

Data Science II Midterm

Megan Panier, Shiyong Wu, and Rita Wang

2025-03-25

Libraries

```
library(readxl) # to import excel files
library(tidyverse)
library(corrplot)
library(ggplot2)
library(tidymodels)
library(glmnet)
library(caret)
library(splines)
library(mgcv)
library(pROC)
library(vip)
library(AppliedPredictiveModeling)
library(tidymodels)
library(earth)
library(pdp)
```

Importing and Organizing Data

```
load("./data/dat1.RData") #importing training data
# Log-transformed antibody level (log_antibody) --> y
initial_training = dat1 #renaming the original training data name

load("./data/dat2.RData") #importing training data
initial_test = dat2 #renaming the original training data name

set.seed(2222)

# partition data into training and validation data sets
datSplit = initial_split(data = initial_training, prop = 0.8)
training = training(datSplit)
validation = testing(datSplit)
```

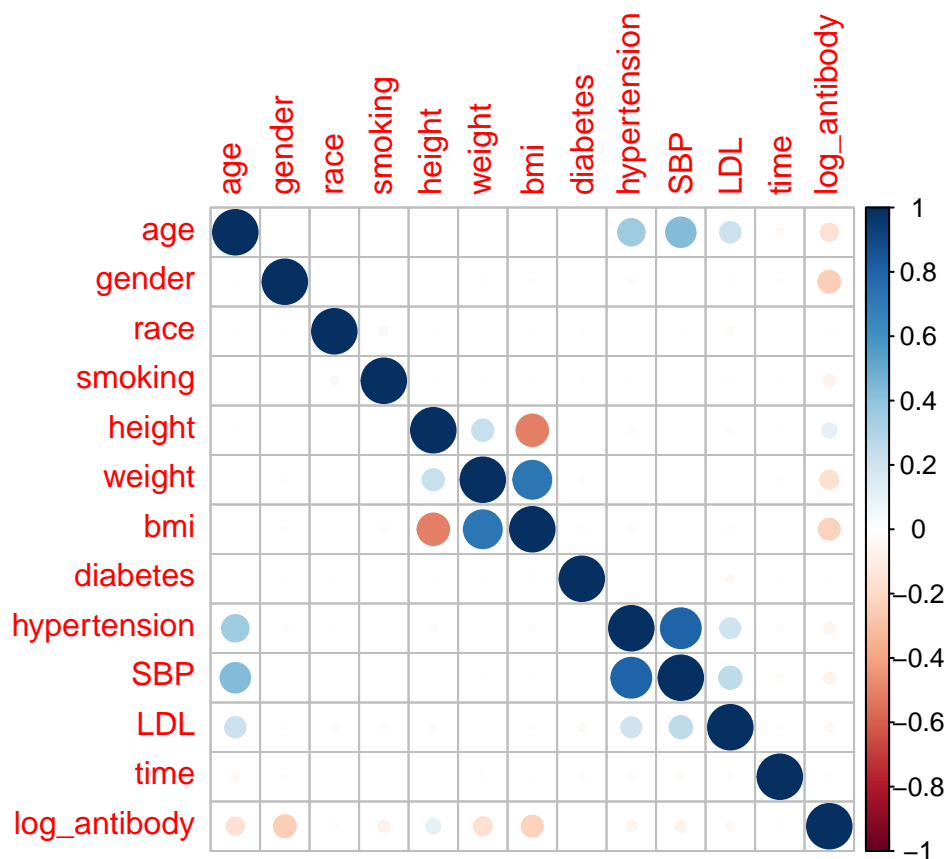
Exploratory Analysis

```

Exploratory_train <- initial_training
Exploratory_train$race <- as.numeric(Exploratory_train$race)
Exploratory_train$smoking <- as.numeric(Exploratory_train$smoking)

train_cor_matrix <- cor(Exploratory_train[, !names(Exploratory_train) %in% c("id")], use = "complete.obs")
corrplot(train_cor_matrix, method = "circle")

```



```
round(train_cor_matrix, 2)
```

```

##          age gender  race smoking height weight  bmi diabetes
## age      1.00 -0.01 -0.01   0.00  -0.01   0.00  0.00   0.00
## gender  -0.01  1.00 -0.01   0.00   0.01  -0.01 -0.02  -0.01
## race    -0.01 -0.01  1.00   0.04  -0.01   0.00  0.01   0.01
## smoking  0.00  0.00  0.04   1.00  -0.01   0.01  0.01   0.01
## height  -0.01  0.01 -0.01  -0.01   1.00   0.23 -0.50  -0.01
## weight   0.00 -0.01  0.00   0.01   0.23   1.00  0.72   0.01
## bmi      0.00 -0.02  0.01   0.01  -0.50   0.72  1.00   0.02
## diabetes 0.00 -0.01  0.01   0.01  -0.01   0.01  0.02   1.00
## hypertension 0.35  0.02  0.00  -0.01   0.03   0.00 -0.02   0.00
## SBP      0.44  0.00  0.01   0.00   0.00  -0.01 -0.01   0.00
## LDL      0.21  0.01 -0.03   0.01   0.02   0.00 -0.02  -0.03
## time     -0.03 -0.02  0.00  -0.01   0.01   0.02  0.01  -0.01
## log_antibody -0.15 -0.24 -0.02 -0.06   0.10  -0.17 -0.23   0.01

```

```
##          hypertension    SBP    LDL    time log_antibody
## age                0.35  0.44  0.21 -0.03        -0.15
## gender              0.02  0.00  0.01 -0.02        -0.24
## race                0.00  0.01 -0.03  0.00        -0.02
## smoking            -0.01  0.00  0.01 -0.01        -0.06
## height              0.03  0.00  0.02  0.01         0.10
## weight              0.00 -0.01  0.00  0.02        -0.17
## bmi                -0.02 -0.01 -0.02  0.01        -0.23
## diabetes            0.00  0.00 -0.03 -0.01         0.01
## hypertension        1.00  0.80  0.20 -0.02        -0.06
## SBP                 0.80  1.00  0.25 -0.03        -0.06
## LDL                 0.20  0.25  1.00 -0.01        -0.04
## time                -0.02 -0.03 -0.01  1.00        -0.01
## log_antibody        -0.06 -0.06 -0.04 -0.01         1.00
```

```
Exploratory_test <- initial_test
Exploratory_test$race <- as.numeric(Exploratory_test$race)
Exploratory_test$smoking <- as.numeric(Exploratory_test$smoking)
str(Exploratory_test)
```

