

## A Tour Around PROC ROBUSTREG

Michael O'Kelly, Quintiles Ireland Ltd, Dublin, Ireland

### ABSTRACT

ROBUSTREG methods are described and assessed: M-estimation, Least median of squares (LMS) estimation, Least trimmed squares (LTS) estimation, S-estimation and MM estimation, which combines starting values for the regression coefficients from LTS or S-estimation, an estimate of the scale from S-estimation, and performs the final estimation of the parameters via M-estimation. The paper presents estimates of regression parameters by SAS and by equivalent offerings in the R language and compares them. Options available for each of the methods are described, with an indication of the consequent changes in the statistical results. Some more general evaluation of robust regression is attempted

### INTRODUCTION

It is over 4 decades since Peter Huber (1964, see also Huber (1973) and Huber (1981)) proposed the M-estimator, but only recently has robust regression software become available in popular statistical packages. Robust regression software offers a large number of choices, both of general approach and of parameterization within each approach. This makes it challenging to compare and indeed validate the offerings of the widely-used statistical packages for robust regression. This paper surveys offerings in the new version of SAS®, version 9, and in the R language, an open-source language similar to S-plus. It should be noted that there are other options for robust regression software not described here. The rank-based method of Hettmansperger and McKean is available as an undocumented option in Minitab. Koch et al. (1998) describe a nonparametric ANCOVA using Cochran-Mantel-Haenszel test whose results can give an adjusted estimate of a treatment effect by inverting the test (see O'Kelly, 2003). This method is robust to outliers in the response, rather than to outliers in the explanatory variables. Some additional software options are surveyed in Yaffee (2006).

It is generally agreed that the following are desirable properties for robust regression methods:

#### *Efficiency*

the rate of convergence of the estimating algorithm to the true value of the parameters

#### *High breakdown value*

where breakdown value measures the proportion of contaminated observations in a data set required to change the estimate of the parameters to any value from  $-\infty$  to  $+\infty$ . It can be argued that *high breakdown* is not very useful as an absolute criterion, or that it can lead to an unduly absolute exclusion of outliers. Nevertheless it is widely used. It covers the idea that a group of related outliers may go undetected in OLS and it is a measure of effectiveness in detecting such a group.

Methods described and assessed with regard to the above two criteria include

#### **Least quantile of squares (LQS), often least median of squares (LMS) estimation**

*relatively inefficient but with relatively high breakdown. Not available in PROC ROBUSTREG.*

#### **Least trimmed squares (LTS) estimation**

*relatively inefficient (although more efficient than LMS) but with relatively high breakdown*

**S-estimation:** find regression estimates **b** yielding residuals whose variance **S** is minimised  
*somewhat more efficient variant on LTS and LMS estimation*

**M-estimation:** minimise a weighted objective function, which could be simply the usual least-squares function  
*relatively efficient*

**MM estimation**

which combines starting values for the regression coefficients from LTS or S-estimation, an estimate of the scale from S-estimation, and performs the final estimation of the parameters via M-estimation. This last is said to combine the high breakdown properties of S-estimation, LTS and LMS with the efficiency of M-estimation.

This paper will outline each of the methods. Options available for each of the five methods are described, with an outline of the consequent changes in the statistical results.

Lu (2005) presented regression results using M-estimation from a self-written macro and compared these with results from SAS PROC ROBUSTREG, noting similarities and some differences. In a similar manner the present paper presents estimates of regression parameters by each statistical package from a selection of data sets and compares them.

**OUTLINE OF THE METHODS AVAILABLE IN PROC ROBUSTREG**

In the quality of its documentation, SAS compares well with competing offerings in the R language (although of course because R is an open-source language, the source code is available for scrutiny). Venables and Ripley (2000) give a concise introduction to the robust methods of their package MASS for the R language, but much of the detail of the algorithms is left for the reader to figure out. The other R packages investigated – rrcov, roblm and robustbase – keep to the R tradition of minimal documentation, again assuming the reader will know the literature and/or read the source code and thus understand the items that can be specified when running the package. SAS, however, provides fairly good documentation, although it still refers, for example, to Rousseeuw et al. (2000), Huber (1981) and Hampel et al. (1986) for some important items. Here I will attempt to give as concrete an idea as possible of how the methods work, while leaving most of the mathematics to the SAS manual.

