STA 4320 CHAP 6.1.1

Prof. He Jiang

```r
require(ISLR2) # Hitters dataset
```

```
## Loading required package: ISLR2
```

```r
require(leaps) # subset selection
```

```
## Loading required package: leaps
```

Hitters dataset and NA terms

The Hitters dataset consists of Major League Baseball data from the 1986 and 1987 seasons.

```r
head(Hitters)
```

```
##                   AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun
## -Andy Allanson      293   66     1   30  29    14     1    293    66      1
## -Alan Ashby         315   81     7   24  38    39    14   3449   835     69
## -Alvin Davis        479  130    18   66  72    76     3   1624   457     63
## -Andre Dawson       496  141    20   65  78    37    11   5628  1575    225
## -Andres Galarraga   321   87    10   39  42    30     2    396   101     12
## -Alfredo Griffin    594  169     4   74  51    35    11   4408  1133     19
##                   CRuns CRBI CWalks League Division PutOuts Assists Errors
## -Andy Allanson       30   29     14      A        E     446      33     20
## -Alan Ashby         321  414    375      N        W     632      43     10
## -Alvin Davis        224  266    263      A        W     880      82     14
## -Andre Dawson       828  838    354      N        E     200      11      3
## -Andres Galarraga    48   46     33      N        E     805      40      4
## -Alfredo Griffin    501  336    194      A        W     282     421     25
##                   Salary NewLeague
## -Andy Allanson        NA         A
## -Alan Ashby        475.0         N
## -Alvin Davis       480.0         A
## -Andre Dawson      500.0         N
## -Andres Galarraga   91.5         N
## -Alfredo Griffin   750.0         A
```

There are NA terms here.

```r
any(is.na(Hitters))
```

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

We can remove rows with NA terms.

```r
dat = na.omit(Hitters)
any(is.na(dat))
```

```
## [1] FALSE
```

Best subset selection

Best subset selection (best is according to the RSS) with default up to 8 variables.

```r
# regsubsets is from the leaps package
sub_sel = regsubsets(Salary ~ ., data = dat)
summary(sub_sel)
```

```
## Subset selection object
## Call: regsubsets.formula(Salary ~ ., data = dat)
## 19 Variables  (and intercept)
##            Forced in Forced out
## AtBat          FALSE      FALSE
## Hits           FALSE      FALSE
## HmRun          FALSE      FALSE
## Runs           FALSE      FALSE
## RBI            FALSE      FALSE
## Walks          FALSE      FALSE
## Years          FALSE      FALSE
## CAtBat         FALSE      FALSE
## CHits          FALSE      FALSE
## CHmRun         FALSE      FALSE
## CRuns          FALSE      FALSE
## CRBI           FALSE      FALSE
## CWalks         FALSE      FALSE
## LeagueN        FALSE      FALSE
## DivisionW      FALSE      FALSE
## PutOuts        FALSE      FALSE
## Assists        FALSE      FALSE
## Errors         FALSE      FALSE
## NewLeagueN     FALSE      FALSE
## 1 subsets of each size up to 8
## Selection Algorithm: exhaustive
##          AtBat Hits HmRun Runs RBI Walks Years CAtBat CHits CHmRun CRuns CRBI
## 1  ( 1 ) " "   " "  " "   " "  " " " "   " "   " "    " "   " "    " "   "*"
## 2  ( 1 ) " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "   "*"
## 3  ( 1 ) " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "   "*"
## 4  ( 1 ) " "   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "   "*"
## 5  ( 1 ) "*"   "*"  " "   " "  " " " "   " "   " "    " "   " "    " "   "*"
## 6  ( 1 ) "*"   "*"  " "   " "  " " "*"   " "   " "    " "   " "    " "   "*"
## 7  ( 1 ) " "   "*"  " "   " "  " " "*"   " "   "*"    "*"   "*"    " "   " "
## 8  ( 1 ) "*"   "*"  " "   " "  " " "*"   " "   " "    " "   "*"    "*"   " "
##          CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN
## 1  ( 1 ) " "    " "     " "       " "     " "     " "    " "
## 2  ( 1 ) " "    " "     " "       " "     " "     " "    " "
## 3  ( 1 ) " "    " "     " "       "*"     " "     " "    " "
## 4  ( 1 ) " "    " "     "*"       "*"     " "     " "    " "
## 5  ( 1 ) " "    " "     "*"       "*"     " "     " "    " "
## 6  ( 1 ) " "    " "     "*"       "*"     " "     " "    " "
## 7  ( 1 ) " "    " "     "*"       "*"     " "     " "    " "
## 8  ( 1 ) "*"    " "     "*"       "*"     " "     " "    " "
```

