P8106 Midterm - Code

Kate Colvin (KAC2301), Jeong Yun (Lizy) Choi (JC6452), and Flora Pang (FP2513)

Exploratory Analysis

Loading in Data

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

dat1 <- dat1 %>% janitor::clean_names()
dat2 <- dat2 %>%janitor::clean_names()
```

Producing Summary Table

Training and test data have the same distribution of demographic characteristics; there is a difference in time since vaccination and log-transformed antibody levels between training and test data

```
# Combining data for summary table, data cleaning
dat1_com <- dat1 %>% mutate(set = "Training Data")
dat2_com <- dat2 %>% mutate(set = "Testing Data")

dat <- dat1_com %>%
  rbind(dat2_com) %>%
  rename(days_vaccinated = time) %>%
  mutate(race = as.character(race), smoking = as.character(smoking)) %>%
  mutate(race = case_match(
        race, "1" ~ "White", "2" ~ "Asian", "3" ~ "Black", "4" ~ "Hispanic"),
         gender = case_match(gender, 1 ~ "Male", 0 ~ "Female"),
         smoking = case_match(
           smoking, "0" ~ "Never", "1" ~ "Former", "2" ~ "Current"))

# Summary table
dat %>% select(!id) %>%
  tbl_summary(
    by = set,
    label = list(age = "Age", gender = "Gender", race = "Race", smoking = "Smoking",
                 height = "Height (cm)", weight = "Weight (kg)", bmi = "BMI",
                 diabetes = "Diabetes", hypertension = "Hypertension",
                 sbp = "Systolic Blood Pressure (mmHg)", ldl = "LDL Cholesterol (mg/dL)",
                 days_vaccinated = "Time Since Vaccinated (days)",
                 log_antibody = "Log-Transformed Antibody Level")) %>%
  add_overall() %>% add_p() %>%
  modify_caption("Summary of Patient Testing and Training Data (N=6000)") %>%
  as_gt() %>% tab_options(table.font.size = 10)
```

```
## The following errors were returned during `as_gt()`:
```

Table 1: Summary of Patient Testing and Training Data (N=6000)

Characteristic	**Overall** N = 6,000[1]	**Testing Data** N = 1,000[1]	**Training Data** N = 5,000[1]	**p-value**
Age	60.0 (57.0, 63.0)	60.0 (57.0, 63.0)	60.0 (57.0, 63.0)	
Gender				
Female	3,082 (51%)	509 (51%)	2,573 (51%)	
Male	2,918 (49%)	491 (49%)	2,427 (49%)	
Race				
Asian	333 (5.6%)	55 (5.5%)	278 (5.6%)	
Black	1,235 (21%)	199 (20%)	1,036 (21%)	
Hispanic	548 (9.1%)	83 (8.3%)	465 (9.3%)	
White	3,884 (65%)	663 (66%)	3,221 (64%)	
Smoking				
Current	589 (9.8%)	103 (10%)	486 (9.7%)	
Former	1,800 (30%)	296 (30%)	1,504 (30%)	
Never	3,611 (60%)	601 (60%)	3,010 (60%)	
Height (cm)	170.1 (166.1, 174.2)	170.2 (166.1, 174.2)	170.1 (166.1, 174.3)	
Weight (kg)	80 (75, 85)	80 (75, 84)	80 (75, 85)	
BMI	27.60 (25.80, 29.50)	27.60 (25.80, 29.60)	27.60 (25.80, 29.50)	
Diabetes	929 (15%)	157 (16%)	772 (15%)	
Hypertension	2,754 (46%)	456 (46%)	2,298 (46%)	
Systolic Blood Pressure (mmHg)	130 (124, 135)	130 (124, 135)	130 (124, 135)	
LDL Cholesterol (mg/dL)	110 (96, 124)	112 (96, 124)	110 (96, 124)	
Time Since Vaccinated (days)	116 (82, 152)	171 (140, 205)	106 (76, 138)	
Log-Transformed Antibody Level	10.06 (9.65, 10.45)	9.93 (9.50, 10.32)	10.09 (9.68, 10.48)	

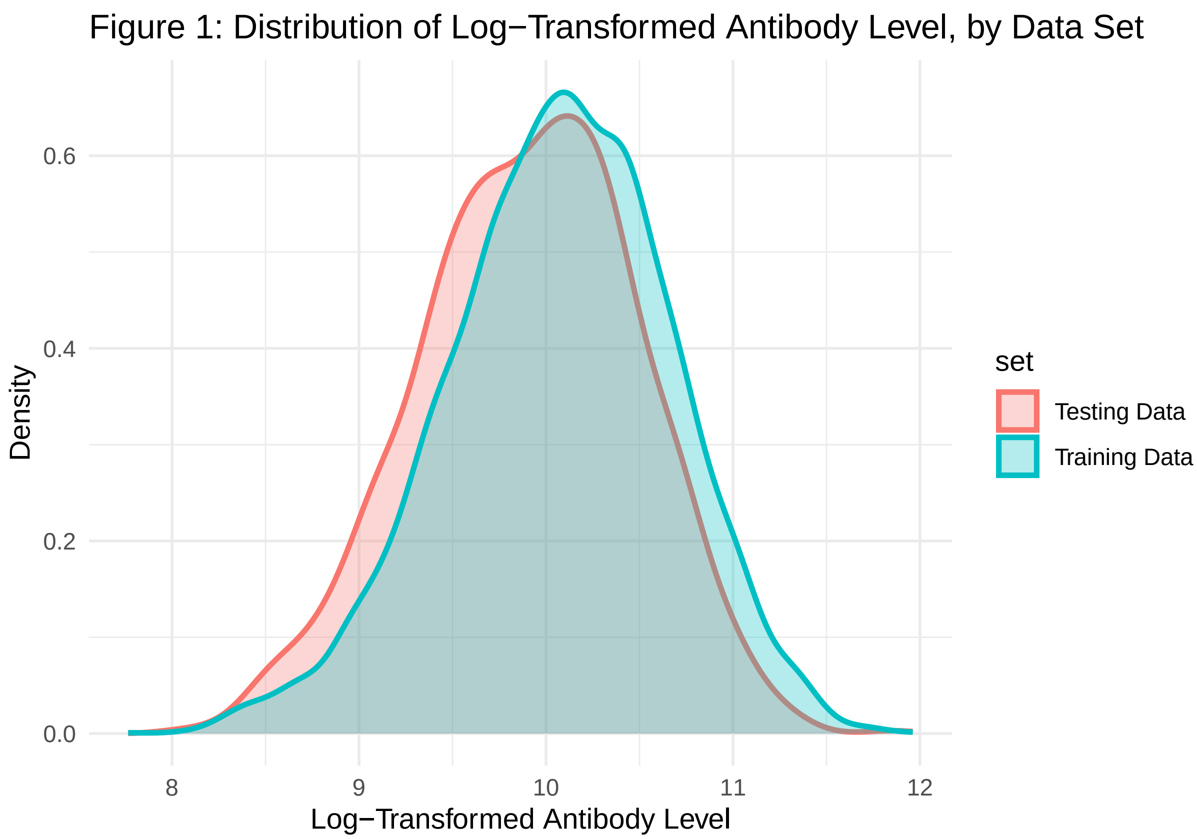
[1] Median (Q1, Q3); n (%)

