Build Prediction Model

Minghe Wang, Zebang Zhang(zz3309)

2025-03-27

Exploratory Data Analysis

```r
load("./data/dat1.RData")
load("./data/dat2.RData")

# no missing data
all(is.na(dat1))
```

```
## [1] FALSE
```

```r
all(is.na(dat2))
```

```
## [1] FALSE
```

```r
ifelse(all(names(dat1) == names(dat2)), "train and test data have same structure", "train and test data
```

```
## [1] "train and test data have same structure"
```

```r
str(dat1)
```

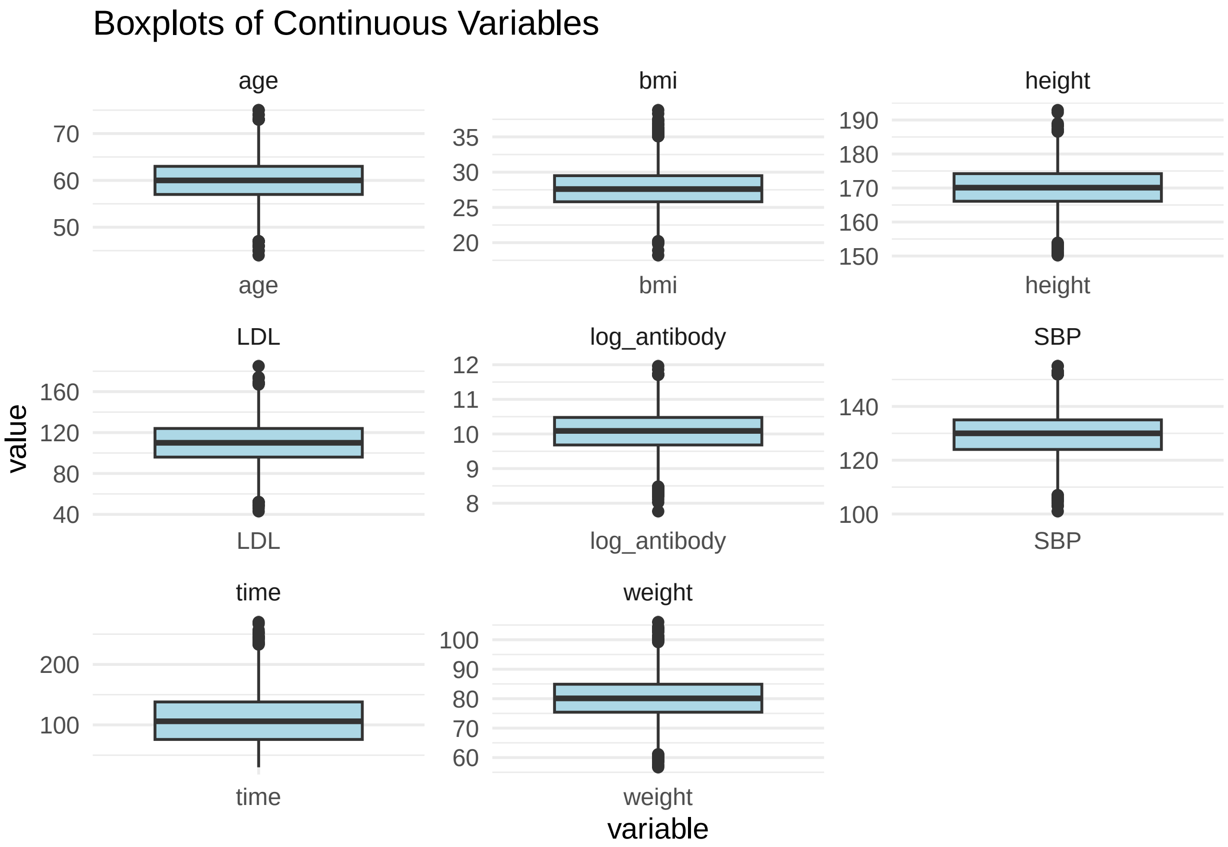
```
## 'data.frame':    5000 obs. of  14 variables:
##  $ id          : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ age         : num  50 71 58 63 56 59 67 62 60 64 ...
##  $ gender      : int  0 1 1 0 1 1 0 1 0 1 ...
##  $ race        : Factor w/ 4 levels "1","2","3","4": 1 1 1 1 1 3 4 1 4 1 ...
##  $ smoking     : Factor w/ 3 levels "0","1","2": 1 1 2 1 1 1 1 1 1 1 ...
##  $ height      : num  176 176 169 167 163 ...
##  $ weight      : num  68.3 69.6 76.9 90 83.9 86.8 91.4 87.7 85.7 76.6 ...
##  $ bmi         : num  22 22.6 27 32.1 31.7 30.8 29.7 28.1 29 31.5 ...
##  $ diabetes    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ hypertension: num  0 1 0 1 0 1 1 0 0 1 ...
##  $ SBP         : num  130 149 127 138 123 132 133 130 129 134 ...
##  $ LDL         : num  82 129 101 93 97 108 89 96 120 135 ...
##  $ time        : num  76 82 168 105 193 143 63 78 61 88 ...
##  $ log_antibody: num  10.65 9.89 10.9 9.91 9.56 ...
```

Univariate analysis(continous & categorical)

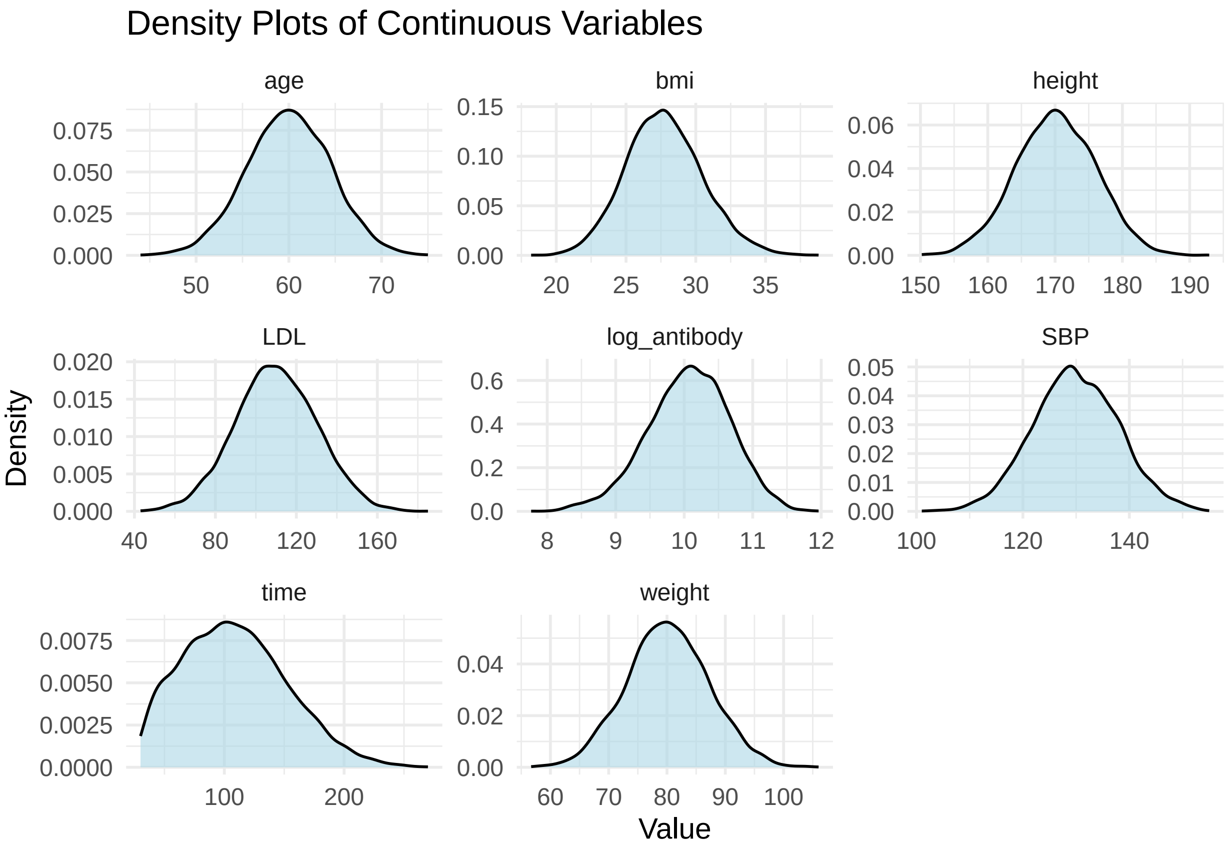
```r
dat1 <- dat1 %>%
  select(-id)

dat2 <- dat2 %>%
  select(-id)
continuous_var <- dat1 %>%
  select(age, height, weight, bmi, SBP, LDL, time, log_antibody)

categorical_var <- dat1 %>%
  select(gender, race, smoking, diabetes, hypertension) %>%
  mutate(
    # Convert binary variables to factors with labels
    gender = factor(gender, levels = c(0, 1), labels = c("Female", "Male")),
    diabetes = factor(diabetes, levels = c(0, 1), labels = c("No", "Yes")),
    hypertension = factor(hypertension, levels = c(0, 1), labels = c("No", "Yes"))
  )

# Continuous:
summary(continuous_var)
```

```
##       age            height          weight            bmi
##  Min.   :44.00   Min.   :150.2   Min.   : 56.70   Min.   :18.20
##  1st Qu.:57.00   1st Qu.:166.1   1st Qu.: 75.40   1st Qu.:25.80
##  Median :60.00   Median :170.1   Median : 80.10   Median :27.60
##  Mean   :59.97   Mean   :170.1   Mean   : 80.11   Mean   :27.74
##  3rd Qu.:63.00   3rd Qu.:174.2   3rd Qu.: 84.90   3rd Qu.:29.50
##  Max.   :75.00   Max.   :192.9   Max.   :106.00   Max.   :38.80
##       SBP             LDL             time        log_antibody
##  Min.   :101.0   Min.   : 43.0   Min.   : 30.0   Min.   : 7.765
##  1st Qu.:124.0   1st Qu.: 96.0   1st Qu.: 76.0   1st Qu.: 9.682
##  Median :130.0   Median :110.0   Median :106.0   Median :10.089
##  Mean   :129.9   Mean   :109.9   Mean   :108.9   Mean   :10.064
##  3rd Qu.:135.0   3rd Qu.:124.0   3rd Qu.:138.0   3rd Qu.:10.478
##  Max.   :155.0   Max.   :185.0   Max.   :270.0   Max.   :11.961
```

```
# Boxplots
continuous_var_long <- continuous_var %>%
  tidyr::pivot_longer(cols = everything(), names_to = "variable", values_to = "value")

ggplot(continuous_var_long, aes(x = variable, y = value)) +
  geom_boxplot(fill = "lightblue") +
  facet_wrap(~variable, scales = "free", ncol = 3) +
  theme_minimal() +
  labs(title = "Boxplots of Continuous Variables")
```



