B27
Hector G. Flores Rodriguez

Stats 110 - HW6

November 22, 2017

4.7

```
library(leaps)
library(HH)
```

```
## Loading required package: lattice

## Loading required package: grid

## Loading required package: latticeExtra

## Loading required package: RColorBrewer

## Loading required package: multcomp

## Loading required package: mvtnorm

## Loading required package: survival

## Loading required package: TH.data

## Loading required package: MASS

##
## Attaching package: 'TH.data'

## The following object is masked from 'package:MASS':
##
##     geyser

## Loading required package: gridExtra
```

```
baseball_times = read.csv("../data/BaseballTimes.csv")
all = regsubsets(Time ~ Runs + Margin + Pitchers + Attendance,
                 data=baseball_times)
summaryHH(all)
```

```
##     model p   rsq  rss adjr2   cp   bic stderr
## 1       P 2 0.800 5984 0.784 2.88 -18.7   21.5
## 2     M-P 3 0.824 5271 0.794 3.22 -17.9   21.0
## 3   M-P-A 4 0.843 4686 0.800 3.87 -17.0   20.6
## 4 R-M-P-A 5 0.856 4312 0.798 5.00 -15.5   20.8
##
## Model variables with abbreviations
##                                   model
## P                              Pitchers
## M-P                     Margin-Pitchers
## M-P-A        Margin-Pitchers-Attendance
## R-M-P-A Runs-Margin-Pitchers-Attendance
##
## model with largest adjr2
## 3
```

```
##
## Number of observations
## 15
```

(a). Maximize R^2

model_a = Runs + Margin + Pitchers + Attendance

(b). Maximize R^2_{adj}

model_b = Margin + Pitchers + Attendance

(c). Minimize Mallow's C_p

model_c = Pitchers

(d). What model to choose to predict game times?

I would choose model_b to predict game times. The model captures most of the variance (relative to the other models) and $C_p = 3.22 < p + 1 = 4$, which is a model worth considering. Furthermore, the data only has 15 observations and if the problem is not sparse enough then having a lot of variables can result in overfitting. Lastly, I wouldn't blindly trust Mallow's C_p on such a small sample size.

Part 2

1. Use momheight and dadheight to predict Height. Show regression results

```
heights = read.table("../data/Hmwk6.txt", sep="\t", header=TRUE)
model = lm("Height ~ momheight + dadheight", data=heights)
summary(model)
```

```
##
## Call:
## lm(formula = "Height ~ momheight + dadheight", data = heights)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -10.4902  -1.2004  -0.1095   1.3077   6.1282
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 16.13189    7.11050   2.269  0.02628 *
## momheight    0.29072    0.10679   2.722  0.00813 **
## dadheight    0.50946    0.07609   6.696    4e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.565 on 72 degrees of freedom
## Multiple R-squared:  0.4972, Adjusted R-squared:  0.4832
## F-statistic:  35.6 on 2 and 72 DF,  p-value: 1.781e-11
```

```
heights$y_hat = predict(model)
head(heights$y_hat)
```

```
## [1] 67.85288 71.85660 68.94659 68.21839 72.07253 68.07162
```

2. Find case diagnostic values.

```
heights$rstandard = rstandard(model)
heights$rstudent = rstudent(model)
heights$hi = hatvalues(model)
heights$cooks = cooks.distance(model)
head(heights[1:5, c("rstandard", "rstudent", "hi", "cooks")])
```

```
##    rstandard   rstudent         hi        cooks
## 1  1.6405779  1.6604766 0.02900539 0.0267999738
## 2 -1.5386737 -1.5537093 0.04537483 0.0375105926
## 3 -0.3787467 -0.3764826 0.05082254 0.0025602684
## 4  0.1115052  0.1107377 0.03077860 0.0001316118
## 5 -0.8278256 -0.8259970 0.04755581 0.0114056625
```