**LEAST QUANTILE OF SQUARES (LQS), OFTEN LEAST MEDIAN OF SQUARES (LMS) AND LEAST TRIMMED SQUARES (LTS) ESTIMATION**

Consider a model

$$E[y] = b_0 + b_1x$$

and let  $r$  be the residual from OLS for this model. Let  $p$  = the number of coefficients in the model = 2.

Although it is not available in PROC ROBUSTREG, LMS is perhaps the easiest of the robust regression methods to describe and serves as a gentle introduction. LMS finds  $\mathbf{b}$  that minimizes the median  $\{r^2\}$ . (Other quantiles can be used). The LMS is difficult to find directly. Software packages tend to find it via approximating algorithms. A simple procedure often used is to draw  $t$  sample sets of size  $p$  from the  $n$  observations, find  $\mathbf{b}$  using OLS for each sample and then calculate median  $\{r^2\}$  for the whole data set using each  $\mathbf{b}$ . The final  $\mathbf{b}$  is that associated with the smallest median  $\{r^2\}$ .

The following simple example of an LMS algorithm at work is due to Olive (2006). Let the data consist of the five  $(x_i, y_i)$  pairs (0,1), (1,2), (2,3), (3,4), and (1,11). Therefore  $n = 5$  and  $p = 2$ . Suppose  $t = 2$ . Suppose then that for the first sample pairs 1 and 5 were drawn. Then observations (0, 1) and (1, 11) were selected. For this sample  $\mathbf{b} = (1, 10)^T$ , the estimated line is  $y = 1 + 10x$ , and the corresponding residuals are 0, -9, -18, -27, and 0. The median squared residual =  $9^2 = 81$  since the ordered squared residuals are 0, 0, 81, 182, and 272. If observations (0, 1) and (3, 4) are selected next, then  $\mathbf{b} = (1, 1)^T$ , and 4 of the residuals are zero. Thus median squared residual for the second sample drawn = 0 and the final  $\mathbf{b} = (1, 1)^T$ . Hence the algorithm produces the fit  $y = 1 + x$ .

LTS estimates are calculated using a sampling algorithm similar to, but more complicated than, that for LMS, with the sum of the squares of the smallest, say,  $h$  of the  $n$  residuals being the criterion to minimize. The most widely-used algorithm to calculate LTS – the one used by SAS – is due to Rousseeuw and Van Driessen (2000). The algorithm, which is has a number of nested parts, is as follows:

1. Find an initial  $\mathbf{b}$  by OLS using a random subset of  $p$  observations.
2. Calculate the residuals for the whole data set from a model with the  $\mathbf{b}$  just calculated.
3. Use the subset of  $h$  observations with the lowest  $\{r^2\}$  to estimate a new  $\mathbf{b}$  via OLS.
4. Repeat from step 2. Each repeat of steps 2+3 is called a concentration step, or a C-step. Rousseeuw and Van Driessen show that, with enough C-steps, the sum of the lowest  $h$  squared residuals will converge. SAS allows the user to specify the number of C-steps to take here, so if the user specifies a small number, the C-steps may not converge at this stage.

The algorithm then repeats the entire process (steps 1-4) a number of times – again, the user can specify how many times. Each of these last repetitions yields an estimate of **b**. The best of these estimates of **b** are then used in another set of C-steps, but this time until convergence. Again, the user can specify how many of the estimates of **b** to choose as the 'best'. The final estimate is the best **b** of these 'best', that is the **b** with the lowest sum of *h* squared residuals.

For both LMS and LTS estimates of **b** can vary considerably depending on the quantile selected (LQS) or the number *h* of residuals to be minimized (LTS). As noted, SAS offers the user three choices for LTS with regard to numbers of repetitions of steps, all of which may also affect the estimate. In particular, consistency of the estimates of **b** should increase with the number of repeats of the entire process (steps 1-4) above (Hawkins and Olive, 2003 and Olive, 2006, Ch 8).

