Homework 2

Minghe Wang

2025-03-16

```
data(Prostate)
summary(Prostate)
```

```
##      lcavol           lweight           age              lbph        
##  Min.   :-1.3471   Min.   :2.375   Min.   :41.00   Min.   :-1.3863  
##  1st Qu.: 0.5128   1st Qu.:3.376   1st Qu.:60.00   1st Qu.:-1.3863  
##  Median : 1.4469   Median :3.623   Median :65.00   Median : 0.3001  
##  Mean   : 1.3500   Mean   :3.653   Mean   :63.87   Mean   : 0.1004  
##  3rd Qu.: 2.1270   3rd Qu.:3.878   3rd Qu.:68.00   3rd Qu.: 1.5581  
##  Max.   : 3.8210   Max.   :6.108   Max.   :79.00   Max.   : 2.3263  
##       svi              lcp             gleason          pgg45       
##  Min.   :0.0000   Min.   :-1.3863   Min.   :6.000   Min.   :  0.00  
##  1st Qu.:0.0000   1st Qu.:-1.3863   1st Qu.:6.000   1st Qu.:  0.00  
##  Median :0.0000   Median :-0.7985   Median :7.000   Median : 15.00  
##  Mean   :0.2165   Mean   :-0.1794   Mean   :6.753   Mean   : 24.38  
##  3rd Qu.:0.0000   3rd Qu.: 1.1787   3rd Qu.:7.000   3rd Qu.: 40.00  
##  Max.   :1.0000   Max.   : 2.9042   Max.   :9.000   Max.   :100.00  
##       lpsa        
##  Min.   :-0.4308  
##  1st Qu.: 1.7317  
##  Median : 2.5915  
##  Mean   : 2.4784  
##  3rd Qu.: 3.0564  
##  Max.   : 5.5829
```

In this exercise, we explore the use of nonlinear models to analyze the "College" dataset, which contains statistics from 565 U.S. colleges, as reported in a previous issue of U.S. News & World Report. The response variable is the out-of-state tuition (Outstate), and the predictors are:

- Apps: Number of applications received
- Accept: Number of applications accepted
- Enroll: Number of new students enrolled
- Top10perc: Pct. new students from top 10% of H.S. class
- Top25perc: Pct. new students from top 25% of H.S. class
- F.Undergrad: Number of fulltime undergraduates
- P.Undergrad: Number of parttime undergraduates
- Room.Board: Room and board costs
- Books: Estimated book costs
- Personal: Estimated personal spending
- PhD: Pct. of faculty with Ph.D.'s
- Terminal: Pct. of faculty with terminal degree
- perc.alumni: Pct. alumni who donate
- Expend: Instructional expenditure per student
- Grad.Rate: Graduation rate