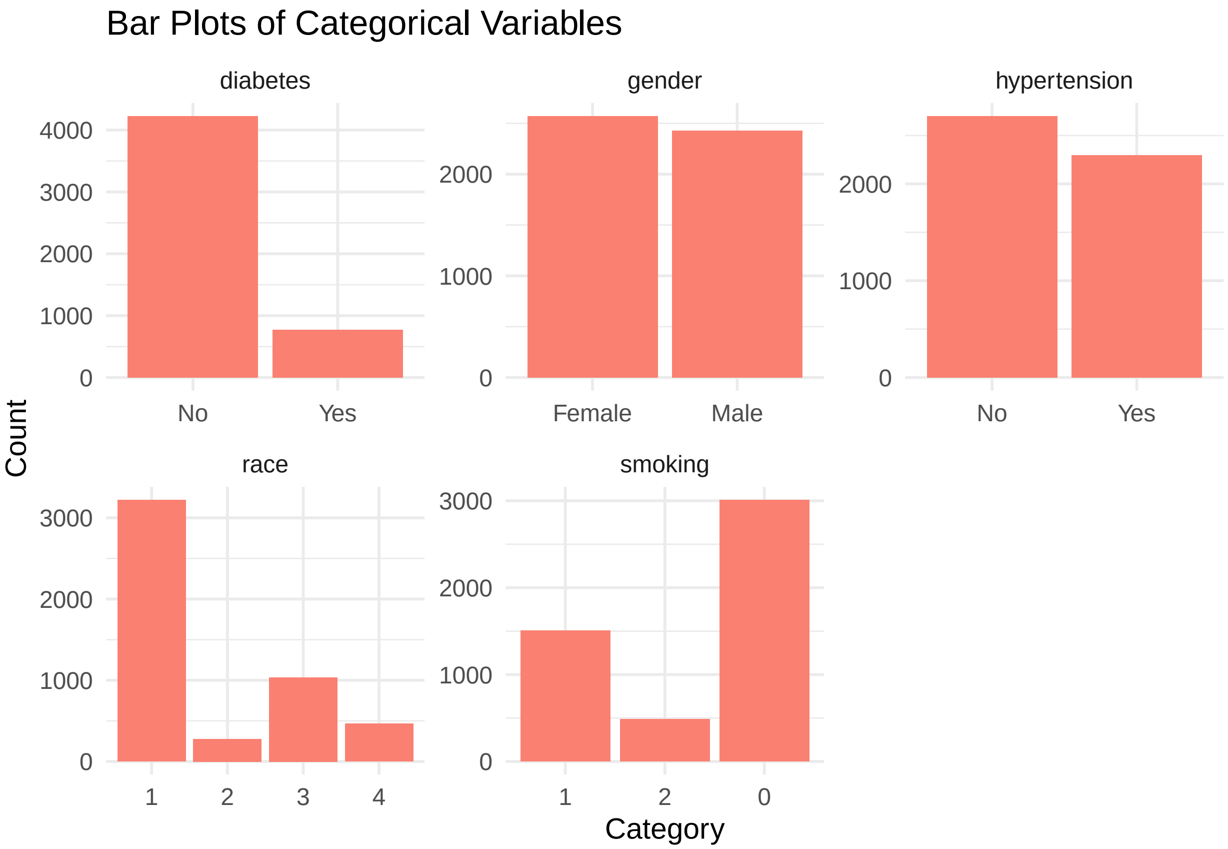
```
ggplot(continuous_var_long, aes(x = value)) +
  geom_density(fill = "lightblue", alpha = 0.6) +
  facet_wrap(~variable, scales = "free", ncol = 3) +
  theme_minimal() +
  labs(title = "Density Plots of Continuous Variables", x = "Value", y = "Density")
```



```r
# Categorical:
summary(continuous_var)
```

```
##       age            height          weight             bmi
##  Min.   :44.00   Min.   :150.2   Min.   : 56.70   Min.   :18.20
##  1st Qu.:57.00   1st Qu.:166.1   1st Qu.: 75.40   1st Qu.:25.80
##  Median :60.00   Median :170.1   Median : 80.10   Median :27.60
##  Mean   :59.97   Mean   :170.1   Mean   : 80.11   Mean   :27.74
##  3rd Qu.:63.00   3rd Qu.:174.2   3rd Qu.: 84.90   3rd Qu.:29.50
##  Max.   :75.00   Max.   :192.9   Max.   :106.00   Max.   :38.80
##       SBP             LDL             time         log_antibody
##  Min.   :101.0   Min.   : 43.0   Min.   : 30.0   Min.   : 7.765
##  1st Qu.:124.0   1st Qu.: 96.0   1st Qu.: 76.0   1st Qu.: 9.682
##  Median :130.0   Median :110.0   Median :106.0   Median :10.089
##  Mean   :129.9   Mean   :109.9   Mean   :108.9   Mean   :10.064
##  3rd Qu.:135.0   3rd Qu.:124.0   3rd Qu.:138.0   3rd Qu.:10.478
##  Max.   :155.0   Max.   :185.0   Max.   :270.0   Max.   :11.961
```

```r
# bar plots
categorical_var_long <- categorical_var %>%
  tidyr::pivot_longer(cols = everything(), names_to = "variable", values_to = "value")

ggplot(categorical_var_long, aes(x = value)) +
  geom_bar(fill = "salmon") +
  facet_wrap(~variable, scales = "free", ncol = 3) +
  theme_minimal() +
  labs(title = "Bar Plots of Categorical Variables", x = "Category", y = "Count")
```



According to the box plot for continuous variables:

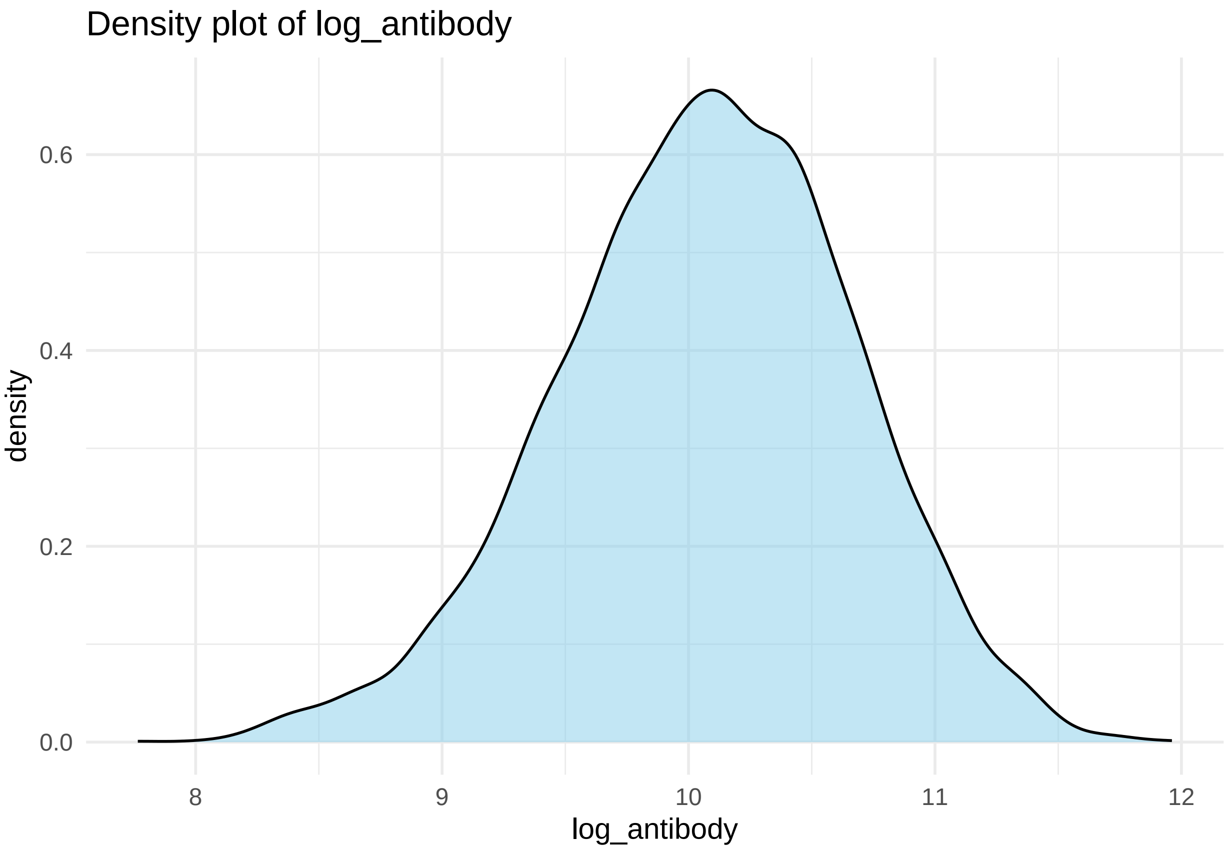
- `Age`, `BMI`, and `SBP` appear reasonably normally distributed, with expected ranges for an adult population; `LDL` cholesterol and `time` since vaccination show a wider range, right-skewness and some outliers, which may impact linear models.
- `log_antibody` (response) appears fairly symmetrical, which supports its use as a continuous response in linear or GAM models.
- Correlations and non-linear trends should be assessed in the next step to guide model form.

