

Robust Regression

Appendix to *An R and S-PLUS Companion to Applied Regression*

John Fox

January 2002

1 *M*-Estimation

Linear least-squares estimates can behave badly when the error distribution is not normal, particularly when the errors are heavy-tailed. One remedy is to remove influential observations from the least-squares fit (see Chapter 6, Section 6.1, in the text). Another approach, termed *robust regression*, is to employ a fitting criterion that is not as vulnerable as least squares to unusual data.

The most common general method of robust regression is *M-estimation*, introduced by Huber (1964).¹ Consider the linear model

$$\begin{aligned}y_i &= \alpha + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \varepsilon_i \\ &= \mathbf{x}'_i \boldsymbol{\beta} + \varepsilon_i\end{aligned}$$

for the i th of n observations. The fitted model is

$$\begin{aligned}y_i &= a + b_1 x_{i1} + b_2 x_{i2} + \cdots + b_k x_{ik} + e_i \\ &= \mathbf{x}'_i \mathbf{b} + e_i\end{aligned}$$

The general *M*-estimator minimizes the *objective function*

$$\sum_{i=1}^n \rho(e_i) = \sum_{i=1}^n \rho(y_i - \mathbf{x}'_i \mathbf{b})$$

where the function ρ gives the contribution of each residual to the objective function. A reasonable ρ should have the following properties:

- $\rho(e) \geq 0$
- $\rho(0) = 0$
- $\rho(e) = \rho(-e)$
- $\rho(e_i) \geq \rho(e_{i'})$ for $|e_i| > |e_{i'}|$

For example, for least-squares estimation, $\rho(e_i) = e_i^2$.

Let $\psi = \rho'$ be the derivative of ρ . Differentiating the objective function with respect to the coefficients, \mathbf{b} , and setting the partial derivatives to 0, produces a system of $k + 1$ estimating equations for the coefficients:

$$\sum_{i=1}^n \psi(y_i - \mathbf{x}'_i \mathbf{b}) \mathbf{x}'_i = \mathbf{0}$$

¹This class of estimators can be regarded as a generalization of maximum-likelihood estimation, hence the term '*M*'-estimation. Huber's 1964 paper introduced *M*-estimation in the context of estimating the 'location' (center) of a distribution; the method was later generalized to regression.

Define the *weight function* $w(e) = \psi(e)/e$, and let $w_i = w(e_i)$. Then the estimating equations may be written as

$$\sum_{i=1}^n w_i (y_i - \mathbf{x}'_i \mathbf{b}) \mathbf{x}'_i = \mathbf{0}$$

Solving the estimating equations is a weighted least-squares problem, minimizing $\sum w_i^2 e_i^2$. The weights, however, depend upon the residuals, the residuals depend upon the estimated coefficients, and the estimated coefficients depend upon the weights. An iterative solution (called *iteratively reweighted least-squares, IRLS*) is therefore required:

1. Select initial estimates $\mathbf{b}^{(0)}$, such as the least-squares estimates.
2. At each iteration t , calculate residuals $e_i^{(t-1)}$ and associated weights $w_i^{(t-1)} = w[e_i^{(t-1)}]$ from the previous iteration.
3. Solve for new weighted-least-squares estimates

$$\mathbf{b}^{(t)} = \left[\mathbf{X}' \mathbf{W}^{(t-1)} \mathbf{X} \right]^{-1} \mathbf{X}' \mathbf{W}^{(t-1)} \mathbf{y}$$

where \mathbf{X} is the model matrix, with \mathbf{x}'_i as its i th row, and $\mathbf{W}^{(t-1)} = \text{diag}\{w_i^{(t-1)}\}$ is the current weight matrix.

Steps 2. and 3. are repeated until the estimated coefficients converge. The asymptotic covariance matrix of \mathbf{b} is

$$\mathcal{V}(\mathbf{b}) = \frac{E(\psi^2)}{[E(\psi')]^2} (\mathbf{X}' \mathbf{X})^{-1}$$

Using $\sum [\psi(e_i)]^2$ to estimate $E(\psi^2)$, and $[\sum \psi'(e_i)/n]^2$ to estimate $[E(\psi')]^2$ produces the *estimated* asymptotic covariance matrix, $\hat{\mathcal{V}}(\mathbf{b})$ (which is not reliable in small samples).