### **M- AND S-ESTIMATION: USING 'MODERATED' OLS WEIGHTS**

Both M-estimation and S-estimation now briefly described use weight functions of the residuals that increase less rapidly than the square of the residuals used by OLS. All of the functions trim or rein in the effect of large residuals on the estimates of **b** and/or the scale parameter, and all have a tuning parameter that the user can alter. The tuning parameter subdues the effect of observations with large residuals, generally at the cost of lowering the efficiency of the procedure.

### **MAXIMUM-LIKELYHOOD-LIKE-ESTIMATION (M-ESTIMATION)**

This method was nicely summarized and discussed in the paper by Lu, presented in 2004 at the annual meeting of the Pharmaceutical Industry SAS Users Group. M-estimates of the **b** are found by iterated re-weighted least squares (IRLS). The weighting function, which dampens the influence of observations with large residuals, is chosen from 10 available in ROBUSTREG. If, as is usual, the variance is not known, an extra iterative step is performed prior to each of the IRLS steps, to estimate the scale parameter for the step. Again, this step to estimate the scale uses a function of the residuals that moderates the effect of large residuals, compared to OLS. And again, the user can specify a tuning parameter for the function.

The two tuning parameters (for the weights used in estimating scale and in estimating **b**) allow the users to choose his or her preferred balance of efficiency and high breakdown. Changing the values of the tuning parameters can alter the results considerably.

The user can also choose from three estimators of the variance of the M-estimates, originally proposed by Huber (1981).

### **S-ESTIMATION**

This method gets its name from the dispersion parameter *S* that appears in an equation that must be solved in this method. The *S* method equates the expected value of weighted residuals (standardized by *S*) to the expected value if the data were Normally distributed. As with the weight functions for the M-estimate, the weight function is not as strongly affected by large residuals as the OLS square function would be. The method finds the **b** that minimize *S*. The algorithm is iterative and requires sampling of the data set. The user can specify the proportion sampled, which can be influential on the final estimates. The user can also specify the tuning parameter for the chosen weight function – there are two weight functions to choose from. As with M-estimation, this tuning parameter determines both the efficiency and the breakdown value of the procedure - the higher the efficiency, the lower the breakdown value. Unfortunately there seems to be but scanty information in the literature about how these attributes vary together in S-estimation. The SAS manual only gives the efficiency and breakdown for a single value of the tuning parameter – the one corresponding to a 25% breakdown value (which gives 72.7% efficiency)

SAS offers the same three estimators of the variance of the S-estimates that are available for the M-estimates. The user can chose from these.

### **MM-ESTIMATION**

MM estimation uses initial estimate of the **b** from a high-breakdown procedure chosen by the user (LTS or S estimation). The breakdown value of the MM-estimate is determined by the breakdown value of this initial estimate of **b** and can thus be kept quite high, because the efficiency of the procedure is determined by the relatively high efficiency of the following steps. The user can select the proportion of residuals to be examined in the LTS (*p* above).

The following estimation steps are then performed:

1. A scale parameter is estimated using an equation similar to that of S-estimation, with the weighted residuals from the **b**. The tuning parameter of this weight function has the same value as that used for the initial S-estimation of the **b**.

2. The scale parameter thus estimated is then used in an M-estimation step that has a second weight function. The tuning parameter for this second weight function determines the efficiency of the estimate of the  $\mathbf{b}$ . As with S-estimation, SAS does not give the user much information about setting the efficiency of this step, which in fact determines the efficiency of the whole procedure in estimating  $\mathbf{b}$ . The manual only gives the value of the tuning parameter that gives 85% efficiency.
3. SAS offers the same three estimators of the variance of the MM-estimates that are available for the M-estimates, plus a fourth. The user can chose from these.

### COMMENTS

There are four robust regression methods available in PROC ROBUSTREG. The user must choose a method, but the array of choices does not end there: within each robust regression method, we have seen that the user may choose a number of important parameters that will affect the value and/or the variance of the regression estimates. The variety of possible results from a single method must give pause to a statistician working in phase IIb, III and IV studies, already trying to control for multiplicity from other sources. The fact that the robust procedures described here work by favouring certain observations over others does appear to make ROBUSTREG unsuitable for the ITT framework.

