Final project

Kindle Zhang

05/09/2024

1 Exploratory analysis and data visualization

1.1 Summary

The original dataset contains COVID-19-related information for 800 participants in a training group and 200 in a testing group. I delete the variable id before all model training. The subsequent exploratory analyses are carried out utilizing the training dataset.

The dataset contains 13 predictors: 7 of them are numeric variables, and 6 of them are categorical variables. The severity is the response which is also a categorical variable. The summary of the training data is shown in **picture** 4 containing variable names and the relevant summary statistics. **picture** 5 shows the general situation.

1.2 Continuous variable

Here is the histogram plot of all the 7 continuous variables: **picture** 6.

1.3 Categorical variable

Here is the histogram plot of all the 7 categorical variables: **picture** 7

1.4 Correlation

The correlation plot generated using the corrplot( ) function is presented in picture 1. I eliminated categorical variables here as only continuous variables are meaningful to examine multicollinearity. The variable bmi, as expected, is mediumly correlated with variables height and weight, and age is slightly correlated with sbp. However, no highly correlated variable pairs were detected, leading us to conclude that multicollinearity is not a concern within this dataset.

2 Model training

2.1 Choosing Model

I will use these models to predict the severity of COVID-19 in patients, including linear models (such as generalized linear regression, elastic net), nonlinear models (such as multivariate adaptive regression splines, generalized additive models), kernel-based models (such as support vector machines), probabilistic models (such as naive Bayes, linear discriminant analysis), and tree models (such as decision trees, random forests). Each of these models has unique strengths and applicability, and I will compare and analyze their performance.

2.2 Linear Models

2.2.1 Glm (Generalized Linear Model)

Glm models the relationship between predictor variables and the response using a linear function. It's useful for cases where linear relationships are evident. In this case, We can get every variable's importance from the **picture** 8.

2.2.2 Glmnet (Elastic Net)

It combines L1 (Lasso) and L2 (Ridge) regularization in a linear model. It can perform variable selection and control model complexity during fitting, suitable for high-dimensional data and col-linearity. According to the **picture** 9, we can know the best parameter here is alpha = 0.3 and lambda = 0.06155. You can get the coefficients in my appendix.

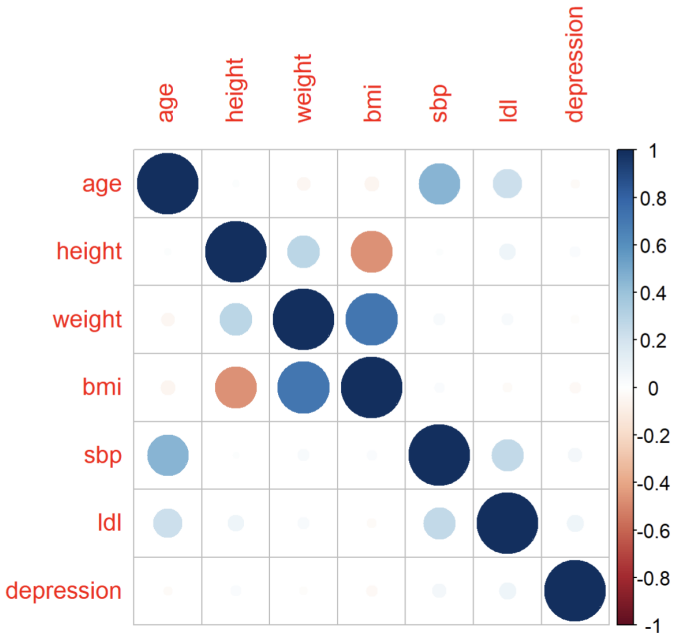


Figure 1: Correleation

2.3 non-linear Models

2.3.1 MARS (Multivariate Adaptive Regression Splines)

It approximates nonlinear relationships using local polynomials, adaptable to complex nonlinear relationships and interactions. According to the **picture** 10, we can know the best parameter here is 6 and 3. The pd plot and importance picture are as follow: **picture** 11 and **picture** 12.

2.3.2 GAM (Generalized Additive Model)

It allows modeling of nonlinear relationships by smoothing functions for each predictor variable. It's useful for exploring nonlinear trends and relationships. The **picture** 13 shows the result plot regarding predictor **depression**.

