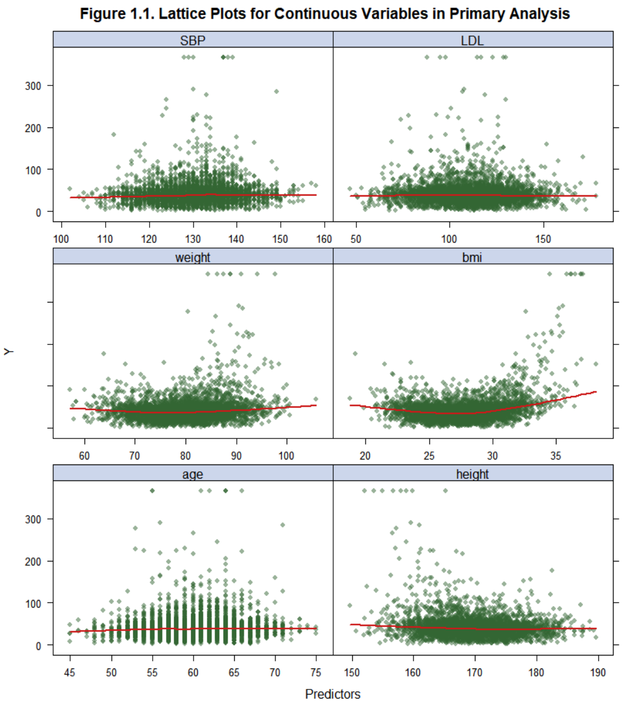


Figure 1.1. Lattice Plots for Continuous Variables in Primary Analysis

## Figure 1.4

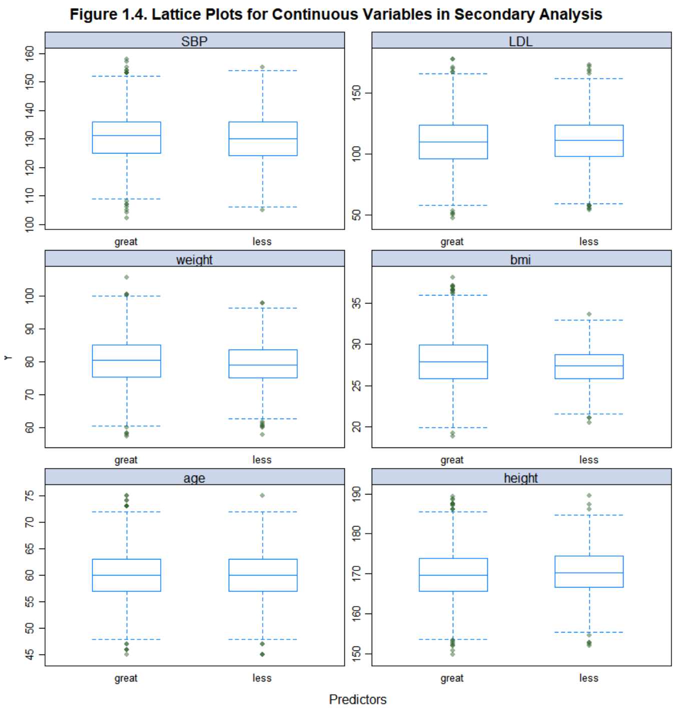
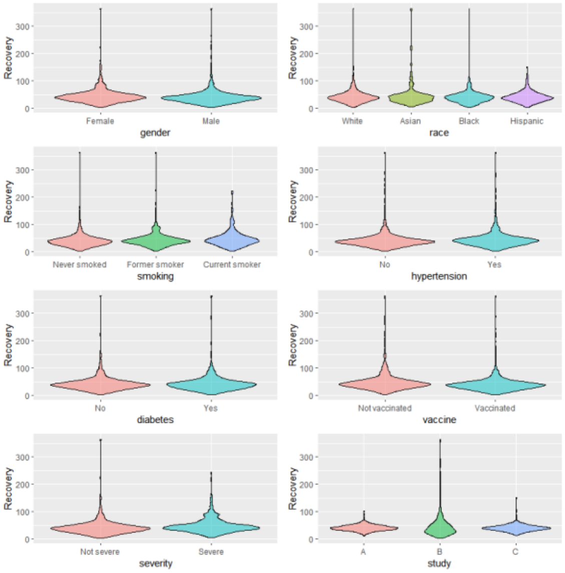


