# MSDS 596 Regression & Time Series

## Lecture 07 Transformation and Model Selection

Department of Statistics
Rutgers University

Nov 14, 2022

# Model Selection

We would like to select a subset of predictors.

- Why not use all of them? Occam's Razor: among several plausible explanations for a phenomenon, the simplest is the best.
- Bias and variance trade-off:
  - Will miss the true function if only few predictors are included (more bias).
  - Will introduce additional variation if too many predictors are used (more variance).
- Need some other criterion for model selection: residual sum of squares (and $R^2$) are always in favor of more predictors (recall homework question)
- A principle: respect the hierarchy of higher- vs lower-order terms (e.g. polynomial regression; interactions)

# Step-wise selection

- Backward elimination.
  - Select a size $\alpha$: 5%, 15%, etc. Usually higher for better prediction performance.
  - Start with the full model with $p$ predictors. Perform a $t$-test for each predictor, and obtain $p$ corresponding $p$-values. Remove the least significant predictor with the largest $p$-value, provided that it is larger than $\alpha$.
  - Refit the model, and repeat the preceding step, until all the predictors are significantly nonzero at the level $\alpha$.

# Step-wise selection

- Forward selection.
  - Select a size $\alpha$: 5%, 15%, etc.
  - Start with the null model with only the intercept. Add a single predictor to the model, and test whether the corresponding coefficient is nonzero. There are $p$ tests to be carried out. If any $p$-value from them is smaller than $\alpha$, then add the predictor with the smallest $p$-value to the model.
  - Now there are $p - 1$ predictors left. Add each single one of them to the model, and test its coefficients. Similarly, if the smallest $p$-value (there are $p - 1$ of them in this step) is less than $\alpha$, add the corresponding predictor to the model.
  - Repeat this procedure until non of the remaining predictor will have a coefficient that is significantly nonzero at level $\alpha$.

# Example: US Census Bureau 1977

Data collected from U.S. Census Bureau on the 50 states from the 1970s.

| | Population | Income | Illiteracy | Life.Exp | Murder | HS.Grad | Frost | Area |
|---|---|---|---|---|---|---|---|---|
| AL | 3615 | 3624 | 2.1 | 69.05 | 15.1 | 41.3 | 20 | 50708 |
| AK | 365 | 6315 | 1.5 | 69.31 | 11.3 | 66.7 | 152 | 566432 |
| AZ | 2212 | 4530 | 1.8 | 70.55 | 7.8 | 58.1 | 15 | 113417 |

# Example: US Census Bureau 1977

We use life expectancy as response and the rest as predictors.

```
> lmod <- lm(Life.Exp ~ ., statedata)
> summary(lmod)

              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.094e+01  1.748e+00  40.586  < 2e-16 ***
Population   5.180e-05  2.919e-05   1.775   0.0832 .
Income      -2.180e-05  2.444e-04  -0.089   0.9293
Illiteracy   3.382e-02  3.663e-01   0.092   0.9269
Murder      -3.011e-01  4.662e-02  -6.459 8.68e-08 ***
HS.Grad      4.893e-02  2.332e-02   2.098   0.0420 *
Frost       -5.735e-03  3.143e-03  -1.825   0.0752 .
Area        -7.383e-08  1.668e-06  -0.044   0.9649
---
Residual standard error: 0.7448 on 42 degrees of freedom
Multiple R-squared:  0.7362,^^IAdjusted R-squared:  0.6922
F-statistic: 16.74 on 7 and 42 DF,  p-value: 2.534e-10
```

Backward elimination. At each stage, remove the predictor with the largest p-value over $\alpha = 0.05$. Area is the first to go.

