# Midterm

Sining Leng

2025-03-24

Load data

```
load("dat1.RData")
load("dat2.RData")

dat1 =
  dat1 |>
  select(-id)

dat2 =
  dat2 |>
  select(-id)
```

## Explorey Data Analysis and Visualization

### Summary statistics

```
summarize_cat = function(data, var_name) {

  var_label = deparse(substitute(var_name))

  out_df =
    data |>
    count({{ var_name }}) |>
    mutate(
      Percent = round(100 * n / sum(n), 1),
      Variable = var_label,
      Level = as.character({{ var_name }})
    ) |>
    rename(N = n) |>
    select(Variable, Level, N, Percent)

  return(out_df)
}

gender <- summarize_cat(dat1, gender)
race <- summarize_cat(dat1, race)
smoking <- summarize_cat(dat1, smoking)
diabetes <- summarize_cat(dat1, diabetes)
```

```r
hypertension <- summarize_cat(dat1, hypertension)
bind_rows(gender, race, smoking, diabetes, hypertension) %>%
  knitr::kable()
```

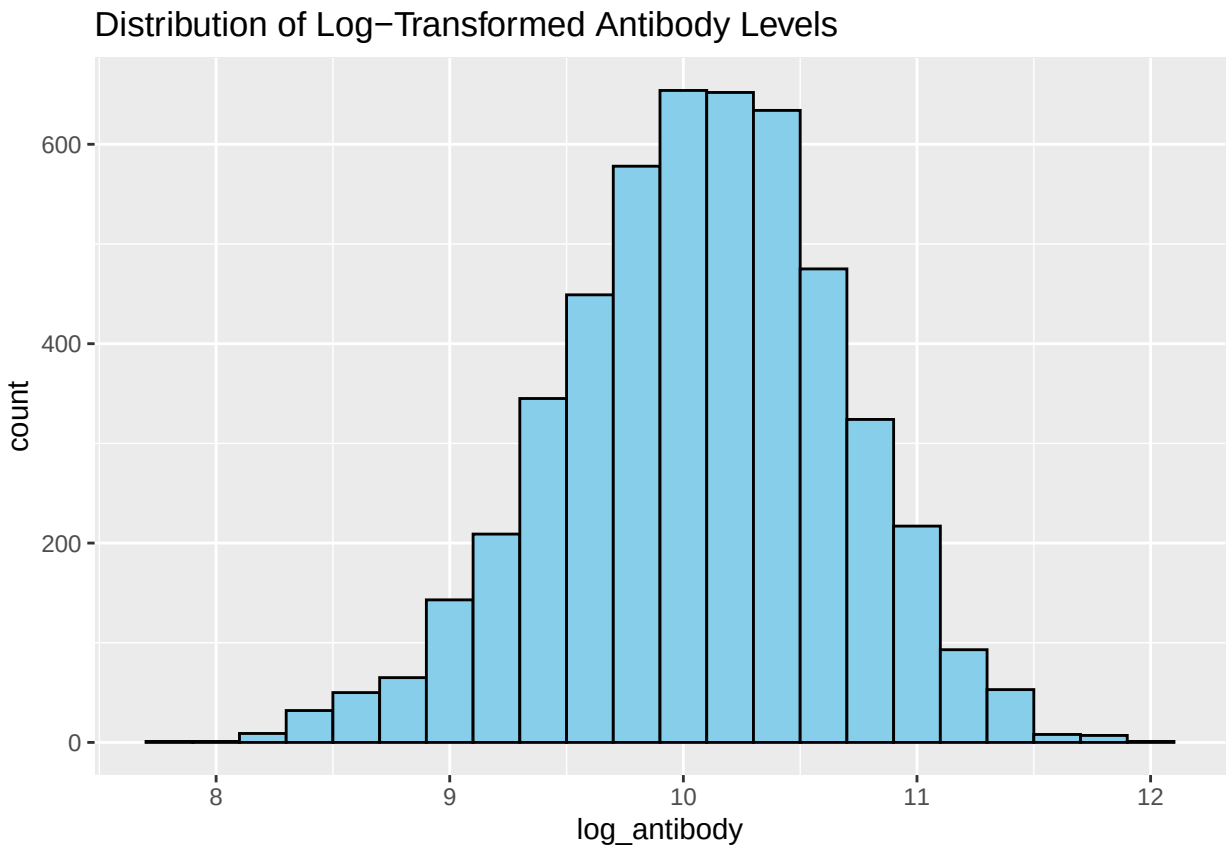
| Variable | Level | N | Percent |
|---|---|---|---|
| gender | 0 | 2573 | 51.5 |
| gender | 1 | 2427 | 48.5 |
| race | 1 | 3221 | 64.4 |
| race | 2 | 278 | 5.6 |
| race | 3 | 1036 | 20.7 |
| race | 4 | 465 | 9.3 |
| smoking | 0 | 3010 | 60.2 |
| smoking | 1 | 1504 | 30.1 |
| smoking | 2 | 486 | 9.7 |
| diabetes | 0 | 4228 | 84.6 |
| diabetes | 1 | 772 | 15.4 |
| hypertension | 0 | 2702 | 54.0 |
| hypertension | 1 | 2298 | 46.0 |

```r
summarize_cont = function(data, var_name) {

  var_label = deparse(substitute(var_name))

  out_df =
    data |>
    summarize(
      Variable = var_label,
      Median = round(median({{ var_name }}, na.rm = TRUE), 1),
      Q1 = round(quantile({{ var_name }}, 0.25, na.rm = TRUE), 1),
      Q3 = round(quantile({{ var_name }}, 0.75, na.rm = TRUE), 1)
    ) |>
    mutate(
      IQR = paste0("[", Q1, ", ", Q3, "]")
    ) |>
    select(Variable, Median, IQR)

  return(out_df)
}

age = summarize_cont(dat1, age)
bmi = summarize_cont(dat1, bmi)
height = summarize_cont(dat1, height)
weight = summarize_cont(dat1, weight)
SBP = summarize_cont(dat1, SBP)
LDL = summarize_cont(dat1, LDL)
log_anti = summarize_cont(dat1, log_antibody)
bind_rows(age, bmi, height, weight, SBP, LDL, log_anti) %>%
  knitr::kable()
```

