Robust Regression in Stata

Ben Jann

University of Bern, jann@soz.unibe.ch

10th German Stata Users Group meeting
Berlin, June 1, 2012

Outline

- Introduction
- Estimators for Robust Regression
- Stata Implementation
- Example

Introduction

- Least-squares regression is a major workhorse in applied research.
- It is mathematically convenient and has great statistical properties.
- As is well known, the LS estimator is
  - BUE (best unbiased estimator) under normally distributed errors
  - BLUE (best linear unbiased estimator) under non-normal error distributions
- Furthermore, it is very robust in a technical sense (i.e. it is easily computable under almost any circumstance).

Introduction

- However, under non-normal errors better (i.e. more efficient) (non-linear) estimators exist.
  - For example, efficiency of the LS estimator can be poor if the error distribution has fat tails (such as, e.g., the t-distribution with few degrees of freedom).
- In addition, the properties of the LS estimator only hold under the assumption that the data comply to the suggested data generating process.
  - This may be violated, for example, if the data are "contaminated" by a secondary process (e.g. coding errors).

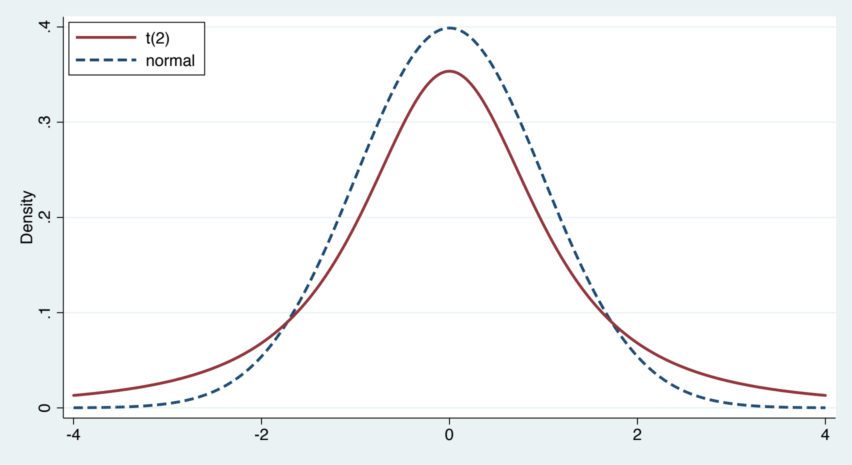
Why is Low Efficiency a Problem?

- An inefficient (yet unbiased) estimator gives the right answer on average over many samples.
- Most of the times, however, we only have one specific sample.
- An inefficient estimator has a large variation from sample to sample. This means that the estimator tends to be too sensitive to the particularities of the given sample.
- As a consequence, results from an inefficient estimator can be grossly misleading in a specific sample.

Why is Low Efficiency a Problem?

- Consider data from model

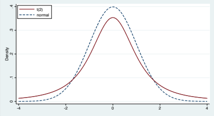
$$Y = \beta_1 + \beta_2 X + e \quad \text{with} \quad \beta_1 = \beta_2 = 0 \quad \text{and} \quad e \sim t(2)$$



Why is Low Efficiency a Problem?

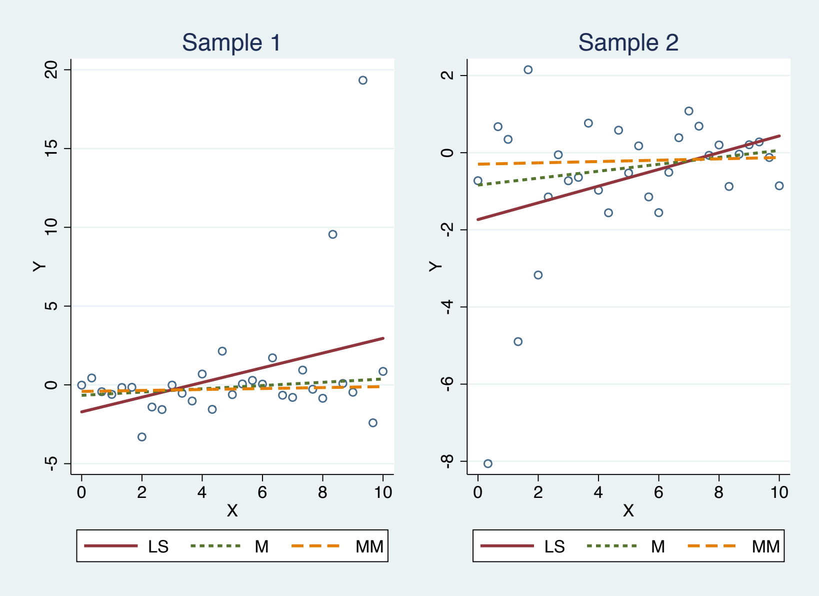
- Consider data from model

$$Y = \beta_1 + \beta_2 X + e \quad \text{with} \quad \beta_1 = \beta_2 = 0 \quad \text{and} \quad e \sim t(2)$$

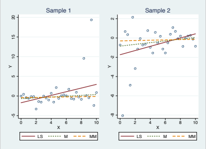



```
. two (function y = normalden(x), range(-4 4) lw(*2) lp(dash)) ///
>     (function y = tden(2,x)    , range(-4 4) lw(*2)) ///
>     , ytitle(Density) xtitle("") ysize(3) ///
>     legend(order(2 "t(2)" 1 "normal") col(1) ring(0) pos(11))
```

Why is Low Efficiency a Problem?



