A Fast Algorithm for S–estimates of Regression

M. Salibian–Barrera[1] and V.J. Yohai[2]

[1] Department of Statistics, University of British Columbia, Room 333, 6356 Agricultural Road, Vancouver, Canada V6T 1Z2

[2] Departamento de Matemáticas, Facultad de Ciencias Exactas y Naturales, Universidad de Buenos Aires, Ciudad Universitaria, Pabellón 1, 1426 Buenos Aires

Keywords: S–estimates, Fast algorithm.

Abstract

Equivariant high–breakdown point regression estimates are computationally expensive, and the corresponding algorithms become unfeasible for moderately large number of regressors.

One important advance to improve the computational speed of one such estimator is the fast–LTS algorithm of Rousseeuw and Van Driessen (2002. They proposed a modification of the subsampling algorithm for the Least Trimmed Squares (LTS) estimate, which they called *fast–LTS*, that considerably improves its performance. Given any initial value, they define the so–called "concentration step"(C–step for short) that improves the objective function. This step is applied to all the candidates obtained by subsampling, and it brings each candidate closer to the solution of the optimization problem. If the C–step is applied a sufficient (finite) number of times, a local minimum of the objective function is obtained. They show that the fast–LTS is much faster than the approximating algorithms for the LTS–estimate that do not use the C–step.

In this talk we present an algorithm for S–estimates (see Rousseeuw and Yohai, 1984) similar to the fast–LTS. This algorithm, that we call "fast–S", is based on modifying each candidate with a step that improves the S–optimality criterion, and thus allows to reduce the number of subsamples required to obtain a desired breakdown point with high probability.

Consider a sample $(y_1, \mathbf{x_1}), ..., (y_n, \mathbf{x_n})$ from a regression model

$$y_i = \beta' \mathbf{x}_i + u_i, 1 \leq i \leq n,$$

and let the S–estimate be defined by

$$\beta = \arg\min s(u_1(\beta), ..., u_n(\beta)),$$

where $u_i(\beta) = y_i - \beta' \mathbf{x}_i$ and the M–scale $s(u_1, ..., u_n)$ is defined by

$$\frac{1}{n} \sum_{i=1}^{n} \rho\left(\frac{u_i}{s}\right) = b. \tag{1}$$

The improvement step applied to a candidate β obtained by subsampling is as follows:

1. Compute the residuals $\widehat{\mathbf{u}}(\beta) = (\widehat{u}_1(\beta), ..., \widehat{u}_n(\beta))$.
2. Compute an approximate scale $\widehat{s}$ of $\widehat{\mathbf{u}}(\beta)$ by applying to eequation (1) one step of the Newton–Raphson algorithm starting from the MAD.
3. Compute the weights
$$w_i = \frac{\psi(\widehat{u}_i(\beta)/s)}{\widehat{u}_i(\beta)/s}, \tag{2}$$
where $\psi = \rho'$.

4. The improved candidate β^* is obtained by weighted least squares with weights defined by (2).

The reason why S–estimators may be expected to behave more robustly than the LTS–estimate is that they have smaller asymptotic bias and smaller asymptotic variance in contamination neighborhoods.

Our Monte Carlo simulation gives empirical evidence of the better robustness behavior of the S–estimator than the LTS–estimator when these estimates are computed by means of the corresponding fast–algorithms. On the other hand, the speeds of the two fast–algorithms are similar.

References

P. J. Rousseeuw and K. Van Driessen (2002). Computing LTS regression for large data sets. *Estadistica,* 54, 163–190.

P. J. Rousseeuw. and V. J. Yohai (1984). Robust regression by means of S–estimators. In *Robust and Nonlinear Time Series.* J. Franke, W. Hardle and D. Martin, eds. Lecture Notes in Statistics, 26, 256–272. Berlin: Springer–Verlag.