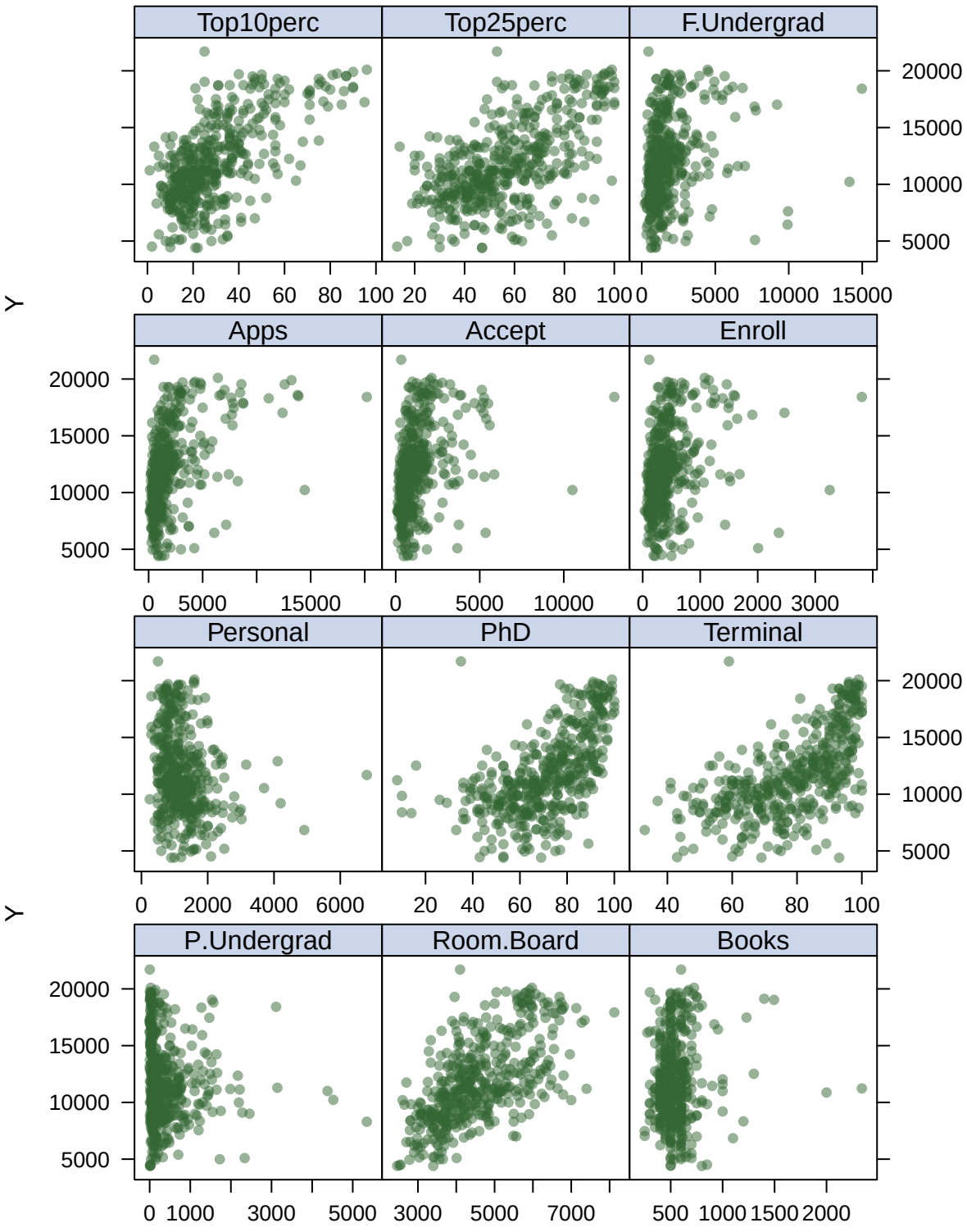
- S.F.Ratio: Student/faculty ratio

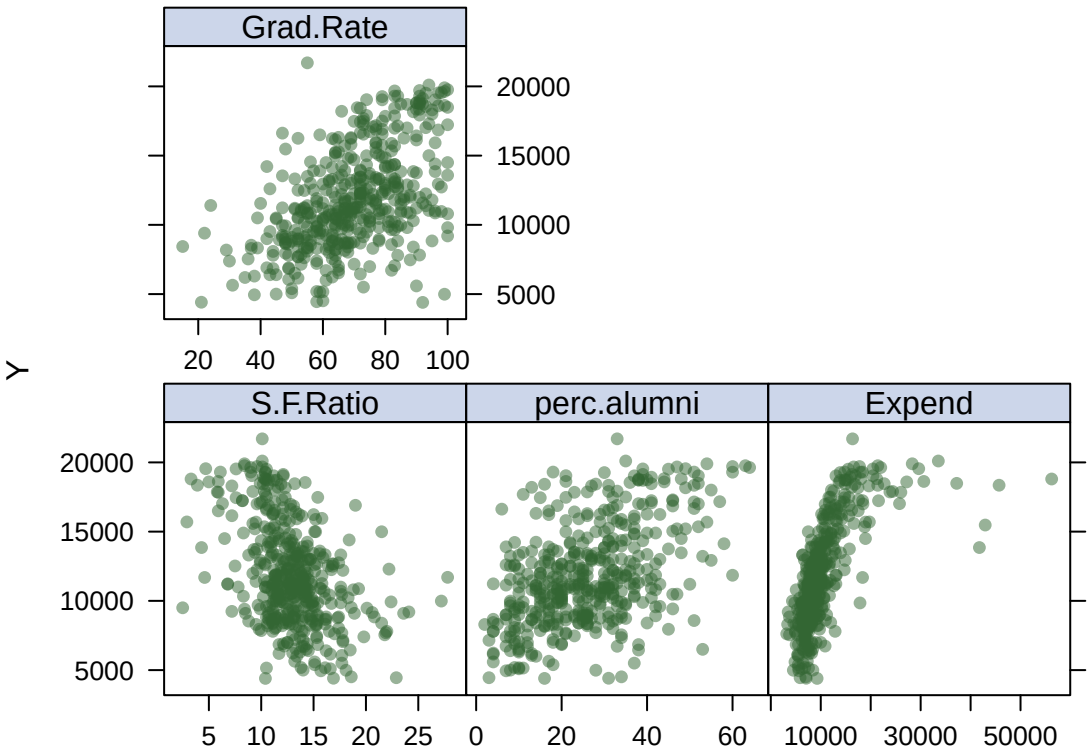
Partition the dataset into two parts: training data (80%) and test data (20%).

```
colleges <- read_csv("./College.csv")
```

```
## Rows: 565 Columns: 18
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): College
## dbl (17): Apps, Accept, Enroll, Top10perc, Top25perc, F.Undergrad, P.Undergr...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
set.seed(123)
train_indices <- createDataPartition(colleges$Outstate, p=0.8, list=FALSE)
# n <- nrow(colleges)
# train_indices <- sample(seq_len(n), size = 0.8 * n)
train_data <- colleges[train_indices, ]
test_data  <- colleges[-train_indices, ]
train_data <- train_data[, -1]
test_data <- test_data[, -1]

y_train <- train_data$Outstate

x <- model.matrix(Outstate ~ ., train_data)[, -1]
y <- y_train

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)
# Set the modified parameters as the global style for the trellis graph
trellis.par.set(theme1)

# svi and gleason were not included in the plot (they take discrete values)
featurePlot(x, y, plot = "scatter", labels = c("", "Y"),
            type = c("p"), layout = c(3, 2))
```


Top10perc
Top25perc
F.Undergrad
Apps
Accept
Enroll
Personal
PhD
Terminal
P.Undergrad
Room.Board
Books
Y
Y



Note Most partial scatter plot does not indicate a clear linearity between predictors and response variable `Ourstate`

(a) Fit smoothing spline models to predict out-of-state tuition (`Outstate`) using the percentage of alumni who donate (`perc.alumni`) as the only predictor, across a range of degrees of freedom. Plot the fitted curve for each degree of freedom. Describe the patterns you observe as the degrees of freedom change. Choose an appropriate degree of freedom for the model and plot this optimal fit. Explain the criteria you used to select the degree of freedom.

```
x_train <- train_data$perc.alumni
y_train <- train_data$Outstate

dfs <- seq(2, 10, by = 1)
pred_all <- data.frame()
p <- ggplot(data = train_data, aes(x = perc.alumni, y = Outstate)) + geom_point(color =rgb(.2, .4, .2,

x_train.grid <- seq(-10, 80, 1)

for (i in seq_along(dfs)) {
  fit.ss <- smooth.spline(train_data$perc.alumni, train_data$Outstate, df = dfs[i], cv = TRUE)
  pred <- predict(fit.ss, x_train.grid)
  pred_tmp <- data.frame(perc.alumni = x_train.grid,
                         pred = pred$y,
                         df = factor(dfs[i]))  # Convert df to factor for a proper legend
  pred_all <- rbind(pred_all, pred_tmp)
}
```

