P8106-hw2

Renjie Wei rw2844

3/7/2022

```
library(caret)
library(splines)
library(mgcv) # for gam model
library(pdp) # for partial dependence plot
library(earth) # implement MARS
library(tidyverse)
library(ggplot2)
```

Partition the dataset into two parts: training data (80%) and test data (20%).

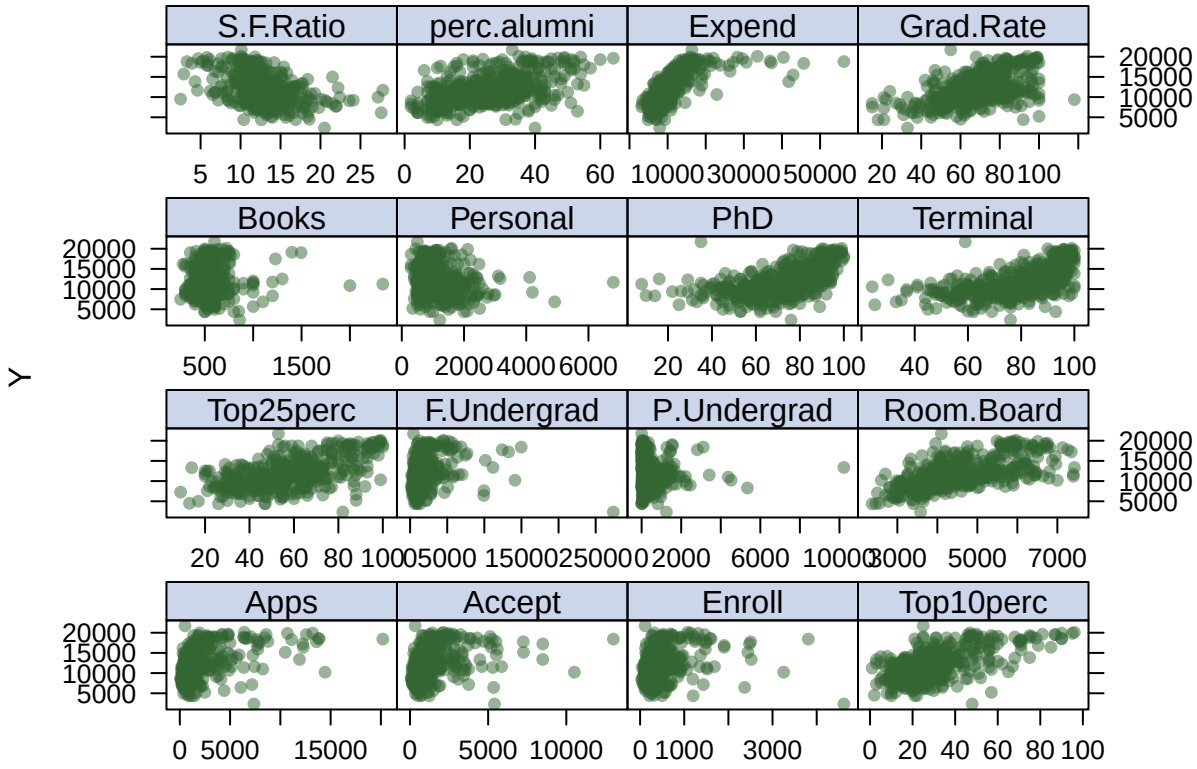
```
set.seed(2022)
college_data <- read.csv("./College.csv") %>% select(!College)
college_data <- na.omit(college_data)
college_mtx <- model.matrix(Outstate ~.,college_data)[,-1]
trainRows <- createDataPartition(y = college_data$Outstate, p = 0.8, list = FALSE)
# design matrix
train_data <- college_data[trainRows,]

test_data <- college_data[-trainRows,]
```

(a) Perform exploratory data analysis using the training data (e.g., scatter plots of response vs. predictors).

```
theme1 <- trellis.par.get()
theme1$plot.symbol$col <- rgb(.2, .4, .2, .5)
theme1$plot.symbol$pch <- 16
theme1$plot.line$col <- rgb(.8, .1, .1, 1)
theme1$plot.line$lwd <- 2
theme1$strip.background$col <- rgb(.0, .2, .6, .2)
trellis.par.set(theme1)


featurePlot(train_data %>% select(!Outstate), train_data$Outstate, plot = "scatter", labels = c("","Y")
```



From the scatterplots, we can explore the relationships between out-of-state tuition and other predictors. There are some nonlinear trend in `Expend`, `Books`, `Personal`, `PhD`, `Terminal`, `F.Undergrad`, `P.Undergrad`, `Apps`, `Accept`, `Enroll`.

```
predictors_ = c()
rsq_lm = c()
for (var_name in colnames(train_data %>% select(-Outstate))) {
    formula_  = paste("Outstate ~", var_name)
    model_ = lm(formula = formula(formula_), data = train_data)
    summary_ = summary(model_)
    rsq_ = summary_$r.squared
    predictors_ = c(predictors_, var_name)
    rsq_lm = c(rsq_lm, rsq_)
}
slr_res <- data.frame(
    variable = predictors_,
    r_squared = rsq_lm
)
arrange(slr_res, r_squared) %>% knitr::kable(digits = 3)
```

variable	r_squared
Books	0.002
P.Undergrad	0.002
F.Undergrad	0.039
Personal	0.044