2.4 Probabilistic Models

2.4.1 LDA (Linear Discriminant Analysis)

It's a linear classifier based on Bayesian theory that classifies by estimating class prior probabilities and conditional probabilities.

2.4.2 QDA (Quadratic Discriminant Analysis)

It's similar to LDA but assumes different covariance matrices for each class, hence more flexible and capable of capturing nonlinear relationships

2.4.3 NB (Naive Bayes)

It's a simple and fast classifier based on Bayes' theorem and the assumption of conditional independence between features. It's suitable for high-dimensional data and text classification. Here is a result plot **picture** 14.

2.5 Tree Models

2.5.1 Classification tree (Recursive Partitioning)

It constructs decision trees using recursive partitioning, suitable for exploring complex interactions in data. Here is the result plot **picture** 15. The best cp is 0.0007526.

2.5.2 Classification tree with one SE

It's a decision tree model based on rpart, uses 1 standard error as the pruning parameter during model selection to avoid over-fitting. Here is the result plot **picture** 16. The best cp is 0.0285655.

2.5.3 compare two rpart

I compare two methods' classification results and find that the method with one se is more simple in classifying data. Here is the results **picture** 17 and **picture** 18

2.5.4 Random Forest

It predicts by constructing multiple decision trees, each based on different subsets of data and randomly selected features. It has high prediction accuracy and robustness, suitable for high-dimensional data and complex relationships. The results ROC with different parameter is as follows: **picture** 19. The best tune is 3 randomly selected predictors and 10 minimal node size.

2.5.5 Adaboosting

It's an ensemble learning method that iteratively trains a series of weak classifiers (usually decision trees) and adjusts sample weights based on their performance to obtain a strong classifier. It's suitable for handling complex nonlinear relationships. The results ROC with different parameter is as follows: **picture** 20. The best tune is 2000 trees, 2 interaction depth, 0.002 shrinkage.

2.6 Kernel-based Models

2.6.1 SVML (Support Vector Machines with Linear Kernel)

It uses a linear kernel function in support vector machine modeling, suitable for linearly separable or approximately separable cases. The result ROC with different cost is in **picture** 21. The best tune is 0.002745041 in cost.

2.6.2 SVMR (Support Vector Machines with Radial Kernel)

It uses a radial basis function as the kernel function in support vector machine modeling, capable of handling nonlinear relationships and has strong generalization ability. The result ROC with different cost is in **picture** 22. The best tune is 0.001464391 in sigma and 64.87185 in cost.

3 Result

3.1 Cv performance

After simulating patients with all models, I want to evaluate the fitting performance of each model through resampling and identify the one with the highest ROC. The result **picture** 2 is as follows:

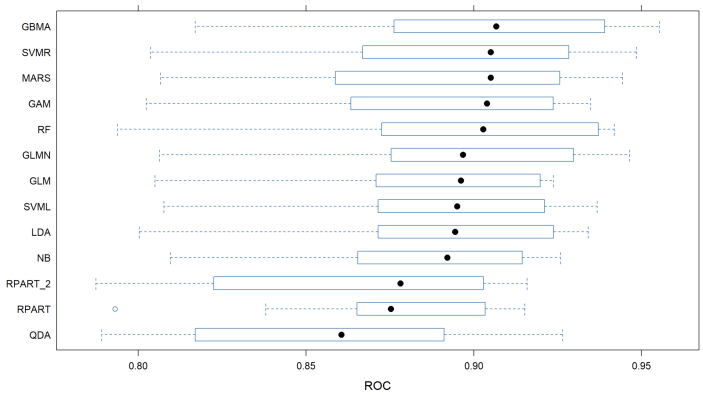


Figure 2: compare CV performance

According to the result, the method **Adaboosting** has the best performance. This is the summary of the result(the 7 most important variables)

Here is also a pdp plot which regarding the predictor **sbp**: **picture** 23

3.2 Predict performance

Now let's look at the test data performance.