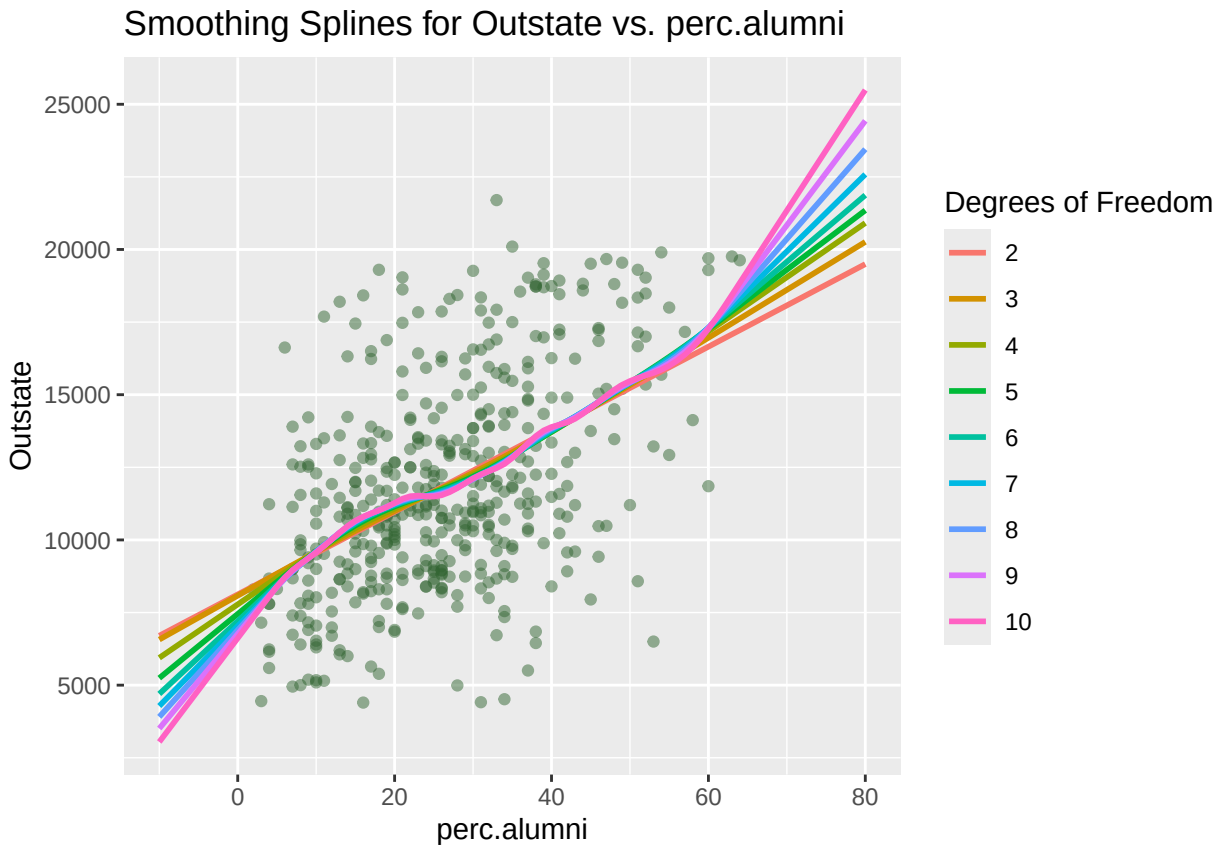
```
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
```

```
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
## Warning in smooth.spline(train_data$perc.alumni, train_data$Outstate, df =
## dfs[i], : cross-validation with non-unique 'x' values seems doubtful
```

```r
# Add the smoothing lines with color mapped to the degree of freedom
p <- p + geom_line(data = pred_all,
                   aes(x = perc.alumni, y = pred, color = df),
                   size = 1)
```

```
## 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.
```

```r
# Optionally, add a title and theme adjustments
p + labs(title = "Smoothing Splines for Outstate vs. perc.alumni",
         color = "Degrees of Freedom",
         x = "perc.alumni",
         y = "Outstate")
```