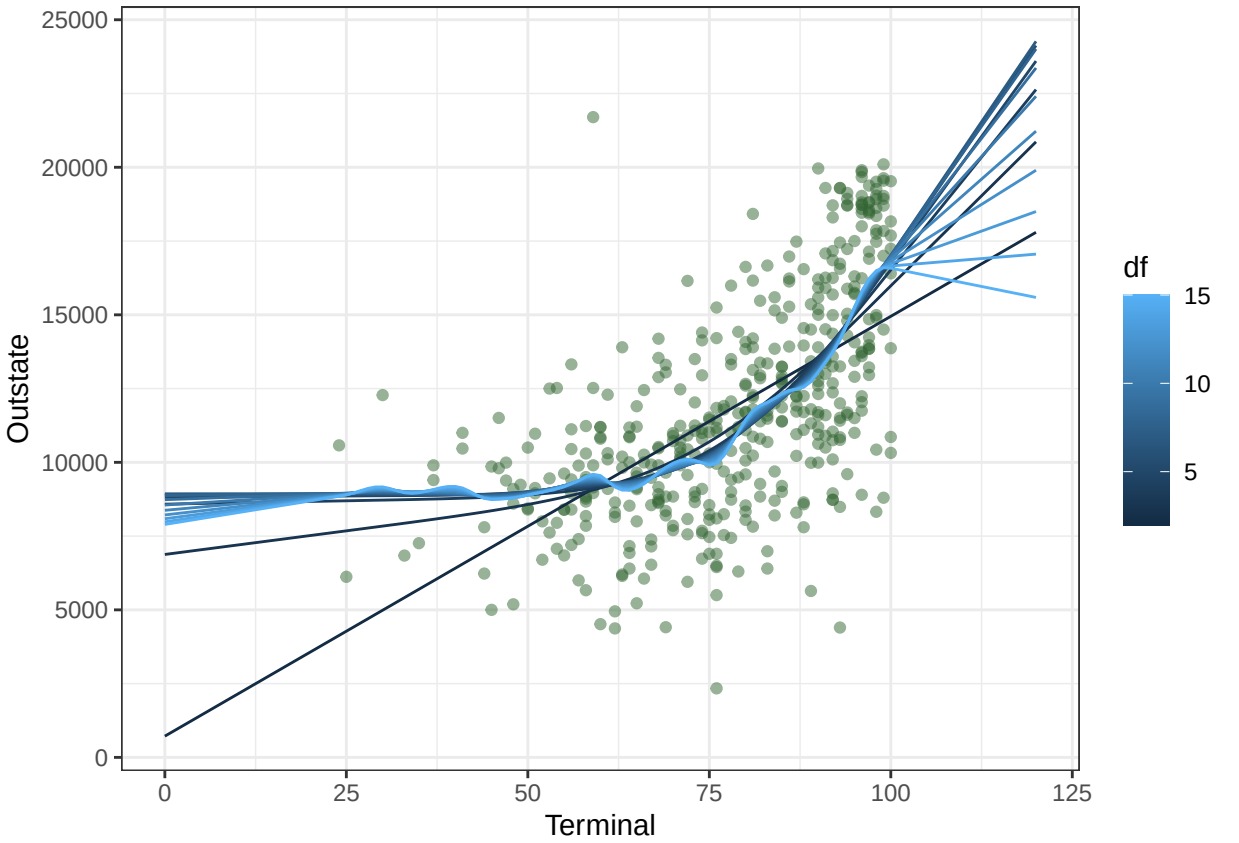
variable	r_squared
Enroll	0.069
Accept	0.146
S.F.Ratio	0.179
Apps	0.227
perc.alumni	0.231
Grad.Rate	0.287
Top25perc	0.339
PhD	0.339
Terminal	0.369
Top10perc	0.392
Room.Board	0.407
Expend	0.434

By fitting simple linear models using each variables as the only predictor, summarizing the R^2 of the model, we can see that linear model do not fully illustrate the relationship between predictors and the predicted variable `Outstate`.