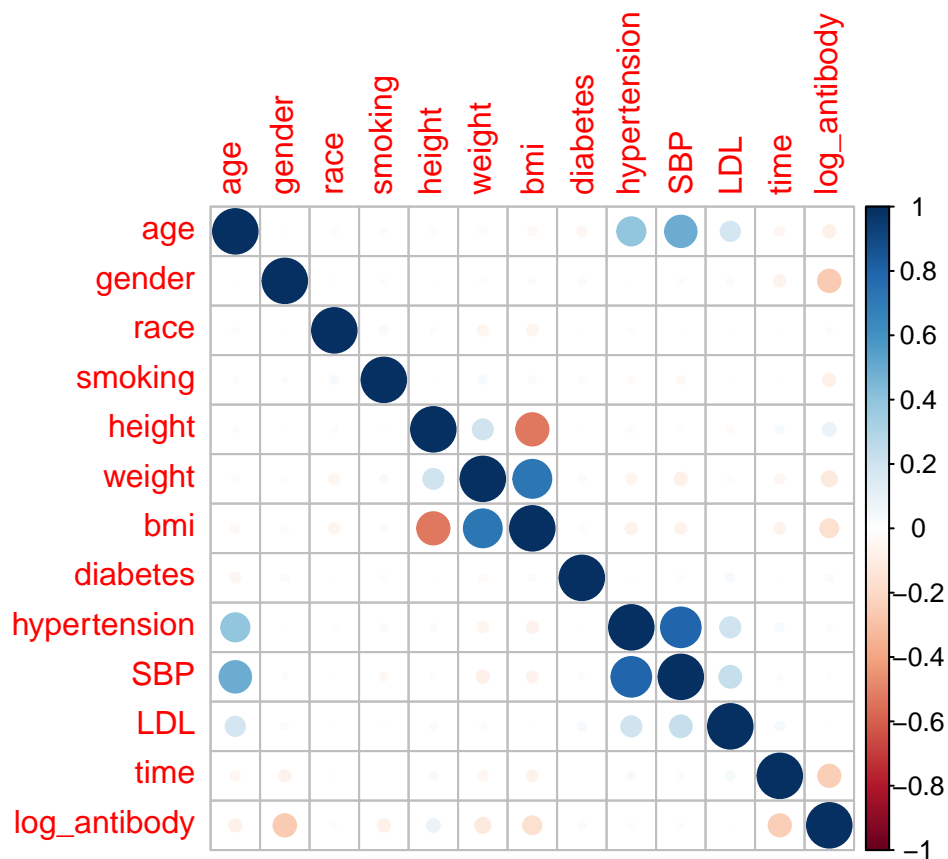
```
## 'data.frame':    1000 obs. of  14 variables:
## $ id      : int  5001 5002 5003 5004 5005 5006 5007 5008 5009 5010 ...
## $ age     : num  58 62 71 59 69 56 65 61 62 68 ...
## $ gender  : int  0 0 0 1 1 0 0 1 0 0 ...
## $ race    : num  4 1 4 1 1 1 1 1 1 4 ...
## $ smoking : num  2 2 1 1 1 1 1 2 1 1 ...
## $ height  : num  176 168 179 170 166 ...
## $ weight  : num  86.4 82.4 79.2 81 74.8 74.8 69.2 81.3 82.1 74.4 ...
## $ bmi     : num  27.7 29.4 24.6 28 27 26.6 22.4 27.4 30.7 26.7 ...
## $ diabetes: int  0 1 1 0 1 0 0 0 0 0 ...
## $ hypertension: num  0 0 1 0 1 0 1 0 1 1 ...
## $ SBP     : num  130 123 145 123 150 121 132 120 142 137 ...
## $ LDL     : num  115 118 149 119 142 112 127 76 86 123 ...
## $ time    : num  205 229 206 163 240 206 285 185 124 127 ...
## $ log_antibody: num  9.81 9.08 10.43 9.83 9.07 ...
```

```
test_cor_matrix <- cor(Exploratory_test[, !names(Exploratory_test) %in% c("id")], use = "complete.obs")
round(test_cor_matrix, 2)
```

```
##          age gender  race smoking height weight  bmi diabetes
## age      1.00  0.00  0.02  -0.02  0.02  -0.02 -0.03  -0.05
## gender   0.00  1.00 -0.01  -0.01  0.01  -0.02 -0.02  -0.03
## race     0.02 -0.01  1.00   0.03  0.02  -0.05 -0.06   0.00
## smoking  -0.02 -0.01  0.03   1.00  0.00  0.03  0.02  -0.02
## height   0.02  0.01  0.02   0.00  1.00  0.20 -0.53  -0.01
## weight   -0.02 -0.02 -0.05   0.03  0.20  1.00  0.72  -0.03
## bmi      -0.03 -0.02 -0.06   0.02 -0.53  0.72  1.00  -0.02
## diabetes -0.05 -0.03  0.00  -0.02 -0.01 -0.03 -0.02   1.00
## hypertension 0.40  0.00 -0.01  -0.02  0.02  -0.06 -0.07  -0.01
## SBP       0.50  0.02  0.00  -0.03 -0.01 -0.08 -0.06  -0.01
## LDL       0.19 -0.02  0.00   0.00 -0.02 -0.02  0.00   0.04
## time      -0.04 -0.07 -0.01   0.00  0.03  -0.05 -0.06   0.00
## log_antibody -0.08 -0.25 -0.01  -0.08  0.08  -0.11 -0.16   0.03
```

```
##      hypertension    SBP    LDL    time log_antibody
## age                0.40  0.50  0.19 -0.04      -0.08
## gender              0.00  0.02 -0.02 -0.07      -0.25
## race               -0.01  0.00  0.00 -0.01      -0.01
## smoking            -0.02 -0.03  0.00  0.00      -0.08
## height              0.02 -0.01 -0.02  0.03       0.08
## weight             -0.06 -0.08 -0.02 -0.05      -0.11
## bmi                -0.07 -0.06  0.00 -0.06      -0.16
## diabetes            -0.01 -0.01  0.04  0.00       0.03
## hypertension        1.00  0.79  0.21  0.03       0.02
## SBP                 0.79  1.00  0.24  0.02      -0.01
## LDL                 0.21  0.24  1.00  0.04       0.00
## time                0.03  0.02  0.04  1.00      -0.25
## log_antibody        0.02 -0.01  0.00 -0.25       1.00
```

```
corrplot(test_cor_matrix, method = "circle")
```