```
## x For variable `age` (`set`) and "p.value" statistic: The package "cardx" (>=
##   0.2.3) is required.
## x For variable `bmi` (`set`) and "p.value" statistic: The package "cardx" (>=
##   0.2.3) is required.
## x For variable `days_vaccinated` (`set`) and "p.value" statistic: The package
##   "cardx" (>= 0.2.3) is required.
## x For variable `diabetes` (`set`) and "p.value" statistic: The package "cardx"
##   (>= 0.2.3) is required.
## x For variable `gender` (`set`) and "p.value" statistic: The package "cardx"
##   (>= 0.2.3) is required.
## x For variable `height` (`set`) and "p.value" statistic: The package "cardx"
##   (>= 0.2.3) is required.
## x For variable `hypertension` (`set`) and "p.value" statistic: The package
##   "cardx" (>= 0.2.3) is required.
## x For variable `ldl` (`set`) and "p.value" statistic: The package "cardx" (>=
##   0.2.3) is required.
## x For variable `log_antibody` (`set`) and "p.value" statistic: The package
##   "cardx" (>= 0.2.3) is required.
## x For variable `race` (`set`) and "p.value" statistic: The package "cardx" (>=
##   0.2.3) is required.
## x For variable `sbp` (`set`) and "p.value" statistic: The package "cardx" (>=
##   0.2.3) is required.
## x For variable `smoking` (`set`) and "p.value" statistic: The package "cardx"
##   (>= 0.2.3) is required.
## x For variable `weight` (`set`) and "p.value" statistic: The package "cardx"
##   (>= 0.2.3) is required.
```

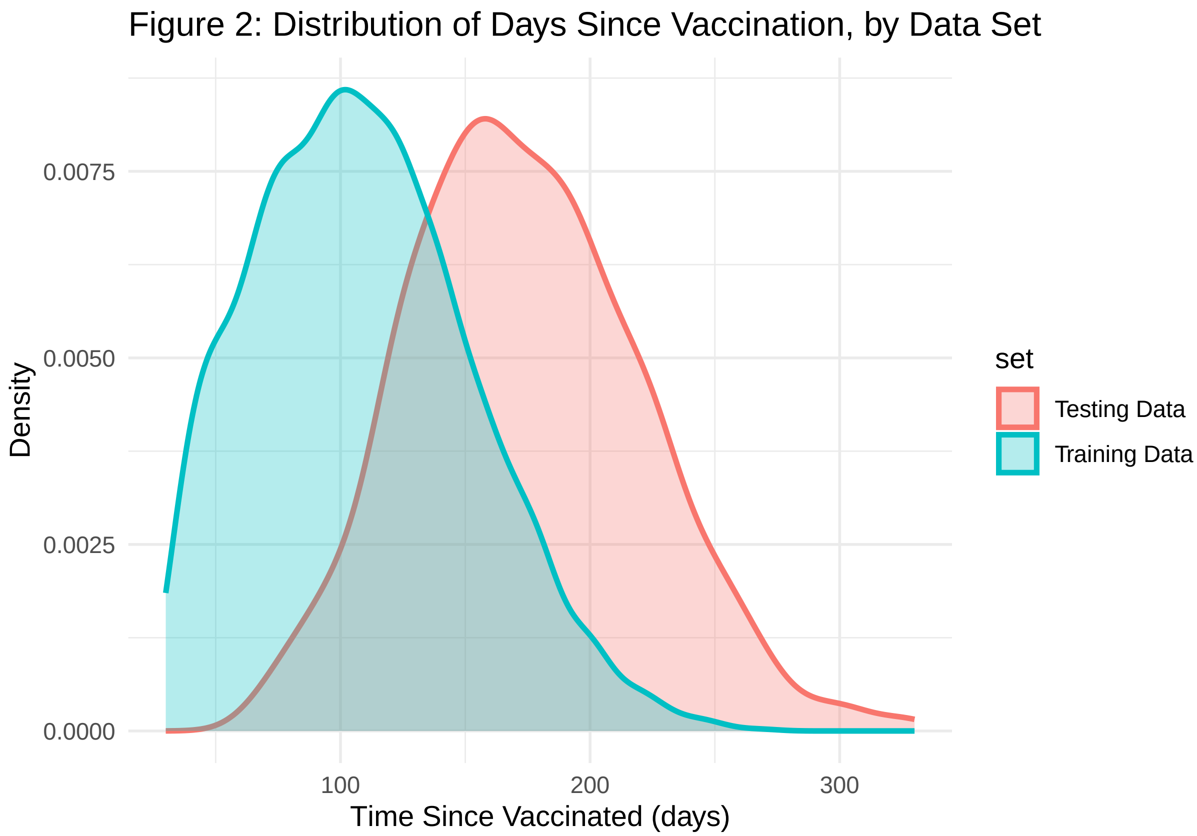
Histograms of Differing Variables by Training and Test Set

```r
# Antibody level
plot_sets <- dat %>%
```

```r
  ggplot(aes(x = log_antibody,
             fill = set,
             color = set)) +
  geom_density(alpha = 0.3, linewidth = 1) +
  labs(x = "Log-Transformed Antibody Level",
       y = "Density",
       title = "Figure 1: Distribution of Log-Transformed Antibody Level, by Data Set") +
  theme_minimal()

