Final Project Code

Tianshu Liu, Lincole Jiang, Jiong Ma

Contents

1 Data Import **3**

2 Data partition **5**

3 Primary Analysis **5**
3.1 Exploratory analysis and data visualization 5
3.1.1 Data Frame Summary 5
3.2 Model Training 12
3.2.1 Linear Model 12
3.2.2 LASSO 12
3.2.3 Ridge 15
3.2.4 Elastic Net 17
3.2.5 Principal components regression (PCR) 19
3.2.6 Partial Least Squares (PLS) 21
3.2.7 Generalized Additive Model (GAM) 23
3.2.8 Multivariate Adaptive Regression Splines (MARS) 25
3.2.9 K-Nearest Neighbour (KNN) 27
3.2.10 Bagging 28
3.2.11 Random Forest 28
3.2.12 Boosting 28
3.2.13 Regression Trees 28
3.3 Model Selection 28
3.4 Training / Testing Error 31

4 Secondary Analysis **32**
4.1 Exploratory analysis and data visualization 32
4.1.1 Data Frame Summary 32
4.2 Model Training 36
4.2.1 Logistic Regression 36
4.2.2 Penalized Logistic Regression 36
4.2.3 Generalized Additive Model (GAM) for classification 36
4.2.4 Multivariate Adaptive Regression Splines (MARS) for classification 36
4.2.5 Linear Discriminant Analysis (LDA) 36
4.2.6 Quadratic Discriminant Analysis (QDA) 36
4.2.7 Naive Bayes (NB) 36
4.2.8 Bagging 36
4.2.9 Random Forest 36
4.2.10 Boosting 36
4.2.11 Classification Trees 36
4.2.12 Support Vector Machine (SVM) 36
4.2.13 Hierarchical Clustering 36
4.2.14 Principal Component Analysis (PCA) 36

4.3 Model Selection 36
4.4 Training / Testing Error 36

```
library(tidyverse)
library(summarytools)
library(corrplot)
library(caret)
library(vip)
```

1 Data Import

```
# import data
load("./recovery.RData")

set.seed(3196)
lts.dat <- dat[sample(1:10000, 2000),]
set.seed(2575)
lincole.dat <- dat[sample(1:10000, 2000),]
set.seed(5509)
amy.dat <- dat[sample(1:10000, 2000),]

dat1 <- lts.dat %>%
  merge(lincole.dat, all = TRUE) %>%
  na.omit() %>%
  select(-id) %>%
  mutate(
    gender = as.factor(gender),
    race = as.factor(race),
    smoking = as.factor(smoking),
    hypertension = as.factor(hypertension),
    diabetes = as.factor(diabetes),
    vaccine = as.factor(vaccine),
    severity = as.factor(severity),
    study = as.factor(study))

dat2 <- lts.dat %>%
  merge(amy.dat, all = TRUE) %>%
  na.omit() %>%
  select(-id) %>%
  mutate(
    gender = as.factor(gender),
    race = as.factor(race),
    smoking = as.factor(smoking),
    hypertension = as.factor(hypertension),
    diabetes = as.factor(diabetes),
    vaccine = as.factor(vaccine),
    severity = as.factor(severity),
    study = as.factor(study))

dat3 <- lincole.dat %>%
  merge(amy.dat, all = TRUE) %>%
  na.omit() %>%
  select(-id) %>%
  mutate(
    gender = as.factor(gender),
```

```
    race = as.factor(race),
    smoking = as.factor(smoking),
    hypertension = as.factor(hypertension),
    diabetes = as.factor(diabetes),
    vaccine = as.factor(vaccine),
    severity = as.factor(severity),
    study = as.factor(study))

dat <- dat1
summary(dat)
```

```
##       age          gender    race     smoking      height          weight
##  Min.   :45.00   0:1842   1:2372   0:2223   Min.   :151.2   Min.   : 56.70
##  1st Qu.:57.00   1:1781   2: 172   1:1034   1st Qu.:166.2   1st Qu.: 75.40
##  Median :60.00            3: 716   2: 366   Median :170.2   Median : 80.20
##  Mean   :60.06            4: 363            Mean   :170.2   Mean   : 80.13
##  3rd Qu.:63.00                              3rd Qu.:174.2   3rd Qu.: 84.80
##  Max.   :77.00                              Max.   :188.6   Max.   :103.40
##       bmi         hypertension diabetes      SBP             LDL         vaccine
##  Min.   :19.70   0:1891       0:3065   Min.   :102.0   Min.   : 28.0   0:1469
##  1st Qu.:25.80   1:1732       1: 558   1st Qu.:125.0   1st Qu.: 97.0   1:2154
##  Median :27.60                         Median :130.0   Median :110.0
##  Mean   :27.73                         Mean   :130.2   Mean   :110.5
##  3rd Qu.:29.40                         3rd Qu.:136.0   3rd Qu.:124.0
##  Max.   :39.80                         Max.   :158.0   Max.   :174.0
##  severity study    recovery_time
##  0:3289   A: 728   Min.   :  3.00
##  1: 334   B:2171   1st Qu.: 28.00
##           C: 724   Median : 38.00
##                    Mean   : 42.87
##                    3rd Qu.: 49.00
##                    Max.   :365.00
```

```
bin.dat1 <- dat1 %>%
  mutate(recovery_time = ifelse(recovery_time > 30, ">30", "<=30")) %>%
  mutate(recovery_time = factor(recovery_time, levels = c("<=30", ">30")))

bin.dat2 <- dat2 %>%
  mutate(recovery_time = ifelse(recovery_time > 30, ">30", "<=30")) %>%
  mutate(recovery_time = factor(recovery_time, levels = c("<=30", ">30")))

bin.dat3 <- dat3 %>%
  mutate(recovery_time = ifelse(recovery_time > 30, ">30", "<=30")) %>%
  mutate(recovery_time = factor(recovery_time, levels = c("<=30", ">30")))

bin.dat <- bin.dat1
summary(bin.dat)
```

```
##       age          gender    race     smoking      height          weight
##  Min.   :45.00   0:1842   1:2372   0:2223   Min.   :151.2   Min.   : 56.70
##  1st Qu.:57.00   1:1781   2: 172   1:1034   1st Qu.:166.2   1st Qu.: 75.40
##  Median :60.00            3: 716   2: 366   Median :170.2   Median : 80.20
##  Mean   :60.06            4: 363            Mean   :170.2   Mean   : 80.13
##  3rd Qu.:63.00                              3rd Qu.:174.2   3rd Qu.: 84.80
```

```
##  Max.   :77.00                              Max.   :188.6   Max.   :103.40
##       bmi         hypertension diabetes      SBP             LDL         vaccine
##  Min.   :19.70   0:1891       0:3065   Min.   :102.0   Min.   : 28.0   0:1469
##  1st Qu.:25.80   1:1732       1: 558   1st Qu.:125.0   1st Qu.: 97.0   1:2154
##  Median :27.60                         Median :130.0   Median :110.0
##  Mean   :27.73                         Mean   :130.2   Mean   :110.5
##  3rd Qu.:29.40                         3rd Qu.:136.0   3rd Qu.:124.0
##  Max.   :39.80                         Max.   :158.0   Max.   :174.0
##  severity study    recovery_time
##  0:3289   A: 728   <=30:1102
##  1: 334   B:2171   >30 :2521
##           C: 724
##
##
##
```

2 Data partition

```
# data partition
dat.matrix <- model.matrix(recovery_time ~ ., dat)[ ,-1]

set.seed(2023)
trainRows <- createDataPartition(y = dat$recovery_time, p = 0.8, list = FALSE)

train.dat <- dat[trainRows,]
train.bin.dat <- bin.dat[trainRows,]

train.x <- dat.matrix[trainRows,]
train.y <- dat$recovery_time[trainRows]
train.bin.y <- bin.dat$recovery_time[trainRows]

test.x <- dat.matrix[-trainRows,]
test.y <- dat$recovery_time[-trainRows]
test.bin.y <- bin.dat$recovery_time[-trainRows]
```

3 Primary Analysis

3.1 Exploratory analysis and data visualization

```
# data summary
st_options(plain.ascii = FALSE,
           style = "rmarkdown",
           dfSummary.silent = TRUE,
           footnote = NA,
           subtitle.emphasis = FALSE)
dfSummary(train.dat)
```

3.1.1 Data Frame Summary

train.dat
Dimensions: 2900 x 15
Duplicates: 0

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	age [numeric]	Mean (sd) : 60.1 (4.5) min < med < max: 45 < 60 < 77 IQR (CV) : 6 (0.1)	33 distinct values	: : . : : : : : . . : : : : : .	2900 (100.0%)	0 (0.0%)
2	gender [factor]	1. 0 2. 1	1468 (50.6%) 1432 (49.4%)	IIIIIIIIII IIIIIIIII	2900 (100.0%)	0 (0.0%)
3	race [factor]	1. 1 2. 2 3. 3 4. 4	1909 (65.8%) 132 (4.6%) 568 (19.6%) 291 (10.0%)	IIIIIIIIIIIII III II	2900 (100.0%)	0 (0.0%)
4	smoking [factor]	1. 0 2. 1 3. 2	1763 (60.8%) 845 (29.1%) 292 (10.1%)	IIIIIIIIIIII IIIII II	2900 (100.0%)	0 (0.0%)
5	height [numeric]	Mean (sd) : 170.2 (6) min < med < max: 151.2 < 170.1 < 188.6 IQR (CV) : 8 (0)	312 distinct values	: : : : . : : : . : : : : . : : : : : .	2900 (100.0%)	0 (0.0%)
6	weight [numeric]	Mean (sd) : 80.2 (7) min < med < max: 57.1 < 80.3 < 103.4 IQR (CV) : 9.5 (0.1)	361 distinct values	. : . : : : : : : . : : : : . . : : : : : : .	2900 (100.0%)	0 (0.0%)
7	bmi [numeric]	Mean (sd) : 27.8 (2.7) min < med < max: 19.7 < 27.7 < 39.8 IQR (CV) : 3.6 (0.1)	160 distinct values	: . : : : : : . : : : : : : : : : : .	2900 (100.0%)	0 (0.0%)
8	hypertension [factor]	1. 0 2. 1	1514 (52.2%) 1386 (47.8%)	IIIIIIIIII IIIIIIIII	2900 (100.0%)	0 (0.0%)
9	diabetes [factor]	1. 0 2. 1	2446 (84.3%) 454 (15.7%)	IIIIIIIIIIIIIIII III	2900 (100.0%)	0 (0.0%)
10	SBP [numeric]	Mean (sd) : 130.2 (8.1) min < med < max: 104 < 130 < 158 IQR (CV) : 11 (0.1)	54 distinct values	: : . : : : . . : : : : . : : : : : :	2900 (100.0%)	0 (0.0%)
11	LDL [numeric]	Mean (sd) : 110.3 (19.9) min < med < max: 32 < 110 < 174 IQR (CV) : 27 (0.2)	116 distinct values	. : : : : : : : . : : : : : . : : : : : : .	2900 (100.0%)	0 (0.0%)
12	vaccine [factor]	1. 0 2. 1	1192 (41.1%) 1708 (58.9%)	IIIIIIII IIIIIIIIIII	2900 (100.0%)	0 (0.0%)
13	severity [factor]	1. 0 2. 1	2619 (90.3%) 281 (9.7%)	IIIIIIIIIIIIIIIIII I	2900 (100.0%)	0 (0.0%)
14	study [factor]	1. A 2. B 3. C	580 (20.0%) 1750 (60.3%) 570 (19.7%)	IIII IIIIIIIIIIII III	2900 (100.0%)	0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
15	recovery_time [numeric]	Mean (sd) : 43 (30.5) min < med < max: 3 < 38 < 365 IQR (CV) : 21 (0.7)	144 distinct values	: : : : : : : : : .	2900 (100.0%)	0 (0.0%)

```
skimr::skim_without_charts(train.dat)
```

Table 2: Data summary

Name	train.dat
Number of rows	2900
Number of columns	15
Column type frequency:	
factor	8
numeric	7
Group variables	None

Variable type: factor

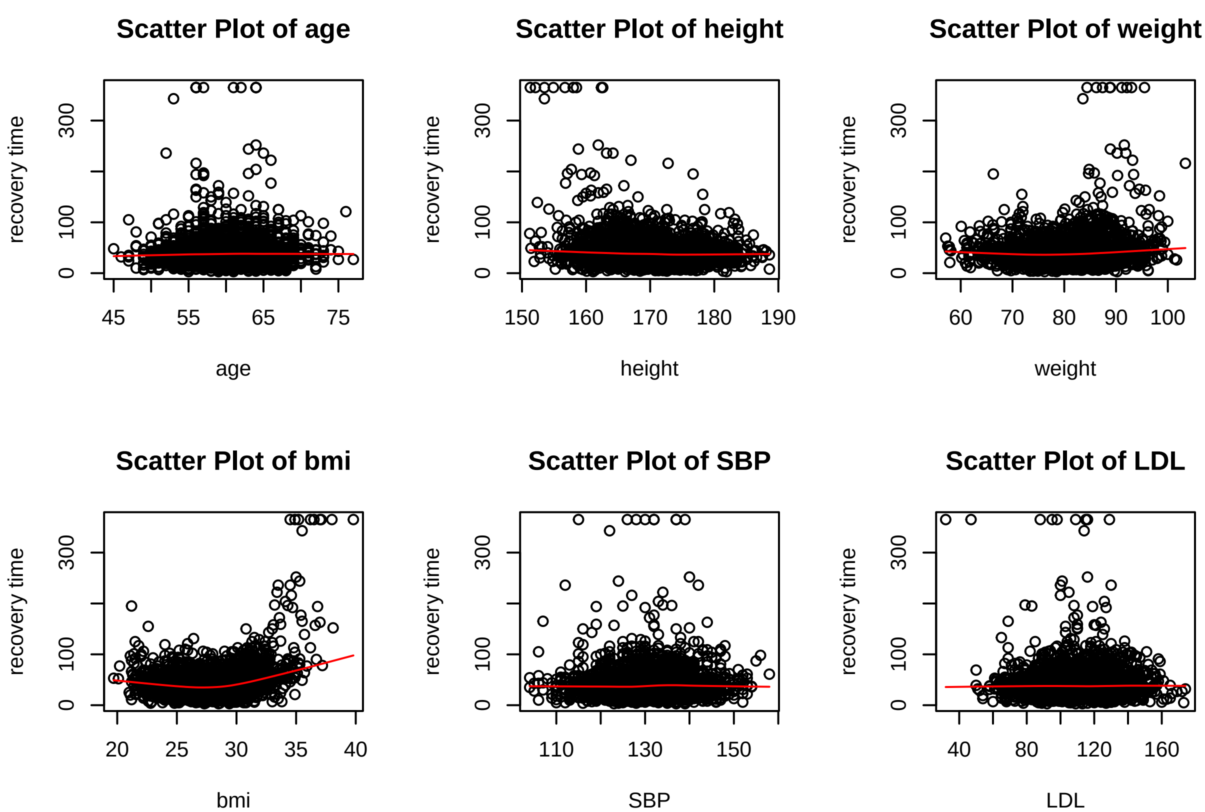
skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
gender	0	1	FALSE	2	0: 1468, 1: 1432
race	0	1	FALSE	4	1: 1909, 3: 568, 4: 291, 2: 132
smoking	0	1	FALSE	3	0: 1763, 1: 845, 2: 292
hypertension	0	1	FALSE	2	0: 1514, 1: 1386
diabetes	0	1	FALSE	2	0: 2446, 1: 454
vaccine	0	1	FALSE	2	1: 1708, 0: 1192
severity	0	1	FALSE	2	0: 2619, 1: 281
study	0	1	FALSE	3	B: 1750, A: 580, C: 570