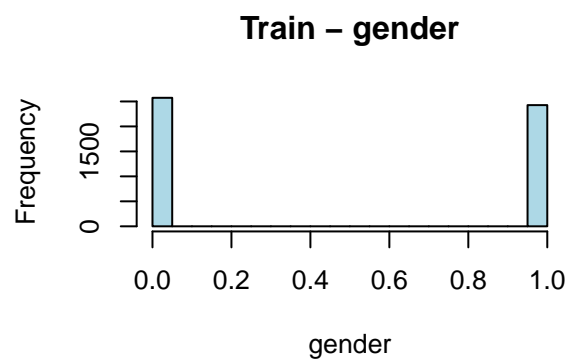
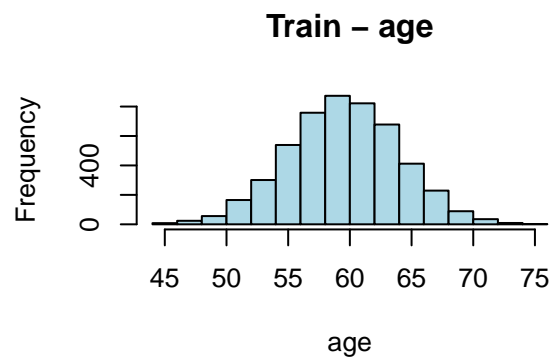
```
measurement<-data.frame(
  Train_Mean = sapply(Exploratory_train, mean, na.rm = TRUE),
  Test_Mean = sapply(Exploratory_test, mean, na.rm = TRUE),
  Train_SD = sapply(Exploratory_train, sd, na.rm = TRUE),
  Test_SD = sapply(Exploratory_test, sd, na.rm = TRUE),
  Train_Min = sapply(Exploratory_train, min, na.rm = TRUE),
  Train_Max = sapply(Exploratory_train, max, na.rm = TRUE),
  Test_Min = sapply(Exploratory_test, min, na.rm = TRUE),
```

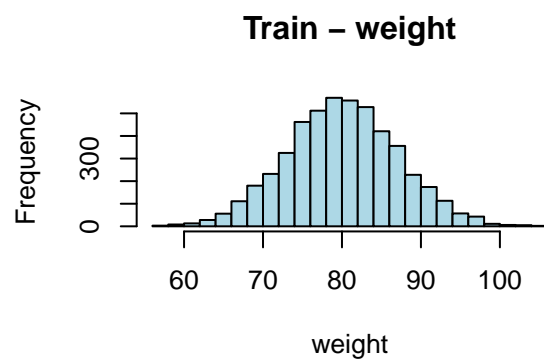
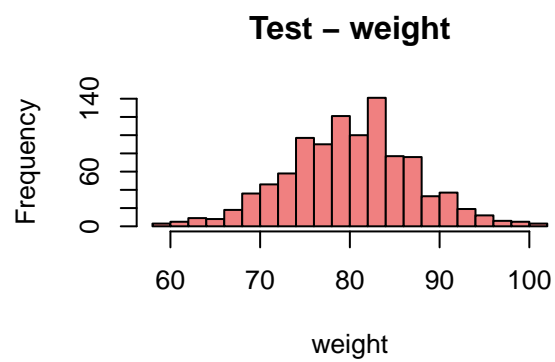
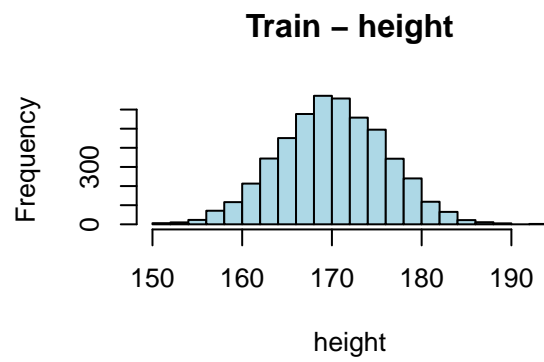
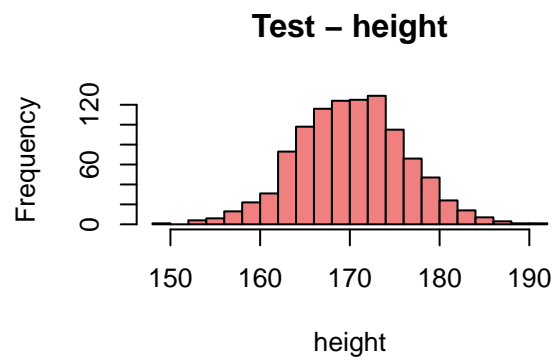
```
Test_Max = supply(Exploratory_test, max, na.rm = TRUE))
round(measurement, 2)
```

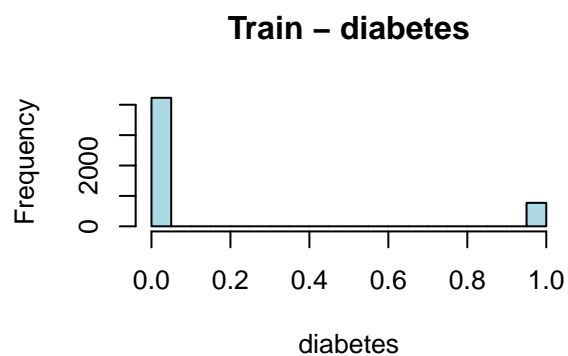
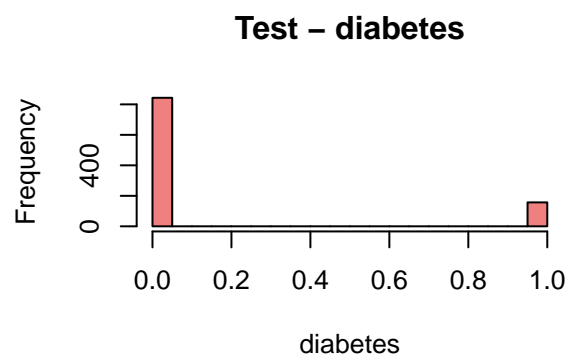
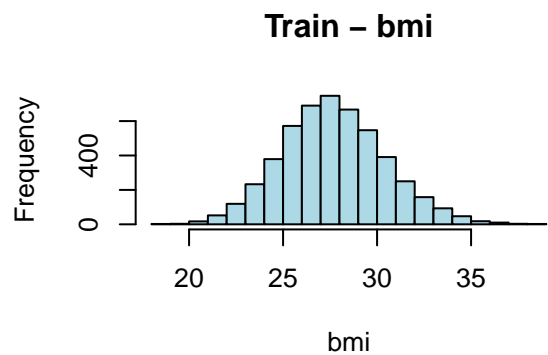
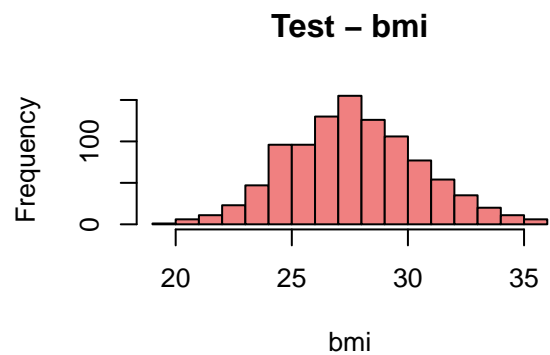
```
##           Train_Mean Test_Mean Train_SD Test_SD Train_Min Train_Max Test_Min
## id           2500.50   5500.50  1443.52  288.82         1.00   5000.00   5001.00
## age           59.97    60.02    4.50    4.45        44.00    75.00    46.00
## gender         0.49     0.49    0.50    0.50         0.00     1.00     0.00
## race           1.75     1.70    1.08    1.05         1.00     4.00     1.00
## smoking        1.50     1.50    0.67    0.68         1.00     3.00     1.00
## height        170.13   170.22    5.94    6.02       150.20   192.90   149.40
## weight         80.11    80.13    7.06    7.05        56.70   106.00    58.80
## bmi            27.74    27.72    2.76    2.82        18.20    38.80    19.80
## diabetes        0.15     0.16    0.36    0.36         0.00     1.00     0.00
## hypertension    0.46     0.46    0.50    0.50         0.00     1.00     0.00
## SBP            129.90   129.61    8.00    8.20       101.00   155.00   106.00
## LDL            109.91   110.25   20.15   20.32        43.00   185.00    46.00
## time           108.86   173.77   43.42   46.78        30.00   270.00    61.00
## log_antibody    10.06     9.90    0.60    0.59         7.77    11.96     8.05
```

```
##           Test_Max
## id           6000.00
## age           75.00
## gender         1.00
## race           4.00
## smoking        3.00
## height        190.60
## weight        101.60
## bmi            35.80
## diabetes        1.00
## hypertension    1.00
## SBP            156.00
## LDL            174.00
## time           330.00
## log_antibody    11.85
```

