Homework 5

Xinyi Lin

11/24/2018

library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.2.1 ──

## ✔ ggplot2 3.1.0 ✔ purrr 0.2.5  
## ✔ tibble 1.4.2 ✔ dplyr 0.7.8  
## ✔ tidyr 0.8.2 ✔ stringr 1.3.1  
## ✔ readr 1.1.1 ✔ forcats 0.3.0

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(faraway)  
library(leaps)

## Input and tidy data

data(state)  
head(state.x77)

## Population Income Illiteracy Life Exp Murder HS Grad Frost  
## Alabama 3615 3624 2.1 69.05 15.1 41.3 20  
## Alaska 365 6315 1.5 69.31 11.3 66.7 152  
## Arizona 2212 4530 1.8 70.55 7.8 58.1 15  
## Arkansas 2110 3378 1.9 70.66 10.1 39.9 65  
## California 21198 5114 1.1 71.71 10.3 62.6 20  
## Colorado 2541 4884 0.7 72.06 6.8 63.9 166  
## Area  
## Alabama 50708  
## Alaska 566432  
## Arizona 113417  
## Arkansas 51945  
## California 156361  
## Colorado 103766

state\_clean\_df =  
 as.tibble(state.x77) %>%   
 janitor::clean\_names()  
  
state\_clean\_df

## # A tibble: 50 x 8  
## population income illiteracy life\_exp murder hs\_grad frost area  
## <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 3615 3624 2.1 69.0 15.1 41.3 20 50708  
## 2 365 6315 1.5 69.3 11.3 66.7 152 566432  
## 3 2212 4530 1.8 70.6 7.8 58.1 15 113417  
## 4 2110 3378 1.9 70.7 10.1 39.9 65 51945  
## 5 21198 5114 1.1 71.7 10.3 62.6 20 156361  
## 6 2541 4884 0.7 72.1 6.8 63.9 166 103766  
## 7 3100 5348 1.1 72.5 3.1 56 139 4862  
## 8 579 4809 0.9 70.1 6.2 54.6 103 1982  
## 9 8277 4815 1.3 70.7 10.7 52.6 11 54090  
## 10 4931 4091 2 68.5 13.9 40.6 60 58073  
## # ... with 40 more rows

## Question 1-a)

### Backward

all\_fit = lm(life\_exp ~ ., data = state\_clean\_df)  
step(all\_fit, direction='backward')

## Start: AIC=-22.18  
## life\_exp ~ population + income + illiteracy + murder + hs\_grad +   
## frost + area  
##   
## Df Sum of Sq RSS AIC  
## - area 1 0.0011 23.298 -24.182  
## - income 1 0.0044 23.302 -24.175  
## - illiteracy 1 0.0047 23.302 -24.174  
## <none> 23.297 -22.185  
## - population 1 1.7472 25.044 -20.569  
## - frost 1 1.8466 25.144 -20.371  
## - hs\_grad 1 2.4413 25.738 -19.202  
## - murder 1 23.1411 46.438 10.305  
##   
## Step: AIC=-24.18  
## life\_exp ~ population + income + illiteracy + murder + hs\_grad +   
## frost  
##   
## Df Sum of Sq RSS AIC  
## - illiteracy 1 0.0038 23.302 -26.174  
## - income 1 0.0059 23.304 -26.170  
## <none> 23.298 -24.182  
## - population 1 1.7599 25.058 -22.541  
## - frost 1 2.0488 25.347 -21.968  
## - hs\_grad 1 2.9804 26.279 -20.163  
## - murder 1 26.2721 49.570 11.569  
##   
## Step: AIC=-26.17  
## life\_exp ~ population + income + murder + hs\_grad + frost  
##   
## Df Sum of Sq RSS AIC  
## - income 1 0.006 23.308 -28.161  
## <none> 23.302 -26.174  
## - population 1 1.887 25.189 -24.280  
## - frost 1 3.037 26.339 -22.048  
## - hs\_grad 1 3.495 26.797 -21.187  
## - murder 1 34.739 58.041 17.456  
##   
## Step: AIC=-28.16  
## life\_exp ~ population + murder + hs\_grad + frost  
##   
## Df Sum of Sq RSS AIC  
## <none> 23.308 -28.161  
## - population 1 2.064 25.372 -25.920  
## - frost 1 3.122 26.430 -23.877  
## - hs\_grad 1 5.112 28.420 -20.246  
## - murder 1 34.816 58.124 15.528

