MSiA 400 Lab Advanced Regression with R

Oct 27, 2014

Young-Woong Park

Variable Selection in Multiple Regression

- Goal: choose the candidate variables to obtain a regression model that contains the "best" subset of regressor variables

- Methods
  - All possible regression
  - Stepwise regression

- Criteria
  - $C_p = \frac{SSE_p}{MSE_p} + 2p - n$
  - AIC $= n \log \frac{SSE_p}{n} + 2p$
  - Adjusted $r^2 = 1 - \frac{SSR/(n-p-1)}{SST/(n-1)}$

Data Set for Multiple Regression

- Data set: Wine quality (white wine)
  - Number of observations: 4898
  - Number of attributes: 11 + output attribute
  - Input attributes: fixed acidity, volatile acidity, citric acid, residual sugar,
    chlorides, free sulfur dioxide, total sulfur dioxide, density,
    pH, sulphates, alcohol
  - Output attribute: quality (score between 0 and 10)

Ref: P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis., Modeling wine preferences by data mining from physicochemical properties, Decision Support Systems, Elsevier, 47(4):547-553. ISSN: 0167-9236.

- Read the data set

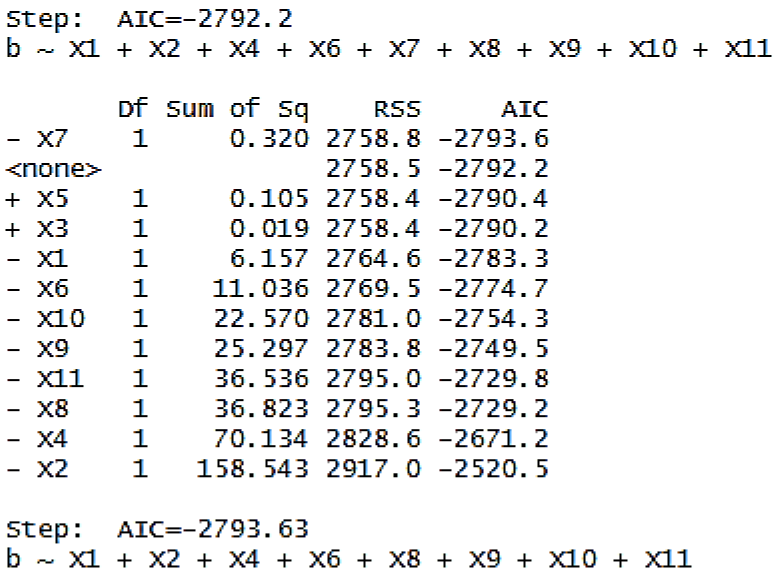
```
> wine <- read.delim(“…/whitewine.txt");
> y = wine[,1];
> x = wine[,2:length(wine[1,])];
```

Stepwise Regression with AIC

- Stepwise regression

```
> library(MASS)
> reg = lm(y~., data=x)
> reg.step = stepAIC(object=reg, direction="both")
```

"forward"
"backward"

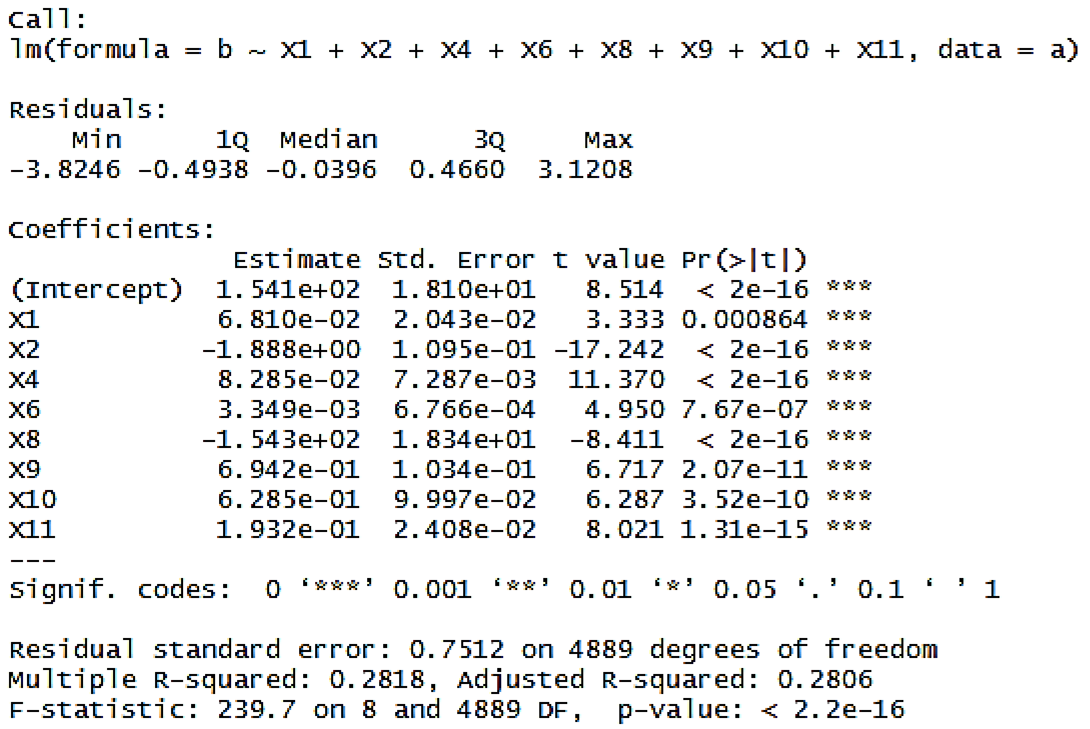


```
Step:  AIC=-2792.2
b ~ x1 + x2 + x4 + x6 + x7 + x8 + x9 + x10 + x11

       Df Sum of Sq    RSS     AIC
- x7    1     0.320 2758.8 -2793.6
<none>              2758.5 -2792.2
+ x5    1     0.105 2758.4 -2790.4
+ x3    1     0.019 2758.4 -2790.2
- x1    1     6.157 2764.6 -2783.3
- x6    1    11.036 2769.5 -2774.7
- x10   1    22.570 2781.0 -2754.3
- x9    1    25.297 2783.8 -2749.5
- x11   1    36.536 2795.0 -2729.8
- x8    1    36.823 2795.3 -2729.2
- x4    1    70.134 2828.6 -2671.2
- x2    1   158.543 2917.0 -2520.5

Step:  AIC=-2793.63
b ~ x1 + x2 + x4 + x6 + x8 + x9 + x10 + x11
```