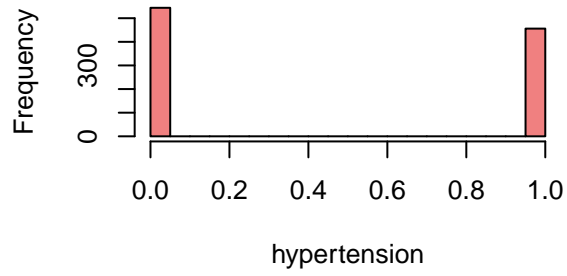
```
train_vars <- initial_training[, !names(initial_training) %in% c("id", "race", "smoking")]
test_vars  <- initial_test[, !names(initial_test) %in% c("id", "race", "smoking")]
par(mfrow = c(2, 2))
for (var in names(train_vars)) {
  hist(test_vars[[var]],
        main = paste("Test -", var),
        xlab = var,
        col = "lightcoral",
        breaks = 20)
  hist(train_vars[[var]],
        main = paste("Train -", var),
        xlab = var,
        col = "lightblue",
        breaks = 20)
}
```



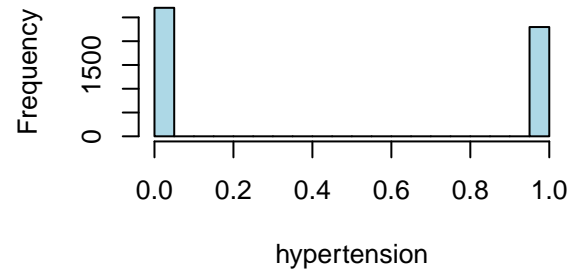




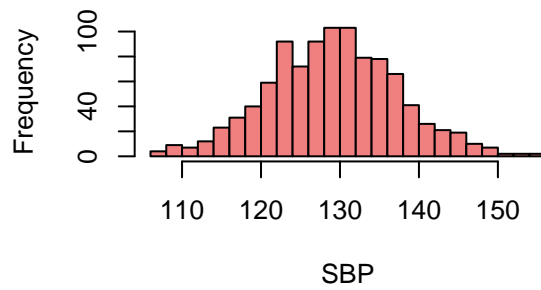
Test – hypertension



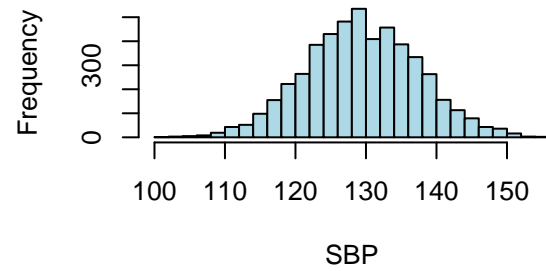
Train – hypertension

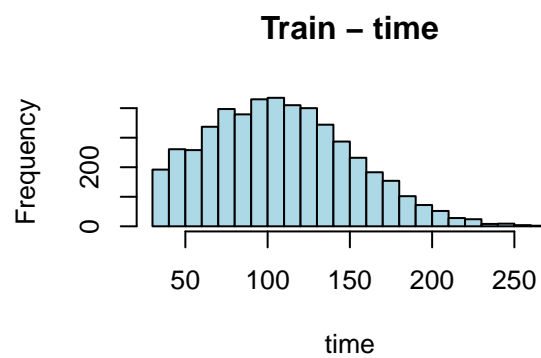
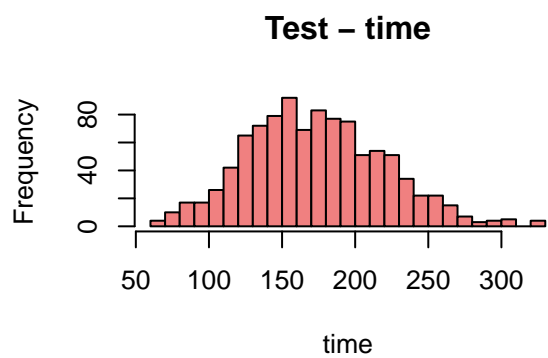
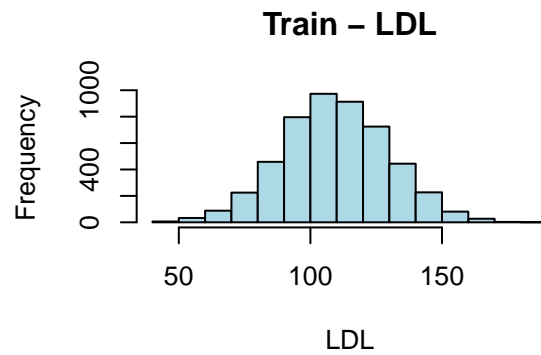
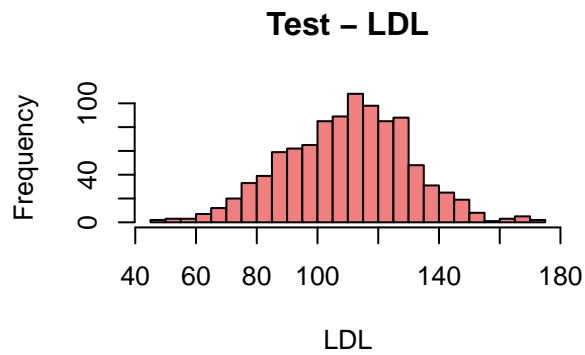


