Module 7: Recommended Exercises

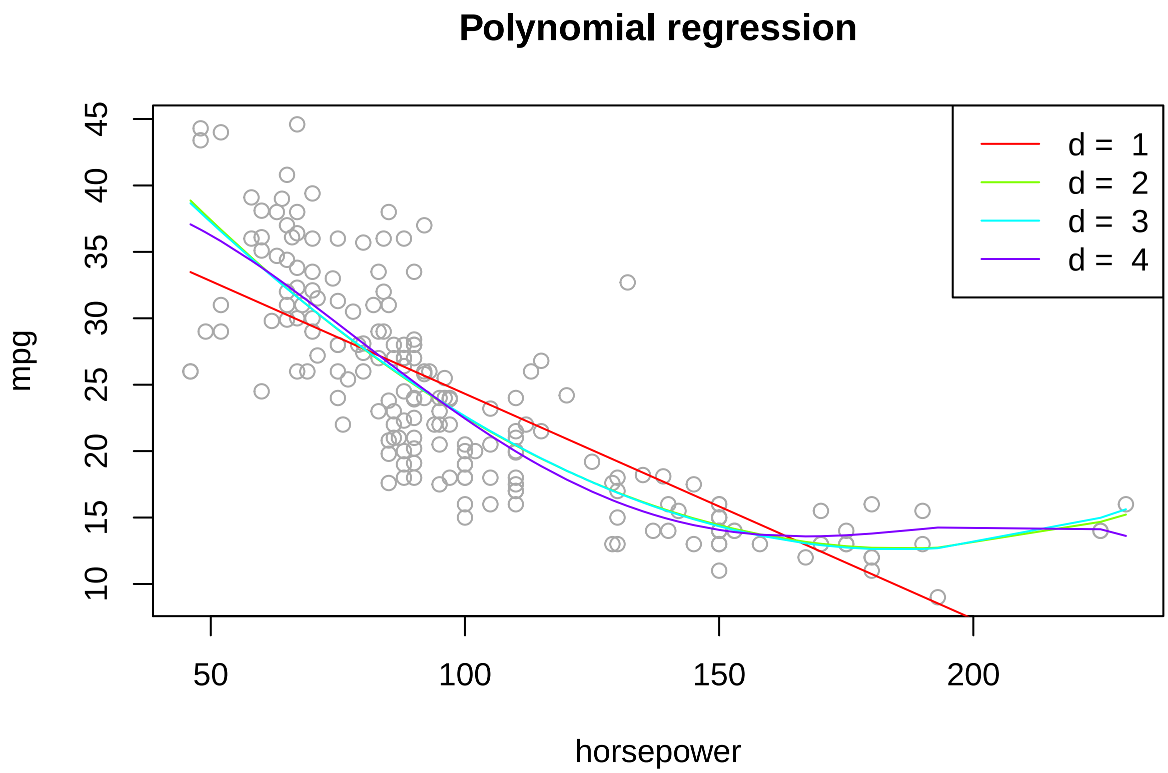
Statistical Learning V2021

alexaoh

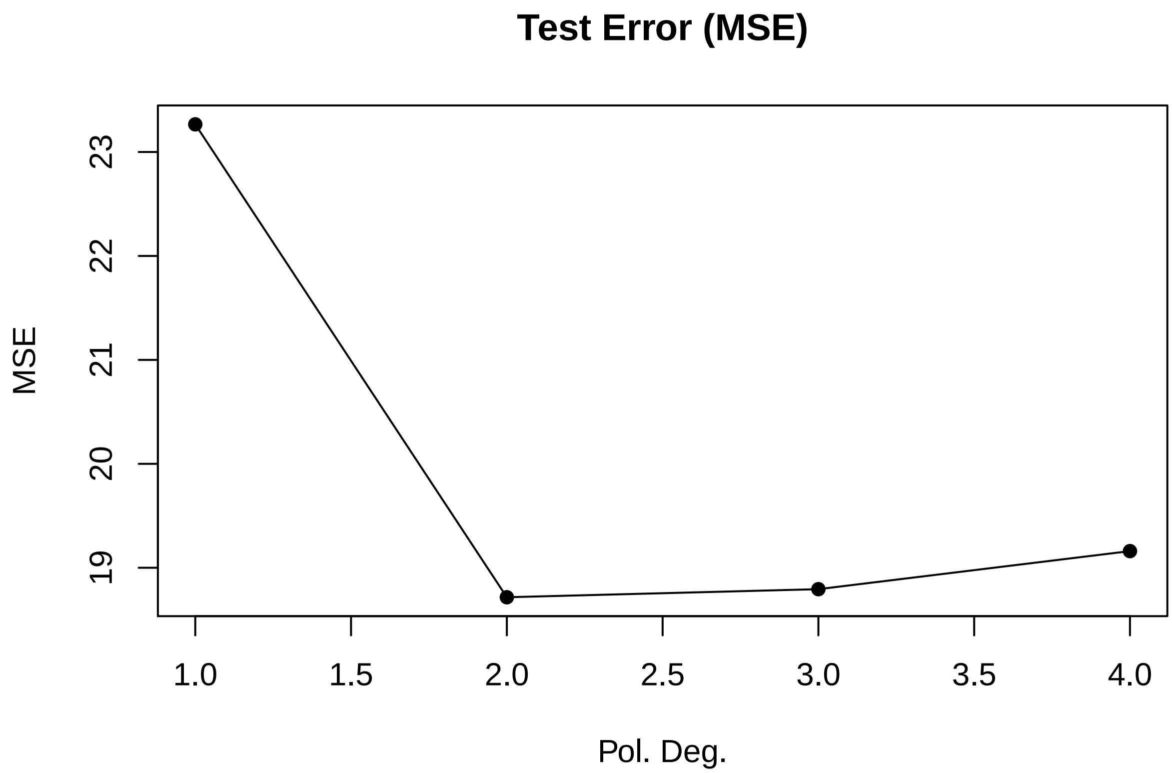
30 mars, 2021

Problem 1

```r
library(ISLR)
# extract only the two variables from Auto
ds = Auto[c("horsepower", "mpg")]
n = nrow(ds)
# which degrees we will look at
deg = 1:4
set.seed(1)
# training ids for training set
tr = sample.int(n = n, size = n/2)
# plot of training data
plot(ds[tr, ], col = "darkgrey", main = "Polynomial regression")

colors <- rainbow(n = length(deg))

MSE = sapply(deg, function(d) {
    fit <- lm(mpg ~ poly(horsepower, d), data = ds[tr, ])
    lines(sort(ds[tr, 1]), fit$fit[order(ds[tr, 1])], col = colors[d])

    return(mean((predict(fit, ds[-tr, ]) - ds[-tr, 2])^2))
})
legend("topright", legend = paste("d = ", deg), col = colors, lty = 1)
```

