Data Science II Midterm Project Analysis

Camille Okonkwo

Contents

```r
library(tidymodels)
library(splines)
library(caret)
library(glmnet)
library(table1)
library(kableExtra)
```

Background

To gain a better understanding of the factors that predict recovery time from COVID-19 illness, a study was designed to combine three existing cohort studies that have been tracking participants for several years. The study collects recovery information through questionnaires and medical records, and leverages existing data on personal characteristics prior to the pandemic. The ultimate goal is to develop a prediction model for recovery time and identify important risk factors for long recovery time.

Data

The dataset in "recovery.RData" includes data from 3000 participants.

Here is a description of each variable:

- ID (`id`): Participant ID
- Gender (`gender`): 1 = Male, 0 = Female
- Race/ethnicity (`race`): 1 = White, 2 = Asian, 3 = Black, 4 = Hispanic
- Smoking (`smoking`): Smoking status; 0 = Never smoked, 1 = Former smoker, 2 = Current smoker
- Height (`height`): Height (in centimeters)
- Weight (`weight`): Weight (in kilograms)
- BMI (`bmi`): Body Mass Index; BMI = weight (in kilograms) / height (in meters) squared
- Hypertension (`hypertension`): 0 = No, 1 = Yes
- Diabetes (`diabetes`): 0 = No, 1 = Yes
- Systolic blood pressure (`SBP`): Systolic blood pressure (in mm/Hg)
- LDL cholesterol (`LDL`): LDL (low-density lipoprotein) cholesterol (in mg/dL)
- Vaccination status at the time of infection (`vaccine`): 0 = Not vaccinated, 1 = Vaccinated
- Severity of COVID-19 infection (`severity`): 0 = Not severe, 1= Severe
- Study (`study`): The study (A/B) that the participant belongs to
- Time to recovery (`recovery_time`): Time from COVID-19 infection to recovery in days

Data Preparation

Partition the dataset into two parts: training data (80%) and test data (20%) with `tidymodels`.

```r
load("data/recovery.RData")

dat = dat |>
  drop_na() |>
  select(-id)

set.seed(2)

# create a random split of 80% training and 20% test data
data_split <- initial_split(data = dat, prop = 0.8)

# partitioned datasets
training_data = training(data_split)
testing_data = testing(data_split)

# training data
x <- model.matrix(recovery_time ~ ., training_data)[, -1] # matrix of predictors
head(x)
y <- training_data$recovery_time # vector of response
```

```
# testing data
x2 <- model.matrix(recovery_time ~ .,testing_data)[, -1] # matrix of predictors
y2 <- testing_data$recovery_time # vector of response
```

Exploratory analysis and data visualization

```
dat_ds <- dat |>
  mutate(across(.fns = as.factor)) |>
  rename_with(~str_to_title(.x), everything()) |>
  mutate(
    Age = as.numeric(Age),
    Gender = factor(Gender, levels = c(0, 1), labels = c("Female", "Male")),
    `Race/Ethnicity` = factor(Race, levels = c(1, 2, 3, 4), labels = c("White", "Asian", "Black", "Hispa
    `Smoking status` = factor(Smoking, levels = c(0, 1, 2), labels = c("Never smoked", "Former smoker",
    Height = as.numeric(Height),
    Weight = as.numeric(Weight),
    `Body Mass Index` = as.numeric(Bmi),
    Hypertension = factor(Hypertension, levels = c(0, 1), labels = c("No", "Yes")),
    Diabetes = factor(Diabetes, levels = c(0, 1), labels = c("No", "Yes")),
    `Systolic Blood Pressure` = as.numeric(Sbp),
    `Low-density lipoprotein cholesterol` = as.numeric(Ldl),
    `Vaccination status at the time of infection` = factor(Vaccine, levels = c(0, 1), labels = c("Not va
    `Severity of COVID-19 infection` = factor(Severity, levels = c(0, 1), labels = c("Not severe", "Seve
    `Time from COVID-19 infection to recovery` = as.numeric(Recovery_time),
    Study = factor(Study, levels = c("A", "B"), labels = c("Study A", "Study B"))
    )
```

Descriptive Statistics Table

```
library(summarytools)

st_options(plain.ascii = FALSE,
           style = "rmarkdown",
           dfSummary.silent = TRUE,
           footnote = NA,
           subtitle.emphasis = FALSE)

dfSummary(dat)
```

```
## ### Data Frame Summary
## **dat**
## **Dimensions:** 3000 x 15
## **Duplicates:** 0
##
## ----------------------------------------------------------------------------------------------------
## No   Variable         Stats / Values              Freqs (% of Valid)    Graph                    Val
## ---- ---------------- --------------------------- --------------------- ------------------------ ---
## 1    age\             Mean (sd) : 60.2 (4.5)\     34 distinct values    \ \ \ \ \ \ : .\         300
##      [numeric]        min < med < max:\                                 \ \ \ \ \ \ : :\         (10
##                       42 < 60 < 79\                                     \ \ \ \ \ \ : :\
##                       IQR (CV) : 6 (0.1)                                \ \ \ \ . : : .\
##                                                                         \ \ \ \ : : : :
##
## 2    gender\          Min   : 0\                  0 : 1544 (51.5%)\     IIIIIIIIII \             300
##      [integer]        Mean : 0.5\                 1 : 1456 (48.5%)      IIIIIIIII                (10
##                       Max   : 1
##
```

```
## 3    race\            1\. 1\                      1967 (65.6%)\         IIIIIIIIIIIII \          3000
##      [factor]         2\. 2\                      158 ( 5.3%)\          I \                      (100
##                       3\. 3\                      604 (20.1%)\          IIII \
##                       4\. 4                       271 ( 9.0%)           I
##
## 4    smoking\         1\. 0\                      1822 (60.7%)\         IIIIIIIIIIII \           3000
##      [factor]         2\. 1\                      859 (28.6%)\          IIIII \                  (100
##                       3\. 2                       319 (10.6%)           II
##
## 5    height\          Mean (sd) : 169.9 (6)\      313 distinct values   \ \ \ \ \ \ \ \ : :\     3000
##      [numeric]        min < med < max:\                                 \ \ \ \ \ \ \ \ : :\     (100
##                       147.8 < 169.9 < 188.6\                            \ \ \ \ \ \ . : : .\
##                       IQR (CV) : 7.9 (0)                                \ \ \ \ \ \ : : : :\
##                                                                         \ \ \ \ . : : : : .
##
## 6    weight\          Mean (sd) : 80 (7.1)\       364 distinct values   \ \ \ \ \ \ \ \ : .\     3000
##      [numeric]        min < med < max:\                                 \ \ \ \ \ \ \ \ : :\     (100
##                       55.9 < 79.8 < 103.7\                              \ \ \ \ \ \ : : : :\
##                       IQR (CV) : 9.6 (0.1)                              \ \ \ \ . : : : : .\
##                                                                         \ \ . : : : : : : .
##
## 7    bmi\             Mean (sd) : 27.8 (2.8)\     163 distinct values   \ \ \ \ \ \ . :\         3000
##      [numeric]        min < med < max:\                                 \ \ \ \ \ \ : : :\       (100
##                       18.8 < 27.6 < 38.9\                               \ \ \ \ \ \ : : :\
##                       IQR (CV) : 3.7 (0.1)                              \ \ \ \ : : : : :\
##                                                                         \ \ . : : : : : .
##
## 8    hypertension\    Min  : 0\                   0 : 1508 (50.3%)\     IIIIIIIIII \             3000
##      [numeric]        Mean : 0.5\                 1 : 1492 (49.7%)      IIIIIIIII                (100
##                       Max  : 1
##
## 9    diabetes\        Min  : 0\                   0 : 2537 (84.6%)\     IIIIIIIIIIIIIIII \       3000
##      [integer]        Mean : 0.2\                 1 :  463 (15.4%)      III                      (100
##                       Max  : 1
##
## 10   SBP\             Mean (sd) : 130.5 (8)\      52 distinct values    \ \ \ \ \ \ \ \ : .\     3000
##      [numeric]        min < med < max:\                                 \ \ \ \ \ \ \ \ : : .\   (100
##                       105 < 130 < 156\                                  \ \ \ \ \ \ : : : :\
##                       IQR (CV) : 11 (0.1)                               \ \ \ \ . : : : : .\
##                                                                         \ \ . : : : : : : .
##
## 11   LDL\             Mean (sd) : 110.5 (19.8)\   114 distinct values   \ \ \ \ \ \ \ \ \ \ :\   3000
##      [numeric]        min < med < max:\                                 \ \ \ \ \ \ \ \ : : .\   (100
##                       28 < 110 < 178\                                   \ \ \ \ \ \ \ \ : : :\
##                       IQR (CV) : 27 (0.2)                               \ \ \ \ \ \ . : : : .\
##                                                                         \ \ \ \ . : : : : : .
##
## 12   vaccine\         Min  : 0\                   0 : 1212 (40.4%)\     IIIIIIII \               3000
##      [integer]        Mean : 0.6\                 1 : 1788 (59.6%)      IIIIIIIIIII              (100
##                       Max  : 1
##
## 13   severity\        Min  : 0\                   0 : 2679 (89.3%)\     IIIIIIIIIIIIIIIII \      3000
##      [integer]        Mean : 0.1\                 1 :  321 (10.7%)      II                       (100
##                       Max  : 1
```

```
##
## 14   study\           1\. A\                      2000 (66.7%)\         IIIIIIIIIIIII \         300
##      [character]      2\. B                       1000 (33.3%)          IIIIII                   (10
##
## 15   recovery_time\   Mean (sd) : 42.2 (23.2)\    140 distinct values   : :\                     300
##      [numeric]        min < med < max:\                                 : :\                     (10
##                       2 < 39 < 365\                                     : :\
##                       IQR (CV) : 18 (0.5)                               : :\
##                                                                         : : .
## ----------------------------------------------------------------------------------------------------
```