We predict with new testing data and get their roc and auc, the picture shows the result: **picture** 3

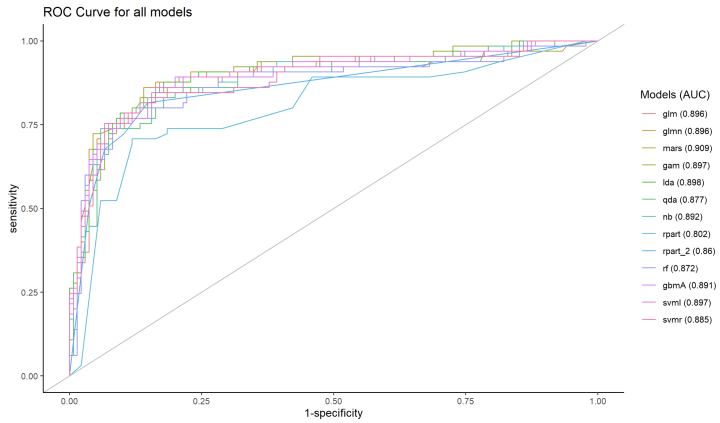


Figure 3: ROC and AUC in all models

We can find that the **MARS** model has the best performance in predicting new data. The coefficients are as follows:

Coefficient	Value
(Intercept)	2.338
vaccine	-3.229
h(bmi-27.2)	0.261
h(139-sbp)	-0.168
h(124-sbp) * vaccine	0.210
h(sbp-124) * vaccine	-0.055

Table 1: MARS Coefficients

The ROC curve is as follow: **picture**24:

The table shows the confusion matrix result and we can calculate the misclassification error rate is 1 - 0.87 = 0.13.

3.3 Conclusion

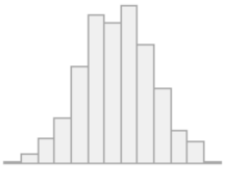
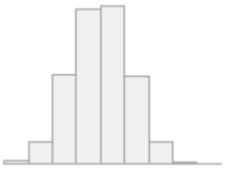
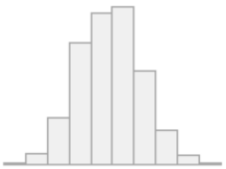
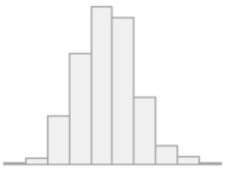
As mentioned earlier, when we focus on the goodness of fit of the model, using the AdaBoosting model is better, whereas when we prioritize practical prediction applications, MARS performs better.

$$\text{logit}(\text{severity}) = \begin{cases} 2.33844275 - 3.22940600 \times \text{vaccine} & \text{if vaccine is active} \\ 2.33844275 - 0.16830784 \times \text{h}(139 - \text{sbp}) & \text{if h}(139 - \text{sbp}) \text{ is acti} \\ 2.33844275 - 0.05526456 \times \text{h}(sbp - 124) \times \text{vaccine} + 0.21038093 \times \text{h}(124 - \text{sbp}) \times \text{vaccine} & \text{if both h}(sbp - 124) \text{ a} \\ 2.33844275 + 0.26113285 \times \text{h}(bmi - 27.2) & \text{if h}(bmi - 27.2) \text{ is act} \end{cases}$$

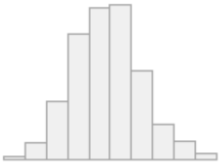
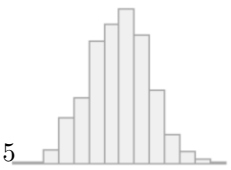
We can use this formula to predict the patient's severity.

4 Appendix

4.1 Plots and Tables

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	text.graph	Valid	Missing
1	age [numeric]	Mean (sd) : 60 (4.3) min < med < max: 46 < 60 < 72 IQR (CV) : 6 (0.1)	26 distinct values		: . : . : : : : : : : : : . . : : : : : : :	800 (100.0%)	0 (0.0%)
2	gender [factor]	1. 0 2. 1	\410 (51.2%) \390 (48.8%)		IIIIIIIIII IIIIIIIII	800 (100.0%)	0 (0.0%)
3	race [factor]	1. 1 2. 2 3. 3 4. 4	\521 (65.1%) 50 (6.2%) \149 (18.6%) 80 (10.0%)		IIIIIIIIIIII I III II	800 (100.0%)	0 (0.0%)
4	smoking [factor]	1. 0 2. 1 3. 2	\467 (58.4%) \248 (31.0%) 85 (10.6%)		IIIIIIIIII IIIIII II	800 (100.0%)	0 (0.0%)
5	height [numeric]	Mean (sd) : 170 (6.1) min < med < max: 150.2 < 170 < 190.3 IQR (CV) : 8.4 (0)	248 distinct values		: : : : : : : : : : : : . : : : .	800 (100.0%)	0 (0.0%)
6	weight [numeric]	Mean (sd) : 79.4 (7.3) min < med < max: 56.6 < 79.3 < 104.8 IQR (CV) : 10 (0.1)	280 distinct values		. : : : : : : : : : . : : : : . : : : : : : .	800 (100.0%)	0 (0.0%)
7	bmi [numeric]	Mean (sd) : 27.5 (2.7) min < med < max: 19.6 < 27.6 < 37.4 IQR (CV) : 3.3 (0.1)	133 distinct values		: . : : : : : : : : : . : : : : : .	800 (100.0%)	0 (0.0%)