Test – SBP



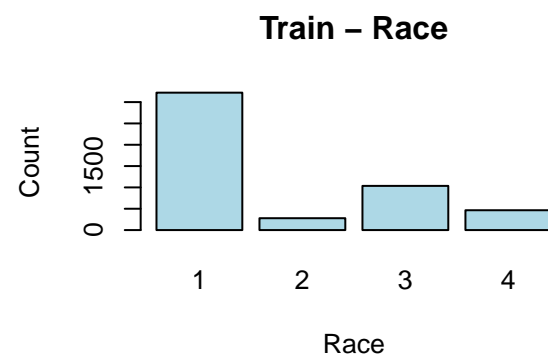
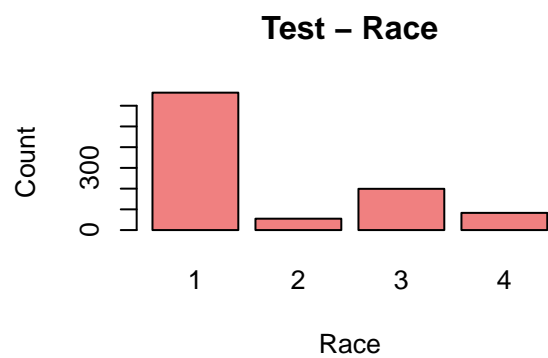
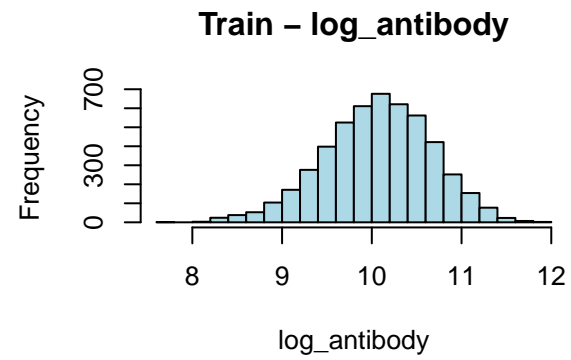
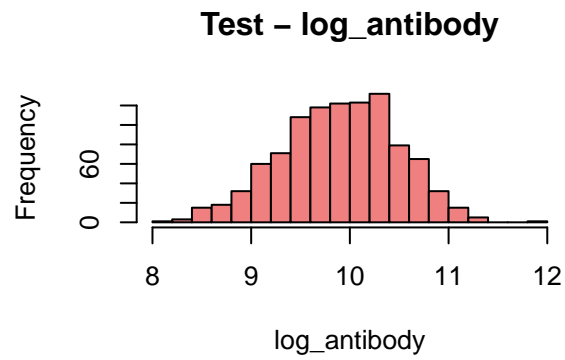
Train – SBP





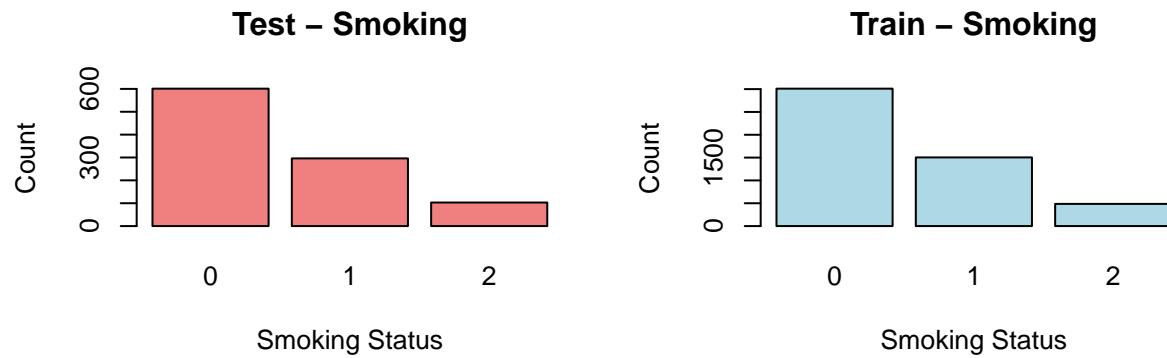
```
barplot(table(initial_test$race),
        main = "Test - Race",
        col = "lightcoral",
        xlab = "Race",
        ylab = "Count")

barplot(table(initial_training$race),
        main = "Train - Race",
        col = "lightblue",
        xlab = "Race",
        ylab = "Count")
```



```
barplot(table(initial_test$smoking),
  main = "Test - Smoking",
  col = "lightcoral",
  xlab = "Smoking Status",
  ylab = "Count")

barplot(table(initial_training$smoking),
  main = "Train - Smoking",
  col = "lightblue",
  xlab = "Smoking Status",
  ylab = "Count")
```



Linear Regression

```
model = lm(log_antibody ~ age + gender + race + smoking + bmi + diabetes +
            hypertension + LDL + time, data = training)
```

```
# View the model summary
summary(model)
```

```
##
## Call:
## lm(formula = log_antibody ~ age + gender + race + smoking + bmi +
##     diabetes + hypertension + LDL + time, data = training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.13184 -0.35446  0.03155  0.38071  1.57178
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.279e+01  1.540e-01  83.056 < 2e-16 ***
## age         -1.873e-02  2.094e-03  -8.945 < 2e-16 ***
## gender       -2.770e-01  1.741e-02 -15.916 < 2e-16 ***
## race2        -1.263e-02  3.877e-02  -0.326  0.7447
```

```
## race3      -8.852e-03  2.193e-02  -0.404   0.6864
## race4      -4.747e-02  3.027e-02  -1.569   0.1168
## smoking1    2.452e-02  1.944e-02   1.262   0.2072
## smoking2   -1.757e-01  2.986e-02  -5.885  4.3e-09 ***
## bmi         -5.054e-02  3.170e-03 -15.941 < 2e-16 ***
## diabetes    1.688e-03  2.445e-02   0.069   0.9450
## hypertension -3.728e-03  1.882e-02  -0.198   0.8430
## LDL         -8.155e-05  4.500e-04  -0.181   0.8562
## time        -3.720e-04  1.996e-04  -1.864   0.0624 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5495 on 3987 degrees of freedom
## Multiple R-squared:  0.139, Adjusted R-squared:  0.1364
## F-statistic: 53.62 on 12 and 3987 DF, p-value: < 2.2e-16
```

```
predictions_train = predict(model, newdata = validation)

# RMSE
rmse_train = sqrt(mean((predictions_train - validation$log_antibody)^2))
rmse_train
```

```
## [1] 0.5636857
```

```
# R^2
rsq_train = 1 - sum((predictions_train - validation$log_antibody)^2) /
  sum((mean(training$log_antibody) - validation$log_antibody)^2)
rsq_train
```

```
## [1] 0.1648078
```

```
generalization = predict(model, newdata = initial_test)

# Calculate RMSE for dat2
rmse_dat2 = sqrt(mean((generalization - initial_test$log_antibody)^2))
rmse_dat2
```