Best subset selection can handle any amount of variables.

```
# regsubsets is from the leaps package
sub_sel = regsubsets(Salary ~ ., data = dat, nvmax = 19)
sub_res = summary(sub_sel)
```

We can see more results from the best subset selection. For example, given the number of variables (p) we want to keep, we select that row to see which variables are included.

```
sub_res$which[3,]
```

```
## (Intercept)       AtBat        Hits       HmRun        Runs         RBI
##        TRUE       FALSE        TRUE       FALSE       FALSE       FALSE
##       Walks       Years      CAtBat       CHits      CHmRun       CRuns
##       FALSE       FALSE       FALSE       FALSE       FALSE       FALSE
##        CRBI      CWalks     LeagueN   DivisionW     PutOuts     Assists
##        TRUE       FALSE       FALSE       FALSE        TRUE       FALSE
##      Errors  NewLeagueN
##       FALSE       FALSE
```

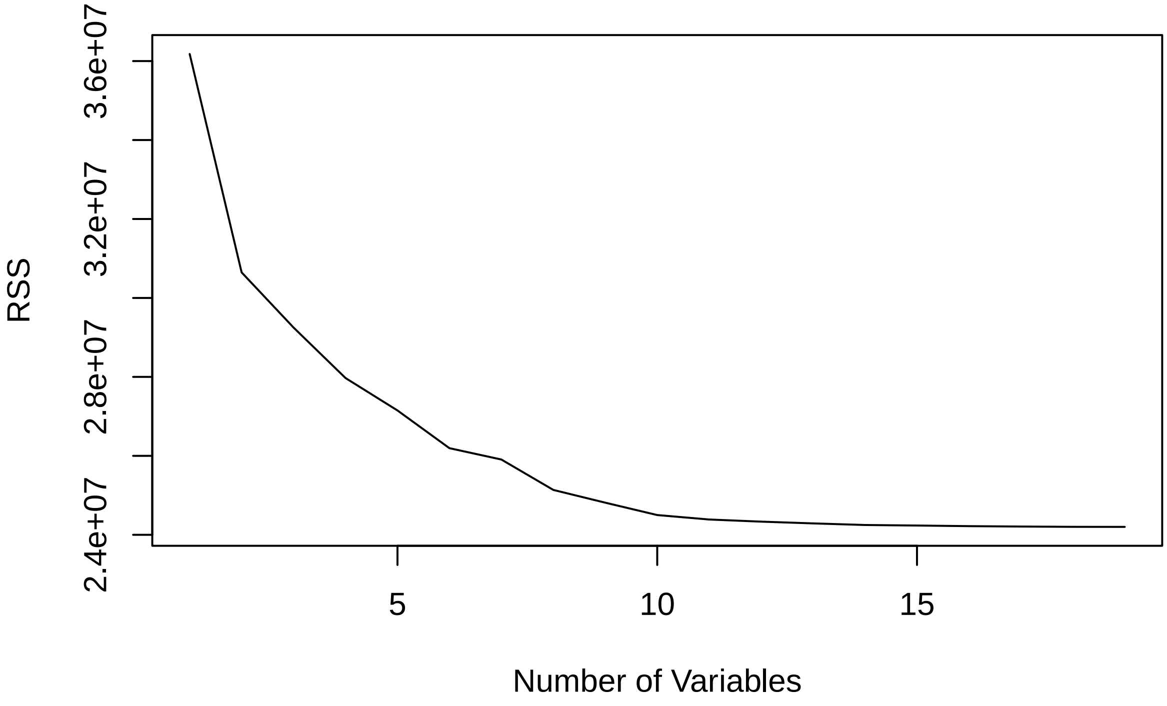
We can also see R2. Note that R2 always increases as we add more variables.

```
sub_res$rsq
```

```
##  [1] 0.3214501 0.4252237 0.4514294 0.4754067 0.4908036 0.5087146 0.5141227
##  [8] 0.5285569 0.5346124 0.5404950 0.5426153 0.5436302 0.5444570 0.5452164
## [15] 0.5454692 0.5457656 0.5459518 0.5460945 0.5461159
```

Residual squared error vs number of variables.