##   
## Call:  
## lm(formula = life\_exp ~ population + murder + hs\_grad + frost,   
## data = state\_clean\_df)  
##   
## Coefficients:  
## (Intercept) population murder hs\_grad frost   
## 7.103e+01 5.014e-05 -3.001e-01 4.658e-02 -5.943e-03

### Forward

start\_fit = lm(life\_exp ~ 1, data = state\_clean\_df)  
step(start\_fit, direction = 'forward', scope = list(upper = all\_fit, lower = start\_fit))

## Start: AIC=30.44  
## life\_exp ~ 1  
##   
## Df Sum of Sq RSS AIC  
## + murder 1 53.838 34.461 -14.609  
## + illiteracy 1 30.578 57.721 11.179  
## + hs\_grad 1 29.931 58.368 11.737  
## + income 1 10.223 78.076 26.283  
## + frost 1 6.064 82.235 28.878  
## <none> 88.299 30.435  
## + area 1 1.017 87.282 31.856  
## + population 1 0.409 87.890 32.203  
##   
## Step: AIC=-14.61  
## life\_exp ~ murder  
##   
## Df Sum of Sq RSS AIC  
## + hs\_grad 1 4.6910 29.770 -19.925  
## + population 1 4.0161 30.445 -18.805  
## + frost 1 3.1346 31.327 -17.378  
## + income 1 2.4047 32.057 -16.226  
## <none> 34.461 -14.609  
## + area 1 0.4697 33.992 -13.295  
## + illiteracy 1 0.2732 34.188 -13.007  
##   
## Step: AIC=-19.93  
## life\_exp ~ murder + hs\_grad  
##   
## Df Sum of Sq RSS AIC  
## + frost 1 4.3987 25.372 -25.920  
## + population 1 3.3405 26.430 -23.877  
## <none> 29.770 -19.925  
## + illiteracy 1 0.4419 29.328 -18.673  
## + area 1 0.2775 29.493 -18.394  
## + income 1 0.1022 29.668 -18.097  
##   
## Step: AIC=-25.92  
## life\_exp ~ murder + hs\_grad + frost  
##   
## Df Sum of Sq RSS AIC  
## + population 1 2.06358 23.308 -28.161  
## <none> 25.372 -25.920  
## + income 1 0.18232 25.189 -24.280  
## + illiteracy 1 0.17184 25.200 -24.259  
## + area 1 0.02573 25.346 -23.970  
##   
## Step: AIC=-28.16  
## life\_exp ~ murder + hs\_grad + frost + population  
##   
## Df Sum of Sq RSS AIC  
## <none> 23.308 -28.161  
## + income 1 0.0060582 23.302 -26.174  
## + illiteracy 1 0.0039221 23.304 -26.170  
## + area 1 0.0007900 23.307 -26.163

##   
## Call:  
## lm(formula = life\_exp ~ murder + hs\_grad + frost + population,   
## data = state\_clean\_df)  
##   
## Coefficients:  
## (Intercept) murder hs\_grad frost population   
## 7.103e+01 -3.001e-01 4.658e-02 -5.943e-03 5.014e-05

### Stepwise

all\_fit = lm(life\_exp ~ ., data = state\_clean\_df)  
step(all\_fit, direction = 'both')