# Example: US Census Bureau 1977 - Backward elimination

Backward elimination. At each stage, remove the predictor with the largest p-value over $\alpha = 0.05$. Area is the first to go.

```
> lmod <- update(lmod, . ~ . - Area)
> summary(lmod)
```

```
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.099e+01  1.387e+00  51.165  < 2e-16 ***
Population   5.188e-05  2.879e-05   1.802   0.0785 .
Income      -2.444e-05  2.343e-04  -0.104   0.9174
Illiteracy   2.846e-02  3.416e-01   0.083   0.9340
Murder      -3.018e-01  4.334e-02  -6.963 1.45e-08 ***
HS.Grad      4.847e-02  2.067e-02   2.345   0.0237 *
Frost       -5.776e-03  2.970e-03  -1.945   0.0584 .
```

Next one up is Illiteracy.

# Example: US Census Bureau 1977 - Backward elimination

Backward elimination. At each stage, remove the predictor with the largest p-value over $\alpha = 0.05$. Area is the first to go.

```
> lmod <- update(lmod, . ~ . - Area)
> summary(lmod)

              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.099e+01  1.387e+00  51.165  < 2e-16 ***
Population   5.188e-05  2.879e-05   1.802   0.0785 .
Income      -2.444e-05  2.343e-04  -0.104   0.9174
Illiteracy   2.846e-02  3.416e-01   0.083   0.9340
Murder      -3.018e-01  4.334e-02  -6.963 1.45e-08 ***
HS.Grad      4.847e-02  2.067e-02   2.345   0.0237 *
Frost       -5.776e-03  2.970e-03  -1.945   0.0584 .
```

Next one up is Illiteracy.

# Example: US Census Bureau 1977 - Backward elimination

```
> lmod <- update(lmod, . ~ . - Illiteracy)
> summary(lmod)

              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.107e+01  1.029e+00  69.067  < 2e-16 ***
Population   5.115e-05  2.709e-05   1.888   0.0657 .
Income      -2.477e-05  2.316e-04  -0.107   0.9153
Murder      -3.000e-01  3.704e-02  -8.099 2.91e-10 ***
HS.Grad      4.776e-02  1.859e-02   2.569   0.0137 *
Frost       -5.910e-03  2.468e-03  -2.395   0.0210 *
```

Next one up is Income.

# Example: US Census Bureau 1977 - Backward elimination

```
> lmod <- update(lmod, . ~ . - Illiteracy)
> summary(lmod)

              Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.107e+01  1.029e+00  69.067  < 2e-16 ***
Population   5.115e-05  2.709e-05   1.888   0.0657 .
Income      -2.477e-05  2.316e-04  -0.107   0.9153
Murder      -3.000e-01  3.704e-02  -8.099 2.91e-10 ***
HS.Grad      4.776e-02  1.859e-02   2.569   0.0137 *
Frost       -5.910e-03  2.468e-03  -2.395   0.0210 *
```

Next one up is Income.

# Example: US Census Bureau 1977 - Backward elimination

```
> lmod <- update(lmod, . ~ . - Income)
> summary(lmod)

             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 *
```

Next one up is Population, although its p-value is close to the critical value $\alpha = 5\%$ so it's a close call.

# Example: US Census Bureau 1977 - Backward elimination

```
> lmod <- update(lmod, . ~ . - Income)
> summary(lmod)

              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 *
```

Next one up is Population, although its p-value is close to the critical value $\alpha = 5\%$ so it's a close call.

# Example: US Census Bureau 1977 - Backward elimination

```
> lmod <- update(lmod, . ~ . - Population)
> summary(lmod)

             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 **
---
Residual standard error: 0.7427 on 46 degrees of freedom
Multiple R-squared:  0.7127,^^IAdjusted R-squared:  0.6939
F-statistic: 38.03 on 3 and 46 DF,  p-value: 1.634e-12
```

Notice that the multiple $R^2$ for this model is 0.7127, whereas the full model $R^2$ is 0.7362.