Figure 1.4. Lattice Plots for Continuous Variables in Secondary Analysis

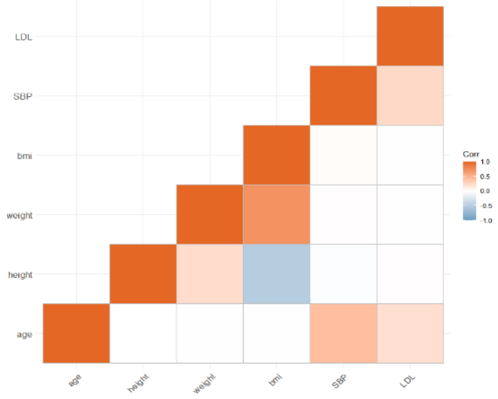
## Figure 1.2

Figure 1.2. Violin Plots for Categorical Variables in Primary Analysis



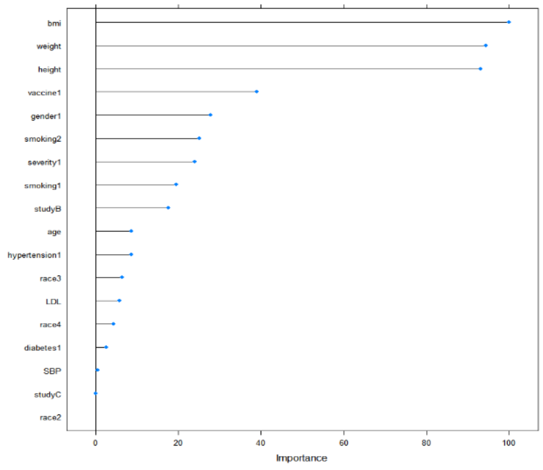
## Figure 1.3

Figure 1.3. Correlation matrix for continuous datasat



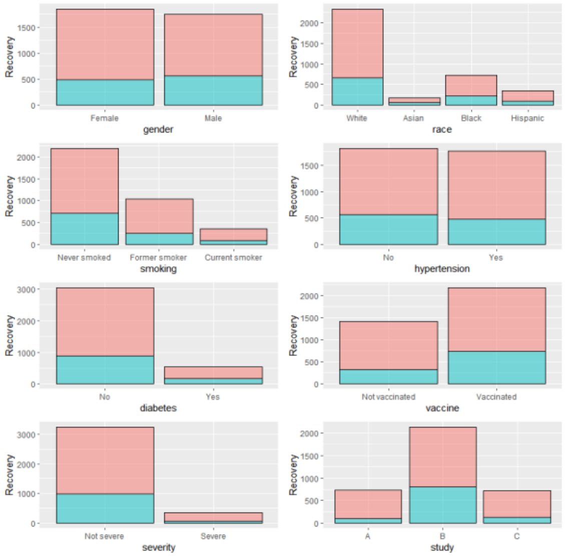
## Figure 2

Figure 2. Linear Model Variable's Importance Plot

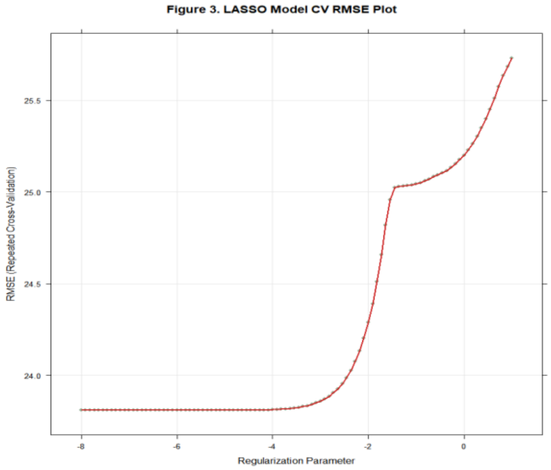


## Figure 1.5

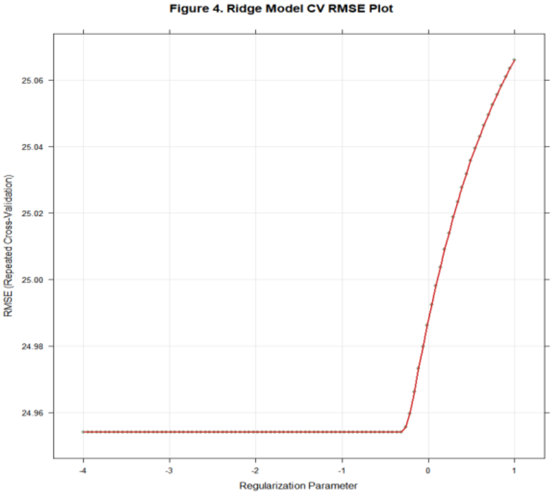
Figure 1.5. Bar Plots for Categorical Variables in Secondary Analysis