As against this, it seems that MM-estimation is the clear favourite amongst the four methods, so one might well feel able to choose it in advance in a clinical trial protocol. And if the wide variety of tuning and other parameters is felt to allow too much flexibility and room for data dredging, a counter-strategy would in theory be either to decide in advance to accept defaults, or carefully to study the documentation and decide in advance on how to fine-tune the methods to obtain the desired combination of high breakdown and efficiency. However, given that some of the attractiveness of robust methods lies in their ability to detect groups of outliers of unknown size, an inflexible approach such as is demanded by much of later phase clinical research may be thought to go against the very spirit of robust regression.

In our next section we will look at how estimates from the methods can vary.

**COMPARISON OF RESULTS FROM ROBUST REGRESSION PROCEDURES IN SAS AND R**

For the well known stackloss data, table 1 gives the results for the LTS, M and MM robust regression methods from SAS ROBUSTREG and from a number of software offerings in the R language.

**Table 1: Stack loss data**

Robust method	Source	Estimates of regression coefficients			
		Intercep <i>t</i>	Airflow	Water temp	Acid. conc.
		-			
	SAS: OLS	39.9197	0.7156	1.2953	-0.1521
	(SD)	11.896	0.1349	0.368	0.1563
		-			
Least trimmed squares	SAS robust	35.4078	0.8462	0.4453	-0.0924
	R: rrcov	35.4078	0.8462	0.4453	-0.0924
	R: MASS	35.2079	0.9129	0.2921	-0.1011
		-			
M-estimation	SAS robust	42.2853	0.9275	0.6507	-0.1123
	(SD)	8.2357	0.1177	0.318	0.1085
	R: MASS	42.2853	0.9275	0.6507	-0.1123
	(SD)	9.5316	0.1081	0.2949	0.1252
		-			
MM-estimation	SAS robust	-41.526	0.9388	0.5797	-0.1129
	(SD)	7.8634	0.1126	0.2985	0.1039
	R: MASS	-41.523	0.9388	0.5795	-0.1129
	(SD)	9.307	0.1055	0.2879	0.1223
	R: roblm	41.5246	0.9389	0.5796	-0.1129
	(SD)	5.2978	0.1174	0.263	0.07
R:	-				
	robustbase	37.2631	0.8129	0.5344	-0.0712
	(SD)	3.0594	0.0644	0.1734	0.047

>33% relative difference across packages for the same robust method

In LTS, The proportion of residuals chosen to be minimised had a considerable effect on estimates of location. The author could not find a way to produce standard deviations of the location estimates directly from any of the packages for LTS estimation. It is worth noting that, except for the MASS package, LTS software tends to emphasise what is referred to as the “weighted least square” result. In fact, the “weighting” used in SAS’s main output is a zero/one weighting – what is presented is simply an OLS estimate omitting outliers and/or high-leverage points uncovered by the LTS procedure. The SAS output would benefit from a clearer heading.

For table 1, the weighting function used in all the algorithms was Tukey’s bisquare. A tuning constant of  $c=4.68501$  was used in the R packages and it was seen from roblm documentation that this corresponded to a choice of efficiency  $EFF=0.95$  in SAS ROBUSTREG. In the MM-estimation, for the initial S-estimation a tuning constant of 1.54764 was chosen. This was the default in the R packages and was explicitly specified in SAS ROBUSTREG. Huber’s so-called “proposal 2” was chosen to estimate the variance of the estimates where a choice was allowed. Thus the results above were generated with as many as possible of the choices and tuning constants chosen to be consistent across the packages.

## PhUSE 2006

Estimates of the regression coefficients in SAS differ somewhat across the robust methods. If we look at the ratio of the estimates across the regressions for a single covariate, we find that the largest ratio among the SAS estimates is 1.46 for Water Temperature, whose coefficients had estimates of 0.4453 and 0.6507 from LTS and M-estimation, respectively. Estimates of the standard deviation (SD) of those coefficients seem more consistent, with the largest ratio being 1.065, again for Water Temperature (0.318 and 0.2985 from M-estimation and MM-estimation, respectively).