## Start: AIC=-22.18  
## life\_exp ~ population + income + illiteracy + murder + hs\_grad +   
## frost + area  
##   
## Df Sum of Sq RSS AIC  
## - area 1 0.0011 23.298 -24.182  
## - income 1 0.0044 23.302 -24.175  
## - illiteracy 1 0.0047 23.302 -24.174  
## <none> 23.297 -22.185  
## - population 1 1.7472 25.044 -20.569  
## - frost 1 1.8466 25.144 -20.371  
## - hs\_grad 1 2.4413 25.738 -19.202  
## - murder 1 23.1411 46.438 10.305  
##   
## Step: AIC=-24.18  
## life\_exp ~ population + income + illiteracy + murder + hs\_grad +   
## frost  
##   
## Df Sum of Sq RSS AIC  
## - illiteracy 1 0.0038 23.302 -26.174  
## - income 1 0.0059 23.304 -26.170  
## <none> 23.298 -24.182  
## - population 1 1.7599 25.058 -22.541  
## + area 1 0.0011 23.297 -22.185  
## - frost 1 2.0488 25.347 -21.968  
## - hs\_grad 1 2.9804 26.279 -20.163  
## - murder 1 26.2721 49.570 11.569  
##   
## Step: AIC=-26.17  
## life\_exp ~ population + income + murder + hs\_grad + frost  
##   
## Df Sum of Sq RSS AIC  
## - income 1 0.006 23.308 -28.161  
## <none> 23.302 -26.174  
## - population 1 1.887 25.189 -24.280  
## + illiteracy 1 0.004 23.298 -24.182  
## + area 1 0.000 23.302 -24.174  
## - frost 1 3.037 26.339 -22.048  
## - hs\_grad 1 3.495 26.797 -21.187  
## - murder 1 34.739 58.041 17.456  
##   
## Step: AIC=-28.16  
## life\_exp ~ population + murder + hs\_grad + frost  
##   
## Df Sum of Sq RSS AIC  
## <none> 23.308 -28.161  
## + income 1 0.006 23.302 -26.174  
## + illiteracy 1 0.004 23.304 -26.170  
## + area 1 0.001 23.307 -26.163  
## - population 1 2.064 25.372 -25.920  
## - frost 1 3.122 26.430 -23.877  
## - hs\_grad 1 5.112 28.420 -20.246  
## - murder 1 34.816 58.124 15.528

##   
## Call:  
## lm(formula = life\_exp ~ population + murder + hs\_grad + frost,   
## data = state\_clean\_df)  
##   
## Coefficients:  
## (Intercept) population murder hs\_grad frost   
## 7.103e+01 5.014e-05 -3.001e-01 4.658e-02 -5.943e-03

According to the results, when using three methods, we get same ‘best subset’ which ‘population, murder, hs\_grad and frost’.

## Question 1-b)

fitted\_model = lm(formula = life\_exp ~ population + murder + hs\_grad + frost,   
 data = state\_clean\_df)  
  
summary(fitted\_model)

##   
## Call:  
## lm(formula = life\_exp ~ population + murder + hs\_grad + frost,   
## data = state\_clean\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.47095 -0.53464 -0.03701 0.57621 1.50683   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.103e+01 9.529e-01 74.542 < 2e-16 \*\*\*  
## population 5.014e-05 2.512e-05 1.996 0.05201 .   
## murder -3.001e-01 3.661e-02 -8.199 1.77e-10 \*\*\*  
## hs\_grad 4.658e-02 1.483e-02 3.142 0.00297 \*\*   
## frost -5.943e-03 2.421e-03 -2.455 0.01802 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7197 on 45 degrees of freedom  
## Multiple R-squared: 0.736, Adjusted R-squared: 0.7126   
## F-statistic: 31.37 on 4 and 45 DF, p-value: 1.696e-12

cor(state\_clean\_df)

## population income illiteracy life\_exp murder  
## population 1.00000000 0.2082276 0.10762237 -0.06805195 0.3436428  
## income 0.20822756 1.0000000 -0.43707519 0.34025534 -0.2300776  
## illiteracy 0.10762237 -0.4370752 1.00000000 -0.58847793 0.7029752  
## life\_exp -0.06805195 0.3402553 -0.58847793 1.00000000 -0.7808458  
## murder 0.34364275 -0.2300776 0.70297520 -0.78084575 1.0000000  
## hs\_grad -0.09848975 0.6199323 -0.65718861 0.58221620 -0.4879710  
## frost -0.33215245 0.2262822 -0.67194697 0.26206801 -0.5388834  
## area 0.02254384 0.3633154 0.07726113 -0.10733194 0.2283902  
## hs\_grad frost area  
## population -0.09848975 -0.3321525 0.02254384  
## income 0.61993232 0.2262822 0.36331544  
## illiteracy -0.65718861 -0.6719470 0.07726113  
## life\_exp 0.58221620 0.2620680 -0.10733194  
## murder -0.48797102 -0.5388834 0.22839021  
## hs\_grad 1.00000000 0.3667797 0.33354187  
## frost 0.36677970 1.0000000 0.05922910  
## area 0.33354187 0.0592291 1.00000000