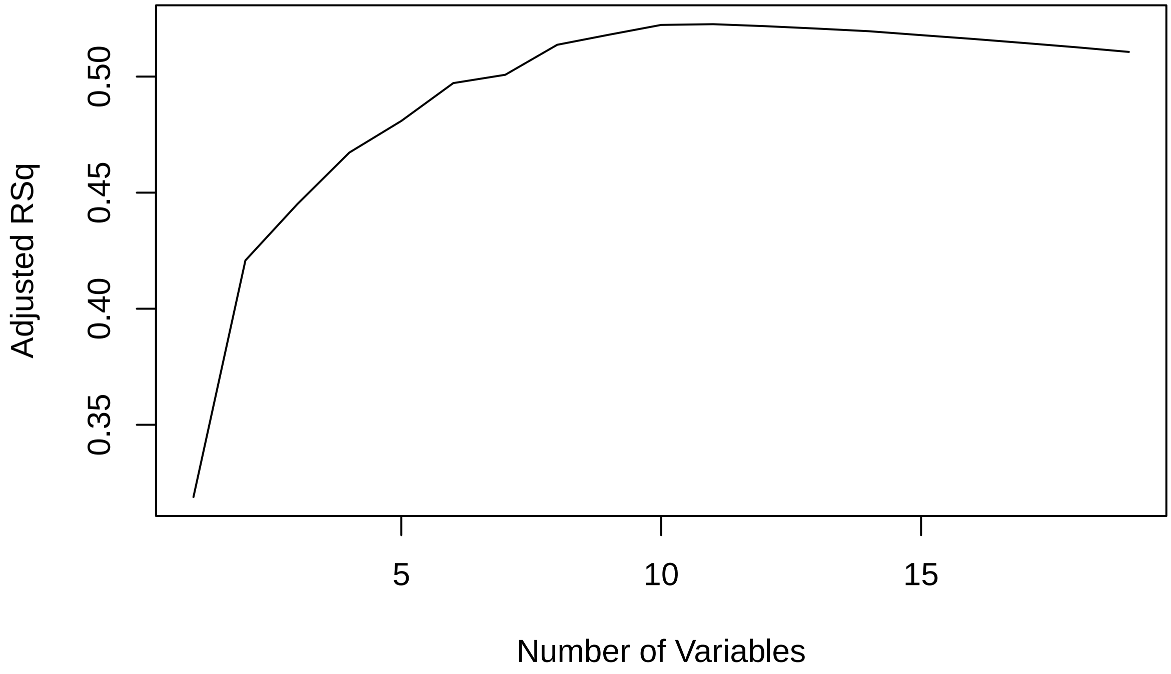
```
plot(sub_res$rss, xlab = "Number of Variables", ylab = "RSS", type = "l")
```



```
# type = "l" connects the dots
```

Adjusted R2 vs number of variables.

```r
plot(sub_res$adjr2, xlab = "Number of Variables", ylab = "Adjusted RSq", type = "l")
```