(b) Fit smoothing spline models using `Terminal` as the only predictor of `Outstate` for a range of degrees of freedom, as well as the degree of freedom obtained by generalized cross-validation, and plot the resulting fits. Describe the results obtained.

```
terminal.grid <- seq(from = 0, to = 120, by = 1)
ps <- list()
for (i in 1:63){
    fit_ <- smooth.spline(train_data$Terminal, train_data$Outstate, df = i+1 )
    df_ <- fit_$df
    if(df_< i){
        next
    }
    pred_<- predict(fit_,
                    x = terminal.grid)
    preddf_ <- data.frame(pred = pred_$y,
                          terminal = terminal.grid,
                          df = rep(i+1, length(pred_$y)))
    ps <- rbind(ps, preddf_)
}

p.mass <- ggplot(data = train_data, aes(x = Terminal, y = Outstate)) +
    geom_point(color = rgb(.2, .4, .2, .5))
p.mass + geom_line(aes(x = terminal, y = pred, group = df, color = df), data = ps[which(ps$df <= 15),])
```



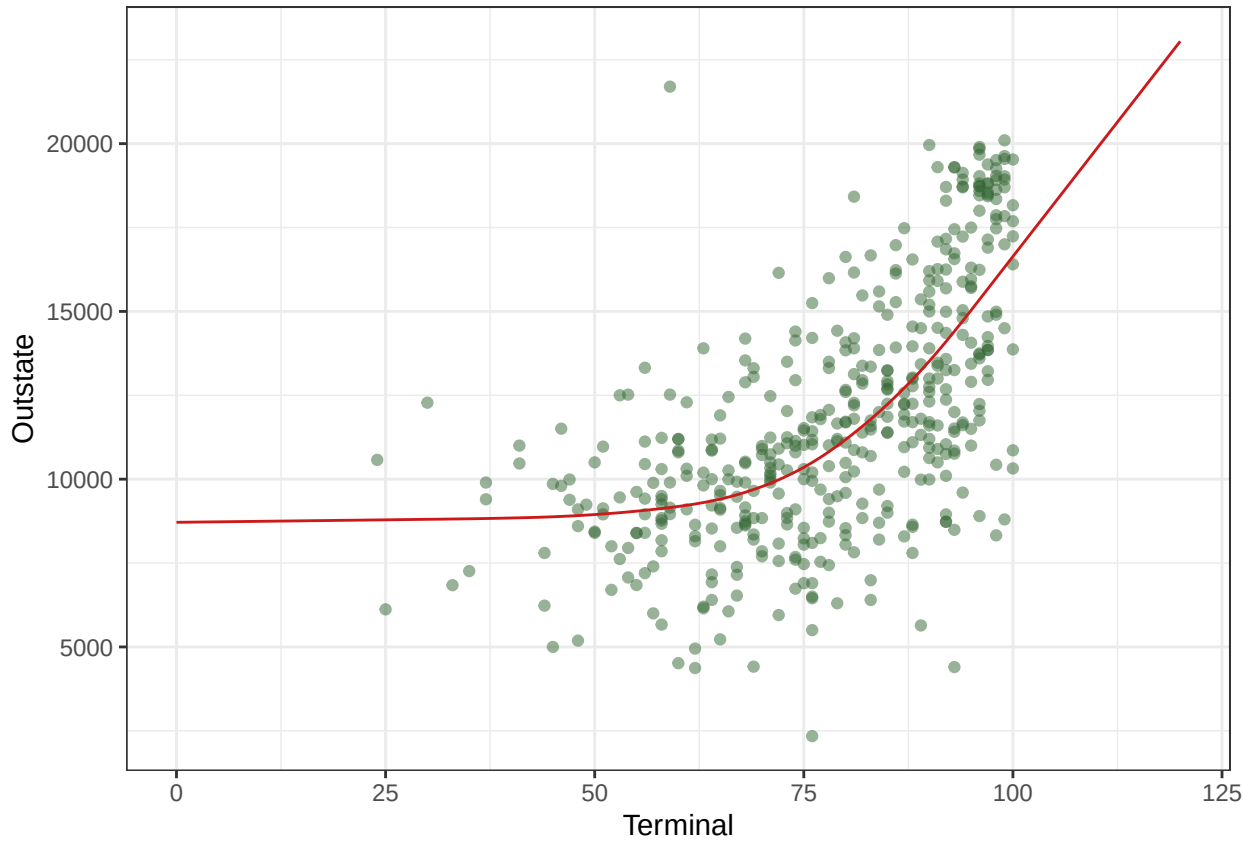
The plot above shows the fitted smoothing spline models using Terminal as the only predictor of `Outstate` for a range of degree of freedom from 2 to 15. As the `df` increase, the fitted spline curve become more and more wiggly.

```
fit.ss <- smooth.spline(train_data$Terminal, train_data$Outstate)
ss.df <- fit.ss$df

pred.ss <- predict(fit.ss,
                   x = terminal.grid)

pred.ss.df <- data.frame(pred = pred.ss$y,
                         terminal = terminal.grid)

p <- ggplot(data = train_data, aes(x = Terminal, y = Outstate)) +
     geom_point(color = rgb(.2, .4, .2, .5))
p + geom_line(aes(x = terminal, y = pred), data = pred.ss.df,
          color = rgb(.8, .1, .1, 1)) + theme_bw()
```



The degree of freedom obtained by generalized cross-validation is 4.3645446. The fitted spline curve is relatively smooth than the curves fitted with high degree of freedom. Since their shapes are very close, we would like to choose the simpler model, which is the GCV model.

