Stat 8054: Branch and Bound

Charles J. Geyer

School of Statistics
University of Minnesota

April 12, 2023

The Branch and Bound Algorithm

Furnival, G. M. and Wilson, R. W., Jr. (1974).
Regressions by leaps and bounds.
Technometrics, **16**, 499–511.
doi:10.1080/00401706.1974.10489231
Reprinted, *Technometrics*, **42**, 69–79.

Hand, D. J. (1981).
Branch and bound in statistical data analysis.
Journal of the Royal Statistical Society, Series D, (The Statistician), **30**, 1–13.
doi:10.2307/2987699

Furnival and Wilson (1974) is almost maximally unreadable but introduced branch and bound into statistics. Hand (1981) is very readable.

The Branch and Bound Algorithm (cont.)

The branch and bound algorithm originated in computer science. When a search over a huge but finite space is attempted — for example when a chess playing program searches for its next move — the branch and bound algorithm makes the search much more efficient by using bounds on the objective function to prune large parts of the search tree.

Although huge improvements are possible (if the bounds are good), generally an exponential time problem remains exponential time. So branch and bound does not allow arbitrarily large problems to be done.

Useful, but not magic.

The Branch and Bound Algorithm (cont.)

Typical use in statistics is frequentist model selection.

Consider a regression problem with p predictors and 2^p possible models when any subset of the predictors is allowed to specify a model.

Exponential time means the naive algorithm that simply fits 2^p models takes time exponential in p.

Branch and bound is also exponential time, but typically much faster, sometimes thousands of times faster.

Thus many problems that cannot be done by the naive algorithm are easily done by branch and bound. But other problems are too large for branch and bound.

Penalized Likelihood and Least Squares

The key idea for model selection is not to use least squares or maximum likelihood. They always pick the supermodel containing all submodels under consideration. This usually "overfits" the data.

Hence we minimize least squares plus a penalty or maximize log likelihood minus a penalty.

Mallows' C_p

$$
\begin{aligned}
C_p &= \frac{\text{SSResid}_p}{\hat{\sigma}^2} + 2p - n \\
&= \frac{\text{SSResid}_p - \text{SSResid}_k}{\hat{\sigma}^2} + p - (k - p) \\
&= (k - p)(F_{k-p,n-k} - 1) + p
\end{aligned}
$$

where SSResid_p is the sum of squares of residuals for model with p predictors, $\hat{\sigma}^2 = \text{SSResid}_k/(n-k)$ is the estimated error variance for the largest model under consideration with k predictors, and $F_{p,k}$ is the F statistic for the F test for comparison of these two models. If small model is correct, then $C_p \approx p$. All such models must be considered reasonably good fits.

Akaike Information Criterion (AIC)

Akaike (1973)

$$\mathrm{AIC}(m) = -2l(\hat{\theta}_m) + 2p$$

for a model m with p parameters.

Hurvich and Tsai (1989)

$$\mathrm{AIC}_c(m) = -2l(\hat{\theta}_m) + 2p + \frac{2p(p+1)}{n-p-1}$$

$$\mathrm{AIC}_c(m) = \mathrm{AIC}(m) + \frac{2p(p+1)}{n-p-1}$$

corrects for small-sample bias.

Bayes Information Criterion (BIC)

Schwarz (1978)

$$\text{BIC}(m) = -2l(\hat{\theta}_m) + p\log(n)$$

for a model m with p parameters.

Chooses much smaller models than AIC.

Consistent when true models is one of models under consideration. AIC inconsistent in this case.

R Package Leaps

Old S had an implementation of branch and bound. The function was called `leaps` after the title of Furnival and Wilson (1974). R has more or less the same thing in the `leaps` function in the `leaps` package (on-line help)

An Rweb example is given on this web page.

Bounds

To fix ideas suppose we are using AIC for model selection. In the branch and bound algorithm we need bounds for the criterion function evaluated over a set M of models that is not necessarily the whole family under consideration.

Let gcs(M) denote the greatest common submodel of all the models in M. This is not necessarily an element of M. In the regression setting where models are specified by the predictor variables they include, gcs(M) has those and only those predictors contained in all elements of M.

Let lcs(M) denote the least common supermodel of all the models in M. In the regression setting, lcs(M) has those and only those predictors contained in any element of M.

Bounds (cont.)

Let $\hat{\theta}_m$ denote the maximum likelihood estimate for model m, and let p_m denote the number of parameters for model m. Recall

$$\text{AIC}(m) = -2l(\hat{\theta}_m) + 2p_m$$

Bounds are

$$\begin{aligned} \text{AIC}(m) &\geq -2l(\hat{\theta}_{\text{lcs}(M)}) + 2p_{\text{gcs}(M)}, \qquad m \in M \\ \text{AIC}(m) &\leq -2l(\hat{\theta}_{\text{gcs}(M)}) + 2p_{\text{lcs}(M)}, \qquad m \in M \end{aligned}$$

Similar bounds are available for C_p, for BIC and for AIC_c.

Bounds (cont.)