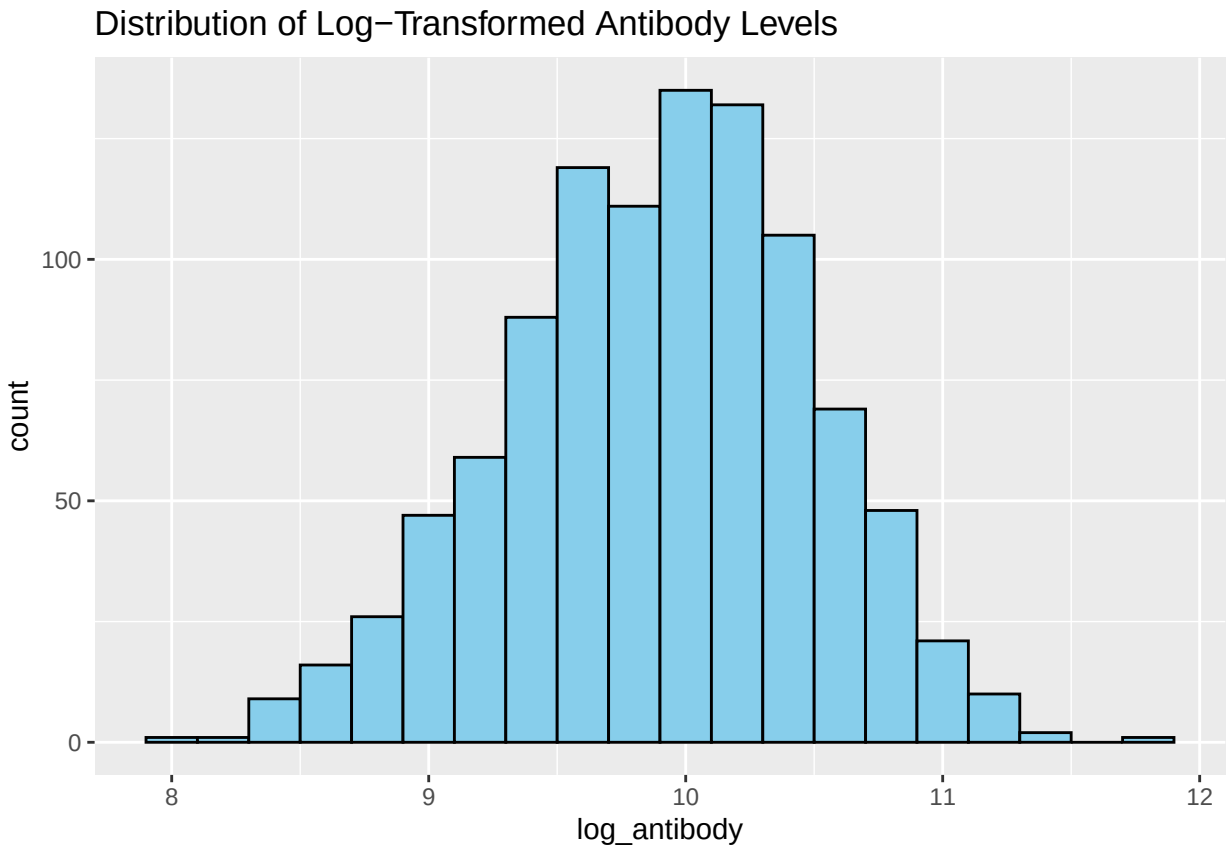
| Variable | Median | IQR |
|---|---|---|
| age | 60.0 | [57, 63] |
| bmi | 27.6 | [25.8, 29.5] |
| height | 170.1 | [166.1, 174.2] |
| weight | 80.1 | [75.4, 84.9] |
| SBP | 130.0 | [124, 135] |
| LDL | 110.0 | [96, 124] |
| log_antibody | 10.1 | [9.7, 10.5] |

### Distribution of antibody levels

```
ggplot(dat1, aes(x = log_antibody)) +
  geom_histogram(binwidth = 0.2, fill = 'skyblue', color = 'black') +
  labs(title = "Distribution of Log-Transformed Antibody Levels")
```