According to the summary results, we can find the p-value of population variable is slightly bigger than 0.05 which is a close call, so we try to compare models with and without population variable.

fitted\_less\_model = lm(formula = life\_exp ~ murder + hs\_grad + frost,   
 data = state\_clean\_df)  
  
summary(fitted\_less\_model)

##   
## Call:  
## lm(formula = life\_exp ~ murder + hs\_grad + frost, data = state\_clean\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.5015 -0.5391 0.1014 0.5921 1.2268   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 71.036379 0.983262 72.246 < 2e-16 \*\*\*  
## murder -0.283065 0.036731 -7.706 8.04e-10 \*\*\*  
## hs\_grad 0.049949 0.015201 3.286 0.00195 \*\*   
## frost -0.006912 0.002447 -2.824 0.00699 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7427 on 46 degrees of freedom  
## Multiple R-squared: 0.7127, Adjusted R-squared: 0.6939   
## F-statistic: 38.03 on 3 and 46 DF, p-value: 1.634e-12

anova(fitted\_model, fitted\_less\_model)

## Analysis of Variance Table  
##   
## Model 1: life\_exp ~ population + murder + hs\_grad + frost  
## Model 2: life\_exp ~ murder + hs\_grad + frost  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 45 23.308   
## 2 46 25.372 -1 -2.0636 3.9841 0.05201 .  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

The adjuested r-square of the model without ‘population’ is slightly less than the adjuested r-square of the model with ‘population’ and AIC of the model with ‘population’ also perform better, so keeping the population variable is a better choice.

## Question 1-c)

add\_illiteracy\_model = lm(formula = life\_exp ~ murder + hs\_grad + frost + illiteracy,   
 data = state\_clean\_df)  
  
summary(add\_illiteracy\_model)

##   
## Call:  
## lm(formula = life\_exp ~ murder + hs\_grad + frost + illiteracy,   
## data = state\_clean\_df)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1.48906 -0.51040 0.09793 0.55193 1.33480   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 71.519958 1.320487 54.162 < 2e-16 \*\*\*  
## murder -0.273118 0.041138 -6.639 3.5e-08 \*\*\*  
## hs\_grad 0.044970 0.017759 2.532 0.01490 \*   
## frost -0.007678 0.002828 -2.715 0.00936 \*\*   
## illiteracy -0.181608 0.327846 -0.554 0.58236   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.7483 on 45 degrees of freedom  
## Multiple R-squared: 0.7146, Adjusted R-squared: 0.6892   
## F-statistic: 28.17 on 4 and 45 DF, p-value: 9.547e-12

cor(state\_clean\_df)

## population income illiteracy life\_exp murder  
## population 1.00000000 0.2082276 0.10762237 -0.06805195 0.3436428  
## income 0.20822756 1.0000000 -0.43707519 0.34025534 -0.2300776  
## illiteracy 0.10762237 -0.4370752 1.00000000 -0.58847793 0.7029752  
## life\_exp -0.06805195 0.3402553 -0.58847793 1.00000000 -0.7808458  
## murder 0.34364275 -0.2300776 0.70297520 -0.78084575 1.0000000  
## hs\_grad -0.09848975 0.6199323 -0.65718861 0.58221620 -0.4879710  
## frost -0.33215245 0.2262822 -0.67194697 0.26206801 -0.5388834  
## area 0.02254384 0.3633154 0.07726113 -0.10733194 0.2283902  
## hs\_grad frost area  
## population -0.09848975 -0.3321525 0.02254384  
## income 0.61993232 0.2262822 0.36331544  
## illiteracy -0.65718861 -0.6719470 0.07726113  
## life\_exp 0.58221620 0.2620680 -0.10733194  
## murder -0.48797102 -0.5388834 0.22839021  
## hs\_grad 1.00000000 0.3667797 0.33354187  
## frost 0.36677970 1.0000000 0.05922910  
## area 0.33354187 0.0592291 1.00000000

The correlation of hs\_grad and illiteracy is -0.657 and when adding illiteracy in model, the coefficient of hs\_grad change slightly, thus there are low association between hs\_grad and illiteracy and my subset only contain hs\_grad.

