36-350: Data Mining

Handout 17
October 22, 2003

Variable selection for a linear model

As you add predictors to a model, some of the predictors you previously added may become unimportant, and should be deleted. This can be done by inspecting its p-value, the condensed sum of squares, or a partial residual plot.

Partial model—A model where one predictor has been removed, by clipping or condensing.
Clipped model—The predictor is removed by setting its coefficient to zero. Other coefficients are the same.
Condensed model—The model is re-fitted without the predictor. All coefficients may change.
Clipped partial residuals—Residuals of the clipped model.
Condensed partial residuals—Residuals of the condensed model.
In this class, "partial residual" always means "condensed partial residual."

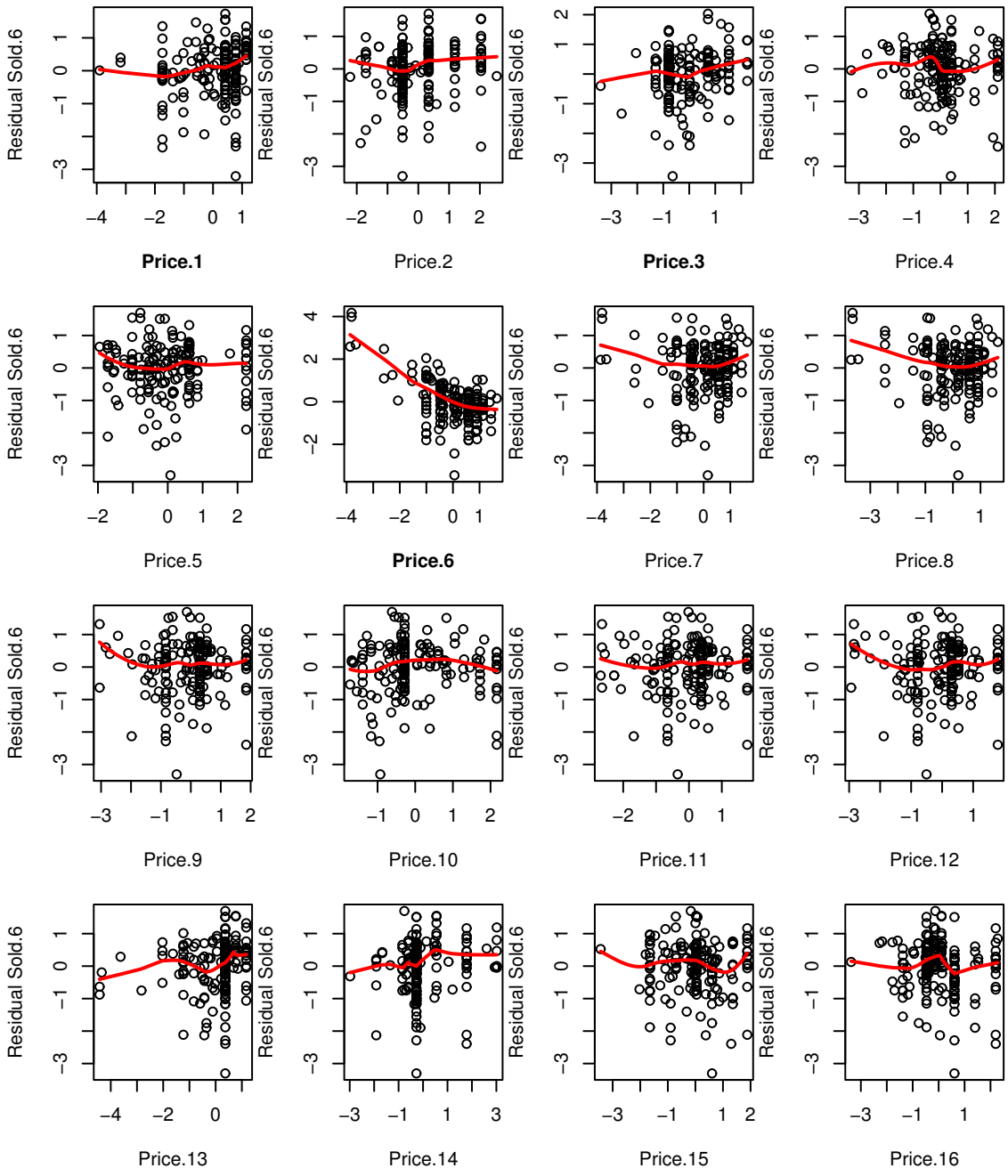
Besides making the model simpler, deleting predictors can make the model *better*, because it prevents overfitting a small data set. We should only keep predictors that have a significant impact on the predictions made by the model.

