# Data Science II Midterm

Megan Panier, Shiying Wu, and Rita Wang

2025-03-25

## Libraries

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

## Importing and Organizing Data

```r
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)
```

## Linear Regression

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

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

```
##
## Call:
## lm(formula = log_antibody ~ age + gender + race + smoking + height +
##     weight + bmi + diabetes + hypertension + SBP + LDL + time,
##     data = training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.14743 -0.35065  0.03211  0.37738  1.53018
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  28.2069787  2.6457948  10.661  < 2e-16 ***
## age          -0.0196829  0.0021607  -9.110  < 2e-16 ***
## gender       -0.2797813  0.0173438 -16.132  < 2e-16 ***
## race2        -0.0139482  0.0386090  -0.361   0.7179
## race3        -0.0080486  0.0218346  -0.369   0.7124
## race4        -0.0463573  0.0301577  -1.537   0.1243
## smoking1      0.0219875  0.0193608   1.136   0.2562
## smoking2     -0.1815792  0.0297480  -6.104 1.13e-09 ***
## height       -0.0919586  0.0154999  -5.933 3.23e-09 ***
## weight        0.0953372  0.0164227   5.805 6.93e-09 ***
## bmi          -0.3264716  0.0471923  -6.918 5.32e-12 ***
## diabetes      0.0030653  0.0243426   0.126   0.8998
## hypertension -0.0287531  0.0290736  -0.989   0.3227
## SBP           0.0024700  0.0019002   1.300   0.1937
## LDL          -0.0001017  0.0004518  -0.225   0.8219
## time         -0.0003804  0.0001988  -1.914   0.0557 .
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.5471 on 3984 degrees of freedom
## Multiple R-squared:  0.147,  Adjusted R-squared:  0.1438
## F-statistic: 45.78 on 15 and 3984 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] 0.5639064
```

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

```
## [1] 0.1641537
```

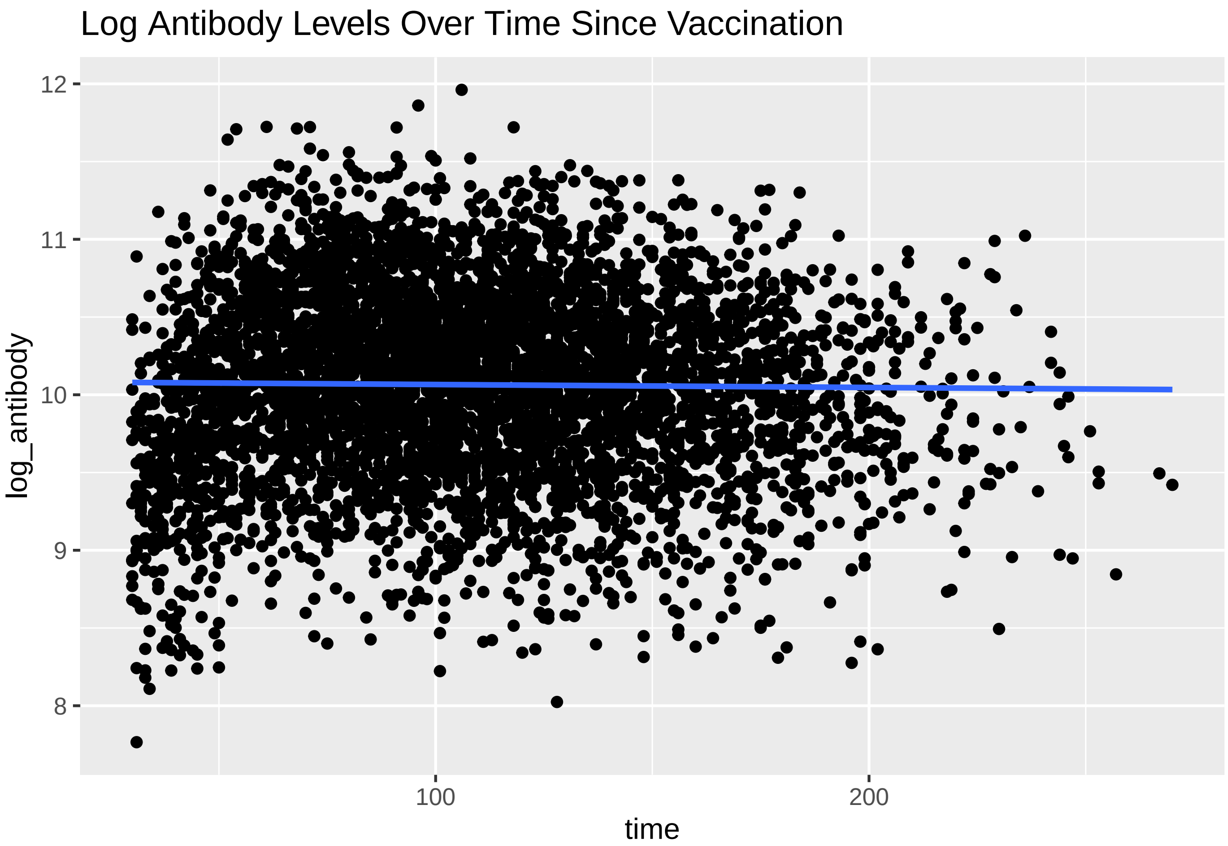
```r
generalization = predict(model, newdata = initial_test)

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

