Lecture 12:
Variable Selection
STAT 452, Spring 2021

Variable Selection

The process of selecting which predictors to include in a model is referred to as **variable selection**.

Why should we perform variable selection?

(1) *Interpretation*: A model with fewer variables is easier to interpret and explain.

(2) *Prediction*: A model with too many predictors can over-fit the training data, and perform poorly when making predictions on the test set.

(3) *Cost*: It is cheaper to collect data on fewer variables.

Backwards Elimination

One of the simplest methods for variable selection:

(1) Start with the full model that includes all the predictor variables.

(2) Identify the predictor with the largest p-value (this is the predictor that is the least statistically significant in the model).

    (a) If the p-value is large (say, greater than 10%), remove that predictor and refit the model; then return to step 2.

    (b) If the p-value is small (say, smaller than 10%), stop since all the remaining predictors are significant.

Other cut-offs for the p-value can be used such as 5%. Generally, smaller cut-offs result in more predictors being eliminated.

Example: Census Data (1970)

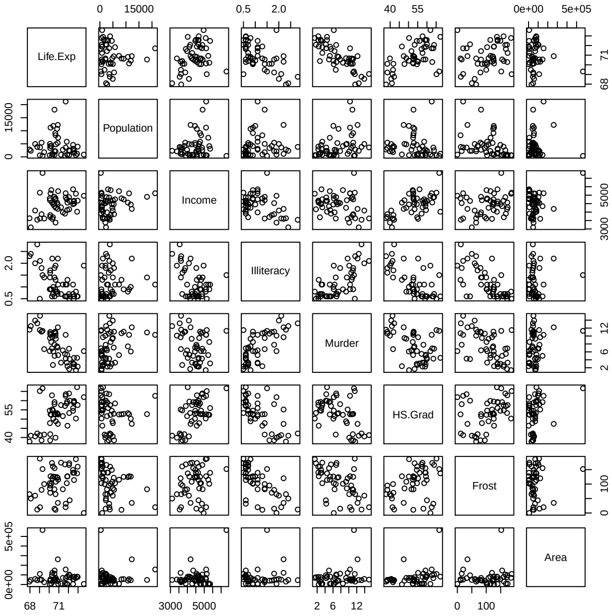
- We illustrate this method using U.S. Census Bureau data on 50 states from the 1970s. This is a base R data set (no package necessary).[1]
- The response variable is `Life.Exp`, life expectancy in years.
- The predictors are:
  - `Population`: population estimate
  - `Income`: per capita income
  - `Illiteracy`: percent illiterate
  - `Murder`: murder rate per 100,000 population
  - `HS.Grad`: percent high-school graduates
  - `Frost`: mean number of days with minimum temperature below freezing in capital city
  - `Area`: land area in square miles

[1] Example from Julian Faraway, *Linear Models with R*, 1st edition, pp. 122–125

```
> statedata <- data.frame(state.x77, row.names=state.abb)

> head(statedata)
   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
AR       2110   3378        1.9    70.66   10.1    39.9    65  51945
CA      21198   5114        1.1    71.71   10.3    62.6    20 156361
CO       2541   4884        0.7    72.06    6.8    63.9   166 103766

> dim(statedata)
[1] 50  8
```