To construct a linear model with only the most important predictors:

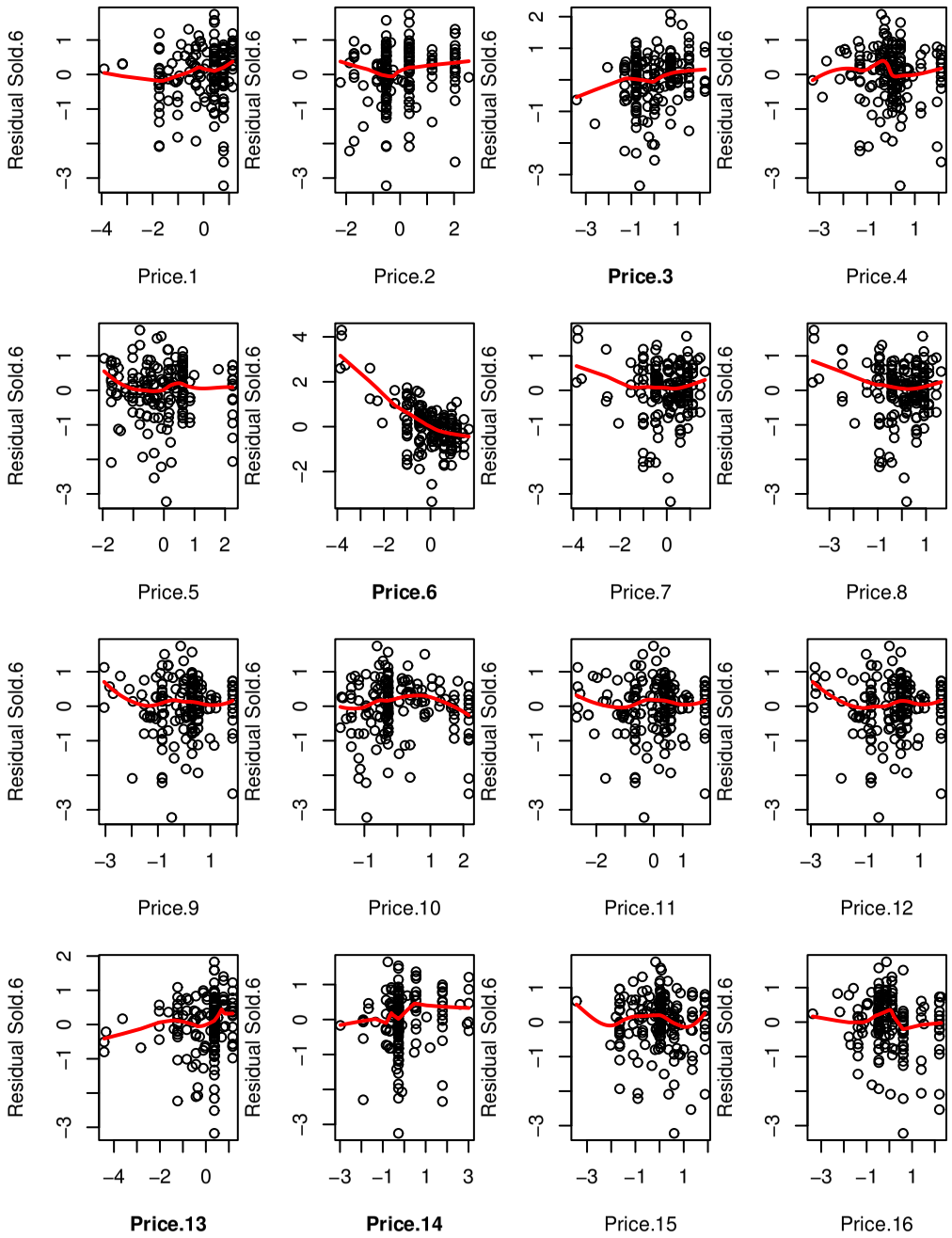
1. Plot the response versus each predictor. Add the most important to the model.
2. Plot the residuals versus each predictor not in the model. Add the most important to the model.
3. Plot the partial residuals versus each predictor in the model. Delete unimportant predictors from the model.
4. Iterate back to 2.