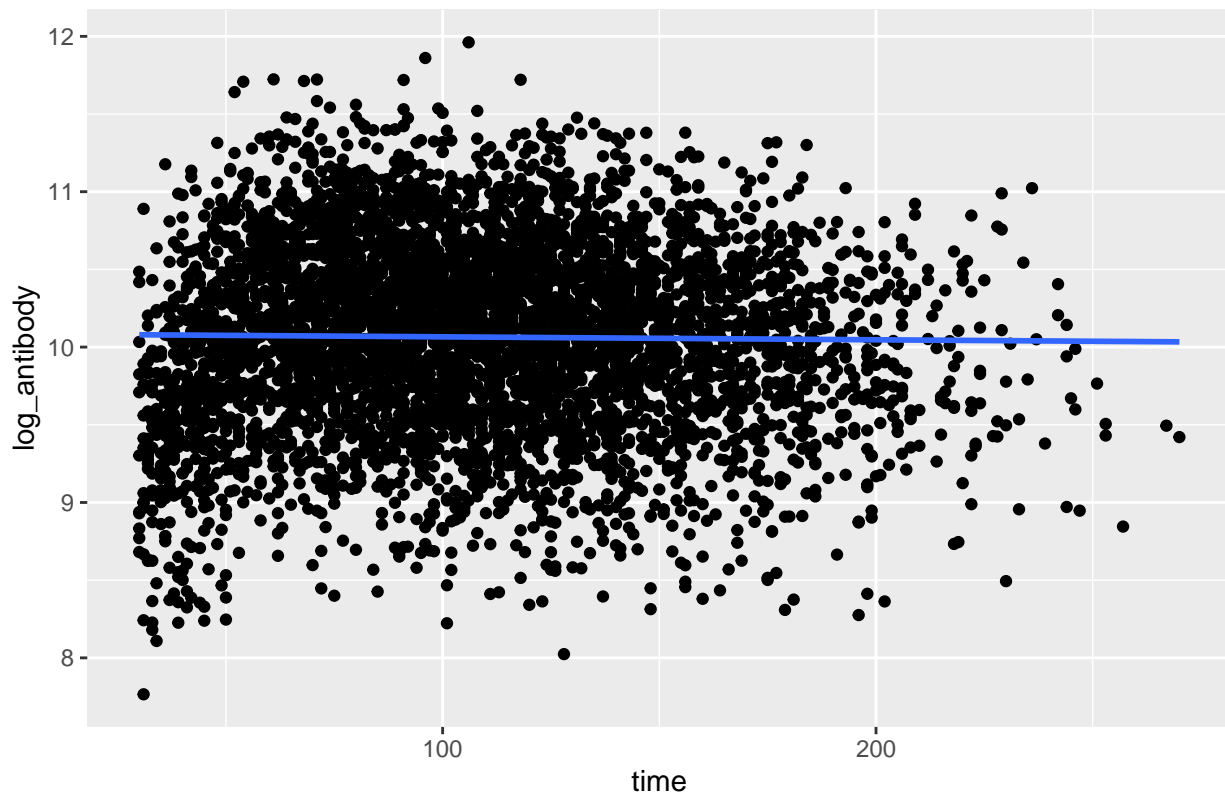
```
## [1] 0.568555
```

```
# Calculate R-squared for dat2
rsq_dat2 = 1 - sum((generalization - initial_test$log_antibody)^2) /
  sum((mean(initial_test$log_antibody) - initial_test$log_antibody)^2)
rsq_dat2
```

```
## [1] 0.06204078
```

```
ggplot(initial_training, aes(x = time, y = log_antibody)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  labs(title = "Log Antibody Levels Over Time Since Vaccination")
```

Log Antibody Levels Over Time Since Vaccination



GAM MODEL

```
#####GAM MODEL#####

#converted some of the variables to factor for smoother flow
training <- training %>% mutate(across(c(race, smoking, gender), as.factor))
validation <- validation %>% mutate(across(c(race, smoking, gender), as.factor))
dat2 <- dat2 %>% mutate(across(c(race, smoking, gender), as.factor))

## GAM Model Specification
gam_spec <- gen_additive_mod(
  select_features = FALSE,
  adjust_deg_free = NULL
) %>%
  set_mode("regression") %>%
  set_engine("mgcv", method = "REML")

## Fitting the GAM
gam_fit <- gam_spec %>%
  fit(log_antibody ~ s(age) + s(bmi) + s(time) + gender + race + smoking +
    diabetes + hypertension + s(LDL),
    data = training)

## Predictions compared to the validation set
```

```

gam_preds <- predict(gam_fit, new_data = validation) %>%
  bind_cols(validation)

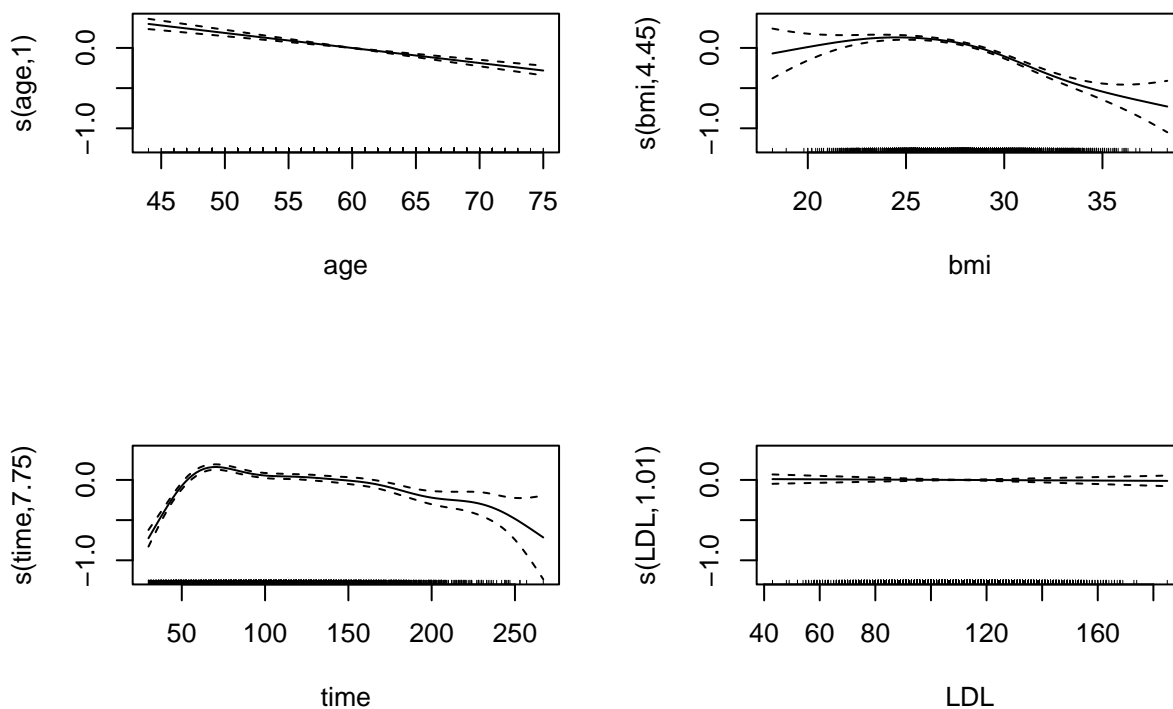
#rmse and rsq for performance eval
gam_rmse <- rmse(gam_preds, truth = log_antibody, estimate = .pred)
gam_rsqa <- rsq(gam_preds, truth = log_antibody, estimate = .pred)

#summary
summary(gam_fit$fit)

##
## Family: gaussian
## Link function: identity
##
## Formula:
## log_antibody ~ s(age) + s(bmi) + s(time) + gender + race + smoking +
##      diabetes + hypertension + s(LDL)
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  10.221959   0.017117 597.175  < 2e-16 ***
## gender1      -0.280876   0.016615 -16.905  < 2e-16 ***
## race2        -0.005040   0.037004  -0.136    0.892
## race3        -0.009887   0.020920  -0.473    0.637
## race4        -0.040831   0.028897  -1.413    0.158
## smoking1      0.023987   0.018552   1.293    0.196
## smoking2     -0.186128   0.028521  -6.526 7.6e-11 ***
## diabetes      0.002641   0.023321   0.113    0.910
## hypertension -0.007043   0.017971  -0.392    0.695
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##              edf Ref.df      F p-value
## s(age)       1.002  1.003 87.242 <2e-16 ***
## s(bmi)       4.447  5.476 64.860 <2e-16 ***
## s(time)     7.752  8.498 38.562 <2e-16 ***
## s(LDL)      1.011  1.021  0.128  0.738
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.215   Deviance explained = 21.9%
## -REML = 3141.8   Scale est. = 0.27444   n = 4000

#visual
plot(gam_fit$fit, pages = 1)