# Time since vaccination (days)
plot_days <- dat %>%
  ggplot(aes(x = days_vaccinated,
             fill = set,
             color = set)) +
  geom_density(alpha = 0.3, linewidth = 1) +
  labs(x = "Time Since Vaccinated (days)",
       y = "Density",
       title = "Figure 2: Distribution of Days Since Vaccination, by Data Set") +
  theme_minimal()

plot_sets
```

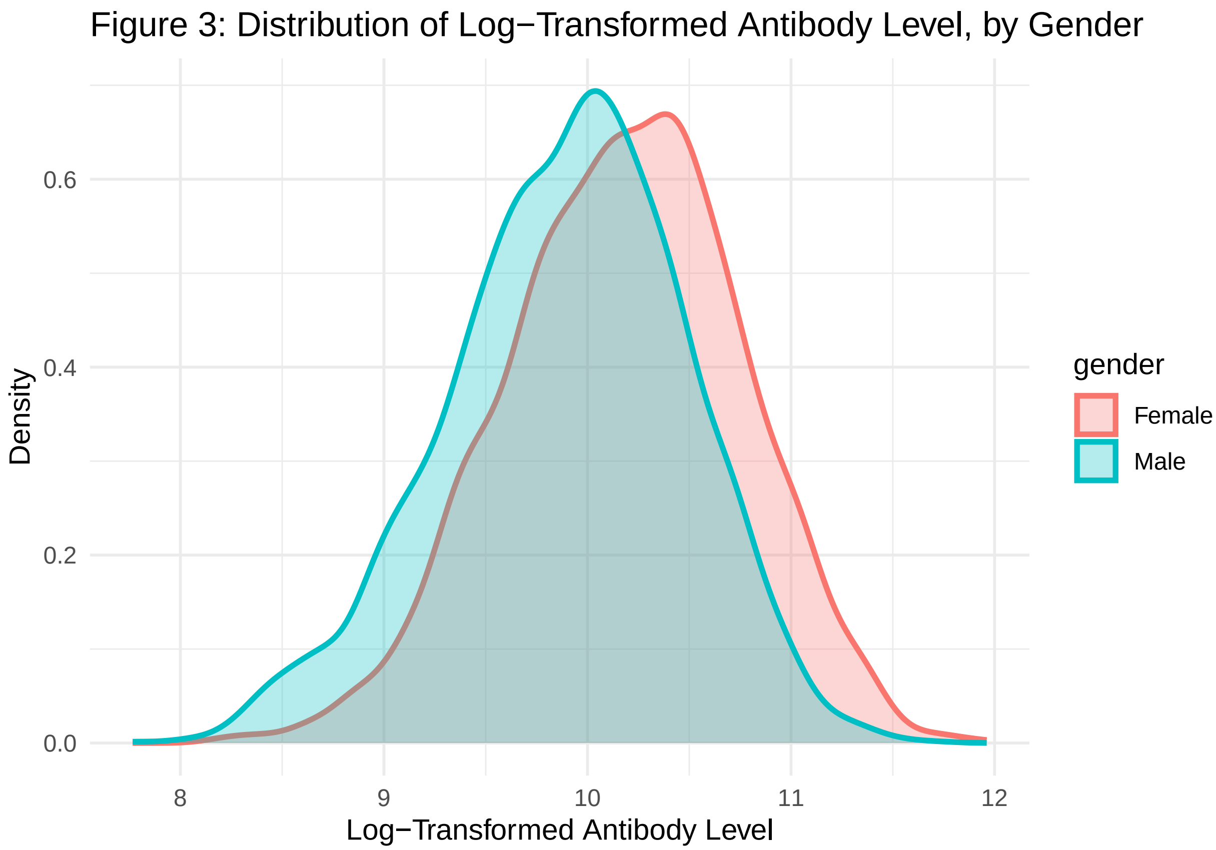



```r
plot_days
```



Plots of Log-Transformed Antibody Level, by Categorical Variables

```r
# Antibody level, by gender
plot_gender <- dat %>%
  ggplot(aes(x = log_antibody, fill = gender, color = gender)) +
  geom_density(alpha = 0.3, linewidth = 1) +
  labs(x = "Log-Transformed Antibody Level", y = "Density",
       title = "Figure 3: Distribution of Log-Transformed Antibody Level, by Gender") +
  theme_minimal()
plot_gender
```




```
strip_markdown <- function(x) {gsub("\\*\\*", "", x)}

dat %>% select(gender, log_antibody) %>%
  tbl_summary(by = gender) %>% add_p() %>%
  modify_caption("Log-Transformed Antibody Level, by Gender") %>%
  as_kable() %>%
  footnote(general_title = "", general = "Median (Q1, Q3), Wilcoxon Rank Sum Test") %>%
  strip_markdown()
```

```
## The following errors were returned during `as_kable()`:
## x For variable `log_antibody` (`gender`) and "p.value" statistic: The package
##   "cardx" (>= 0.2.3) is required.
```

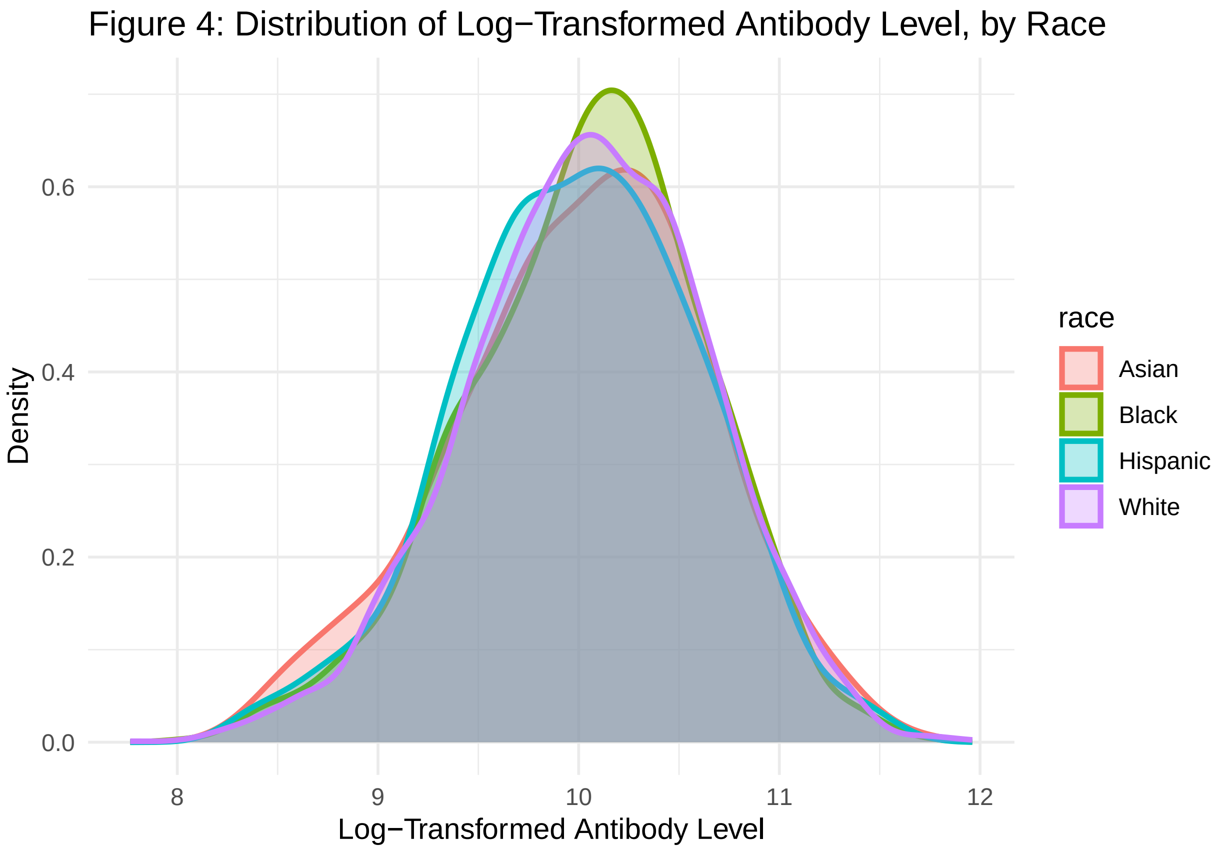
Table 2: Log-Transformed Antibody Level, by Gender

Characteristic	Female N = 3,082	Male N = 2,918	p-value
log_antibody	10.20 (9.79, 10.58)	9.93 (9.51, 10.30)	

Median (Q1, Q3), Wilcoxon Rank Sum Test