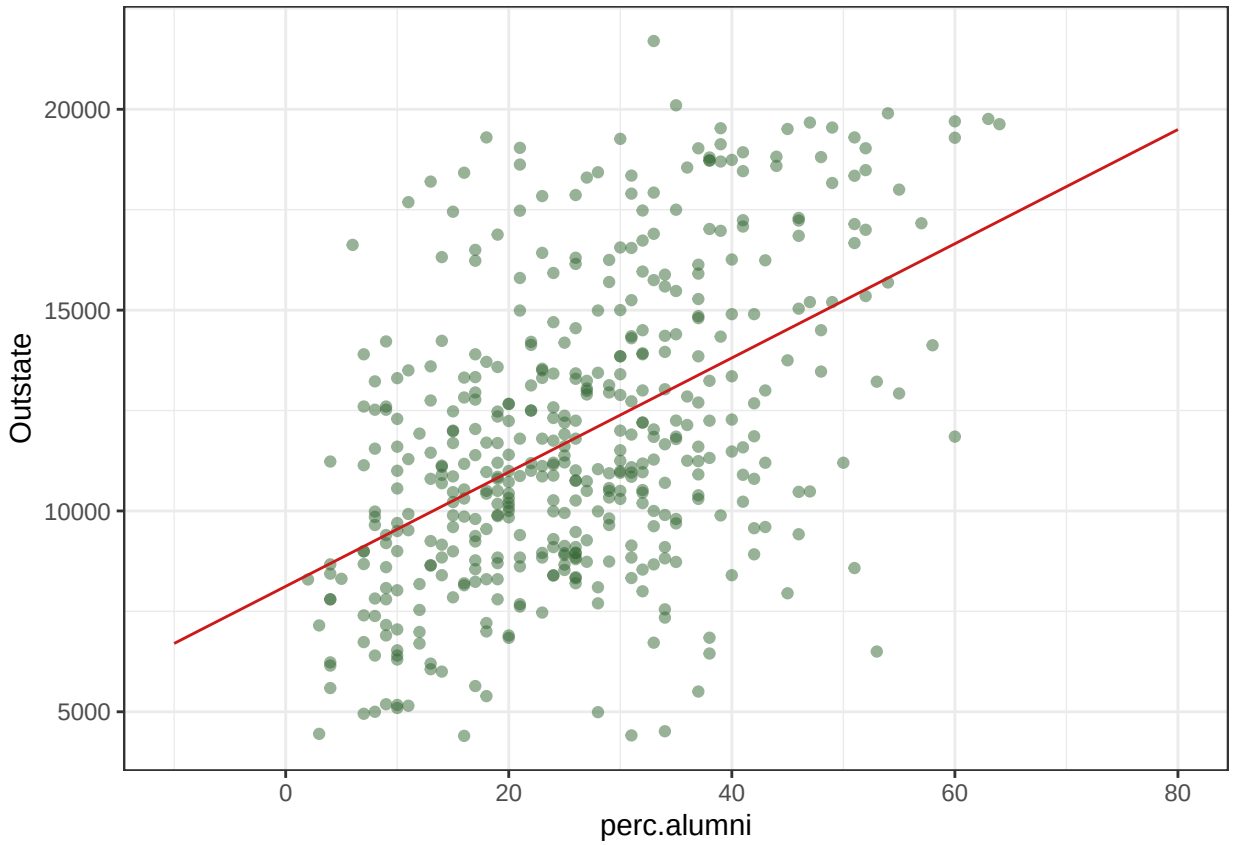

```
# Select the optimal degree of freedom using cross-validation
x_train.grid <- seq(-10, 80, 1)
fit_optim <- smooth.spline(x_train, y_train, cv = TRUE)
```

```
## Warning in smooth.spline(x_train, y_train, cv = TRUE): cross-validation with
## non-unique 'x' values seems doubtful
```

```
pred_optim <- predict(fit_optim, x_train.grid)
optimal_df <- fit_optim$df
cat("Optimal degrees of freedom selected by CV:", optimal_df, "\n")
```

```
## Optimal degrees of freedom selected by CV: 2.00025
```

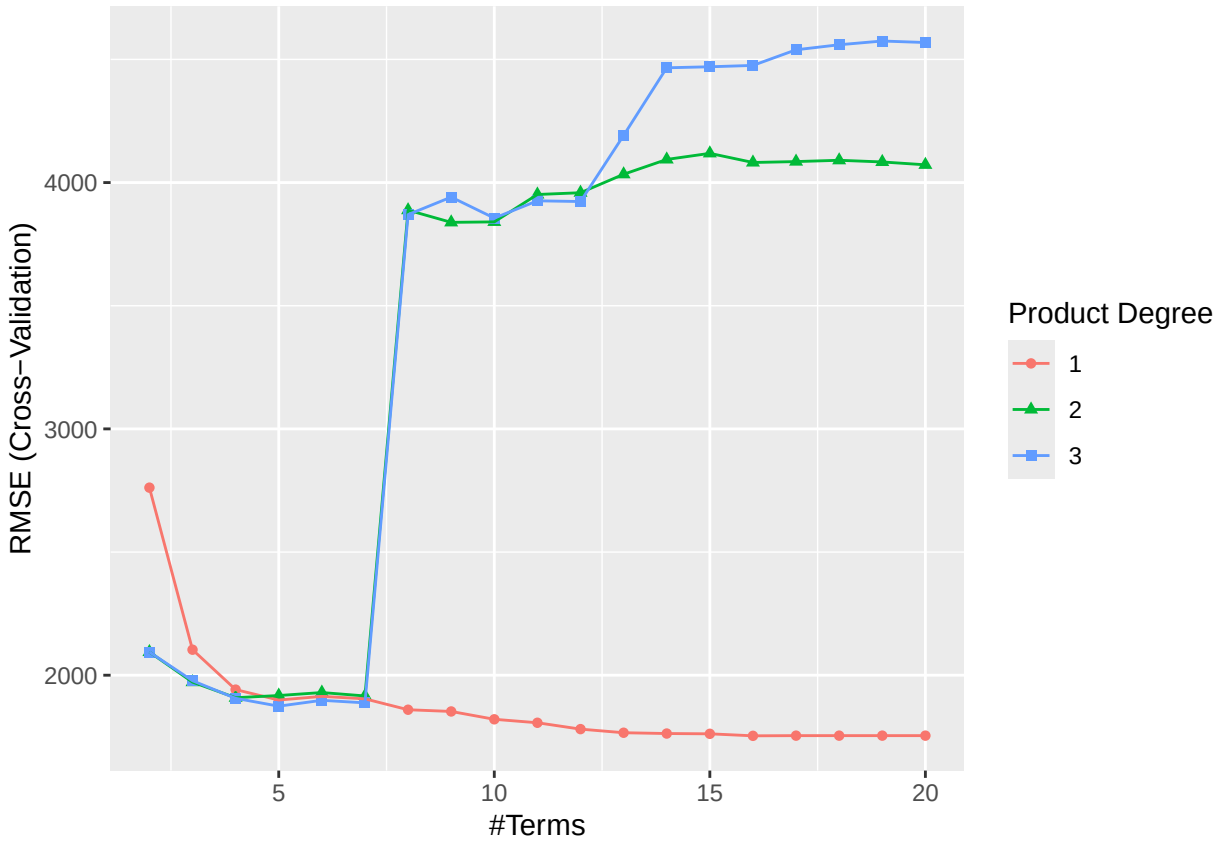
```
pred.ss.df_optim <-data.frame(pred = pred_optim$y, prec.alumni = x_train.grid)
ggplot(data = train_data, aes(x = perc.alumni, y = Outstate)) +
  geom_point(color =rgb(.2, .4, .2, .5)) +
  geom_line(aes(x = prec.alumni, y = pred), data = pred.ss.df_optim, color = rgb(.8, .1, .1, 1)) +
  theme_bw()
```