```



```
# Testing GAM model with the test data
gam_test_pred = predict(gam_fit, new_data = initial_test)$pred
# Calculating RMSE of the test data
gam_test_error = mean((gam_test_pred - initial_test$log_antibody)^2)
gam_test_error # 0.2879954
```

```
## [1] 0.287913
```

MARS MODEL

```
###RESOLVE THIS BEFORE WE SUBMIT###
## I did this in both of my parts --> if you guys are okay with it maybe we can do this in the data prep

training <- training %>% mutate(across(c(race, smoking, gender), as.factor))
validation <- validation %>% mutate(across(c(race, smoking, gender), as.factor))
dat2 <- dat2 %>% mutate(across(c(race, smoking, gender), as.factor))

## Cross-Validation Setup
set.seed(2222)
cv_folds <- vfold_cv(training, v = 10)

## MARS Model Specification
mars_spec <- mars(num_terms = tune(), prod_degree = tune()) %>%
```



```

set_engine("earth") %>%
set_mode("regression")

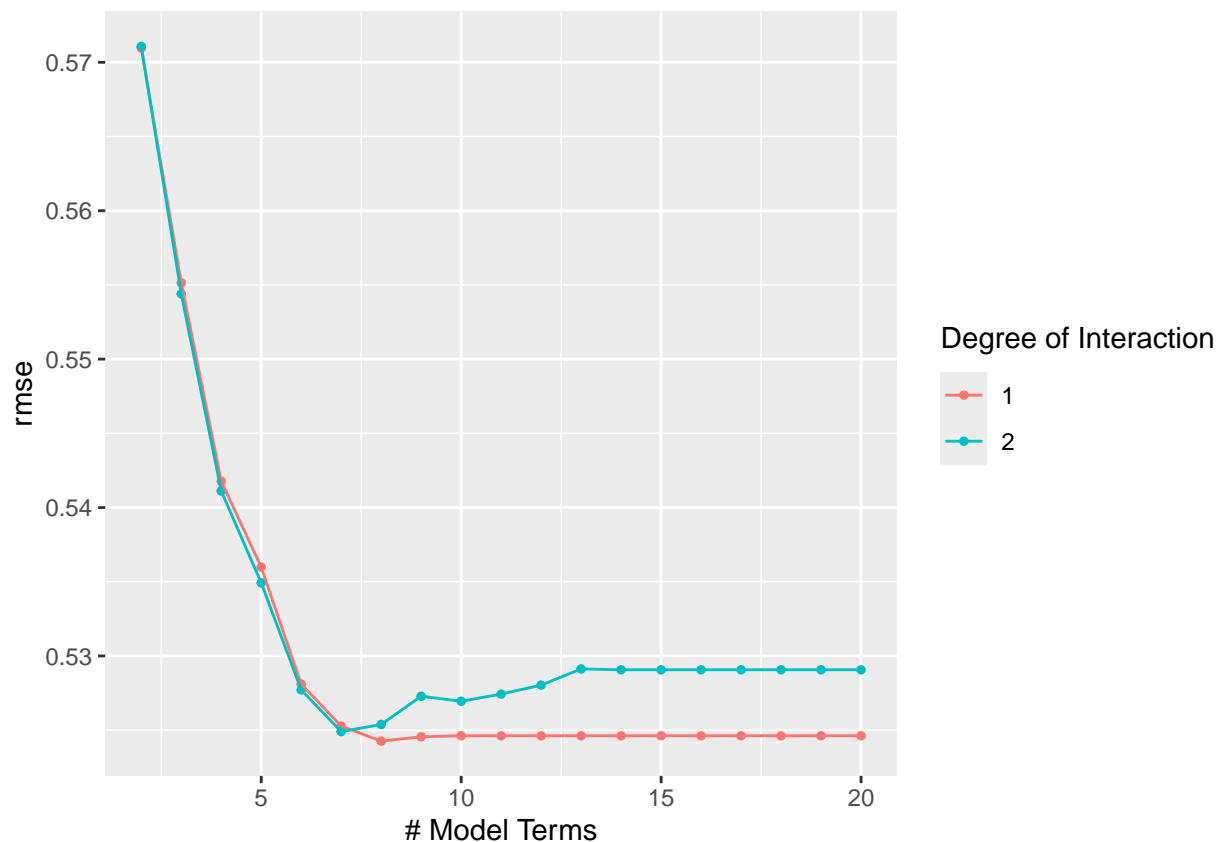
## Hyperparameter Grid
mars_grid_set <- parameters(num_terms(range = c(2, 20)), prod_degree(range = c(1, 2)))
mars_grid <- grid_regular(mars_grid_set, levels = c(20, 4))

## setting up the workflow
mars_workflow <- workflow() %>%
  add_model(mars_spec) %>%
  add_formula(log_antibody ~ age + gender + race + smoking +
              bmi + diabetes + hypertension + LDL + time)

## Hyperparameter Tuning
set.seed(2222)
mars_tune <- tune_grid(
  mars_workflow,
  resamples = cv_folds,
  grid = mars_grid
)

# Visualizing the tuning results
autoplot(mars_tune, metric = "rmse")