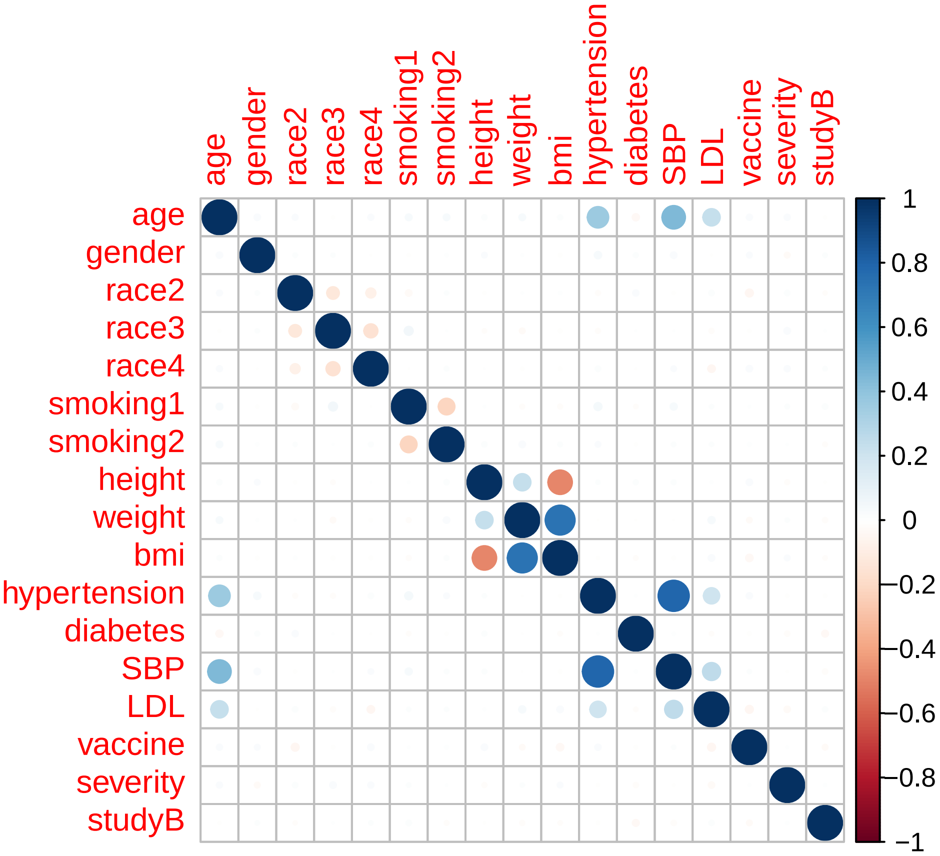
```r
library(table1)
library(kableExtra)

units(dat_ds$Height) <- "cm"
units(dat_ds$Weight) <- "kg"
units(dat_ds$`Body Mass Index`) <- "kg/m^2"
units(dat_ds$`Systolic Blood Pressure`) <- "mm/Hg"
units(dat_ds$`Low-density lipoprotein cholesterol`) <- "mg/dL"
units(dat_ds$`Time from COVID-19 infection to recovery`) <- "days"

descriptive_table <- table1(~ Age + Gender + `Race/Ethnicity` + `Smoking status` + Height + Weight + `Bo
                            data = dat_ds,
                            overall = "Total",
                            caption = "Descriptive Statistics")

t1kable(descriptive_table)
```

Correlation matrix of training data

```r
library(corrplot)
corrplot(cor(x), method = "circle", type = "full")
```

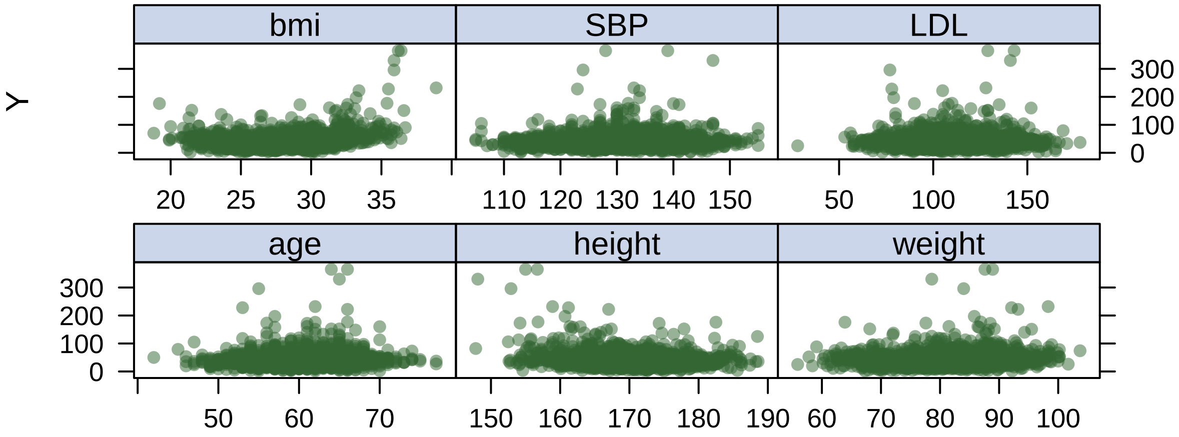
Table 1: Descriptive Statistics

	Study A	Study B	Total
	(N=2000)	(N=1000)	(N=3000)
Age			
Mean (SD)	17.2 (4.52)	17.2 (4.38)	17.2 (4.47)
Median [Min, Max]	17.0 [1.00, 34.0]	17.0 [2.00, 33.0]	17.0 [1.00, 34.0]
Gender			
Female	1036 (51.8%)	508 (50.8%)	1544 (51.5%)
Male	964 (48.2%)	492 (49.2%)	1456 (48.5%)
Race/Ethnicity			
White	1312 (65.6%)	655 (65.5%)	1967 (65.6%)
Asian	108 (5.4%)	50 (5.0%)	158 (5.3%)
Black	408 (20.4%)	196 (19.6%)	604 (20.1%)
Hispanic	172 (8.6%)	99 (9.9%)	271 (9.0%)
Smoking status			
Never smoked	1225 (61.3%)	597 (59.7%)	1822 (60.7%)
Former smoker	557 (27.9%)	302 (30.2%)	859 (28.6%)
Current smoker	218 (10.9%)	101 (10.1%)	319 (10.6%)
Height (cm)			
Mean (SD)	160 (58.8)	161 (59.1)	160 (58.9)
Median [Min, Max]	160 [1.00, 313]	161 [2.00, 312]	160 [1.00, 313]
Weight (kg)			
Mean (SD)	181 (70.0)	182 (70.5)	182 (70.2)
Median [Min, Max]	178 [1.00, 364]	182 [3.00, 358]	180 [1.00, 364]
Body Mass Index (kg/m^2)			
Mean (SD)	77.6 (27.5)	77.6 (28.3)	77.6 (27.8)
Median [Min, Max]	77.0 [1.00, 162]	76.0 [2.00, 163]	76.5 [1.00, 163]
Hypertension			
No	998 (49.9%)	510 (51.0%)	1508 (50.3%)
Yes	1002 (50.1%)	490 (49.0%)	1492 (49.7%)
Diabetes			
No	1678 (83.9%)	859 (85.9%)	2537 (84.6%)
Yes	322 (16.1%)	141 (14.1%)	463 (15.4%)
Systolic Blood Pressure (mm/Hg)			
Mean (SD)	26.6 (8.02)	26.3 (7.88)	26.5 (7.97)
Median [Min, Max]	27.0 [1.00, 52.0]	26.0 [1.00, 51.0]	26.0 [1.00, 52.0]
Low-density lipoprotein cholesterol (mg/dL)			
Mean (SD)	58.3 (19.7)	58.7 (19.7)	58.4 (19.7)
Median [Min, Max]	58.0 [1.00, 114]	58.0 [3.00, 112]	58.0 [1.00, 114]
Vaccination status at the time of infection			
Not vaccinated	797 (39.9%)	415 (41.5%)	1212 (40.4%)
Vaccinated	1203 (60.2%)	585 (58.5%)	1788 (59.6%)
Severity of COVID-19 infection			
Not severe	1785 (89.3%)	894 (89.4%)	2679 (89.3%)
Severe	215 (10.8%)	106 (10.6%)	321 (10.7%)
Time from COVID-19 infection to recovery (days)			
Mean (SD)	39.4 (11.1)	42.8 (28.1)	40.5 (18.7)
Median [Min, Max]	39.0 [9.00, 107]	36.0 [1.00, 140]	38.0 [1.00, 140]