```
# Antibody level, by race
plot_race <- dat %>%
  ggplot(aes(x = log_antibody, fill = race, color = race)) +
  geom_density(alpha = 0.3, linewidth = 1) +
  labs(x = "Log-Transformed Antibody Level",
       y = "Density",
       title = "Figure 4: Distribution of Log-Transformed Antibody Level, by Race") +
```

```
  theme_minimal()

plot_race
```




```
dat %>%
  select(race, log_antibody) %>%
  tbl_summary(by = race) %>%
    add_p() %>%
  modify_caption("Log-Transformed Antibody Level, by Race") %>%
  as_kable() %>%
  footnote(general_title = "",
           general = "Median (Q1, Q3), Kruskal-Wallis Rank Sum Test") %>%
  strip_markdown()
```

```
## The following errors were returned during `as_kable()`:
## x For variable `log_antibody` (`race`) and "p.value" statistic: The package
##   "cardx" (>= 0.2.3) is required.
```

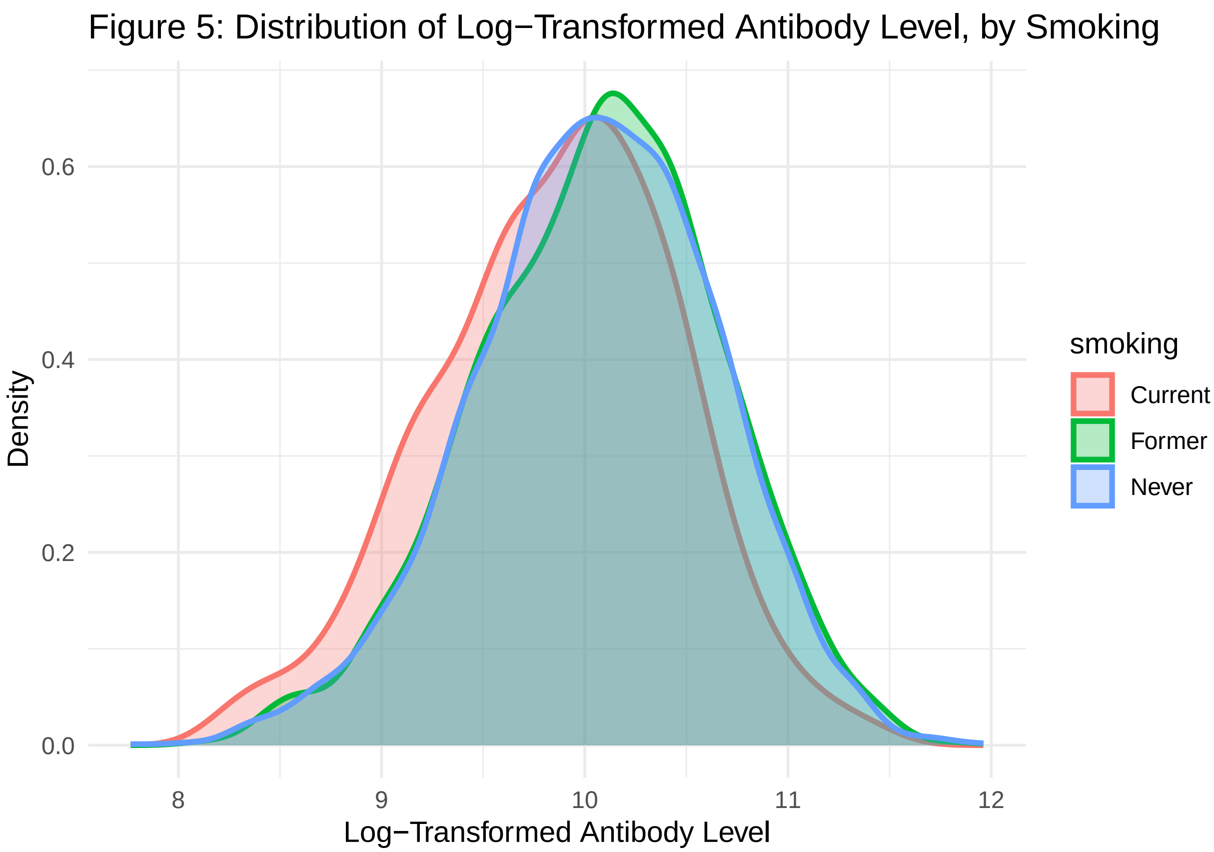
Table 3: Log-Transformed Antibody Level, by Race

Characteristic	Asian N = 333	Black N = 1,235	Hispanic N = 548	White N = 3,884	p-value
log_antibody	10.06 (9.62, 10.44)	10.08 (9.65, 10.44)	10.03 (9.61, 10.42)	10.06 (9.65, 10.46)	

Median (Q1, Q3), Kruskal-Wallis Rank Sum Test

```
# Antibody level, by smoking status
plot_smoking <- dat %>%
```

```r
  ggplot(aes(x = log_antibody, fill = smoking, color = smoking)) +
  geom_density(alpha = 0.3, linewidth = 1) +
  labs(x = "Log-Transformed Antibody Level",
       y = "Density",
       title = "Figure 5: Distribution of Log-Transformed Antibody Level, by Smoking") +
  theme_minimal()
plot_smoking
```




