Math 158, Spring 2013
Jo Hardin
Multiple Regression IV – R code
Model Building

Consider the multiple regression model:

$$\begin{aligned}
E[Y] &= \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_4 X_4 + \beta_5 X_5 + \beta_6 X_6 \\
Y &= \text{state ave SAT score} \\
X_1 &= \text{\% of eligible seniors who took the exam, } \textit{takers} \\
X_2 &= \text{median income of families of test takers, } \textit{income} \\
X_3 &= \text{ave number of years of formal eduction, } \textit{years} \\
X_4 &= \text{\% of test takers who attend public school, } \textit{public} \\
X_5 &= \text{total state expenditure on public secondary schools (\$100 /student), } \textit{expend} \\
X_6 &= \text{median percentile rank of test takers within their secondary school class, } \textit{rank}
\end{aligned}$$

```
> sat.data <- read.table("sat.csv", header=T, sep=",")
> attach(sat.data)
> sat.n <- nrow(sat.data)                # be careful with missing values!!
> ltakers <- log(takers)                 # variable is quite right skewed
```

AIC and BIC in R

1.
   ```
   > sat.lm0 <- lm(sat ~ 1)
   > summary(sat.lm0)

   Coefficients:
               Estimate Std. Error t value Pr(>|t|)
   (Intercept)   948.45      10.21   92.86   <2e-16 ***
   ---
   Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

   Residual standard error: 71.5 on 48 degrees of freedom

   > sat.sse0 <- sum(resid(sat.lm0) ^2)
   > sat.n + sat.n*log(2*pi) + sat.n * log(sat.sse0 / sat.n) + 2 * (1+1)
   [1] 560.4736
   > AIC(sat.lm0, k=2)
   [1] 560.4736
   > sat.n + sat.n * log(2*pi) + sat.n*log(sat.sse0/sat.n) + log(sat.n)*(1+1)
   [1] 564.2573
   > AIC(sat.lm0, k=log(sat.n))
   [1] 564.2573
   ```

2.

```
> sat.lm1 <- lm(sat ~ ltakers)
> summary(sat.lm1)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 1112.408     12.386   89.81   <2e-16 ***
ltakers      -59.175      4.167  -14.20   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 31.41 on 47 degrees of freedom
Multiple R-squared: 0.811,      Adjusted R-squared: 0.807
F-statistic: 201.7 on 1 and 47 DF,  p-value: < 2.2e-16

> sat.sse1 <- sum(resid(sat.lm1) ^2)
> sat.n + sat.n*log(2*pi) + sat.n * log(sat.sse1 / sat.n) + 2 * (2+1)
[1] 480.832
> AIC(sat.lm1, k=2)
[1] 480.832
> sat.n + sat.n * log(2*pi) + sat.n*log(sat.sse1/sat.n) + log(sat.n) * (2+1)
[1] 486.5075
> AIC(sat.lm1, k=log(sat.n))
[1] 486.5075
```

These notes belong at the end... I'm putting them here to save paper and keep the formatting reasonably clear.

- Notice that C_p and F-tests use a "full" model MSE. Typically, the MSE will only be an unbiased predictor of σ^2 in backwards variable selection.

- SBC usually results in fewer parameters in the model than AIC.

- Using different selection criteria may lead to different models (there is no one best model).

- The order in which variables are entered does not necessarily represent their importance. As a variable entered early on can be dropped at a later stage because it is predicted well from the other explanatory variables that have been subsequently added to the model.

Forward Variable Selection: F-tests

```
> add1(lm(sat~1), sat~ ltakers + income + years + public + expend +
                        rank, test="F")
Single term additions
Model:
sat ~ 1
        Df Sum of Sq    RSS    AIC  F value     Pr(F)
<none>                245376    419
ltakers  1    199007  46369    340 201.7138 < 2.2e-16 ***
income   1    102026 143350    395  33.4513 5.711e-07 ***
years    1     26338 219038    416   5.6515   0.02156 *
public   1      1232 244144    421   0.2371   0.62856
expend   1       386 244991    421   0.0740   0.78683
rank     1    190297  55079    348 162.3828 < 2.2e-16 ***
```

```
> add1(lm(sat~ ltakers), sat~ ltakers + income + years + public +
                        expend + rank, test="F")
Single term additions
Model:
sat ~ ltakers
       Df Sum of Sq   RSS   AIC F value     Pr(F)
<none>              46369   340
income  1       785 45584   341  0.7922  0.378064
years   1      6364 40006   335  7.3170  0.009546 **
public  1       449 45920   341  0.4497  0.505838
expend  1     20523 25846   313 36.5274 2.489e-07 ***
rank    1       871 45498   341  0.8807  0.352900
```

```
> add1(lm(sat~ ltakers + expend), sat~ ltakers + income + years +
                          public + expend + rank, test="F")
Single term additions
Model:
sat ~ ltakers + expend
       Df Sum of Sq     RSS     AIC F value  Pr(F)
<none>              25845.8   313.1
income  1      53.3 25792.5   315.0  0.0930 0.7617
years   1    1248.2 24597.6   312.7  2.2835 0.1377
public  1       1.3 25844.5   315.1  0.0023 0.9624
rank    1    1053.6 24792.2   313.1  1.9124 0.1735
```

Note: `Sum of Sq` refers to the SSR(new variable | current model) (additional reduction in SSE). `RSS` is the SSE for the model that contains the current variables and the new variable.