# Example: US Census Bureau 1977 - Backward elimination

Note. Again, variables removed from the model may still be related to the response. For example, even if we removed Illiteracy early on, a simpler model using it as a predictor may still be significant:

```
> summary(lm(Life.Exp ~ Illiteracy+Murder+Frost, statedata))

             Estimate Std. Error t value Pr(>|t|)
(Intercept) 74.556717   0.584251 127.611  < 2e-16 ***
Illiteracy  -0.601761   0.298927  -2.013  0.04998 *
Murder      -0.280047   0.043394  -6.454 6.03e-08 ***
Frost       -0.008691   0.002959  -2.937  0.00517 **
---
Residual standard error: 0.7911 on 46 degrees of freedom
Multiple R-squared:  0.6739,^^IAdjusted R-squared:  0.6527
F-statistic: 31.69 on 3 and 46 DF,  p-value: 2.915e-11
```

# Caveats of testing-based procedures

- Can miss the "optimal" model because variables are added/dropped one at a time.
- $p$-values used in the procedure should not be treated too literally. Lots of multiple testing that was not accounted for properly.
    - Selective inference: a literature that attempts to address multiple and sequential testing in model selection.
- Stepwise variable selection tends to pick models that are smaller than desirable for prediction purposes.
- Variables that are dropped can still be correlated with the response. While they provide little additional explanatory effect beyond those variables already included in the model, it would be wrong to say that these variables are unrelated to the response.
- Any variable selection method must be understood in context of the underlying purpose of the investigation.

# Criterion-based procedures

- Adjusted $R^2$ (written $R_a^2$):

$$R_a^2 = 1 - \frac{\text{RSS}/(n-p-1)}{\text{Total SS}/(n-1)} = 1 - \frac{n-1}{n-p-1}(1-R^2) = 1 - \frac{\hat{\sigma}^2_{\text{model}}}{\hat{\sigma}^2_{\text{null}}}.$$

  Choose the model with largest $R_a^2$.
- The regular $R^2 = 1 - RSS/TSS$, which increases whenever a predictor is added.
- In comparison, adding a predictor will only increase $R_a^2$ if it has some predictive value.

# Information criteria

- Akaike Information Criterion (AIC):

$$AIC = -2\log L\left(\hat{\beta}\right) + 2\left(p+1\right) \overset{\text{(for regression)}}{=} n\log\left(RSS/n\right) + 2\left(p+1\right) + \text{const.}$$

- Bayes Information Criterion (BIC):

$$BIC = -2\log L\left(\hat{\beta}\right) + \left(p+1\right)\cdot\log(n)$$

- BIC penalizes larger models more heavily, and tend to prefer smaller models in comparison to AIC.

# Criterion-based procedures

- Mallows's $C_p$:

$$C_p = \frac{\text{RSS}}{\check{\sigma}^2} + 2(p+1) - n,$$

  where
  - RSS is obtained by the model to be evaluated.
  - $\check{\sigma}^2$ is the estimate of $\sigma^2$ obtained by the saturated model (i.e. the one with all predictors).
- $C_p$ is an estimate of the average mean squared prediction error

$$\frac{1}{\sigma^2} \sum_i \mathbb{E}\left(\hat{y}_i - \mathbb{E}\left(y_i\right)\right)^2,$$

  which should be small if the model predicts well (in sample).
- We desire models with a small number of predictors, and $C_p$ less or equal to the number of predictors (including intercept).

# Example: savings data

```
> data(savings, package='faraway')
> head(savings, 3)
             sr pop15 pop75     dpi ddpi
Australia 11.43 29.35  2.87 2329.68 2.87
Austria   12.07 23.32  4.41 1507.99 3.93
Belgium   13.17 23.80  4.43 2108.47 3.82
```

Fit a full model with savings rate as response:

```
> out1 <- lm(sr ~ pop15+pop75+dpi+ddpi,savings); summary(out1)

              Estimate Std. Error t value Pr(>|t|)
(Intercept) 28.5660865  7.3545161   3.884 0.000334 ***
pop15       -0.4611931  0.1446422  -3.189 0.002603 **
pop75       -1.6914977  1.0835989  -1.561 0.125530
dpi         -0.0003369  0.0009311  -0.362 0.719173
ddpi         0.4096949  0.1961971   2.088 0.042471 *
```