1.1 Objective Functions

Figure 1 compares the objective functions, and the corresponding ψ and weight functions for three M -estimators: the familiar least-squares estimator; the *Huber* estimator; and the Tukey *bisquare* (or *biweight*) estimator. The objective and weight functions for the three estimators are also given in Table 1.

Both the least-squares and Huber objective functions increase without bound as the residual e departs from 0, but the least-squares objective function increases more rapidly. In contrast, the bisquare objective function levels eventually levels off (for $|e| > k$). Least-squares assigns equal weight to each observation; the weights for the Huber estimator decline when $|e| > k$; and the weights for the bisquare decline as soon as e departs from 0, and are 0 for $|e| > k$.

The value k for the Huber and bisquare estimators is called a *tuning constant*; smaller values of k produce more resistance to outliers, but at the expense of lower efficiency when the errors are normally distributed. The tuning constant is generally picked to give reasonably high efficiency in the normal case; in particular, $k = 1.345\sigma$ for the Huber and $k = 4.685\sigma$ for the bisquare (where σ is the standard deviation of the errors) produce 95-percent efficiency when the errors are normal, and still offer protection against outliers.

In an application, we need an estimate of the standard deviation of the errors to use these results. Usually a robust measure of spread is employed in preference to the standard deviation of the residuals. For example, a common approach is to take $\hat{\sigma} = \text{MAR}/0.6745$, where MAR is the median absolute residual.

2 Bounded-Influence Regression

Under certain circumstances, M -estimators can be vulnerable to high-leverage observations. A key concept in assessing influence is the *breakdown point* of an estimator: The breakdown point is the fraction of ‘bad’