In addition to the differences resulting from different methods of robust estimation, there are differences in the results of different software, when every attempt is made to use consistent tuning parameters across the software packages. The R package `rrcov` and SAS PROC ROBUSTREG give identical estimates of the location parameters using LTS regression. The R package MASS results for the LTS method differ, especially for the Water Temperature variable.

SAS and the MASS package had identical results for their M-estimates of location. The estimates of scale were close but not identical.

Finally, and perhaps surprisingly, given the large number of steps and options, there was good agreement for estimates of location between SAS, the MASS package and R's `robim` package, but they differed somewhat with regard to the estimate of scale. Results from the R package `robustbase`, however, did not agree with those from the other R packages or from SAS.

**Table 2: GDP data**

*Estimates of regression coefficients*

<i>Robust method</i>	<i>Source</i>	<i>Intercep t</i>	<i>Lab force growth</i>	<i>Relative GDP gap</i>	<i>Equipmnt investmn t</i>	<i>Non- equipmnt investmn t</i>
	SAS: OLS	-0.0143	-0.0298	0.0203	0.2654	0.0264
	(SD)	0.0103	0.1984	0.0092	0.0653	0.0348
<b>Least trimmed squares</b>	SAS robust	-0.0333	0.2721	0.0319	0.3697	0.0806
	R: <code>rrcov</code>	-0.0333	0.2721	0.0319	0.3697	0.0806
	R: MASS	-0.0275	0.1156	0.0221	0.297	0.1178
<b>M-estimation</b>	SAS robust	-0.0247	0.104	0.025	0.2968	0.0885
	(SD)	0.0094	0.18	0.0079	0.0667	0.0336
	R: MASS	-0.0247	0.104	0.025	0.2968	0.0885
	(SD)	0.0097	0.1868	0.0086	0.0615	0.0328
<b>MM-estimation</b>	SAS robust	-0.0239	0.0882	0.0247	0.2921	0.0874
	(SD)	0.0095	0.1808	0.008	0.0649	0.0333
	R: MASS	-0.0238	0.0866	0.0247	0.2916	0.0872
	(SD)	0.0096	0.1862	0.0086	0.0613	0.0327
	R: <code>robim</code>	-0.0239	0.0882	0.0247	0.2921	0.0874
	(SD)	0.0092	0.1724	0.0065	0.0828	0.0352
	R: <code>robustbase</code>	-0.0283	0.171	0.0266	0.325	0.0887
	(SD)	0.0098	0.1829	0.0081	0.1064	0.0461

>33% relative difference in package results

Differences between the packages did also occur for the larger data set of DeLong and Summers (1991) analysed by Zaman et al., (Table 2, 61 observations, vs. 21 in the stack loss data set). However, the differences were on a similar scale to those that occurred with the smaller data set.

## CONCLUSION

1. The smaller the proportion  $h/n$  of the observations used to calculate the criterion to be minimized, the more resistant to groups of outliers; but the less consistent the estimator – each subset tends to estimate quite different slopes, especially where there are many explanatory variables. This problem is alleviated by the use of C-steps (Hawkins and Olive, 2003).
2. In practice robust regression is most often used to identify outliers. The robust regression ignores or downplays outliers and/or groups of outliers in estimating a robust  $\mathbf{b}$ . The robust  $\mathbf{b}$ , designed to be undistorted by outliers, can then be used to identify true outliers. Thus the emphasis is on identifying outliers, as opposed to finding a model that takes outliers into account in some way. The practice of using robust regression principally to identify outliers is reinforced by the way SAS presents its output from ROBUSTREG, where the OLS minus outliers identified by the robust regression (entitled a “final weighted least squares” analysis) is presented by default, rather than the robust regression results themselves. The use of selected non-outlying observations from the data set means that these methods are more likely to be of use in early phase research or data-exploratory work rather than in the ITT setting.
3. Given 1 and 2, formal inference seems difficult.
4. The theory of many robust estimators relies on asymptotics, which may not apply.
5. The variety of choice in the methods, their complexity and the obscurity of the documentation pose practical difficulties for the researcher wishing to use robust regression.
6. The method of Hettmansperger and McKean (1998), uses a norm based on the Wilcoxon test statistics in place of OLS’s Euclidean norm. This method is not as yet widely available for the practicing statistician, but may offer a robust alternative more suitable for formal inference than the methods available in SAS’s PROC ROBUSTREG.