# Example: savings data

```
> data(savings, package='faraway')
> head(savings, 3)
             sr pop15 pop75     dpi ddpi
Australia 11.43 29.35  2.87 2329.68 2.87
Austria   12.07 23.32  4.41 1507.99 3.93
Belgium   13.17 23.80  4.43 2108.47 3.82
```

Fit a full model with savings rate as response:

```
> out1 <- lm(sr ~ pop15+pop75+dpi+ddpi,savings); summary(out1)

              Estimate Std. Error t value Pr(>|t|)
(Intercept) 28.5660865  7.3545161   3.884 0.000334 ***
pop15       -0.4611931  0.1446422  -3.189 0.002603 **
pop75       -1.6914977  1.0835989  -1.561 0.125530
dpi         -0.0003369  0.0009311  -0.362 0.719173
ddpi         0.4096949  0.1961971   2.088 0.042471 *
```

# Example: savings data

...and compare with two reduced models:

```
> out2=update(out1, . ~ . - dpi); summary(out2)
lm(formula = sr ~ pop15 + pop75 + ddpi, data = savings)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  28.1247     7.1838   3.915 0.000297 ***
pop15        -0.4518     0.1409  -3.206 0.002452 **
pop75        -1.8354     0.9984  -1.838 0.072473 .
ddpi          0.4278     0.1879   2.277 0.027478 *


> out3=update(out2, . ~ . - pop75); summary(out3)
lm(formula = sr ~ pop15 + ddpi, data = savings)
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 15.59958    2.33439   6.682 2.48e-08 ***
pop15       -0.21638    0.06033  -3.586 0.000796 ***
ddpi         0.44283    0.19240   2.302 0.025837 *
```

# Example: savings data

...and compare with two reduced models:

```
> out2=update(out1, . ~ . - dpi); summary(out2)
lm(formula = sr ~ pop15 + pop75 + ddpi, data = savings)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  28.1247     7.1838   3.915 0.000297 ***
pop15        -0.4518     0.1409  -3.206 0.002452 **
pop75        -1.8354     0.9984  -1.838 0.072473 .
ddpi          0.4278     0.1879   2.277 0.027478 *
```

```
> out3=update(out2, . ~ . - pop75); summary(out3)
lm(formula = sr ~ pop15 + ddpi, data = savings)
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 15.59958    2.33439   6.682 2.48e-08 ***
pop15       -0.21638    0.06033  -3.586 0.000796 ***
ddpi         0.44283    0.19240   2.302 0.025837 *
```

# Example: savings data - model selection

Compare their adjusted $R^2$, AIC and BIC:

```
> print(c(summary(out1)$adj.r.squared,
          summary(out2)$adj.r.squared,
          summary(out3)$adj.r.squared))
[1] 0.2796525 0.2932620 0.2574811

> print(c(AIC(out1),AIC(out2),AIC(out3)))
[1] 282.1961 280.3414 281.8861

> print(c(BIC(out1),BIC(out2),BIC(out3)))
[1] 293.6683 289.9015 289.5342
```

# Example: savings data - model selection

Compare their adjusted $R^2$, AIC and BIC:

```
> print(c(summary(out1)$adj.r.squared,
          summary(out2)$adj.r.squared,
          summary(out3)$adj.r.squared))
[1] 0.2796525 0.2932620 0.2574811

> print(c(AIC(out1),AIC(out2),AIC(out3)))
[1] 282.1961 280.3414 281.8861

> print(c(BIC(out1),BIC(out2),BIC(out3)))
[1] 293.6683 289.9015 289.5342
```

# Cross Validation

$K$-fold Cross validation:

- Randomly partition data into $K$ groups. Use $K - 1$ groups to fit the model and use the fitted model to predict the last group and obtain prediction errors.