Feature Plot of continuous variables

```
theme1 <- trellis.par.get()
theme1$plot.symbol$col <- rgb(.2, .4, .2, .5)
theme1$plot.symbol$pch <- 16
theme1$plot.line$col <- rgb(.8, .1, .1, 1)
theme1$plot.line$lwd <- 2
theme1$strip.background$col <- rgb(.0, .2, .6, .2)
trellis.par.set(theme1)

featurePlot(
  x[, -c(2, 3, 4, 5, 6, 7, 11, 12, 15, 16, 17) ],
  y,
  plot = "scatter",
  labels = c("", "Y"),
  type = c("p"),
  layout = c(3, 3))
```



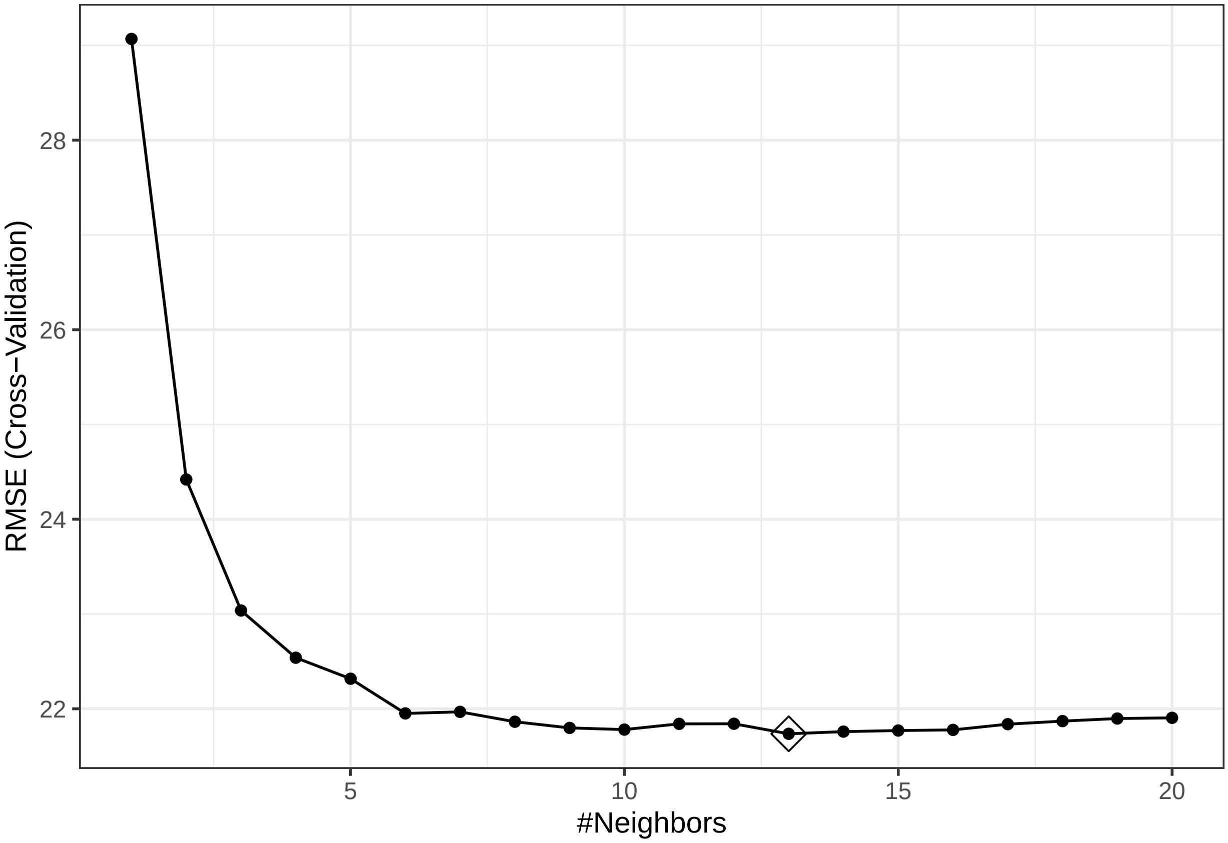
Seems to be mostly linear, with some outliers.

Model Fitting in caret

```
# setting a 10-fold cross-validation
ctrl <- trainControl(method = "cv",
                     number = 10,
                     selectionFunction = "best")
```

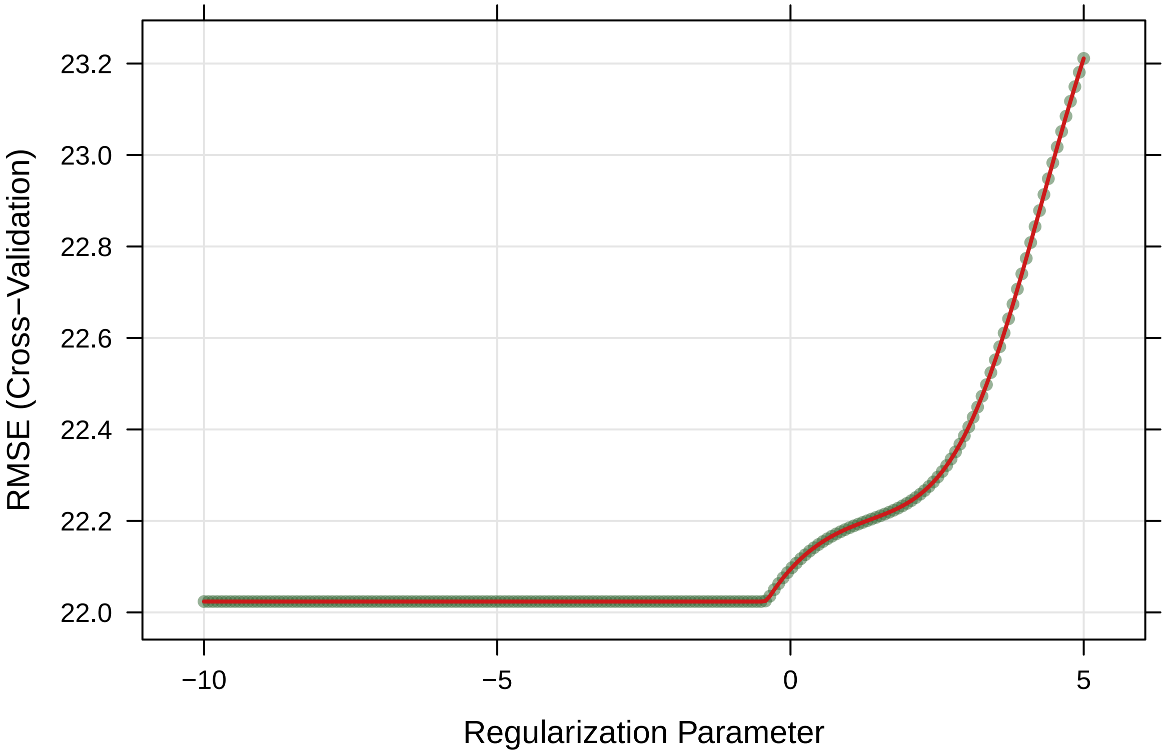
KNN

```
# knn using `caret`
set.seed(2)

knn.fit <- train(x, y,
                 method = "knn",
                 trControl = ctrl,
                 tuneGrid = expand.grid(k = seq(from = 1, to = 20, by = 1)))

ggplot(knn.fit, highlight = TRUE) + theme_bw()
```



Ridge Regression

```
# ridge using `caret`
set.seed(2)

ridge.fit <- train(x, y,
                   method = "glmnet",
                   tuneGrid = expand.grid(alpha = 0,
                                          lambda = exp(seq(5, -10, length=200))),
                   trControl = ctrl)

plot(ridge.fit, xTrans = log)
```



```
ridge.fit$bestTune
```

```
##     alpha    lambda
## 127     0 0.6050086
```

```
# coefficients in the final model
coef(ridge.fit$finalModel, s = ridge.fit$bestTune$lambda)
```