0.45

8	diabetes [factor]	1. 0 2. 1	\679 (84.9%) \121 (15.1%)		IIIIIIIIIIIIIII III	800 (100.0%)	0 (0.0%)
9	hypertension [factor]	1. 0 2. 1	\432 (54.0%) \368 (46.0%)		IIIIIIIIII IIIIIIIII	800 (100.0%)	0 (0.0%)
10	sbp [numeric]	Mean (sd) : 129.8 (8) min < med < max: 109 < 130 < 154 IQR (CV) : 11 (0.1)	46 distinct values		. . : : : : . : : : . . : : : : : . : : : : : : : . .	800 (100.0%)	0 (0.0%)
11	ldl [numeric]	Mean (sd) : 110.3 (20.1) min < med < max: 41 < 111 < 174 IQR (CV) : 25 (0.2)	103 distinct values		. : : : : . : : : . : : : : . : : : : : : .	800 (100.0%)	0 (0.0%)

	Reference: not_severe	Reference: severe
Prediction: not_severe	127	18
Prediction: severe	8	47
Accuracy	0.87	
95% CI	(0.8153, 0.9133)	
No Information Rate	0.675	
P-Value [Acc ¿ NIR]	1.77e-10	
Kappa	0.6914	
McNemar's Test P-Value	0.07756	
Sensitivity	0.7231	
Specificity	0.9407	
Pos Pred Value	0.8545	
Neg Pred Value	0.8759	
Prevalence	0.3250	
Detection Rate	0.2350	
Detection Prevalence	0.2750	
Balanced Accuracy	0.8319	
'Positive' Class	severe	

Table 2: Confusion Matrix and Statistics

Name	training_data
Number of rows	800
Number of columns	14
_____	
Column type frequency:	
factor	7
numeric	7
_____	
Group variables	None