Variable type: numeric

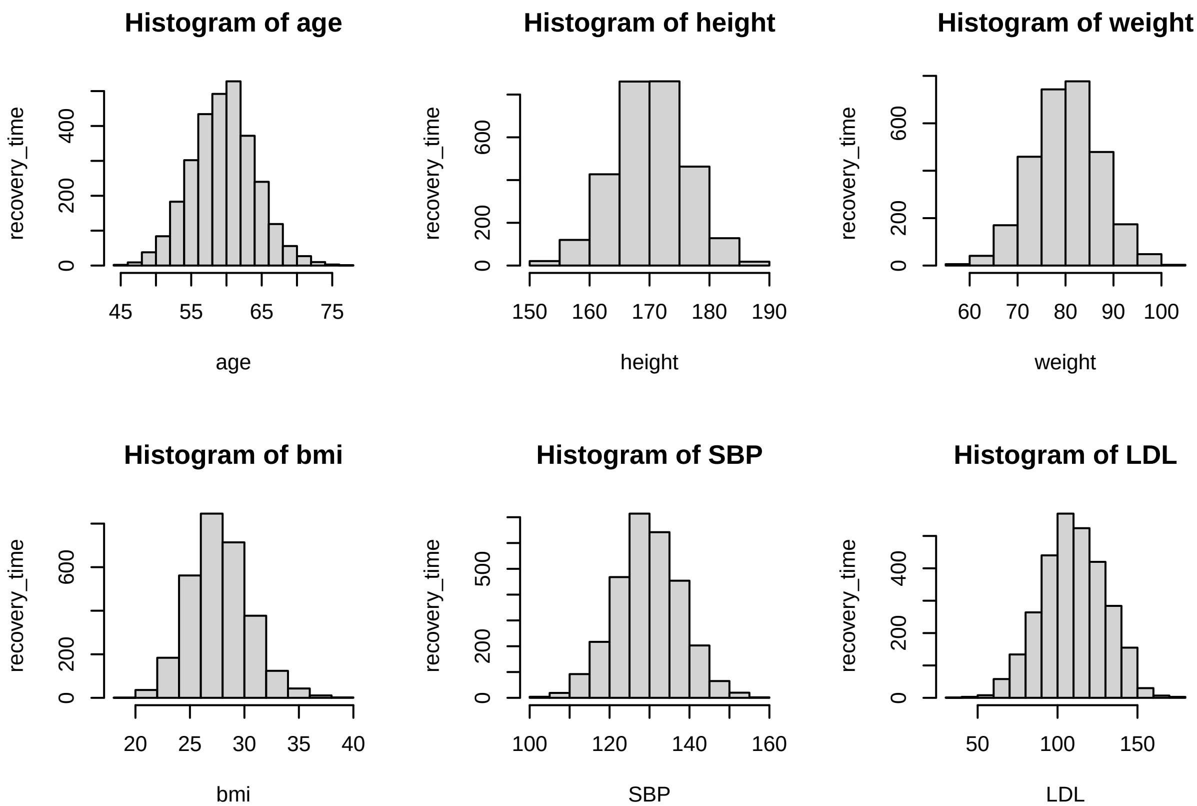
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
age	0	1	60.07	4.51	45.0	57.0	60.00	63.0	77.0
height	0	1	170.17	6.04	151.2	166.1	170.15	174.1	188.6
weight	0	1	80.20	7.00	57.1	75.4	80.30	84.9	103.4
bmi	0	1	27.76	2.73	19.7	25.9	27.70	29.5	39.8
SBP	0	1	130.19	8.08	104.0	125.0	130.00	136.0	158.0
LDL	0	1	110.27	19.87	32.0	97.0	110.00	124.0	174.0
recovery_time	0	1	43.02	30.51	3.0	28.0	38.00	49.0	365.0

```
##################################################################
## Remember to edit the next chunk if you do any modification here:)
##################################################################
```

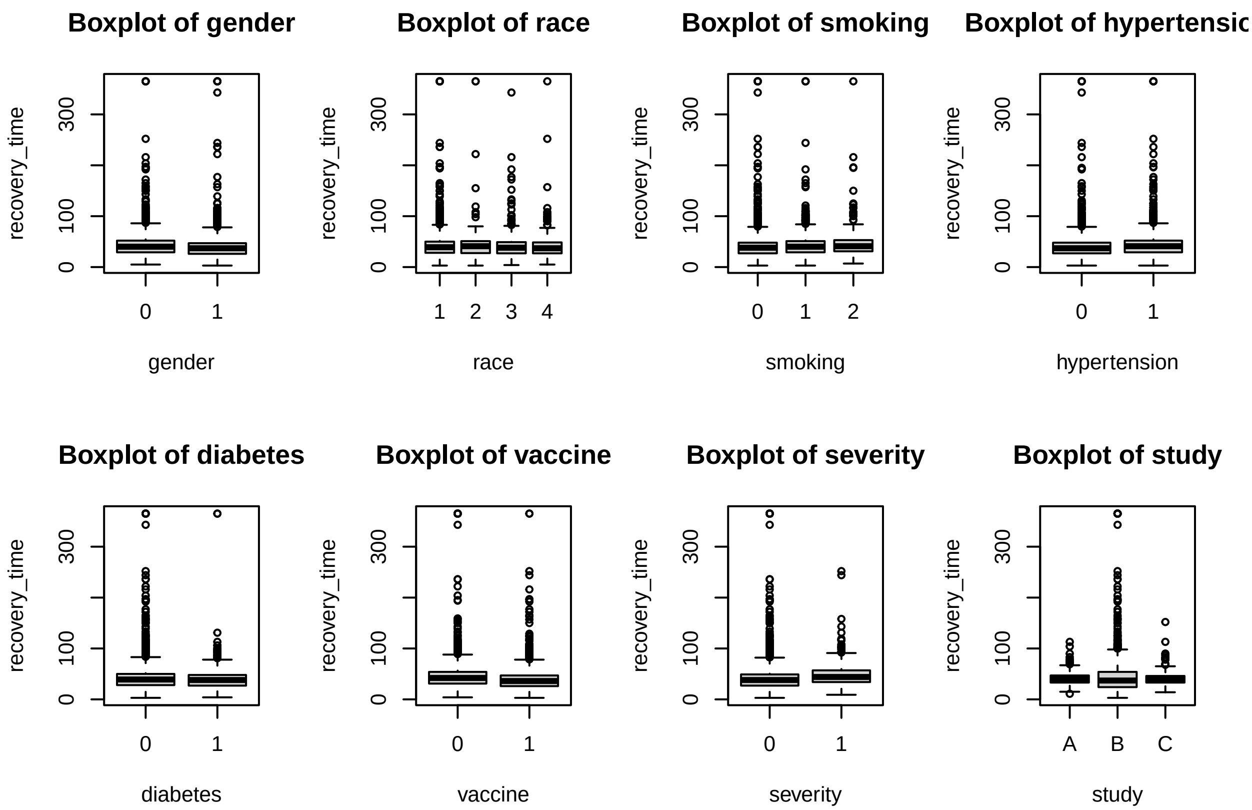
```r
# EDA
# library(GGally)
# ggpairs(dat)

cts_var = c("age", "height", "weight", "bmi", "SBP", "LDL")
fct_var = c("gender", "race", "smoking", "hypertension", "diabetes", "vaccine", "severity", "study")

# scatter plot of continuous predictors
par(mfrow=c(2, 3))
for (i in 1:length(cts_var)){
  var = cts_var[i]
  plot(recovery_time~train.dat[,var],
       data = train.dat,
       ylab = "recovery time",
       xlab = var,
       main = str_c("Scatter Plot of ", var))
  lines(stats::lowess(train.dat[,var], train.dat$recovery_time), col = "red", type = "l")
}
```



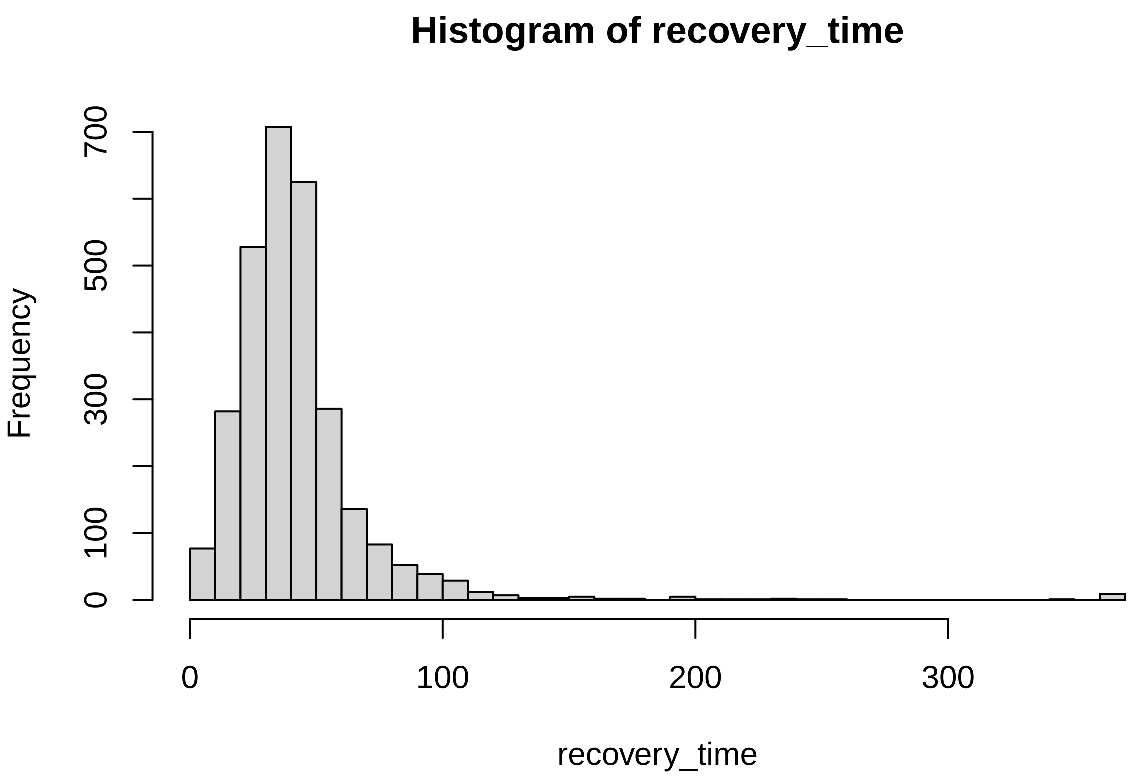

```r
for (i in 1:length(cts_var)){
  var = cts_var[i]
  hist(train.dat[,var],
       ylab = "recovery_time",
       xlab = var,
       main = str_c("Histogram of ", var))
}
```



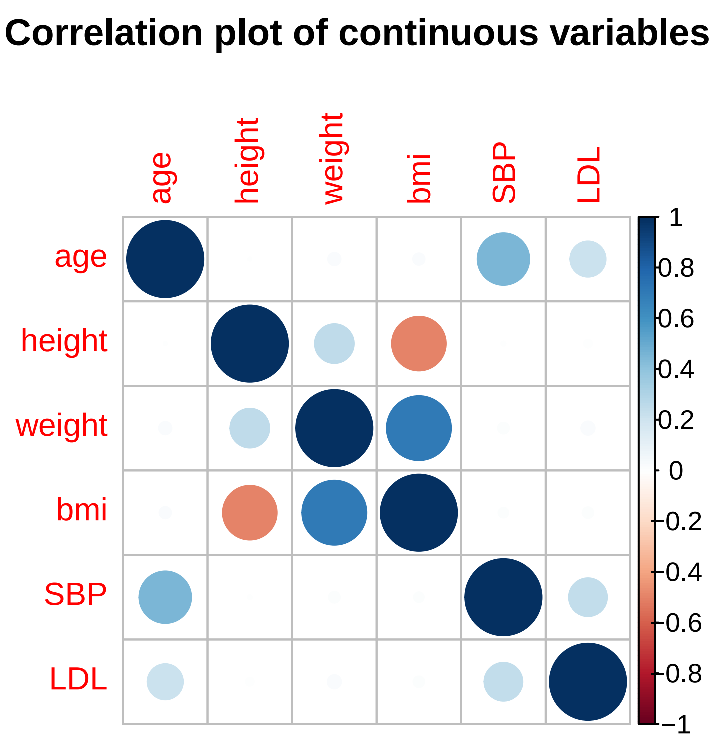

```
# boxplot of categorical predictors
par(mfrow=c(2, 4))
for (i in 1:length(fct_var)){
  var = fct_var[i]
  plot(recovery_time~train.dat[,var],
       data = train.dat,
       ylab = "recovery_time",
       xlab = var,
       main = str_c("Boxplot of ", var))
}
```




```
# histogram of response
par(mfrow=c(1, 1))
hist(train.dat$recovery_time,
     breaks = 50,
     main = "Histogram of recovery_time",
     xlab = "recovery_time")
```




```
# correlation
par(mfrow=c(1, 1))
corrplot(cor(train.dat[,cts_var]), method = "circle", type = "full",
         title = "Correlation plot of continuous variables",
         mar = c(2, 2, 4, 2))
```