Why is Low Efficiency a Problem?

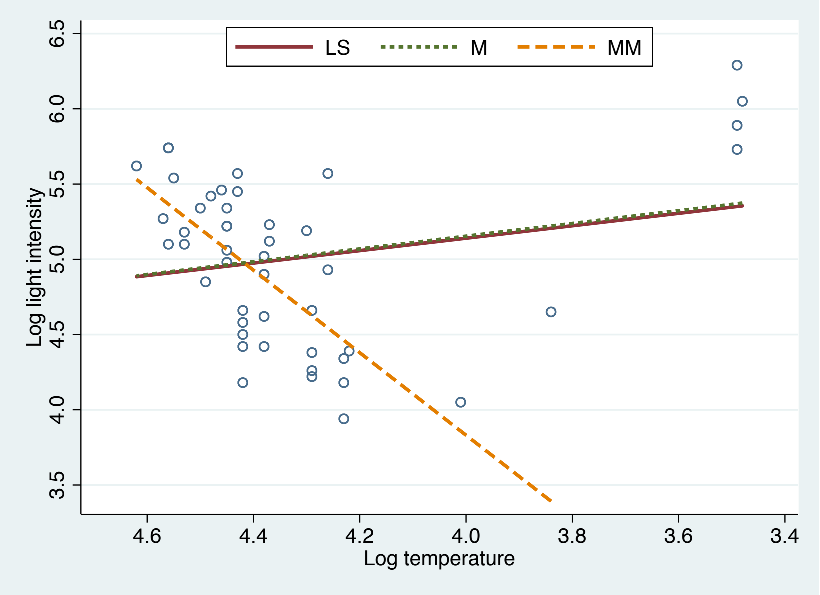



```
. drop _all

. set obs 31
obs was 0, now 31

. generate x = (_n-1)/3

. forvalues i = 1/2 {
  2.      local seed: word `i´ of 669 776
  3.      set seed `seed´
  4.      generate y = 0 + 0 * x + rt(2)
  5.      quietly robreg m y x
  6.      predict m
  7.      quietly robreg mm y x
  8.      predict mm
  9.      two (scatter y x, msymbol(Oh) mcolor(*.8)) ///
>          (lfit y x, lwidth(*2)) ///
>          (line m mm x, lwidth(*2 *2) lpattern(shortdash dash)) ///
>          , nodraw name(g`i´, replace) ytitle("Y") xtitle("X") ///
>          title(Sample `i´) scale(*1.1) ///
>          legend(order(2 "LS" 3 "M" 4 "MM") rows(1))
 10.      drop y m mm
 11. }

. graph combine g1 g2
```

Why is Contamination a Problem?

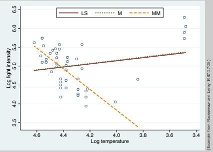
- Assume that the data are generated by two processes.
  - A main process we are interested in.
  - A secondary process that "contaminates" the data.
- The LS estimator will then give an answer that is an "average" of both processes.
- Such results can be meaningless because they represent neither the main process nor the secondary process (i.e. the LS results are biased estimates for both processes).
- It might be sensible to have an estimator that only picks up the main processes. The secondary process can then be identified as deviation from the first (by looking at the residuals).

Hertzsprung-Russell Diagram of Star Cluster CYG OB1



(Example from Rousseeuw and Leroy 1987:27-28)

Hertzsprung-Russell Diagram of Star Cluster CYG OB1




```
. use starsCYG, clear

. quietly robreg m log_light log_Te

. predict m

. quietly robreg mm log_light log_Te

. predict mm

. two (scatter log_light log_Te, msymbol(Oh) mcolor(*.8)) ///
>     (lfit log_light log_Te, lwidth(*2)) ///
>     (line m log_Te, sort lwidth(*2) lpattern(shortdash)) ///
>     (line mm log_Te if log_Te>3.5, sort lwidth(*2) lpattern(dash)) ///
>     , xscale(reverse) xlabel(3.4(0.2)4.7, format(%9.1f)) ///
>     xtitle("Log temperature") ylabel(3.5(0.5)6.5, format(%9.1f)) ///
>     ytitle("Log light intensity") ///
>     legen(order(2 "LS" 3 "M" 4 "MM") rows(1) ring(0) pos(12))
```

Efficiency of Robust Regression

- Efficiency under non-normal errors
  - A robust estimator should be efficient if the the errors do not follow a normal distribution.
- Relative efficiency
  - In general, robust estimators should be relatively efficient for a wide range of error distributions (including the normal distribution).
  - For a given error distribution, the "relative efficiency" of a robust estimator can be determined as follows:

$$RE = \frac{\text{variance of the maximum-likelihood estimator}}{\text{variance of the robust estimator}}$$

  - Interpretation: Fraction of sample with which the ML estimator is still as efficient as the robust estimator.
- Gaussian efficiency
  - Efficiency of a robust estimator under normal errors (compared to the LS estimator, which is equal to the ML estimator in this case).