## Question 3

state\_criterion\_df =  
 state\_clean\_df %>%   
 as.data.frame() %>%   
 select(life\_exp, everything())  
  
# Printing the 2 best models of each size, using the Cp criterion:  
leaps(x = state\_criterion\_df[,2:8], y = state\_criterion\_df[,1], nbest = 1, method = "Cp")

## $which  
## 1 2 3 4 5 6 7  
## 1 FALSE FALSE FALSE TRUE FALSE FALSE FALSE  
## 2 FALSE FALSE FALSE TRUE TRUE FALSE FALSE  
## 3 FALSE FALSE FALSE TRUE TRUE TRUE FALSE  
## 4 TRUE FALSE FALSE TRUE TRUE TRUE FALSE  
## 5 TRUE TRUE FALSE TRUE TRUE TRUE FALSE  
## 6 TRUE TRUE TRUE TRUE TRUE TRUE FALSE  
## 7 TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
##   
## $label  
## [1] "(Intercept)" "1" "2" "3" "4"   
## [6] "5" "6" "7"   
##   
## $size  
## [1] 2 3 4 5 6 7 8  
##   
## $Cp  
## [1] 16.126760 9.669894 3.739878 2.019659 4.008737 6.001959 8.000000

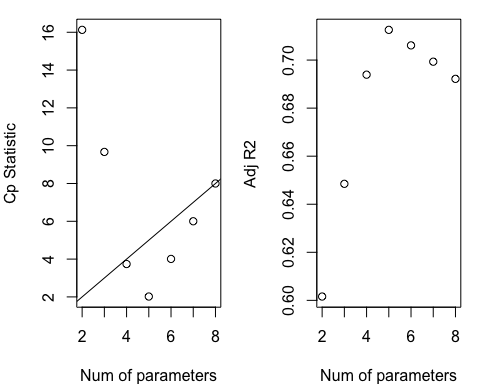
# Printing the 2 best models of each size, using the adjusted R^2 criterion:  
leaps(x = state\_criterion\_df[,2:8], y = state\_criterion\_df[,1], nbest = 1, method = "adjr2")

## $which  
## 1 2 3 4 5 6 7  
## 1 FALSE FALSE FALSE TRUE FALSE FALSE FALSE  
## 2 FALSE FALSE FALSE TRUE TRUE FALSE FALSE  
## 3 FALSE FALSE FALSE TRUE TRUE TRUE FALSE  
## 4 TRUE FALSE FALSE TRUE TRUE TRUE FALSE  
## 5 TRUE TRUE FALSE TRUE TRUE TRUE FALSE  
## 6 TRUE TRUE TRUE TRUE TRUE TRUE FALSE  
## 7 TRUE TRUE TRUE TRUE TRUE TRUE TRUE  
##   
## $label  
## [1] "(Intercept)" "1" "2" "3" "4"   
## [6] "5" "6" "7"   
##   
## $size  
## [1] 2 3 4 5 6 7 8  
##   
## $adjr2  
## [1] 0.6015893 0.6484991 0.6939230 0.7125690 0.7061129 0.6993268 0.6921823

# Summary of models for each size (one model per size)  
b = regsubsets(life\_exp ~ ., data = state\_criterion\_df)  
 (rs = summary(b))