```r
dat %>% select(smoking, log_antibody) %>%
  tbl_summary(by = smoking) %>%
    add_p() %>%
  modify_caption("Log-Transformed Antibody Level, by Smoking Status") %>%
  as_kable() %>%
  footnote(general_title = "",
           general = "Median (Q1, Q3), Kruskal-Wallis Rank Sum Test") %>%
  strip_markdown()
```

```
## The following errors were returned during `as_kable()`:
## x For variable `log_antibody` (`smoking`) and "p.value" statistic: The package
##   "cardx" (>= 0.2.3) is required.
```

Table 4: Log-Transformed Antibody Level, by Smoking Status

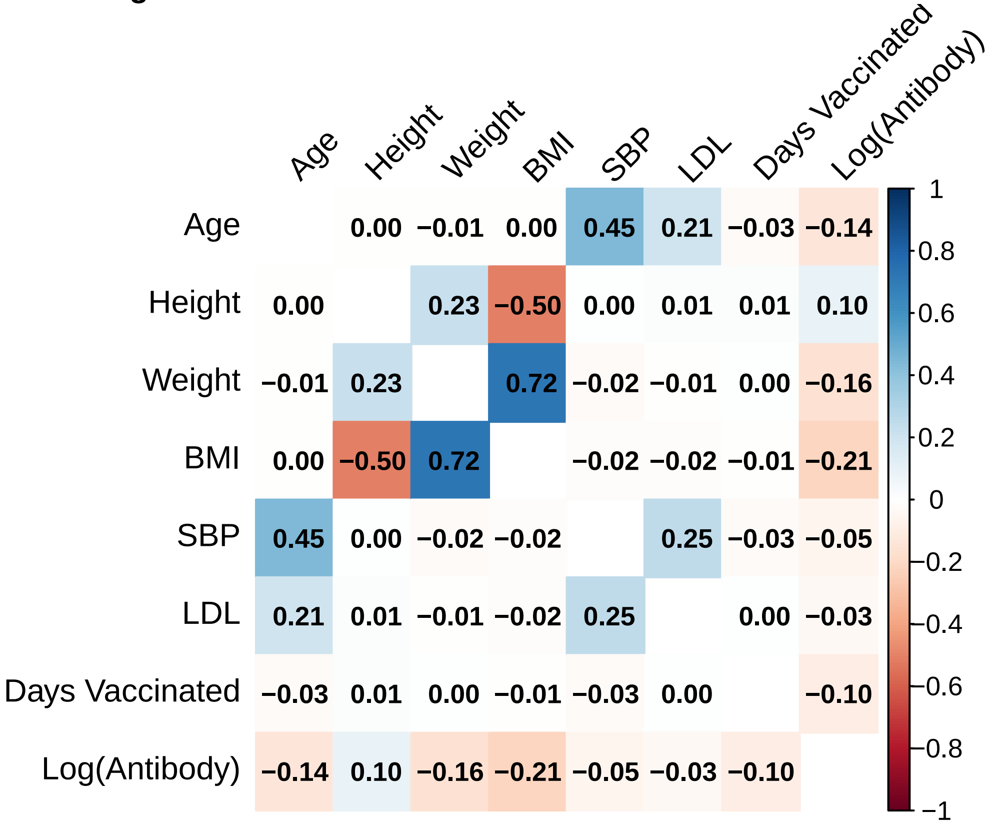
Characteristic	Current N = 589	Former N = 1,800	Never N = 3,611	p-value
log_antibody	9.91 (9.46, 10.28)	10.10 (9.66, 10.48)	10.07 (9.68, 10.46)	

Median (Q1, Q3), Kruskal-Wallis Rank Sum Test

Correlation Matrix of Numerical Variables

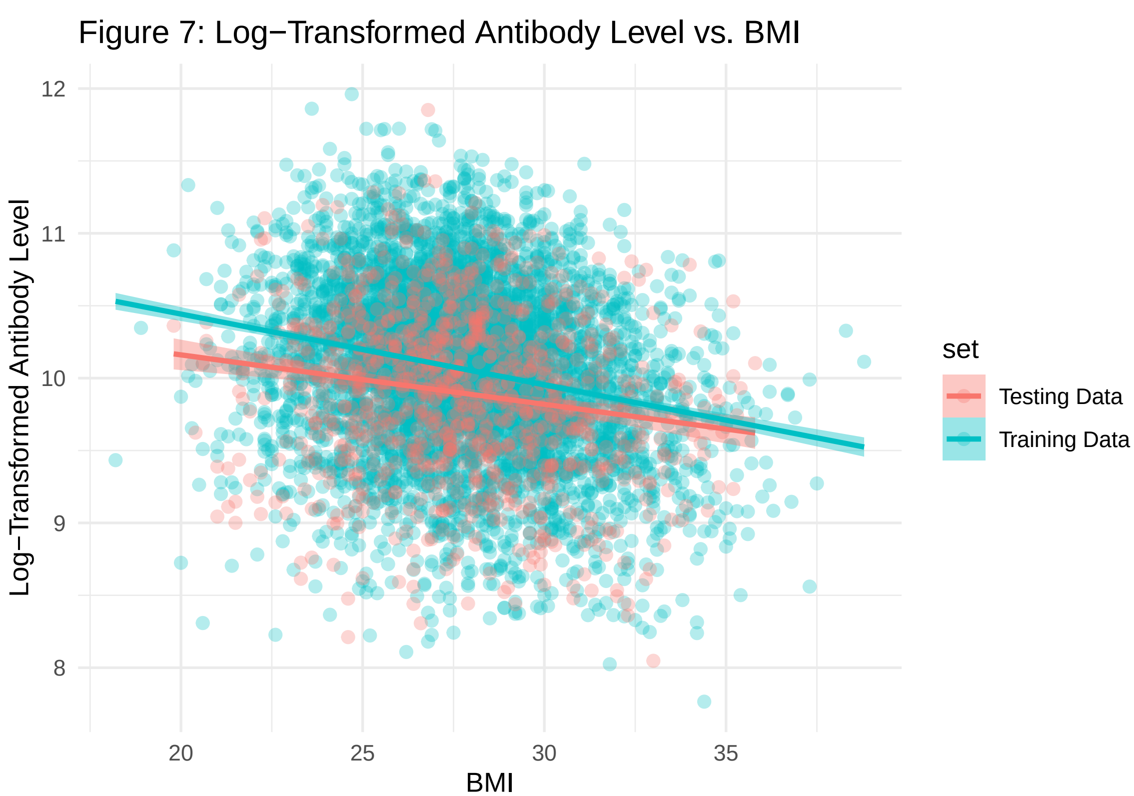
```r
cor_matrix <- dat %>%
  select(age, height, weight, bmi, sbp, ldl, days_vaccinated, log_antibody) %>%
  rename("Age" = age,
         "Height" = height,
         "Weight" = weight,
         "BMI" = bmi,
         "SBP" = sbp,
         "LDL" = ldl,
         "Days Vaccinated" = days_vaccinated,
         "Log(Antibody)" = log_antibody) %>%
  cor()

