4.4 Variable Selection

Variable Selection

Variable Selection

The task of determining which predictors are associated with the response, in order to fit a single model involving only those predictors, is referred to as **variable selection**.

There are two distinctly different approaches to choosing the potential subsets of predictor variables, namely **Best Subset Selection** and **Stepwise Methods**.

Best Subset Selection

This approach considers all 2^m possible regression equations and identifies the subset of the predictors of a given size that maximises a measure of fit or minimises an information criterion.

With a **fixed** number of terms in the regression model, all four criteria for evaluating a subset of predictor variables (R^2_{adj}, AIC, AICc and BIC) agree that the best choice is the set of predictors with the smallest value of the residual sum of squares.

Note, however, when the comparison is across models with different numbers of predictors the four methods (R^2_{adj}, AIC, AICc and BIC) can give entirely different results.

Example: Best Subset Selection

In this example, we wish to predict a baseball player's `Salary` by various statistics associated with performance in the previous year. We will use the $regsubsets()$ function to perform best subset selection by identifying the best model that contains a given number of predictors, where best is quantified using RSS.

First, load the `Hitters` data and omit `NA` values:

```
library(ISLR)
names(Hitters)

## [1] "AtBat"    "Hits"     "HmRun"  "Runs"   "RBI"
## [6] "Walks"    "Years"    "CAtBat" "CHits"  "CHmRun"
## [11] "CRuns"   "CRBI"     "CWalks" "League" "Division"
## [16] "PutOuts" "Assists" "Errors" "Salary" "NewLeague"

dim(Hitters)

## [1] 322 20

sum(is.na(Hitters$Salary))

## [1] 59

Hitters=na.omit(Hitters)
dim(Hitters)

## [1] 263 20

sum(is.na(Hitters))

## [1] 0
```

Then, we use the $regsubsets()$ function for variable selection:

```r
library(ISLR)
library(leaps)
regfit.full=regsubsets(Salary~., Hitters)
summary(regfit.full)
```

```
Subset selection object
Call: regsubsets.formula(Salary ~ ., Hitters)
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
1  ( 1 )  " "    " "   " "    " "   " "  " "    " "    " "     " "    " "     " "
2  ( 1 )  " "    "*"   " "    " "   " "  " "    " "    " "     " "    " "     " "
3  ( 1 )  " "    "*"   " "    " "   " "  " "    " "    " "     " "    " "     " "
4  ( 1 )  " "    "*"   " "    " "   " "  " "    " "    " "     " "    " "     " "
5  ( 1 )  "*"    "*"   " "    " "   " "  " "    " "    " "     " "    " "     " "
6  ( 1 )  "*"    "*"   " "    " "   " "  "*"    " "    " "     " "    " "     " "
7  ( 1 )  " "    "*"   " "    " "   " "  "*"    " "    "*"     "*"    "*"     " "
8  ( 1 )  "*"    "*"   " "    " "   " "  "*"    " "    " "     " "    "*"     "*"
           CRBI CWalks LeagueN DivisionW PutOuts Assists Errors NewLeagueN
1  ( 1 )  "*"  " "    " "     " "       " "     " "     " "    " "
2  ( 1 )  "*"  " "    " "     " "       " "     " "     " "    " "
3  ( 1 )  "*"  " "    " "     " "       "*"     " "     " "    " "
4  ( 1 )  "*"  " "    " "     "*"       "*"     " "     " "    " "
5  ( 1 )  "*"  " "    " "     "*"       "*"     " "     " "    " "
6  ( 1 )  "*"  " "    " "     "*"       "*"     " "     " "    " "
7  ( 1 )  " "  " "    " "     "*"       "*"     " "     " "    " "
8  ( 1 )  " "  "*"    " "     "*"       "*"     " "     " "    " "
```

An asterisk indicates that a given variable is included in the corresponding model. For example, this example shows that the best two-variable model contains only `Hits` and `CRBI`. Note that by default, *regsubsets()* reports only results up to the best eight-variable model. This can be easily changed by using the **nvmax** option.