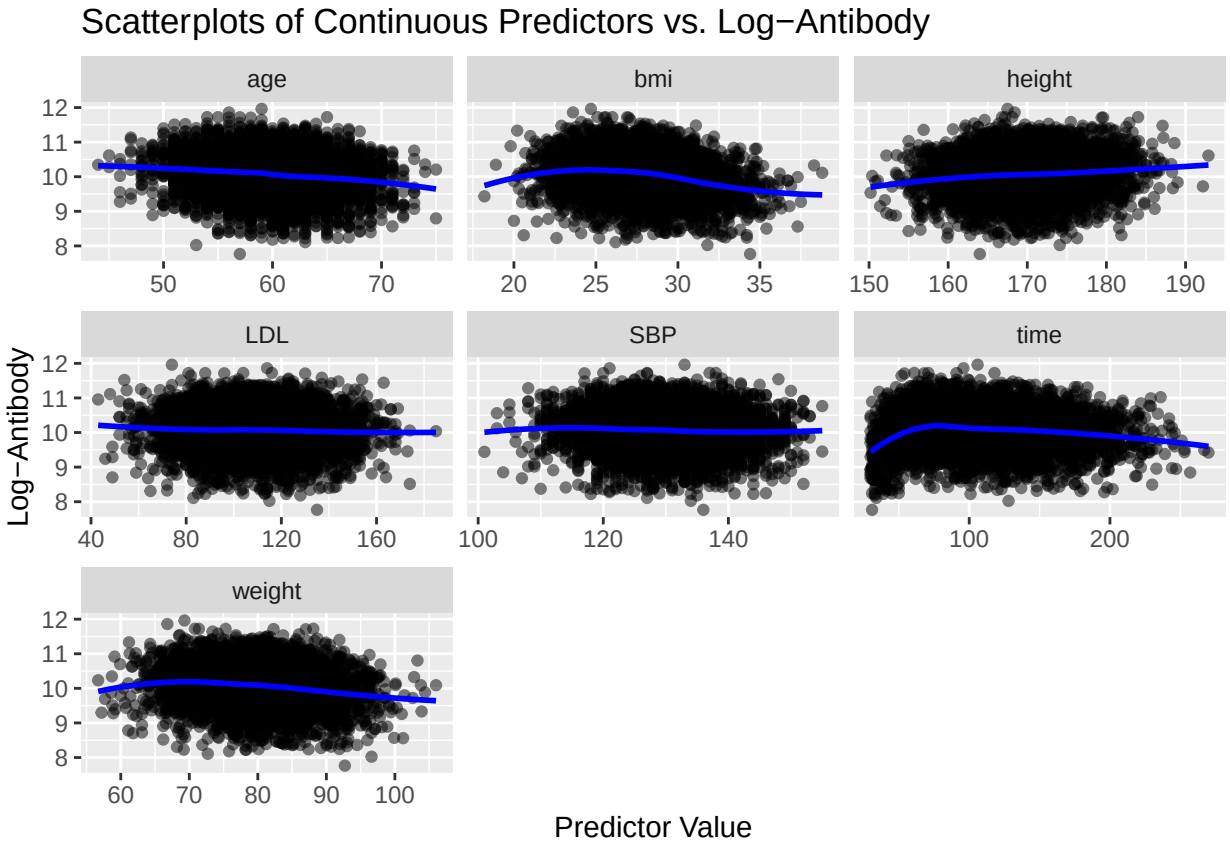

```
ggplot(dat2, aes(x = log_antibody)) +
  geom_histogram(binwidth = 0.2, fill = 'skyblue', color = 'black') +
  labs(title = "Distribution of Log-Transformed Antibody Levels")
```




### Scatterplots of Continuous Predictors vs. Log-Antibody

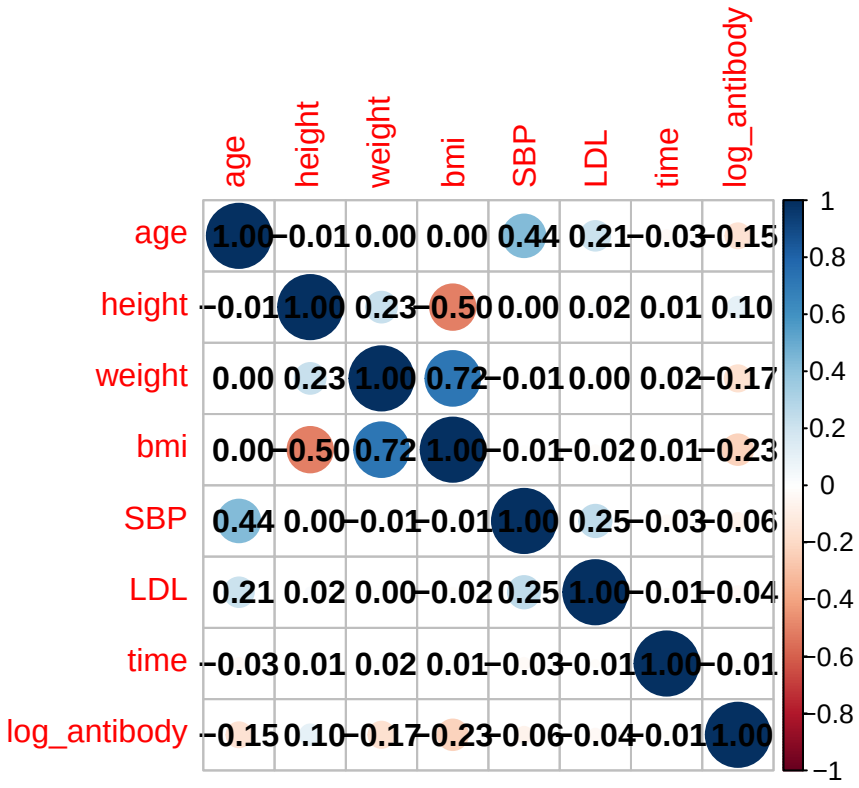
```
dat1 %>%
  select(age, height, weight, bmi, SBP, LDL, time, log_antibody) %>%
  pivot_longer(
    cols = c(age,  height, weight, bmi, SBP, LDL, time),
    names_to = "predictor",
    values_to = "value"
  ) %>%
  ggplot(aes(x = value, y = log_antibody)) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "loess", se = FALSE, color = "blue") +
  facet_wrap(~ predictor, scales = "free_x") +
  labs(
    x = "Predictor Value",
    y = "Log-Antibody",
    title = "Scatterplots of Continuous Predictors vs. Log-Antibody"
  )
```