Breakdown Point of Robust Regression

- Robust estimators should be resistant to a certain degree of data contamination.
- Consider a mixture distribution

$$F_\varepsilon = (1-\varepsilon)F_\theta + \varepsilon G$$

  where F_θ is the main distribution we are interested in and G is a secondary distribution that contaminates the data.
- The breakdown point ε^* of an estimator $\hat{\theta}(F_\varepsilon)$ is the largest value for ε, for which $\hat{\theta}(F_\varepsilon)$ as a function of G is bounded.
  - This is the maximum fraction of contamination that is allowed before $\hat{\theta}$ can take on any value depending on G.
- The LS estimator has a breakdown point of zero (as do many of the fist generation robust regression estimators).

First Generation Robust Regression Estimators

- A number of robust regression estimators have been developed as generalizations of robust estimators of location.
- In the regression context, however, these estimators have a low breakdown point if the design matrix X is not fixed.
- The best known first-generation estimator is the so called M-estimator by Huber (1973).
- An extension are so called GM- or bounded influence estimators that, however, do not really solve the low breakdown point problem.

First Generation Robust Regression Estimators

- The M-estimator is defined as

$$\hat{\beta}^{\mathrm{M}} = \arg\min_{\hat{\beta}} \sum_{i=1}^{n} \rho\left(\frac{Y_i - X_i^T\hat{\beta}}{\sigma}\right)$$

  where ρ is a suitable "objective function".

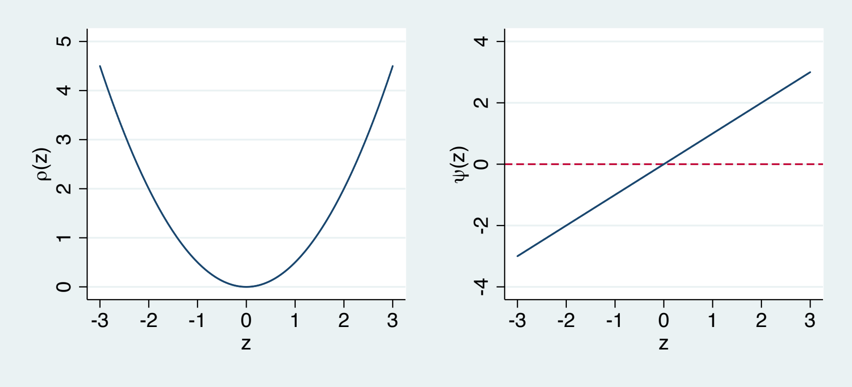
- Assuming σ to be known, the M-estimate is found by solving

$$\sum_{i=1}^{n} \psi\left(\frac{Y_i - X_i^T\hat{\beta}}{\sigma}\right) X_i = 0$$

  where ψ is the first derivative of ρ.

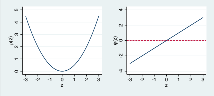
First Generation Robust Regression Estimators

- Different choices for ρ lead to different variants of M-estimators.
- For example, setting $\rho(z) = \frac{1}{2}z^2$ we get the LS estimator. This illustrates that LS is a special case of the M-estimator.
- ρ and ψ of the LS estimator look as follows:



First Generation Robust Regression Estimators

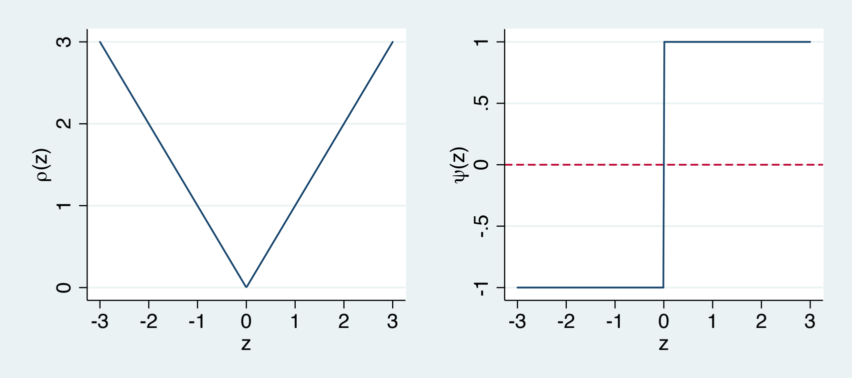
- Different choices for ρ lead to different variants of M-estimators.
- For example, setting $\rho(z) = \frac{1}{2}z^2$ we get the LS estimator. This illustrates that LS is a special case of the M-estimator.
- ρ and ψ of the LS estimator look as follows:




```
. two function y = .5*x^2, range(-3 3) xlabel(-3(1)3) ///
>     ytitle("{&rho}(z)") xtitle(z) nodraw name(rho, replace)
. two function y = x, range(-3 3) xlabel(-3(1)3) yline(0, lp(dash)) ///
>     ytitle("{&psi}(z)") xtitle(z) nodraw name(psi, replace)
. graph combine rho psi, ysize(2.5) scale(*2)
```

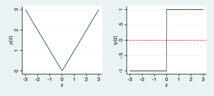
First Generation Robust Regression Estimators

- To get an M-estimator that is more robust to outliers than LS we have to define ρ so that it grows slower than the ρ of LS.
- In particular, it seems reasonable to chose ρ such that ψ is bounded (ψ is roughly equivalent to the influence of a data point).
- A possible choice is to set $\rho(z) = |z|$. This leads to the median regression (a.k.a. L_1-estimator, LAV, LAD).