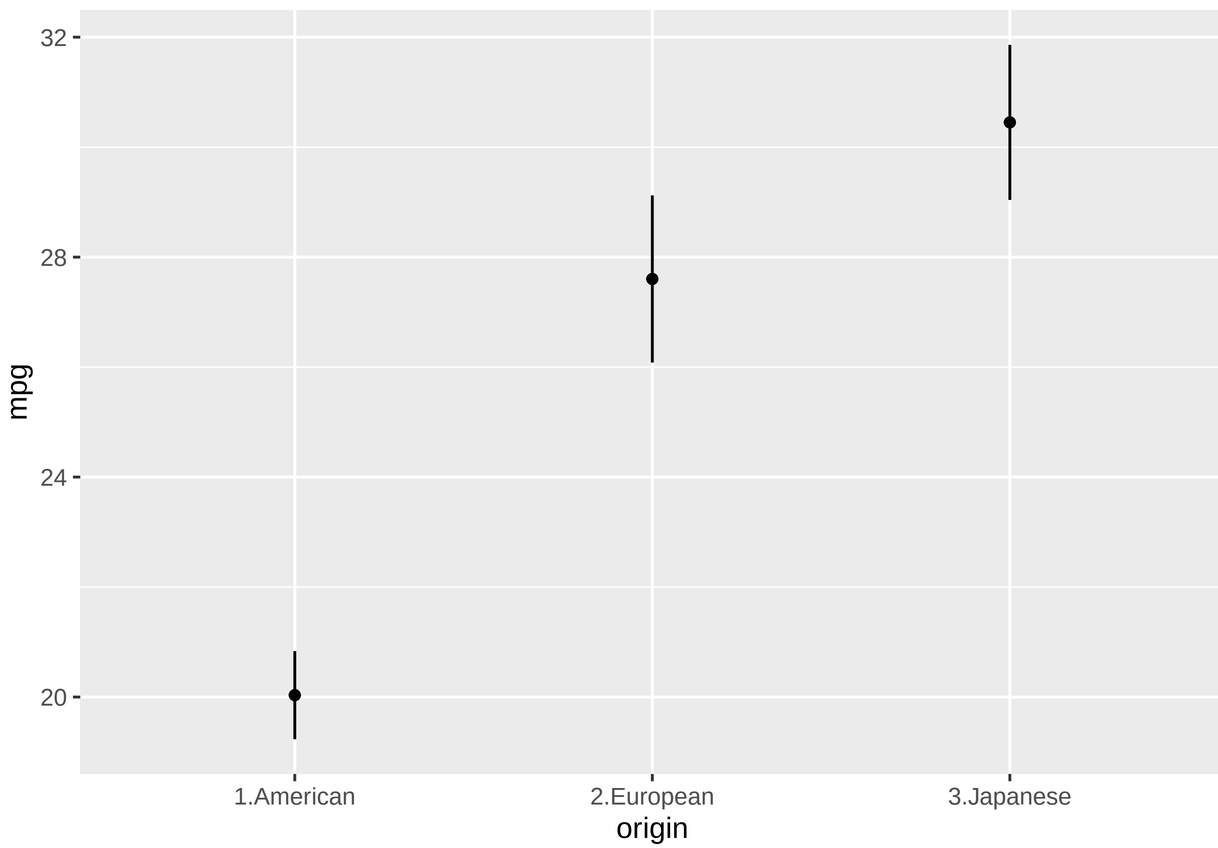



```
# Plot MSE
plot(MSE, type = "o", xlab = "Pol. Deg.", main = "Test Error (MSE)",
    pch = 16)
```



Problem 2

```
attach(Auto)
fit2 <- lm(mpg ~ factor(origin))
dframe <- data.frame(origin = as.factor(sort(unique(origin))))
pred <- predict(fit2, dframe, se = T)

# Data frame including CI (z_alpha/2 = 1.96).
dat <- data.frame(origin = dframe, mpg = pred$fit, lwr = pred$fit - 1.96 *
    pred$se.fit, upr = pred$fit + 1.96 * pred$se.fit)

# Plot the fitted/predicted values and CI
ggplot(dat, aes(x = origin, y = mpg)) + geom_point() + geom_segment(aes(x = origin,
    y = lwr, xend = origin, yend = upr)) + scale_x_discrete(labels = c(`1` = "1.American",
    `2` = "2.European", `3` = "3.Japanese"))
```



Problem 3

Now, let us look at the `Wage` data set. The section on Additive Models (slides 28-34 in the pdf) explains how we can create an AM by adding components together. One component we saw is a natural spline in `year` with one knot. Derive the expression for the design matrix $\mathbf{X}_2$ from the natural spline basis

$$b_1(x_i) = x_i, \quad b_{k+2}(x_i) = d_k(x_i) - d_K(x_i),\ k = 0, \ldots, K-1,$$
$$d_k(x_i) = \frac{(x_i - c_k)^3_+ - (x_i - c_{K+1})^3_+}{c_{K+1} - c_k}.$$

From the slides that are referenced to above, we know that the design matrix $\mathbf{X}_2$ is

$$\mathbf{X}_2 = \begin{pmatrix} x_{12} & \left[\frac{1}{6}(x_{12} - 2003)^3 - \frac{1}{3}(x_{12} - 2006)^3_+\right] \\ x_{22} & \left[\frac{1}{6}(x_{22} - 2003)^3 - \frac{1}{3}(x_{22} - 2006)^3_+\right] \\ \vdots & \vdots \\ x_{n2} & \left[\frac{1}{6}(x_{n2} - 2003)^3 - \frac{1}{3}(x_{n2} - 2006)^3_+\right] \end{pmatrix},$$

when having a knot at $c_1 = 2006$ and boundary knots at $c_0 = 2003$ and $c_2 = 2009$. The reason behind this matrix is given in the following.

