MATH 829: Introduction to Data Mining and Analysis
Subset selection

Dominique Guillot

Departments of Mathematical Sciences
University of Delaware

February 19, 2016

Testing multiple coefficients

We saw before how to use the t-statistic to test

$$H_0 : \beta_i = 0$$
$$H_1 : \beta_i \neq 0.$$

Given $\{i_1, i_2, \dots, i_k\} \subset \{1, 2, \dots, p\}$, we want to rigorously test

$$H_0 : \beta_{i_1} = \beta_{i_2} = \cdots = \beta_{i_k} = 0$$
$$H_1 : \beta_{i_1} \neq 0 \text{ or } \beta_{i_2} \neq 0 \text{ or } \dots \text{ or } \beta_{i_k} \neq 0.$$

We use the F statistic

$$F = \frac{(\mathrm{RSS}_0 - \mathrm{RSS}_1)/(p - p_0)}{\mathrm{RSS}_1/(n-p)},$$

where

$\mathrm{RSS}_1 =$ residual sum of squares for full model,

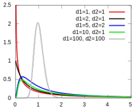
$\mathrm{RSS}_0 =$ residual sum of squares for the nested smaller model.

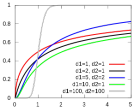
Can be seen as a measure of the *change in residual sum-of-squares per additional parameter in the bigger model.*

Testing multiple coefficients (cont.)

Under the H_0 assumption that the smaller model is correct, the F statistic has an F-distribution

$$F \sim F_{p-p_0, n-p}.$$





To test if a group of coefficients are 0:

1. Compute the F-statistic.
2. Reject H_0 for large values of the F-statistic.

Python

A simple illustration of the previous ideas.

```
import numpy as np
import statsmodels.api as sm

# Generate random data
n = 50

epsilon = np.random.randn(n,1)  # Try varying the sample size

X = np.random.randn(n,5)

y = 3*X[:,0] + 4*X[:,1] + epsilon  # Try changing coefficients

results = sm.OLS(y,X).fit()

print(results.summary())

R = [[0,0,1,0,0],
     [0,0,0,1,0],
     [0,0,0,0,1]]

print(results.f_test(R))

R = [[1,0,0,0,0],[0,1,0,0,0]]

print(results.f_test(R))
```

Python (cont.)