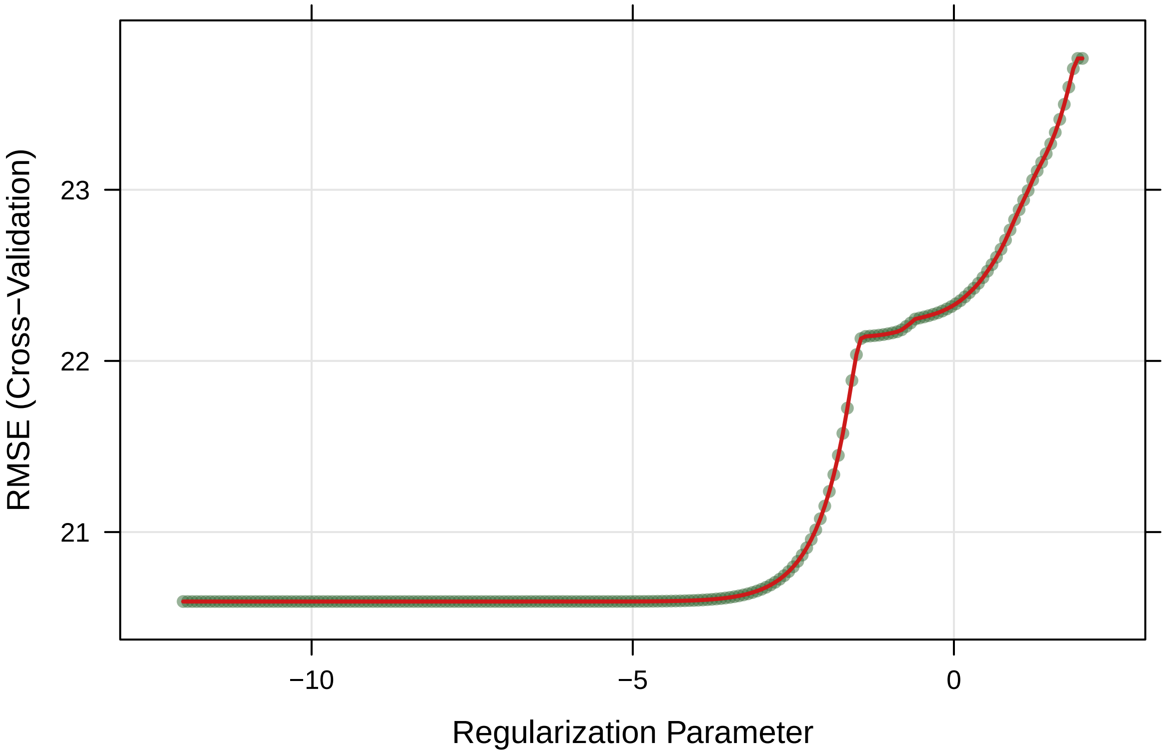
```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)  -117.01411494
## age             0.19251501
## gender         -2.48566998
## race2           2.10893672
## race3          -1.55369621
## race4          -0.65563498
## smoking1        2.42795618
## smoking2        2.89381550
## height          0.52869977
## weight         -0.89924951
## bmi             4.44256063
## hypertension    2.31676844
## diabetes       -2.04500225
## SBP             0.08663018
## LDL            -0.03359715
## vaccine        -6.87690608
## severity        8.27607544
## studyB          5.74811996
```

```
ridge.pred <- predict(ridge.fit, newdata = model.matrix(recovery_time ~ ., testing_data)[,-1])

# test error
mean((ridge.pred - testing_data[, "recovery_time"])^2)
```

```
## [1] 336.6451
```

Lasso

```
set.seed(2)

# lasso using caret
lasso.fit <- train(x, y,
                   method = "glmnet",
                   tuneGrid = expand.grid(alpha = 1,
                                          lambda = exp(seq(2, -12, length=200))),
                   trControl = ctrl)

plot(lasso.fit, xTrans = log)
```



```
lasso.fit$bestTune
```

```
##    alpha      lambda
## 75     1 0.001120512
```

```
# coefficients in the final model
coef(lasso.fit$finalModel, lasso.fit$bestTune$lambda)
```

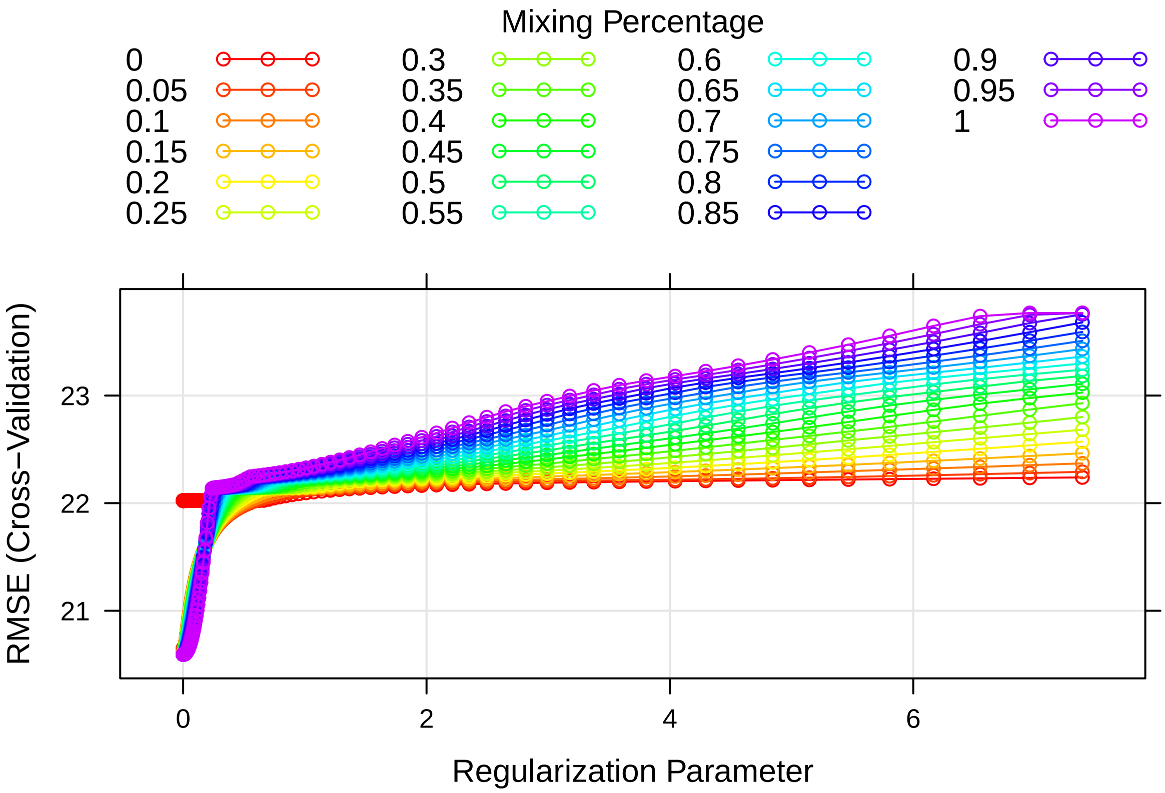
```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)  -2.325013e+03
## age           2.005281e-01
## gender       -2.878633e+00
## race2         1.293972e+00
## race3        -1.832518e+00
## race4        -2.147234e-02
## smoking1      2.506334e+00
## smoking2      2.819656e+00
## height        1.355073e+01
## weight       -1.468902e+01
## bmi           4.406210e+01
## hypertension  2.133432e+00
## diabetes     -1.615766e+00
## SBP           7.587321e-02
## LDL          -3.832476e-02
## vaccine      -6.808441e+00
## severity      8.205208e+00
## studyB        5.720852e+00
```

Elastic Net (???)

```r
set.seed(2)

# elastic net using caret
enet.fit <- train(x, y,
                   method = "glmnet",
                   tuneGrid = expand.grid(alpha = seq(0, 1, length = 21),
                                          lambda = exp(seq(2, -10, length=200))),
                   trControl = ctrl)

enet.fit$bestTune
```

```
##      alpha      lambda
## 1659   0.4 0.001499666
```

```r
myCol <- rainbow(25)
myPar <- list(superpose.symbol = list(col = myCol),
              superpose.line = list(col = myCol))
plot(enet.fit, par.settings = myPar)
```



```r
# coefficients in the final model
coef(enet.fit$finalModel, enet.fit$bestTune$lambda)
```

```
## 18 x 1 sparse Matrix of class "dgCMatrix"
##                          s1
## (Intercept)  -2.287199e+03
## age           2.005688e-01
## gender       -2.874084e+00
## race2         1.310441e+00
## race3        -1.830401e+00
## race4        -3.555828e-02
## smoking1      2.507871e+00
## smoking2      2.825084e+00
## height        1.332763e+01
## weight       -1.445287e+01
## bmi           4.338401e+01
## hypertension  2.138628e+00
## diabetes     -1.625301e+00
## SBP           7.607963e-02
## LDL          -3.832161e-02
## vaccine      -6.813749e+00
## severity      8.211536e+00
## studyB        5.725047e+00
```

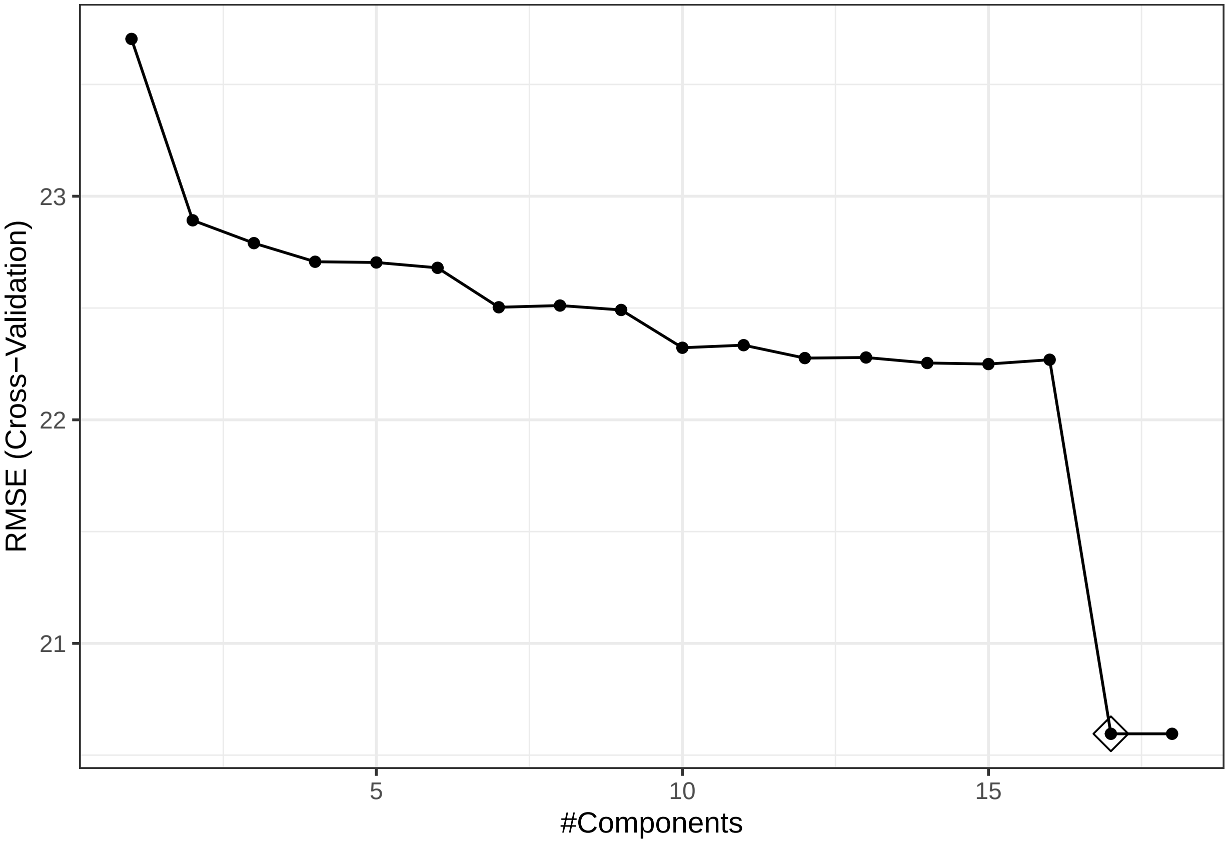
PCR

```r
set.seed(2)

# pcr using caret
```

```
pcr.fit <- train(x, y,
                 method = "pcr",
                 tuneGrid = data.frame(ncomp = 1:18),
                 trControl = ctrl,
                 preProcess = c("center", "scale"))

predy2.pcr2 <- predict(pcr.fit, newdata = x2)

mean((y2 - predy2.pcr2)^2)
```

```
## [1] 327.5411
```

```
ggplot(pcr.fit, highlight = TRUE) + theme_bw()
```