Figure 5: Summary 2

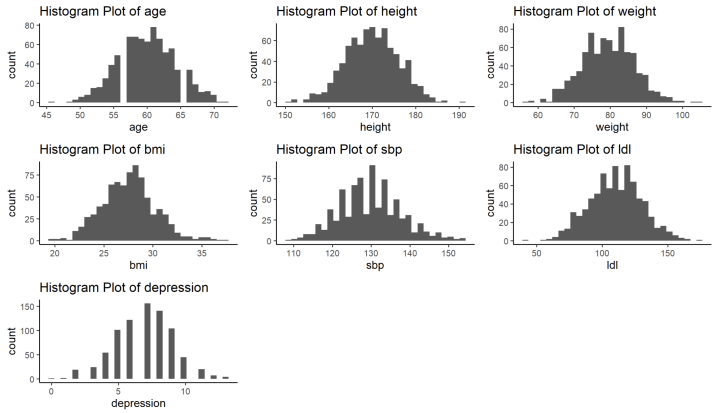


Figure 6: Continuous variable

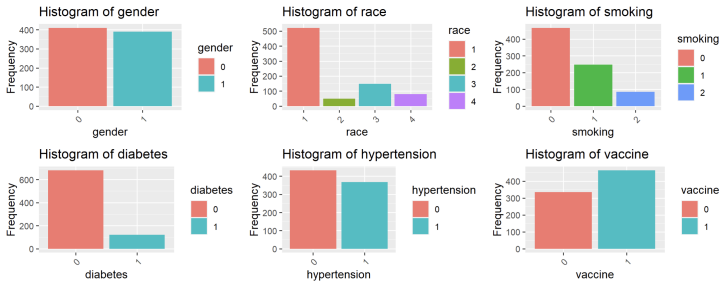


Figure 7: Categorical variable

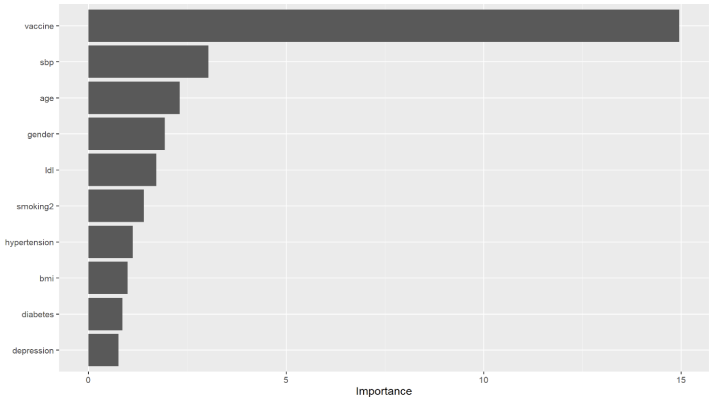


Figure 8: GLM importance

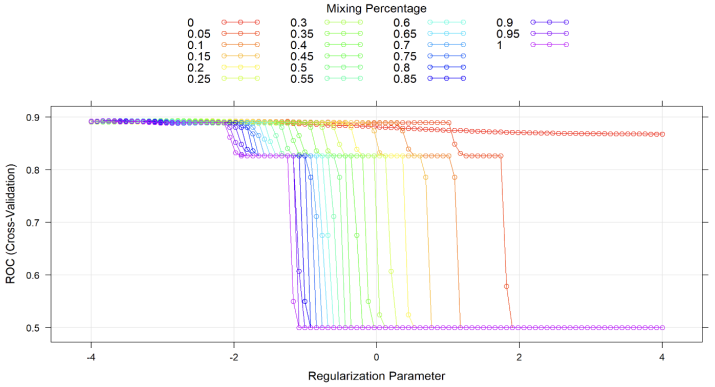


Figure 9: ROC with different parameter

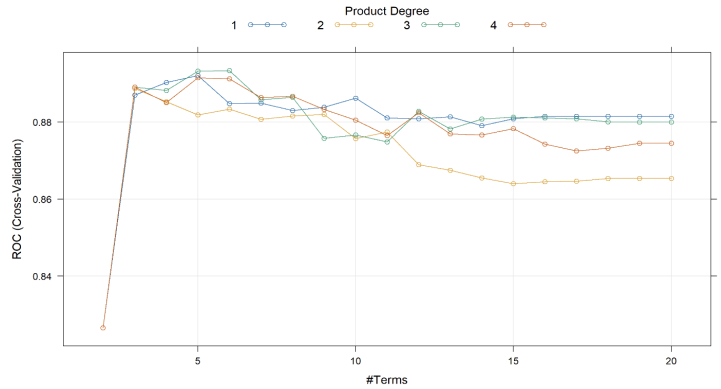