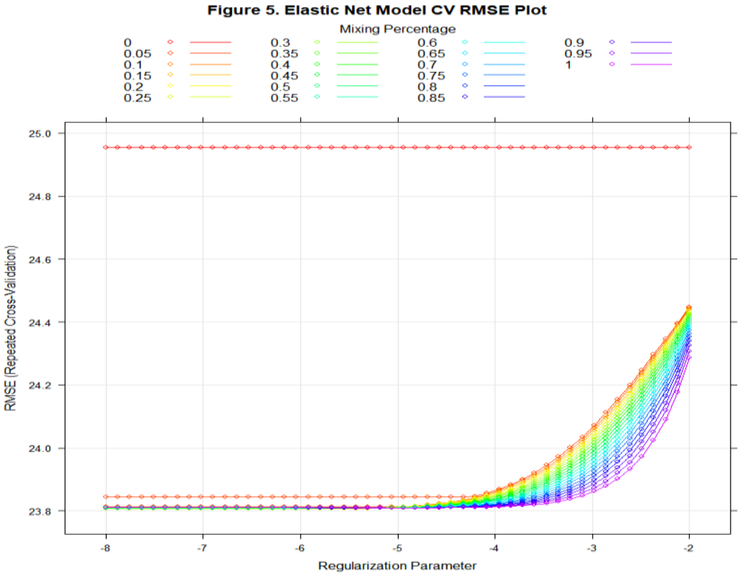
## Figure 3



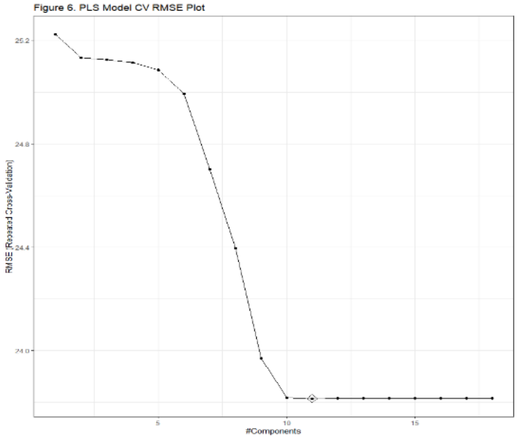
## Figure 4



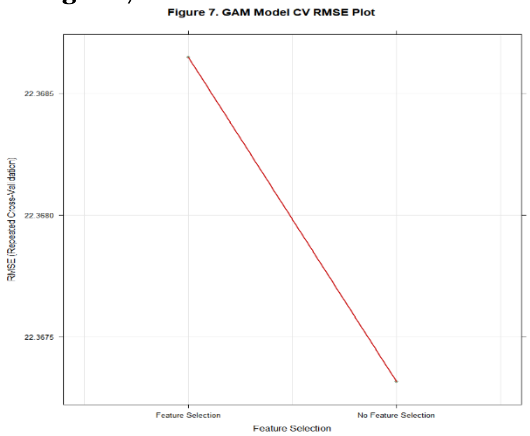
## Figure 5



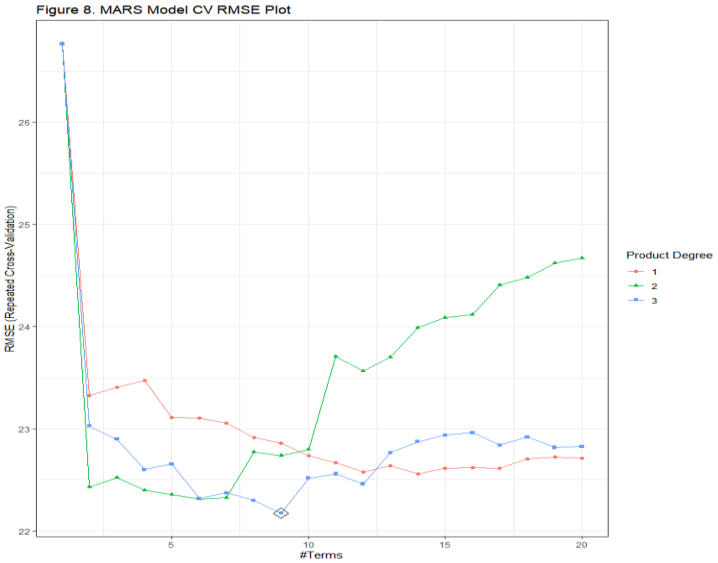