Life.Exp
Population
Income
Illiteracy
Murder
HS.Grad
Frost
Area
0
15000
0.5
2.0
40
55
0e+00
5e+05
68
71
3000
5000
2
6
12
0
100

```
> lm1 <- lm(Life.Exp ~ ., data=statedata)
> summary(lm1)

Coefficients:
              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
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7448 on 42 degrees of freedom
Multiple R-squared:  0.7362,Adjusted R-squared:  0.6922
F-statistic: 16.74 on 7 and 42 DF,  p-value: 2.534e-10
```

```
> lm2 <- update(lm1, ~ . - Area)
> summary(lm2)

Coefficients:
               Estimate  Std. Error t value Pr(>|t|)
(Intercept) 70.98931852  1.38745441  51.165  < 2e-16 ***
Population   0.00005188  0.00002879   1.802   0.0785 .
Income      -0.00002444  0.00023429  -0.104   0.9174
Illiteracy   0.02845881  0.34163295   0.083   0.9340
Murder      -0.30182314  0.04334432  -6.963 1.45e-08 ***
HS.Grad      0.04847232  0.02066727   2.345   0.0237 *
Frost       -0.00577576  0.00297023  -1.945   0.0584 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7361 on 43 degrees of freedom
Multiple R-squared:  0.7361,Adjusted R-squared:  0.6993
F-statistic: 19.99 on 6 and 43 DF,  p-value: 5.362e-11
```

```
> lm3 <- update(lm2, ~ . - Illiteracy, data=statedata)
> summary(lm3)

Coefficients:
               Estimate  Std. Error t value Pr(>|t|)
(Intercept) 71.06575094  1.02894145  69.067  < 2e-16 ***
Population   0.00005115  0.00002709   1.888   0.0657 .
Income      -0.00002477  0.00023160  -0.107   0.9153
Murder      -0.30000770  0.03704182  -8.099 2.91e-10 ***
HS.Grad      0.04775797  0.01859079   2.569   0.0137 *
Frost       -0.00590986  0.00246778  -2.395   0.0210 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7277 on 44 degrees of freedom
Multiple R-squared:  0.7361,Adjusted R-squared:  0.7061
F-statistic: 24.55 on 5 and 44 DF,  p-value: 1.019e-11
```

```
> lm4 <- update(lm3, ~ . - Income, data=statedata)
> summary(lm4)

Coefficients:
               Estimate  Std. Error t value Pr(>|t|)
(Intercept) 71.02712853  0.95285296  74.542  < 2e-16 ***
Population   0.00005014  0.00002512   1.996  0.05201 .
Murder      -0.30014880  0.03660946  -8.199 1.77e-10 ***
HS.Grad      0.04658225  0.01482706   3.142  0.00297 **
Frost       -0.00594329  0.00242087  -2.455  0.01802 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7197 on 45 degrees of freedom
Multiple R-squared:  0.736,Adjusted R-squared:  0.7126
F-statistic: 31.37 on 4 and 45 DF,  p-value: 1.696e-12
```

Drawbacks

Backwards selection is a dated method that has some drawbacks:

- By dropping variables one-at-a-time it is possible to miss the "optimal" model.
- The p-values will overstate the importance of the remaining predictors. The removal of less significant predictors tends to increase the significance of the remaining predictors.

Model Selection Criterion

- The model containing all of the predictors will always have the smallest RSS, or equivalently the largest R^2. Therefore, RSS and R^2 are not suitable for selecting the best model among a collection of models with different numbers of predictors.
- Model selection criterion can be used compare regression models with different numbers of predictors. Here we consider the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC). These criteria penalize for increasing the number of predictors variables.
- Variable selection algorithms that use model selection criterion can search a much wider collection of potential models than backwards elimination. The repeated use hypothesis testing in backwards elimination is also dubious, and criterion such as the AIC have better theoretical justification.

Model Selection Criterion: AIC

$$\text{AIC} = n \log(\text{RSS}/n) + 2p$$

- n is the sample size, and p is the number of predictor variables
- The AIC can be used to compare regression models, estimated using the same data set, that differ in the number of predictors.
- The *smaller* the value of the AIC the better the model. So if we are comparing several models, we should select the model with the smallest AIC value.
- The AIC provides a balance between goodness-of-fit (small RSS) and model complexity (large p).

Remarks:

- Note that the AIC cannot be used to compare regression models with different response transformations (e.g., $\log(Y)$ versus Y). However, it can be used to compare models with different predictor transformations.
- Theoretical justification for the AIC comes from maximum-likelihood estimation and information theory (beyond the scope of this class).

Model Selection Criterion: BIC

An alternative to the AIC, is the Bayesian Information Criterion (BIC) which is defined as

$$\mathrm{BIC} = n \log(\mathrm{RSS}/n) + p \log(n)$$

- When $n \geq 8$, $\log(n) > 2$ and so the penalty term in the BIC is greater than the penalty term in the AIC.
- Consequently, the BIC tends to select models with fewer predictors than the AIC.

Backwards Stepwise Selection using AIC

Description of procedure:

1. Start with the full model will all p predictor variables.
2. At each step, the predictor variable is removed that results in the largest reduction in the AIC.
3. The procedure stops when the AIC starts to increase, or all variables have been removed.

Backwards Stepwise Selection using AUC

- The `step()` function can be used to easily implement backwards stepwise selection in R. By default the `step()` function uses the AIC.

- Backwards stepwise selection searches through $p + (p-1) + (p-2) \cdots + 1 = p(p+1)/2$ models. So it searches through a larger number of models than backwards elimination using p-values.