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




### Correlation Matrix of Continuous Variables

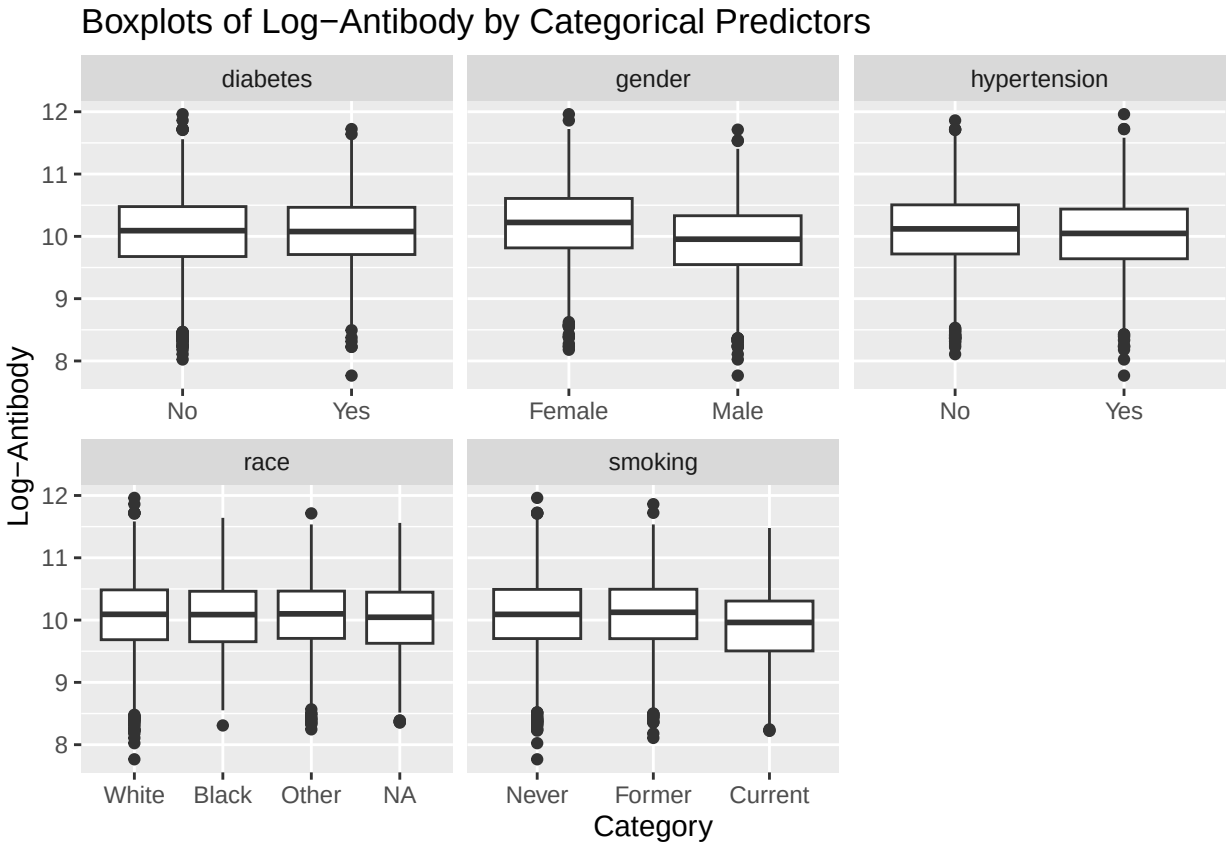
```
dat1 %>%
  select(age, height, weight, bmi, SBP, LDL, time, log_antibody) %>%
  cor(use = "complete.obs") %>%
  corrplot(type = "full",
           title = "Correlation Matrix of Continuous Variables",
           addCoef.col = "black",
           mar = c(0,0,2,0))
```




Boxplots of log_antibody by categorical variables

```
dat1 %>%
  select(log_antibody, gender, race, smoking, diabetes, hypertension) %>%
  mutate(
    gender = factor(dat1$gender,
                    levels = c(0, 1),
                    labels = c("Female", "Male")),
    race = factor(dat1$race,
                  levels = c(1, 2, 3),
                  labels = c("White", "Black", "Other")),
    smoking = factor(dat1$smoking,
                     levels = c(0, 1, 2),
                     labels = c("Never", "Former", "Current")),
    diabetes = factor(dat1$diabetes,
                      levels = c(0, 1),
                      labels = c("No", "Yes")),
    hypertension = factor(dat1$hypertension,
                          levels = c(0, 1),
                          labels = c("No", "Yes"))
  ) %>%
  pivot_longer(
    cols = c(gender, race, smoking, diabetes, hypertension),
    names_to = "predictor",
```

```
    values_to = "category"
  ) %>%
  ggplot(aes(x = category, y = log_antibody)) +
  geom_boxplot() +
  facet_wrap(~ predictor, scales = "free_x") +
  labs(
    x = "Category",
    y = "Log-Antibody",
    title = "Boxplots of Log-Antibody by Categorical Predictors"
  )
```




## Models Building

cross-validation

```
set.seed(123)
ctrl = trainControl(method = "cv", number = 10)
```