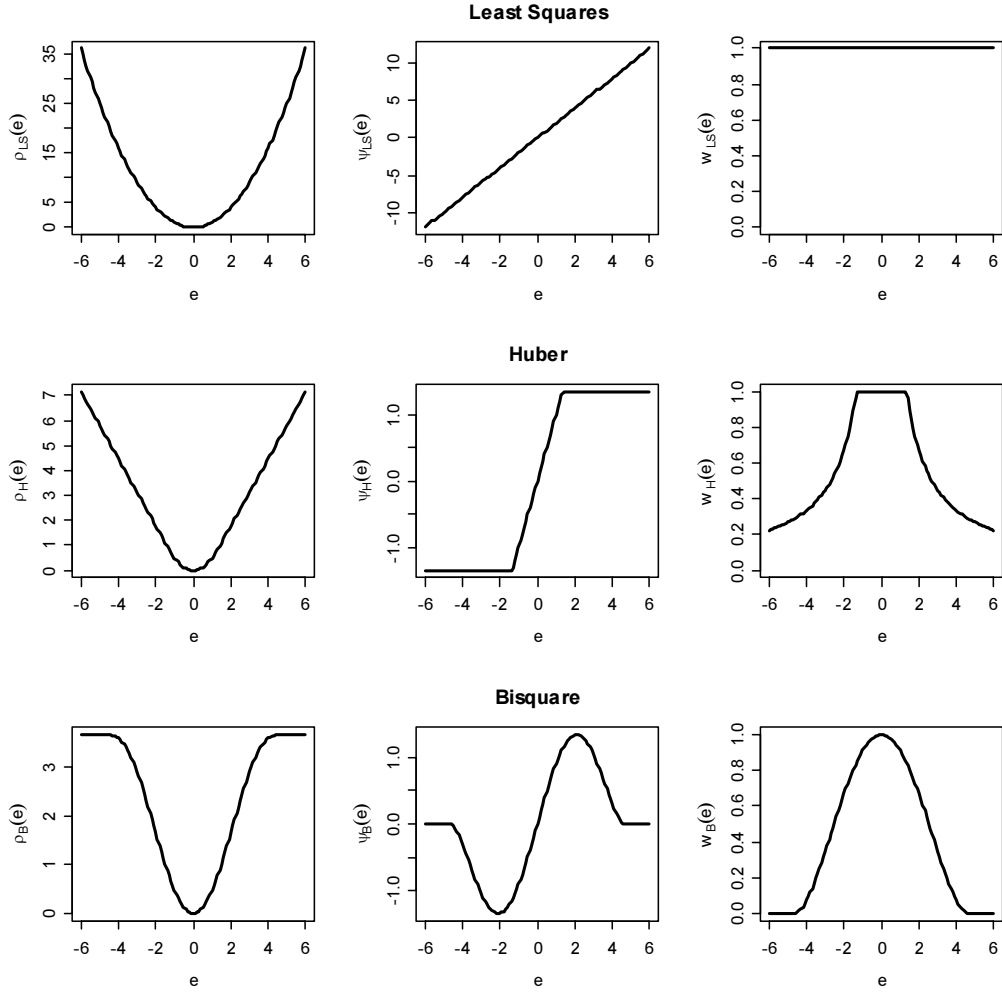


Figure 1: Objective, ψ , and weight functions for the least-squares (top), Huber (middle), and bisquare (bottom) estimators. The tuning constants for these graphs are $k = 1.345$ for the Huber estimator and $k = 4.685$ for the bisquare. (One way to think about this scaling is that the standard deviation of the errors, σ , is taken as 1.)

<i>Method</i>	<i>Objective Function</i>	<i>Weight Function</i>
Least-Squares	$\rho_{LS}(e) = e^2$	$w_{LS}(e) = 1$
Huber	$\rho_H(e) = \begin{cases} \frac{1}{2}e^2 & \text{for } e \leq k \\ k e - \frac{1}{2}k^2 & \text{for } e > k \end{cases}$	$w_H(e) = \begin{cases} 1 & \text{for } e \leq k \\ k/ e & \text{for } e > k \end{cases}$
Bisquare	$\rho_B(e) = \begin{cases} \frac{k^2}{6} \left\{ 1 - \left[1 - \left(\frac{e}{k} \right)^2 \right]^3 \right\} & \text{for } e \leq k \\ k^2/6 & \text{for } e > k \end{cases}$	$w_B(e) = \begin{cases} \left[1 - \left(\frac{e}{k} \right)^2 \right]^2 & \text{for } e \leq k \\ 0 & \text{for } e > k \end{cases}$

Table 1: Objective function and weight function for least-squares, Huber, and bisquare estimators.

data that the estimator can tolerate without being affected to an arbitrarily large extent. For example, in the context of estimating the center of a distribution, the mean has a breakdown point of 0, because even *one* bad observation can change the mean by an arbitrary amount; in contrast the median has a breakdown point of 50 percent.

There are also regression estimators that have breakdown points of nearly 50 percent. One such bounded-influence estimator is *least-trimmed squares (LTS)* regression.

The residuals from the fitted regression model are

$$\begin{aligned} e_i &= y_i - (a + b_1x_{i1} + b_2x_{i2} + \cdots + b_kx_{ik}) \\ &= y_i - \mathbf{x}'_i\mathbf{b} \end{aligned}$$

Let us order the squared residuals from smallest to largest:

$$(e^2)_{(1)}, (e^2)_{(2)}, \dots, (e^2)_{(n)}$$

The LTS estimator chooses the regression coefficients \mathbf{b} to minimize the sum of the smallest m of the squared residuals,

$$\text{LTS}(\mathbf{b}) = \sum_{i=1}^m (e^2)_{(i)}$$

where, typically, $m = \lfloor n/2 \rfloor + \lfloor (k+2)/2 \rfloor$ (i.e., a little more than half of the observations), and the ‘floor’ brackets, $\lfloor \cdot \rfloor$, denote rounding down to the next smallest integer.