First Generation Robust Regression Estimators

- To get an M-estimator that is more robust to outliers than LS we have to define ρ so that it grows slower than the ρ of LS.
- In particular, it seems reasonable to chose ρ such that ψ is bounded (ψ is roughly equivalent to the influence of a data point).
- A possible choice is to set $\rho(z) = |z|$. This leads to the median regression (a.k.a. L_1-estimator, LAV, LAD).




```
. two function y = abs(x), range(-3 3) xlabel(-3(1)3) ///
>     ytitle("{&rho}(z)") xtitle(z) nodraw name(rho, replace)
. two function y = sign(x), range(-3 3) xlabel(-3(1)3) yline(0, lp(dash)) ///
>     ytitle("{&psi}(z)") xtitle(z) nodraw name(psi, replace)
. graph combine rho psi, ysize(2.5) scale(*2)
```

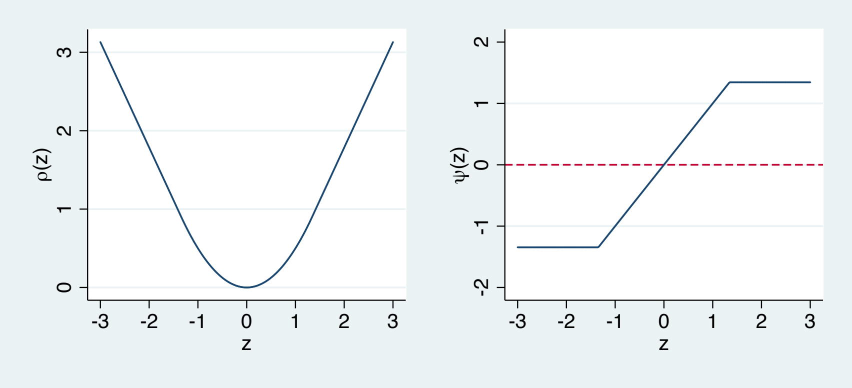
First Generation Robust Regression Estimators

- Unfortunately, the LAV-estimator has low gaussian efficiency (63.7%).
- This lead Huber (1964) to define an objective function that combines the good efficiency of LS and the robustness of LAV.
- Huber's ρ and ψ are given as:

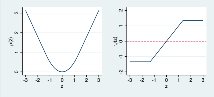
$$\rho^{\mathrm{H}}(z) = \begin{cases} \frac{1}{2}z^2 & \text{if } |z| \leq k \\ k|z| - \frac{1}{2}k^2 & \text{if } |z| > k \end{cases} \quad \text{and} \quad \psi^{\mathrm{H}}(z) = \begin{cases} k & \text{if } z > k \\ z & \text{if } |z| \leq k \\ -k & \text{if } z < -k \end{cases}$$

- Parameter k determines the gaussian efficiency of the estimator. For example, for $k = 1.35$ gaussian efficiency is 95%.
  - approaches LS if $k \to \infty$
  - approaches LAV if $k \to 0$

First Generation Robust Regression Estimators



First Generation Robust Regression Estimators




```
. local k 1.345

. two function y = cond(abs(x)<=`k´ , .5*x^2 , `k´*abs(x) - 0.5*`k´^2), ///
>     range(-3 3) xlabel(-3(1)3) ///
>     ytitle("{&rho}(z)") xtitle(z) nodraw name(rho, replace)

. two function y = cond(abs(x)<=`k´ , x, sign(x)*`k´), ///
>     range(-3 3) xlabel(-3(1)3) yline(0, lp(dash)) ///
>     ytitle("{&psi}(z)") xtitle(z) nodraw name(psi, replace)

. graph combine rho psi, ysize(2.5) scale(*2)
```

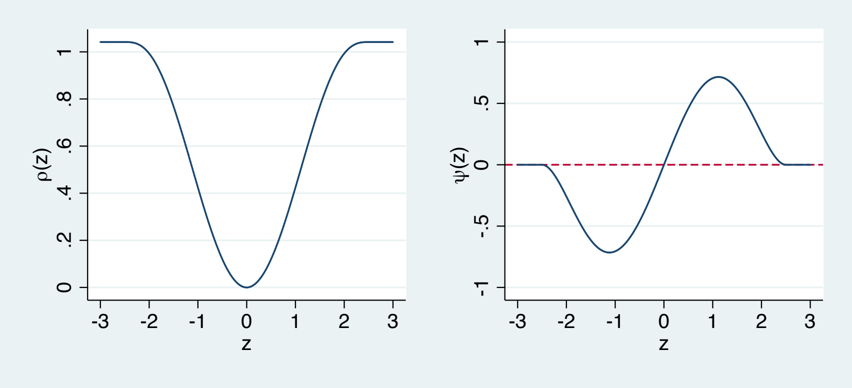
First Generation Robust Regression Estimators

- The Huber M-estimator belongs to the class of monotone M-estimators (the advantage of which is that there are no local minima in the optimization problem).
- Even better results in terms of efficiency and robustness can be achieved by so called "redescending" M-estimators that completely ignore large outliers.
- A popular example is the bisquare or biweight objective function suggested by Beaton and Tukey (1974):

$$\rho^{\mathrm{B}}(z) = \begin{cases} \frac{k^2}{6}\left(1-\left(1-(z/k)^2\right)^3\right) & \text{if } |z| \leq k \\ \frac{k^2}{6} & \text{if } |z| > k \end{cases}$$

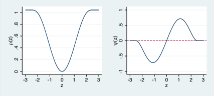
$$\psi^{\mathrm{B}}(z) = \begin{cases} z\left(1-(z/k)^2\right)^2 & \text{if } |z| \leq k \\ 0 & \text{if } |z| > k \end{cases}$$

First Generation Robust Regression Estimators



- Again, k determines gaussian efficiency (e.g. 95% for $k = 4.69$).
- Optimization has local minima. Therefore, the bisquare M is often used with starting values from Huber M (as in Stata's `rreg`).