# Example: savings data - K-fold cross validation

Use five-fold cross-validation to estimate prediction error (function `cv.glm()` from the `boot` package):

```
> set.seed(123)
> print(cv.glm(data=savings,glm(sr~pop15+pop75+dpi+ddpi,
                 data=savings),K=5)$delta[1])
[1] 16.46066

> print(cv.glm(data=savings,glm(sr~pop15+pop75+ddpi,
                 data=savings), K=5)$delta[1])
[1] 15.16503

> print(cv.glm(data=savings,glm(sr~pop15+ddpi,
                 data=savings), K=5)$delta[1])
[1] 17.57572
```

# Finding candidate models

- All subsets. Preferred, but may be computationally impossible!

| $k$ | 1 | 10 | 20 | 30 | 40 |
|---|---|---|---|---|---|
| # models | 1 | 1K | 1M | 1B | 1T |
| time | 1/100s | 0.17m | 2.9h | 124d | 348y |

  (Moore's law: computer speed doubles every two years, allowing one more variable.)

- Forward search. Start from just the intercept. Each step, try every predictor not in the model and add the one that reduces RSS the most.
- Backward search. Start with full model. Each step, try every predictor in the model and remove the one that increases RSS the least.
- Sequential replacement. For each given size $k$, starting from some initial choice of a subset of size $k$ (may be random). Each step, replace one predictor in the subset by some one not in the subset, choose the replacement that reduces the RSS the most. Keep doing until no replacement reduces RSS further.

# Example: US Census Bureau 1977 - candidate model

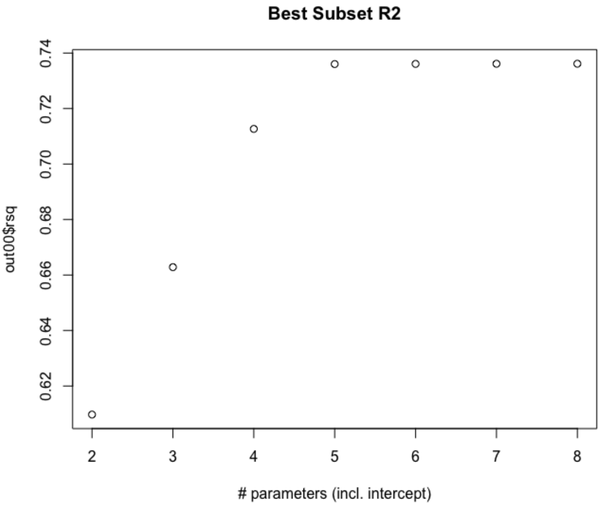
We use an 'exhaustive' search on the US Census Bureau data, predicting life expectancy. Can also use 'forward', 'backward', and 'seqrep' (sequential replacement) search methods.

```
> library(leaps)
> out0=regsubsets(Life.Exp~., data=statedata,
                  method='exhaustive', nvmax=7)
> out00=summary(out0)
> out00$outmat
```

```
         Population Income Illiteracy Murder HS.Grad Frost Area
1  ( 1 ) " "        " "    " "        "*"    " "     " "   " "
2  ( 1 ) " "        " "    " "        "*"    "*"     " "   " "
3  ( 1 ) " "        " "    " "        "*"    "*"     "*"   " "
4  ( 1 ) "*"        " "    " "        "*"    "*"     "*"   " "
5  ( 1 ) "*"        "*"    " "        "*"    "*"     "*"   " "
6  ( 1 ) "*"        "*"    "*"        "*"    "*"     "*"   " "
7  ( 1 ) "*"        "*"    "*"        "*"    "*"     "*"   "*"
```