(c) Fit a generalized additive model (GAM) using all the predictors. Plot the results and explain your findings. Report the test error.

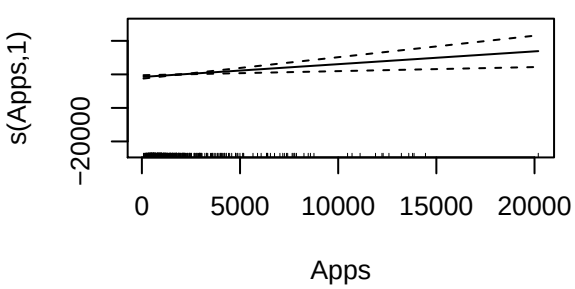
```
gam.m1 <- gam(Outstate~ s(Apps) + s(Accept) + s(Enroll)+ s(Top10perc)+s(Top25perc)+s(F.Undergrad)+s(P.U
summary(gam.m1)
```

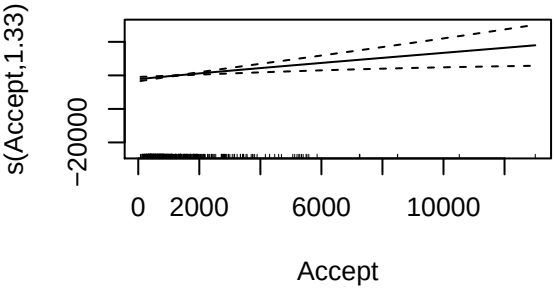
```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Outstate ~ s(Apps) + s(Accept) + s(Enroll) + s(Top10perc) + s(Top25perc) +
##     s(F.Undergrad) + s(P.Undergrad) + s(Room.Board) + s(Books) +
##     s(Personal) + s(PhD) + s(Terminal) + s(S.F.Ratio) + s(perc.alumni) +
##     s(Expend) + s(Grad.Rate)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 11851.63      72.39   163.7   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

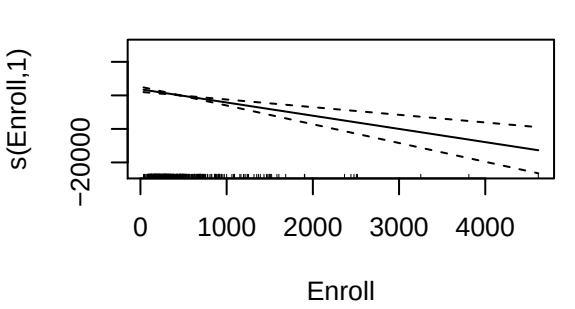
```