## Figure 6



## Figure 7



## Figure 8



## Figure 9.1

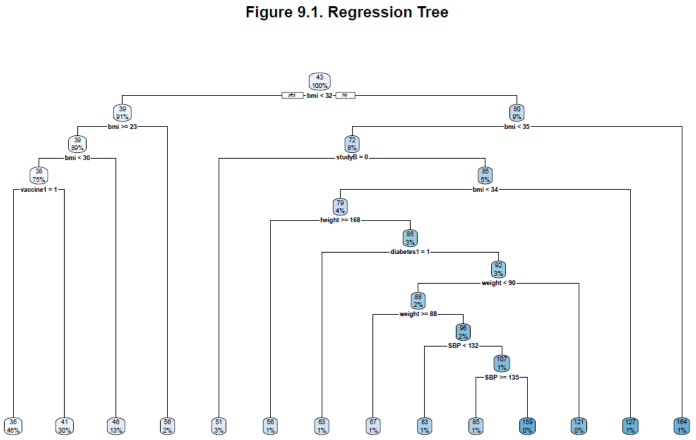


Figure 9.1. Regression Tree

## Figure 9.2

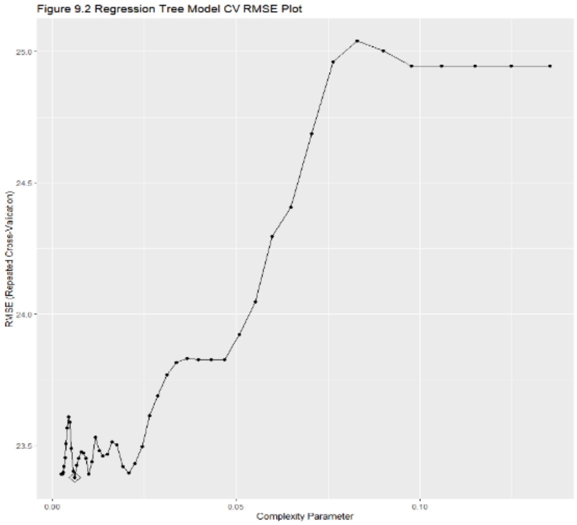


Figure 9.2 Regression Tree Model CV RMSE Plot