## REFERENCES

- Brownlee, KA (1960, 2nd ed. 1965) *Statistical Theory and Methodology in Science and Engineering*. New York: Wiley.
- De Long, JB and LH Summers (1991), “Equipment investment and economic growth” *Quarterly Journal of Economics*, 106, 445-501.
- Hampel, F.R., Ronchetti, E.M., Rousseeuw, P.J. and Stahel, W.A., 1986, *Robust Statistics, The Approach Based on Influence Functions*, New York: John Wiley and Sons, Inc.
- Hettmansperger, TP and JW McKean, 1998, *Robust nonparametric statistical methods*. London: Arnold.
- Hawkins, D.M., and Olive, D.J. (2002), Inconsistency of Resampling Algorithms for High Breakdown Regression Estimators and a New Algorithm, (with discussion), *Journal of the American Statistical Association*, 97, 136-159.
- Huber, Peter. *Robust Estimation of location parameter*. *Annals of Mathematical Statistics*, 35 (1), 1964.
- Huber, Peter, (1973) Robust Regression: Asymptotics, Conjectures and Monte Carlo, *Ann. Stat.*, 1, 799-821
- Huber, Peter, 1981, *Robust statistics*, Hoboken: John Wiley and Sons, Inc.
- Koch, Gary G., Catherine M. Tangen, Jin-Whan Jung and Ingrid A. Amara, 1998, Issues for covariance analysis of dichotomous and ordered categorical data from randomized clinical trials and non-parametric strategies for addressing them, *Statistics in Medicine*, 17, pp. 1863-1892.
- Lu, Shuang, 2004, An Experiment with Experimental PROC ROBUSTREG, *Proceedings of the 2004 meeting of the Pharmaceutical Industry SAS Users Group*, see also <http://www.lexjansen.com/pharmasug/2004/technicaltechniques/tt16.pdf>
- O’Kelly, Michael, 2003, Calculating estimates of an effect in stratified nonparametric analysis, in *Proceedings, 18th International Workshop on Statistical Modelling*, (eds. Geert Verbeke, Geert Molenburghs, Marc Aerts and Steffen Fieuws), pp. 349-354, Leuven: Katholieke Universiteit Leuven.
- Olive, David J., 2006, *Applied robust statistics*, <http://www.math.siu.edu/olive/ol-bookp.htm>

Rousseeuw, PJ and Van Driessen, K (2000), An Algorithm for Positive-Breakdown Regression Based on Concentration Steps, *Data Analysis: Scientific Modelling and Practical Applications*, ed. W. Gaul, O. Opitz and M Schader, New York: Springer-Verlag, 335-346

Venables, WN and BD Ripley, 1999, *Modern applied statistics with S-PLUS*, New York: Springer Verlag

Yaffee, RA, *Robust Regression Analysis: Some Popular Statistical Package Options*  
<http://www.nyu.edu/its/socsci/Docs/RobustReg2.pdf>

Zaman Azad, Peter J. Rousseeuw and Mehmet Orhan, 2001, Econometric Applications of High-Breakdown Robust Regression Techniques, *Economics Letters*, Vol. 71, No. 1, pp. 1-8

### **ACKNOWLEDGMENTS**

I would like to thank Quintiles Ireland for sponsoring this research

### **CONTACT INFORMATION**

Your comments and questions are valued and encouraged. Contact the author at:

Michael O'Kelly  
Quintiles Ireland Ltd.  
East Point Business Park  
Fairview  
Dublin 3.  
Work Phone: +353 1 8195432

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.