### LASSO

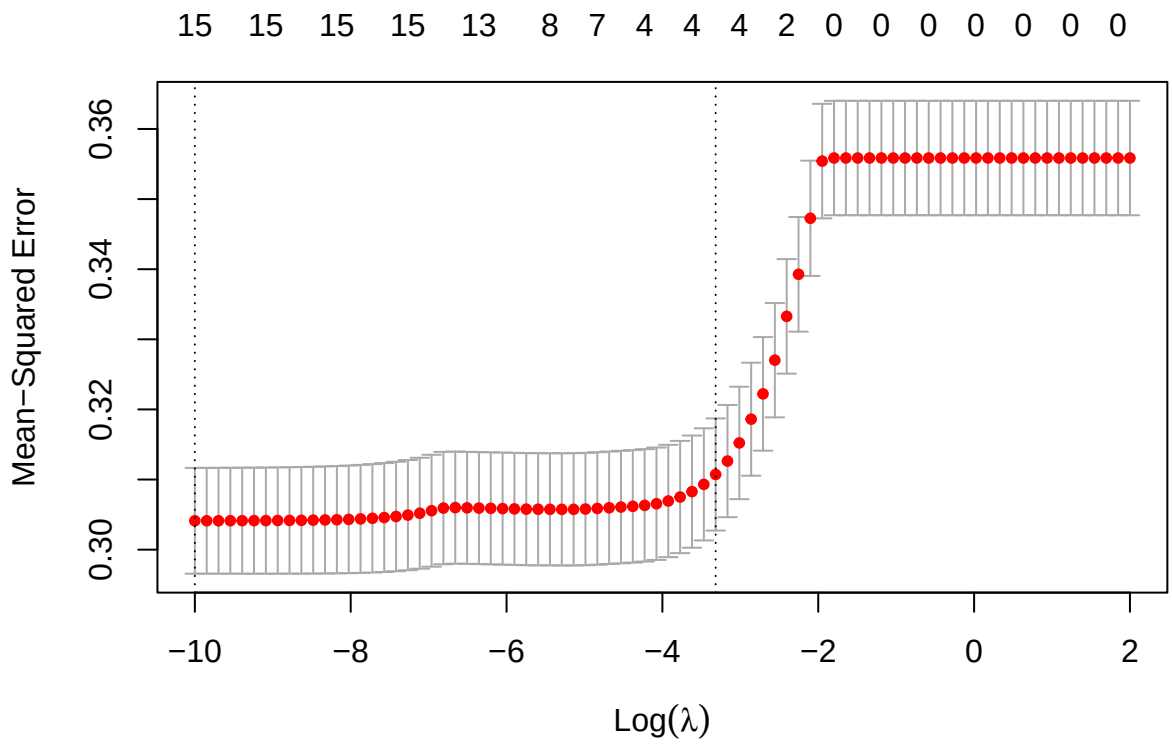
```r
x <- model.matrix(log_antibody ~ . , data = dat1)[, -1]
y <- dat1$log_antibody

cv_lasso <- cv.glmnet(x, y, alpha = 1, lambda = exp(seq(2, -10, length = 80)))
cv_lasso$lambda.min
```

```
## [1] 4.539993e-05
```

```r
cv_lasso$lambda.1se
```

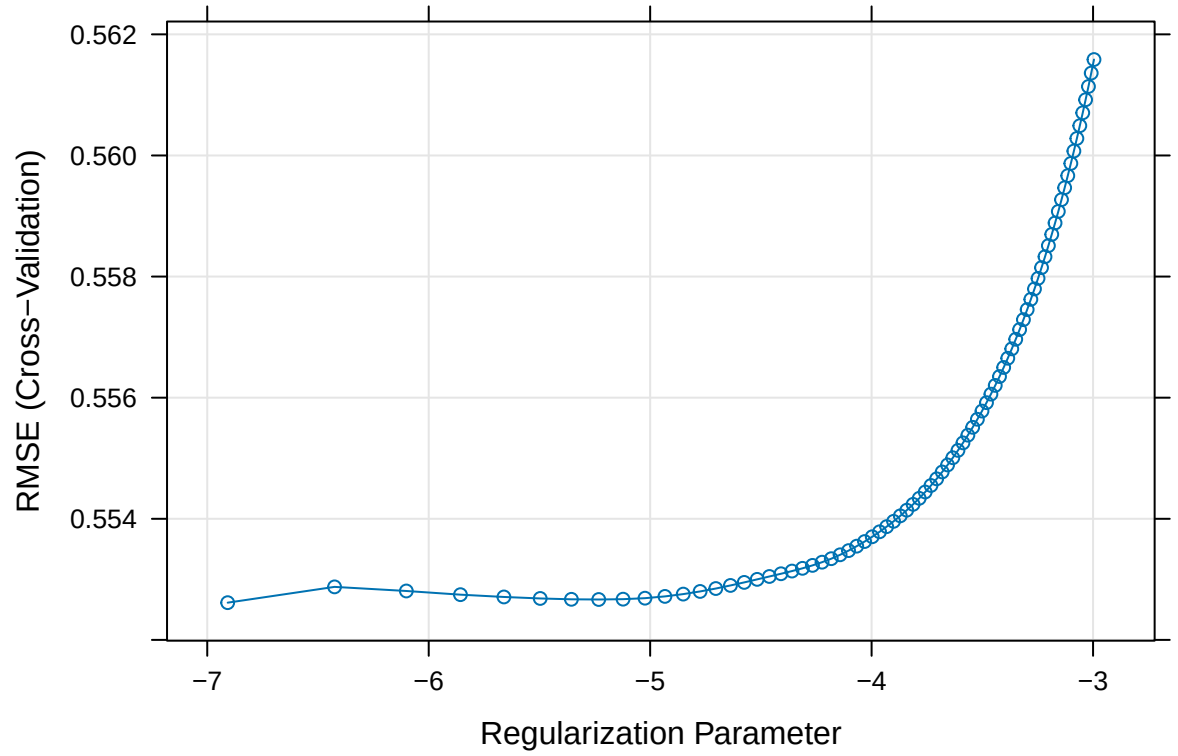
```
## [1] 0.0362812
```

```r
plot(cv_lasso)
```




```r
model_lasso <- train(
  log_antibody ~ .,
  data = dat1,
  method = "glmnet",
  trControl = ctrl,
  tuneGrid = expand.grid(alpha = 1,
                         lambda = seq(0.001, 0.05, length = 80))

)
```

```r
plot(model_lasso, xTrans = log)
```




```r
model_lasso$bestTune
```

```
##   alpha lambda
## 1     1  0.001
```