Since we are using only one knot, we set $K = 1$. Moreover, the first column of the matrix is always given by the functions $b_1(x_i) = x_i$. Since $K = 1$, $k = 0$ is the only value that k takes. Hence,

$$
\begin{aligned}
b_2(x_i) = d_0(x_i) - d_1(x_i) &= \frac{(x_i - c_0)_+^3 - (x_i - c_2)_+^3}{c_2 - c_0} - \frac{(x_i - c_1)_+^3 - (x_i - c_2)_+^3}{c_2 - c_1} \\
&= \frac{(x_i - 2003)_+^3 - (x_i - 2009)_+^3}{6} - \frac{(x_i - 2006)_+^3 - (x_i - 2009)_+^3}{3} \\
&= \frac{1}{6}(x_i - 2003)_+^3 - \frac{1}{3}(x_i - 2006)_+^3 + \frac{1}{6}(x_i - 2009)_+^3.
\end{aligned}
$$

Furthermore, since $2003 \le x_i \le 2009$, $(x_i - 2009)_+^3 = 0$. Hence, $b_2(x_i) = \frac{1}{6}(x_i - 2003)_+^3 - \frac{1}{3}(x_i - 2006)_+^3$. Finally, the design matrix is constructed by setting

$$
\mathbf{X}_2 = \begin{pmatrix} b_1(x_1) & b_2(x_1) \\ b_1(x_2) & b_2(x_2) \\ \vdots & \vdots \\ b_1(x_n) & b_2(x_n) \end{pmatrix}.
$$

Problem 4

Continuation of Problem 3. Write code that produces $\mathbf{X}$.

```r
attach(Wage)
# X_1
mybs = function(x, knots) {
    cbind(x, x^2, x^3, sapply(knots, function(y) pmax(0, x - y)^3))
}
d = function(c, cK, x) (pmax(0, x - c)^3 - pmax(0, x - cK)^3)/(cK - c)
# X_2
myns = function(x, knots) {
    kn = c(min(x), knots, max(x))
    K = length(kn)
    sub = d(kn[K - 1], kn[K], x)
    cbind(x, sapply(kn[1:(K - 2)], d, kn[K], x) - sub)
}
# X_3
myfactor = function(x) model.matrix(~x)[, -1]

# Define the X-matrix below.

knots.age <- c(40, 60)
knot.year <- 2006

X <- cbind(1, mybs(age, knots.age), myns(year, knot.year), myfactor(education))

# fitted model with our X
myhat = lm(wage ~ X - 1)$fit
# fitted model with gam
yhat = gam(wage ~ bs(age, knots = c(40, 60)) + ns(year, knots = 2006) +
    education)$fit
# are they equal?
all.equal(myhat, yhat)
```

```
#> [1] TRUE
```