##
## Approximate significance of smooth terms:
##                   edf Ref.df      F  p-value
## s(Apps)         1.000  1.000  8.592  0.00357 **
## s(Accept)       1.328  1.556  6.482  0.00295 **
## s(Enroll)       1.000  1.000 22.615 3.04e-06 ***
## s(Top10perc)    7.933  8.644  1.446  0.11119
## s(Top25perc)    1.335  1.608  0.401  0.55859
## s(F.Undergrad) 6.523  7.537  5.490 4.53e-06 ***
## s(P.Undergrad) 1.000  1.000  0.002  0.96052
## s(Room.Board)  2.208  2.796 18.475  < 2e-16 ***
## s(Books)        1.951  2.446  2.015  0.16342
## s(Personal)     1.000  1.000  4.832  0.02850 *
## s(PhD)          5.708  6.805  2.025  0.05792 .
## s(Terminal)     1.000  1.000  0.620  0.43153
## s(S.F.Ratio)    3.597  4.536  1.724  0.13944
## s(perc.alumni) 1.748  2.204  6.585  0.00107 **
## s(Expend)       6.169  7.325 17.544  < 2e-16 ***
## s(Grad.Rate)    3.916  4.898  3.724  0.00284 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.825   Deviance explained = 84.3%
## GCV = 2.6577e+06  Scale est. = 2.3736e+06  n = 453
```