## GAM

```r
model_gam <- train(
  log_antibody ~ .,
  data = dat1,
  method = "gam",
  trControl = ctrl
)

model_gam$finalModel
```

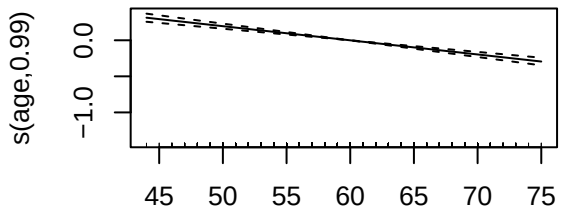
```
##
## Family: gaussian
## Link function: identity
##
```

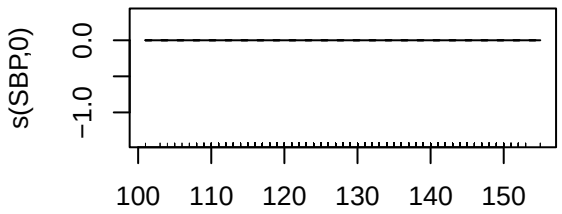
```
## Formula:
## .outcome ~ gender + race2 + race3 + race4 + smoking1 + smoking2 +
##     diabetes + hypertension + s(age) + s(SBP) + s(LDL) + s(bmi) +
##     s(time) + s(height) + s(weight)
##
## Estimated degrees of freedom:
## 0.991 0.000 0.000 4.179 7.892 1.234 0.000
##  total = 23.3
##
## GCV score: 0.2786734
```

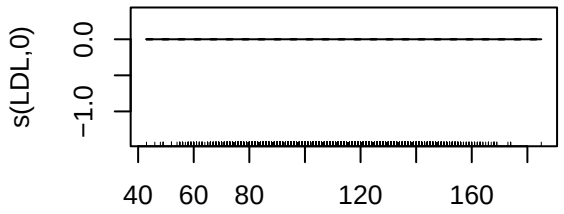
```
model_gam$bestTune
```

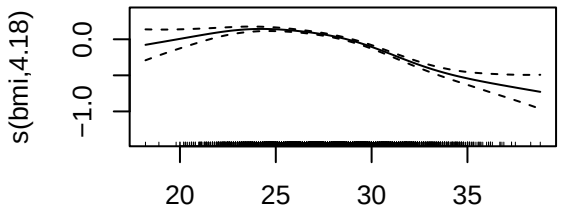
```
##   select method
## 2   TRUE GCV.Cp
```

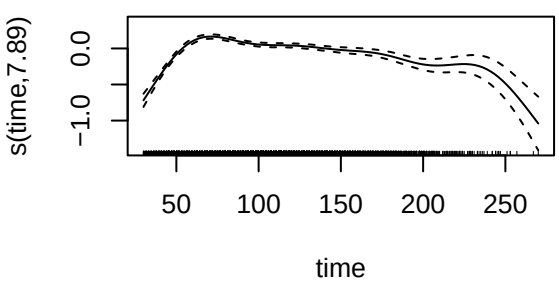
```
par(mfrow = c(2,2))
plot(model_gam$finalModel)
```