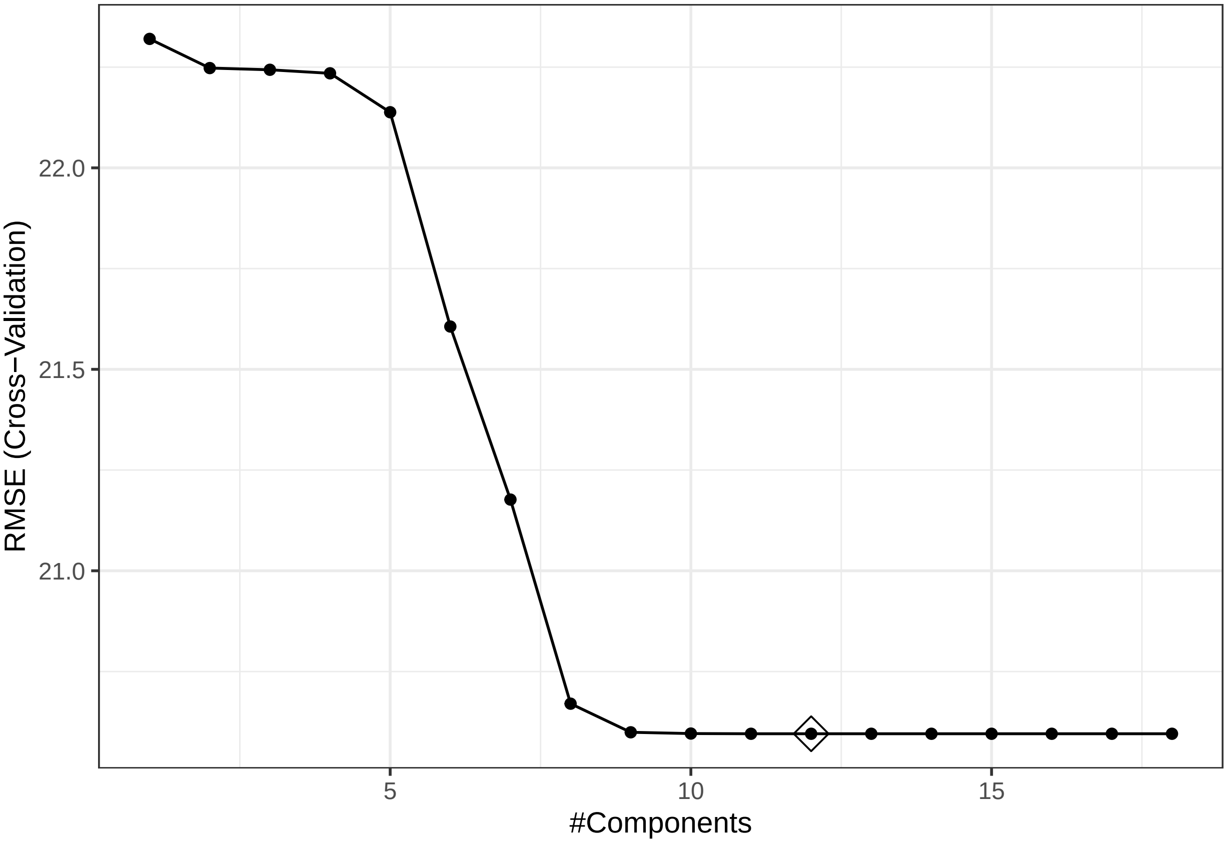
PLS

```
set.seed(2)

# pls using caret
pls.fit <- train(x, y,
                 method = "pls",
                 tuneGrid = data.frame(ncomp = 1:18),
                 trControl = ctrl,
                 preProcess = c("center", "scale"))

predy2.pls2 <- predict(pls.fit, newdata = x2)
```

```
mean((y2 - predy2.pls2)^2)
```

```
## [1] 327.5415
```

```
ggplot(pls.fit, highlight = TRUE) + theme_bw()
```



GAM

```
set.seed(2)

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

gam.fit$bestTune
```

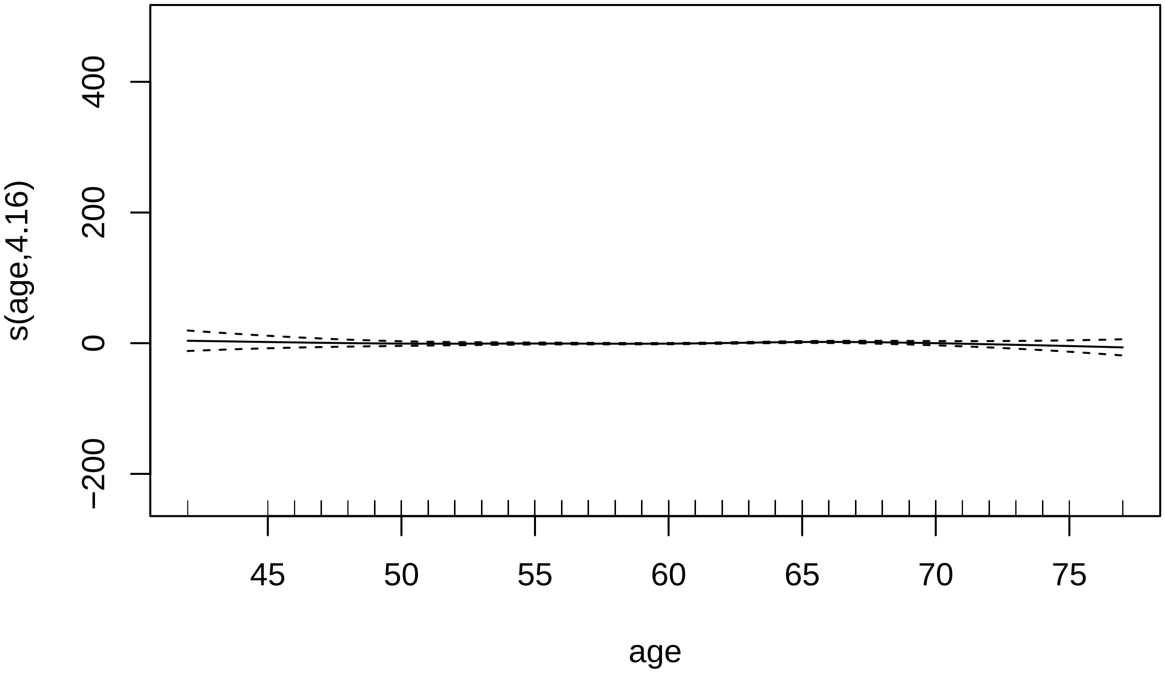
```
##   select method
## 1  FALSE GCV.Cp
```