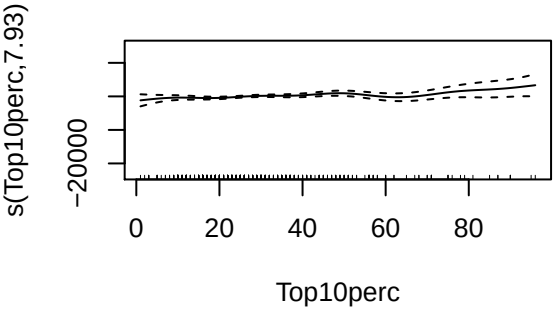
The model summary shows that some `edf` equals to `1`, which means these predictors may have linear relationships with `Outstate`.

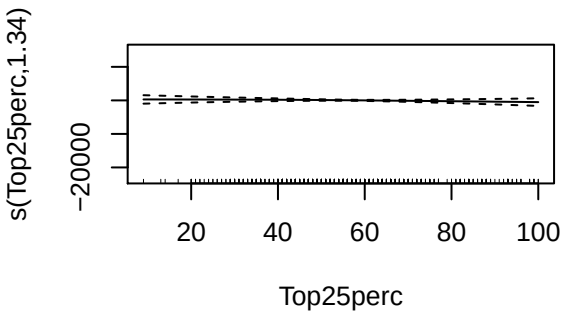
```
plot.gam(gam.m1, pages = 4)
```

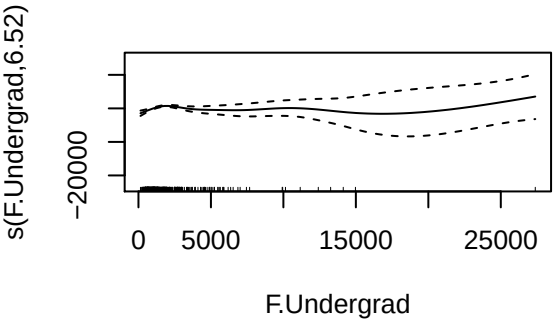

s(Apps,1)
−20000
0
5000
10000
15000
20000
Apps

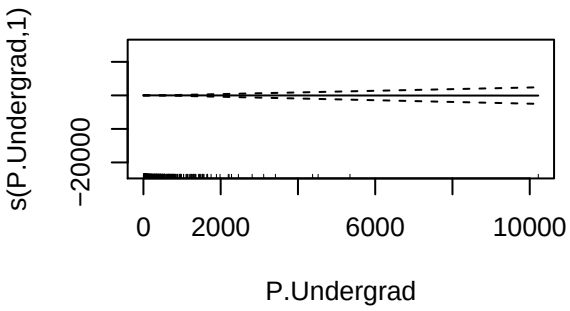

s(Accept,1.33)
−20000
0
2000
6000
10000
Accept

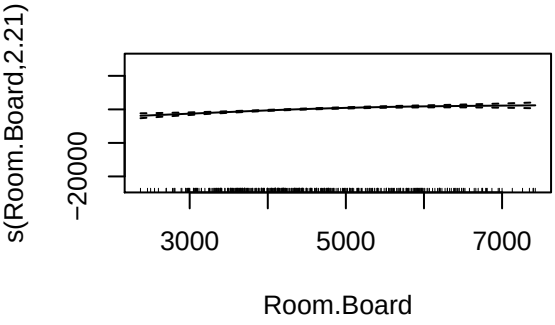

s(Enroll,1)
−20000
0
1000
2000
3000
4000
Enroll

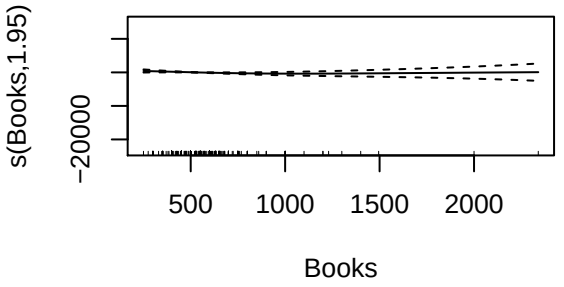

s(Top10perc,7.93)
−20000
0
20
40
60
80
Top10perc

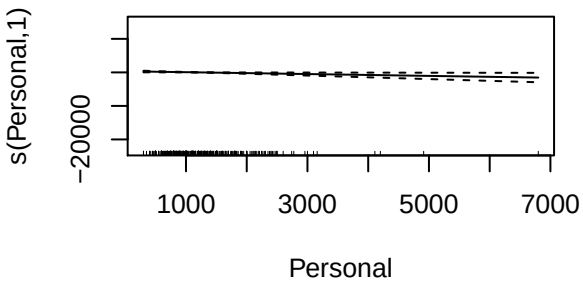

s(Top25perc,1.34)
−20000
20
40
60
80
100
Top25perc

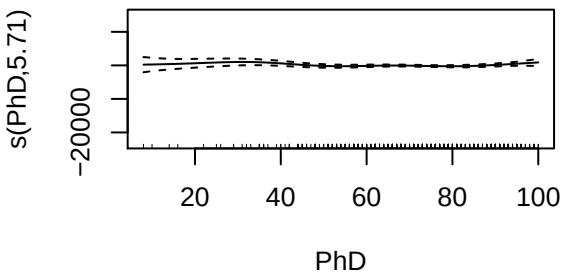

s(F.Undergrad,6.52)
−20000
0
5000
15000
25000
F.Undergrad