## Subset selection object  
## Call: regsubsets.formula(life\_exp ~ ., data = state\_criterion\_df)  
## 7 Variables (and intercept)  
## Forced in Forced out  
## population FALSE FALSE  
## income FALSE FALSE  
## illiteracy FALSE FALSE  
## murder FALSE FALSE  
## hs\_grad FALSE FALSE  
## frost FALSE FALSE  
## area FALSE FALSE  
## 1 subsets of each size up to 7  
## Selection Algorithm: exhaustive  
## population income illiteracy murder hs\_grad frost area  
## 1 ( 1 ) " " " " " " "\*" " " " " " "   
## 2 ( 1 ) " " " " " " "\*" "\*" " " " "   
## 3 ( 1 ) " " " " " " "\*" "\*" "\*" " "   
## 4 ( 1 ) "\*" " " " " "\*" "\*" "\*" " "   
## 5 ( 1 ) "\*" "\*" " " "\*" "\*" "\*" " "   
## 6 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" " "   
## 7 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*"

# Plots of Cp and Adj-R2 as functions of parameters  
par(mar = c(4,4,1,1))  
par(mfrow = c(1,2))  
  
plot(2:(length(rs$cp) + 1), rs$cp, xlab = "Num of parameters", ylab = "Cp Statistic")  
abline(0,1)  
  
plot(2:(length(rs$cp) + 1), rs$adjr2, xlab = "Num of parameters", ylab = "Adj R2")



According to the Cp and adjusted r-square results, number of parameters are 4 to 8 are better models, so we count AIC and BIC of these models.

# AIC of the 3-predictor model:  
fitted\_4\_model <- lm(life\_exp ~ murder + hs\_grad + frost, data = state\_criterion\_df)  
AIC(fitted\_4\_model)

## [1] 117.9743

# BIC  
AIC(fitted\_4\_model, k = log(length(state\_criterion\_df$life\_exp)))

## [1] 127.5344

# AIC of the 4-predictor model:  
fitted\_5\_model <- lm(life\_exp ~ murder + hs\_grad + frost + population, data = state\_criterion\_df)  
AIC(fitted\_5\_model)

## [1] 115.7326

# BIC  
AIC(fitted\_5\_model, k = log(length(state\_criterion\_df$life\_exp)))

## [1] 127.2048

# AIC of the 5-predictor model:  
fitted\_6\_model <- lm(life\_exp ~ murder + hs\_grad + frost + population + income, data = state\_criterion\_df)  
AIC(fitted\_6\_model)

## [1] 117.7196

# BIC  
AIC(fitted\_6\_model, k = log(length(state\_criterion\_df$life\_exp)))

## [1] 131.1038

# AIC of the 6-predictor model:  
fitted\_7\_model <- lm(life\_exp ~ murder + hs\_grad + frost + population + income + illiteracy, data = state\_criterion\_df)  
AIC(fitted\_7\_model)

## [1] 119.7116

# BIC  
AIC(fitted\_7\_model, k = log(length(state\_criterion\_df$life\_exp)))

## [1] 135.0077

# AIC of the 7-predictor model:  
fitted\_8\_model <- lm(life\_exp ~ murder + hs\_grad + frost + population + income + illiteracy + area, data = state\_criterion\_df)  
AIC(fitted\_8\_model)

## [1] 121.7092

# BIC  
AIC(fitted\_8\_model, k = log(length(state\_criterion\_df$life\_exp)))

## [1] 138.9174

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| models | model 1 | model 2 | model 3 | model 4 | model 5 |
| number of parameters | p = 4 | p = 5 | p = 6 | p = 7 | p = 8 |
| Cp | 3.7399 | 2.0197 | 4.0087 | 6.0020 | 8.0000 |
| adjusted r-square | 0.6939 | 0.7126 | 0.7061 | 0.6993 | 0.6922 |
| AIC | 117.974 | 115.733 | 117.720 | 119.712 | 121.709 |
| BIC | 127.534 | 127.205 | 131.104 | 135.008 | 138.917 |

# How do the 6- and 4-predictors models compare in terms of AIC, R-adj, Cp?  
  
#############################################################################  
# A more compact way to look at the test-based results #  
#############################################################################  
  
best <- function(model, ...)   
{  
 subsets <- regsubsets(formula(model), model.frame(model), ...)  
 subsets <- with(summary(subsets),  
 cbind(p = as.numeric(rownames(which)), which, rss, rsq, adjr2, cp, bic))  
   
 return(subsets)  
}