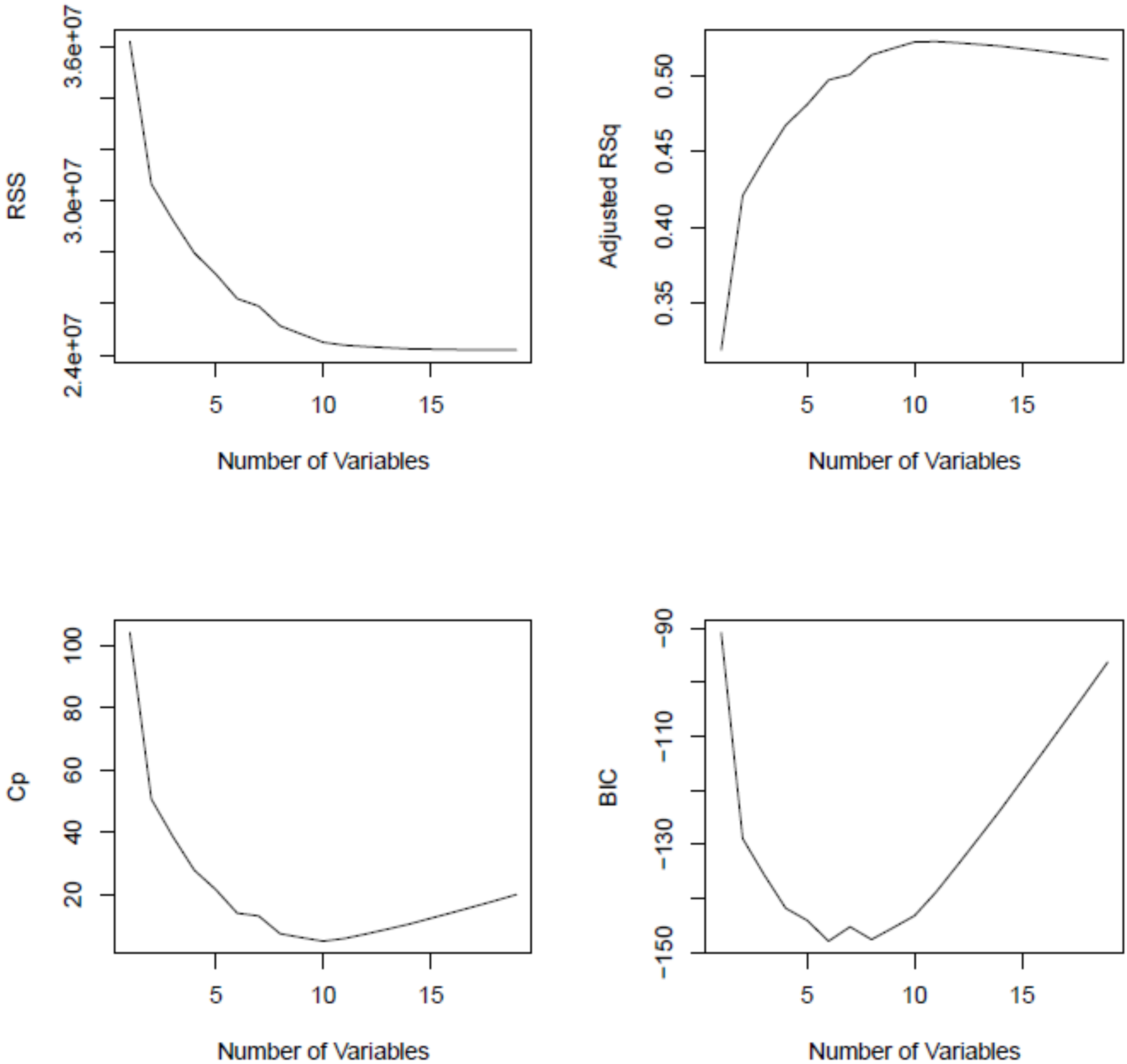
The $summary()$ function also returns R^2, RSS, R^2_{adj}, C_p and BIC. We can examine these to try to select the best overall model.

```
library(ISLR)
library(leaps)
regfit.full=regsubsets(Salary~., Hitters, nvmax=19)
reg.summary=summary(regfit.full)

names(reg.summary)
```

```
[1] "which" "rsq" "rss" "adjr2" "cp" "bic" "outmat" "obj"
```