First Generation Robust Regression Estimators



- Again, k determines gaussian efficiency (e.g. 95% for $k = 4.69$).
- Optimization has local minima. Therefore, the bisquare M is often used with starting values from Huber M (as in Stata's `rreg`).

```
. local k 2.5

. two fun y = cond(abs(x)<=`k´, `k´^2/6*(1-(1- (x/`k´)^2)^3), `k´^2/6), ///
>     range(-3 3) xlabel(-3(1)3) ///
>     ytitle("{&rho}(z)") xtitle(z) nodraw name(rho, replace)

. two function y = cond(abs(x)<=`k´, x*(1- (x/`k´)^2)^2, 0), ///
>     range(-3 3) xlabel(-3(1)3) yline(0, lp(dash)) ///
>     ytitle("{&psi}(z)") xtitle(z) nodraw name(psi, replace)

. graph combine rho psi, ysize(2.5) scale(*2)
```

First Generation Robust Regression Estimators

- Computation of M-estimators
  - M-estimators can be computed using an IRWLS algorithm (iteratively reweighted least squares).
  - The procedure iterates between computing weights from given parameters and computing parameters from given weights until convergence.
  - The error variance is computed from the residuals using some robust estimator of scale such as the (normalized) median absolute deviation.
- Breakdown point of M-estimators
  - M-estimators such as LAV, Huber, or bisquare are robust to Y-outliers (as long as a robust estimate for σ is used).
  - However, if X-outliers with high leverage are possible, then the breakdown point drops to zero and not much is gained compared to LS.

Second Generation Robust Regression Estimators

- A number of robust regression estimators have been proposed to tackle the problem of a low breakdown point in case of X outliers.
- Early examples are LMS (least median of squares) and LTS (least trimmed squares) (Rousseeuw and Leroy 1987).
- LMS minimizes the median of the squared residuals

$$\hat{\beta}^{\mathrm{LMS}} = \arg\min_{\hat{\beta}} \mathrm{MED}(r(\hat{\beta})_1^2, \ldots, r(\hat{\beta})_n^2)$$

  and has a breakdown point of approximately 50%.
    - It finds the "narrowest" band through the data that contains at least 50% of the data.

Second Generation Robust Regression Estimators

- The LTS estimator follows a similar idea, but also takes into account how the data are distributed within the 50% band.
- It minimizes the variance of the 50% smallest residuals:

$$\hat{\beta}^{\mathrm{LTS}} = \arg\min_{\hat{\beta}} \sum_{i=1}^{h} r(\hat{\beta})^2_{(i)} \quad \text{with} \quad h = \lfloor n/2 \rfloor + 1$$

  where $r(\hat{\beta})_{(i)}$ are the ordered residuals.
- LMS and LTS are attractive because of their high breakdown point and their nice interpretation.
- However, gaussian efficiency is terrible (0% and 7%, respectively).
- Furthermore, estimation is tedious (jumpy objective function; lots of local minima).

Second Generation Robust Regression Estimators

- A better alternative is the so called S-estimator.
- Similar to LS, the S-estimator minimizes the variance of the residuals. However, it uses a robust measure for the variance.
- It is defined as

$$\hat{\beta}^{S} = \arg\min_{\hat{\beta}} \hat{\sigma}(r(\hat{\beta}))$$

where $\hat{\sigma}(r)$ is an M-estimator of scale, found as the solution of

$$\frac{1}{n-p}\sum_{i=1}^{n} \rho\left(\frac{Y_i - x_i^T\hat{\beta}}{\hat{\sigma}}\right) = \delta$$

with δ as a suitable constant to ensure consistency.

Second Generation Robust Regression Estimators

- For ρ the bisquare function is commonly employed.
- Depending on the value of the tuning constant k of the bisquare function, the S-estimator can reach a breakdown point of 50% ($k = 1.55$) without sacrificing as much efficiency as LMS or LTS (gaussian efficiency is 28.7%).
- Similar to LMS/LTS, estimation of S is tedious because there are local minima. However the objective function is relatively smooth so that computational shortcuts can be used.