Answer From the scatter plot of `prec.alumni` vs `Outstate`, we cannot observe a clear trend of linearity. Then we use smoothing spline to plot the smooth curve for our data, it shows that as degree of freedom increases, the regression line would be more fit out scattered data, which is more wiggled. The range of degree of freedom we choose is from 2 to 10 so that we can observe straight regression lines when dof are low and wiggled lines when dof are high. We use the built-in GCV to select an optimal degree of freedom (= 2), the regression line with optimal dof indicates a linear relationship between x and y.

(b) Train a multivariate adaptive regression spline (MARS) model to predict the response variable. Report the regression function. Present the partial dependence plot of an arbitrary predictor in your model. Report the test error.

```
library(plotmo)
ctrl1 <-trainControl(method = "cv", number = 10)
set.seed(123)
mars_grid <-expand.grid(degree = 1:3,nprune = 2:20)
x <- model.matrix(Outstate ~ . , train_data)
y <- y_train
mars.fit <- train(x, y,
                  method = "earth",
                  tuneGrid = mars_grid,
                  trControl = ctrl1)
ggplot(mars.fit)
```




```
mars.fit$bestTune
```

```
##    nprune degree
## 15     16      1
```

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

```
##         (Intercept)     h(Expend-15622)  h(Room.Board-4460)  h(4460-Room.Board)
##       10684.3159852          -0.7227653           0.3113264          -1.1274658
##     h(79-Grad.Rate)    h(1300-Personal) h(F.Undergrad-1350) h(1350-F.Undergrad)
##         -28.9480175           1.0471977          -0.4456624          -1.2719896
##        h(Apps-2694)   h(21-perc.alumni)      h(Expend-6898)       h(862-Enroll)
##           0.3774909         -87.2568633           0.7187916           4.9263485
##      h(2165-Accept)
##          -2.0063276
```

```
bestMARS <- mars.fit$finalModel
summary(bestMARS)
```

```
## Call: earth(x=matrix[453,17], y=c(11250,13500,1...), keepxy=TRUE, degree=1,
##             nprune=16)
##
##                     coefficients
## (Intercept)            10684.3160
## h(Apps-2694)               0.3775
## h(2165-Accept)            -2.0063
## h(862-Enroll)              4.9263
## h(1350-F.Undergrad)       -1.2720
## h(F.Undergrad-1350)       -0.4457
```

```
## h(4460-Room.Board)       -1.1275
## h(Room.Board-4460)        0.3113
## h(1300-Personal)          1.0472
## h(21-perc.alumni)       -87.2569
## h(Expend-6898)            0.7188
## h(Expend-15622)          -0.7228
## h(79-Grad.Rate)         -28.9480
##
## Selected 13 of 22 terms, and 9 of 17 predictors (nprune=16)
## Termination condition: RSq changed by less than 0.001 at 22 terms
## Importance: Expend, Room.Board, perc.alumni, Accept, F.Undergrad, Apps, ...
## Number of terms at each degree of interaction: 1 12 (additive model)
## GCV 2825823    RSS 1142705503    GRSq 0.7887777    RSq 0.8106129
```