cor_plot <- corrplot(cor_matrix,
                     main = "Figure 6: Correlation Matrix of Numerical Variables",
                     mar=c(0,0,1,0), cex.main = 1,
                     method = "color",
                     addCoef.col = "black",
                     tl.col = "black",
                     number.cex = 0.8,
                     tl.srt = 45,
                     order = 'original',
                     diag = F)
```

Figure 6: Correlation Matrix of Numerical Variables



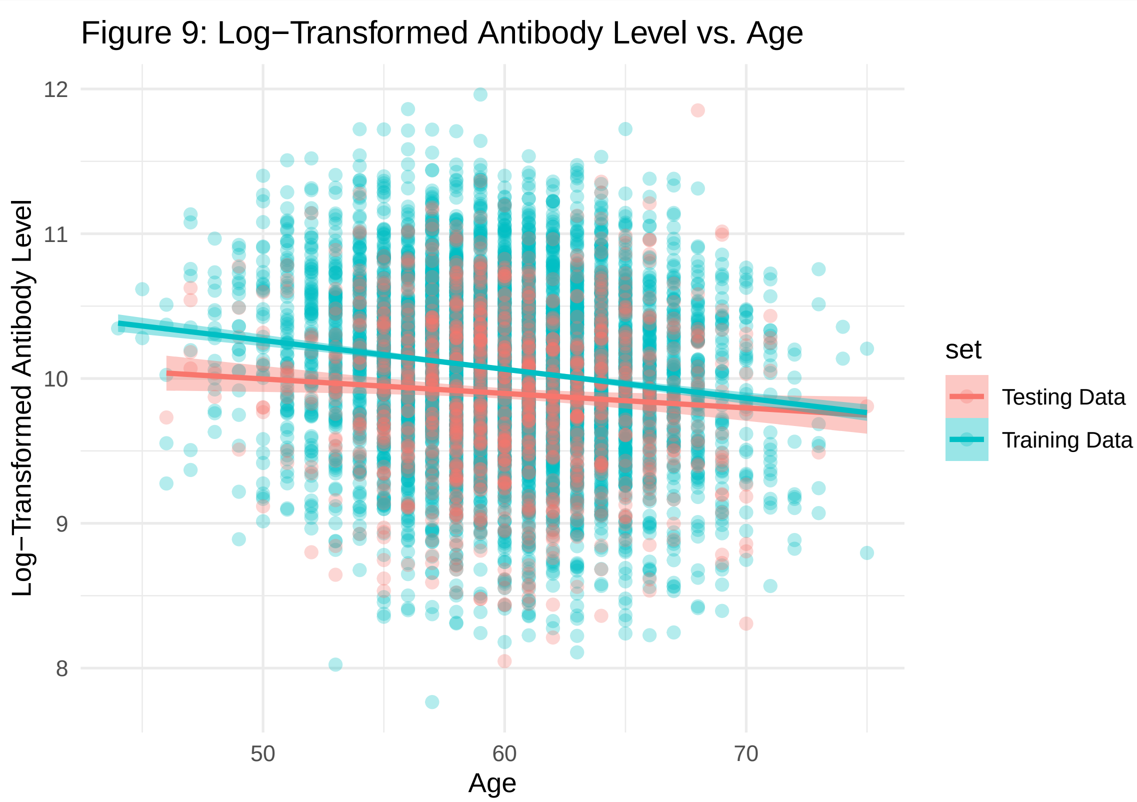
Plots of Log-Transformed Antibody Level vs. Selected Numerical Variables

```
# Antibody level vs. BMI
plot_bmi <- dat %>% ggplot(aes(x = bmi, y = log_antibody, fill = set, color = set)) +
  geom_point(alpha = 0.3, size = 2) +
  geom_smooth(method = "lm") +
  labs(y = "Log-Transformed Antibody Level", x = "BMI",
       title = "Figure 7: Log-Transformed Antibody Level vs. BMI") +
  theme_minimal()

# Antibody level vs. Weight
plot_weight <- dat %>%
  ggplot(aes(x = weight, y = log_antibody, fill = set, color = set)) +
  geom_point(alpha = 0.3, size = 2) +
  geom_smooth(method = "lm") +
  labs(y = "Log-Transformed Antibody Level", x = "Weight",
       title = "Figure 8: Log-Transformed Antibody Level vs. Weight") +
  theme_minimal()

# Antibody level vs. Age
plot_age <- dat %>% ggplot(aes(x = age, y = log_antibody, fill = set, color = set)) +
  geom_point(alpha = 0.3, size = 2) + geom_smooth(method = "lm") +
  labs(y = "Log-Transformed Antibody Level", x = "Age",
       title = "Figure 9: Log-Transformed Antibody Level vs. Age") +
  theme_minimal()

plot_bmi
```



```
plot_weight
```


```
plot_age
```



Model Selection and Training

Since the response variable (log_antibody) is continuous, this project will consider the following models:

- Multiple Linear Regression (MLR) - as a baseline.
- LASSO Regression – to improve predictive performance by selecting important predictors.
- MARS model – allow remain in regression but also capture nonlinear effects

After comparing model performance, the best model will be based on cross-validation results.

Data Pre-processing

```
# Converting categorical variables into factors
dat1 <- dat1 %>%
  mutate(
    gender = factor(gender, levels = c(0, 1), labels = c("Female", "Male")),
    race = factor(race, levels = c(1, 2, 3, 4), labels = c("White", "Asian", "Black", "Hispanic")),
    smoking = factor(smoking, levels = c(0, 1, 2), labels = c("Never", "Former", "Current")),
    diabetes = factor(diabetes),
    hypertension = factor(hypertension)
  )

dat2 <- dat2 %>%
  mutate(
    gender = factor(gender, levels = c(0, 1), labels = c("Female", "Male")),
    race = factor(race, levels = c(1, 2, 3, 4), labels = c("White", "Asian", "Black", "Hispanic")),
    smoking = factor(smoking, levels = c(0, 1, 2), labels = c("Never", "Former", "Current")),
    diabetes = factor(diabetes),
    hypertension = factor(hypertension)
  )
```

```
sum(is.na(dat1))
```

```
## [1] 0
```

```
sum(is.na(dat2))
```

```
## [1] 0
```

```
dat1 <- dat1 %>% select(-id)
dat2 <- dat2 %>% select(-id)
```

```
# Split training data into training (80%) and validation (20%)
set.seed(123)
train_index <- createDataPartition(dat1$log_antibody, p = 0.8, list = FALSE)
train_data <- dat1[train_index, ]
valid_data <- dat1[-train_index, ]
```

Training multiple linear regression model