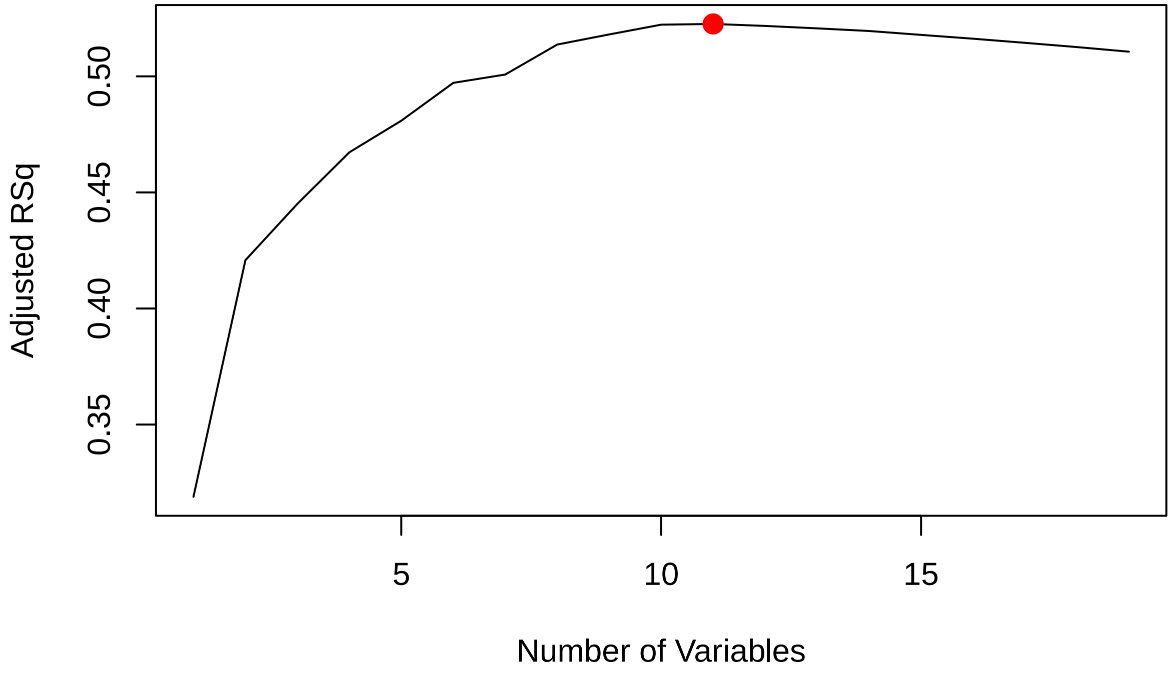


To see the number of variables leading to the highest adjusted R2:

```r
which.max(sub_res$adjr2)
```

```
## [1] 11
```

```r
# the red dot indicates the highest adjusted R2
plot(sub_res$adjr2, xlab = "Number of Variables", ylab = "Adjusted RSq", type = "l")
points(11, sub_res$adjr2[11], col = "red", cex = 2, pch = 20)
```