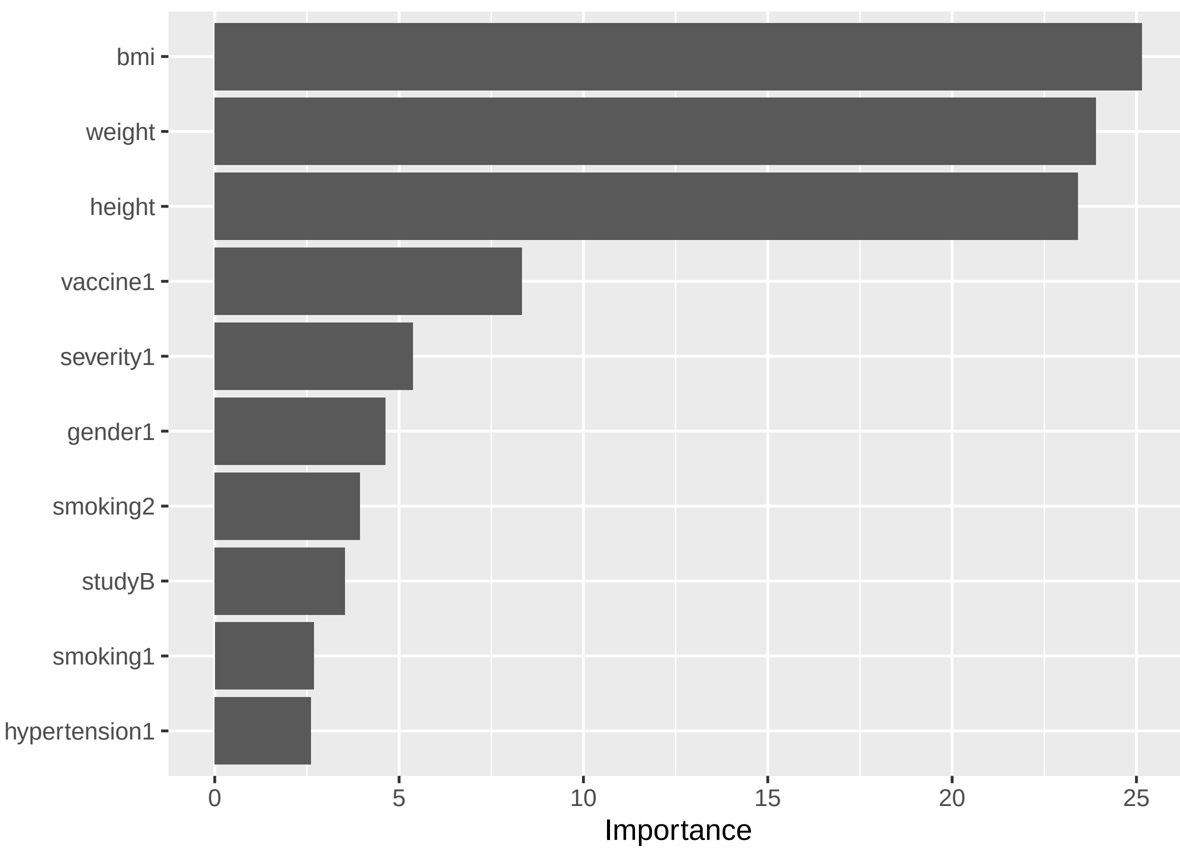
3.2 Model Training

3.2.1 Linear Model

```
ctrl1 <- trainControl(method = "cv", number = 10)
set.seed(2023)

lm.fit <- train(train.x, train.y,
                method = "lm",
                trControl = ctrl1)

coef(lm.fit$finalModel)
```

```
##   (Intercept)           age       gender1         race2         race3
## -3.190120e+03  1.163953e-01 -4.443893e+00  2.189010e+00 -6.599719e-01
##         race4      smoking1      smoking2        height        weight
## -1.156806e+00  2.905693e+00  6.427376e+00  1.866280e+01 -2.014323e+01
##           bmi hypertension1     diabetes1           SBP           LDL
##  6.056969e+01  4.165589e+00 -1.152370e+00 -7.863399e-02 -4.215262e-02
##      vaccine1     severity1        studyB        studyC
## -8.133542e+00  8.747096e+00  4.368587e+00 -6.869681e-01
```

```
vip(lm.fit$finalModel)
```



3.2.2 LASSO

```
set.seed(2023)
lasso.fit <- train(train.x, train.y,
```

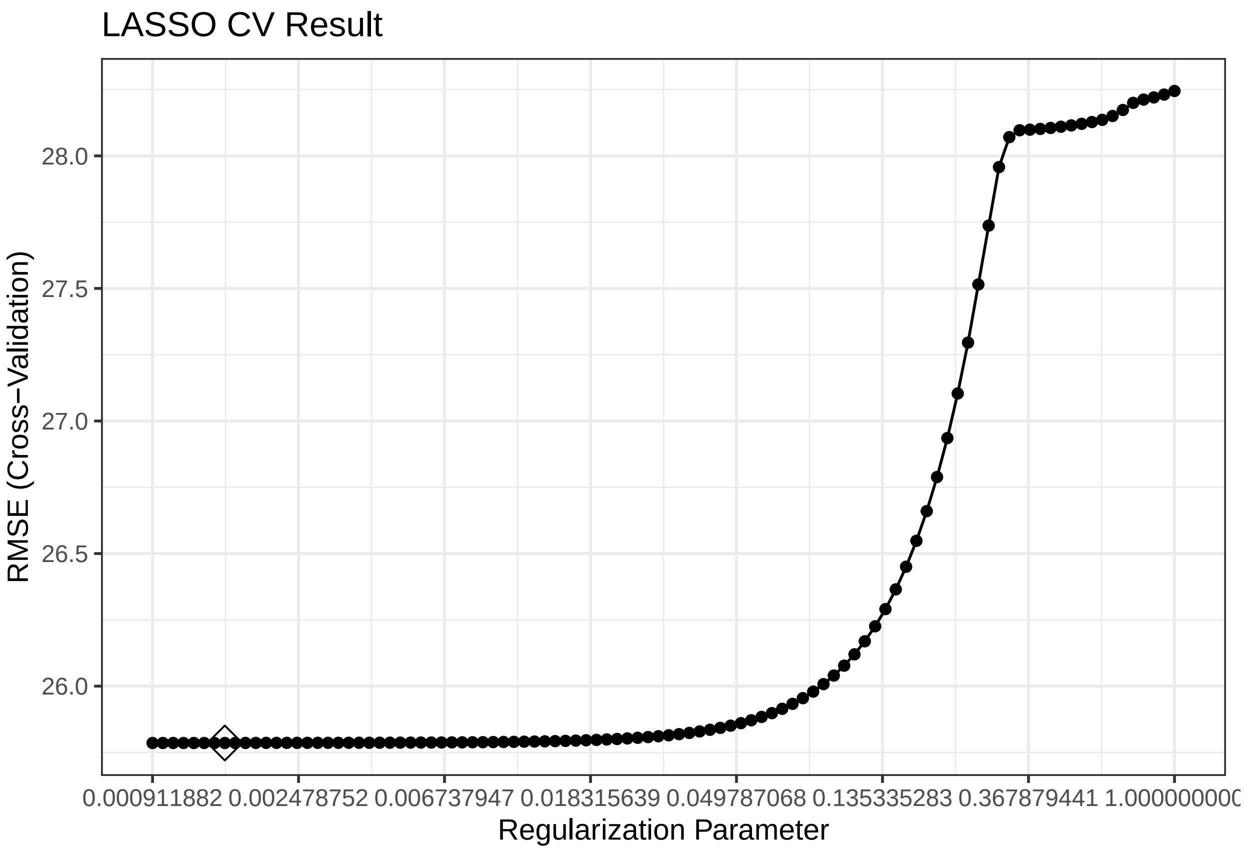
```r
                    method = "glmnet",
                    tuneGrid = expand.grid(
                      alpha = 1,
                      lambda = exp(seq(0, -7, length=100))),
                    trControl = ctrl1)

lasso.fit$bestTune
```

```
##   alpha      lambda
## 8     1 0.001495865
```

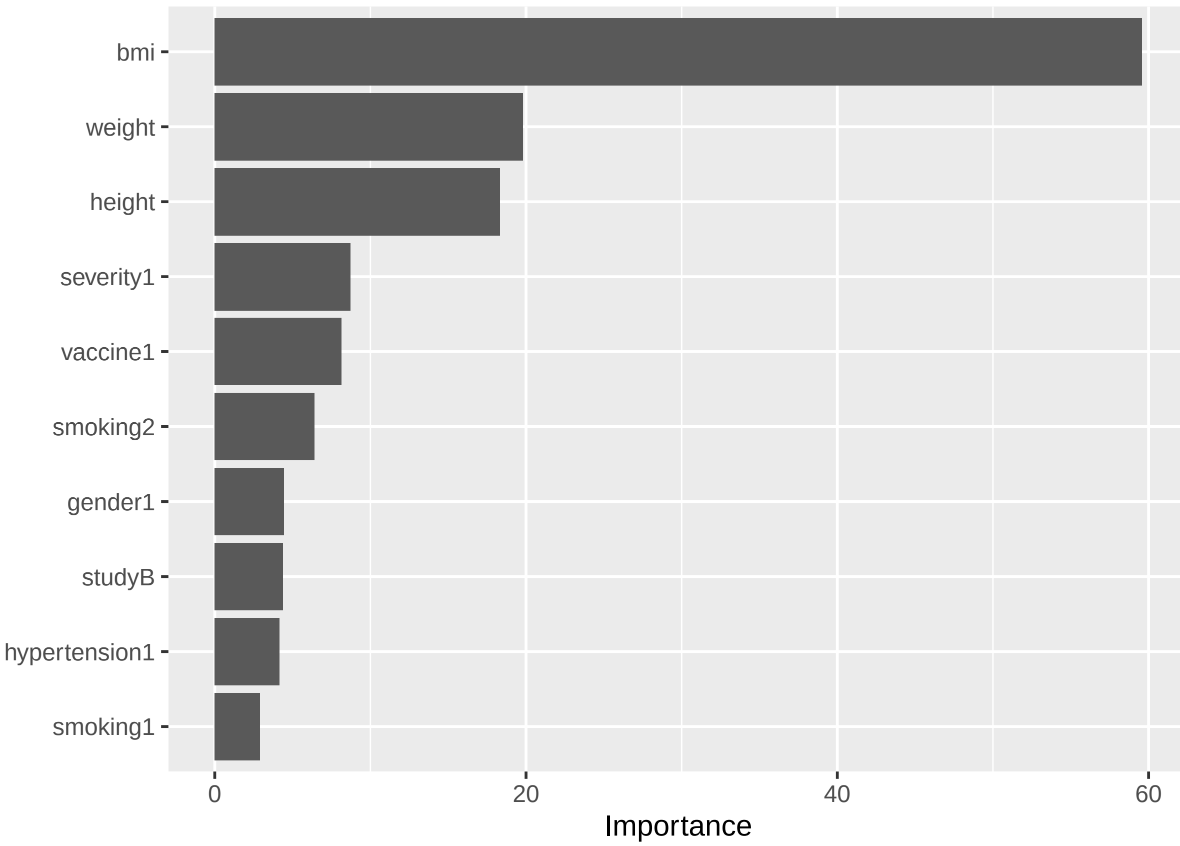
```r
coef(lasso.fit$finalModel, s = lasso.fit$bestTune$lambda)
```

```
## 19 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)   -3.134172e+03
## age            1.153955e-01
## gender1       -4.441866e+00
## race2          2.191861e+00
## race3         -6.681255e-01
## race4         -1.149670e+00
## smoking1       2.901232e+00
## smoking2       6.400802e+00
## height         1.833161e+01
## weight        -1.979266e+01
## bmi            5.956877e+01
## hypertension1  4.150461e+00
## diabetes1     -1.160249e+00
## SBP           -7.746419e-02
## LDL           -4.212203e-02
## vaccine1      -8.147730e+00
## severity1      8.730928e+00
## studyB         4.369356e+00
## studyC        -6.781352e-01
```

```r
ggplot(lasso.fit, highlight = TRUE) +
  labs(title="LASSO CV Result") +
  scale_x_continuous(trans='log',n.breaks = 10) +
  theme_bw()
```




```r
ggsave("./figure/lasso_cv.jpeg", dpi = 500)

vip(lasso.fit$finalModel)
```



3.2.3 Ridge

```
set.seed(2023)
ridge.fit <- train(train.x, train.y,
                   method = "glmnet",
                   tuneGrid = expand.grid(alpha = 0,
                                          lambda = exp(seq(1, -5, length=100))),
                   trControl = ctrl1)

ridge.fit$bestTune
```

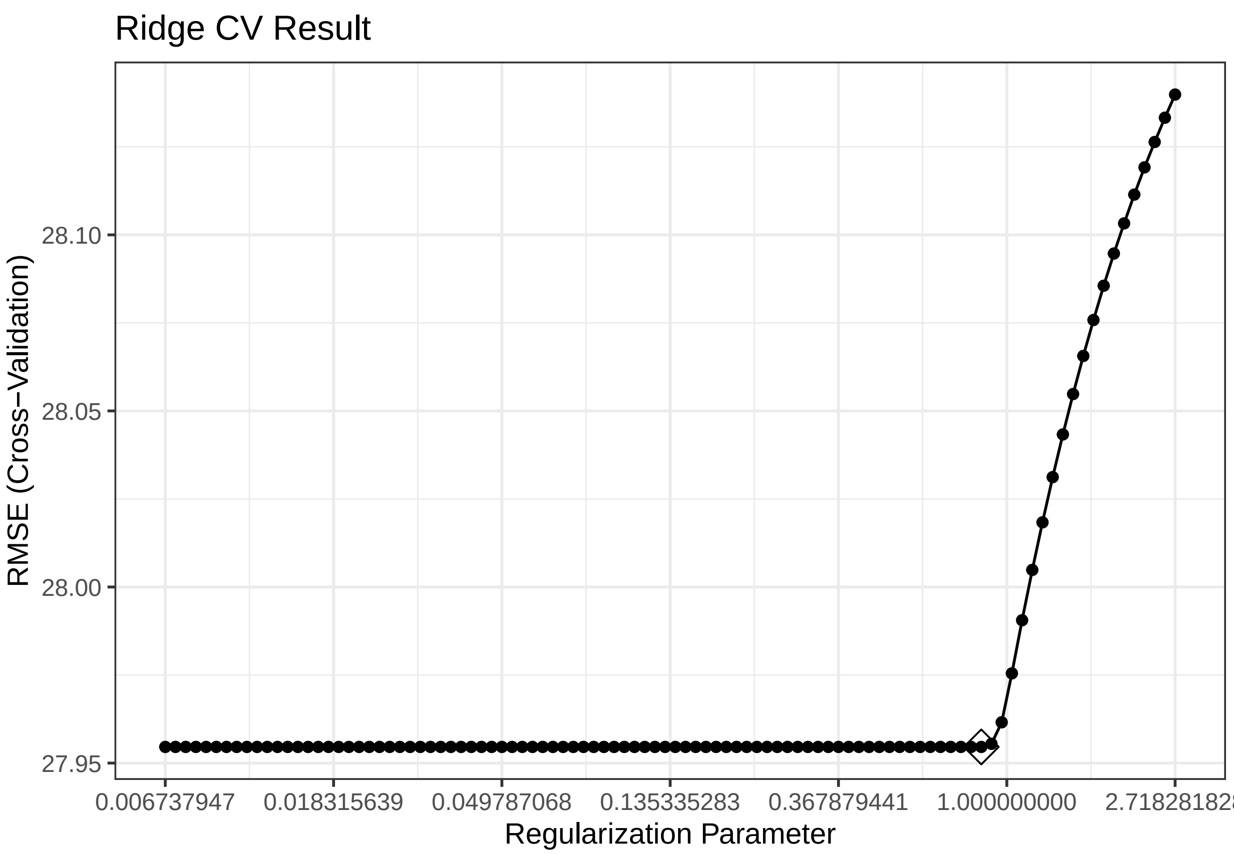
```
##    alpha    lambda
## 81     0 0.8594049
```

```
coef(ridge.fit$finalModel, s = ridge.fit$bestTune$lambda)
```