According to the bar plot for categorical variables:

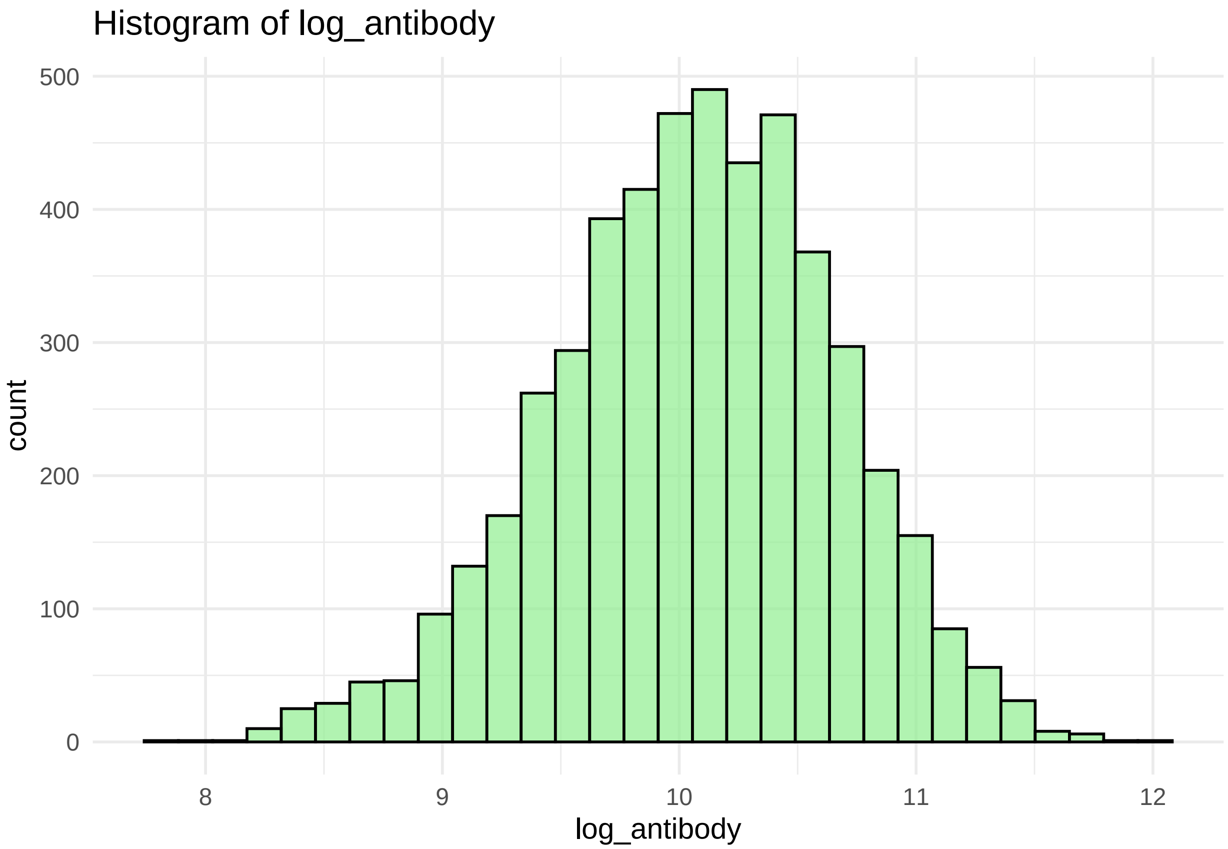
- `Gender` is fairly balanced between Female and Male;
- `Race` is skewed, with a majority of participants identifying as White (Category 1). Other racial/ethnic groups are underrepresented;
- `Smoking` status shows that the majority are never smokers (Category 0), with fewer current and former smokers;
- A large proportion of participants do not have `diabetes`;
- A moderate split exists for `hypertension`, which may contribute meaningfully to clinical outcome variation
- Demographically, the population is balanced by gender but skewed by race and smoking status.

Overall, we believe the response variable `log_antibody` is well-behaved, and further correlation analysis(eg. bivariate) is needed.

```r
ggplot(dat1, aes(x = log_antibody)) +
  geom_density(fill = "skyblue", alpha = 0.5) +
  ggtitle("Density plot of log_antibody") +
  xlab("log_antibody") +
  theme_minimal()
```

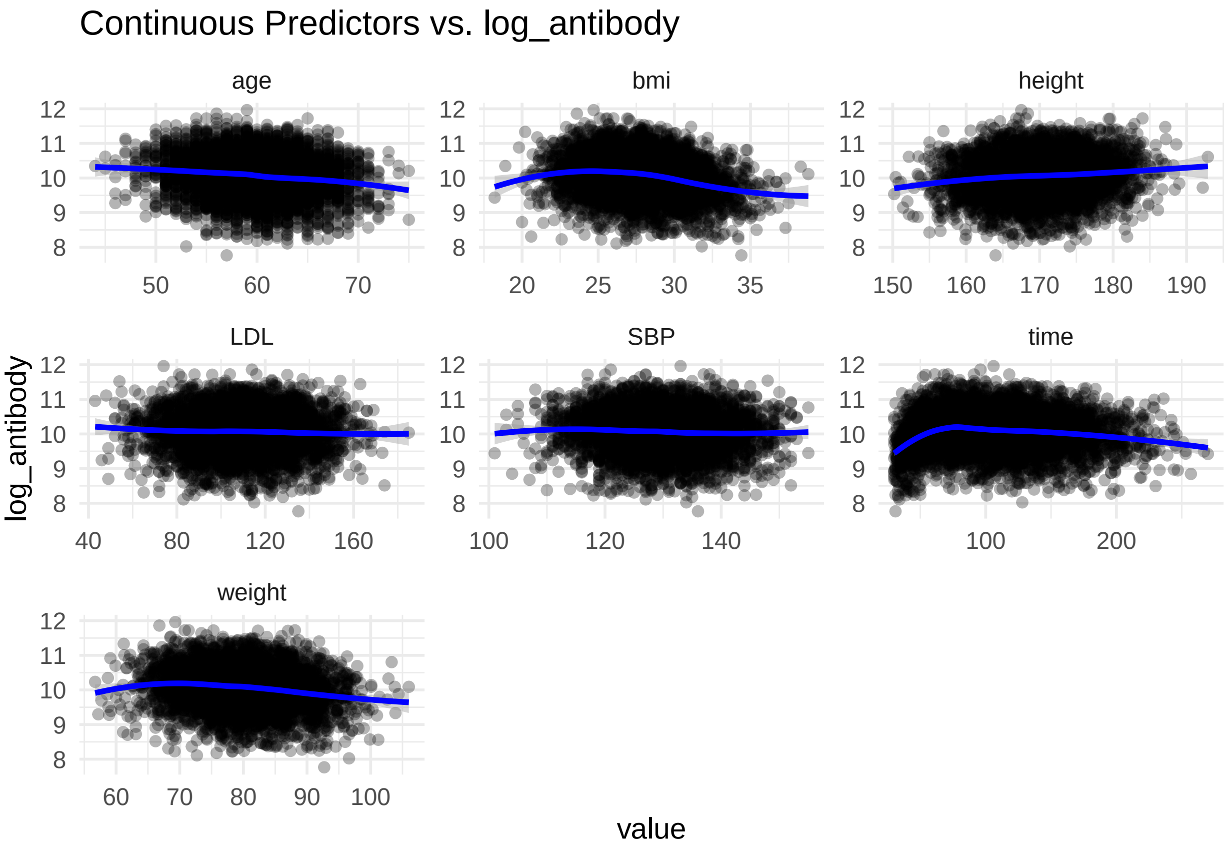


```
ggplot(dat1, aes(x = log_antibody)) +
  geom_histogram(bins = 30, fill = "lightgreen", color = "black", alpha = 0.7) +
  ggtitle("Histogram of log_antibody") +
  xlab("log_antibody") +
  theme_minimal()
```



```
# continous variable
continuous_var_long <- dat1 %>%
  select(age, height, weight, bmi, SBP, LDL, time, log_antibody) %>%
  tidyr::pivot_longer(cols = -log_antibody, names_to = "variable", values_to = "value")

# Scatterplots with smoothing lines
ggplot(continuous_var_long, aes(x = value, y = log_antibody)) +
  geom_point(alpha = 0.3) +
  geom_smooth(method = "loess", color = "blue") +
  facet_wrap(~variable, scales = "free", ncol = 3) +
  theme_minimal() +
  labs(title = "Continuous Predictors vs. log_antibody")
```