Figure 10: ROC with different parameter

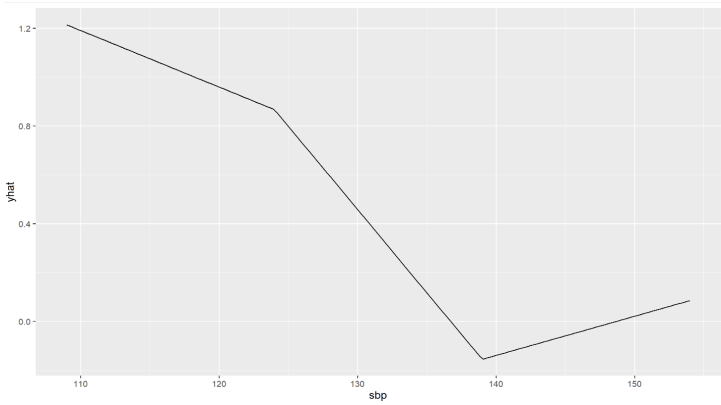


Figure 11: pdp of variable sbp

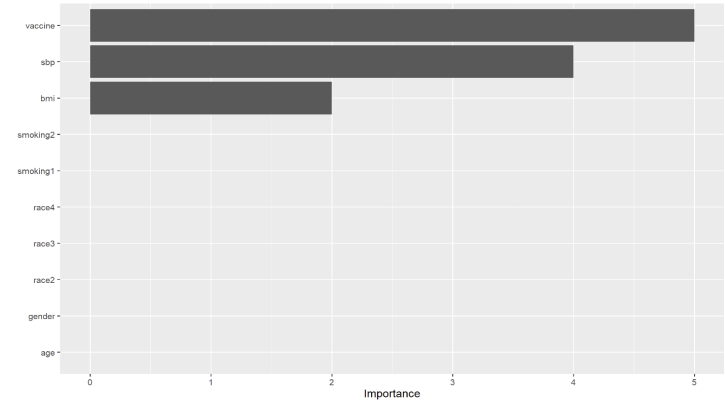


Figure 12: important variable

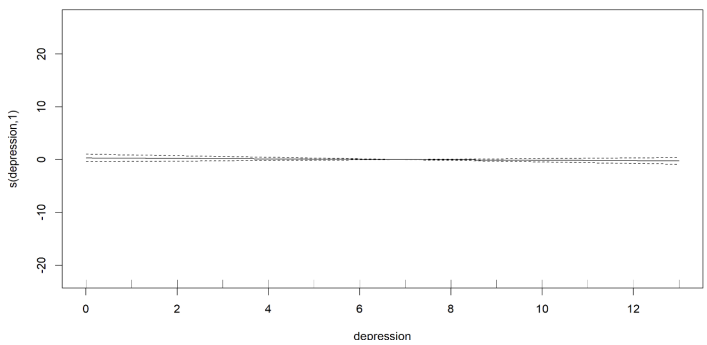


Figure 13: result plot regarding depression

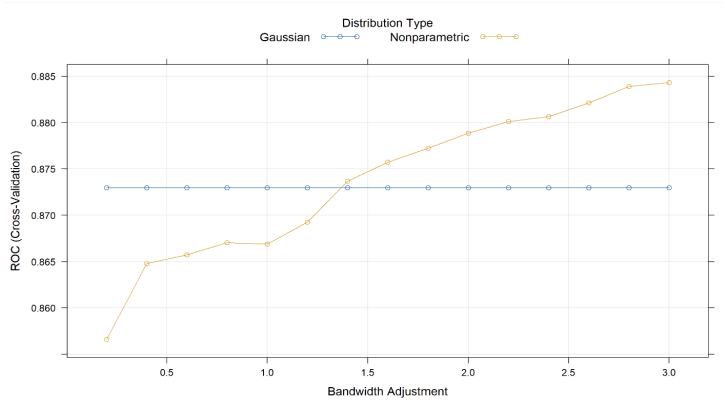


Figure 14: ROC picture in NB

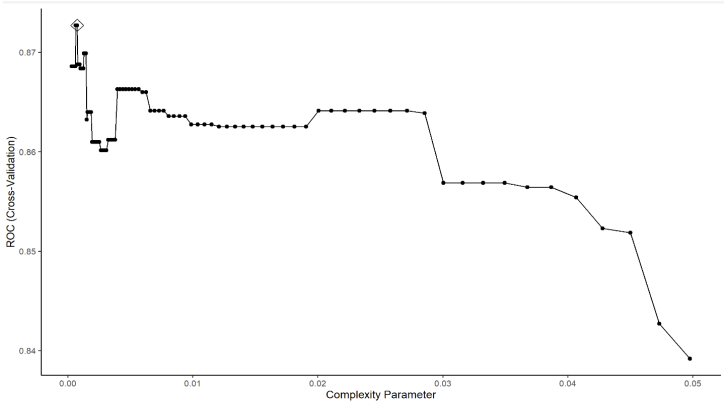