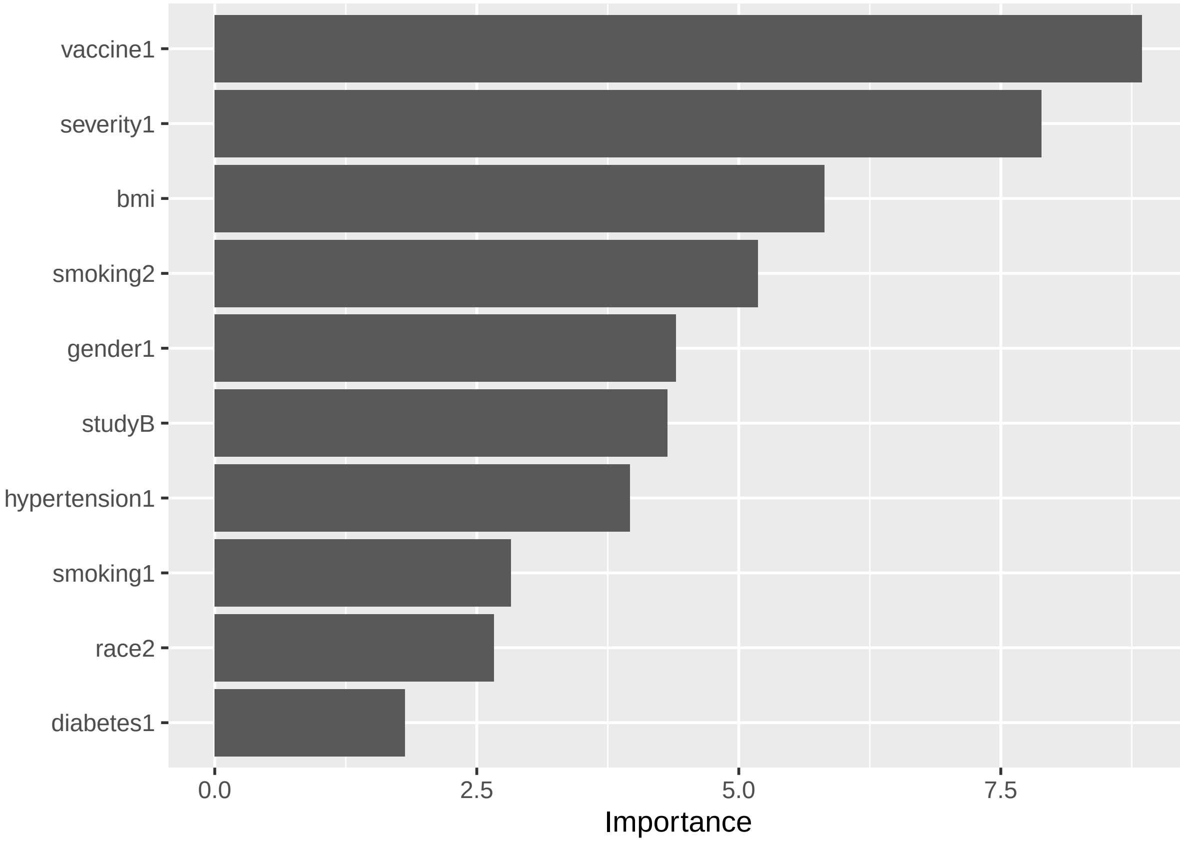
```
## 19 x 1 sparse Matrix of class "dgCMatrix"
##                          s1
## (Intercept)   -131.33806374
## age              0.09731228
## gender1         -4.40320528
## race2            2.66527141
## race3           -1.32710400
## race4           -1.12570977
## smoking1         2.82624366
## smoking2         5.18400128
## height           0.60404463
```

```
## weight          -1.01341715
## bmi              5.81922510
## hypertension1    3.96367066
## diabetes1       -1.81677375
## SBP             -0.06303616
## LDL             -0.04440780
## vaccine1        -8.84608080
## severity1        7.88676978
## studyB           4.32156225
## studyC          -0.51357417
```

```
ggplot(ridge.fit,highlight = TRUE) +
  scale_x_continuous(trans='log', n.breaks = 6) +
  labs(title="Ridge CV Result") +
  theme_bw()
```




```
ggsave("./figure/ridge_cv.jpeg", dpi = 500)

vip(ridge.fit$finalModel)
```



3.2.4 Elastic Net

```
set.seed(2023)
enet.fit <- train(train.x, train.y,
                  method = "glmnet",
                  tuneGrid = expand.grid(
                    alpha = seq(0, 1, length = 21),
                    lambda = exp(seq(0, -8, length = 100))),
                  trControl = ctrl1)

enet.fit$bestTune
```

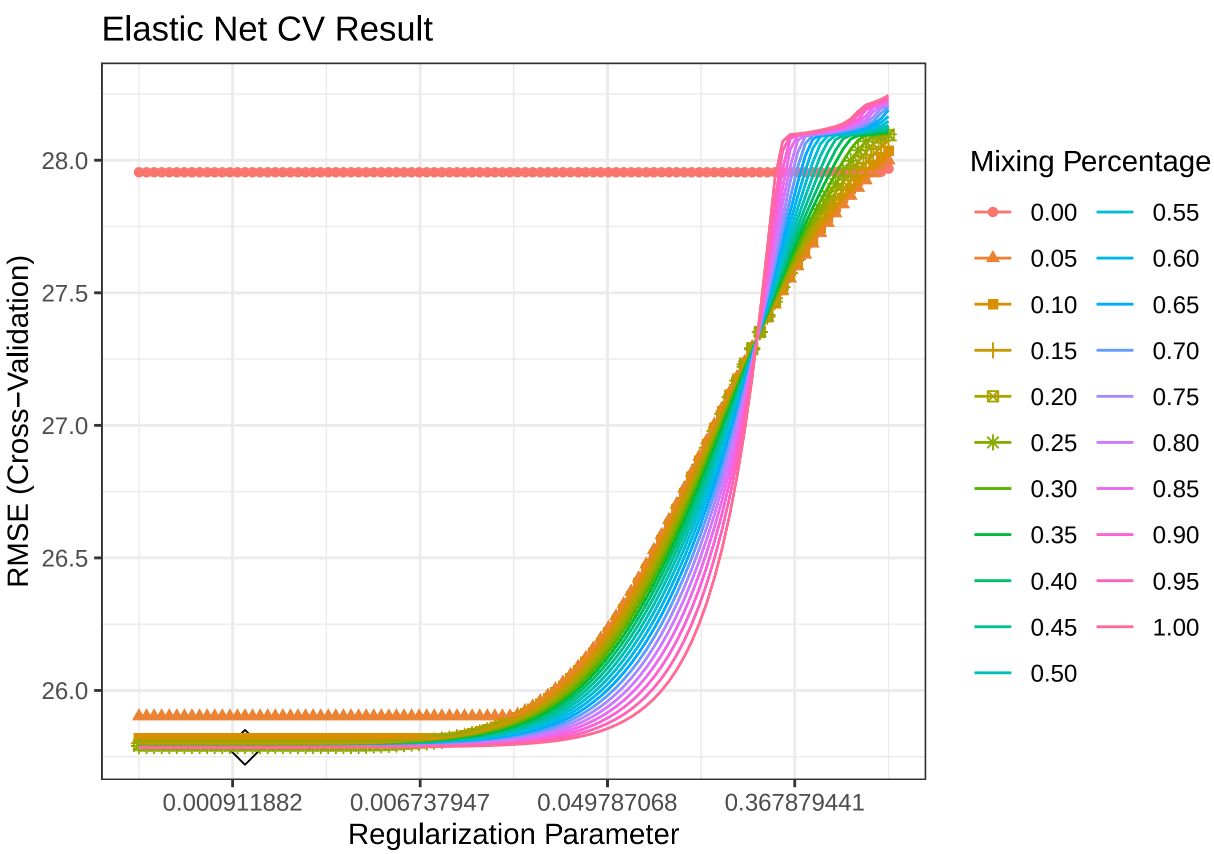
```
##      alpha      lambda
## 1815   0.9 0.001039842
```

```
coef(enet.fit$finalModel, enet.fit$bestTune$lambda)
```

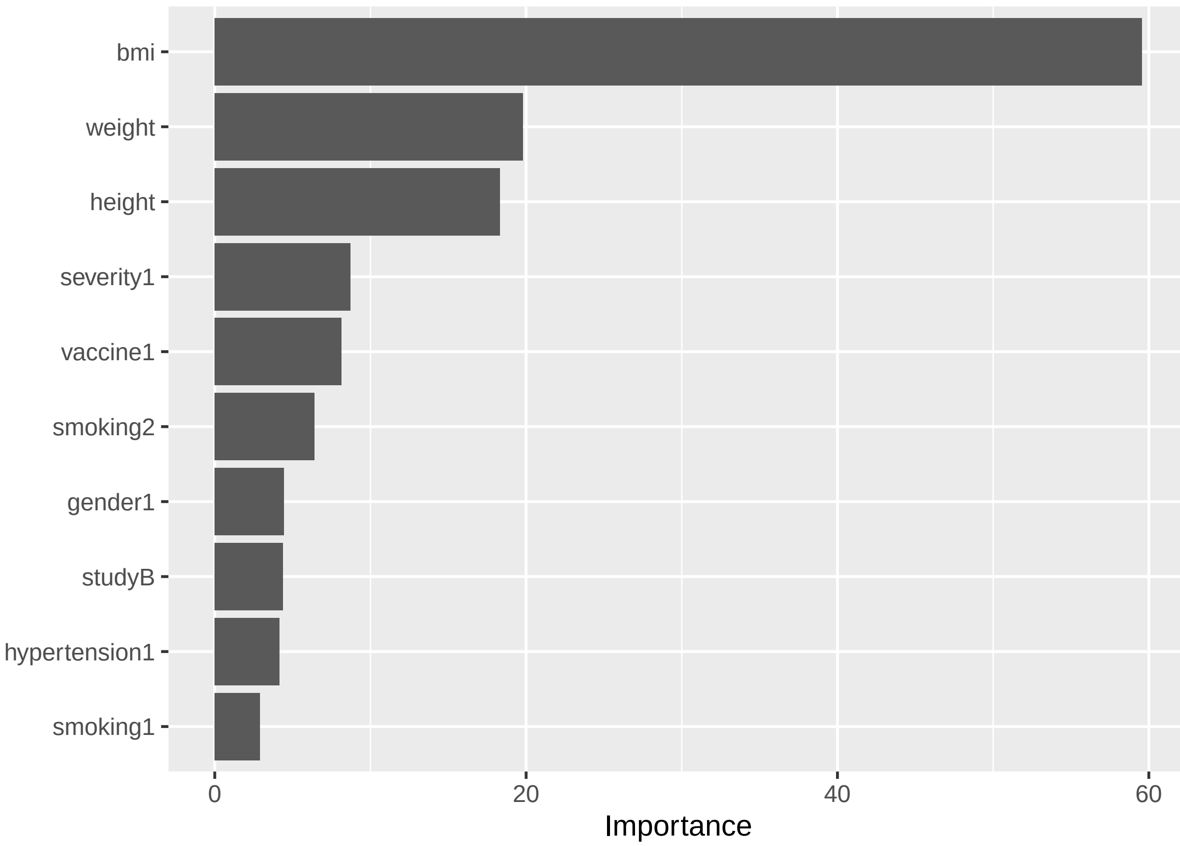
```
## 19 x 1 sparse Matrix of class "dgCMatrix"
##                            s1
## (Intercept)   -3.133363e+03
## age            1.156446e-01
## gender1       -4.443015e+00
## race2          2.194049e+00
## race3         -6.697538e-01
## race4         -1.151993e+00
## smoking1       2.902929e+00
## smoking2       6.403008e+00
```

```
## height          1.832705e+01
## weight         -1.978780e+01
## bmi             5.955488e+01
## hypertension1   4.156169e+00
## diabetes1      -1.161920e+00
## SBP            -7.786025e-02
## LDL            -4.215546e-02
## vaccine1       -8.149202e+00
## severity1       8.732536e+00
## studyB          4.370077e+00
## studyC         -6.790033e-01
```

```
ggplot(enet.fit, highlight = TRUE) +
  scale_x_continuous(trans='log', n.breaks = 6) +
  labs(title ="Elastic Net CV Result") +
  theme_bw()
```

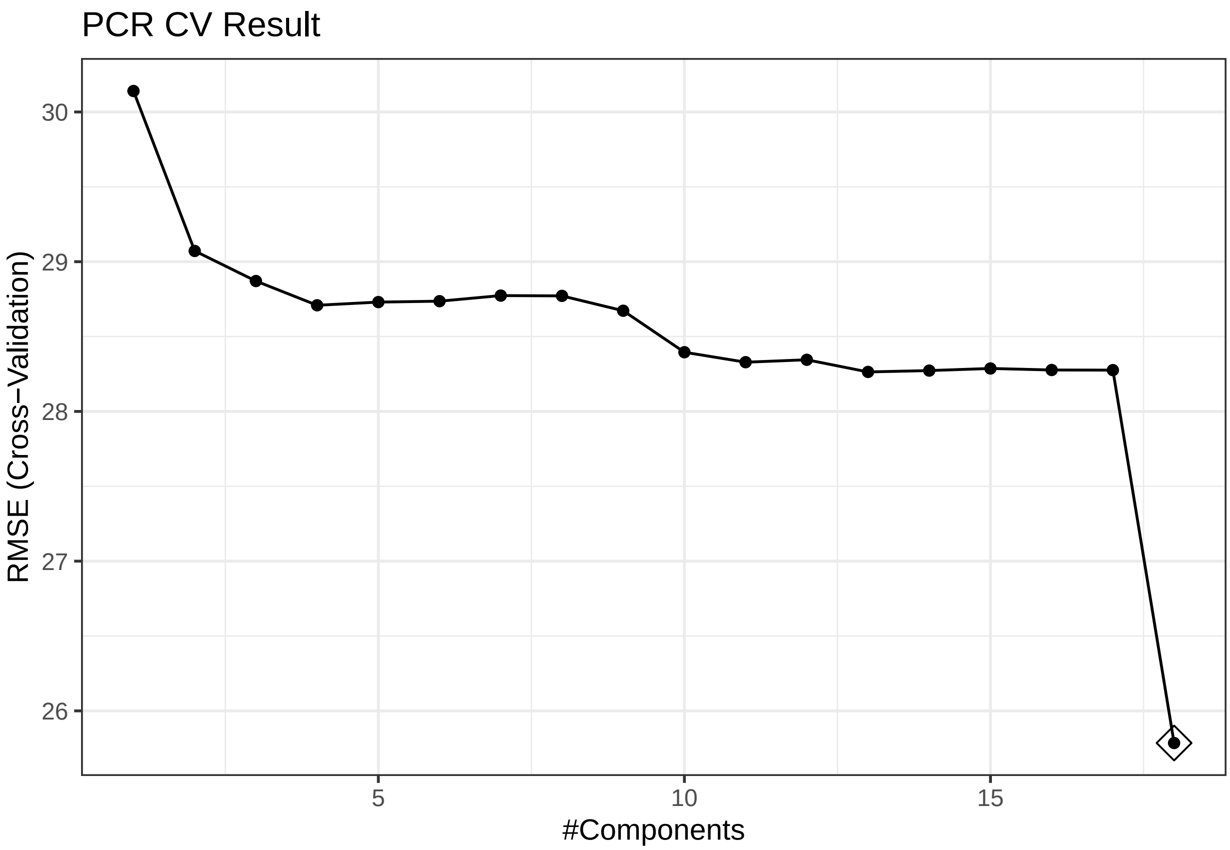



```
ggsave("./figure/enet_cv.jpeg", dpi = 500)

vip(enet.fit$finalModel)
```