## Figure 10

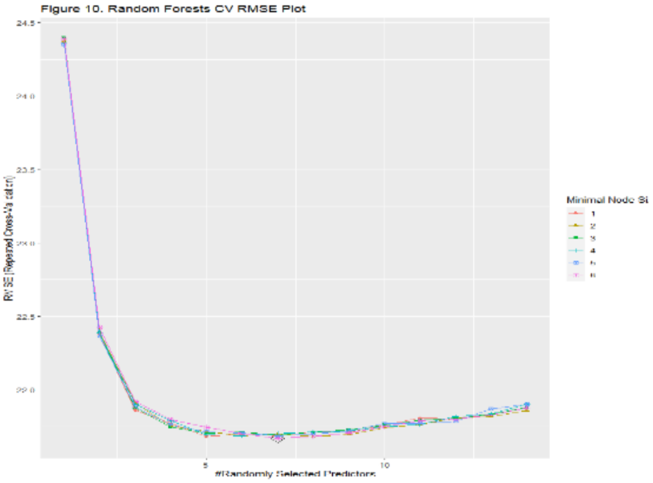


Figure 10. Random Forests CV RMSE Plot

## Figure 11

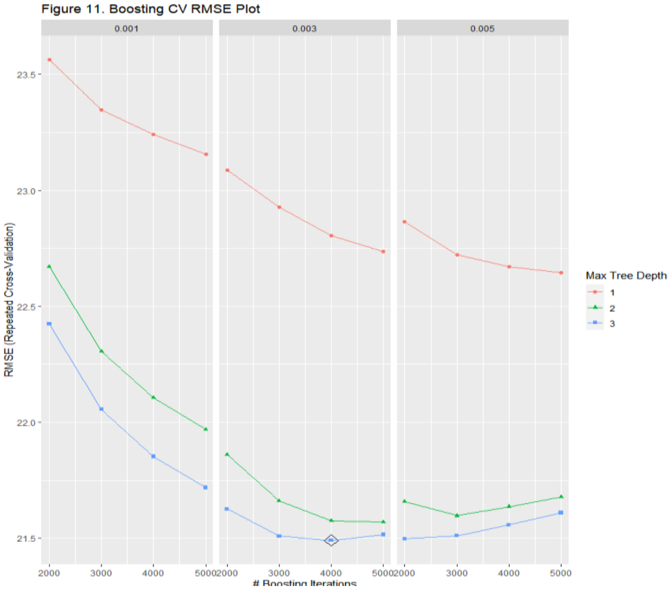
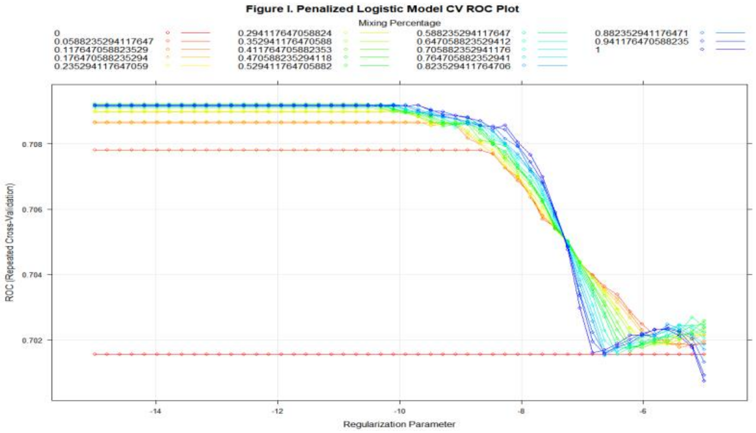
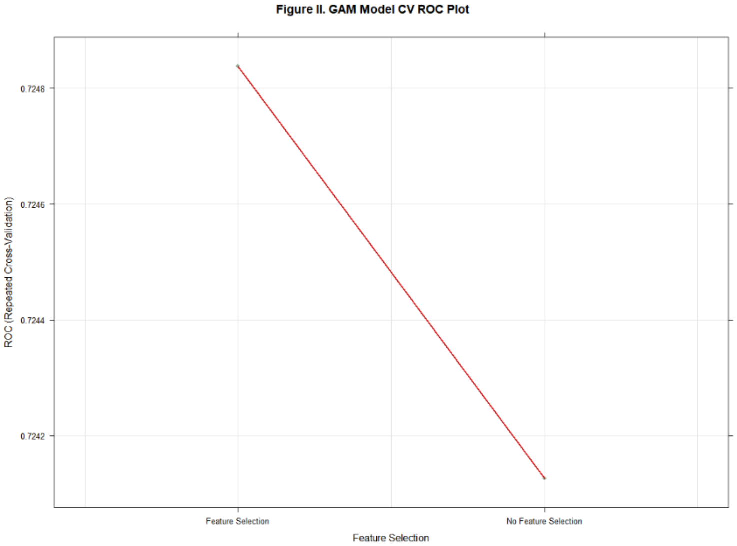