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                      y   R-squared:                       0.954
Model:                            OLS   Adj. R-squared:                  0.949
Method:                 Least Squares   F-statistic:                     187.2
Date:                Tue, 19 Jan 2016   Prob (F-statistic):           6.23e-29
Time:                        12:40:31   Log-Likelihood:                -78.513
No. Observations:                  50   AIC:                             167.0
Df Residuals:                      45   BIC:                             176.6
Df Model:                           5
==============================================================================
                 coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------
x1             3.3360      0.208     16.071      0.000         2.918     3.754
x2             4.0380      0.167     24.139      0.000         3.701     4.375
x3            -0.1904      0.167     -1.143      0.259        -0.526     0.145
x4             0.1282      0.186      0.689      0.495        -0.247     0.503
x5             0.1163      0.155      0.751      0.456        -0.195     0.428
==============================================================================
Omnibus:                        0.748   Durbin-Watson:                   2.074
Prob(Omnibus):                  0.688   Jarque-Bera (JB):                0.755
Skew:                          -0.002   Prob(JB):                        0.686
Kurtosis:                       2.398   Cond. No.                         1.91
==============================================================================
<F test: F=array([[ 0.76049081]]), p=[[ 0.52218257]], df_denom=45, df_num=3>
<F test: F=array([[ 390.38886666]]), p=[[  3.69709216e-29]], df_denom=45, df_num
=2>
```


Subset selection

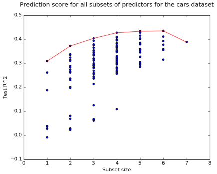
- We saw before that the OLS is the *best linear unbiased estimator* for β.
- However, biased estimators can significantly improve the performance (e.g. reduce prediction error).

We now explore various approaches that can be used to select an appropriate subset of variables in a linear regression.

Best subset selection: Given $k \in \{1, \ldots, p\}$, we find the subset of size k of $\{1, \ldots, p\}$ that minimizes the prediction error.

- Note: there are $\binom{p}{k}$ subsets of size k and 2^k possible subsets. So the procedure is only computationally feasible for small values of p.
- The leaps and bounds procedure (Furnival and Wilson, 1974) makes this feasible for p as large as 30 or 40.

Best subset selection: cars dataset

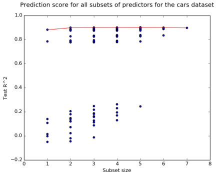


Best subset = ['Mileage','Liter','Doors','Cruise','Sound', 'Leather'].
Not included = ['Cylinder']

Best subset of 4 elements: ['Mileage','Liter','Cruise','Leather']

Best subset selection: cars dataset, Chevrolet

Restricting to Chevrolet only:



Forward- and Backward- stepwise regression

- Best subset selection performs well, but is too computationally intensive to be useful in practice.

Two natural "greedy" variants of the best subset selection technique:

- **Forward stepwise regression**: starts with the intercept $\overline{y}$, and then sequentially adds into the model the predictor that most improves the fit.
- **Backward stepwise regression**: starts with the full model, and sequentially deletes the predictor that has the least impact on the fit (smallest Z-score or t-score).

Can be used even when the number of variables is very large.
However,

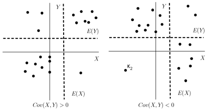
- Greedy approach: doesn't guarantee a global optimum.
- Less rigorous than other methods, less supporting theory.

Nevertheless, the stepwise approaches often return predictors similar to the predictors obtained from more complex methods with better theory.

Correlation

Recall: **Covariance** is a measure of linear dependence between random variables:

$$\mathrm{Cov}(X, Y) = E\left((X - E(X))(Y - E(Y))\right).$$



Properties:

1. $\mathrm{Cov}(\cdot, \cdot)$ is bilinear and symmetric.
2. $\mathrm{Cov}(X, X) = \mathrm{Var}(X)$.
3. $\mathrm{Cov}(X, Y) = E(XY) - E(X)E(Y)$.
4. X, Y independent $\Rightarrow \mathrm{Cov}(X, Y) = 0$.

Correlation

How can we tell if variables have a linear relationship?

The correlation (coefficient) between X and Y is given by:

$$\rho = \rho(X, Y) = \frac{\mathrm{Cov}(X, Y)}{\sqrt{\mathrm{Var}(X)\,\mathrm{Var}(Y)}}.$$

The correlation coefficient is a measure of the linear dependence between two random variables.

Theorem: Assume $\mathrm{Var}(X), \mathrm{Var}(Y) < \infty$. The correlation coefficient $\rho(X, Y)$ satisfies

$$-1 \leq \rho(X, Y) \leq 1.$$

Moreover, $\rho(X, Y) = \pm 1$ if and only if $\mathbb{P}(Y = aX + b) = 1$ for some constants a, b. In this case, $a > 0$ if $\rho(X, Y) = 1$ and $a < 0$ if $\rho(X, Y) = -1$.

Forward stagewise regression

- Start with intercept $\overline{y}$, and centered predictors with coefficients initially all 0.
- At each step the algorithm: identify the variable most correlated with the current residual.
- Compute the simple linear regression coefficient of the residual on this chosen variable, and add it to the current coefficient for that variable.
- Continued till none of the variables have correlation with the residuals.

In other words:

- $C = \emptyset$, $\hat{y}_1 = \overline{y}$, $\beta_1 = \cdots = \beta_p = 0$.
- Suppose X_{i_1} is most correlated to y.
$$C \to C \cup \{X_{i_1}\}.$$
- Solve $y - \hat{y}_1 = \alpha_{i_1} X_{i_1} + \epsilon$.
$$\beta_{i_1} \to \beta_{i_1} + \alpha_{i_1}.$$
- etc.

Forward stagewise regression (cont.)

Remarks:

1. Unlike forward-stepwise regression, none of the other variables are adjusted when a term is added to the model.
2. The process can take **more than p** steps to reach the least squares fit.
3. Historically, forward stagewise regression has been dismissed as being inefficient.
4. However, it can be quite competitive, especially in very high-dimensional problems.

Forward stagewise regression (cont.)

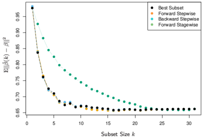


FIGURE 3.6. *Comparison of four subset-selection techniques on a simulated linear regression problem $Y = X^T\beta + \varepsilon$. There are $N = 300$ observations on $p = 31$ standard Gaussian variables, with pairwise correlations all equal to 0.85. For 10 of the variables, the coefficients are drawn at random from a $N(0, 0.4)$ distribution; the rest are zero. The noise $\varepsilon \sim N(0, 6.25)$, resulting in a signal-to-noise ratio of 0.64. Results are averaged over 50 simulations. Shown is the mean-squared error of the estimated coefficient $\hat\beta(k)$ at each step from the true β.*

ESL, Fig. 3.6