```
## [1] 0.5662817
```

```r
# 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] 0.06952672
```

```r
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")
```

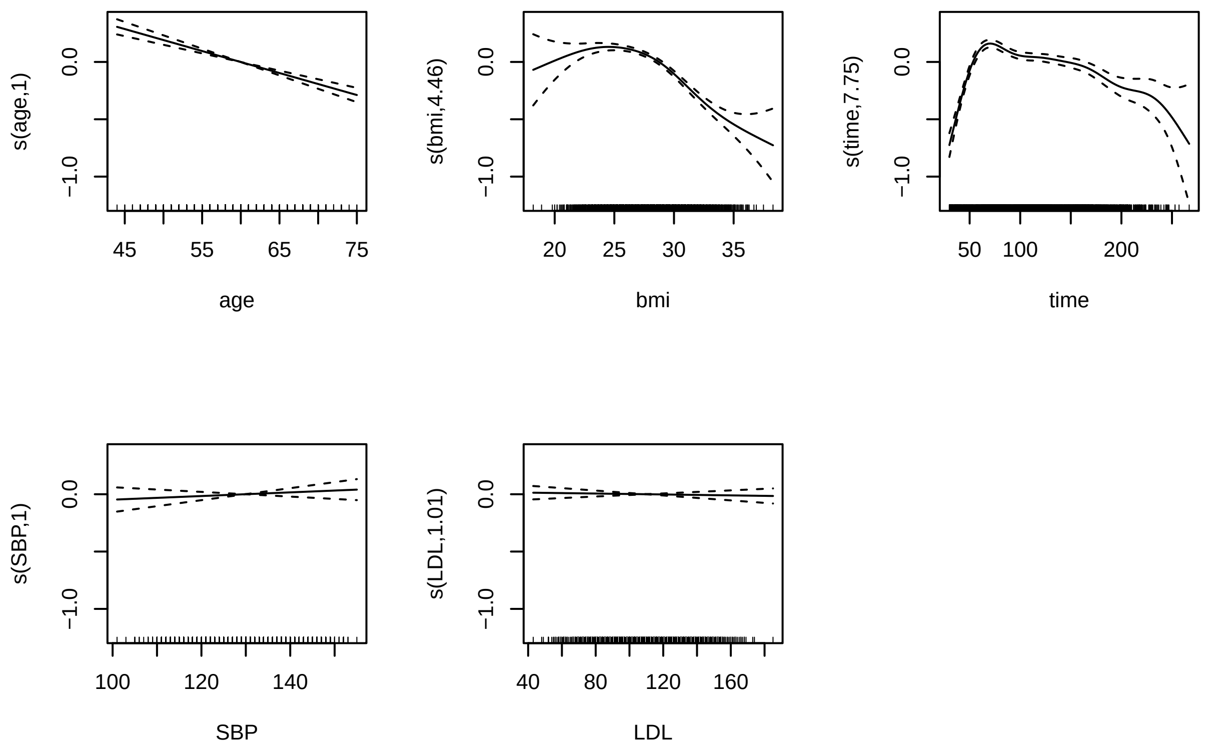


```r
###########################GAM MODEL##########################

#converted some of the variables to factor for smoother flow
```

```r
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(SBP) + 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_rsq <- rsq(gam_preds, truth = log_antibody, estimate = .pred)