Example: Baseball Data

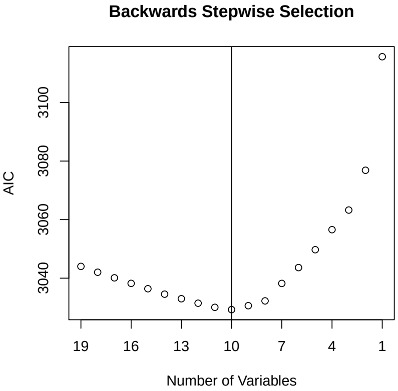
- To illustrate backwards stepwise selection we use Major League Baseball Data from the 1986 and 1987 seasons. The data set is called `Hitters`, and can be accessed from the `ISLR` library.
- The data set contains $n = 322$ observation of major league players. After removing some missing data there are $n = 263$ observations.
- The response variable is `Salary`, a baseball player's 1987 annual salary in thousands of dollars.
- There are 19 predictor variables related to the player's performance (e.g, `AtBat`, number of times at bat in 1986; `HmRun`, number of home runs in 1986; `CHmRun` number of home runs during his career; etc.).

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

> dim(Hitters)
[1] 322  20

# remove missing data
> Hitters2 <- na.omit(Hitters)
> dim(Hitters2)
[1] 263  20
```

Backwards stepwise selection applied to the `Hitters` data set. The plot shows the AIC of the best fitting model versus the number of predictor variables at each iteration. We see that the procedure selects a 10 predictor model (has smallest AIC).



The `step()` function can be used perform stepwise selection. The function is convenient since it outputs the selected regression model (an `lm` object).

```
> lm_full <- lm(Salary ~ ., data=Hitters2)
> lm2 <- step(lm_full)
> summary(lm2)
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  162.53544   66.90784   2.429 0.015830 *
AtBat         -2.16865    0.53630  -4.044 7.00e-05 ***
Hits           6.91802    1.64665   4.201 3.69e-05 ***
Walks          5.77322    1.58483   3.643 0.000327 ***
CAtBat        -0.13008    0.05550  -2.344 0.019858 *
CRuns          1.40825    0.39040   3.607 0.000373 ***
CRBI           0.77431    0.20961   3.694 0.000271 ***
CWalks        -0.83083    0.26359  -3.152 0.001818 **
DivisionW   -112.38006   39.21438  -2.866 0.004511 **
PutOuts        0.29737    0.07444   3.995 8.50e-05 ***
Assists        0.28317    0.15766   1.796 0.073673 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 311.8 on 252 degrees of freedom
Multiple R-squared:  0.5405,Adjusted R-squared:  0.5223
F-statistic: 29.64 on 10 and 252 DF,  p-value: < 2.2e-16
```

We can also use the step() function to select a model using the BIC by specifying the argument k=log(n) (note by default k=2 which specifies the penalty for the AIC). When using the BIC, 8 predictors are selected (less than AIC).

```
> n <- nrow(Hitters2)
> lm3 <- step(lm_full, k=log(n))
> summary(lm3)
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  117.15204   65.07016   1.800 0.072985 .
AtBat         -2.03392    0.52282  -3.890 0.000128 ***
Hits           6.85491    1.65215   4.149 4.56e-05 ***
Walks          6.44066    1.52212   4.231 3.25e-05 ***
CRuns          0.70454    0.24869   2.833 0.004981 **
CRBI           0.52732    0.18861   2.796 0.005572 **
CWalks        -0.80661    0.26395  -3.056 0.002483 **
DivisionW   -123.77984   39.28749  -3.151 0.001824 **
PutOuts        0.27539    0.07431   3.706 0.000259 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 314.7 on 254 degrees of freedom
Multiple R-squared:  0.5281,Adjusted R-squared:  0.5133
F-statistic: 35.54 on 8 and 254 DF,  p-value: < 2.2e-16
```

Remarks

- If interpretation is the goal, then automated variable selection methods should not be used as a substitute for thinking about the context of your data and the validity selected model.
  - Assess whether the signs and magnitudes of the coefficients make sense in the selected model.
  - It also might be worthwhile to assess collinearity and to manually remove or combine correlated variables.
  - Look at scatterplots of the data and residuals plots. These diagnostics might indicate problems with nonlinearity or nonconstant variance, and thus motivate transformations that fix these issues.
- If prediction is the goal, then it is recommended to do some form of cross-validation. Withhold some data and evaluate how well the model performs on that withheld, validation set.