3.2.5 Principal components regression (PCR)

```
set.seed(2023)
pcr.fit <- train(train.x,
                 train.y,
                 method = "pcr",
                 tuneGrid  = data.frame(ncomp = 1:ncol(train.x)),
                 trControl = ctrl1,
                 preProcess = c("center", "scale"))
ggplot(pcr.fit, highlight = TRUE) +
  labs(title  ="PCR CV Result") +
  theme_bw()
```




```r
ggsave("./figure/pcr_cv.jpeg", dpi = 500)

pcr.fit$bestTune
```

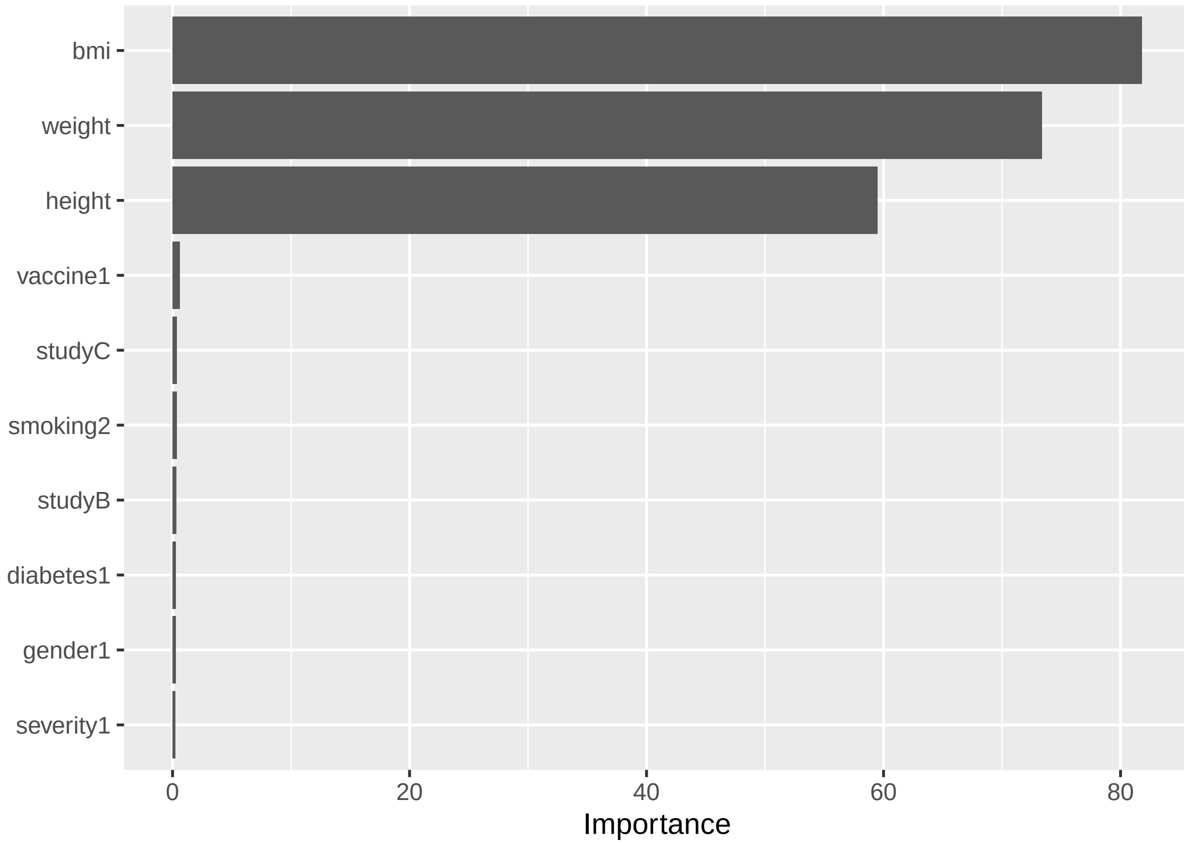
```
##    ncomp
## 18    18
```

```r
coef(pcr.fit$finalModel)
```

```
## , , 18 comps
##
##                     .outcome
## age                0.5252538
## gender1           -2.2221586
## race2              0.4563464
## race3             -0.2619635
## race4             -0.3476329
## smoking1           1.3205684
## smoking2           1.9344423
## height           112.6936931
## weight          -141.0001175
## bmi              165.1518985
## hypertension1      2.0811234
## diabetes1         -0.4188178
## SBP               -0.6356938
## LDL               -0.8376686
## vaccine1          -4.0025673
## severity1          2.5879846
```

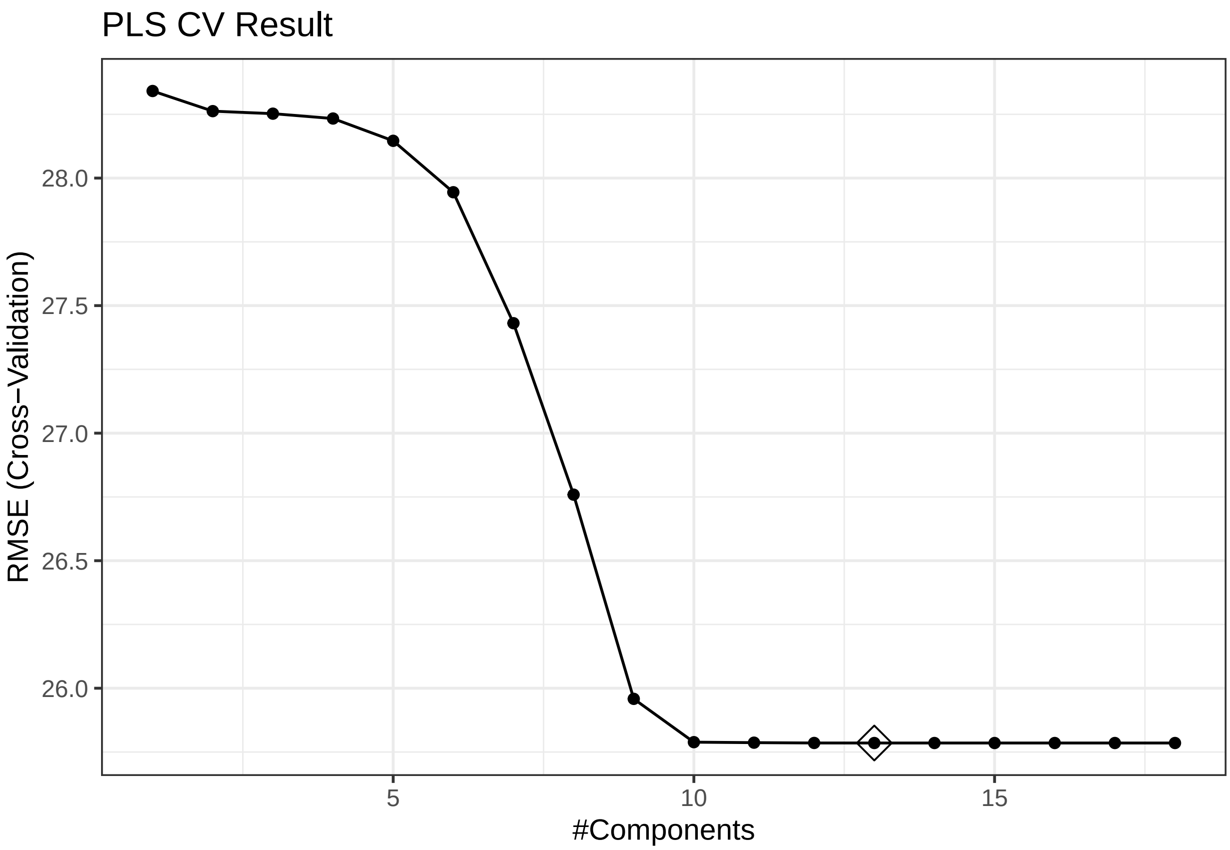
```
## studyB            2.1374000
## studyC           -0.2730416
```

```
vip(pcr.fit$finalModel)
```



3.2.6 Partial Least Squares (PLS)

```
set.seed(2023)
pls.fit <- train(train.x,
                 train.y,
                 method = "pls",
                 tuneGrid = data.frame(ncomp = 1:ncol(train.x)),
                 trControl = ctrl1,
                 preProcess = c("center", "scale"))
ggplot(pls.fit, highlight = TRUE) +
  labs(title  ="PLS CV Result") +
  theme_bw()
```




```
ggsave("./figure/pls_cv.jpeg", dpi = 500)

pls.fit$bestTune
```

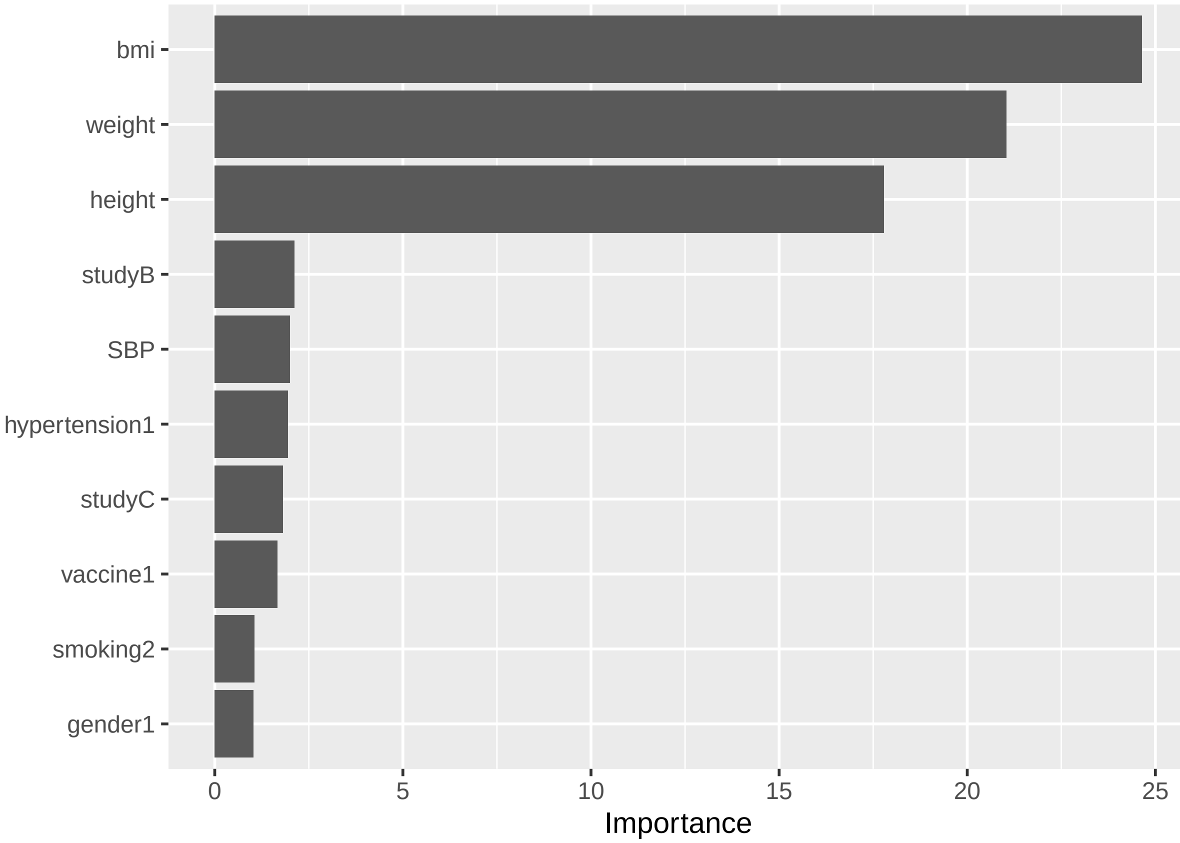
```
##    ncomp
## 13    13
```

```
coef(pls.fit$finalModel)
```

```
## , , 13 comps
##
##                    .outcome
## age               0.5253162
## gender1          -2.2224171
## race2             0.4564699
## race3            -0.2616135
## race4            -0.3472528
## smoking1          1.3206873
## smoking2          1.9344789
## height          112.6936914
## weight         -141.0001239
## bmi             165.1518926
## hypertension1     2.0811255
## diabetes1        -0.4187817
## SBP              -0.6356784
## LDL              -0.8377705
## vaccine1         -4.0025291
## severity1         2.5877989
```

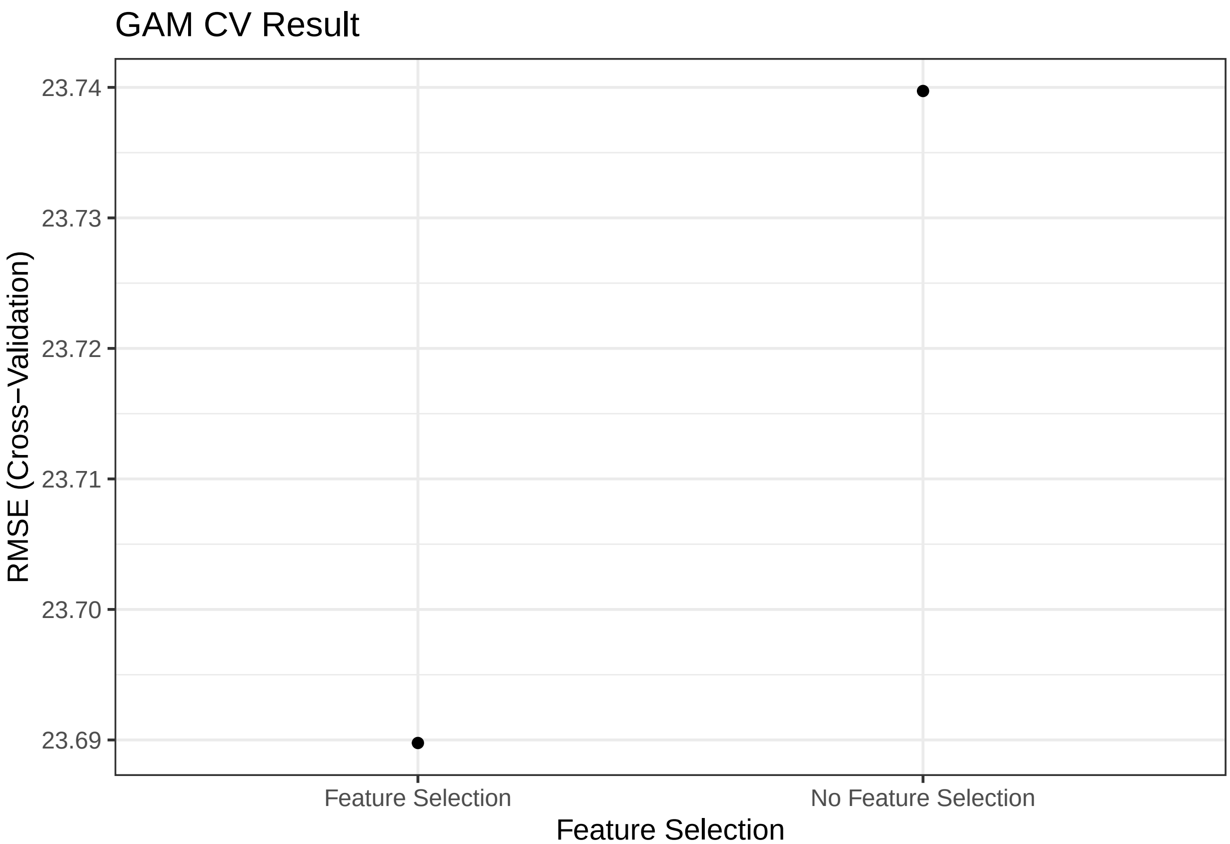
```
## studyB            2.1374098
## studyC           -0.2730417
```

```
vip(pls.fit$finalModel)
```