Stepwise Regression with AIC (Cont.)

- Displaying the summary

```
> summary(reg.step)
```



```
Call:
lm(formula = b ~ X1 + X2 + X4 + X6 + X8 + X9 + X10 + X11, data = a)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.8246 -0.4938 -0.0396  0.4660  3.1208 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  1.541e+02  1.810e+01   8.514  < 2e-16 ***
X1           6.810e-02  2.043e-02   3.333 0.000864 ***
X2          -1.888e+00  1.095e-01 -17.242  < 2e-16 ***
X4           8.285e-02  7.287e-03  11.370  < 2e-16 ***
X6           3.349e-03  6.766e-04   4.950 7.67e-07 ***
X8          -1.543e+02  1.834e+01  -8.411  < 2e-16 ***
X9           6.942e-01  1.034e-01   6.717 2.07e-11 ***
X10          6.285e-01  9.997e-02   6.287 3.52e-10 ***
X11          1.932e-01  2.408e-02   8.021 1.31e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7512 on 4889 degrees of freedom
Multiple R-squared: 0.2818, Adjusted R-squared: 0.2806 
F-statistic: 239.7 on 8 and 4889 DF,  p-value: < 2.2e-16
```

Stepwise Regression with AIC (Cont.)

- Calculating / Referencing Statistics

```
> formula(reg.step);                      # print the formula of the model
> AIC(reg. step);                         # print AIC value of the model
> summary(reg. step)$r.squared;           # print $r^2$ value of the model
> summary(reg. step)$adj.r.squared;       # print adjusted $r^2$ value
> e = resid(reg. step);                   # define residuals
> SSE = sum(e^2);                         # calculate Sum of Squared errors
> SAE = sum(abs(e));                      # calculate Sum of Absolute errors
```

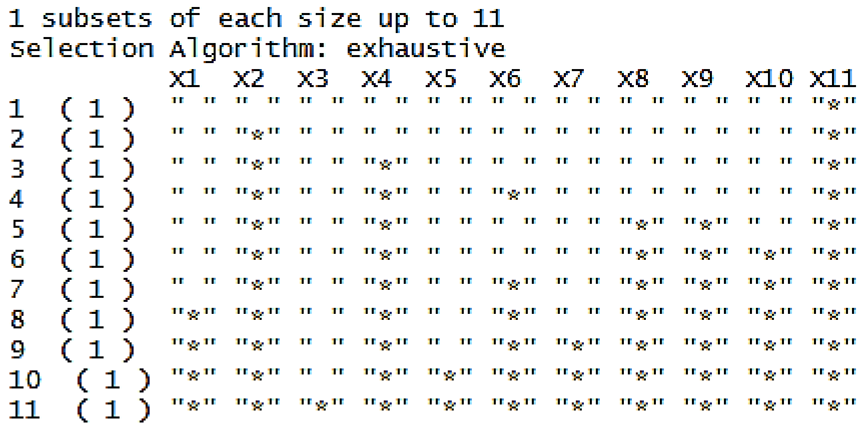
Variable Selection Using Package

- Load package leaps

```
> library(leaps)
```

- Finding the best subset
  - For number of variables p=1,2,…,nvmax,
    Find nbest best subsets with cardinality p

```
> reg.exh = regsubsets(x,y, nbest=1, nvmax=length(y), method="exhaustive");
> summary(reg.exh)
```