While the LTS criterion is easily described, the mechanics of fitting the LTS estimator are complicated (see, for example, Rousseeuw and Leroy, 1987). Moreover, bounded-influence estimators can produce unreasonable results in certain circumstances (Stefanski, 1991), and there is no simple formula for coefficient standard errors.²

3 An Illustration: Duncan’s Occupational-Prestige Regression

Duncan’s occupational-prestige regression was introduced in Chapter 1 and described further in Chapter 6 on regression diagnostics. The least-squares regression of `prestige` on `income` and `education` produces the following results:

```
> library(car) # mostly for the Duncan data set
> data(Duncan)
> mod.ls <- lm(prestige ~ income + education, data=Duncan)
> summary(mod.ls)
```

Call:

```
lm(formula = prestige ~ income + education, data = Duncan)
```

Residuals:

Min	1Q	Median	3Q	Max
-29.538	-6.417	0.655	6.605	34.641

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	-6.0647	4.2719	-1.42	0.16
income	0.5987	0.1197	5.00	1.1e-05
education	0.5458	0.0983	5.56	1.7e-06

Residual standard error: 13.4 on 42 degrees of freedom

Multiple R-Squared: 0.828, Adjusted R-squared: 0.82

F-statistic: 101 on 2 and 42 DF, p-value: 1.11e-016

²Statistical inference for the LTS estimator can easily be performed by bootstrapping, however. See the Appendix on bootstrapping for an example.

Recall from the previous discussion of Duncan's data that two observations, ministers and railroad conductors, serve to decrease the `income` coefficient substantially and to increase the `education` coefficient, as we may verify by omitting these two observations from the regression:

```
> mod.ls.2 <- update(mod.ls, subset=-c(6,16))
> summary(mod.ls.2)

Call:
lm(formula = prestige ~ income + education, data = Duncan, subset = -c(6,
16))
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-28.61  -5.90   1.94   5.62  21.55
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -6.4090     3.6526  -1.75  0.0870
income         0.8674     0.1220   7.11 1.3e-08
education     0.3322     0.0987   3.36 0.0017
```

```
Residual standard error: 11.4 on 40 degrees of freedom
Multiple R-Squared:  0.876,    Adjusted R-squared:  0.87
F-statistic: 141 on 2 and 40 DF,  p-value:    0
```

Alternatively, let us compute the Huber M -estimator for Duncan's regression model, employing the `rlm` (robust linear model) function in the MASS library:

```
> library(MASS)
> mod.huber <- rlm(prestige ~ income + education, data=Duncan)
> summary(mod.huber)
```

```
Call: rlm.formula(formula = prestige ~ income + education, data = Duncan)
```

```
Residuals:
    Min       1Q   Median       3Q      Max
-30.12  -6.89   1.29   4.59  38.60
```

```
Coefficients:
              Value Std. Error t value
(Intercept) -7.111   3.881    -1.832
income       0.701   0.109     6.452
education    0.485   0.089     5.438
```

```
Residual standard error: 9.89 on 42 degrees of freedom
```

```
Correlation of Coefficients:
```

```
      (Intercept) income
income   -0.297
education -0.359   -0.725
```

The `summary` method for `rlm` objects prints the correlations among the coefficients; to suppress this output, specify `correlation=FALSE`. The Huber regression coefficients are between those produced by the least-squares fit to the full data set and by the least-squares fit eliminating the occupations `minister` and `conductor`.

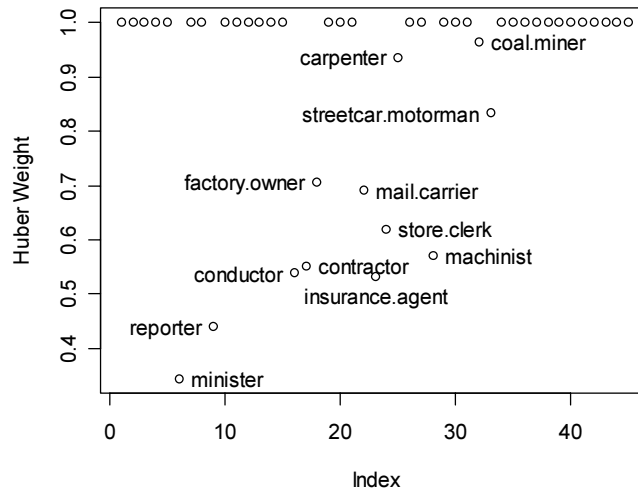


Figure 2: Weights from the robust Huber estimator for the regression of `prestige` on `income` and `education`. Observations with weights less than 1 were identified interactively with the mouse.