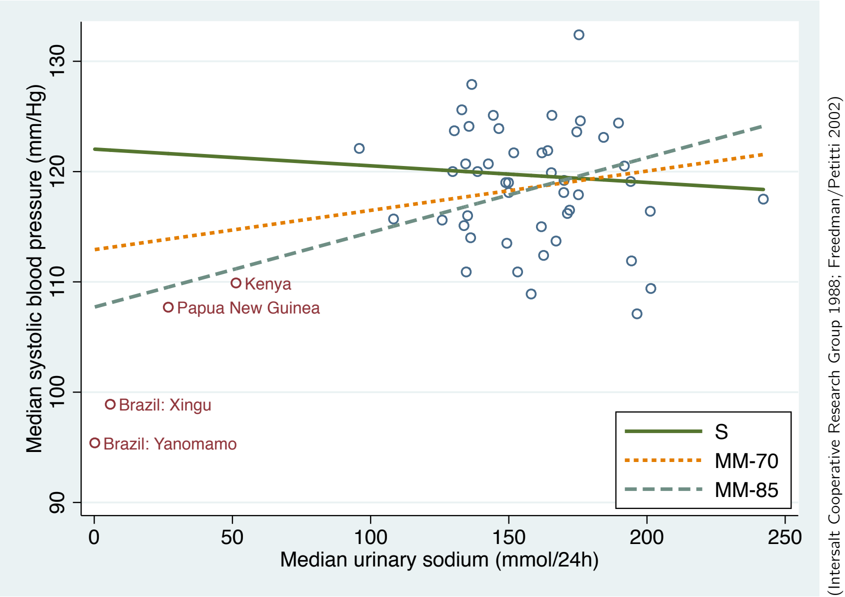
Second Generation Robust Regression Estimators

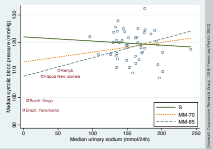
- The gaussian efficiency of the S-estimator is still unsatisfactory.
- The problem is that in case of gaussian errors too much information is thrown away.
- High efficiency while preserving a high breakdown point is possible by combining an S- and an M-estimator.
- This is the so called MM-estimator. It works as follows:
  1. Retrieve an initial estimate for β and an estimate for σ using the S-estimator with a 50% breakdown point.
  2. Apply a redescending M-estimator (bisquare) using $\hat{\beta}^S$ as starting values (while keeping $\hat{\sigma}$ fixed).

Second Generation Robust Regression Estimators

- The higher the efficiency of the M-estimator in the second step, the higher the maximum bias due to data contamination. An efficiency of 85% is suggested as a good compromise ($k = 3.44$).
- However, it can also be sensible to try different values to see how the estimates change depending on k.

Second Generation Robust Regression Estimators



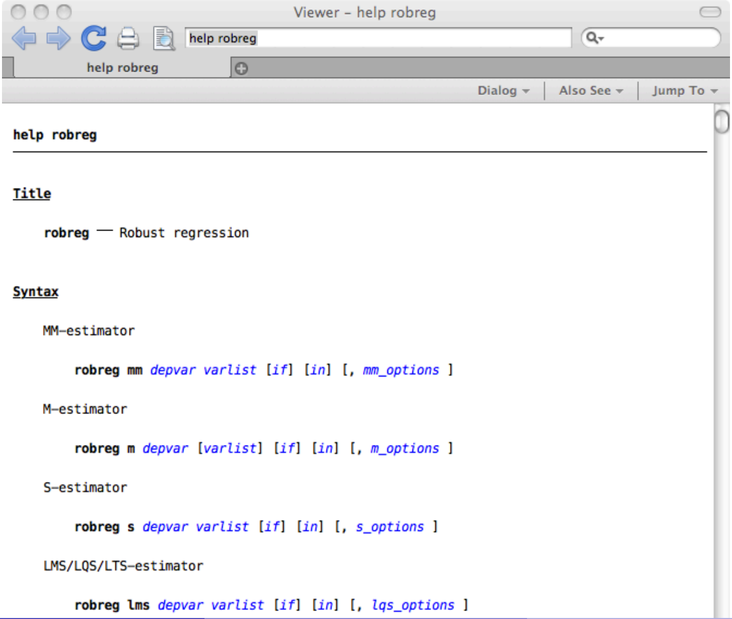



```
. use intersalt/intersalt, clear
. qui robreg s msbp mus
. predict s
. qui robreg mm msbp mus
. predict mm85
. qui robreg mm msbp mus, eff(70)
. predict mm70
. two (scatter msbp mus if mus>60, msymbol(Oh) mcolor(*.8)) ///
>     (scatter msbp mus if mus<60, msymbol(Oh) mlabel(centre)) ///
>     (line s mus, sort lwidth(*2)) ///
>     (line mm70 mus, sort lwidth(*2) lpattern(shortdash)) ///
>     (line mm85 mus, sort lwidth(*2) lpattern(dash)) ///
>     , ytitle("`: var lab msbp´") ///
>     legen(order(3 "S" 4 "MM-70" 5 "MM-85") cols(1) ring(0) pos(4))
```

Stata Implementation

- Official Stata has the `rreg` command.
  - It is essentially an M-estimator (Huber follwed by bisquare), but also includes an initial step that removes high-leverage outliers (based on Cook's D). Nonetheless, it has a low breakdown point.
- High breakdown estimators are provided by the `robreg` user command.
  - Supports MM, M, S, LMS, and LTS estimation.
  - Provides robust standard errors for MM, M, and S estimation.
  - Implements a fast algorithm for the S-estimator.
  - Provides options to set efficiency and breakdown point.
  - Available from SSC.