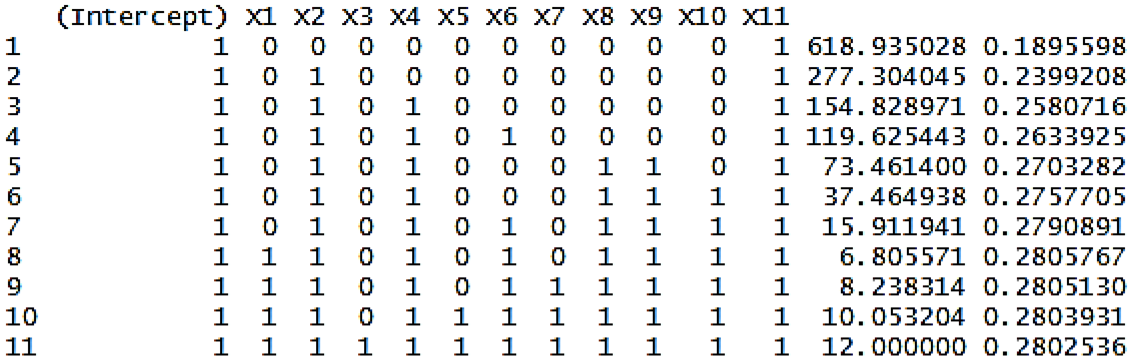


```
1 subsets of each size up to 11
Selection Algorithm: exhaustive
          X1  X2  X3  X4  X5  X6  X7  X8  X9  X10 X11
1  ( 1 )  " " " " " " " " " " " " " " " " " " " " "*"
2  ( 1 )  " " "*" " " " " " " " " " " " " " " " " "*"
3  ( 1 )  " " "*" " " "*" " " " " " " " " " " " " "*"
4  ( 1 )  " " "*" " " "*" " " "*" " " " " " " " " "*"
5  ( 1 )  " " "*" " " "*" " " " " " " "*" "*" " " "*"
6  ( 1 )  " " "*" " " "*" " " " " " " "*" "*" "*" "*"
7  ( 1 )  " " "*" " " "*" " " "*" " " "*" "*" "*" "*"
8  ( 1 )  "*" "*" " " "*" " " "*" " " "*" "*" "*" "*"
9  ( 1 )  "*" "*" " " "*" " " "*" "*" "*" "*" "*" "*"
10  ( 1 ) "*" "*" " " "*" "*" "*" "*" "*" "*" "*" "*"
11  ( 1 ) "*" "*" "*" "*" "*" "*" "*" "*" "*" "*" "*"
```

Variable Selection Using Package (Cont.)

- Calculating / Referencing Statistics

```
> summary(reg.exh)$which
> summary(reg.exh)$cp
> summary(reg.exh)$adjr2
> cbind(summary(reg.exh)$which, summary(reg.exh)$cp, summary(reg.exh)$adjr2)
```

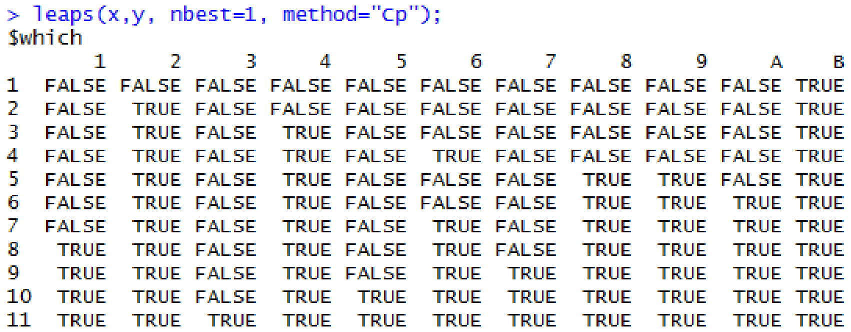


```
   (Intercept) x1 x2 x3 x4 x5 x6 x7 x8 x9 x10 x11
1            1  0  0  0  0  0  0  0  0  0   0   1 618.935028 0.1895598
2            1  0  1  0  0  0  0  0  0  0   0   1 277.304045 0.2399208
3            1  0  1  0  1  0  0  0  0  0   0   1 154.828971 0.2580716
4            1  0  1  0  1  0  1  0  0  0   0   1 119.625443 0.2633925
5            1  0  1  0  1  0  0  0  1  1   0   1  73.461400 0.2703282
6            1  0  1  0  1  0  0  0  1  1   1   1  37.464938 0.2757705
7            1  0  1  0  1  0  1  0  1  1   1   1  15.911941 0.2790891
8            1  1  1  0  1  0  1  0  1  1   1   1   6.805571 0.2805767
9            1  1  1  0  1  0  1  1  1  1   1   1   8.238314 0.2805130
10           1  1  1  0  1  1  1  1  1  1   1   1  10.053204 0.2803931
11           1  1  1  1  1  1  1  1  1  1   1   1  12.000000 0.2802536
```