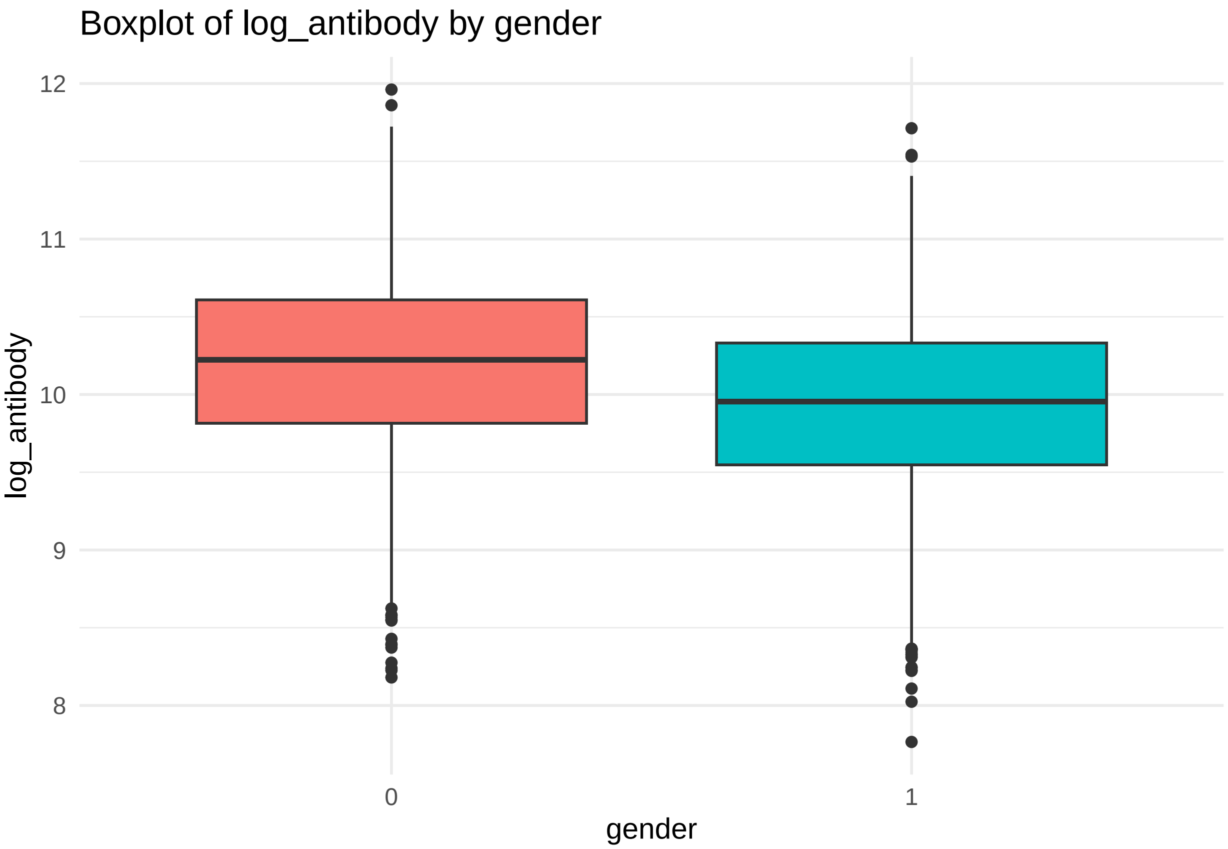
```
## `geom_smooth()` using formula = 'y ~ x'
```

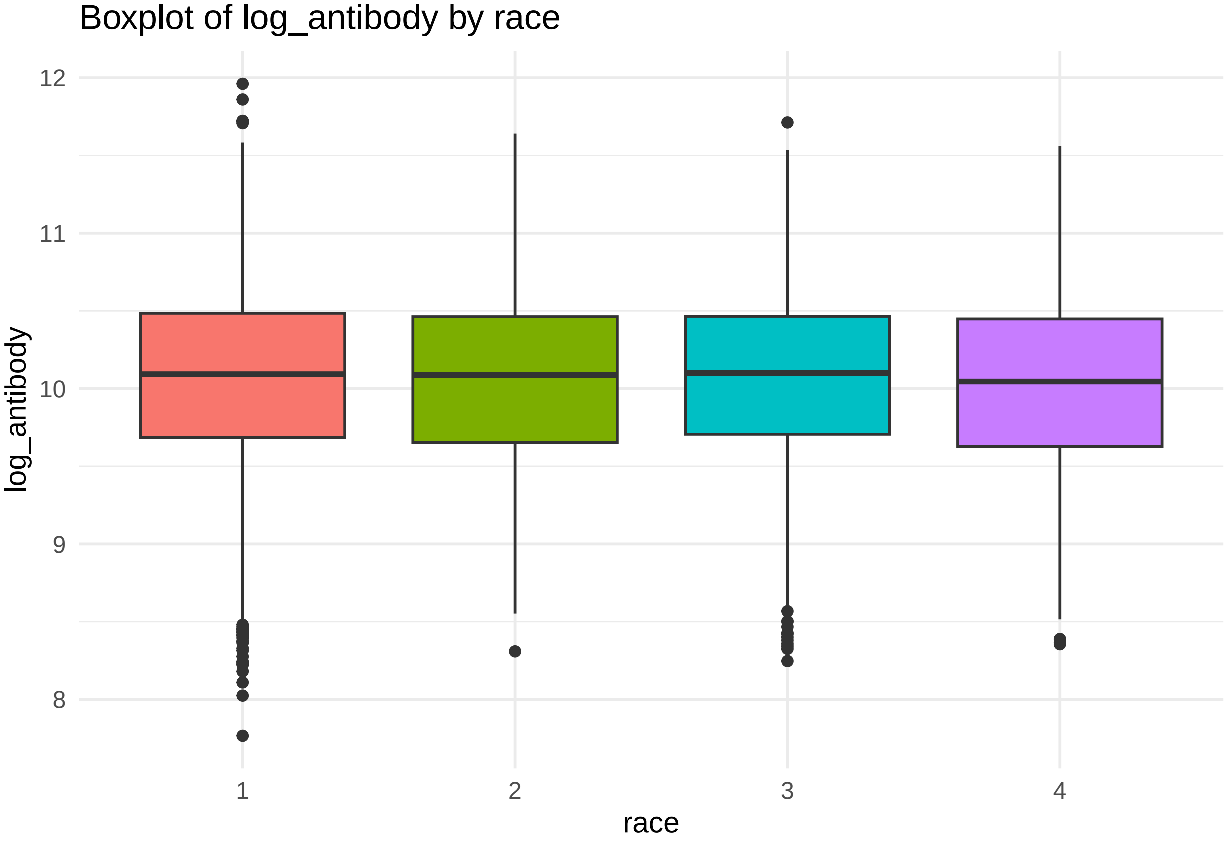


Using LOESS method, we observe linearity between predictors and the response. The plot shows that `bmi`, `time`, and `weight` has clear non linear trend against resopnse `log_antibody`, indicating potential need to use GAM or non linear model.

