```r
# load libraries
library(tidyverse)
library(caret)
library(vip)
library(rpart)
library(rpart.plot)
```

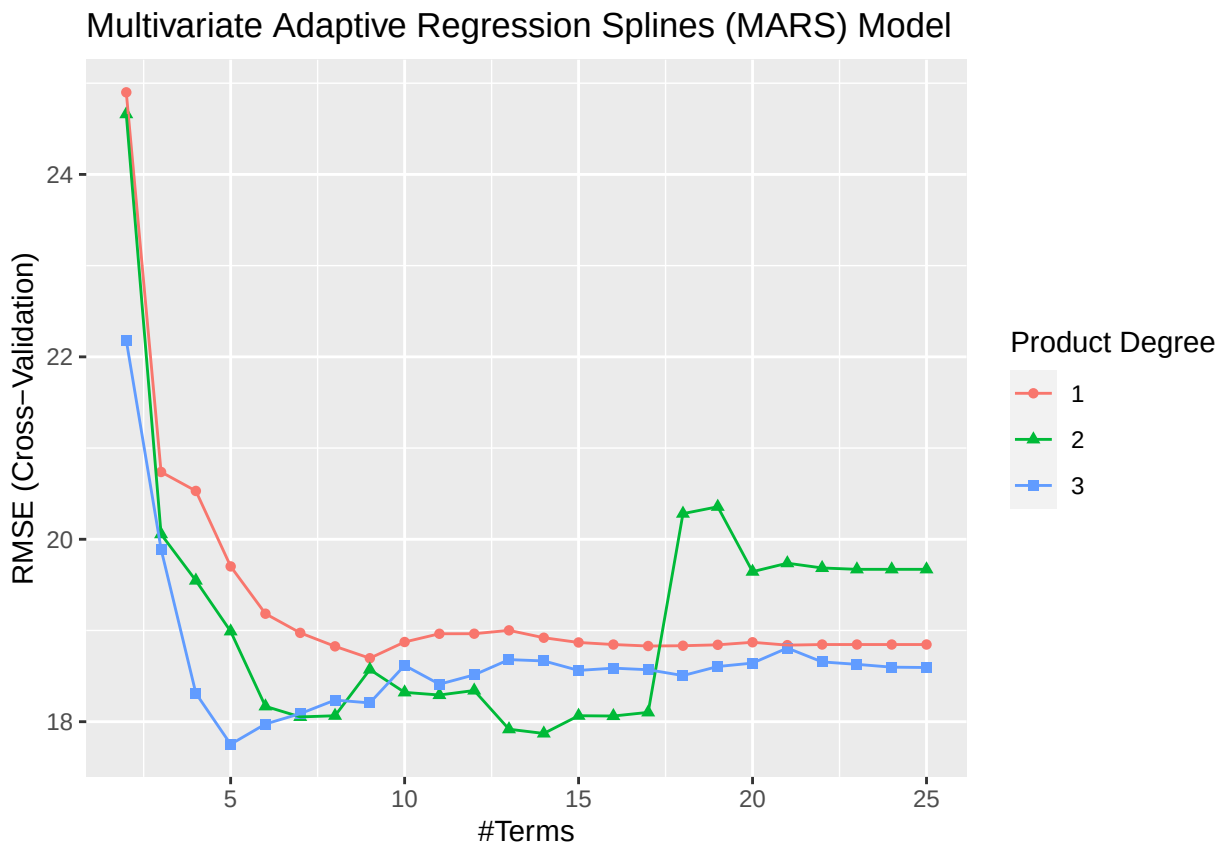
```r
# import and clean dataset
load("data/recovery.rdata")

dat %>%
  na.omit()

set.seed(5929)
final1 =
  dat[sample(1:10000, 2000),] %>%
  janitor::clean_names() %>%
  mutate (gender=as.factor(gender),
          hypertension=as.factor(hypertension),
          diabetes=as.factor(diabetes),
          vaccine=as.factor(vaccine),
          severity=as.factor(severity),
          study=as.factor(study))

set.seed(3186)
final2 =
  dat[sample(1:10000, 2000),] %>%
  janitor::clean_names() %>%
  mutate (gender=as.factor(gender),
          hypertension=as.factor(hypertension),
          diabetes=as.factor(diabetes),
          vaccine=as.factor(vaccine),
          severity=as.factor(severity),
          study=as.factor(study))

# merge 2 dataset
final = merge(final1, final2, all=TRUE) %>%
  select(-id)

# add a new binary variable
final = merge(final1, final2, all=TRUE) %>%
  select(-id) %>%
  mutate(status=case_when(recovery_time <=30 ~ 0,
                          recovery_time >30 ~ 1,
                          ) )
# Partition the dataset into two parts: training data (80%) and test data (20%)
set.seed(2023)
trRows <- createDataPartition(y = final$recovery_time, p = 0.8, list = FALSE)

# training data (80%)
trainData <- final[trRows, ]
trainData_matrix <- model.matrix(recovery_time ~., final)[ ,-1]
x <- trainData_matrix[trRows,]
```

```r
y <- final$recovery_time[trRows]

# testing data (20%)
testData <- final[-trRows, ]
testData_matrix = model.matrix(recovery_time ~., testData)[,-1]
y2 <- final$recovery_time[-trRows]

# choose methods
ctrl1 = trainControl(method = "cv", number = 10)
```