It is instructive to extract and plot (in Figure 2) the final weights employed in the robust fit, identifying observations with weights less than 1 using the mouse:

```
> plot(mod.huber$w, ylab="Huber Weight")
> identify(1:45, mod.huber$w, rownames(Duncan))
[1] 6 9 16 17 18 22 23 24 25 28 32 33
```

Ministers and conductors are among the observations that receive the smallest weight.

Next, I employ `rlm` to compute the bisquare estimator for Duncan's regression. Start-values for the IRLS procedure are potentially more critical for the bisquare estimator; specifying the argument `method='MM'` to `rlm` requests bisquare estimates with start values determined by a preliminary bounded-influence regression. To use this option, it is necessary first to attach the `lqs` library, which contains functions for bounded-influence regression:

```
> library(lqs)
> mod.bisq <- rlm(prestige ~ income + education, data=Duncan, method='MM')
> summary(mod.bisq, cor=F)
```

```
Call: rlm.formula(formula = prestige ~ income + education, data = Duncan,
  method = "MM")
```

Residuals:

Min	1Q	Median	3Q	Max
-29.87	-6.63	1.44	4.47	42.40

Coefficients:

	Value	Std. Error	t value
(Intercept)	-7.389	3.908	-1.891
income	0.783	0.109	7.149
education	0.423	0.090	4.710

Residual standard error: 9.79 on 42 degrees of freedom

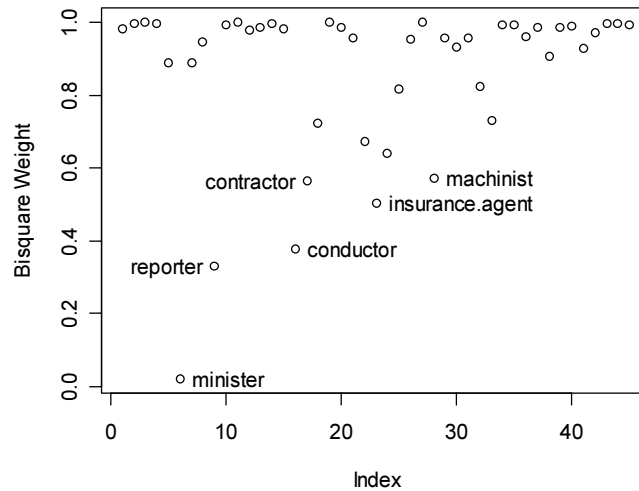


Figure 3: Weights from the robust bisquare estimator for the regression of `prestige` on `income` and `education`. Observations accorded relatively small weight were identified interactively with the mouse.

Compared to the Huber estimates, the bisquare estimate of the `income` coefficient is larger, and the estimate of the `education` coefficient is smaller. Figure 3 shows a graph of the weights from the bisquare fit, interactively identifying the observations with the smallest weights:

```
> plot(mod.bisq$w, ylab="Bisquare Weight")
> identify(1:45, mod.bisq$w, rownames(Duncan))
[1] 6 9 16 17 23 28
```

Finally, I use the `ltsreg` function in the `lqs` library to fit Duncan's model by LTS regression:³

```
> mod.lts <- ltsreg(prestige ~ income + education, data=Duncan)
> mod.lts
Call:
lqs.formula(formula = prestige ~ income + education, data = Duncan,
  method = "lts")
```

```
Coefficients:
(Intercept)      income      education
      -7.015         0.804         0.432
```

```
Scale estimates 7.77 7.56
```

In this case, the results are similar to those produced by the M -estimators. Note that the `print` method for bounded-influence regression gives the regression coefficients and two estimates of the variation ('scale') of the errors. There is no `summary` method for this class of models.

References

Huber, P. J. 1964. "Robust Estimation of a Location Parameter." *Annals of Mathematical Statistics* 35:73–101.

³LTS regression is also the default method for the `lqs` function, which additionally can fit other bounded-influence estimators.

Rousseeuw, R. J. & A. M. Leroy. 1987. *Robust Regression and Outlier Detection*. New York: Wiley.

Stefanski, L. A. 1991. "A Note on High-Breakdown Estimators." *Statistics and Probability Letters* 11:353–358.