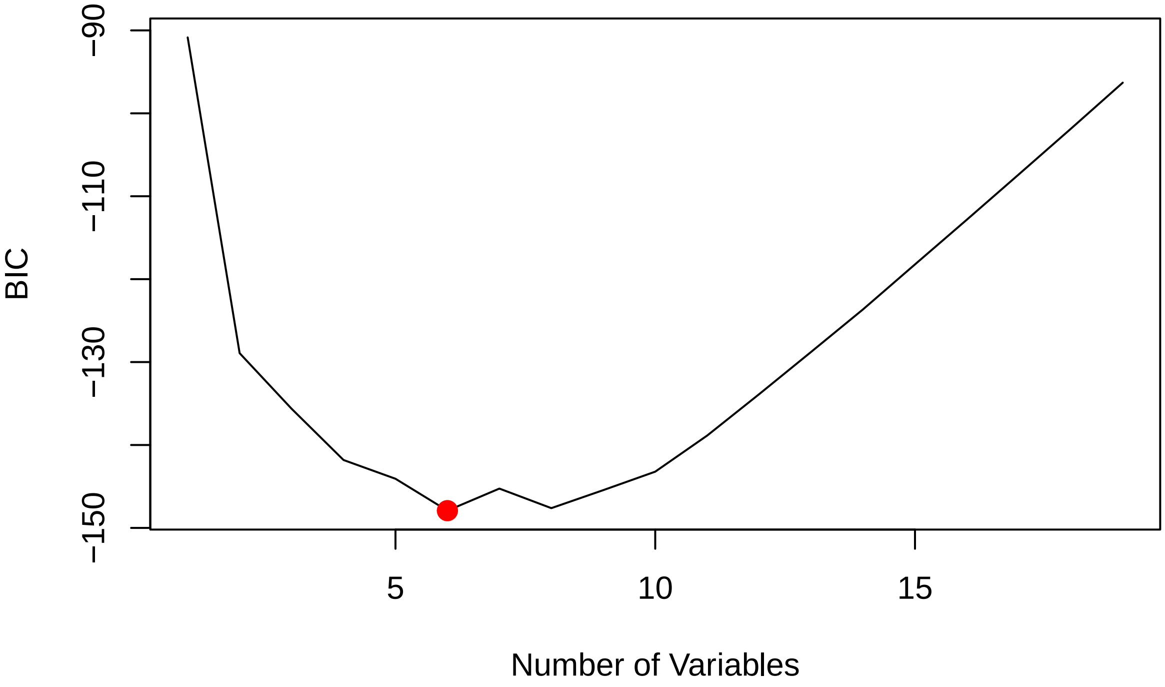


Using the BIC criteria, we select 6 as the number of variables.

```
which.min(sub_res$bic)
```

```
## [1] 6
```

```
plot(sub_res$bic, xlab = "Number of Variables", ylab = "BIC", type = "l")
points(6, sub_res$bic[6], col = "red", cex = 2, pch = 20)
```



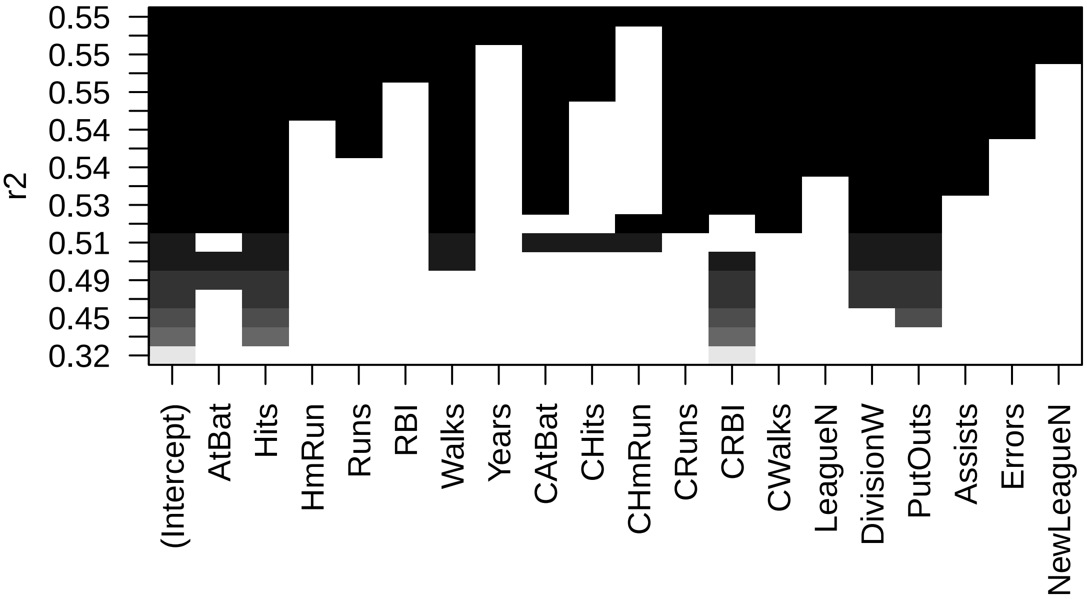
We can see which of the 6 variables have been selected.

```
coef(sub_sel, 6)
```

```
##  (Intercept)        AtBat         Hits        Walks         CRBI    DivisionW 
##   91.5117981   -1.8685892    7.6043976    3.6976468    0.6430169 -122.9515338 
##      PutOuts 
##    0.2643076
```