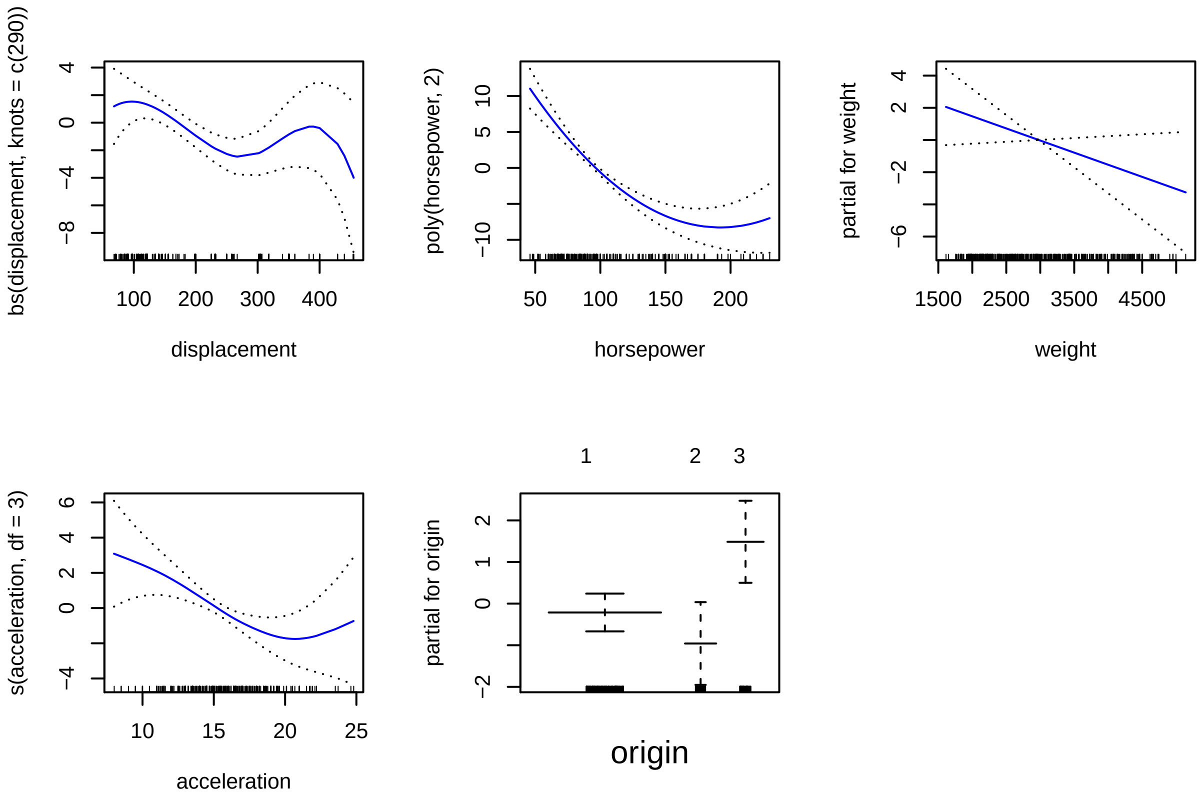
```r
# Yes, they are equal!
```

Problem 5

```r
Auto$origin <- as.factor(Auto$origin)
gamobject <- gam(mpg ~ bs(displacement, knots = c(290)) + poly(horsepower,
    2) + weight + s(acceleration, df = 3) + origin, data = Auto)
summary(gamobject)
```

```
#>
#> Call: gam(formula = mpg ~ bs(displacement, knots = c(290)) + poly(horsepower,
#>     2) + weight + s(acceleration, df = 3) + origin, data = Auto)
#> Deviance Residuals:
#>      Min       1Q   Median       3Q      Max
#> -11.5172  -2.3774  -0.2538   1.7982  15.9994
#>
#> (Dispersion Parameter for gaussian family taken to be 14.1747)
#>
#>     Null Deviance: 23818.99 on 391 degrees of freedom
#> Residual Deviance: 5372.203 on 378.9999 degrees of freedom
#> AIC: 2166.599
#>
#> Number of Local Scoring Iterations: NA
#>
#> Anova for Parametric Effects
#>                                   Df  Sum Sq Mean Sq  F value    Pr(>F)
#> bs(displacement, knots = c(290))   4 16705.2  4176.3 294.6301 < 2.2e-16 ***
#> poly(horsepower, 2)                2  1283.6   641.8  45.2786 < 2.2e-16 ***
#> weight                             1   318.9   318.9  22.4970 2.985e-06 ***
#> s(acceleration, df = 3)            1   128.1   128.1   9.0362 0.0028231 **
#> origin                             2   213.8   106.9   7.5422 0.0006137 ***
#> Residuals                        379  5372.2    14.2
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Anova for Nonparametric Effects
#>                                  Npar Df Npar F   Pr(F)
#> (Intercept)
#> bs(displacement, knots = c(290))
#> poly(horsepower, 2)
#> weight
#> s(acceleration, df = 3)                2 2.9111 0.05563 .
#> origin
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
par(mfrow = c(2, 3))
plot(gamobject, se = TRUE, col = "blue")
```


bs(displacement, knots = c(290))
displacement
100 200 300 400
poly(horsepower, 2)
horsepower
50 100 150 200
partial for weight
weight
1500 2500 3500 4500
s(acceleration, df = 3)
acceleration
10 15 20 25
partial for origin
1 2 3
origin