#summary
summary(gam_fit$fit)
```

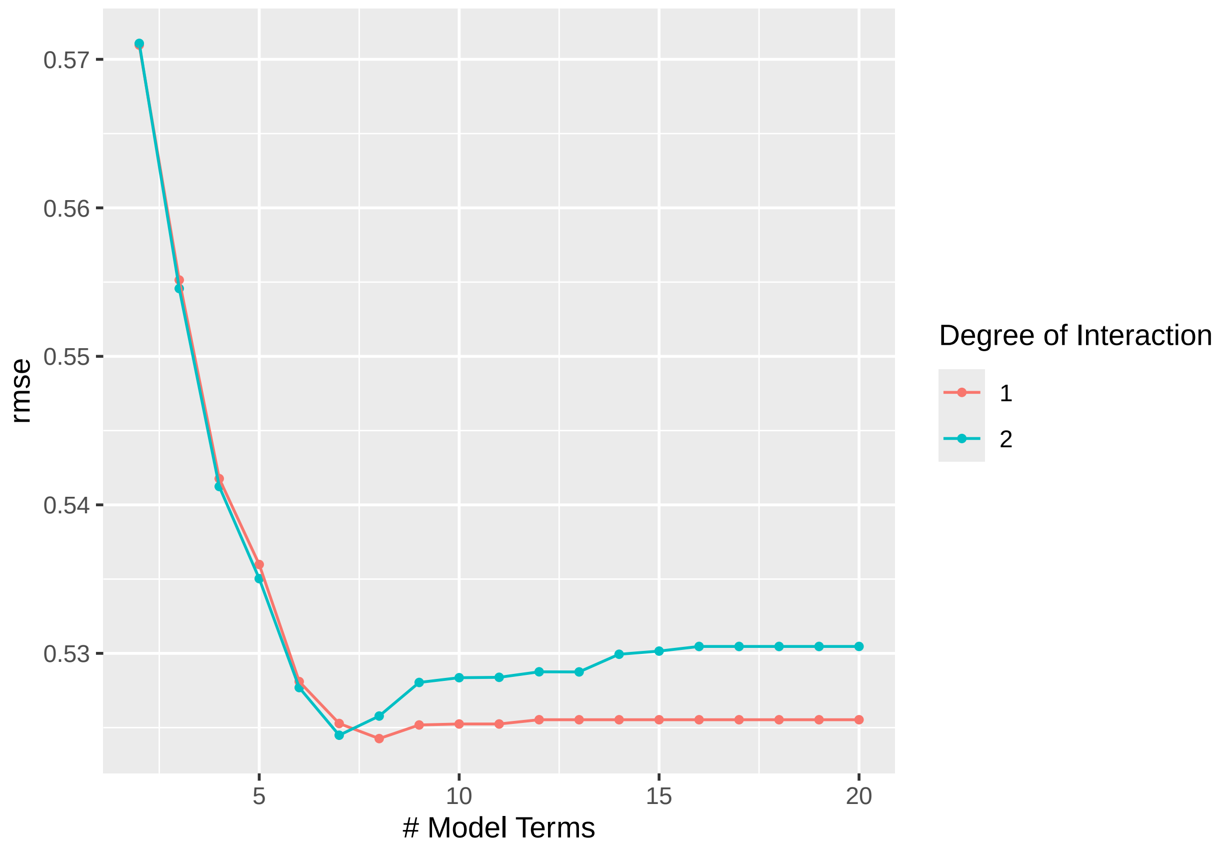
```
## 
## Family: gaussian 
## Link function: identity 
## 
## Formula:
## log_antibody ~ s(age) + s(bmi) + s(time) + gender + race + smoking + 
##     diabetes + hypertension + s(SBP) + s(LDL)
## 
## Parametric coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  10.230578   0.019746 518.109  < 2e-16 ***
## gender1      -0.280558   0.016619 -16.881  < 2e-16 ***
## race2        -0.004618   0.037008  -0.125    0.901    
## race3        -0.009939   0.020921  -0.475    0.635    
## race4        -0.041619   0.028912  -1.440    0.150    
## smoking1      0.023789   0.018553   1.282    0.200    
## smoking2     -0.186250   0.028522  -6.530  7.4e-11 ***
## diabetes      0.002780   0.023322   0.119    0.905    
## hypertension -0.025688   0.027861  -0.922    0.357    
## ---
## 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)  1.001  1.001 85.602  <2e-16 ***
## s(bmi)  4.455  5.484 64.804  <2e-16 ***
## s(time) 7.750  8.496 38.455  <2e-16 ***
## s(SBP)  1.002  1.004  0.762   0.382
## s(LDL)  1.005  1.010  0.214   0.650
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.215   Deviance explained =   22%
## -REML = 3144.8  Scale est. = 0.27446   n = 4000
```

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