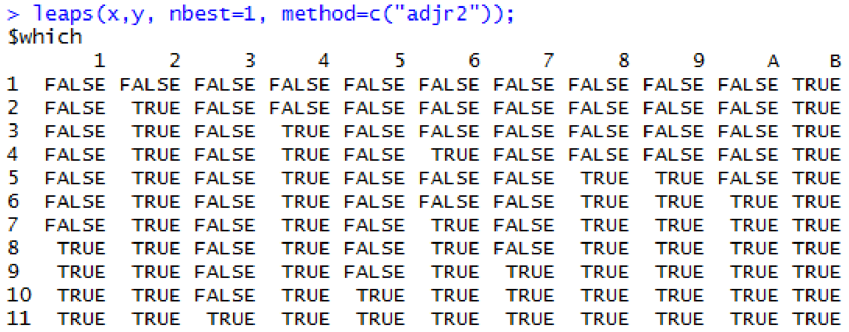
Variable Selection Using Package (Cont.)

- Optimizing various criteria

```
> leaps(x,y,nbest=1,method=“Cp”)
> leaps(x,y,nbest=1,method=“adjr2”)
```

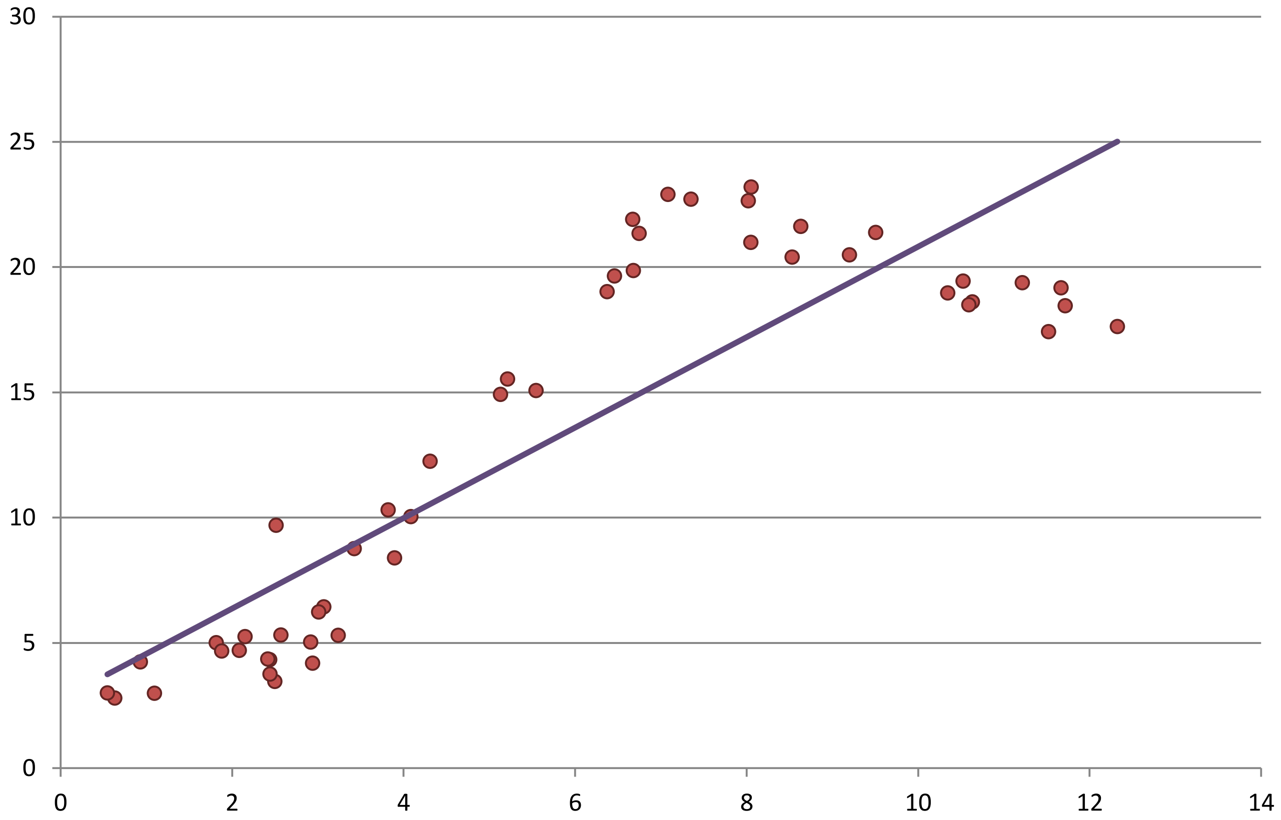


```
> leaps(x,y, nbest=1, method="Cp");
$which
       1     2     3     4     5     6     7     8     9     A     B
1  FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
2  FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
3  FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
4  FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
5  FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE
6  FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
7  FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
8   TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
9   TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
10  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