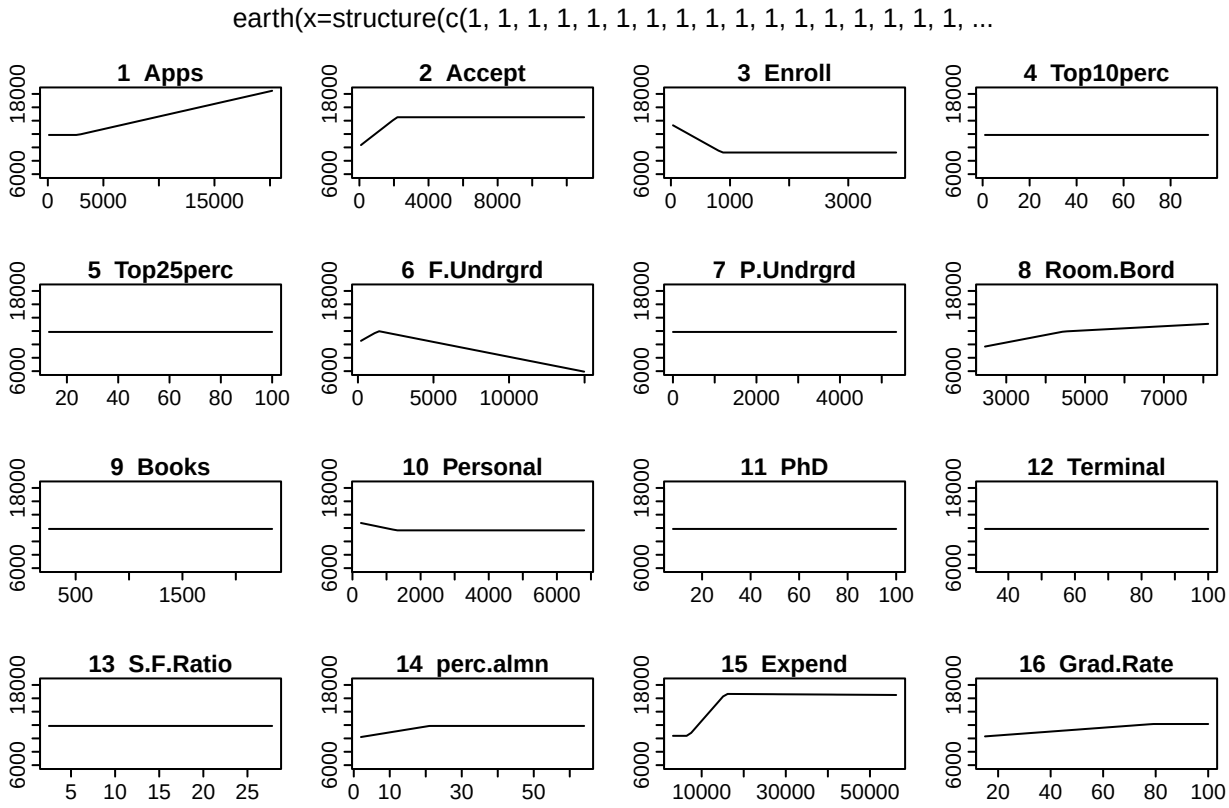
```
plotmo(bestMARS,
        nresponse = 1,
        degree2 = FALSE,
        varnames = "Apps"
)
```

```
## Warning: predict.earth ignored argument 'varnames'
```

```
## Warning: Cannot determine which variables to plot (use all1=TRUE?)
##              ncol(x) 16 < nrow(modvars) 17
##              colnames(x)=c(Apps,Accept,Enroll,Top10perc,Top25perc,F.Undergrad,P.Undergrad,Room.Board
##              rownames(modvars)=c((Intercept),Apps,Accept,Enroll,Top10perc,Top25perc,F.Undergrad,P.Un
```

```
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
## Warning: predict.earth ignored argument 'varnames'
```

```
##  plotmo grid:    Apps Accept Enroll Top10perc Top25perc F.Undergrad P.Undergrad
##                  1110    845    328        25        55        1247         191
##  Room.Board Books Personal PhD Terminal S.F.Ratio perc.alumni Expend Grad.Rate
##         4400   500     1100  73       81      12.8          26   8953        69
```




```
x_test <- model.matrix(Outstate ~ . , test_data)[, -1]
mars_preds <- predict(bestMARS, newdata = x_test)
test_mse <- mean((mars_preds - test_data$Outstate)^2)
test_rmse <- sqrt(test_mse)
cat("Test RMSE:", test_rmse, "\n")
```

```
## Test RMSE: 1726.206
```

Answer We use 10 fold cross validation process to select the best MARS regression model, which is shown above. The model has nprune = 16 and degree = 1, indicating this is a additive-only, 16 basis function MARS model. The partial dependence plot shows that several variables, such as `PhD`, `Books`, etc, contribute to prediction is negligible after considering more dominant variables, while few variables like `Apps` is clearly influencing the prediction (eg. positively). Test RMSE of the MARS model is 1726.206279

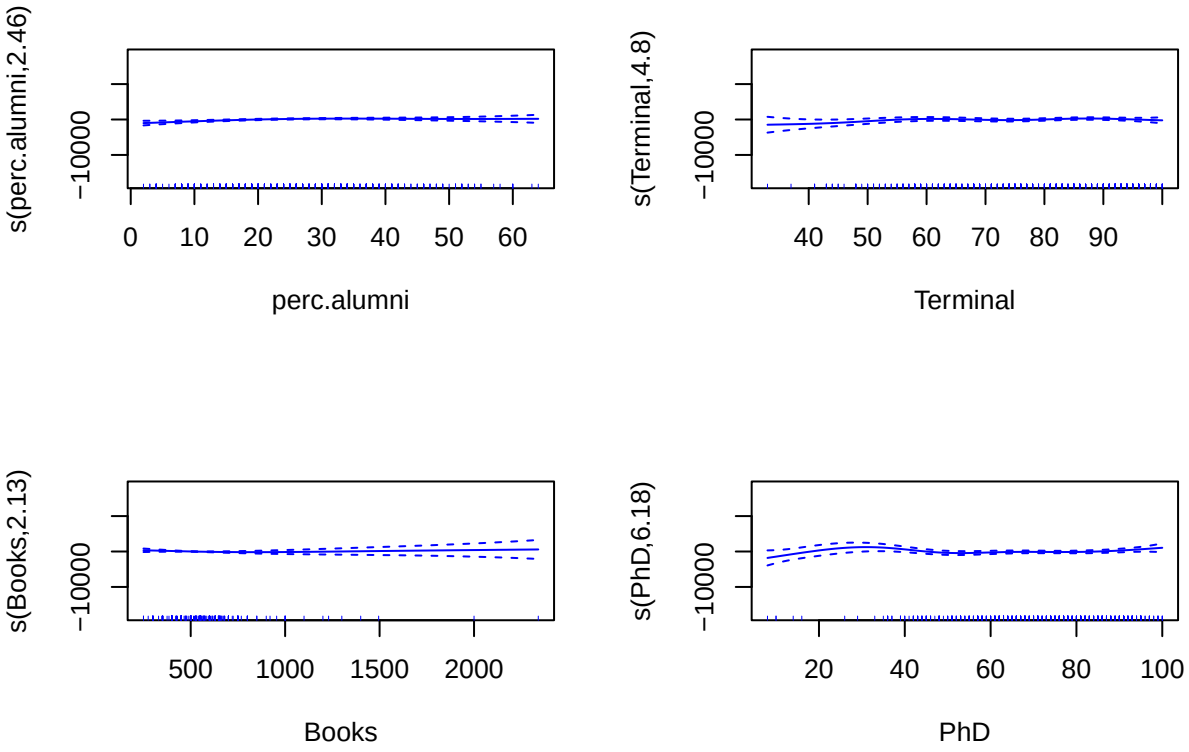
(c) Construct a generalized additive model (GAM) to predict the response variable. For the nonlinear terms included in your model, generate plots to visualize these relationships and discuss your observations. Report the test error.