3.2.7 Generalized Additive Model (GAM)

```
set.seed(2023)
gam.fit <- train(train.x,
                 train.y,
                 method = "gam",
                 tuneGrid = data.frame(select = c(TRUE, FALSE),
                                       method = "GCV.Cp"),
                 trControl = ctrl1)


ggplot(gam.fit) +
  labs(title = "GAM CV Result") +
  theme_bw()
```



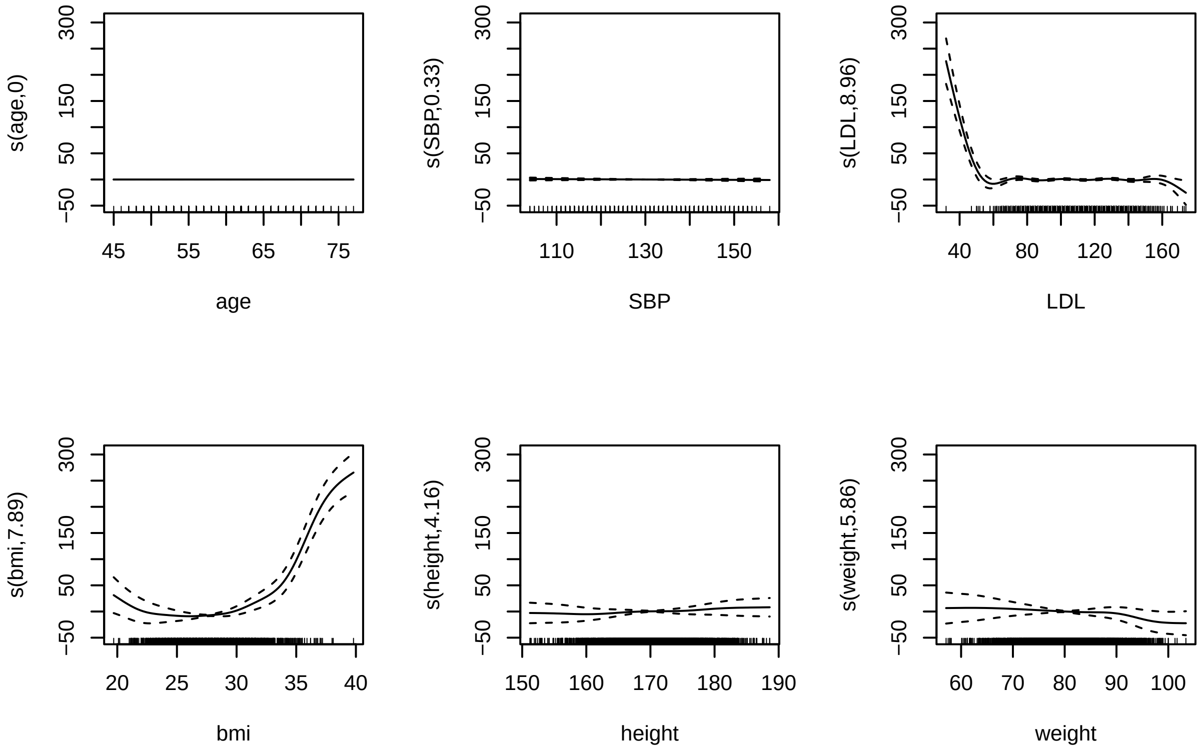

```
ggsave("./figure/gam_cv.jpeg", dpi = 500)

gam.fit$bestTune
```

```
##   select method
## 2   TRUE GCV.Cp
```

```
# coef(gam.fit$finalModel)
gam.fit$finalModel
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## .outcome ~ gender1 + race3 + race4 + smoking1 + smoking2 + hypertension1 +
##     diabetes1 + vaccine1 + severity1 + studyB + studyC + s(age) +
##     s(SBP) + s(LDL) + s(bmi) + s(height) + s(weight)
##
## Estimated degrees of freedom:
## 0.000 0.329 8.959 7.893 4.163 5.856  total = 39.2
##
## GCV score: 524.051
```

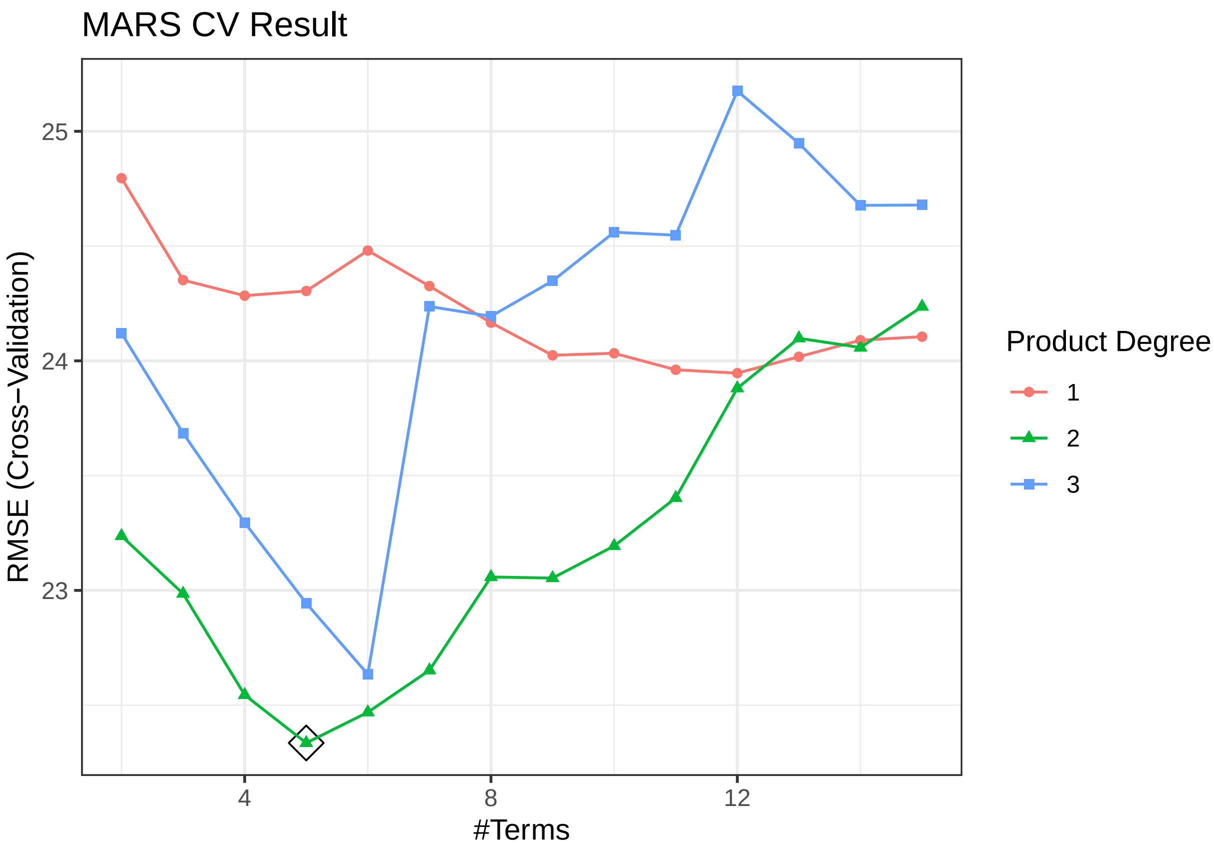
```
par(mfrow=c(2, 3))
plot(gam.fit$finalModel)
```




```
par(mfrow=c(1, 1))
```

3.2.8 Multivariate Adaptive Regression Splines (MARS)

```
mars_grid <- expand.grid(degree = 1:3,
                         nprune = 2:15)
set.seed(2023)
mars.fit <- train(train.x,
                  train.y,
                  method = "earth",
                  tuneGrid = mars_grid,
                  trControl = ctrl1)

ggplot(mars.fit, highlight = TRUE)+
  labs(title  ="MARS CV Result") +
  theme_bw()
```




```r
ggsave("./figure/mars_cv.jpeg", dpi = 500)

mars.fit$bestTune
```

```
##    nprune degree
## 18      5      2
```

```r
coef(mars.fit$finalModel)
```

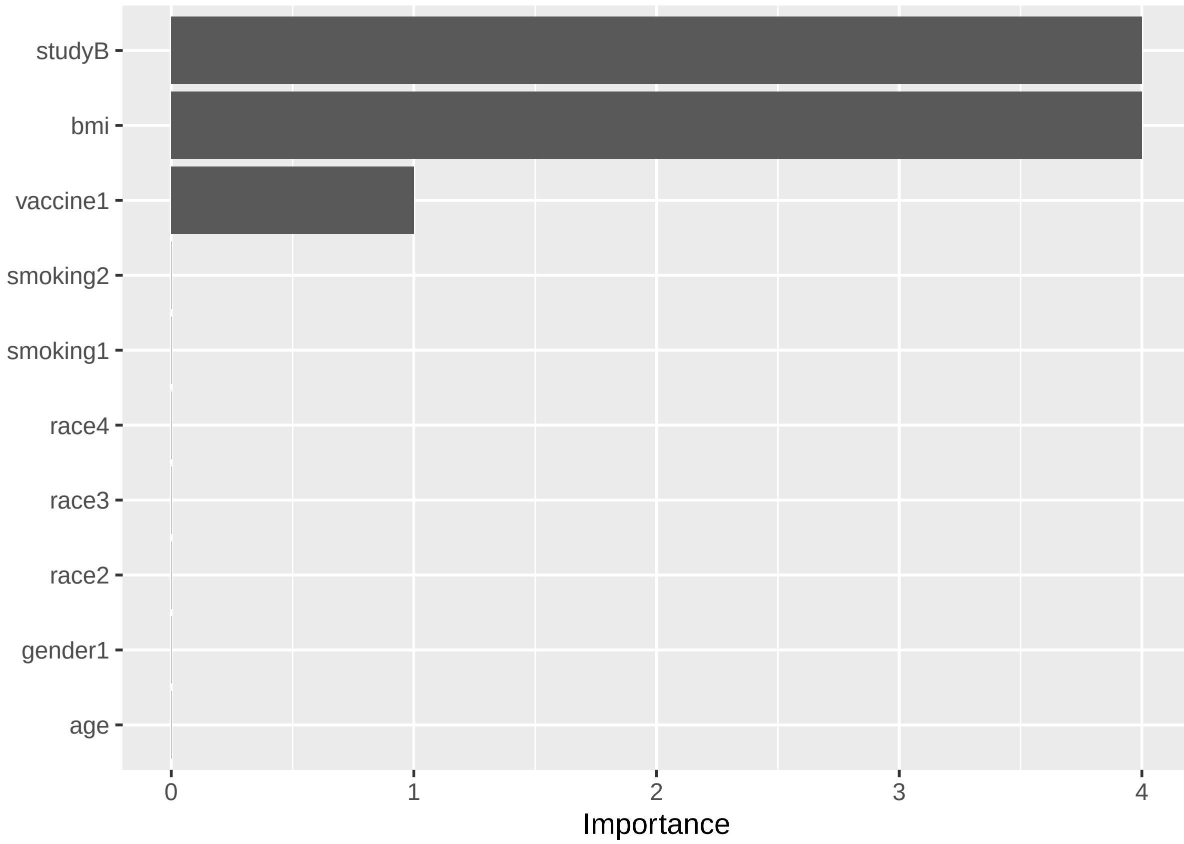
```
##          (Intercept)          h(31.7-bmi) h(bmi-31.7) * studyB
##            19.366730             3.705371            34.383832
##          h(bmi-26.8)             vaccine1
##             6.695655            -7.788338
```

```r
summary(mars.fit$finalModel)
```

```
## Call: earth(x=matrix[2900,18], y=c(40,34,31,50,3...), keepxy=TRUE, degree=2,
##             nprune=5)
##
##                      coefficients
## (Intercept)             19.366730
## vaccine1                -7.788338
## h(bmi-26.8)              6.695655
## h(31.7-bmi)              3.705371
## h(bmi-31.7) * studyB    34.383832
##
## Selected 5 of 25 terms, and 3 of 18 predictors (nprune=5)
## Termination condition: Reached nk 37
## Importance: bmi, studyB, vaccine1, age-unused, gender1-unused, ...
```