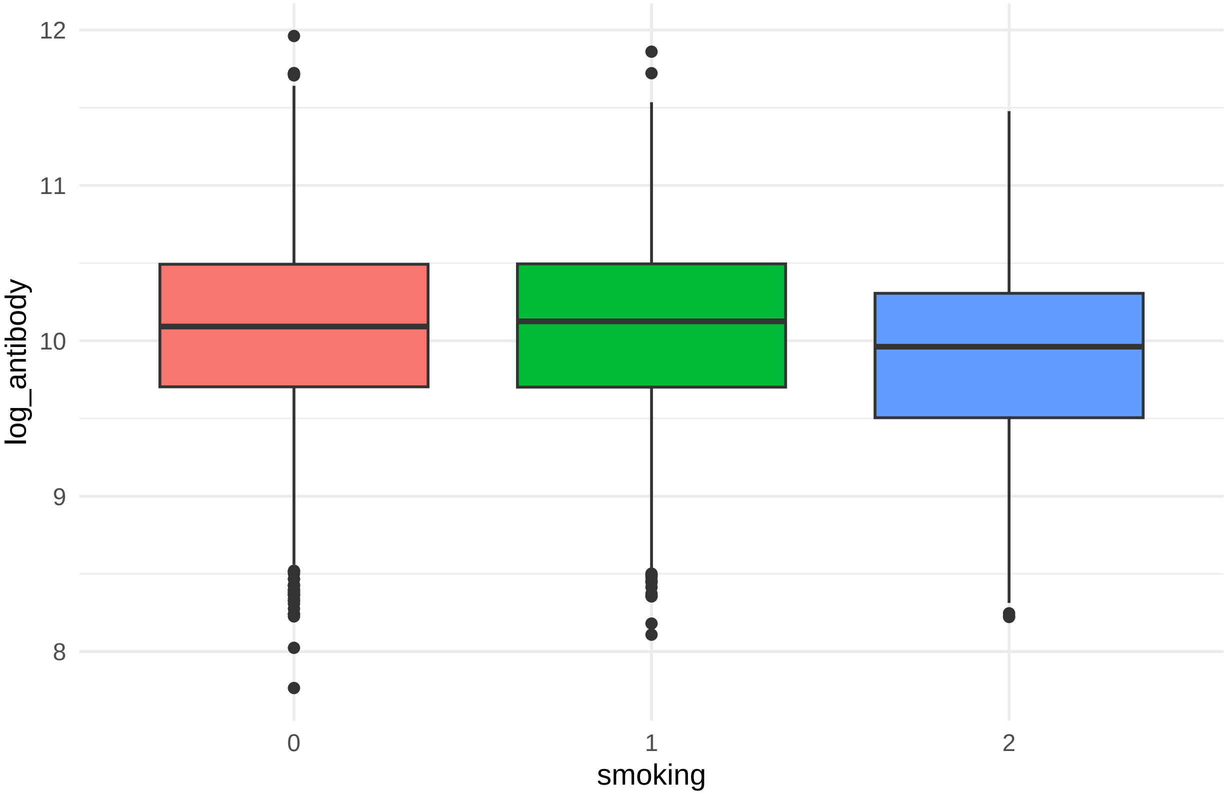
```
# categorical data
categorical_name <- c("gender", "race", "smoking", "diabetes", "hypertension")
dat1[categorical_name] <- lapply(dat1[categorical_name], factor)

for (name in categorical_name) {
  p <- ggplot(dat1, aes_string(x = name, y = "log_antibody", fill = name)) +
    geom_boxplot() +
    ggtitle(paste("Boxplot of log_antibody by", name)) +
    theme_minimal() +
    theme(legend.position = "none")
  print(p)
}
```

```
## Warning: `aes_string()` was deprecated in ggplot2 3.0.0.
## i Please use tidy evaluation idioms with `aes()`.
## i See also `vignette("ggplot2-in-packages")` for more information.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

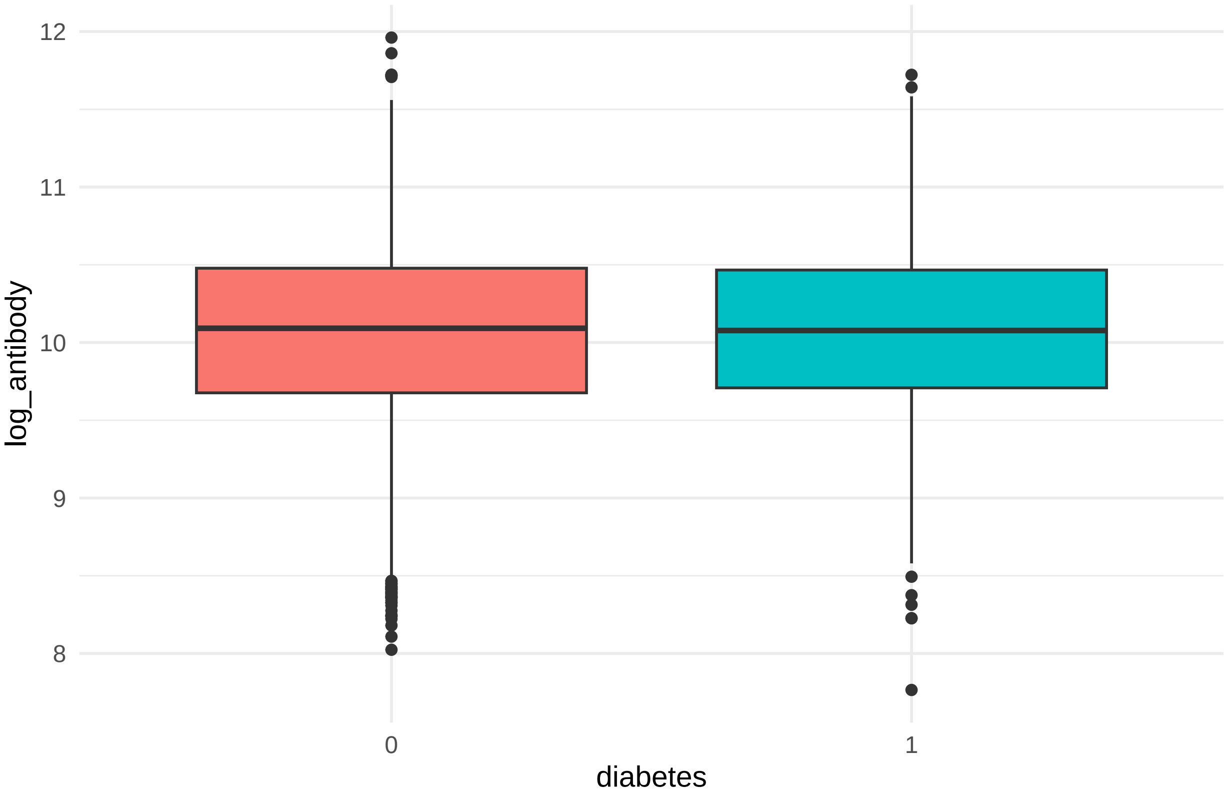

Boxplot of log_antibody by gender

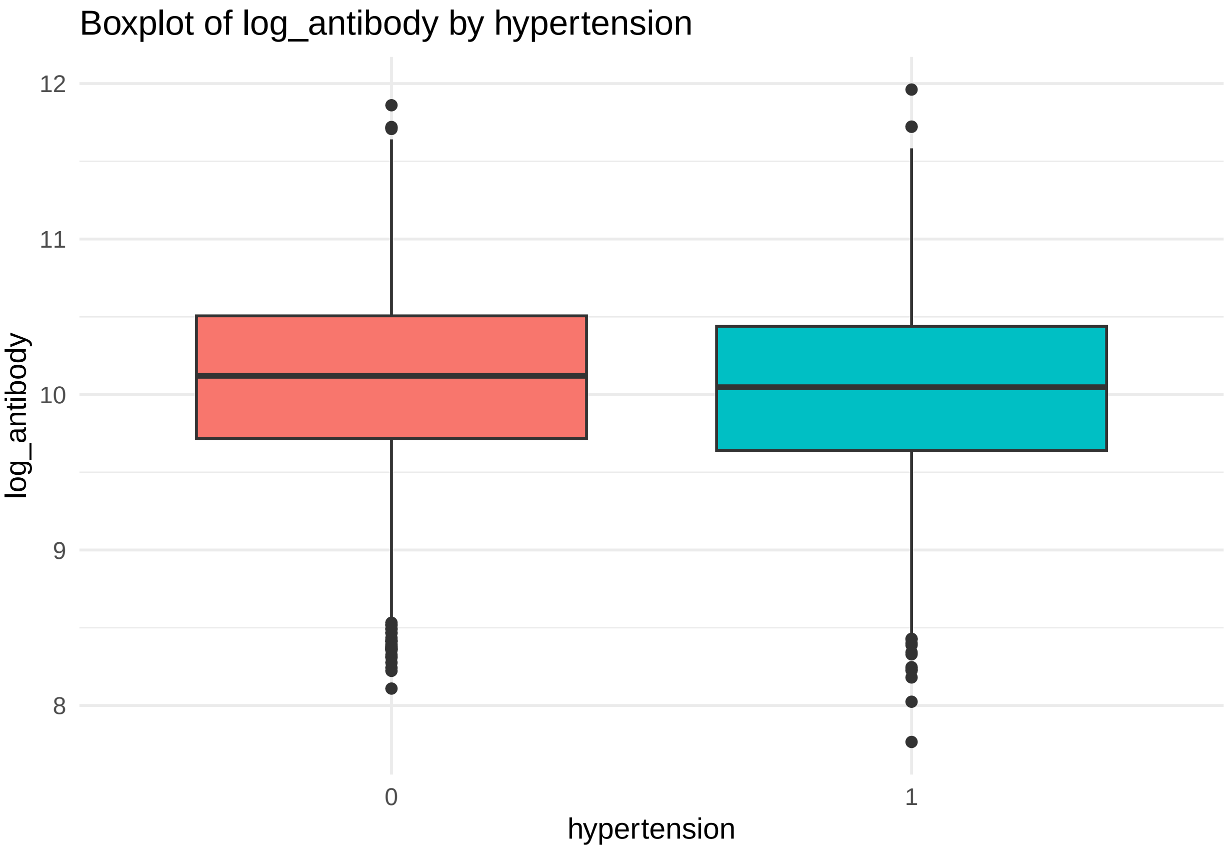

Boxplot of log_antibody by race

Boxplot of log_antibody by smoking



Boxplot of log_antibody by diabetes



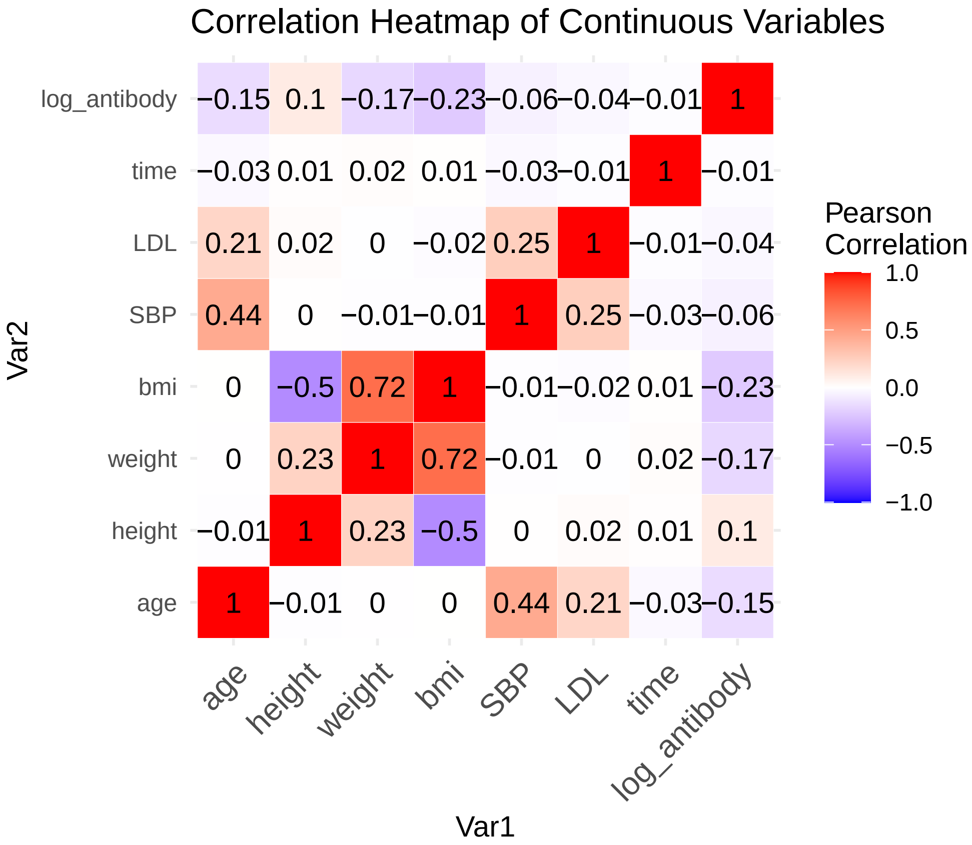


Correlation Analysis

```r
continous_name <- c("age", "height", "weight", "bmi", "SBP", "LDL", "time", "log_antibody")
dat_cont <- dat1[ , continous_name]

# coefficient matrix
cor_matrix <- cor(dat_cont, use = "complete.obs", method = "pearson")

print(round(cor_matrix, 2))
```

```
##                age height weight   bmi   SBP   LDL  time log_antibody
## age           1.00  -0.01   0.00  0.00  0.44  0.21 -0.03        -0.15
## height       -0.01   1.00   0.23 -0.50  0.00  0.02  0.01         0.10
## weight        0.00   0.23   1.00  0.72 -0.01  0.00  0.02        -0.17
## bmi           0.00  -0.50   0.72  1.00 -0.01 -0.02  0.01        -0.23
## SBP           0.44   0.00  -0.01 -0.01  1.00  0.25 -0.03        -0.06
## LDL           0.21   0.02   0.00 -0.02  0.25  1.00 -0.01        -0.04
## time         -0.03   0.01   0.02  0.01 -0.03 -0.01  1.00        -0.01
## log_antibody -0.15   0.10  -0.17 -0.23 -0.06 -0.04 -0.01         1.00
```

```r
cor_melt <- melt(cor_matrix)

ggplot(cor_melt, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1, 1), space = "Lab",
                       name = "Pearson\nCorrelation") +
  geom_text(aes(label = round(value, 2)), color = "black", size = 4) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1)) +
  coord_fixed() +
  ggtitle("Correlation Heatmap of Continuous Variables")
```