Partial residual plot—Each predictor is plotted versus the residuals of a condensed model with that predictor left out. The plot shows the importance of each predictor in the model (like a visual p-value), as well as what predictors need to be added.

Partial residuals of the model `Sold.6 ~ Price.1 + Price.3 + Price.6`:



Partial residuals of the model `Sold.6 ~ Price.6 + Price.3 + Price.13 + Price.14`:



A condensed model will *always* have a larger residual sum of squares. But the difference is statistically significant only when the ratio exceeds $e^{-2/n}$, where n is the number of cases. A simple way to keep track of this is to compute the AIC number of each model:

$$AIC = n \log(RSS/n) + 2(\text{number of coefficients})$$

The model with smallest AIC number is the smallest model with best expected performance. However, this result assumes:

- The response actually is linear
- There are no outliers
- There are no interactions
- The predictors are uncorrelated (!)

Because these assumptions are frequently violated, the visual method is more reliable.

Stepwise selection adds and deletes predictors in an effort to minimize the AIC number.

```
Start:  AIC= -63.51
 Sold.6 ~ Price.6
Step:  AIC= -71.54
 Sold.6 ~ Price.6 + Price.3
Step:  AIC= -76.67
 Sold.6 ~ Price.6 + Price.3 + Price.8
...
Step:  AIC= -89.7
 Sold.6 ~ Price.6 + Price.3 + Price.8 + Price.2 + Price.16 + Price.14 +
    Price.13 + Price.1

              RSS     AIC
<none>     93.697 -89.703
- Price.1  94.906 -89.474
- Price.13 95.226 -88.887
+ Price.7  93.153 -88.717
+ Price.10 93.274 -88.491
+ Price.15 93.353 -88.343
+ Price.11 93.577 -87.926
+ Price.9  93.597 -87.891
+ Price.12 93.600 -87.885
+ Price.4  93.661 -87.772
- Price.14 95.846 -87.758
+ Price.5  93.697 -87.704
```

```
- Price.6  96.285 -86.964
- Price.8  96.575 -86.440
- Price.2  97.201 -85.316
- Price.3  98.277 -83.400
- Price.16 99.644 -80.997

n = 174
```

AIC does not do well here. The chosen model has too many predictors, including `Price.8` which should be redundant given `Price.6`. But it did find `Price.3,13,14` which we know are important.

An overview of the derivation of AIC is provided by Joseph E. Cavanaugh, "Unifying the Derivations for the Akaike and Corrected Akaike Information Criteria," Statistics & Probability Letters 33:201-208, 1997. `http://citeseer.nj.nec.com/cavanaugh97unifying.html`

How to determine the coefficients of a linear model

A linear model for the response y as a function of predictors (x_1, x_2):

$$y = a_1 x_1 + a_2 x_2 + b + (\text{noise})$$

The residual is the difference between the actual response and the predicted response:

$$\text{residual} = y - a_1 x_1 - a_2 x_2 - b$$

Given training data in the form of triples (x_{i1}, x_{i2}, y_i), there are many ways to estimate the coefficients (a_1, a_2, b). **Least-squares** minimizes the residual sum of squares:

$$RSS = \sum_i (y_i - a_1 x_{i1} - a_2 x_{i2} - b)^2$$

As a function of any parameter (a_1, a_2, or b), RSS is a quadratic bowl. The minimum can be found by hill-climbing, and it is unique (no need for random restarts). **Backfitting** is a hill-climbing method:

1. Guess values for all parameters.
2. Update a_1 by solving a one-dimensional least-squares problem:

   $$RSS_1 = \sum_i ((y_i - a_2 x_{i2} - b) - a_1 x_{i1})^2 = \sum_i (r_{i1} - a_1 x_{i1})^2$$

   where r_{i1} is the **clipped partial residual** for predictor 1. Minimizing RSS_1 is easy—just set the derivative equal to zero.
3. Update a_2 by solving an analogous problem:

   $$RSS_2 = \sum_i ((y_i - a_1 x_{i1} - b) - a_2 x_{i2})^2 = \sum_i (r_{i2} - a_2 x_{i2})^2$$

4. Similarly for b.
5. Iterate back to 2.

In other words, each predictor repeatedly tries to approximate the residuals left by the other predictors. Eventually, no further progress can be made, and you are at the optimum.

However, if two predictors are identical and you are not careful, then backfitting can get stuck in a bad solution. To avoid this, you should always start with zeros. Once one predictor is added, the other will not be added. This is the basis of our visual scheme for adding predictors.