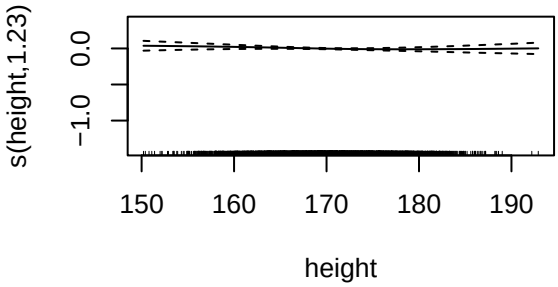


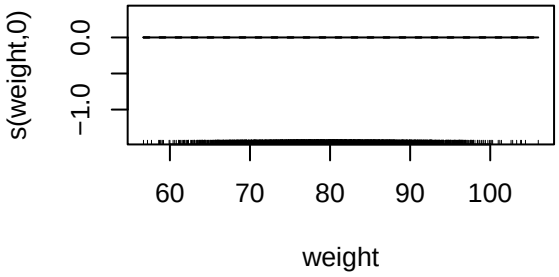



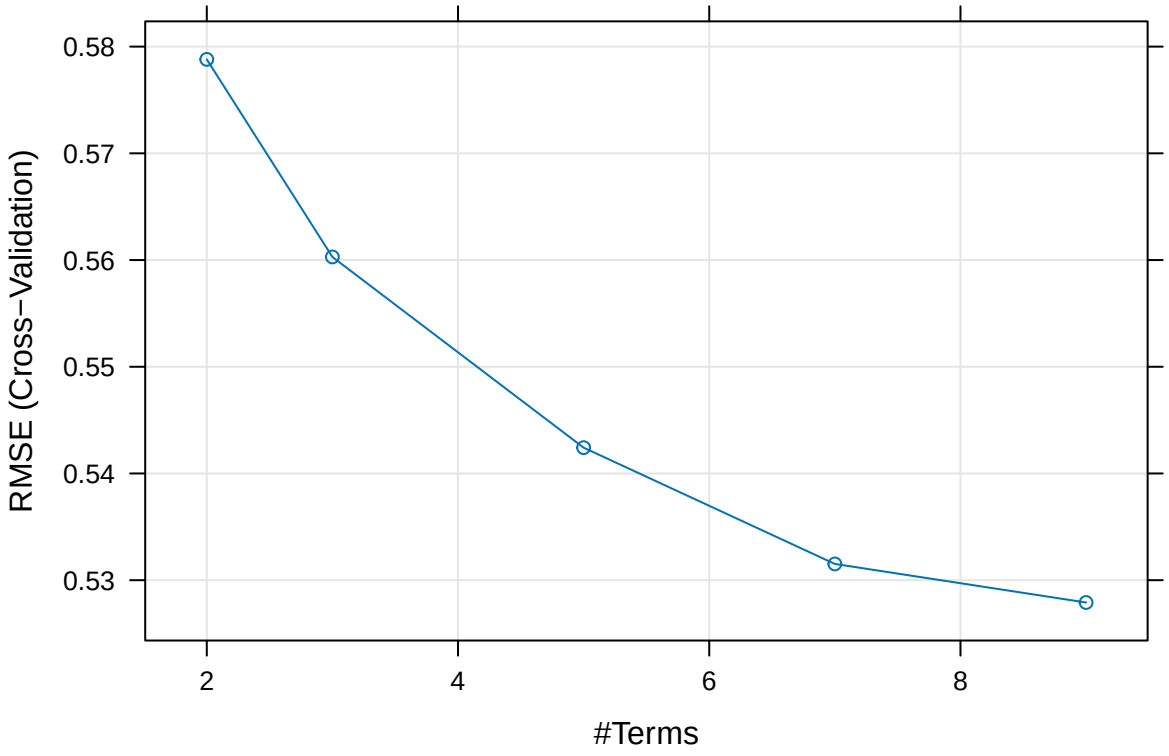



## MARS

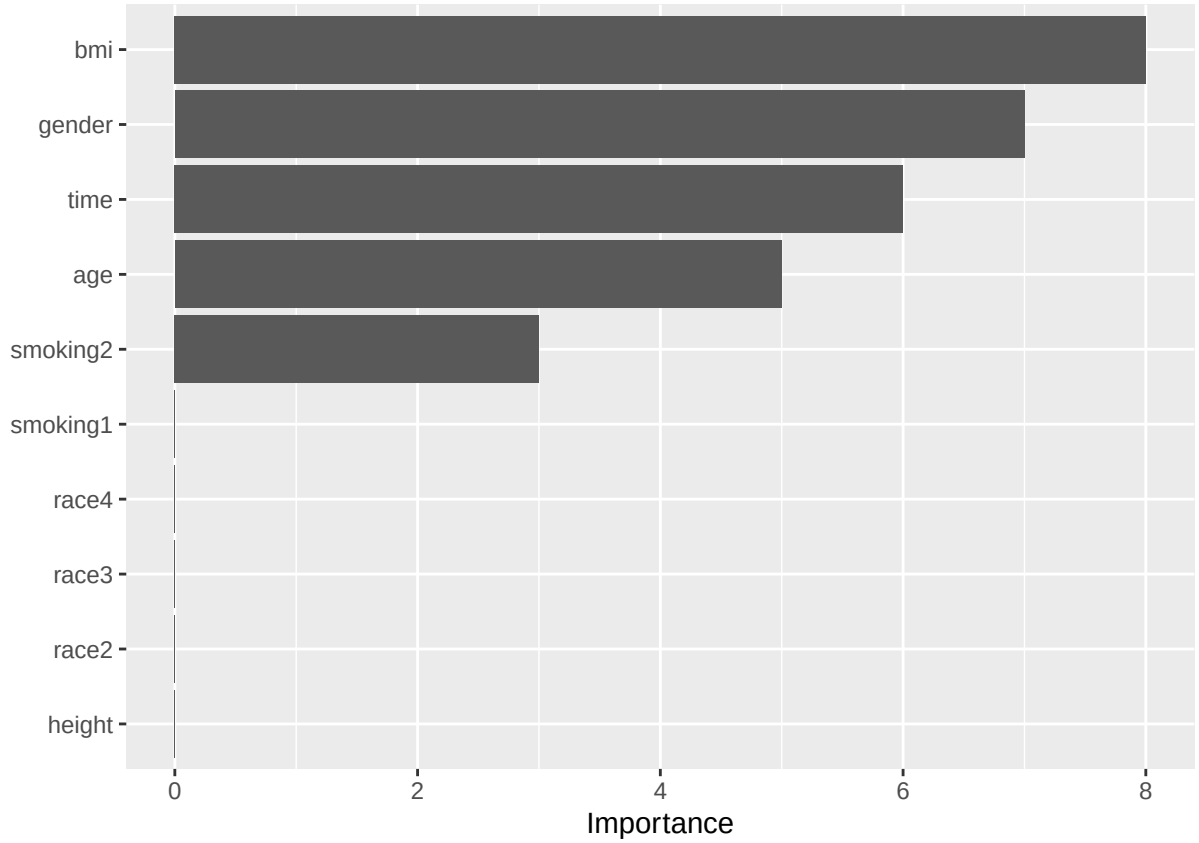
```
model_mars <- train(
  log_antibody ~ .,
  data = dat1,
  method = "earth",
  trControl = ctrl,
  tuneLength = 5
)

plot(model_mars)
```