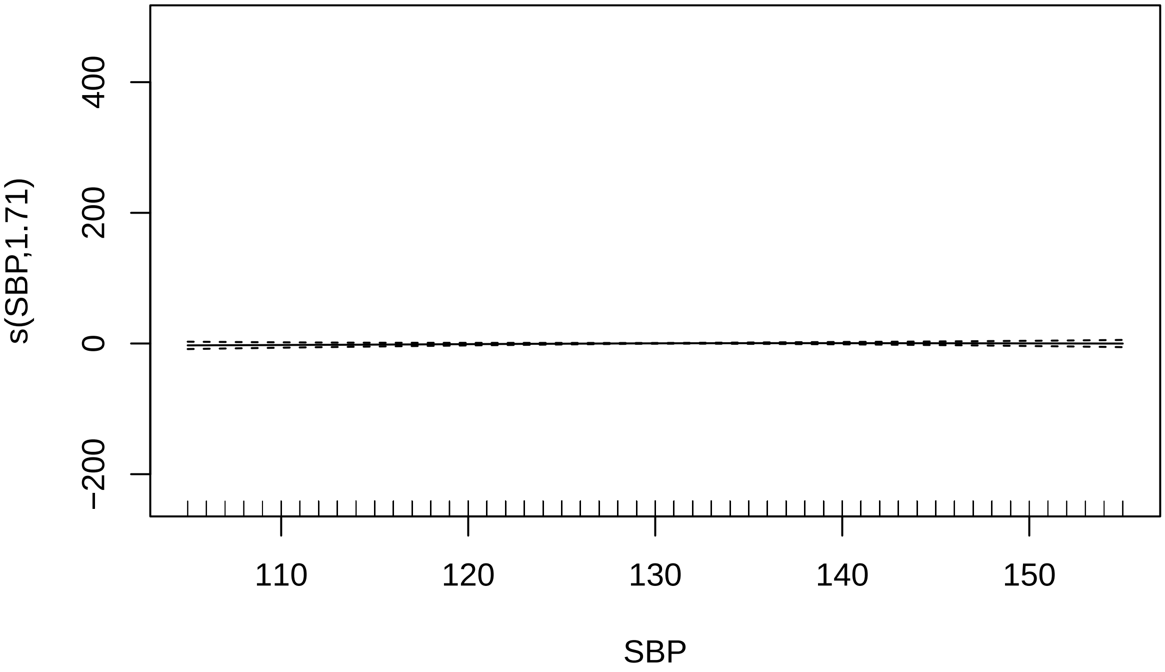
```
gam.fit$finalModel
```

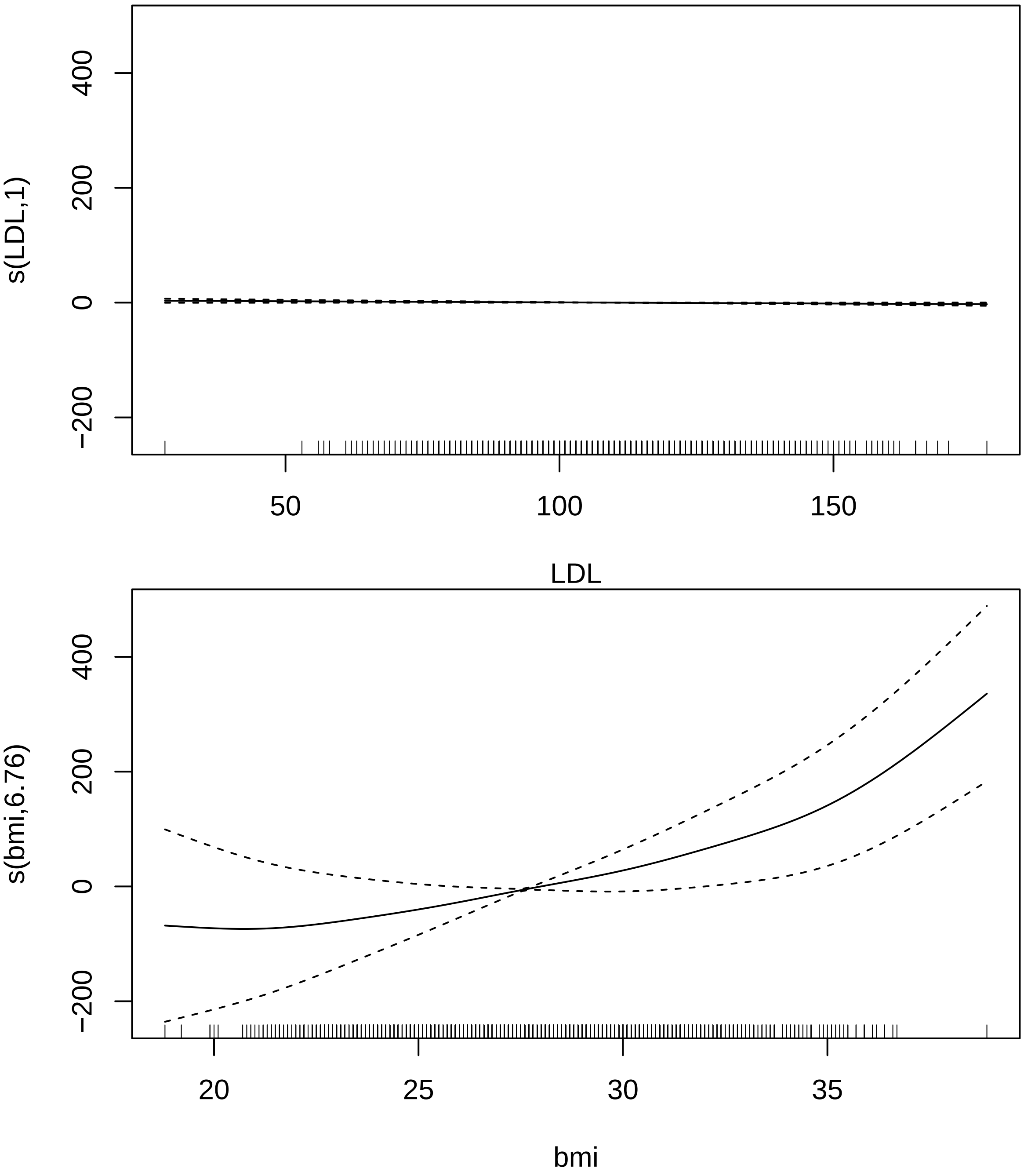
```
##
## Family: gaussian
## Link function: identity
##
## Formula:
```

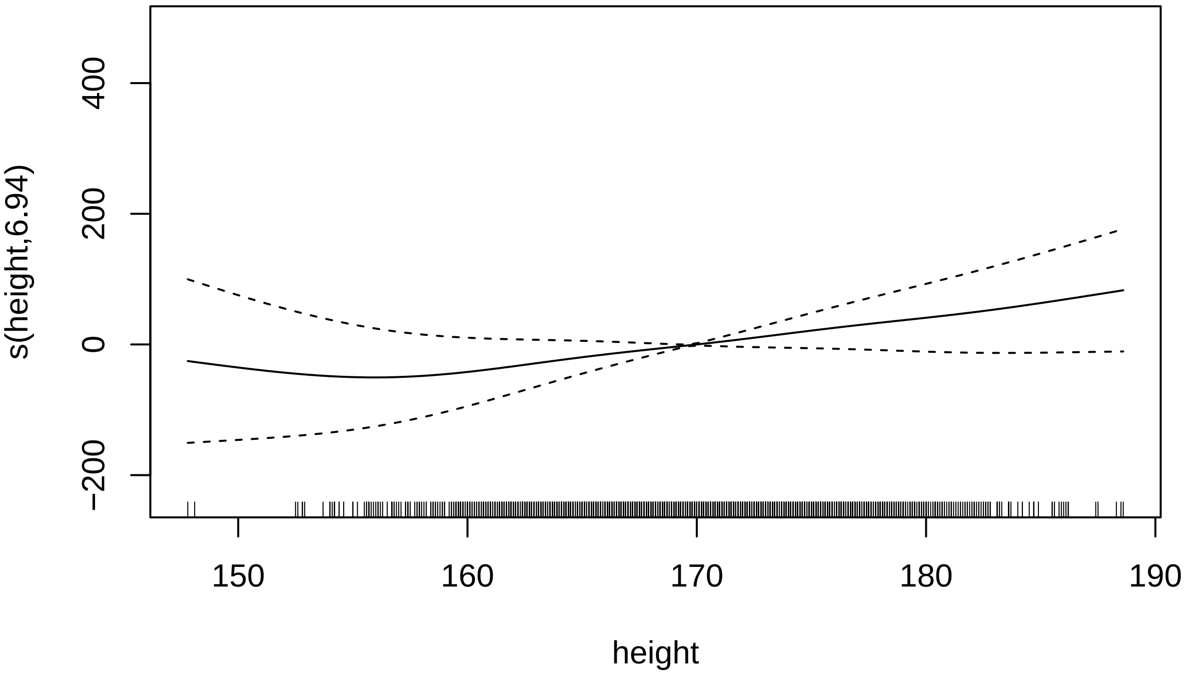
```
## .outcome ~ gender + race2 + race3 + race4 + smoking1 + smoking2 +
##     hypertension + diabetes + vaccine + severity + studyB + s(age) +
##     s(SBP) + s(LDL) + s(bmi) + s(height) + s(weight)
##
## Estimated degrees of freedom:
## 4.16 1.71 1.00 6.76 6.94 3.46  total = 36.02
##
## GCV score: 375.7389
```