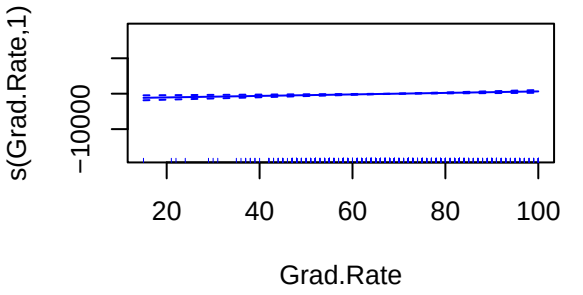
```
set.seed(2)
gam.fit <-train(x, y,
                method = "gam",
                trControl = ctrl1)

bestGAM <- gam.fit$finalModel
bestGAM
```

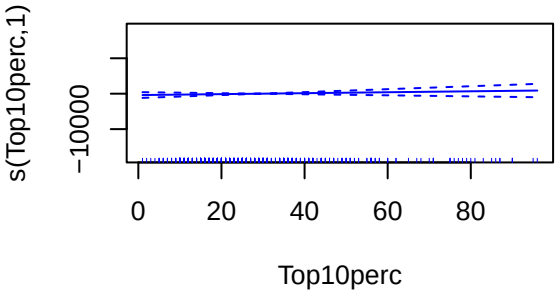
```
##
## Family: gaussian
## Link function: identity
##
```

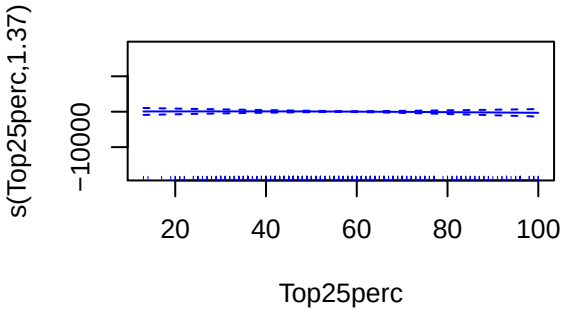
```
## Formula:
## .outcome ~ s(perc.alumni) + s(Terminal) + s(Books) + s(PhD) +
##     s(Grad.Rate) + s(Top10perc) + s(Top25perc) + s(S.F.Ratio) +
##     s(Personal) + s(P.Undergrad) + s(Enroll) + s(Room.Board) +
##     s(Accept) + s(F.Undergrad) + s(Apps) + s(Expend)
##
## Estimated degrees of freedom:
## 2.46 4.80 2.13 6.18 1.00 1.00 1.37
## 3.46 1.00 1.00 1.00 2.17 2.49 5.24
## 1.00 6.19  total = 43.48
##
## GCV score: 2834679
```

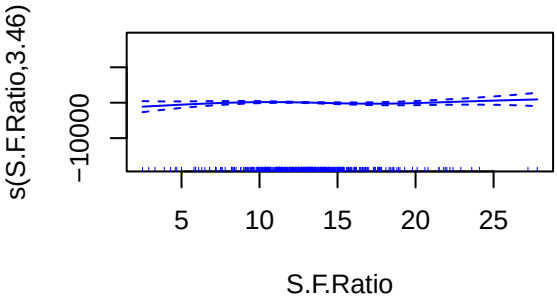
```
par(mfrow=c(2,2))
plot(bestGAM, se = TRUE, col = "blue")
```