```r
for (i in 1:(length(categorical_name)-1)) {
  for (j in (i+1):length(categorical_name)) {
    var1 <- categorical_name[i]
    var2 <- categorical_name[j]
    cat("\nContingency Table:", var1, "vs", var2, "\n")
    tab <- table(dat1[[var1]], dat1[[var2]])
    print(tab)
    cat("Chi-squared Test:\n")
    print(chisq.test(tab))
  }
}
```

```
##
## Contingency Table: gender vs race
##
##          1    2    3    4
##   0 1642  151  542  238
##   1 1579  127  494  227
## Chi-squared Test:
##
```

```
##  Pearson's Chi-squared test
##
## data:  tab
## X-squared = 1.5264, df = 3, p-value = 0.6762
##
##
## Contingency Table: gender vs smoking
##
##        0    1    2
##   0 1554  759  260
##   1 1456  745  226
## Chi-squared Test:
##
##  Pearson's Chi-squared test
##
## data:  tab
## X-squared = 1.4376, df = 2, p-value = 0.4873
##
##
## Contingency Table: gender vs diabetes
##
##        0    1
##   0 2167  406
##   1 2061  366
## Chi-squared Test:
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab
## X-squared = 0.41526, df = 1, p-value = 0.5193
##
##
## Contingency Table: gender vs hypertension
##
##        0    1
##   0 1419 1154
##   1 1283 1144
## Chi-squared Test:
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab
## X-squared = 2.5366, df = 1, p-value = 0.1112
##
##
## Contingency Table: race vs smoking
##
##        0    1    2
##   1 1981  956  284
##   2  149   89   40
##   3  605  325  106
##   4  275  134   56
## Chi-squared Test:
##
```

```
##  Pearson's Chi-squared test
##
## data:  tab
## X-squared = 16.7, df = 6, p-value = 0.01045
##
##
## Contingency Table: race vs diabetes
##
##        0    1
##   1 2725  496
##   2  240   38
##   3  879  157
##   4  384   81
## Chi-squared Test:
##
##  Pearson's Chi-squared test
##
## data:  tab
## X-squared = 2.132, df = 3, p-value = 0.5455
##
##
## Contingency Table: race vs hypertension
##
##        0    1
##   1 1741 1480
##   2  152  126
##   3  566  470
##   4  243  222
## Chi-squared Test:
##
##  Pearson's Chi-squared test
##
## data:  tab
## X-squared = 0.78675, df = 3, p-value = 0.8526
##
##
## Contingency Table: smoking vs diabetes
##
##        0    1
##   0 2556  454
##   1 1259  245
##   2  413   73
## Chi-squared Test:
##
##  Pearson's Chi-squared test
##
## data:  tab
## X-squared = 1.1913, df = 2, p-value = 0.5512
##
##
## Contingency Table: smoking vs hypertension
##
##        0    1
##   0 1621 1389
##   1  814  690
##   2  267  219
## Chi-squared Test:
##
##  Pearson's Chi-squared test
##
## data:  tab
## X-squared = 0.204, df = 2, p-value = 0.903
##
##
## Contingency Table: diabetes vs hypertension
##
##         0    1
##   0 2284 1944
##   1  418  354
## Chi-squared Test:
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  tab
## X-squared = 0.00059731, df = 1, p-value = 0.9805
```

Model Selection

```
# VIF
lm_full <- lm(log_antibody ~ age + gender + smoking + height + weight + bmi +
                diabetes + hypertension + SBP + LDL + time, data = dat1)

vif(lm_full)
```

```
##                    GVIF Df GVIF^(1/(2*Df))
## age            1.258104  1        1.121652
## gender         1.002988  1        1.001493
## smoking        1.002682  2        1.000670
## height       107.111548  1       10.349471
## weight       169.112707  1       13.004334
## bmi          213.764468  1       14.620686
## diabetes       1.001898  1        1.000949
## hypertension   2.791341  1        1.670731
## SBP            3.070211  1        1.752202
## LDL            1.085268  1        1.041762
## time           1.002242  1        1.001120
```

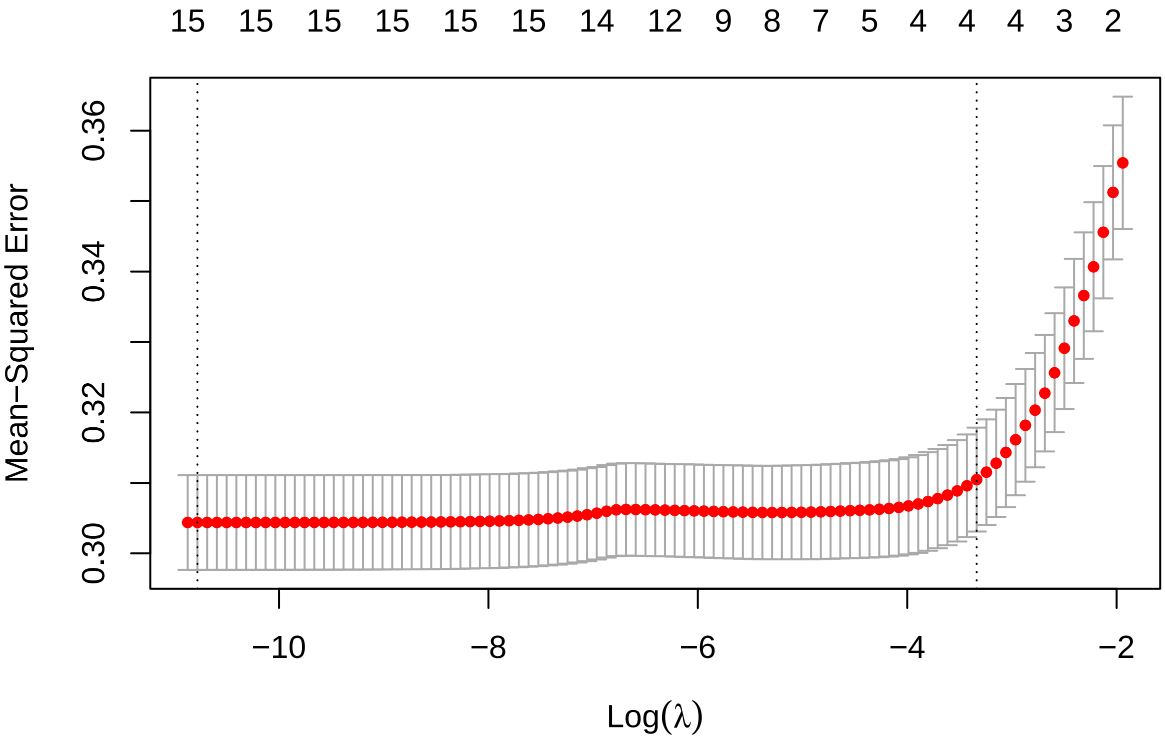
```
x <- model.matrix(log_antibody ~ ., data = dat1)[, -1]
y <- dat1$log_antibody

lasso_cv <- cv.glmnet(x, y, alpha = 1, standardize = TRUE)


lasso_cv$lambda.min
```

```
## [1] 2.082471e-05
```

```
lasso_coef <- coef(lasso_cv, s = "lambda.1se")
print(lasso_coef)
```

```
## 16 x 1 sparse Matrix of class "dgCMatrix"
##                          s1
## (Intercept)   11.92665650
## age           -0.01217839
## gender1       -0.22108381
## race2          .
## race3          .
## race4          .
## smoking1       .
## smoking2      -0.07608186
## height         .
## weight         .
## bmi           -0.03667170
## diabetes1      .
## hypertension1  .
## SBP            .
## LDL            .
## time           .
```

```
plot(lasso_cv)
```



```
ridge_cv <- cv.glmnet(x, y, alpha = 0, standardize = TRUE)

ridge_cv$lambda.min
```

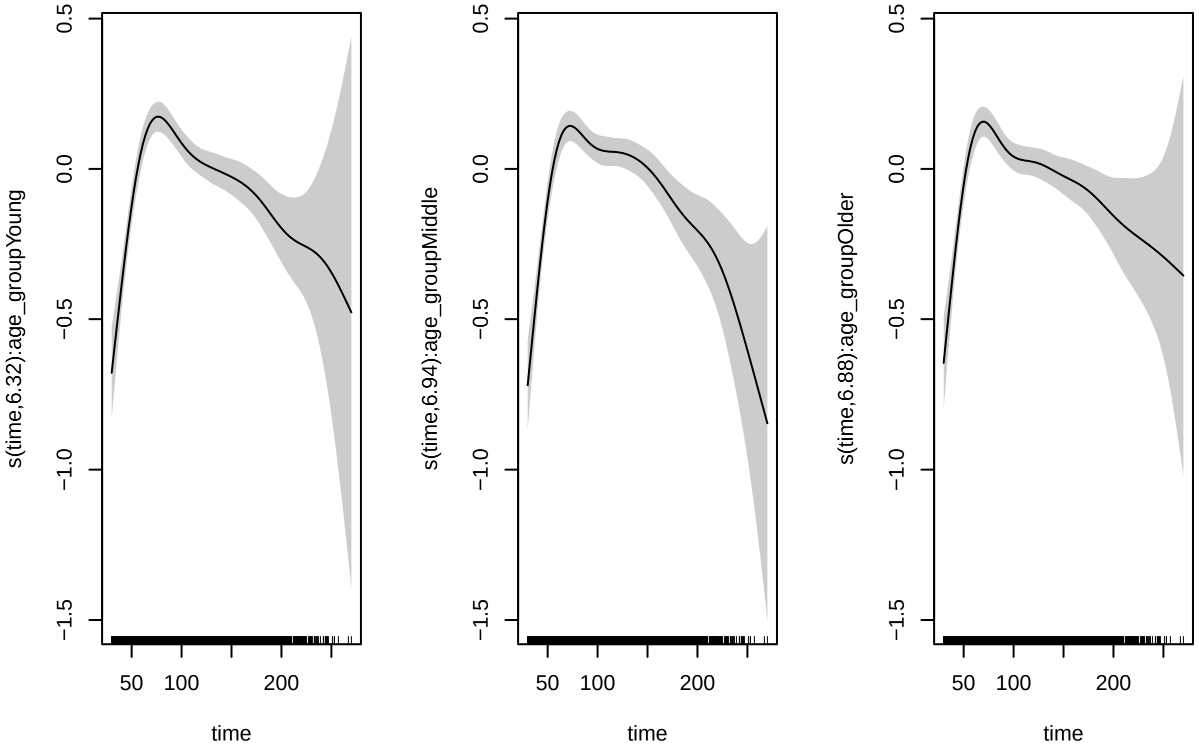
```
## [1] 0.01435366
```