Figure 15: ROC picture in rpart

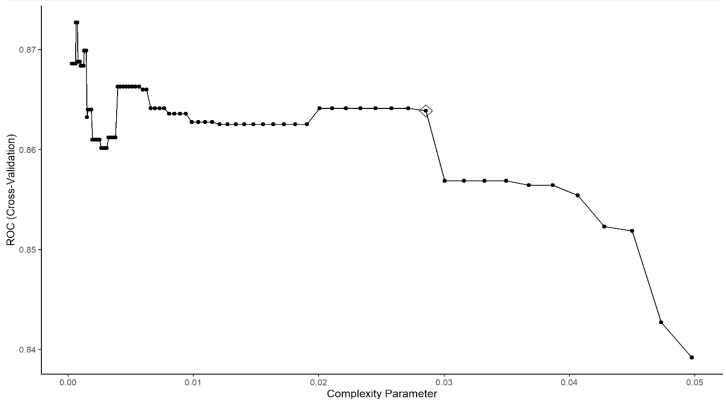


Figure 16: ROC picture in rpart with one SE

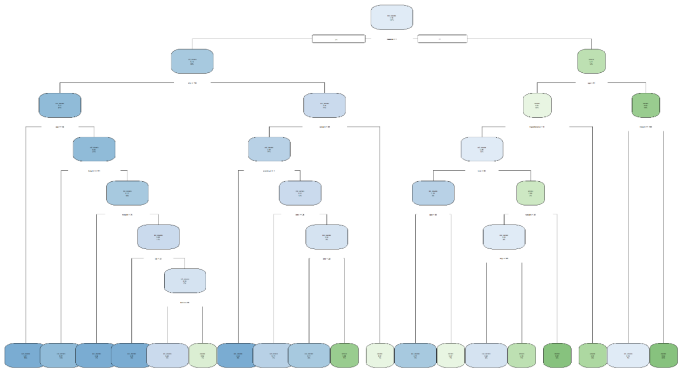


Figure 17: classification picture in rpart with one SE

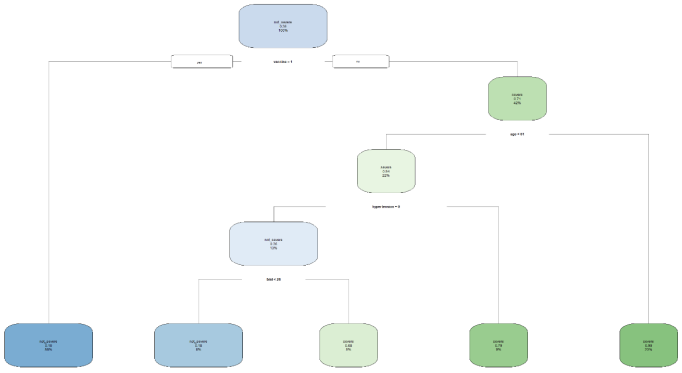


Figure 18: classification picture in rpart with one SE

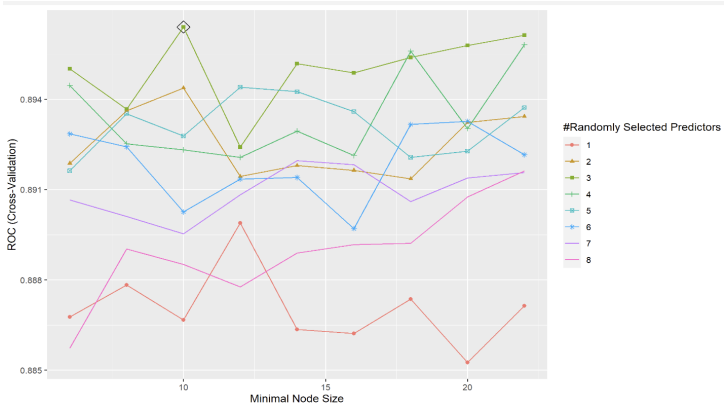


Figure 19: ROC among different parameters