11  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```



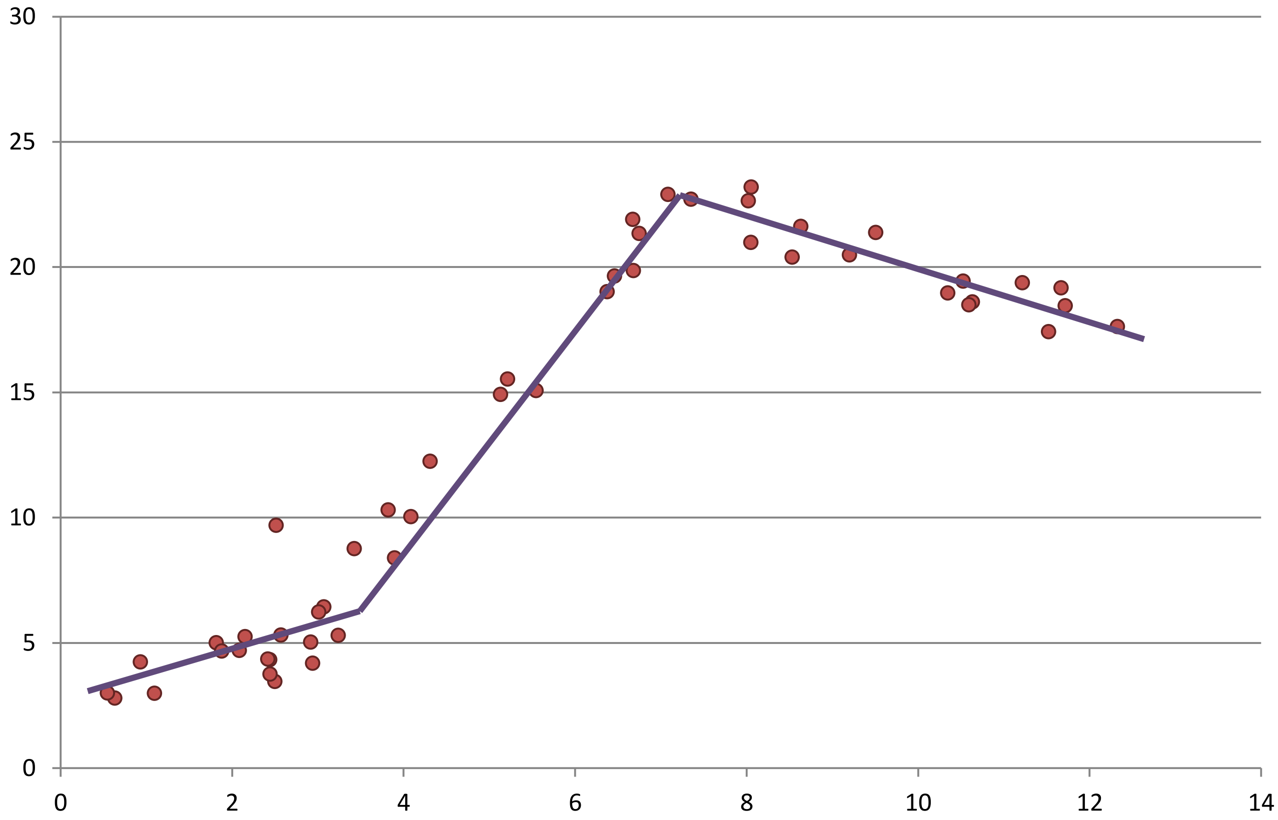
```
> leaps(x,y, nbest=1, method=c("adjr2"));
$which
       1     2     3     4     5     6     7     8     9     A     B
1  FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
2  FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
3  FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
4  FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
5  FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE  TRUE
6  FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
7  FALSE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
8   TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
9   TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
10  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
11  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

Piecewise Regression



Is simple linear regression working?