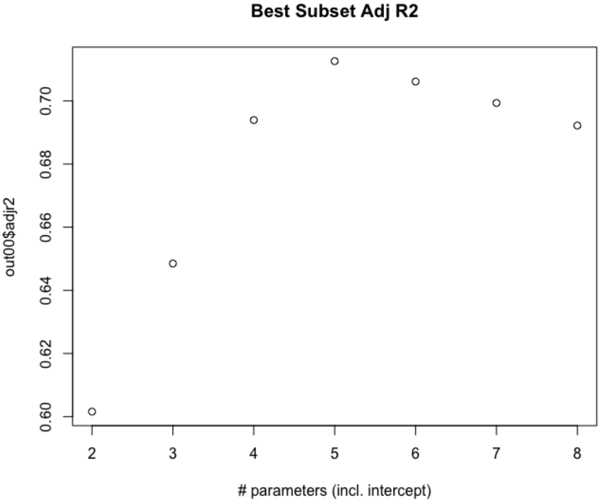
# Example: US Census Bureau 1977 - candidate model

Best $R^2$ among each subset of models with $2, 3, \ldots, 8$ parameters:



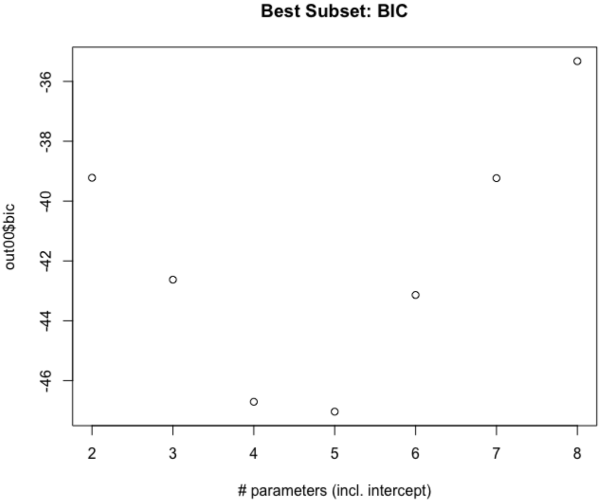
# Example: US Census Bureau 1977 - candidate model

Best adjusted $R^2$ among each subset of models with $2, 3, \ldots, 8$ parameters:



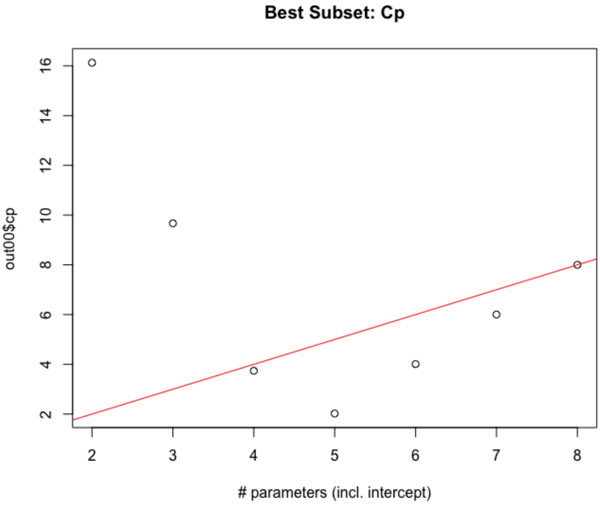
# Example: US Census Bureau 1977 - candidate model

Best BIC among each subset of models with $2, 3, \ldots, 8$ parameters:



# Example: US Census Bureau 1977 - candidate model

Best Mallows's $C_p$ among each subset of models with $2, 3, \ldots, 8$ parameters:



# Example: US Census Bureau 1977 - candidate model

The criteria we looked at consistently pick the model with the intercept plus four predictors:

```
> out00$which[4,]
 (Intercept)  Population      Income  Illiteracy      Murder     HS.G
        TRUE        TRUE       FALSE       FALSE        TRUE        T
      Frost         Area
       TRUE        FALSE
```

# Early stopping rule

- Choose a criterion, usually AIC or BIC. (Will use BIC as an example.)
- Forward search. Start from nothing. Each step, try every predictor not in the model and add the one that reduces the BIC the most. Stop when no predictor outside the model is able to reduce the BIC.
- Backward search. Start with the full model. Each step, try every predictor in the model and remove the one that increases the BIC the most. Stop when dropping any predictor from the current model increases BIC.
- Hybrid search. In each step with a current subset (model), either dropping a predictor or adding one predictor. Choose the one which reduces the BIC the most. Stop if none of them reduces the BIC.