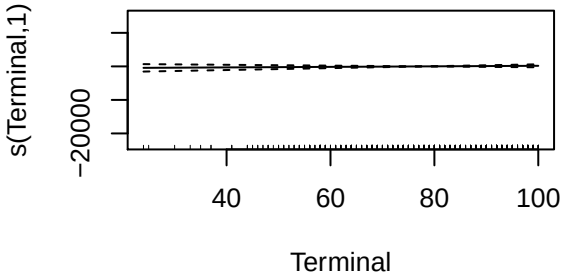

s(P.Undergrad,1)
−20000
0
2000
6000
10000
P.Undergrad

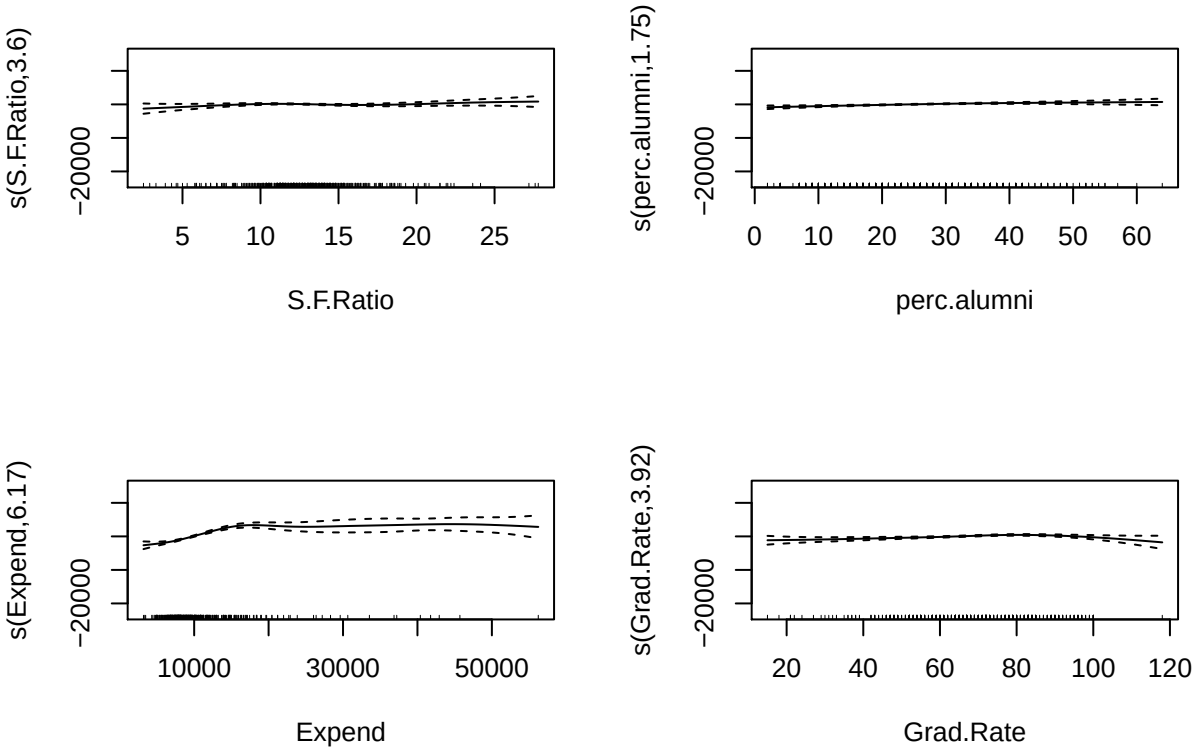

s(Room.Board,2.21)
−20000
3000
5000
7000
Room.Board


s(Books,1.95)
−20000
500
1000
1500
2000
Books


s(Personal,1)
−20000
1000
3000
5000
7000
Personal


s(PhD,5.71)
−20000
20
40
60
80
100
PhD


s(Terminal,1)
−20000
40
60
80
100
Terminal



From the plot we can also see some linear trend in `Apps`, `Enroll`, `P.Undergrad`, `Personal` and `Terminal`.

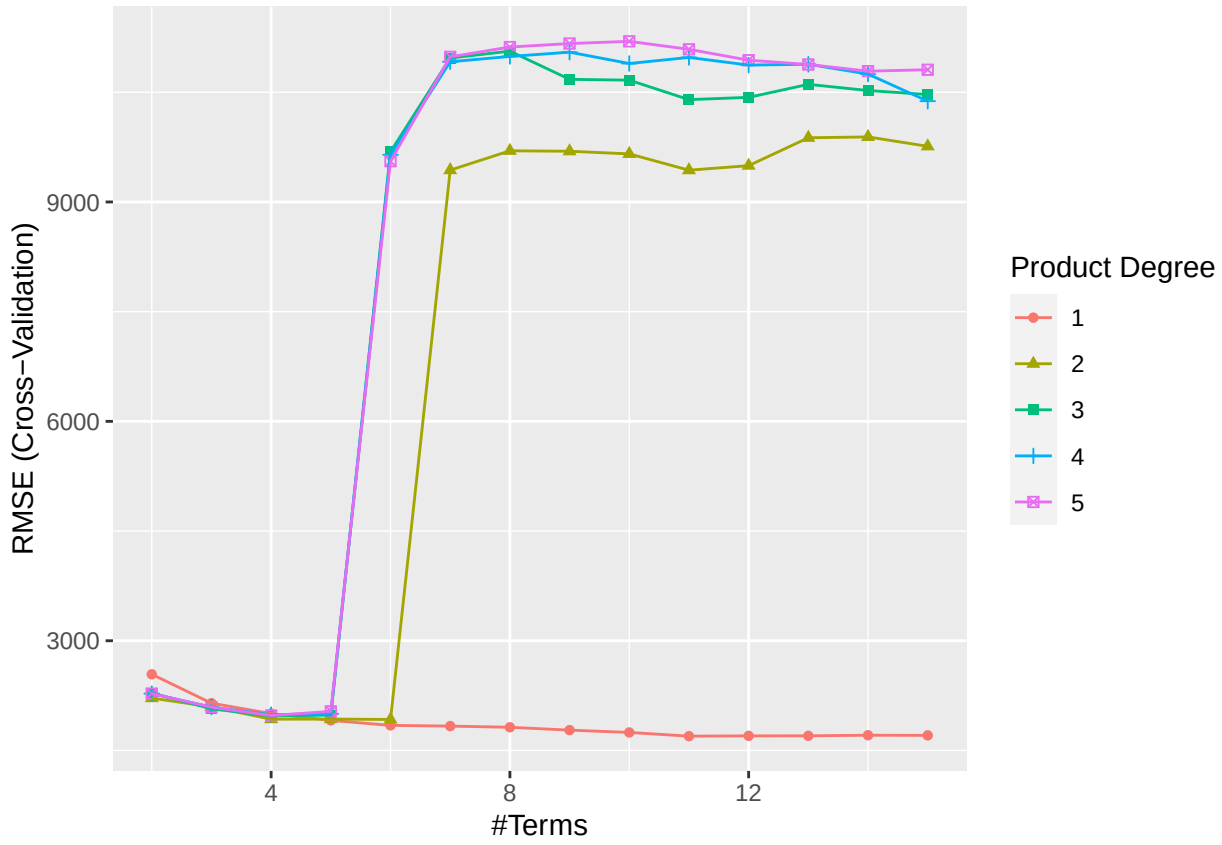
```
gam.test.predict <- predict.gam(gam.m1, newdata = test_data, type = "response")
gam.test.mse <- (RMSE(gam.test.predict, test_data$Outstate))^2
```

The test MSE of the GAM model is 3.4945204×10^6.

(d) Train a multivariate adaptive regression spline (MARS) model using all the predictors. Report the final model. Report the test error.

Train the MARS model using all predictors.