Piecewise Regression

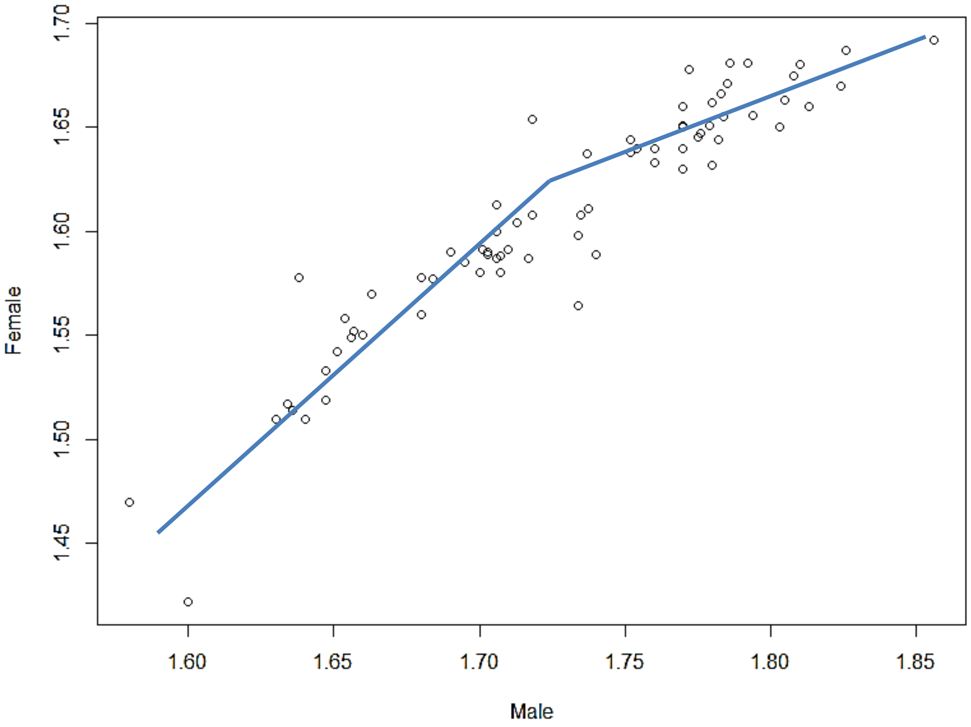


Is simple linear regression working?

Piecewise Regression

- Load height2.txt

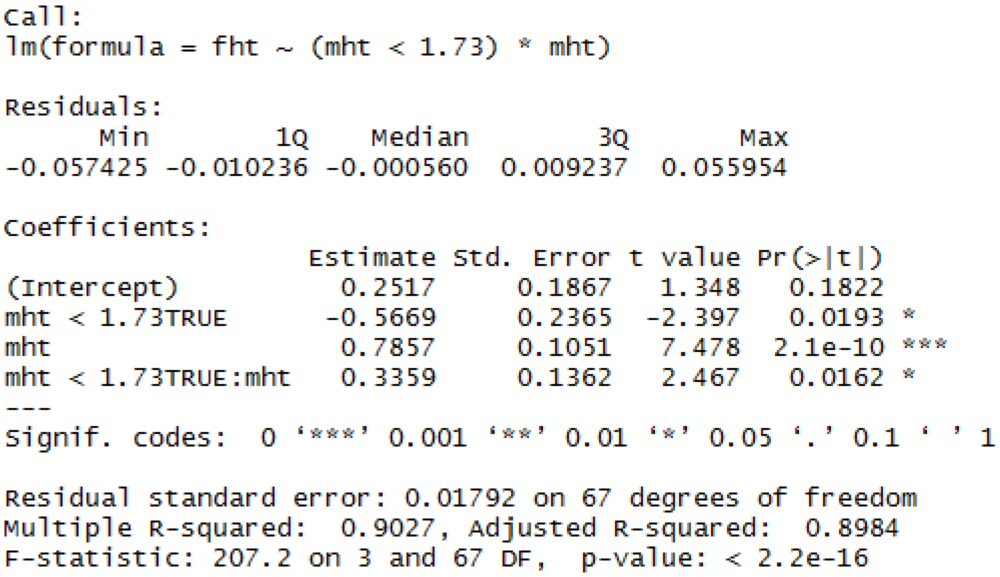
```
> height2 <- read.delim(“…/height2.txt");
> mht = height2[,1];
> fht = height2[,2];
```



Piecewise Regression (Cont.)

- Let us set male height 1.73 as a breakpoint

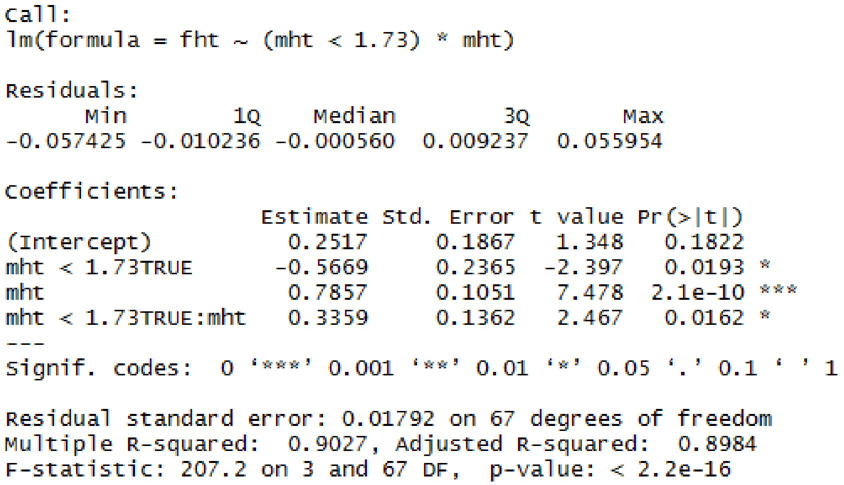
```
> reg.seg = lm(fht ~ (mht<1.73)*mht )
```



```
Call:
lm(formula = fht ~ (mht < 1.73) * mht)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.057425 -0.010236 -0.000560  0.009237  0.055954 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)          0.2517     0.1867   1.348   0.1822    
mht < 1.73TRUE      -0.5669     0.2365  -2.397   0.0193 *  
mht                  0.7857     0.1051   7.478  2.1e-10 ***
mht < 1.73TRUE:mht   0.3359     0.1362   2.467   0.0162 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01792 on 67 degrees of freedom
Multiple R-squared:  0.9027, Adjusted R-squared:  0.8984 
F-statistic: 207.2 on 3 and 67 DF,  p-value: < 2.2e-16
```