3. Identify all of the cases tha need to be investigated using cretira for moderate and extreme values.

```
p = 2 + 1  # k + 1
n = nrow(heights)

high_stud = heights$rstudent[abs(heights$rstudent) > 3]
low_stud = heights$rstudent[abs(heights$rstudent) > 2]
flagH_stud = heights[match(high_stud, heights$rstudent),]
flagL_stud = heights[match(low_stud, heights$rstudent),]

high_lev = heights$hi[heights$hi > 3*p/n]
low_lev = heights$hi[heights$hi > 2*p/n]
flagH_lev = heights[match(high_lev, heights$hi),]
flagL_lev = heights[match(low_lev, heights$hi),]

high_cook = heights$cooks[heights$cooks > 1]
low_cook = heights$cooks[heights$cooks > 0.5]
flagH_cook = heights[match(high_cook, heights$cooks),]
flagL_cook = heights[match(low_cook, heights$cooks),]
```

4. Print ID, momheight, dadheight, height, and y__hat

Studentize residuals:

```
flagL_stud
```

```
##     ID  Sex momheight dadheight Height    y_hat rstandard  rstudent
## 20  36 Male        64        67     75 68.87180  2.410692  2.496775
## 22  38 Male        60        67     73 67.70893  2.094698  2.146535
## 59 131 Male        61        66     57 67.49019 -4.140052 -4.709851
##            hi      cooks
## 20 0.01802712 0.03556223
## 22 0.03046838 0.04596306
## 59 0.02439849 0.14288310
```

```
flagH_stud
```

```
##      ID  Sex momheight dadheight Height    y_hat rstandard  rstudent
## 59 131 Male        61        66     57 67.49019 -4.140052 -4.709851
##             hi     cooks
## 59 0.02439849 0.1428831
```

Leverage:

```
flagL_lev
```

```
##      ID  Sex momheight dadheight Height    y_hat   rstandard    rstudent
## 11  21 Male        54        68     68 66.47408  0.64829993  0.64566938
## 13  25 Male        59        60     64 63.85198  0.06028141  0.05986284
## 41  86 Male        66        55     65 63.33971  0.73423136  0.73185972
## 57 122 Male        60        78     70 73.31300 -1.38510690 -1.39415429
## 65 138 Male        71        76     77 75.49199  0.62579585  0.62313182
##            hi        cooks
## 11 0.15815743 0.0263202141
## 13 0.08380549 0.0001107976
## 41 0.22299854 0.0515732840
## 57 0.13064951 0.0961077083
## 65 0.11760773 0.0173987597
```

```
flagH_lev
```

```
##      ID  Sex momheight dadheight Height    y_hat  rstandard   rstudent
## 11  21 Male        54        68     68 66.47408  0.6482999  0.6456694
## 41  86 Male        66        55     65 63.33971  0.7342314  0.7318597
## 57 122 Male        60        78     70 73.31300 -1.3851069 -1.3941543
##           hi      cooks
## 11 0.1581574 0.02632021
## 41 0.2229985 0.05157328
## 57 0.1306495 0.09610771
```

Cooks:

```
flagL_cook
```

```
##  [1] ID        Sex       momheight dadheight Height    y_hat     rstandard
##  [8] rstudent  hi        cooks
## <0 rows> (or 0-length row.names)
```

```
flagH_cook
```

```
##  [1] ID        Sex       momheight dadheight Height    y_hat     rstandard
##  [8] rstudent  hi        cooks
## <0 rows> (or 0-length row.names)
```

5.

ID: 86. The momheight is 11 inches greater than dadheight. This is quite unusual for the sample given, resulting in predicted hight lower than the actual height, since the dadheight seems to be more significantly significant for predicting male height.

ID: 131. Both momheight and dadheight are reasonably similar. However the true height is shorter than both momheight and dadheight. The predicted height overestimated the actual height.

ID: 36. The true height is much taller than both dadheight and momheight. However the predicted height underestimated the true height by 6 inches.

ID: 122. The dadheigt is much greater than the momheight. However the true height lied approximately between the momheight and dadheight. The predicted height overestimated the true height since the dadheight was quite large.

6.

I would remove ID:86. The momheight being 11 inches greater than the dadheight suggests an outlier since the sample doesn't provide a similar case.