```
mlr_model <- lm(log_antibody ~ ., data = train_data)
summary(mlr_model)
```

```
##
## Call:
## lm(formula = log_antibody ~ ., data = train_data)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.13316 -0.35330  0.02809  0.38037  1.64371
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    26.2267745  2.5711711  10.200  < 2e-16 ***
## age            -0.0214288  0.0021720  -9.866  < 2e-16 ***
## genderMale     -0.2917270  0.0174904 -16.679  < 2e-16 ***
## raceAsian      -0.0217822  0.0392909  -0.554   0.5793
## raceBlack      -0.0014558  0.0219999  -0.066   0.9472
## raceHispanic   -0.0563862  0.0302636  -1.863   0.0625 .
## smokingFormer   0.0183932  0.0194704   0.945   0.3449
## smokingCurrent -0.1821800  0.0307072  -5.933 3.23e-09 ***
## height         -0.0789900  0.0150568  -5.246 1.63e-07 ***
## weight          0.0831140  0.0159211   5.220 1.88e-07 ***
## bmi            -0.2869272  0.0458538  -6.257 4.33e-10 ***
## diabetes1      -0.0009792  0.0239012  -0.041   0.9673
## hypertension1  -0.0030485  0.0292595  -0.104   0.9170
## sbp             0.0006393  0.0019277   0.332   0.7402
## ldl            -0.0002973  0.0004484  -0.663   0.5073
## time           -0.0002329  0.0001997  -1.166   0.2437
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.552 on 3984 degrees of freedom
## Multiple R-squared:  0.1482, Adjusted R-squared:  0.145
## F-statistic:  46.2 on 15 and 3984 DF,  p-value: < 2.2e-16
```

```
# Evaluating model performance on validation data
mlr_pred <- predict(mlr_model, newdata = valid_data)
mlr_rmse <- sqrt(mean((mlr_pred - valid_data$log_antibody)^2))
mlr_rmse
```

```
## [1] 0.5443976
```

Training LASSO regression model

Standardizing numerical variables for LASSO

```
num_vars <- c("age", "height", "weight", "bmi", "sbp", "ldl", "time")

preprocess_params <- preProcess(train_data[, num_vars], method = c("center", "scale"))
train_data[, num_vars] <- predict(preprocess_params, train_data[, num_vars])
valid_data[, num_vars] <- predict(preprocess_params, valid_data[, num_vars])
dat2[, num_vars] <- predict(preprocess_params, dat2[, num_vars]) # Applying same transformation to test
```

```
# Preparing the data matrices for glmnet
x_train <- model.matrix(log_antibody ~ ., train_data)[, -1]  # Removing the intercept
y_train <- train_data$log_antibody

x_valid <- model.matrix(log_antibody ~ ., valid_data)[, -1]
y_valid <- valid_data$log_antibody
```

```r
set.seed(123)
lasso_model <- cv.glmnet(x_train, y_train, alpha = 1) # LASSO with cross validation

best_lambda <- lasso_model$lambda.min
lasso_final <- glmnet(x_train, y_train, alpha = 1, lambda = best_lambda) # final model is based on opti
```

```r
# predicting with LASSO on validation data

lasso_pred <- predict(lasso_final, newx = x_valid)
lasso_rmse <- sqrt(mean((lasso_pred - y_valid)^2))
lasso_rmse
```

```
## [1] 0.5444827
```

Training MARS model

```r
mars_model <- earth(log_antibody ~ ., data = train_data)
summary(mars_model)
```

```
## Call: earth(formula=log_antibody~., data=train_data)
##
##                     coefficients
## (Intercept)           10.6736281
## genderMale            -0.2934673
## smokingCurrent        -0.1914705
## h(2.23884-age)         0.0949434
## h(bmi- -1.53409)      -0.2544287
## h(0.0941101-bmi)      -0.2077791
## h(bmi-2.73542)         0.5673150
## h(-1.17191-time)      -1.4869949
## h(time- -1.17191)     -0.0977731
##
## Selected 9 of 13 terms, and 5 of 15 predictors
## Termination condition: RSq changed by less than 0.001 at 13 terms
## Importance: genderMale, bmi, time, age, smokingCurrent, raceAsian-unused, ...
## Number of terms at each degree of interaction: 1 8 (additive model)
## GCV 0.2785185    RSS 1104.624    GRSq 0.2187119    RSq 0.2249513
```

```r
# predicting with MARS on validation data

mars_pred <- predict(mars_model, newdata = valid_data)
mars_rmse <- sqrt(mean((mars_pred - valid_data$log_antibody)^2))
mars_rmse
```

```
## [1] 0.5286262
```

- The MARS model achieves the lowest RMSE. Therefore MARS will be used as the preferred model for predicting log_antibody. Although further fine-tuning and additional feature exploration could further enhance the model's predictive power.

MARS model tuning

```r
tune_grid <- expand.grid(degree = 1:3, nprune = seq(5, 50, by = 5))

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

```r
mars_tune <- train(log_antibody ~ ., data = train_data, method = "earth",
                   trControl = train_control, tuneGrid = tune_grid)

mars_tune$bestTune
```

```
##   nprune degree
## 2     10      1
```