```
ctrl1 = trainControl(method = "cv", number = 10)
mars_grid <- expand.grid(degree = 1:5, nprune = 2:15)
set.seed(2022)
mars.fit <- train(train_data %>% select(-Outstate), train_data$Outstate, method = "earth", tuneGrid = ma
ggplot(mars.fit)
```



The final model is given by.

```
mars.fit$bestTune
```

```
##    nprune degree
## 10     11      1
```

There is 11 coefficients in our final model. Since the `degree = 1`, there is no products of hinge functions in our model. Present the partial dependence plot of an arbitrary predictor in your final model.

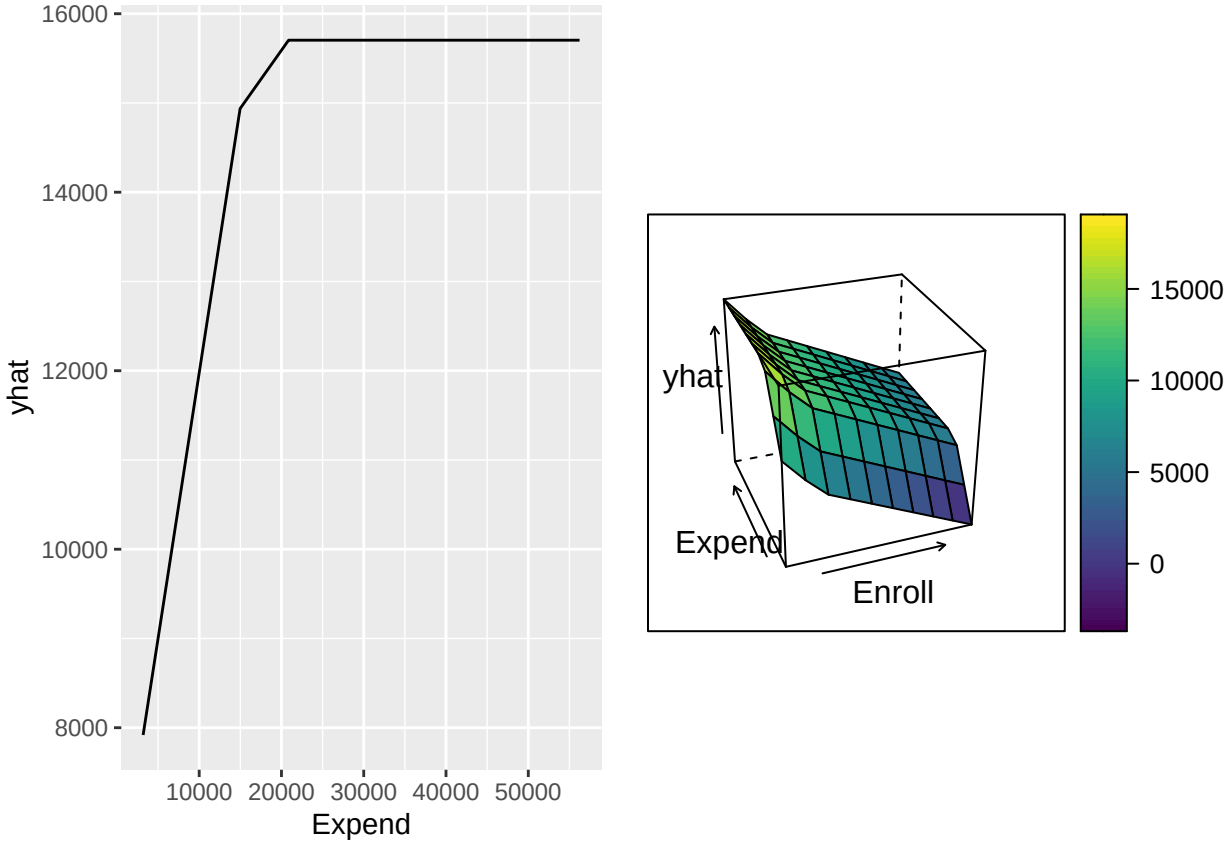
The coefficients are shown below.

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

```
##          (Intercept)     h(16262-Expend)  h(5620-Room.Board) h(1365-F.Undergrad)
##        15638.7944981          -0.5953777          -0.8472446          -1.7238957
##   h(32-perc.alumni)        h(Apps-1422)       h(Enroll-911)       h(911-Enroll)
##         -43.8939153           0.4775894          -2.0881562           5.0659210
##     h(83-Grad.Rate)    h(1323-Personal)      h(1228-Accept)
##         -23.6886814           0.9139770          -2.3199641
```

I made a partial dependence plot of `Accept` , as well as `Accept` and `Expend`.

```
p1 <- pdp::partial(mars.fit, pred.var = c("Expend"), grid.resolution = 10) %>% autoplot()
p2 <- pdp::partial(mars.fit, pred.var = c("Enroll", "Expend"),
grid.resolution = 10) %>%
pdp::plotPartial(levelplot = FALSE, zlab = "yhat", drape = TRUE,
screen = list(z = 20, x = -60))
grid.arrange(p1, p2, ncol = 2)
```




```
mars.predict <- predict(mars.fit, newdata = test_data)
mars.mse <- (RMSE(mars.predict, test_data$Outstate))^2
```

The MSE of MARS model is 3.7801717×10^6

(e) In this data example, do you prefer the use of MARS model over a linear model when predicting the out-of-state tuition? Why?

I fit a linear model and an elastic-net model to determine whether a MARS model is better than linear ones. I compared the test MSE of these models.

```
set.seed(2022)
lm.fit <- train(train_data %>% select(-Outstate), train_data$Outstate, method = "lm")
enet.fit <- train(train_data %>% select(-Outstate), train_data$Outstate, method = "glmnet", tuneGrid = 
#myCol<- rainbow(25)
#myPar <- list(superpose.symbol = list(col = myCol),
#                    superpose.line = list(col = myCol))
```

```r
# plot(enet.fit, par.settings = myPar)
lm.predict <- predict(lm.fit,newdata = test_data %>% select(-Outstate))
lm.mse <- (RMSE(test_data$Outstate, lm.predict))^2
enet.predict <- predict(enet.fit,newdata = test_data %>% select(-Outstate))
enet.mse <- (RMSE(test_data$Outstate, enet.predict))^2

tibble(model = c("MARS", "Multiple Linear Regression", "Elastic Net"),
       `Test MSE` = c(mars.mse, lm.mse, enet.mse)) %>% knitr::kable()
```

model	Test MSE
MARS	3780172
Multiple Linear Regression	4143501
Elastic Net	4164050

We can see that the MARS model has the lowest test MSE. Hence, I prefer MARS model when predicting the out-of-state tuition.