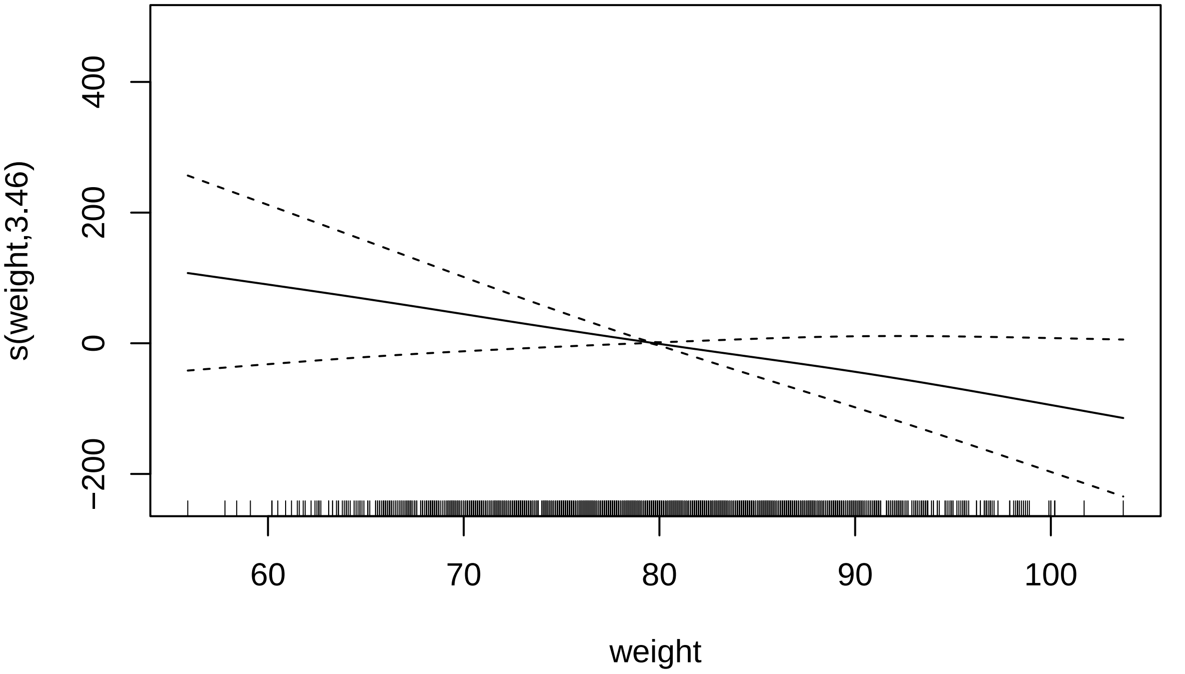
```
plot(gam.fit$finalModel)
```





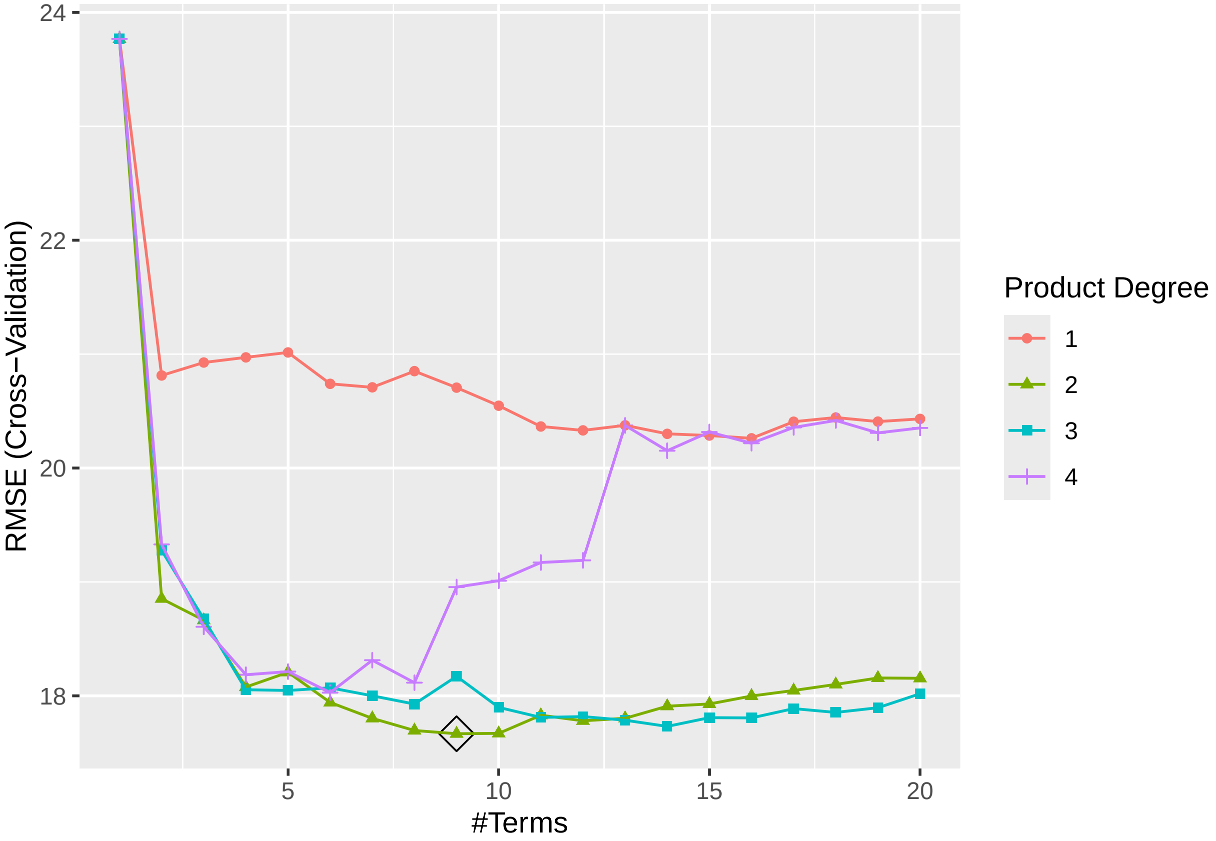

s(LDL,1)
400
200
0
−200
50
100
150
LDL
s(bmi,6.76)
400
200
0
−200
20
25
30
35
bmi





MARS

```
# set grid
mars_grid <- expand.grid(degree = 1:4, nprune = 1:20)

set.seed(2)

# fit a MARS model
mars.fit <- train(x, y,
                  method = "earth",
                   tuneGrid = mars_grid,
                   trControl = ctrl)
# plot
ggplot(mars.fit, highlight = TRUE)
```



```
# best tuning parameters
mars.fit$bestTune
```

```
##    nprune degree
## 29      9      2
```

```
# regression function
mars.fit$finalModel
```

```
## Selected 9 of 27 terms, and 6 of 17 predictors (nprune=9)
## Termination condition: Reached nk 35
## Importance: bmi, studyB, height, vaccine, severity, weight, age-unused, ...
## Number of terms at each degree of interaction: 1 3 5
## GCV 298.3891    RSS 703656.4    GRSq 0.4847193    RSq 0.493275
```

```
# report the regression function
summary(mars.fit)
```

```
## Call: earth(x=matrix[2400,17], y=c(30,39,9,40,50...), keepxy=TRUE, degree=2,
##             nprune=9)
##
##                               coefficients
## (Intercept)                     -5.2669231
## vaccine                         -6.3585662
## h(bmi-24.5)                     7.7118874
## h(30.9-bmi)                     6.9728117
## h(bmi-24.5) * severity          1.8498609
## h(bmi-30.9) * studyB           25.7460428
## h(159-height) * h(bmi-30.9)     2.8177334
## h(85.1-weight) * h(bmi-30.9)   -2.7210715
## h(weight-85.1) * h(bmi-30.9)   -0.4065098
##
## Selected 9 of 27 terms, and 6 of 17 predictors (nprune=9)
## Termination condition: Reached nk 35
## Importance: bmi, studyB, height, vaccine, severity, weight, age-unused, ...
## Number of terms at each degree of interaction: 1 3 5
## GCV 298.3891    RSS 703656.4    GRSq 0.4847193    RSq 0.493275
```

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

```
##                  (Intercept)                  h(30.9-bmi)
##                   -5.2669231                    6.9728117
##         h(bmi-30.9) * studyB                  h(bmi-24.5)
##                   25.7460428                    7.7118874
##  h(159-height) * h(bmi-30.9)                      vaccine
##                    2.8177334                   -6.3585662
##       h(bmi-24.5) * severity h(weight-85.1) * h(bmi-30.9)
##                    1.8498609                   -0.4065098
## h(85.1-weight) * h(bmi-30.9)
##                   -2.7210715
```

```
# test error
pred.mars <- predict(mars.fit, newdata = testing_data)

test.error.mars <- mean((pred.mars - y2)^2)
```

Linear Model

```
set.seed(2)

# fit a linear model
lm.fit <- train(x, y,
                method = "lm",
                trControl = ctrl)

summary(lm.fit)
```