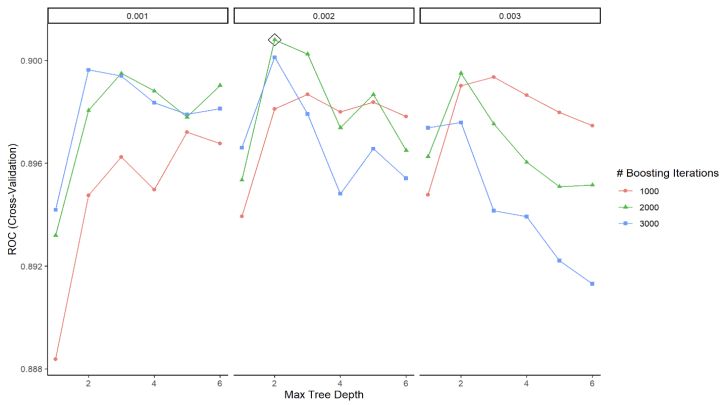


Figure 20: ROC among different parameters

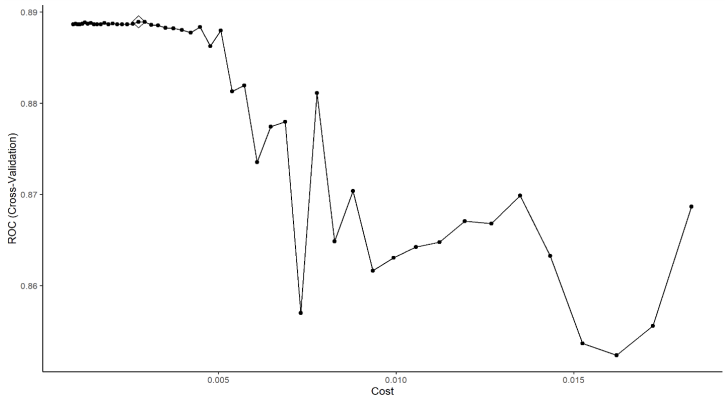


Figure 21: SVML ROC

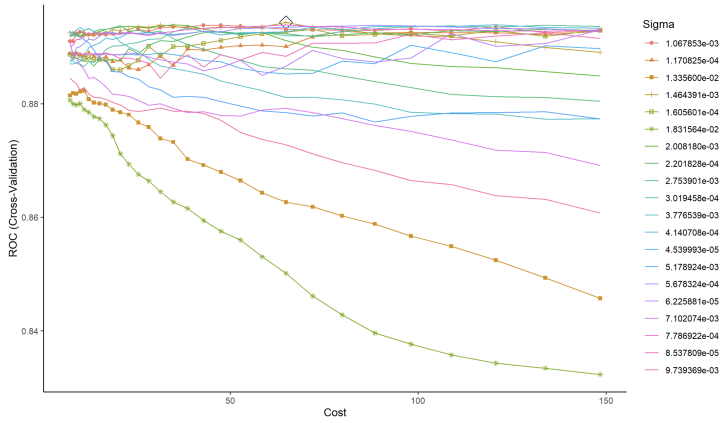


Figure 22: SVMR ROC

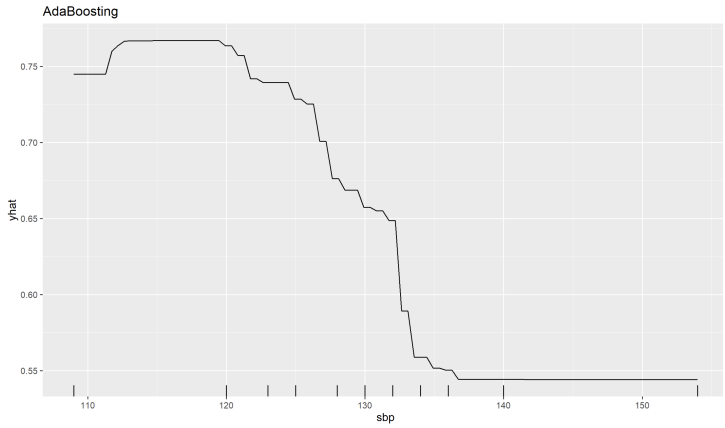


Figure 23: pdp plot regarding sbp

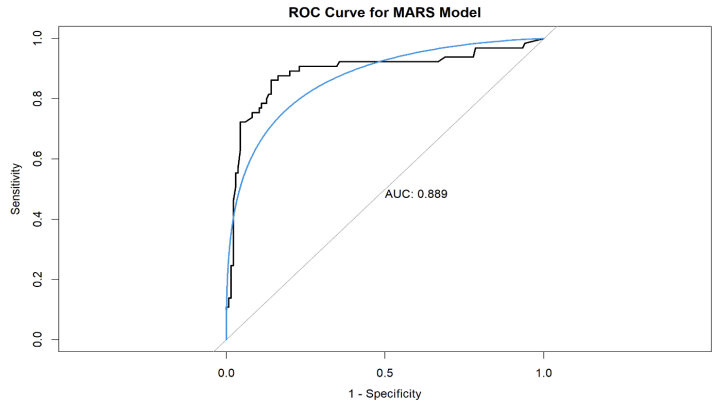


Figure 24: ROC curve in MARS