Figure 11. Boosting CV RMSE Plot

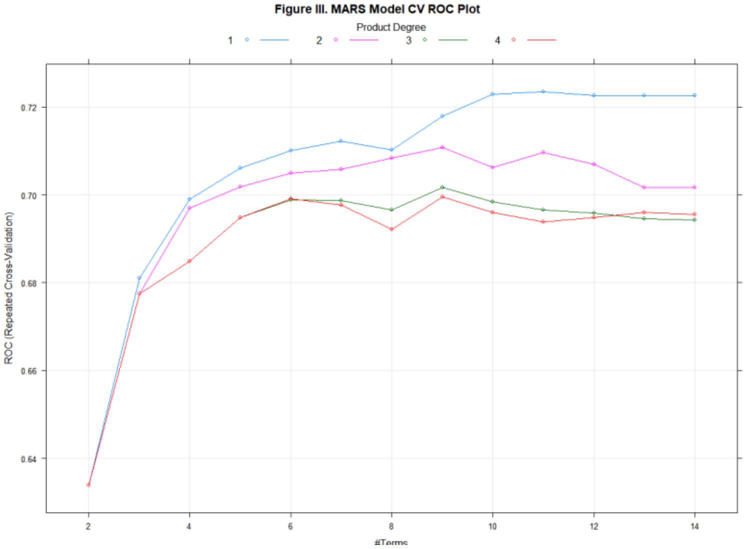
## Figure I



## Figure II



## Figure III



## Figure IV

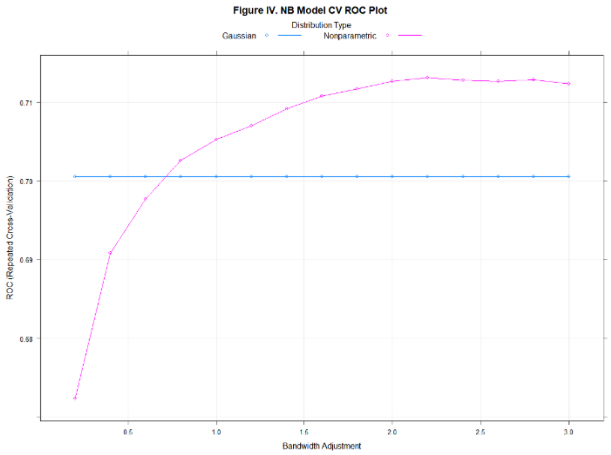


Figure IV. NB Model CV ROC Plot

## Figure V(I)

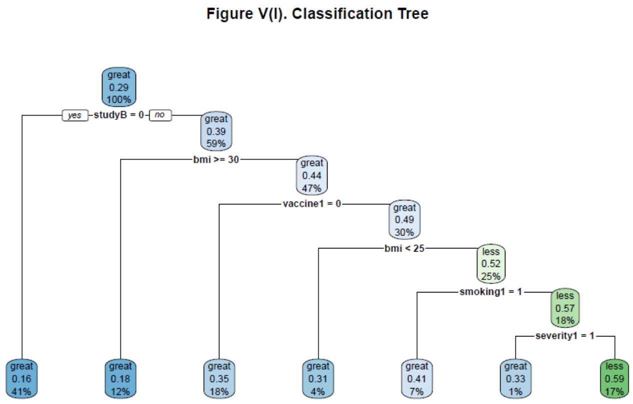


Figure V(I). Classification Tree