Backward Variable Selection: F-tests

```
> drop1(lm(sat ˜ ltakers + income + years + public + expend + rank), test="F")
Single term deletions
Model:
sat ˜ ltakers + income + years + public + expend + rank
        Df Sum of Sq   RSS   AIC F value     Pr(F)
<none>               21397   312
ltakers  1      2150 23547   315  4.2203   0.04620 *
income   1       340 21737   311  0.6681   0.41834
years    1      2532 23928   315  4.9693   0.03121 *
public   1        20 21417   310  0.0393   0.84390
expend   1     10964 32361   330 21.5221 3.404e-05 ***
rank     1      2679 24076   316  5.2587   0.02691 *

> drop1(lm(sat ˜ ltakers + income + years + expend + rank), test="F")
Single term deletions
Model:
sat ˜ ltakers + income + years + expend + rank
        Df Sum of Sq   RSS   AIC F value     Pr(F)
<none>               21417   310
ltakers  1      2552 23968   313  5.1232   0.02871 *
income   1       505 21922   309  1.0147   0.31942
years    1      3011 24428   314  6.0451   0.01805 *
expend   1     12465 33882   330 25.0277 1.003e-05 ***
rank     1      3162 24578   315  6.3480   0.01555 *
```

If you ask to `add1` here (that is, to see whether it makes sense to add either public or income back into the model), neither is significant.

```
> drop1(lm(sat ˜ ltakers + years + expend + rank), test="F")
Single term deletions
Model:
sat ˜ ltakers + years + expend + rank
        Df Sum of Sq   RSS   AIC F value    Pr(F)
<none>               21922   309
ltakers  1      5094 27016   317 10.2249 0.002568 **
years    1      2870 24792   313  5.7606 0.020687 *
expend   1     13620 35542   331 27.3360 4.52e-06 ***
rank     1      2676 24598   313  5.3700 0.025200 *
```

Forward Stepwise: AIC

```
> step(lm(sat~1), sat ~ ltakers + income + years + public + expend +
                     rank,direction = "forward")
Start:  AIC=419.42
sat ~ 1
          Df Sum of Sq    RSS    AIC
+ ltakers  1    199007  46369    340
+ rank     1    190297  55079    348
+ income   1    102026 143350    395
+ years    1     26338 219038    416
<none>                 245376    419
+ public   1      1232 244144    421
+ expend   1       386 244991    421

Step:  AIC=339.78
sat ~ ltakers
         Df Sum of Sq   RSS   AIC
+ expend  1     20523 25846   313
+ years   1      6364 40006   335
<none>                46369   340
+ rank    1       871 45498   341
+ income  1       785 45584   341
+ public  1       449 45920   341

Step:  AIC=313.14
sat ~ ltakers + expend
         Df Sum of Sq     RSS     AIC
+ years   1    1248.2 24597.6   312.7
+ rank    1    1053.6 24792.2   313.1
<none>                25845.8   313.1
+ income  1      53.3 25792.5   315.0
+ public  1       1.3 25844.5   315.1

Step:  AIC=312.71
sat ~ ltakers + expend + years
         Df Sum of Sq     RSS     AIC
+ rank    1    2675.5 21922.1   309.1
<none>                24597.6   312.7
+ public  1     287.8 24309.8   314.1
+ income  1      19.2 24578.4   314.7

Step:  AIC=309.07
sat ~ ltakers + expend + years + rank
         Df Sum of Sq     RSS     AIC
```

```
<none>                21922.1   309.1
+ income  1     505.4 21416.7   309.9
+ public  1     185.0 21737.1   310.7

lm(formula = sat ~ ltakers + expend + years + rank)
(Intercept)      ltakers       expend        years         rank
    399.115      -38.100        3.996       13.147        4.400
```

Backward Stepwise: SBC

```
> step(lm(sat ~ (ltakers + income + years + public + expend + rank)),
                direction = "backward", k=log(sat.n))
Start:  AIC=325.12
sat ~ (ltakers + income + years + public + expend + rank)
          Df Sum of Sq   RSS   AIC
- public   1        20 21417   321
- income   1       340 21737   322
<none>                 21397   325
- ltakers  1      2150 23547   326
- years    1      2532 23928   327
- rank     1      2679 24076   327
- expend   1     10964 32361   342

Step:  AIC=321.28
sat ~ ltakers + income + years + expend + rank
          Df Sum of Sq   RSS   AIC
- income   1       505 21922   319
<none>                 21417   321
- ltakers  1      2552 23968   323
- years    1      3011 24428   324
- rank     1      3162 24578   324
- expend   1     12465 33882   340

Step:  AIC=318.53
sat ~ ltakers + years + expend + rank
          Df Sum of Sq   RSS   AIC
<none>                 21922   319
- rank     1      2676 24598   320
- years    1      2870 24792   321
- ltakers  1      5094 27016   325
- expend   1     13620 35542   338

lm(formula = sat ~ ltakers + years + expend + rank)
(Intercept)      ltakers        years       expend         rank
    399.115      -38.100       13.147        3.996        4.400
```

To get an idea of how complicated your models can get, try this:

```
> step(lm(sat ~ (ltakers + income + years + public + expend + rank)^2),
            direction = "backward")
```