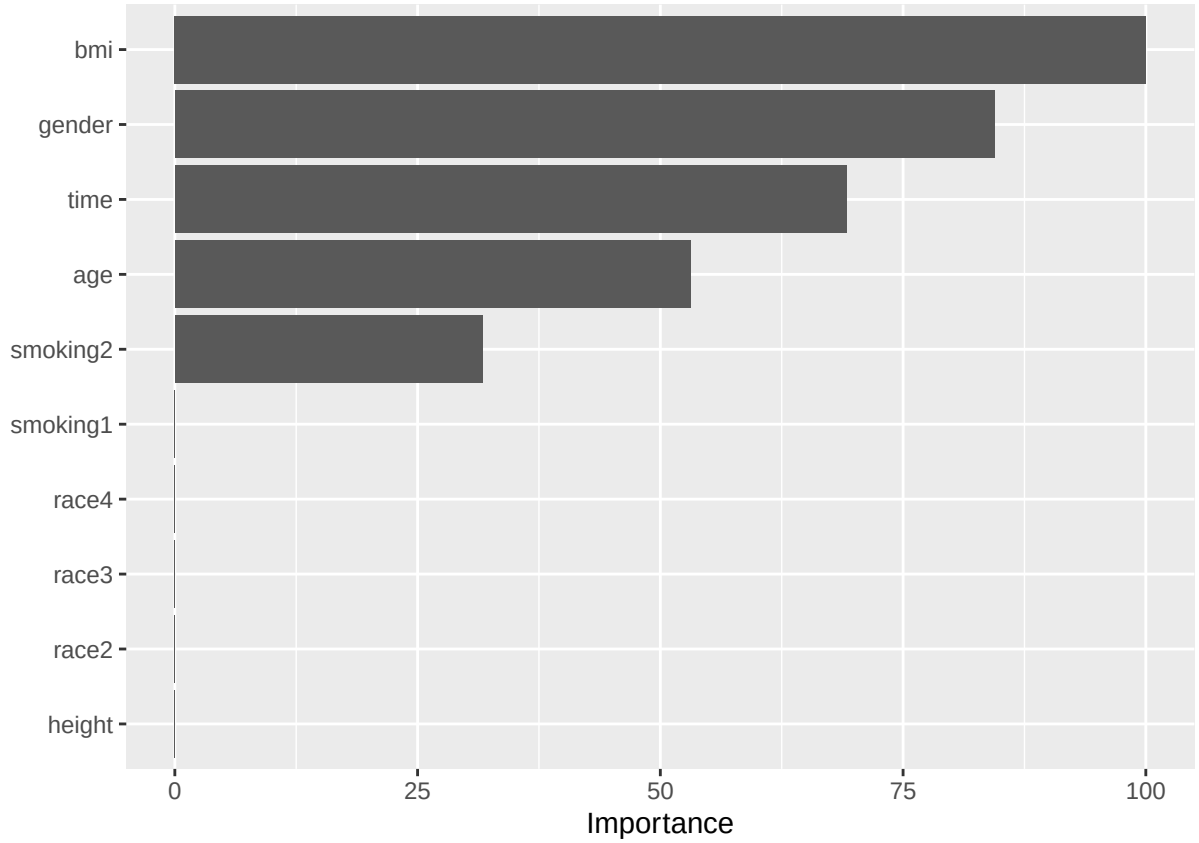

```
coef(model_mars$finalModel)
```

```
##  (Intercept)  h(27.8-bmi)   h(time-57)   h(57-time)       gender    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
```

```
vip(model_mars$finalModel, type = "nsubsets")
```




```
vip(model_mars$finalModel, type = "rss")
```




## Predictions and Model Evaluation

```
pred_lasso <- predict(model_lasso, newdata = dat2)
pred_mars  <- predict(model_mars, newdata = dat2)
pred_gam   <- predict(model_gam, newdata = dat2)

resample = resamples(list(lasso = model_lasso, gam = model_gam, mars = model_mars))
summary(resample)
```

```
##
## Call:
## summary.resamples(object = resample)
##
## Models: lasso, gam, mars
## Number of resamples: 10
##
## MAE
##            Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## lasso 0.4302898 0.4347903 0.4384044 0.4402713 0.4443131 0.4543854    0
## gam   0.4075818 0.4191554 0.4252307 0.4228909 0.4269059 0.4340199    0
## mars  0.3998365 0.4142026 0.4225842 0.4225721 0.4296486 0.4466906    0
##
## RMSE
```

```
##            Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## lasso 0.5419531 0.5451779 0.5484308 0.5526149 0.5607064 0.5679933    0
## gam   0.5114329 0.5250503 0.5291679 0.5284976 0.5310636 0.5503925    0
## mars  0.4983898 0.5196328 0.5275137 0.5279091 0.5352447 0.5623342    0
##
## Rsquared
##            Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## lasso 0.1174197 0.1295564 0.1442448 0.1419938 0.1558283 0.1614893    0
## gam   0.1640261 0.1884887 0.2122991 0.2157631 0.2358544 0.2821440    0
## mars  0.1850330 0.2027610 0.2069534 0.2167496 0.2415450 0.2521676    0
```

# GAM

pred_gam <- predict(gam.fit$finalModel, newdata = dat2) sqrt(mean((pred_gam - y2)^2))

# Antibody level and time

```r
library(splines)
model_spline <- lm(log_antibody ~ ns(time, df = 4), data = dat1)
summary(model_spline)
```

```
##
## Call:
## lm(formula = log_antibody ~ ns(time, df = 4), data = dat1)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.03592 -0.36964  0.01735  0.40630  1.86009
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)        9.47141    0.03945 240.066  < 2e-16 ***
## ns(time, df = 4)1  0.55828    0.04100  13.616  < 2e-16 ***
## ns(time, df = 4)2  0.28006    0.05069   5.525 3.46e-08 ***
## ns(time, df = 4)3  1.16159    0.10205  11.383  < 2e-16 ***
## ns(time, df = 4)4 -0.44128    0.11083  -3.981 6.95e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5792 on 4995 degrees of freedom
## Multiple R-squared:  0.05771,    Adjusted R-squared:  0.05695
## F-statistic: 76.48 on 4 and 4995 DF,  p-value: < 2.2e-16
```