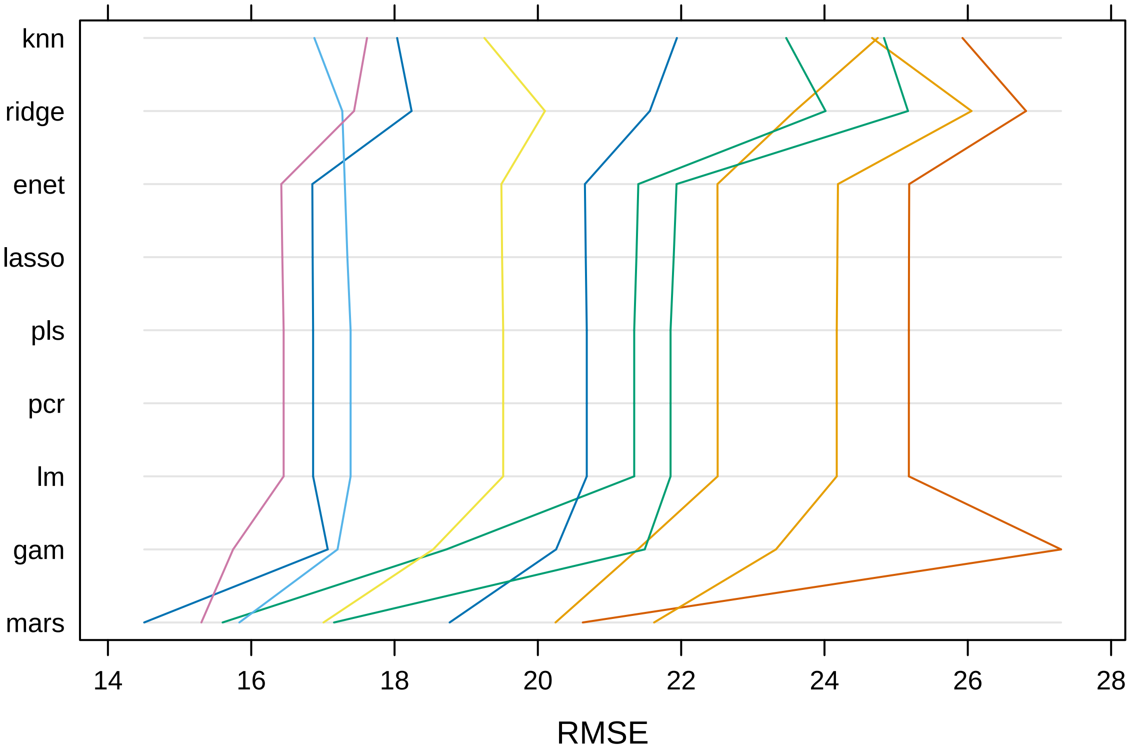
```
##
## Call:
## lm(formula = .outcome ~ ., data = dat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -57.444 -11.456   0.078   8.816 252.382
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.368e+03  1.193e+02 -19.843  < 2e-16 ***
## age           2.008e-01  1.049e-01   1.914 0.055742 .
## gender       -2.888e+00  8.455e-01  -3.415 0.000648 ***
## race2         1.282e+00  1.851e+00   0.692 0.488787
## race3        -1.841e+00  1.086e+00  -1.695 0.090281 .
## race4        -1.404e-02  1.521e+00  -0.009 0.992636
## smoking1      2.510e+00  9.554e-01   2.627 0.008662 **
## smoking2      2.821e+00  1.412e+00   1.998 0.045837 *
## height        1.381e+01  7.001e-01  19.719  < 2e-16 ***
## weight       -1.496e+01  7.395e-01 -20.227  < 2e-16 ***
## bmi           4.483e+01  2.122e+00  21.127  < 2e-16 ***
## hypertension  2.129e+00  1.400e+00   1.521 0.128489
## diabetes     -1.610e+00  1.173e+00  -1.372 0.170040
## SBP           7.581e-02  9.169e-02   0.827 0.408396
## LDL          -3.848e-02  2.252e-02  -1.709 0.087655 .
## vaccine      -6.807e+00  8.638e-01  -7.880 4.95e-15 ***
## severity      8.204e+00  1.360e+00   6.030 1.89e-09 ***
## studyB        5.720e+00  8.989e-01   6.363 2.36e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 20.65 on 2382 degrees of freedom
## Multiple R-squared:  0.2684, Adjusted R-squared:  0.2632
## F-statistic:  51.4 on 17 and 2382 DF,  p-value: < 2.2e-16
```

Model Comparison

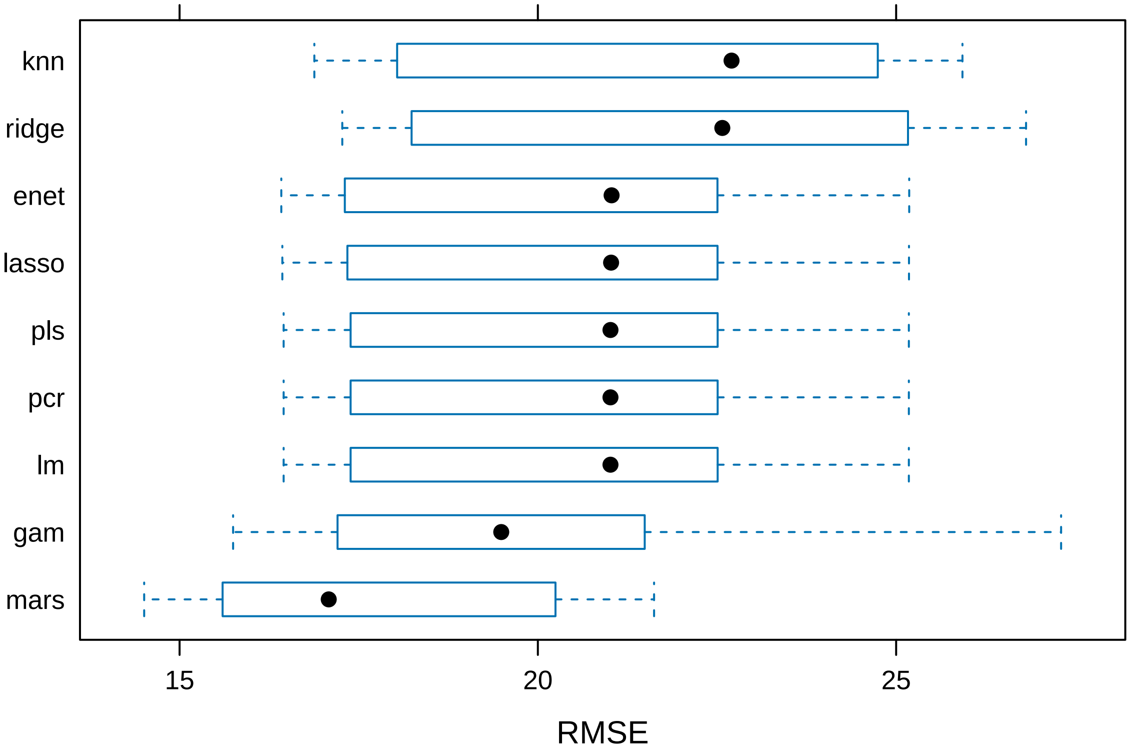
```r
# compare models
resamp <- resamples(list(knn = knn.fit, ridge = ridge.fit, lasso = lasso.fit, enet = enet.fit, pcr = pc

summary(resamp)
```

```
##
## Call:
## summary.resamples(object = resamp)
##
## Models: knn, ridge, lasso, enet, pcr, pls, gam, mars, lm
## Number of resamples: 10
##
## MAE
##           Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## knn   11.89451 12.53209 13.13301 13.23721 13.94318 14.87500    0
## ridge 11.95892 13.09153 13.81815 13.59444 14.22836 14.43166    0
## lasso 12.44244 12.93115 13.69204 13.61849 14.19721 14.85341    0
## enet  12.42711 12.89477 13.68552 13.59745 14.17737 14.81495    0
## pcr   12.46114 12.97479 13.70124 13.64347 14.22029 14.89574    0
## pls   12.46114 12.97480 13.70124 13.64347 14.22027 14.89572    0
## gam   11.87037 12.28822 12.85254 13.00140 13.67807 14.55438    0
## mars  11.02436 11.14337 11.96777 12.01770 12.77086 13.63149    0
## lm    12.46114 12.97479 13.70124 13.64347 14.22029 14.89574    0
##
## RMSE
##           Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## knn   16.88052 18.34044 22.70299 21.73570 24.72416 25.92585    0
## ridge 17.27041 18.70308 22.57431 22.02367 24.87808 26.81337    0
## lasso 16.43374 17.88156 21.02217 20.59409 22.35467 25.17967    0
## enet  16.41942 17.85202 21.02924 20.59399 22.36377 25.18257    0
## pcr   16.45215 17.91897 21.01357 20.59555 22.34488 25.17761    0
## pls   16.45213 17.91897 21.01358 20.59555 22.34487 25.17760    0
## gam   15.74704 17.53776 19.48885 20.10459 21.46701 27.30243    0
## mars  14.50559 15.65849 17.08160 17.66700 19.87678 21.62199    0
## lm    16.45215 17.91897 21.01357 20.59555 22.34488 25.17761    0
##
## Rsquared
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## knn   0.11142603 0.1412739 0.1638373 0.1795177 0.2226442 0.2695723    0
## ridge 0.08418902 0.1242735 0.1453962 0.1492394 0.1759793 0.2068496    0
## lasso 0.15543099 0.2184218 0.2605332 0.2622150 0.2777511 0.3795018    0
## enet  0.15505238 0.2190095 0.2601768 0.2620470 0.2777216 0.3786877    0
## pcr   0.15576465 0.2176829 0.2604864 0.2623602 0.2779419 0.3804001    0
## pls   0.15576657 0.2176826 0.2604868 0.2623603 0.2779424 0.3804004    0
## gam   0.18735879 0.2446136 0.3118875 0.3102145 0.3492249 0.5108512    0
## mars  0.32424416 0.3755638 0.4198455 0.4474544 0.4692360 0.6618530    0
## lm    0.15576465 0.2176829 0.2604864 0.2623602 0.2779419 0.3804001    0
```

```r
parallelplot(resamp, metric = "RMSE")
```



```r
bwplot(resamp, metric = "RMSE")
```



MARS has lowest mean and median RMSE -> model I pick