Stata Implementation



help robreg

Title

robreg — Robust regression

Syntax

MM-estimator

robreg mm *depvar varlist* [*if*] [*in*] [, *mm_options*]

M-estimator

robreg m *depvar* [*varlist*] [*if*] [*in*] [, *m_options*]

S-estimator

robreg s *depvar varlist* [*if*] [*in*] [, *s_options*]

LMS/LQS/LTS-estimator

robreg lms *depvar varlist* [*if*] [*in*] [, *lqs_options*]

Example: Online Actions of Mobile Phones

```
. robreg mm price rating startpr shipcost duration nbids minincr
Step 1: fitting S-estimate
enumerating 50 candidates (percent completed)
0 ──── 20 ──── 40 ──── 60 ──── 80 ──── 100
..................................................
refining 2 best candidates ... done
Step 2: fitting redescending M-estimate
iterating RWLS estimate ........................................... done
MM-Regression (85% efficiency)                    Number of obs    =        99
                                                  Subsamples       =        50
                                                  Breakdown point  =        .5
                                                  M-estimate: k    =  3.443686
                                                  S-estimate: k    =  1.547645
                                                  Scale estimate   = 32.408444
                                                  Robust R2 (w)    = .62236093
                                                  Robust R2 (rho)  = .22709915
```

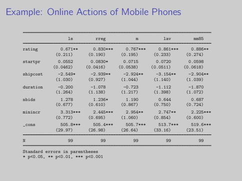
price	Coef.	Robust Std. Err.	z	P>|z|	[95% Conf.	Interval]
rating	.8862042	.274379	3.23	0.001	.3484312	1.423977
startpr	.0598183	.0618122	0.97	0.333	-.0613313	.1809679
shipcost	-2.903518	1.039303	-2.79	0.005	-4.940515	-.8665216
duration	-1.86956	1.071629	-1.74	0.081	-3.969914	.2307951
nbids	.6874916	.7237388	0.95	0.342	-.7310104	2.105994
minincr	2.225189	.5995025	3.71	0.000	1.050185	3.400192
_cons	519.5566	23.51388	22.10	0.000	473.4702	565.6429

(Data from Diekmann et al. 2009)

Example: Online Actions of Mobile Phones

	ls	rreg	m	lav	mm85
rating	0.671**	0.830***	0.767***	0.861***	0.886**
	(0.211)	(0.190)	(0.195)	(0.233)	(0.274)
startpr	0.0552	0.0830*	0.0715	0.0720	0.0598
	(0.0462)	(0.0416)	(0.0538)	(0.0511)	(0.0618)
shipcost	-2.549*	-2.939**	-2.924**	-3.154**	-2.904**
	(1.030)	(0.927)	(1.044)	(1.140)	(1.039)
duration	-0.200	-1.078	-0.723	-1.112	-1.870
	(1.264)	(1.138)	(1.217)	(1.398)	(1.072)
nbids	1.278	1.236*	1.190	0.644	0.687
	(0.677)	(0.610)	(0.867)	(0.750)	(0.724)
minincr	3.313***	2.445***	2.954**	2.747**	2.225***
	(0.772)	(0.695)	(1.060)	(0.854)	(0.600)
_cons	505.8***	505.4***	505.7***	513.7***	519.6***
	(29.97)	(26.98)	(26.64)	(33.16)	(23.51)
N	99	99	99	99	99

Standard errors in parentheses
* p<0.05, ** p<0.01, *** p<0.001



Example: Online Actions of Mobile Phones

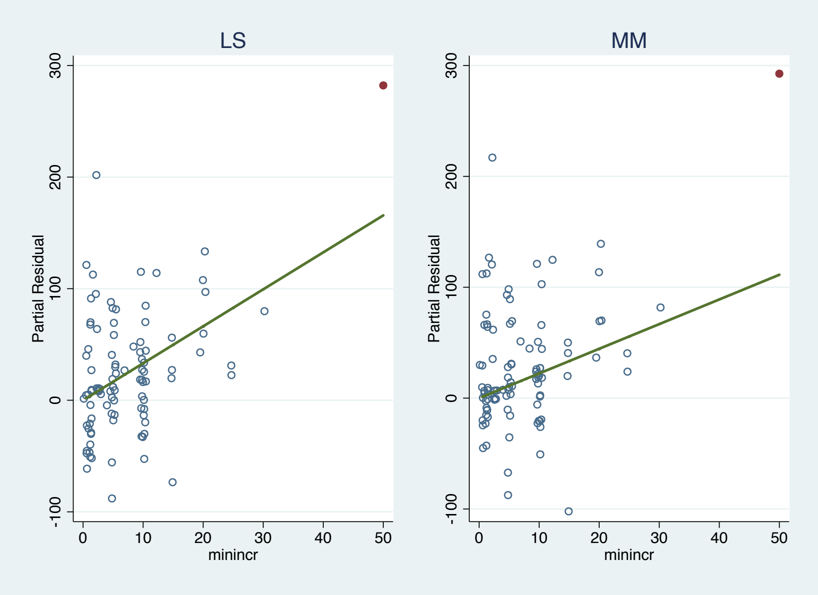
	ls	rreg	m	lav	mm85
rating	0.671**	0.830***	0.767***	0.861***	0.886**
	(0.211)	(0.190)	(0.195)	(0.233)	(0.274)
startpr	0.0552	0.0830*	0.0715	0.0720	0.0598
	(0.0462)	(0.0416)	(0.0538)	(0.0511)	(0.0618)
shipcost	-2.549*	-2.939**	-2.924**	-3.154**	-2.904**
	(1.030)	(0.927)	(1.044)	(1.140)	(1.039)
duration	-0.200	-1.078	-0.723	-1.112	-1.870
	(1.264)	(1.138)	(1.217)	(1.398)	(1.072)
nbids	1.278	1.236*	1.190	0.644	0.687
	(0.677)	(0.610)	(0.867)	(0.750)	(0.724)
minincr	3.313***	2.445***	2.954**	2.747**	2.225***
	(0.772)	(0.695)	(1.060)	(0.854)	(0.600)
_cons	505.8***	505.4***	505.7***	513.7***	519.6***
	(29.97)	(26.98)	(26.64)	(33.16)	(23.51)
N	99	99	99	99	99