Piecewise Regression (Cont.)

- Interpreting the result



```
Call:
lm(formula = fht ~ (mht < 1.73) * mht)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.057425 -0.010236 -0.000560  0.009237  0.055954 

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)    
(Intercept)           0.2517     0.1867   1.348   0.1822    
mht < 1.73TRUE       -0.5669     0.2365  -2.397   0.0193 *  
mht                   0.7857     0.1051   7.478  2.1e-10 ***
mht < 1.73TRUE:mht    0.3359     0.1362   2.467   0.0162 *  
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01792 on 67 degrees of freedom
Multiple R-squared:  0.9027,	Adjusted R-squared:  0.8984 
F-statistic: 207.2 on 3 and 67 DF,  p-value: < 2.2e-16
```

$$fht = 0.2517 - 0.5669 * 1_{mht<1.73} + 0.7857mht + 0.3359 * 1_{mht<1.73} * mht$$

Note

1. For an observation with male height ≥ 1.73, we have

$$\begin{aligned} fht &= 0.2517 - 0.5669 * 0 + 0.7857mht + 0.3359 * 0 * mht \\ &= 0.2517 + 0.7857mht \end{aligned}$$

2. For an observation with male height < 1.73, we have

$$\begin{aligned} fht &= 0.2517 - 0.5669 * 1 + 0.7857mht + 0.3359 * 1 * mht \\ &= -0.3152 + 1.1216mht \end{aligned}$$

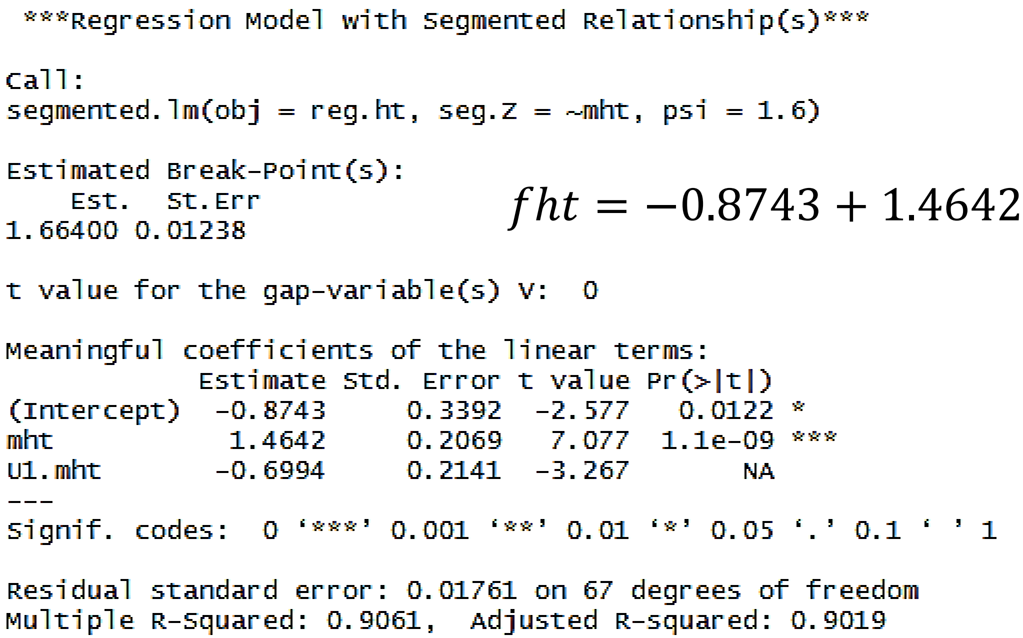
Piecewise Regression Using Package

- Load package segmented

```
> library(segmented)
```

- Let us guess male height 1.73 as a breakpoint

```
> reg.ht = lm(fht ~ mht)
> reg.seg1 = segmented(reg.ht, seg.Z = ~mht, psi=1.73)
```