To simplify notation say our criterion function is $F(m)$ and our upper and lower bounds are

$$F(m) \geq L(M), \qquad m \in M$$
$$F(m) \leq U(M), \qquad m \in M$$

Branch and Bound Recursive Procedure

Input data: a set M of models and a bound $l = F(m)$ for some model m not necessarily in M. Before any models have been evaluated set $l = +\infty$. Each time a model m is evaluated, if $F(m) < l$, then set $l = F(m)$.

This procedure is designed to be called many times for many different sets M, the global variable l keeps track of the lowest value of the criterion seen in all calls so far.

Branch and Bound Recursive Procedure (cont.)

Partition M giving $M_1, \ldots, M_k$.

For $1 \leq i \leq k$, if $l < L(M_i)$, then there is no point in examining any of the models in M_i further. None can be optimal.

For $1 \leq i \leq k$, if $M_i = \{m\}$, then evaluate $F(m)$ adjusting l if necessary.

For $1 \leq i \leq k$, if M_i is not a singleton, then recursively call this procedure with M_i as the given set (so it will be further partitioned).

Branch and Bound Theorem

The branch and bound algorithm is guaranteed to terminate because each step reduces the size of the largest set in the partition so eventually partitions have only one element and the recursion stops.

For each model m in the set M which is the argument to the top level call, the branch and bound algorithm is guaranteed to either evaluate $F(m)$ or prove that m is not optimal because $F(m^*) < F(m)$ for some $m^* \in M$.

Branch and Bound with Cutoff

If test for discarding M_i is $l + c < L(M_i)$, where $c > 0$ is a fixed number (the "cutoff") then branch and bound is guaranteed to evaluate every model m such that

$$F(m) \leq \inf_{m^* \in M} F(m^*) + c,$$

that is, every model with $F(m)$ within c of the optimal value.

Bayesian Model Averaging

Hoeting, J. A., Madigan, D., Raftery, A. E., and Volinsky, C. T. (1999).
Bayesian model averaging: A tutorial (with discussion).
Statistical Science, **19**, 382–417.
Corrected version available at http://www.stat.washington.edu/www/research/online/1999/hoeting.pdf.

Madigan, D. and Raftery, A. E. (1994).
Model selection and accounting for model uncertainty in graphical models using Occam's window.
Journal of the American Statistical Association, **89**, 1535–1546.

Bayesian Model Averaging (cont.)

If one is truly Bayesian and has a problem in which both models and parameters within models are uncertain, one averages over the whole posterior.

For any function $g(m, \theta)$ a function of both model m and within-model parameter θ, calculate posterior mean

$$E\{g(m, \theta) \mid \text{data}\}$$

This usually requires MCMC with dimension jumping (MHG). It is hard to implement. No available software.

Bayesian Model Averaging (cont.)

A reasonable approximation to the Right Thing (average with respect to full posterior) is

$$\frac{\sum_{m \in M} g(m, \hat{\theta}_m) e^{-\frac{1}{2}\,\mathrm{BIC}(m)}}{\sum_{m \in M} e^{-\frac{1}{2}\,\mathrm{BIC}(m)}}$$

where $\hat{\theta}_m$ is the MLE for model m.

This makes sense because $e^{-\frac{1}{2}\,\mathrm{BIC}(m)}$ is approximately the posterior probability of model m for large sample sizes and θ is near $\hat{\theta}_m$ when the sample size is large.

Occam's Window

In order to avoid sums over a huge class of models use

$$\frac{\sum_{m \in M^*} g(m, \hat{\theta}_m) e^{-\frac{1}{2}\,\mathrm{BIC}(m)}}{\sum_{m \in M^*} e^{-\frac{1}{2}\,\mathrm{BIC}(m)}} \tag{1a}$$

where

$$M^* = \left\{ m^* \in M : \mathrm{BIC}(m^*) \leq \left(\inf_{m \in M} \mathrm{BIC}(m^*) \right) + c \right\} \tag{1b}$$

Frequentist Model Averaging

Burnham, K. P. and Anderson, D. R. (2002).
Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach, 2nd ed.
New York: Springer-Verlag.

Hjort N. L. and Claeskens G. (2003).
Frequentist model average estimators.
Journal of the American Statistical Association, **98**, 879–899.
doi:10.1198/016214503000000828

Claeskens, G., and Hjort, N. L. (2008).
Model Selection and Model Averaging.
Cambridge University Press, Cambridge, England.
doi:10.1017/CBO9780511790485.

Efron, B. (2004).
The estimation of prediction error: Covariance penalties
and cross-validation (with discussion).
Journal of the American Statistical Association, **99**, 619–642.
doi:10.1198/016214504000000692

Shen, X. and Huang, H. (2006).
Optimal model assessment, selection and combination.
Journal of the American Statistical Association, **101**, 554–568.
doi:10.1198/016214505000001078

Yang, Y. (2003).
Regression with multiple candidate models: selecting or mixing?
Statistica Sinica, **13**, 783–809.

Frequentist Model Averaging (cont.)

Many different methods of frequentist model averaging. Simplest just replaces BIC in (1a) and (1b) by AIC or AIC_c.

Basically, these procedures are Bayesian if you think like a Bayesian and frequentist if you think like a frequentist.

When there is very little chance of selecting the true model — even assuming one of the models under consideration is true, which is unlikely except in simulations — selecting one model and pretending it is true is just dumb.

There never was a theorem justifying dumb model selection. People did it only because they didn't know what else to do.