Plotting RSS, R^2_{adj}, C_p and BIC for all the models at once will help us decide which model to select.

```
library(ISLR)
library(leaps)
regfit.full=regsubsets(Salary~., Hitters, nvmax=19)
reg.summary=summary(regfit.full)

par(mfrow=c(2,2))
plot(reg.summary$rss, xlab="Number of Variables", ylab="RSS", type="l")
plot(reg.summary$adjr2, xlab="Number of Variables", ylab="Adjusted RSq", type="l")
plot(reg.summary$cp, xlab="Number of Variables", ylab="Cp", type="l")
plot(reg.summary$bic, xlab="Number of Variables", ylab="BIC", type="l")
```



We can see from these plots that, for instance, the model selected using BIC is the six-variable model. This model contains **AtBat**, **Hits**, **Walks**, **CRBI**, **DivisionW** and **PutOuts** as predictors. We can now use the *coef()* function to display the coefficient estimates for this model.

```
library(ISLR)
library(leaps)
regfit.full=regsubsets(Salary~., Hitters, nvmax=19)
coef(regfit.full,6)
```

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

Stepwise Model Selection

This approach is based on examining just a sequential subset of the 2^m possible regression models. Two of the most popular variations are *backward elimination* and *forward selection*.

Backward elimination starts with all potential predictor variables in the regression model. Then, at each step, it deletes the predictor variable such that the resulting model has the lowest value of an information criterion. (This amounts to removing the predictor with the largest p-value each time.)

Forward selection starts with no potential predictor variables in the regression equation. Then, at each step, it adds the predictor such that the resulting model has the lowest value of an information criterion. (This amounts to adding the predictor with the smallest p-value each time.)

Backward elimination and forward selection do not necessarily find the model that minimises the information criteria across all 2^m possible predictor subsets, and there is no guarantee that backward elimination and forward selection will produce the same final model.

Example: Backward and forward selection

Consider backward elimination and forward selection for the `Cheese` data set.

Backward selection:

- We start with the model that has all the parameters
- We check the impact of removing one variable from the model:

```
library(faraway)
data("cheddar")

cheddar.lm <- lm(taste~., cheddar)
drop1(cheddar.lm)
```

```
Single term deletions

Model:
taste ~ acetic + H2S + lactic
       Df Sum of Sq    RSS    AIC
<none>              2668.4 142.64
Acetic  1      0.55 2669.0 140.65
H2S     1   1007.66 3676.1 150.25
Lactic  1    533.32 3201.7 146.11
```

Note that since we do not specify the `scope` argument then all variables can be dropped.

The lowest AIC is obtained for the model where the variable `Acetic` is removed.

- We create a new model without `Acetic` :
- We check the impact of removing one variable from this new model:

```
library(faraway)
data("cheddar")

cheddar.lm2 <- lm(taste~H2S+Lactic, cheddar)
drop1(cheddar.lm2)
```

```
Single term deletions

Model:
taste ~ H2S + lactic
       Df Sum of Sq    RSS    AIC
<none>                2669.0 140.65
H2S     1   1193.52 3862.5 149.74
Lactic  1    617.18 3286.1 144.89
```

The lowest AIC is obtained for the model where no variables are removed. The selected model is then $\mathtt{taste} \sim \mathtt{H2S} + \mathtt{Lactic}$.

Forward selection:

- We start with a model with no variables:

```
library(faraway)
data("cheddar")

cheddar.lm <- lm(taste~1,cheddar)
```

- We check the impact of adding one variable to the model:

```
library(faraway)
data("cheddar")

cheddar.lm <- lm(taste~1,cheddar)
add1(cheddar.lm, scope=~Acetic+H2S+Lactic)
```

```
Single term additions

Model:
taste ~ 1
        Df  Sum of Sq    RSS     AIC
<none>                 7662.9  168.29
Acetic  1      2314.1 5348.7  159.50
H2S     1      4376.7 3286.1  144.89
Lactic  1      3800.4 3862.5  149.74
```

Now you need to specify the `scope` in the `add1` function, i.e. the potential variables to add. The lowest AIC goes for `H2S` so it is included in the model.

- We consider the new model:
- We check if we should add a second variable:

```
library(faraway)
data("cheddar")

cheddar.lm2 <- lm(taste~H2S,cheddar)
add1(cheddar.lm2, scope=~Acetic+H2S+Lactic)
```

```
Single term additions

Model:
taste ~ H2S
       Df Sum of Sq    RSS    AIC
<none>              3286.1 144.89
Acetic  1     84.41 3201.7 146.11
Lactic  1    617.18 2669.0 140.65
```

The lowest AIC is obtained when `Lactic` is included in the model.