```
## Libraries
library(tidymodels)
library(earth)
library(pdp)

###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 pre

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 + SBP + 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")
```



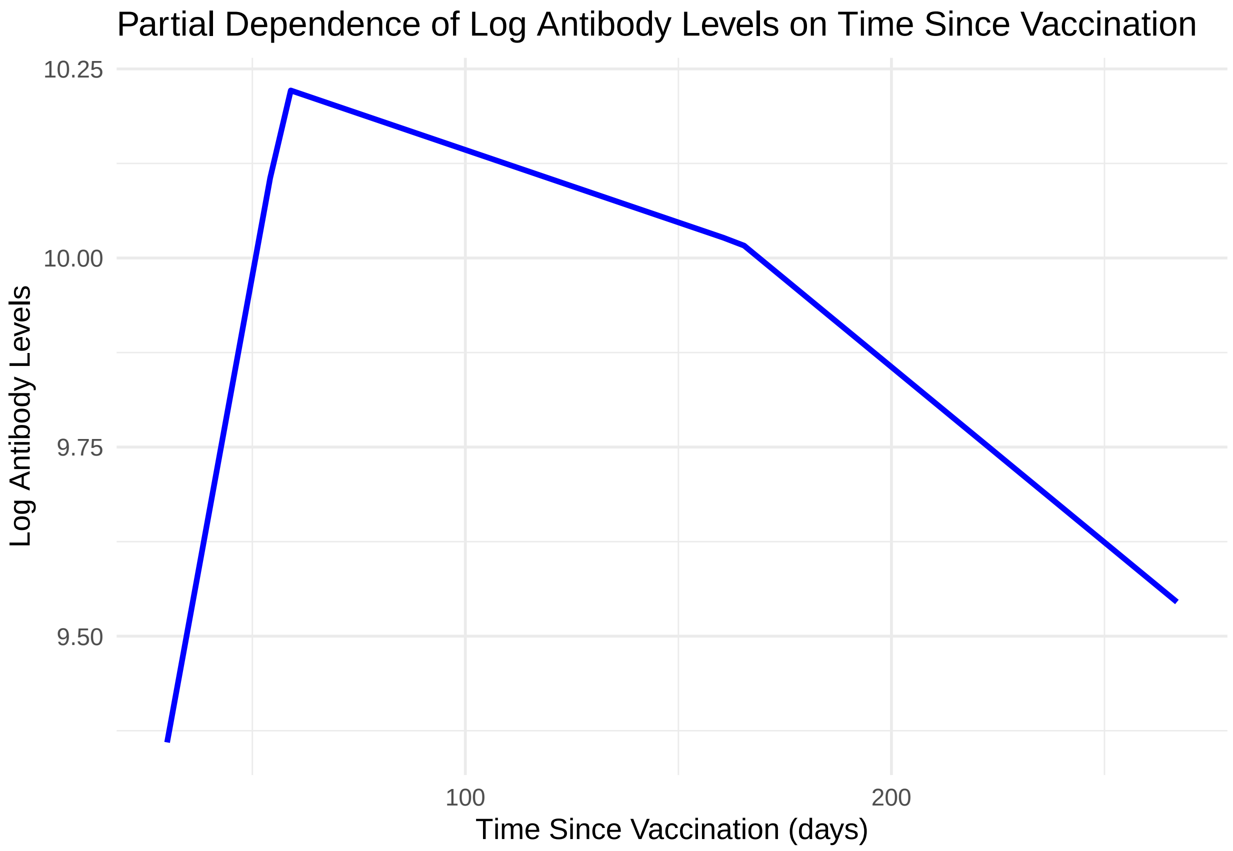
```r
# best hyperparameters Based on RMSE
mars_best <- select_best(mars_tune, metric = "rmse")

## 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 + SBP + 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
```

```r
## 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.
```


Partial Dependence of Log Antibody Levels on Time Since Vaccination
Log Antibody Levels
Time Since Vaccination (days)