```
ridge_coef <- coef(ridge_cv, s = "lambda.min")
print(ridge_coef)
```

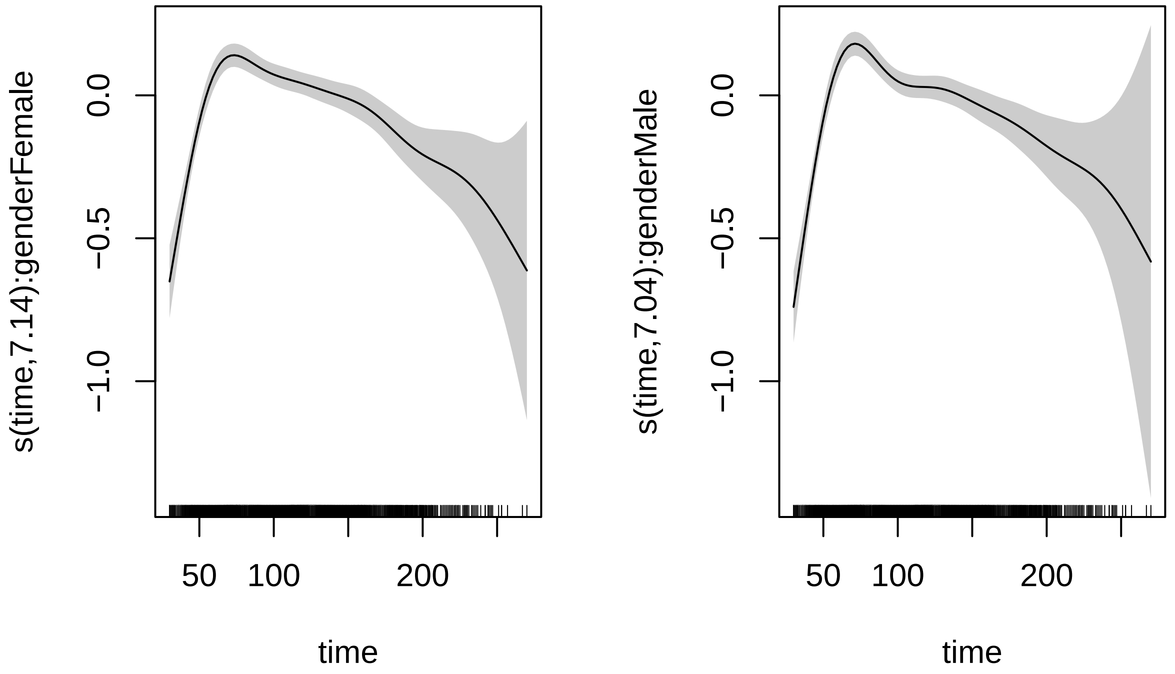
```
## 16 x 1 sparse Matrix of class "dgCMatrix"
##                            s1
## (Intercept)   12.7384560516
## age           -0.0197679168
## gender1       -0.2880871615
## race2         -0.0038318346
## race3         -0.0066654722
## race4         -0.0417690245
## smoking1       0.0242172365
## smoking2      -0.1847225211
## height        -0.0001949552
## weight        -0.0009185524
## bmi           -0.0473178893
## diabetes1      0.0113069924
## hypertension1 -0.0166641972
## SBP            0.0010847735
## LDL           -0.0001614795
## time          -0.0002807285
```

Interaction Analysis

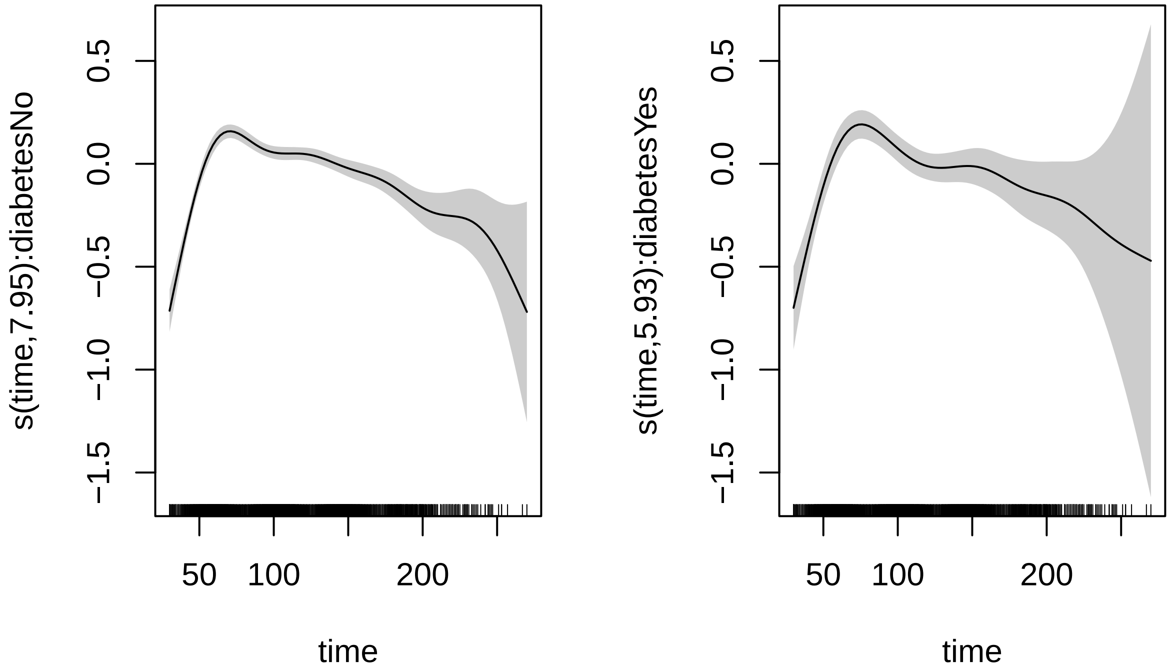
```
dat1_ageGroup <- dat1 %>%
  mutate(age_group = ntile(age, 3)) %>%
  mutate(age_group = factor(age_group, labels = c("Young", "Middle", "Older"))) %>%
  mutate(
    gender = factor(gender, labels = c("Female", "Male")),
    diabetes = factor(diabetes, labels = c("No", "Yes")),
    hypertension = factor(hypertension, labels = c("No", "Yes"))
  )

library(mgcv)

gam_age_interact <- gam(
  log_antibody ~ s(time, by = age_group) + age_group + gender + bmi + SBP + LDL +
    race + smoking + diabetes + hypertension,
  data = dat1_ageGroup,
  method = "GCV.Cp"
)

gam_gender_interact <- gam(
  log_antibody ~ s(time, by = gender) + gender + age + bmi + SBP + LDL +
    race + smoking + diabetes + hypertension,
  data = dat1_ageGroup,
  method = "GCV.Cp"
)

gam_diabetes_interact <- gam(
  log_antibody ~ s(time, by = diabetes) + diabetes + age + bmi + SBP + LDL +
    race + gender + smoking + hypertension,
  data = dat1_ageGroup,
  method = "GCV.Cp"
)
gam_hypertension_interact <- gam(
  log_antibody ~ s(time, by = hypertension) + diabetes + age + bmi + SBP + LDL +
    race + gender + smoking + hypertension,
  data = dat1_ageGroup,
  method = "GCV.Cp"
)
# Plotting smooth terms
par(mfrow = c(1, 3))
plot(gam_age_interact, shade = TRUE)
```



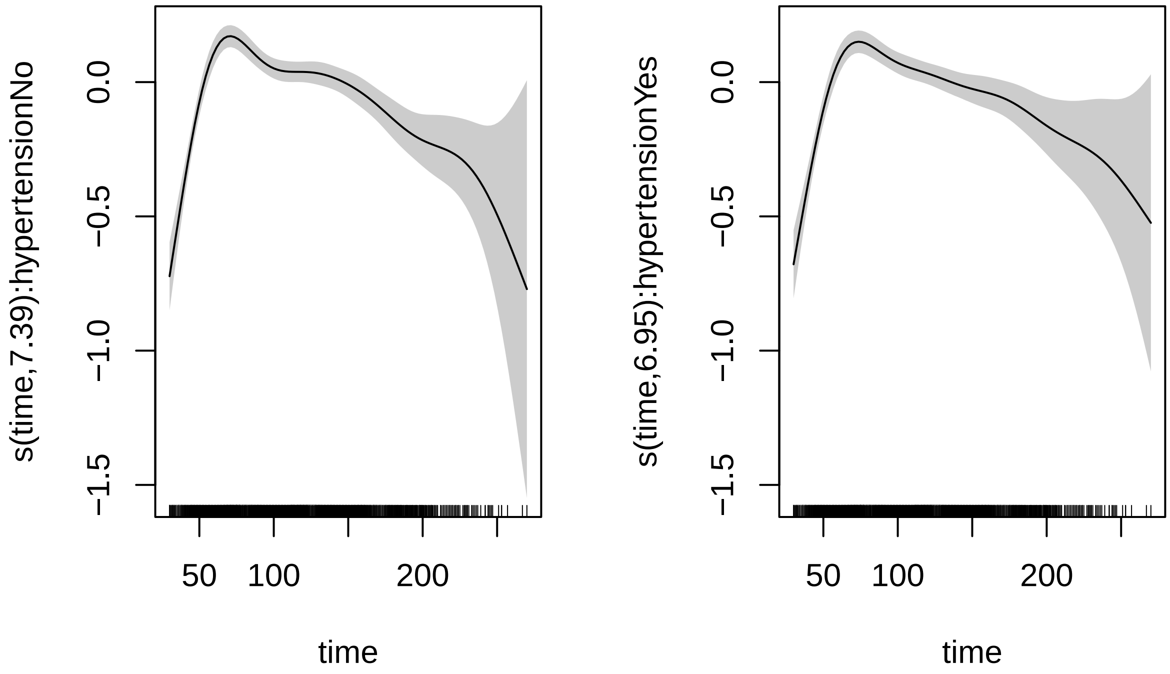
```
par(mfrow = c(1, 2))
plot(gam_gender_interact, shade = TRUE)
```