## Figure V(II)

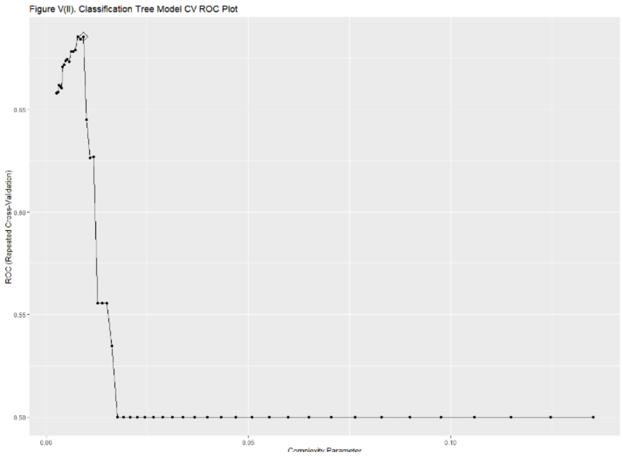


Figure V(II). Classification Tree Model CV ROC Plot

## Figure VI(I)

## Figure VI(II)

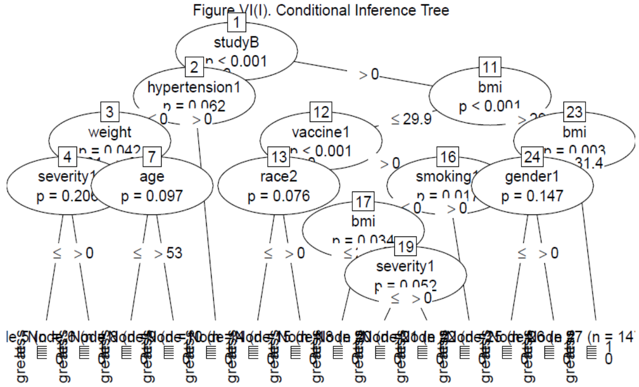


Figure VI(I). Conditional Inference Tree

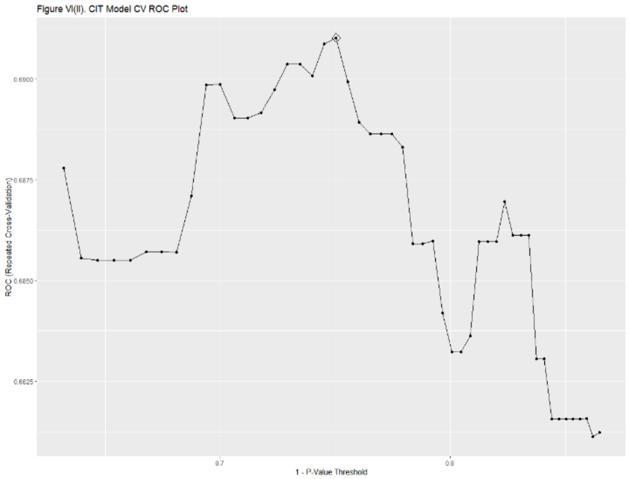
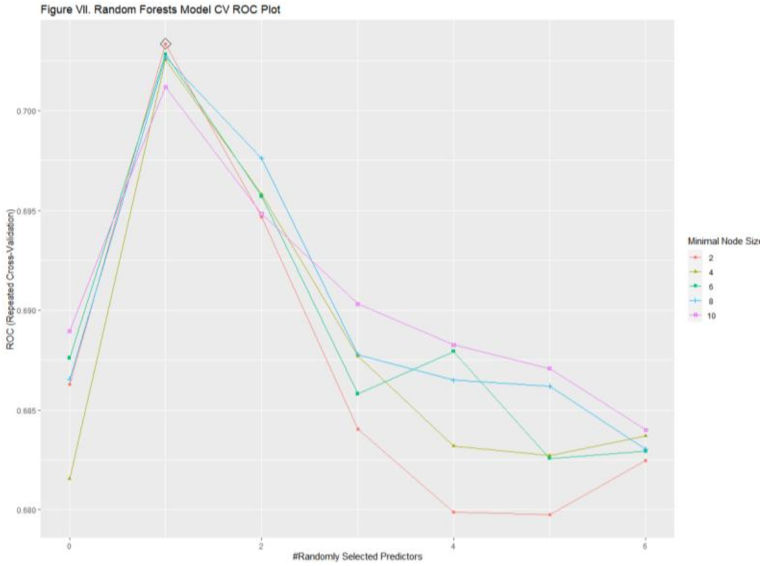
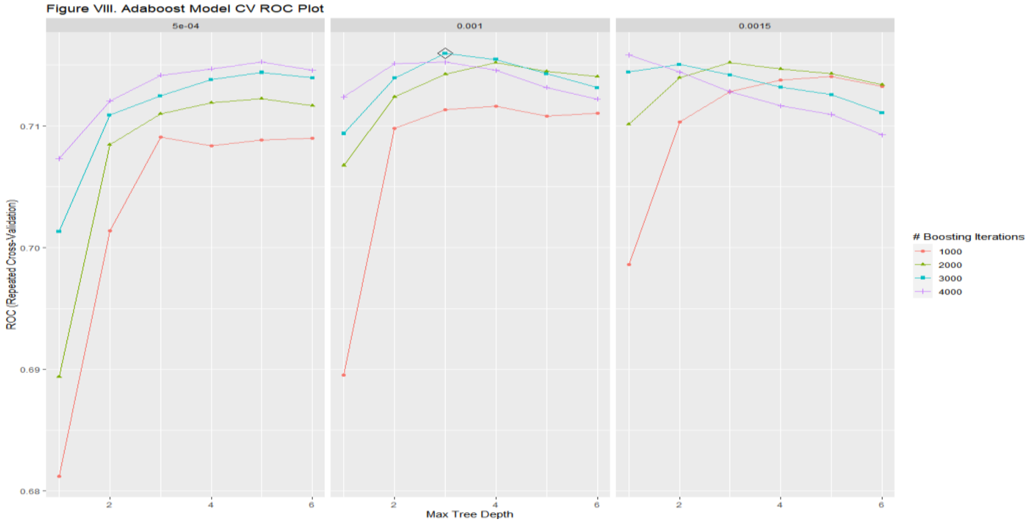