- We consider the new model:
- We check if we should add a third variable:

```
library(faraway)
data("cheddar")

cheddar.lm3 <- lm(taste~H2S+Lactic,cheddar)
add1(cheddar.lm3, scope=~Acetic+H2S+Lactic)
```

```
Single term additions

Model:
taste ~ H2S + lactic
       Df Sum of Sq    RSS    AIC
<none>              2669.0 140.65
Scetic  1     84.41 2668.4 142.64
```

The lowest AIC is obtained when no other variables are included in the model. The selected model is then $\texttt{taste} \sim \texttt{H2S} + \texttt{Lactic}$.

> Now consider the **`Sydneymaximumtemperature`** data set:

<u>Backward selection:</u>

```
mos.df <- read.table("/course/data/mos.df.txt", header=TRUE, quote='"')
mos.lm <- lm(Maxtemp ~ ., mos.df)

drop1(mos.lm, scope~Modst+Modsp+Modthik)
```

```
Single term deletions
```

```
Model:
Maxtemp ~ Modst + Modsp + Modthik
         Df          Sum of Sq     RSS       AIC
<none>                             3305.6    817.06
Modst    1           10.71         3316.3    816.26
Modsp    1           29.49         3335.1    818.34
Modthik  1           1947.60       5253.2    985.99
```

- Dropping **`Modst`** leads to smaller AIC, so we fit the model

```
mos.df <- read.table("/course/data/mos.df.txt", header=TRUE, quote='"')
mos2.lm <- lm(Maxtemp ~ Modsp + Modthik, mos.df)

drop1(mos2.lm, scope~Modsp+Modthik)
```

```
Single term deletions

Model:
Maxtemp ~ Modsp + Modthik
          Df      Sum of Sq      RSS       AIC
<none>                          3316.3    816.26
Modsp    1       32.21          3348.5    817.82
Modthik  1       2734.50        6050.8    1036.15
```

- Since dropping any term lead to an increase in AIC, we conclude with the final model $\texttt{Maxtemp} \sim \texttt{Modsp} + \texttt{Modthik}$

Forward selection:

- Now for forward selection, we start with the null model

```
mos.df <- read.table("/course/data/mos.df.txt", header=TRUE, quote='"')
mos.lm <- lm(Maxtemp~1,mos.df)
add1(mos.lm,scope~Modst+Modsp+Modthik)
```

```
Single term additions

Model:
Maxtemp ~ 1
        Df Sum of Sq     RSS     AIC
<none>                6241.6 1045.60
Modst    1    891.60  5350.0  990.72
Modsp    1    190.72  6050.8 1036.15
Modthik  1   2893.01  3348.5  817.82
```

- Adding **`Modthik`** leads to the largest reduction in AIC, so we next fit

```
mos.df <- read.table("/course/data/mos.df.txt", header=TRUE, quote='"')
mos2.lm <- lm(Maxtemp~Modthik,mos.df)
```

```
add1(mos2.lm, scope~Modst+Modsp+Modthik)
```

```
Single term additions

Model:
Maxtemp ~ Modthik
       Df Sum of Sq    RSS    AIC
<none>                3348.5 817.82
Modst   1    13.432   3335.1 818.34
Modsp   1    32.210   3316.3 816.26
```

- Next we add $\mathtt{Modsp}$, since adding this term reduces the AIC,

```
mos.df <- read.table("/course/data/mos.df.txt", header=TRUE, quote='"')
mos3.lm <- lm(Maxtemp~Modthik+Modsp,mos.df)
add1(mos3.lm, scope~Modst+Modsp+Modthik)
```

```
Single term additions

Model:
Maxtemp ~ Modthik + Modsp
       Df Sum of Sq    RSS    AIC
<none>                3316.3 816.26
Modst   1     10.71   3305.6 817.06
```

- Now since adding Modst would increase the AIC, we stop and conclude with forward selection $\mathtt{Maxtemp} \sim \mathtt{Modsp} + \mathtt{Modthik}$.

Activity in R: Forward and Backward Stepwise Selection

Consider the **Hitters** dataset and use the $regsubsets()$ function with the argument $\texttt{method} = \texttt{"forward"}$ or $\texttt{method} = \texttt{"backward"}$ to perform forward stepwise and backward stepwise selection. What is the best seven variable model?