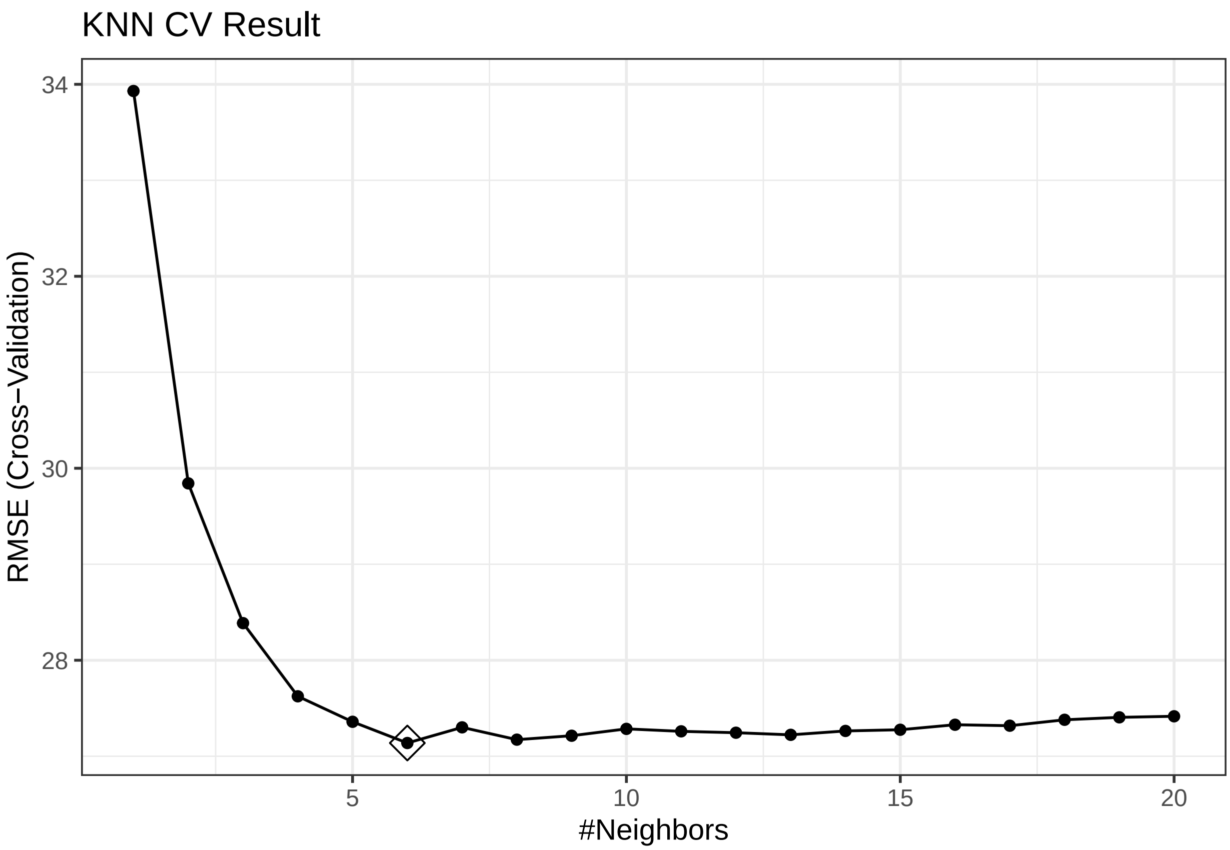
```
## Number of terms at each degree of interaction: 1 3 1
## GCV 491.1694    RSS 1413606    GRSq 0.4723714    RSq 0.4760052
```

```
vip(mars.fit$finalModel)
```



3.2.9 K-Nearest Neighbour (KNN)

```
set.seed(2023)
knn.fit <- train(train.x,
                 train.y,
                 tuneGrid  = data.frame(k = 1:20),
                 method = "knn",
                 trControl = ctrl1)

ggplot(knn.fit, highlight = TRUE) +
  labs(title  ="KNN CV Result") +
  theme_bw()
```




```
ggsave("./figure/knn_cv.jpeg", dpi = 500)

knn.fit$bestTune
```

```
##   k
## 6 6
```

3.2.10 Bagging

3.2.11 Random Forest

3.2.12 Boosting

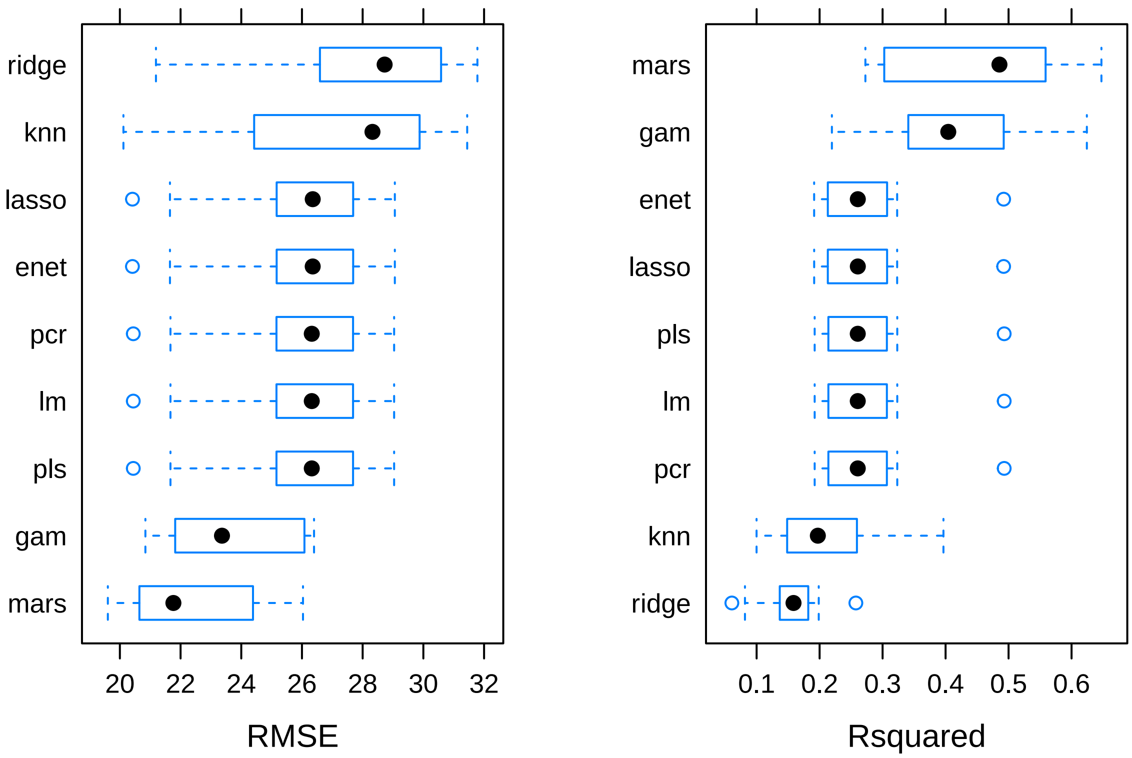
3.2.13 Regression Trees

3.3 Model Selection

```
set.seed(2023)
resamp <- resamples(list(lm = lm.fit,
                         lasso = lasso.fit,
                         ridge = ridge.fit,
                         enet = enet.fit,
                         pcr = pcr.fit,
                         pls = pls.fit,
                         gam = gam.fit,
                         mars = mars.fit,
                         knn = knn.fit))

summary(resamp)
```

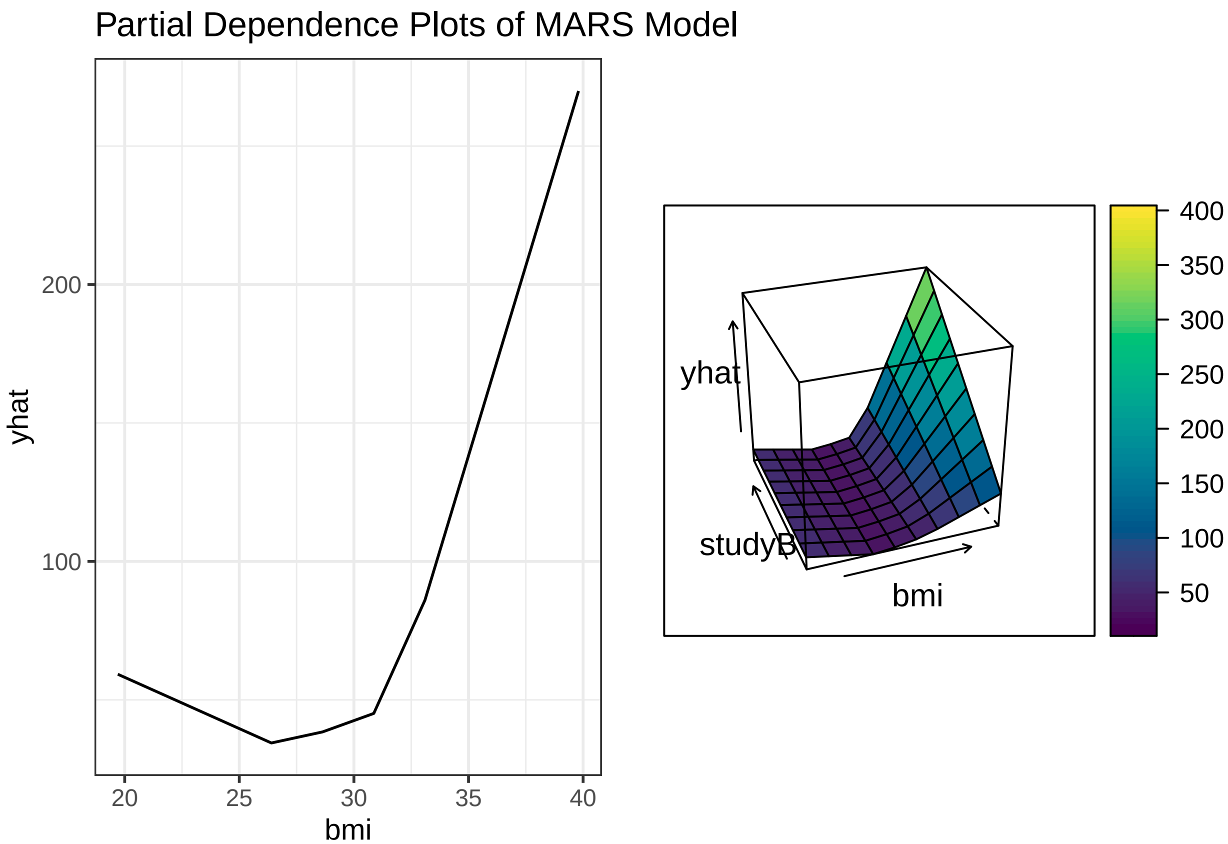
```
##
## Call:
## summary.resamples(object = resamp)
##
## Models: lm, lasso, ridge, enet, pcr, pls, gam, mars, knn
## Number of resamples: 10
##
## MAE
##           Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## lm    15.54483 15.80758 16.63529 16.59842 17.13204 18.12333    0
## lasso 15.51069 15.78658 16.61245 16.57052 17.09219 18.09015    0
## ridge 15.34004 16.62387 16.79935 16.84047 17.23997 18.17959    0
## enet  15.51026 15.78694 16.61217 16.57069 17.09223 18.09088    0
## pcr   15.54483 15.80758 16.63529 16.59842 17.13204 18.12333    0
## pls   15.54482 15.80753 16.63528 16.59840 17.13208 18.12332    0
## gam   14.60392 14.76502 15.40409 15.42678 15.78762 17.02963    0
## mars  14.06187 14.29497 14.88239 14.89479 15.31286 16.10880    0
## knn   14.43602 16.28400 16.79135 16.77166 17.45629 18.38966    0
##
## RMSE
##           Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## lm    20.44180 25.16612 26.32308 25.78528 27.58385 29.03646    0
## lasso 20.41446 25.16779 26.35443 25.78553 27.58286 29.05994    0
## ridge 21.18921 26.76934 28.72409 27.95459 30.39855 31.78080    0
## enet  20.41395 25.16792 26.35390 25.78540 27.58280 29.06018    0
## pcr   20.44180 25.16612 26.32308 25.78528 27.58385 29.03646    0
## pls   20.44179 25.16611 26.32305 25.78526 27.58386 29.03644    0
## gam   20.84135 22.00149 23.36475 23.68977 25.89070 26.39798    0
## mars  19.60380 20.76550 21.76341 22.33527 23.91386 26.03407    0
## knn   20.11678 25.01933 28.32298 27.13762 29.65682 31.44427    0
##
## Rsquared
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## lm    0.19215021 0.2201628 0.2605519 0.2764092 0.3001496 0.4930552    0
## lasso 0.19133277 0.2196625 0.2606385 0.2763222 0.3004686 0.4921785    0
## ridge 0.06069200 0.1374835 0.1585905 0.1552980 0.1812584 0.2575556    0
## enet  0.19132111 0.2196526 0.2606417 0.2763214 0.3004556 0.4921930    0
## pcr   0.19215021 0.2201628 0.2605519 0.2764092 0.3001496 0.4930552    0
## pls   0.19215072 0.2201634 0.2605543 0.2764103 0.3001491 0.4930584    0
## gam   0.21948254 0.3453667 0.4042745 0.4084093 0.4864782 0.6243131    0
## mars  0.27268902 0.3335869 0.4855151 0.4599541 0.5506146 0.6474131    0
## knn   0.09988269 0.1495740 0.1971881 0.2119237 0.2570144 0.3966191    0
```

```r
# jpeg("./figure/resample.jpeg", width = 8, height=6, units="in", res=500)
p1=bwplot(resamp, metric = "RMSE")
p2=bwplot(resamp, metric = "Rsquared")
grid.arrange(p1, p2 ,ncol=2)
```




```
# dev.off()
```

```
p1<- pdp::partial(mars.fit, pred.var = c("bmi"), grid.resolution = 10) %>% autoplot() +
  theme_bw()+
  labs(title = "Partial Dependence Plots of MARS Model")

p2 <-pdp::partial(mars.fit, pred.var = c("bmi", "studyB"),
                   grid.resolution = 10) %>%
      pdp::plotPartial(levelplot = FALSE, zlab = "yhat", drape = TRUE,
                       screen = list(z = 20, x = -60))

# jpeg("./figure/partial_dependence.jpeg", width = 8, height=6, units="in", res=500)
gridExtra::grid.arrange(p1, p2, ncol = 2)
```




```
# dev.off()

# Important variables
varImp(mars.fit$finalModel)
```

```
##            Overall
## bmi      100.00000
## studyB   100.00000
## vaccine1  17.78457
```

3.4 Training / Testing Error

```
# training error
mars.train.pred = predict(mars.fit, newdata = train.x)
RMSE(train.y, mars.train.pred)
```

```
## [1] 22.07828
```

```
# testing error
mars.pred = predict(mars.fit, newdata = test.x)
RMSE(test.y, mars.pred)
```

```
## [1] 22.1712
```

4 Secondary Analysis

4.1 Exploratory analysis and data visualization

```
# data summary
st_options(plain.ascii = FALSE,
           style = "rmarkdown",
           dfSummary.silent = TRUE,
           footnote = NA,
           subtitle.emphasis = FALSE)
dfSummary(train.bin.dat)
```