Standard errors in parentheses
* p<0.05, ** p<0.01, *** p<0.001

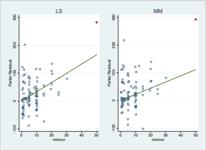
```
. quietly reg price rating startpr shipcost duration nbids minincr
. eststo ls
. quietly rreg price rating startpr shipcost duration nbids minincr
. eststo rreg
. quietly robreg m price rating startpr shipcost duration nbids minincr
. eststo m
. quietly qreg price rating startpr shipcost duration nbids minincr
. eststo lav
. quietly robreg mm price rating startpr shipcost duration nbids minincr
. eststo mm85
. esttab ls rreg m lav mm85, compress se mti nonum
```

	ls	rreg	m	lav	mm85
rating	0.671**	0.830***	0.767***	0.861***	0.886**
	(0.211)	(0.190)	(0.195)	(0.233)	(0.274)
startpr	0.0552	0.0830*	0.0715	0.0720	0.0598
	(0.0462)	(0.0416)	(0.0538)	(0.0511)	(0.0618)
shipcost	-2.549*	-2.939**	-2.924**	-3.154**	-2.904**
	(1.030)	(0.927)	(1.044)	(1.140)	(1.039)

Example: Online Actions of Mobile Phones



Example: Online Actions of Mobile Phones




```
. quietly reg price rating startpr shipcost duration nbids minincr

. predict ls_cpr
(option xb assumed; fitted values)
(6 missing values generated)

. replace ls_cpr = price - ls_cpr + _b[minincr]*minincr
(188 real changes made, 89 to missing)

. generate ls_fit = _b[minincr]*minincr

. quietly robreg mm price rating startpr shipcost duration nbids minincr

. predict mm_cpr
(6 missing values generated)

. replace mm_cpr = price - mm_cpr + _b[minincr]*minincr
(188 real changes made, 89 to missing)

. generate mm_fit = _b[minincr]*minincr

. two (scatter ls_cpr minincr if minincr<40, ms(Oh) mc(*.8) jitter(1)) ///
>     (scatter ls_cpr minincr if minincr>40) ///
>     (line ls_fit minincr, sort lwidth(*2)) ///
>     , title(LS) ytitle(Partial Residual) legend(off) ///
>     name(ls, replace) nodraw

. two (scatter mm_cpr minincr if minincr<40, ms(Oh) mc(*.8) jitter(1)) ///
>     (scatter mm_cpr minincr if minincr>40) ///
>     (line mm_fit minincr, sort lwidth(*2)) ///
>     , title(MM) ytitle(Partial Residual) legend(off) ///
>     name(mm, replace) nodraw

. graph combine ls mm
```

Conclusions

- High breakdown-point robust regression is now available in Stata.
- Should we use it?
  - Some people recommend using robust regression *instead* of classic methods.
  - However, I see it more as a diagnostic tool, yet less tedious then classic regression diagnostics.
  - A good advice is to use classic methods for most of the work, but then check the models using robust regression.
  - If there are differences, then go into details.
- Outlook
  - Robust GLM
  - Robust fixed effects and instrumental variables regression
  - Robust multivariate methods
  - ...

References

- Beaton, A.E., J.W. Tukey. 1974. The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data. Technometrics 16(2): 147-185.
- Diekmann, A., B. Jann, D. Wyder. 2009. Trust and Reputation in Internet Auctions. Pp. 139-165 in: K.S. Cook, C. Snijders, V. Buskens, C. Cheshire (ed.). eTrust. Forming Relationships in the Online World. New York: Russell Sage Foundation.
- Freedman, D.A., D.B. Petitti. 2002. Salt, blood pressure, and public policy. International Journal of Epidemiology 31: 319-320.
- Huber, P.J. 1964. Robust estimation of a location parameter. The Annals of Mathematical Statistics, 35(1): 73-101.
- Huber, P.J. 1973. Robust regression: Asymptotics, conjectures and monte carlo. The Annals of Statistics 1(5): 799-821.
- Intersalt Cooperative Research Group. 1988. Intersalt: an international study of electrolyte excretion and blood pressure: results for 24 hour urinary sodium and potassium excretion. British Medical Journal 297 (6644): 319-328.
- Maronna, R.A., D.R. Martin, V.J. Yohai. 2006. Robust Statistics. Theory and Methods. Chichester: Wiley.
- Rousseeuw, P.J., A.M. Leroy. 1987. Robust Regression and Outlier Detection. New York: Wiley.