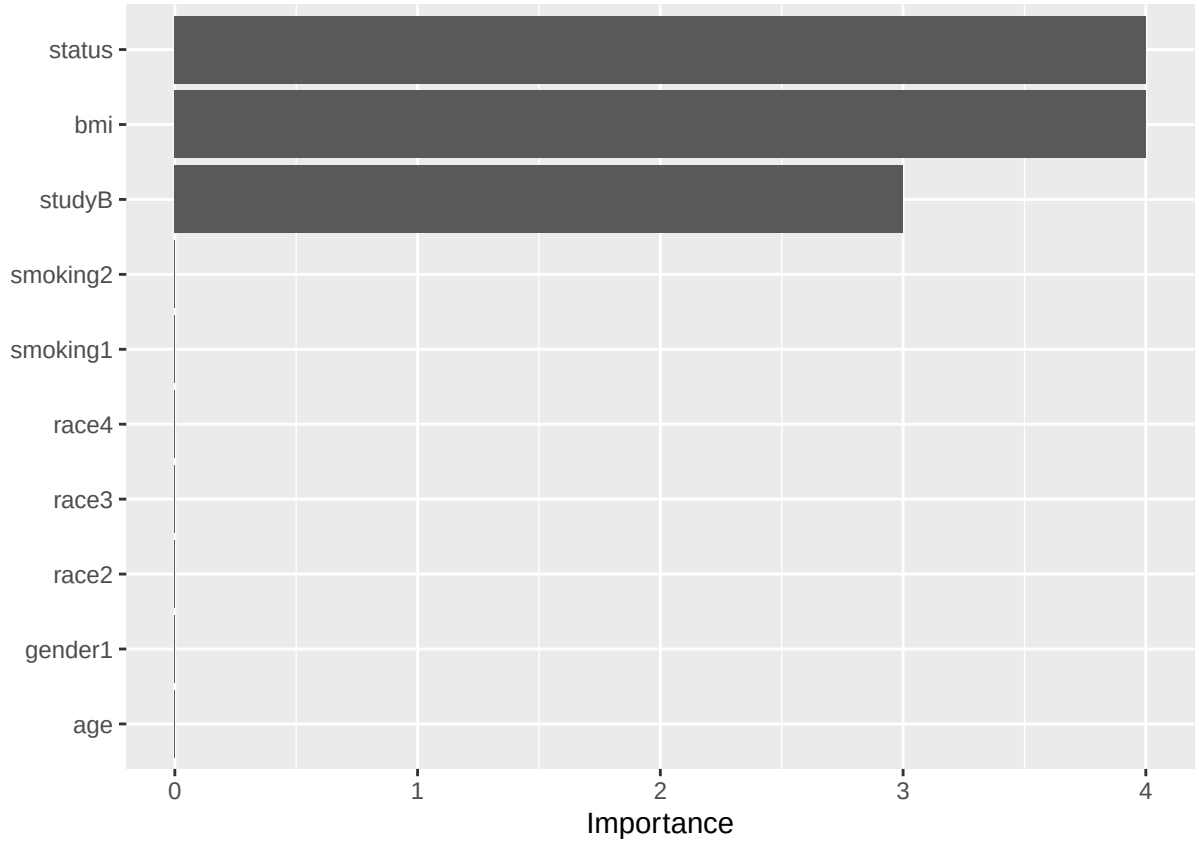
```r
# Multivariate Adaptive Regression Splines (MARS) Model
set.seed(2023)

# perform a grid search to identify optimal combination of hyperparameters
mars_grid <- expand.grid(degree = 1:3, nprune = 2:25)

# fit the model
mars.fit <- train(x, y,
                  method = "earth",
                  tuneGrid = mars_grid,
                  trControl = ctrl1)
```

```
## Loading required package: earth

## Loading required package: Formula

## Loading required package: plotmo

## Loading required package: plotrix

## Loading required package: TeachingDemos
```

```r
# check plot
ggplot(mars.fit) + labs(title = "Multivariate Adaptive Regression Splines (MARS) Model")
```




```
# variable importance plot
vip(mars.fit$finalModel)
```




```
# check results
mars.fit$bestTune
```

```
##    nprune degree
## 52      5      3
```

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

```
##                   (Intercept)                        status
##                      20.84014                      22.44707
##               studyB * status h(30.5-bmi) * studyB * status
##                    -200.22243                      26.50325
## h(bmi-22.3) * studyB * status
##                      24.74839
```

```
# make predictions and test errors
mars.pred <- predict(mars.fit, newdata = testData_matrix)
mean((mars.pred - testData$recovery_time)^2)
```

```
## [1] 303.602
```