```



```

# best hyperparameters Based on RMSE
mars_best <- select_best(mars_tune, metric = "rmse")

## Updated model using best tune
final_mars_spec <- mars_spec %>%
  update(num_terms = mars_best$num_terms,
         prod_degree = mars_best$prod_degree)

##Final MARS
mars_fit <- fit(final_mars_spec,
               formula = log_antibody ~ age + gender + race + smoking +
                        bmi + diabetes + hypertension + LDL + time,
               data = training)

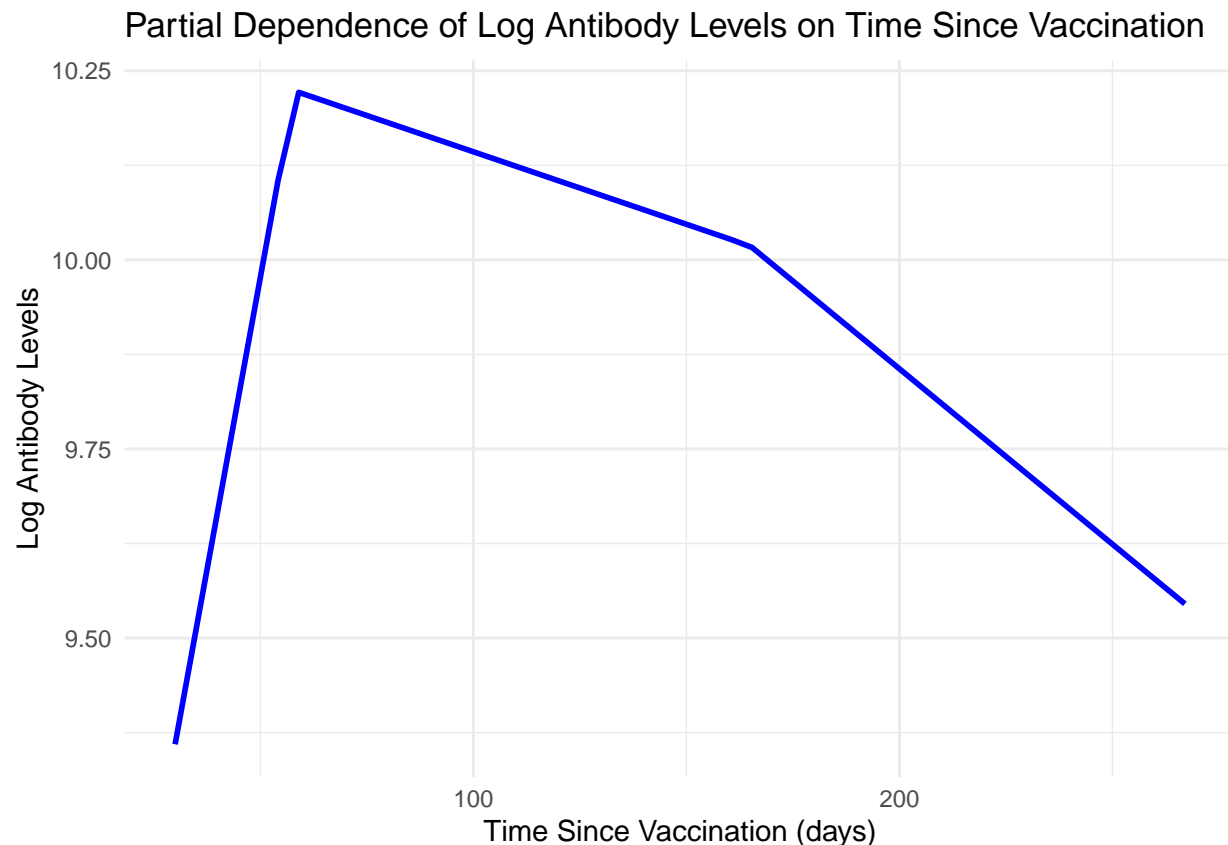
## Extracting fitted MARS and Coefficients
mars_model <- extract_fit_engine(mars_fit)
coef(mars_model)

## (Intercept) h(bmi-27.8) h(time-58) h(58-time) gender1 h(70-age)
## 10.288401336 -0.094250784 -0.001917226 -0.030854679 -0.280399107 0.018948596
## smoking2 h(time-165)
## -0.200881611 -0.002721176

## Partial Dependence Plot for 'time'
pdp_plot <- partial(mars_fit, pred.var = "time", grid.resolution = 50, train = training)
ggplot(pdp_plot, aes(x = time, y = yhat)) +
  geom_line(color = "blue", size = 1) +
  labs(title = "Partial Dependence of Log Antibody Levels on Time Since Vaccination",
       x = "Time Since Vaccination (days)",
       y = "Log Antibody Levels") +
  theme_minimal()

## Warning: Using 'size' aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use 'linewidth' instead.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.

```



```
# Testing MARS model with the test data
```

```
test.pred = predict(mars_fit, new_data = initial_test)
```

```
## Warning in model.frame.default(terms.without.response, data = data, na.action =
```

```
## na.pass, : variable 'gender' is not a factor
```

```
## Error : variable 'gender' was fitted with type "factor" but type "numeric" was supplied
```

```
## Continuing anyway, first few rows of modelframe are
```

```
##      age gender race smoking  bmi diabetes hypertension LDL time
## 5001  58      0   4      1 27.7        0              0 115 205
## 5002  62      0   1      1 29.4        1              0 118 229
## 5003  71      0   4      0 24.6        1              1 149 206
## 5004  59      1   1      0 28.0        0              0 119 163
## 5005  69      1   1      0 27.0        1              1 142 240
## 5006  56      0   1      0 26.6        0              0 112 206
```

```
# Calculating RMSE of the test data
```

```
rmse = sqrt(mean((test.pred$.pred - initial_test$log_antibody)^2))
```

```
rmse # 0.5276064
```

```
## [1] 0.5276064
```