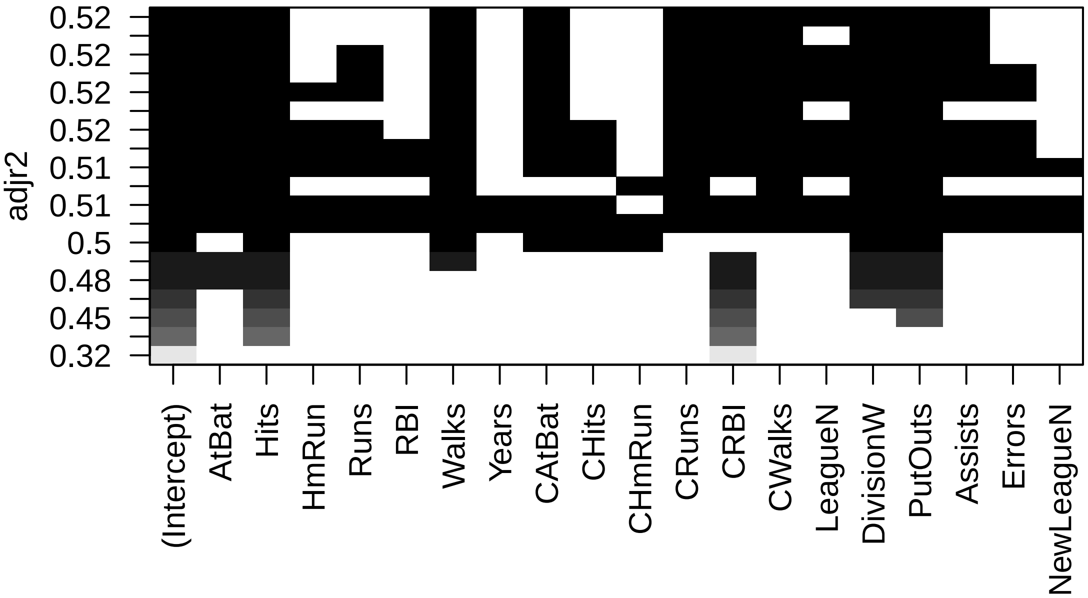
Plot displaying the selected variables and given criteria.

```
plot(sub_sel, scale = "r2")
```



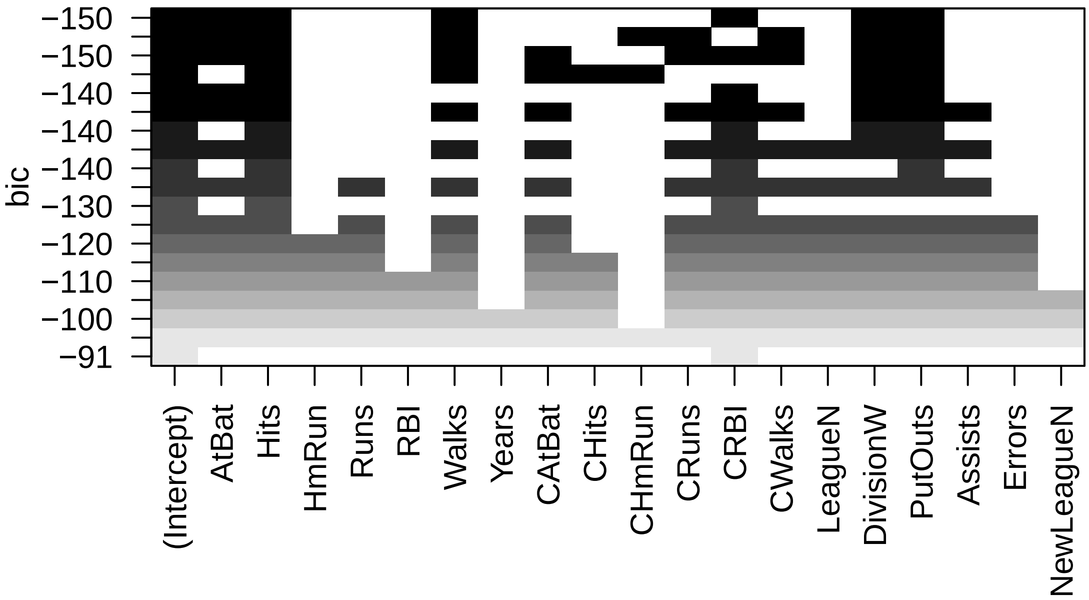
Adjusted R2.

```r
plot(sub_sel, scale = "adjr2")
```



BIC.

```r
plot(sub_sel, scale = "bic")
```


bic
−150
−150
−140
−140
−140
−130
−120
−110
−100
−91
(Intercept)
AtBat
Hits
HmRun
Runs
RBI
Walks
Years
CAtBat
CHits
CHmRun
CRuns
CRBI
CWalks
LeagueN
DivisionW
PutOuts
Assists
Errors
NewLeagueN