Figure VI(II). CIT Model CV ROC Plot

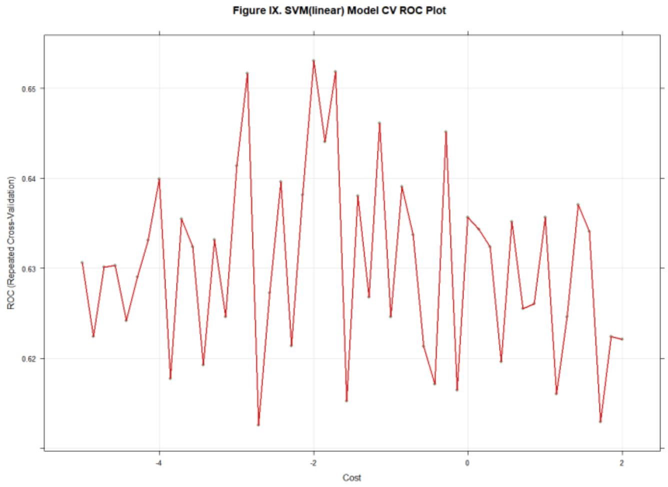
## Figure VII



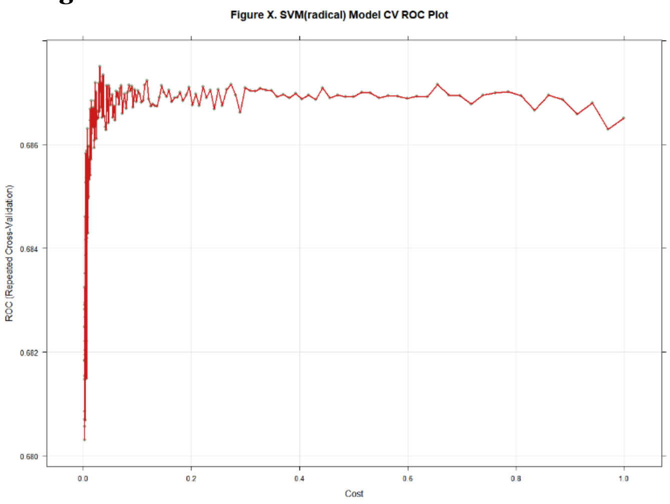
## Figure VIII



## Figure IX



## Figure X



### Figure 10.2

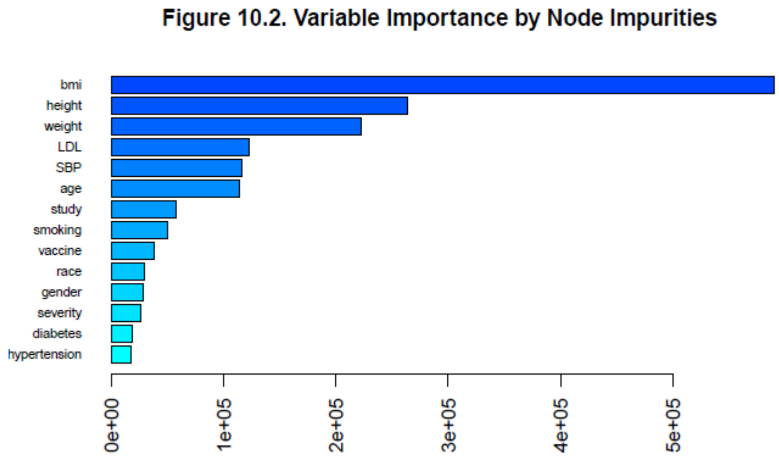


Figure 10.2. Variable Importance by Node Impurities
bmi
height
weight
LDL
SBP
age
study
smoking
vaccine
race
gender
severity
diabetes
hypertension
0e+00
1e+05
2e+05
3e+05
4e+05
5e+05