```r
plot(gam_diabetes_interact, shade = TRUE)
```



```r
plot(gam_hypertension_interact, shade = TRUE)
```



Model Training

```
ctrl1 <- trainControl(method = "cv", number = 10)

train_y <- dat1$log_antibody
train_x <- dat1[, -which(names(dat1) == "log_antibody")]
```

```
set.seed(2)
gam.fit <- train(train_x, train_y,
                 method = "gam",
                 # tuneGrid = data.frame(method = "GCV.Cp", select = c(TRUE,FALSE)),
                 trControl = ctrl1)

gam.fit$bestTune
```

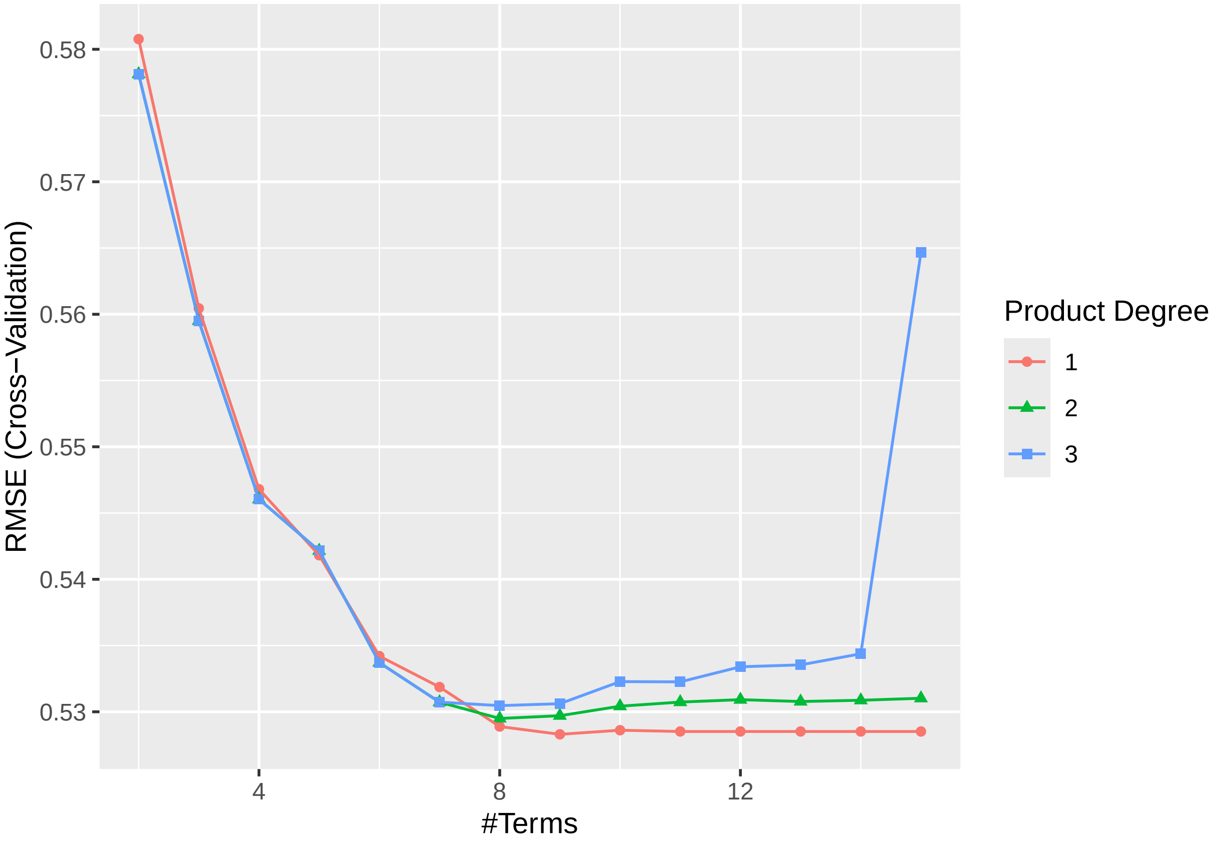
```
##   select method
## 2   TRUE GCV.Cp
```

```
gam.fit$finalModel
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## .outcome ~ gender + diabetes + hypertension + smoking + race +
##     s(age) + s(SBP) + s(LDL) + s(bmi) + s(time) + s(height) +
##     s(weight)
##
## Estimated degrees of freedom:
## 0.991 0.000 0.000 4.179 7.892 1.234 0.000
##  total = 23.3
```

```
##
## GCV score: 0.2786734
```

```
mars_grid <- expand.grid(degree = 1:3,
                         nprune = 2:15)

set.seed(2)
mars.fit <- train(train_x, train_y,
                  method = "earth",
                  tuneGrid = mars_grid,
                  trControl = ctrl1)

ggplot(mars.fit)
```



```
mars.fit$bestTune
```

```
##   nprune degree
## 8      9      1
```

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

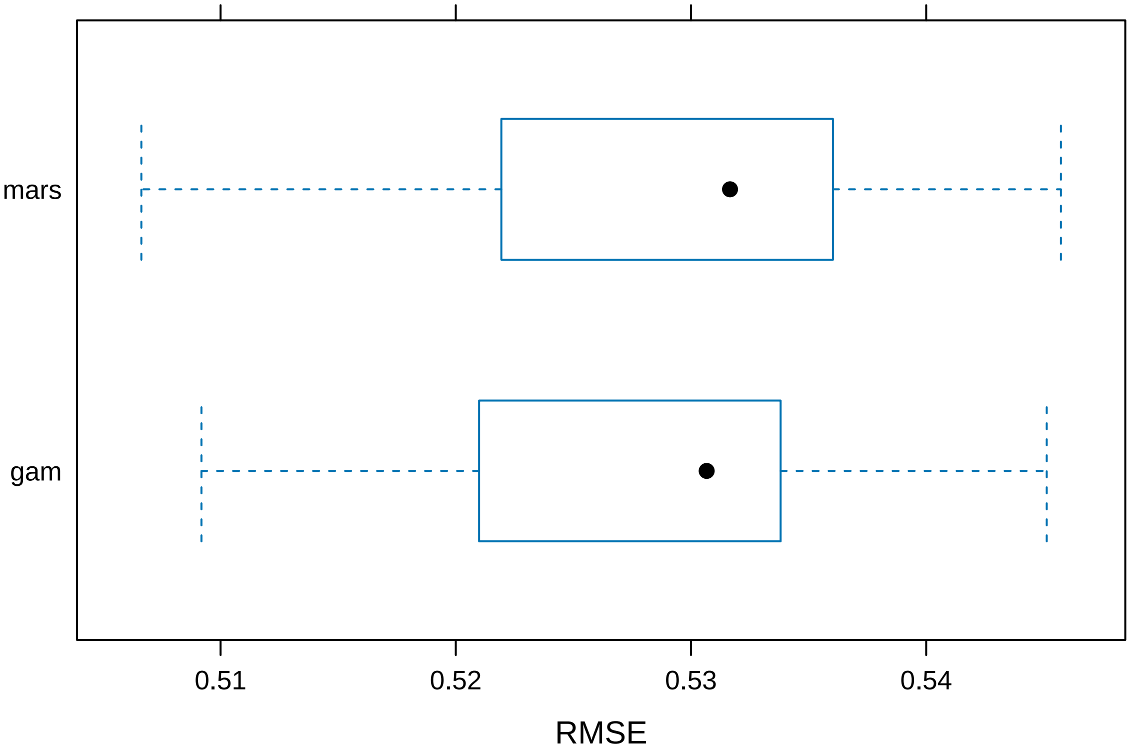
```
##  (Intercept)  h(27.8-bmi)   h(time-57)   h(57-time)      gender1    h(age-59)
## 10.847446930 -0.061997354 -0.002254182 -0.033529326 -0.296290451 -0.022957648
##    h(59-age)     smoking2  h(bmi-23.7)
##  0.016138468 -0.205126851 -0.084380175
```

```
resamp <- resamples(list(mars = mars.fit, gam = gam.fit))
summary(resamp)
```

```
##
```

```
## Call:
## summary.resamples(object = resamp)
##
## Models: mars, gam
## Number of resamples: 10
##
## MAE
##           Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## mars 0.4120189 0.4180233 0.4203065 0.4224208 0.4285348 0.4360995    0
## gam  0.4127242 0.4190074 0.4202804 0.4224455 0.4273258 0.4352565    0
##
## RMSE
##           Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## mars 0.5066327 0.5230870 0.5316602 0.5282995 0.5354905 0.5457286    0
## gam  0.5091877 0.5223781 0.5306669 0.5279212 0.5336806 0.5451253    0
##
## Rsquared
##           Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## mars 0.1766328 0.1941155 0.2028183 0.2159220 0.2369173 0.2730827    0
## gam  0.1795042 0.1955026 0.2071224 0.2170568 0.2376473 0.2735385    0
```

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



```
mars.pred <- predict(mars.fit, newdata = dat2)
# test RMSE
mars_test_rmse = sqrt(mean((mars.pred - dat2[, "log_antibody"])^2))
mars_test_rmse
```

```
## [1] 0.5327718
```

```r
gam.pred <- predict(gam.fit, newdata = dat2)
# test RMSE
gam_test_rmse = sqrt(mean((gam.pred - dat2[, "log_antibody"])^2))
gam_test_rmse
```

```
## [1] 0.5700836
```