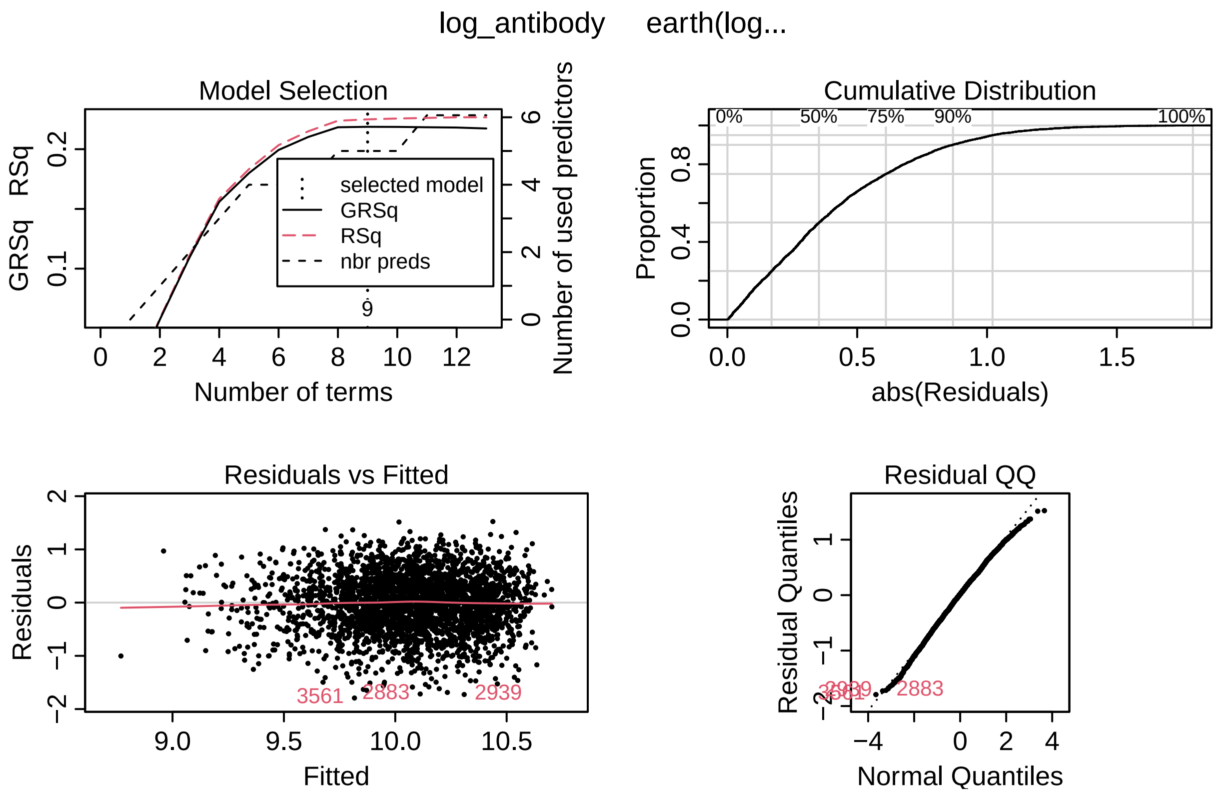
```r
train_control <- trainControl(method = "cv", number = 10)

# Train the MARS model with best tuning parameters
mars_model_tune <- train(log_antibody ~ .,
                         data = train_data,
                         method = "earth",
                         trControl = train_control,
                         tuneGrid = data.frame(nprune = 10, degree = 1))

summary(mars_model_tune)
```

```
## Call: earth(x=matrix[4000,15], y=c(10.65,9.889,1...), keepxy=TRUE, degree=1,
##             nprune=10)
##
##                   coefficients
## (Intercept)         10.6736281
## genderMale          -0.2934673
## smokingCurrent      -0.1914705
## h(2.23884-age)       0.0949434
## h(bmi- -1.53409)    -0.2544287
## h(0.0941101-bmi)    -0.2077791
## h(bmi-2.73542)       0.5673150
## h(-1.17191-time)    -1.4869949
## h(time- -1.17191)   -0.0977731
##
## Selected 9 of 13 terms, and 5 of 15 predictors (nprune=10)
## Termination condition: RSq changed by less than 0.001 at 13 terms
## Importance: genderMale, bmi, time, age, smokingCurrent, raceAsian-unused, ...
## Number of terms at each degree of interaction: 1 8 (additive model)
## GCV 0.2785185    RSS 1104.624    GRSq 0.2187119    RSq 0.2249513
```

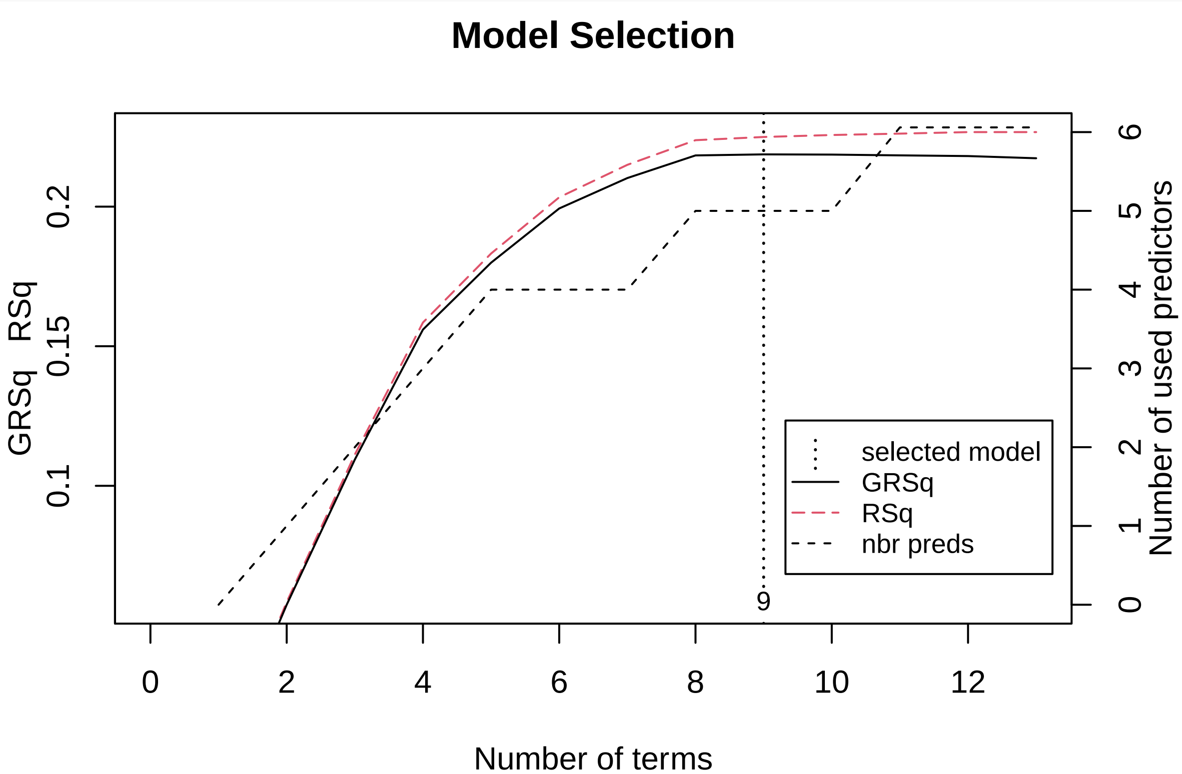
```r
mars_tune_pred <- predict(mars_model_tune, newdata = valid_data)
mars_tune_rmse <- sqrt(mean((mars_tune_pred - valid_data$log_antibody)^2))
mars_tune_rmse
```

```
## [1] 0.5286262
```

```r
plot(mars_model)
```




```r
plot(mars_model, which = 1)
```



Results