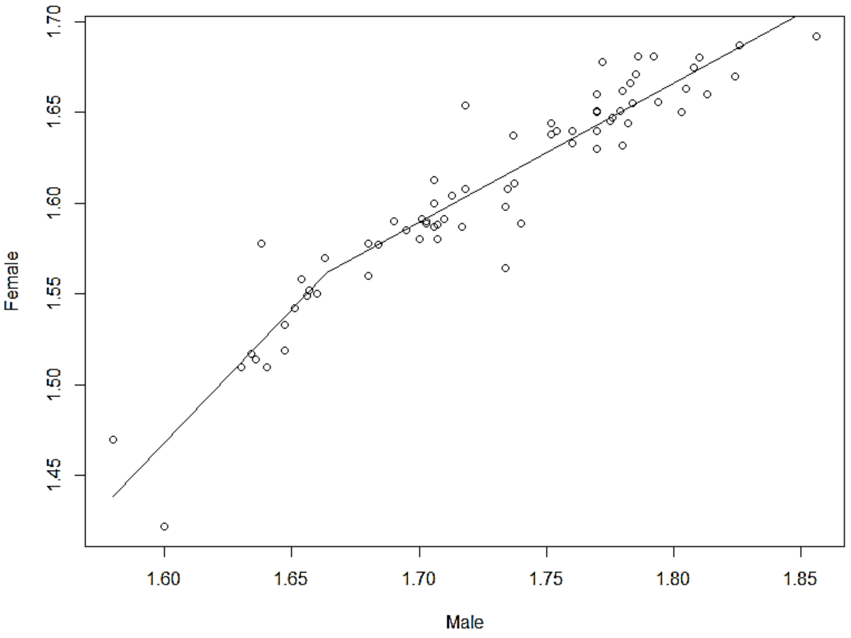


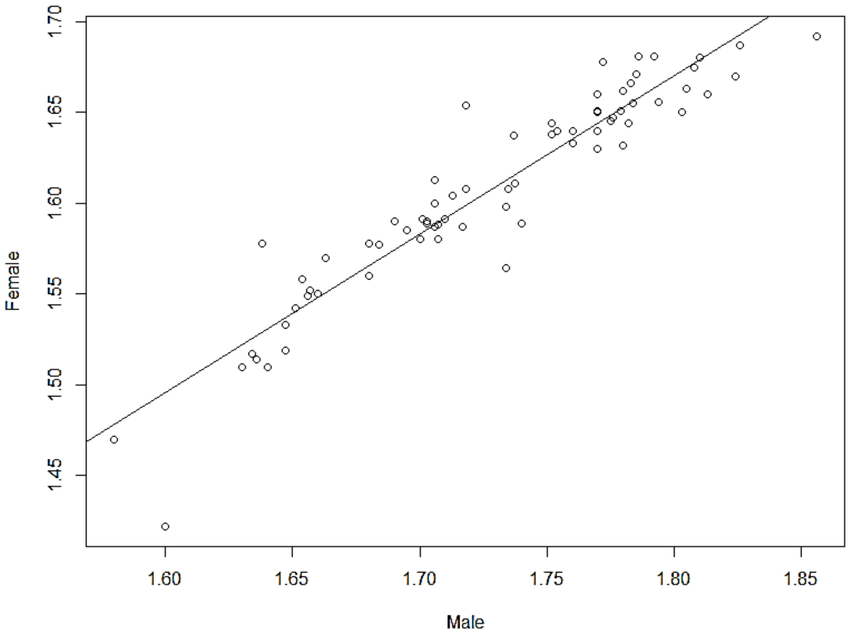
```
 ***Regression Model with Segmented Relationship(s)***

Call:
segmented.lm(obj = reg.ht, seg.Z = ~mht, psi = 1.6)

Estimated Break-Point(s):
    Est.  St.Err
1.66400 0.01238

t value for the gap-variable(s) V:  0

Meaningful coefficients of the linear terms:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -0.8743     0.3392  -2.577   0.0122 *
mht           1.4642     0.2069   7.077  1.1e-09 ***
U1.mht       -0.6994     0.2141  -3.267       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.01761 on 67 degrees of freedom
Multiple R-Squared: 0.9061,  Adjusted R-squared: 0.9019
```

$$fht = -0.8743 + 1.4642 * mht - 0.6994 * 1_{mht<1.664} * mht$$

Piecewise Regression Using Package(Cont.)

- Plotting the result





```
> plot(ht)
> plot(reg.seg1, add=T)
```

```
> plot(ht)
> abline(coef(reg.ht))
```

Piecewise Regression Using Package(Cont.)

- Piecewise regression with multiple breakpoints?

```
> reg.ht = lm(fht ~ mht)
> reg.seg1 = segmented(reg.ht, seg.Z = ~mht, psi=1.73)
> reg.seg2 = segmented(reg.ht, seg.Z = ~mht, psi=c(1.65,1.73))
```