# Example: US Census Bureau 1977 - early stopping

Forward search:

```
> n = nrow(statedata)
> out.null=lm(Life.Exp~1, data=statedata)
> full=formula(lm(Life.Exp~.,statedata))
> out.forward=step(out.null,scope=list(lower=~1,upper=full),
                 k=log(n),direction="forward",trace=FALSE)

> out.forward$coefficients
  (Intercept)        Murder       HS.Grad         Frost    Population
 7.102713e+01 -3.001488e-01  4.658225e-02 -5.943290e-03  5.013998e-05
```

Note. The $k$ argument in function `step()` specifies the criterion. $k$=2 uses AIC, which is the default, and $k = \log(n)$ uses BIC ($n$ is sample size).

# Example: US Census Bureau 1977 - early stopping

Forward search:

```
> out.forward$coefficients
  (Intercept)        Murder       HS.Grad         Frost    Population
 7.102713e+01 -3.001488e-01  4.658225e-02 -5.943290e-03  5.013998e-05
```

Backward search:

```
> out.full=lm(Life.Exp~.,statedata)
> out.backward=step(out.full,scope=list(lower=~1,upper=full),
                  direction="backward",trace=FALSE,k=log(n))
> out.backward$coefficients
  (Intercept)    Population        Murder       HS.Grad         Frost
 7.102713e+01  5.013998e-05 -3.001488e-01  4.658225e-02 -5.943290e-03
```

Hybrid search:

```
> out.both=step(out.full,scope=list(lower=~1,upper=full),
              direction="both",trace=FALSE,k=log(n))
> out.both$coefficients
  (Intercept)    Population        Murder       HS.Grad         Frost
 7.102713e+01  5.013998e-05 -3.001488e-01  4.658225e-02 -5.943290e-03
```

# Example: US Census Bureau 1977 - early stopping

Forward search:

```
> out.forward$coefficients
  (Intercept)        Murder       HS.Grad         Frost    Population
 7.102713e+01 -3.001488e-01  4.658225e-02 -5.943290e-03  5.013998e-05
```

Backward search:

```
> out.full=lm(Life.Exp~.,statedata)
> out.backward=step(out.full,scope=list(lower=~1,upper=full),
                  direction="backward",trace=FALSE,k=log(n))
> out.backward$coefficients
  (Intercept)    Population        Murder       HS.Grad         Frost
 7.102713e+01  5.013998e-05 -3.001488e-01  4.658225e-02 -5.943290e-03
```

Hybrid search:

```
> out.both=step(out.full,scope=list(lower=~1,upper=full),
              direction="both",trace=FALSE,k=log(n))
> out.both$coefficients
  (Intercept)    Population        Murder       HS.Grad         Frost
 7.102713e+01  5.013998e-05 -3.001488e-01  4.658225e-02 -5.943290e-03
```

# Example: US Census Bureau 1977 - early stopping

Forward search:

```
> out.forward$coefficients
  (Intercept)        Murder       HS.Grad        Frost    Population
 7.102713e+01 -3.001488e-01  4.658225e-02 -5.943290e-03  5.013998e-05
```

Backward search:

```
> out.full=lm(Life.Exp~.,statedata)
> out.backward=step(out.full,scope=list(lower=~1,upper=full),
                  direction="backward",trace=FALSE,k=log(n))
> out.backward$coefficients
  (Intercept)    Population        Murder       HS.Grad         Frost
 7.102713e+01  5.013998e-05 -3.001488e-01  4.658225e-02 -5.943290e-03
```

Hybrid search:

```
> out.both=step(out.full,scope=list(lower=~1,upper=full),
              direction="both",trace=FALSE,k=log(n))
> out.both$coefficients
  (Intercept)    Population        Murder       HS.Grad         Frost
 7.102713e+01  5.013998e-05 -3.001488e-01  4.658225e-02 -5.943290e-03
```