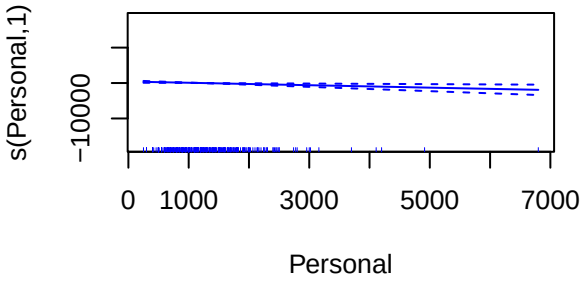


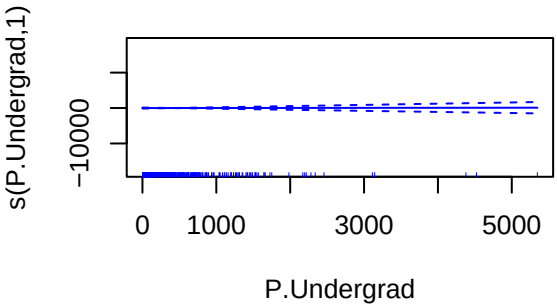

s(Grad.Rate,1)
−10000
20
40
60
80
100
Grad.Rate

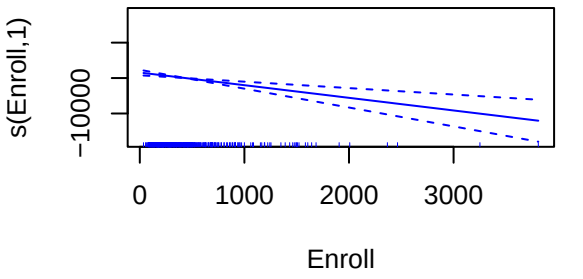

s(Top10perc,1)
−10000
0
20
40
60
80
Top10perc

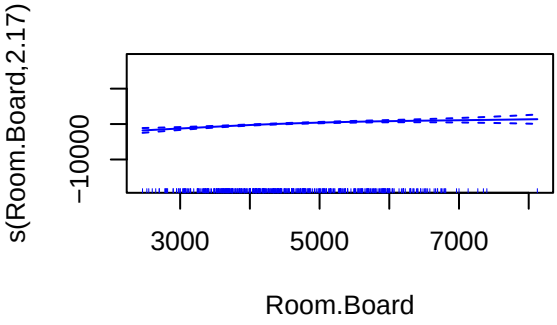

s(Top25perc,1.37)
−10000
20
40
60
80
100
Top25perc

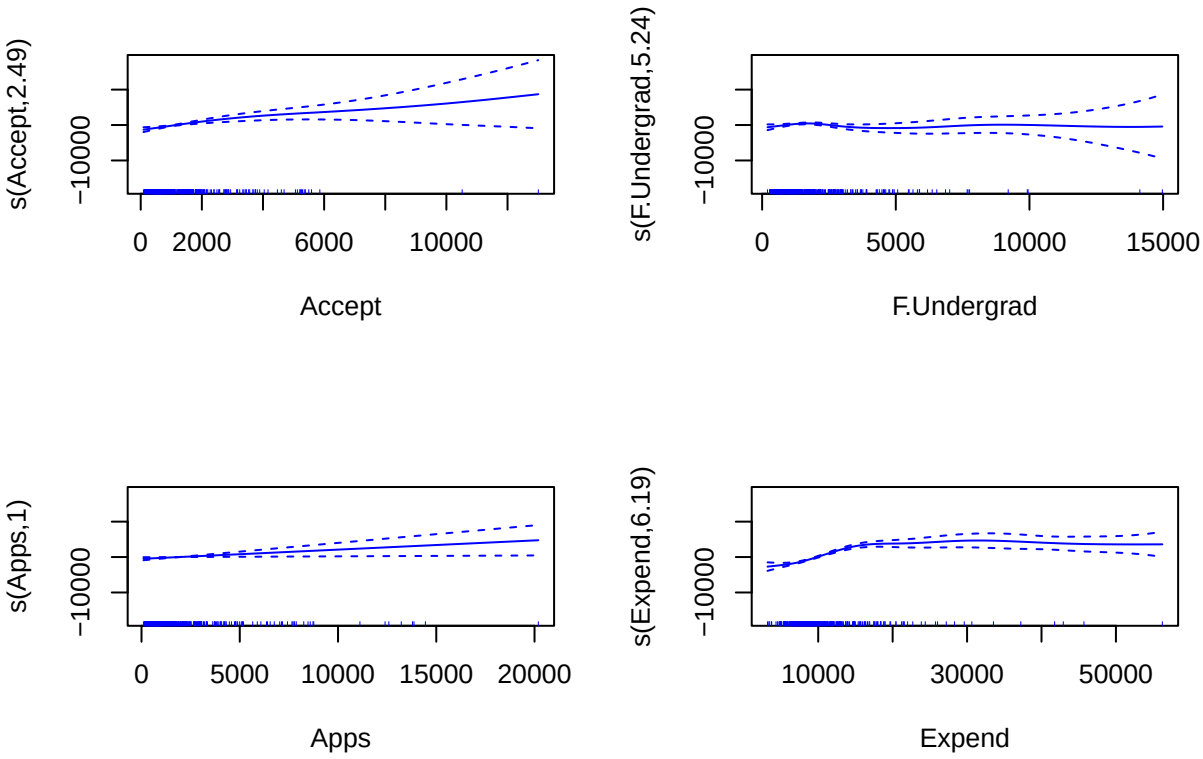

s(S.F.Ratio,3.46)
−10000
5
10
15
20
25
S.F.Ratio


s(Personal,1)
−10000
0
1000
3000
5000
7000
Personal


s(P.Undergrad,1)
−10000
0
1000
3000
5000
P.Undergrad


s(Enroll,1)
−10000
0
1000
2000
3000
Enroll


s(Room.Board,2.17)
−10000
3000
5000
7000
Room.Board




```
gam_preds <- predict(bestGAM, newdata=test_data)
gam_mse <- mean((gam_preds - test_data$Outstate)^2)
gam_rmse <- sqrt(gam_mse)
gam_rmse
```

```
## [1] 1688.746
```

Answer The GAM model identified several variables exhibiting clear non-linear relationships with Out-of-state tuition. Notably, `Expend`, `F.Undergrad`, `Accept`, and `Apps` showed significant non-linear effects. With expenditures having the most pronounced impacts. Other variables, including percentage of alumni donors, student-faculty ratio, and graduation rates, displayed relatively flat smooth functions, suggesting minimal effect on tuition after controlling for other variables. Test RMSE of GAM model is 1688.7455509

(d) In this dataset, would you favor a MARS model over a linear model for predicting out-of- state tuition? If so, why? More broadly, in general applications, do you consider a MARS model to be superior to a linear model? Please share your reasoning.

```
set.seed(123)
# Fit a linear model
lm_model <- lm(Outstate ~ ., data = train_data)
# Perform prediction
lm_preds <- predict(lm_model, newdata = test_data)
# Calculate test error
lm_rmse <- sqrt(mean((lm_preds - test_data$Outstate)^2))
lm_rmse
```

```
## [1] 2052.888
```

```
test_rmse
```

```
## [1] 1726.206
```

Answer For predicting out-of-state tuition in the College dataset, the MARS model appears to offer

significant advantages by flexibly modeling non-linearities and interactions. In this context, I would favor the MARS model over a simple linear model if it demonstrates lower test error and captures important variable relationships that a linear model overlooks.

More broadly, while MARS can be superior in cases where the underlying relationships are complex, the choice between a MARS model and a linear model should be guided by the specific characteristics (eg. linearity) of the data and the balance between interpretability and prediction accuracy.