4.1.1 Data Frame Summary

train.bin.dat
Dimensions: 2900 x 15
Duplicates: 0

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	age [numeric]	Mean (sd) : 60.1 (4.5) min < med < max: 45 < 60 < 77 IQR (CV) : 6 (0.1)	33 distinct values	: : . : : : : : . . : : : : : .	2900 (100.0%)	0 (0.0%)
2	gender [factor]	1. 0 2. 1	1468 (50.6%) 1432 (49.4%)	IIIIIIIIII IIIIIIIII	2900 (100.0%)	0 (0.0%)
3	race [factor]	1. 1 2. 2 3. 3 4. 4	1909 (65.8%) 132 (4.6%) 568 (19.6%) 291 (10.0%)	IIIIIIIIIIIII III II	2900 (100.0%)	0 (0.0%)
4	smoking [factor]	1. 0 2. 1 3. 2	1763 (60.8%) 845 (29.1%) 292 (10.1%)	IIIIIIIIIIII IIIII II	2900 (100.0%)	0 (0.0%)
5	height [numeric]	Mean (sd) : 170.2 (6) min < med < max: 151.2 < 170.1 < 188.6 IQR (CV) : 8 (0)	312 distinct values	: : : : . : : . : : : : . : : : : .	2900 (100.0%)	0 (0.0%)
6	weight [numeric]	Mean (sd) : 80.2 (7) min < med < max: 57.1 < 80.3 < 103.4 IQR (CV) : 9.5 (0.1)	361 distinct values	. : . : : : : : : . : : : : . . : : : : : : .	2900 (100.0%)	0 (0.0%)
7	bmi [numeric]	Mean (sd) : 27.8 (2.7) min < med < max: 19.7 < 27.7 < 39.8 IQR (CV) : 3.6 (0.1)	160 distinct values	: . : : : : : . : : : : : : : : : : .	2900 (100.0%)	0 (0.0%)
8	hypertension [factor]	1. 0 2. 1	1514 (52.2%) 1386 (47.8%)	IIIIIIIIII IIIIIIIII	2900 (100.0%)	0 (0.0%)
9	diabetes [factor]	1. 0 2. 1	2446 (84.3%) 454 (15.7%)	IIIIIIIIIIIIIIII III	2900 (100.0%)	0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
10	SBP [numeric]	Mean (sd) : 130.2 (8.1) min < med < max: 104 < 130 < 158 IQR (CV) : 11 (0.1)	54 distinct values	: : . : : : . . : : : : . : : : : : :	2900 (100.0%)	0 (0.0%)
11	LDL [numeric]	Mean (sd) : 110.3 (19.9) min < med < max: 32 < 110 < 174 IQR (CV) : 27 (0.2)	116 distinct values	. : : : : : : : . : : : : : . : : : : : : .	2900 (100.0%)	0 (0.0%)
12	vaccine [factor]	1. 0 2. 1	1192 (41.1%) 1708 (58.9%)	IIIIIIII IIIIIIIIIII	2900 (100.0%)	0 (0.0%)
13	severity [factor]	1. 0 2. 1	2619 (90.3%) 281 (9.7%)	IIIIIIIIIIIIIIIIII I	2900 (100.0%)	0 (0.0%)
14	study [factor]	1. A 2. B 3. C	580 (20.0%) 1750 (60.3%) 570 (19.7%)	IIII IIIIIIIIIIII III	2900 (100.0%)	0 (0.0%)
15	recovery_time [factor]	1. <=30 2. >30	887 (30.6%) 2013 (69.4%)	IIIIII IIIIIIIIIIIII	2900 (100.0%)	0 (0.0%)

```
skimr::skim_without_charts(train.bin.dat)
```

Table 6: Data summary

Name	train.bin.dat
Number of rows	2900
Number of columns	15
Column type frequency:	
factor	9
numeric	6
Group variables	None

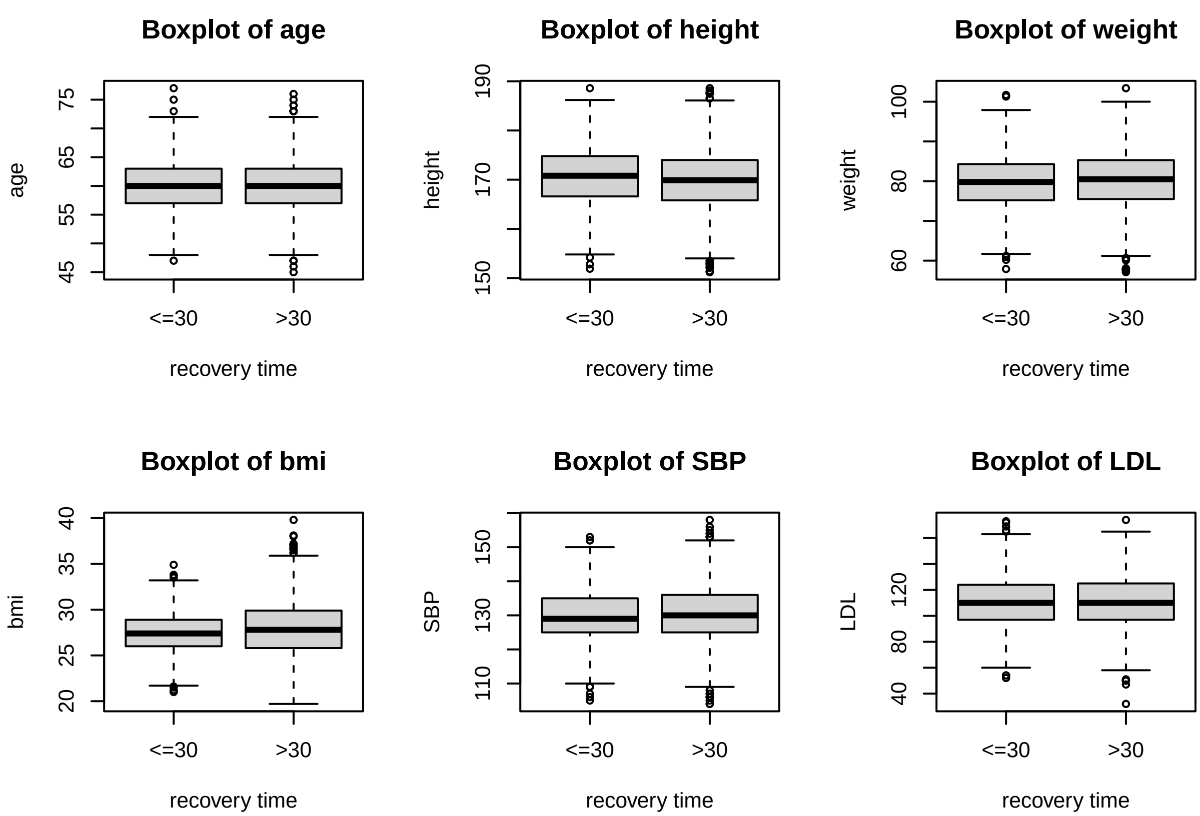
Variable type: factor

skim_variable	n_missing	complete_rate	ordered	n_unique	top_counts
gender	0	1	FALSE	2	0: 1468, 1: 1432
race	0	1	FALSE	4	1: 1909, 3: 568, 4: 291, 2: 132
smoking	0	1	FALSE	3	0: 1763, 1: 845, 2: 292
hypertension	0	1	FALSE	2	0: 1514, 1: 1386
diabetes	0	1	FALSE	2	0: 2446, 1: 454
vaccine	0	1	FALSE	2	1: 1708, 0: 1192
severity	0	1	FALSE	2	0: 2619, 1: 281
study	0	1	FALSE	3	B: 1750, A: 580, C: 570
recovery_time	0	1	FALSE	2	>30: 2013, <=3: 887

Variable type: numeric

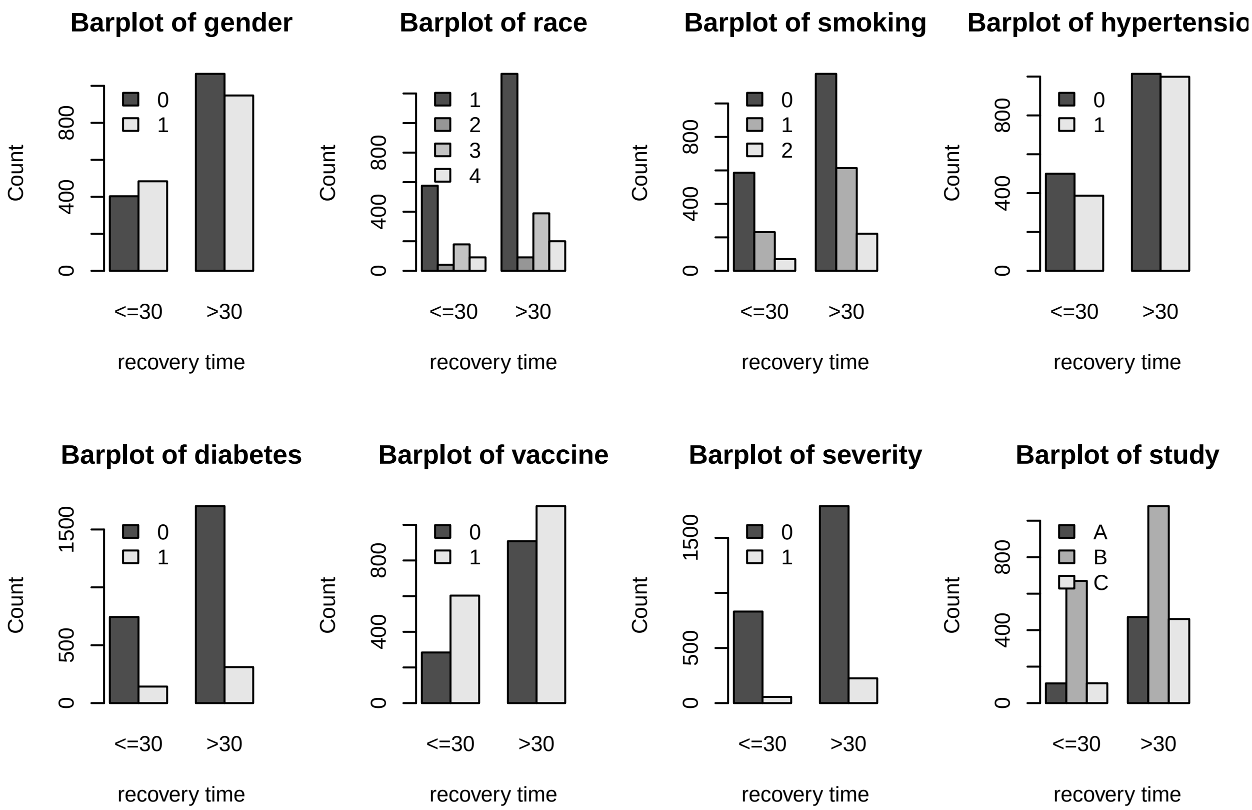
skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
age	0	1	60.07	4.51	45.0	57.0	60.00	63.0	77.0
height	0	1	170.17	6.04	151.2	166.1	170.15	174.1	188.6
weight	0	1	80.20	7.00	57.1	75.4	80.30	84.9	103.4
bmi	0	1	27.76	2.73	19.7	25.9	27.70	29.5	39.8
SBP	0	1	130.19	8.08	104.0	125.0	130.00	136.0	158.0
LDL	0	1	110.27	19.87	32.0	97.0	110.00	124.0	174.0

```r
####################################################################
## Remember to edit the next chunk if you do any modification here:)
####################################################################
# EDA


# boxplot of continuous predictors
par(mfrow=c(2, 3))
for (i in 1:length(cts_var)){
  var = cts_var[i]
  boxplot(train.bin.dat[,var]~recovery_time,
       data = train.bin.dat,
       xlab = "recovery time",
       ylab = var,
       main = str_c("Boxplot of ", var))
}
```

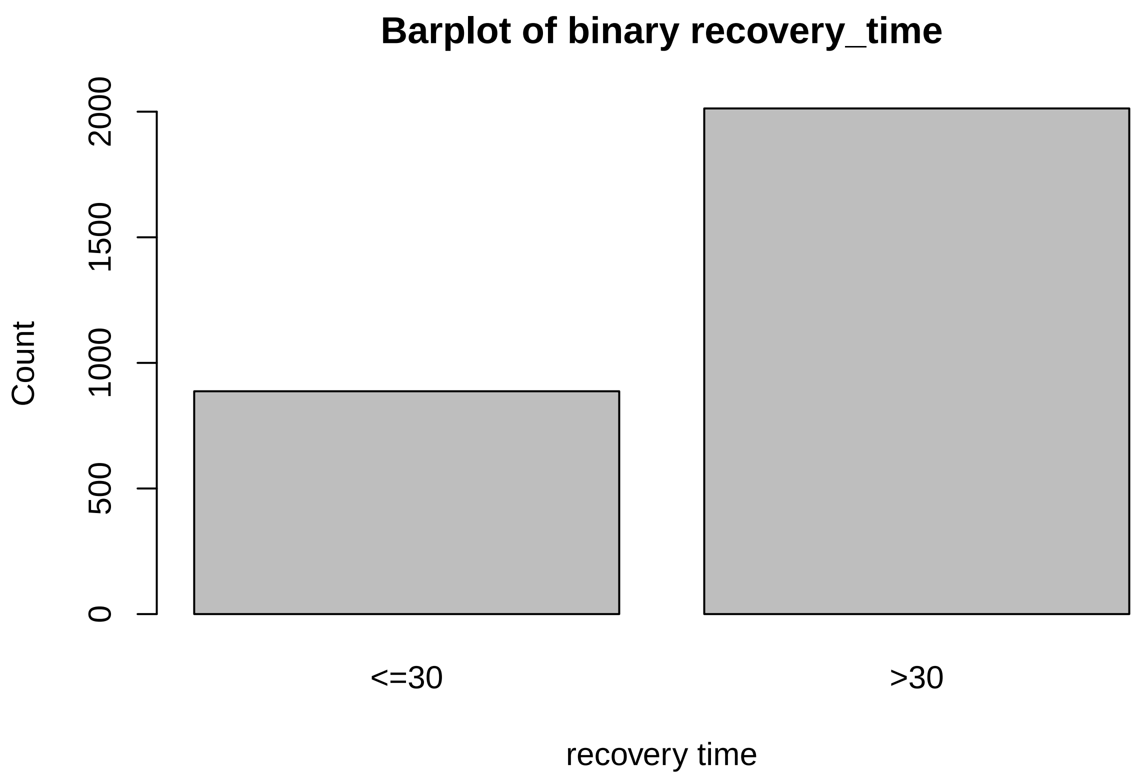



```r
# barplot of categorical predictors
par(mfrow=c(2, 4))
```

```
for (i in 1:length(fct_var)){
  var <- fct_var[i]
  counts <- table(train.bin.dat[,var], train.bin.y)
  barplot(counts, beside = TRUE, legend.text = TRUE,
        xlab = "recovery time",
        ylab = "Count",
        main = str_c("Barplot of ", var),
        args.legend = list(bty = 'n', x = 'topleft'))
}
```




```
# barplot of response
par(mfrow=c(1, 1))
counts <- table(train.bin.y)
barplot(counts,
        xlab = "recovery time",
        ylab = "Count",
        main = "Barplot of binary recovery_time")
```



4.2 Model Training

4.2.1 Logistic Regression

4.2.2 Penalized Logistic Regression

4.2.3 Generalized Additive Model (GAM) for classification

4.2.4 Multivariate Adaptive Regression Splines (MARS) for classification

4.2.5 Linear Discriminant Analysis (LDA)

4.2.6 Quadratic Discriminant Analysis (QDA)

4.2.7 Naive Bayes (NB)

4.2.8 Bagging

4.2.9 Random Forest

4.2.10 Boosting

4.2.11 Classification Trees

4.2.12 Support Vector Machine (SVM)

4.2.13 Hierarchical Clustering

4.2.14 Principal Component Analysis (PCA)

4.3 Model Selection

4.4 Training / Testing Error