Robustness for dummies

Vincenzo Verardi
joint with M. Gassner and D. Ugarte

2012 UK Stata Users Group meeting
Cass Business School, London

September 2012

Types of outliers

Robust estimators

Consider regression model

$$Y_i = X_i^t\theta + \varepsilon_i$$

where Y_i is the dependent variable, X_i is the vector of covariates and ε_i is the error term $(i = 1, ..., n)$.

To estimate θ, an aggregate prediction error, based on residuals $r_i(\theta) = Y_i - X_i^t\theta$, is minimized.

- **LS-estimator:** $\hat{\theta}_{LS} = \arg\min_{\theta} \sum_{i=1}^{n} r_i^2(\theta)$ (`regress`)
  fragile to all types of outliers

Robust estimators

Consider regression model

$$Y_i = X_i^t \theta + \varepsilon_i$$

where Y_i is the dependent variable, X_i is the vector of covariates and ε_i is the error term $(i = 1, ..., n)$.

To estimate θ, an aggregate prediction error, based on residuals $r_i(\theta) = Y_i - X_i^t \theta$, is minimized.

- **LS-estimator:** $\hat{\theta}_{LS} = \arg\min_{\theta} \sum_{i=1}^{n} r_i^2(\theta)$ (`regress`)
  <u>fragile to all types of outliers</u>
- **M-estimators:** $\hat{\theta}_M = \arg\min_{\theta} \sum_{i=1}^{n} \rho\left(\frac{r_i(\theta)}{\sigma}\right)$ (`qreg, rreg`)
  <u>fragile to bad leverage points</u>

Robust estimators

Consider regression model

$$Y_i = X_i^t\theta + \varepsilon_i$$

where Y_i is the dependent variable, X_i is the vector of covariates and ε_i is the error term $(i = 1, ..., n)$.

To estimate θ, a measure s of the dispersion of the residuals $r_i(\theta) = Y_i - X_i^t\theta$ is minimized.

- **LS-estimator:** $\hat{\theta}_{LS} = \arg\min_{\theta} \frac{1}{n}\sum_{i=1}^{n} r_i^2(\theta)$ or equivalently

Robust estimators

Consider regression model

$$Y_i = X_i^t\theta + \varepsilon_i$$

where Y_i is the dependent variable, X_i is the vector of covariates and ε_i is the error term $(i = 1, ..., n)$.

To estimate θ, a measure s of the dispersion of the residuals $r_i(\theta) = Y_i - X_i^t\theta$ is minimized.

- **LS-estimator:** $\hat{\theta}_{LS} = \arg\min_{\theta} \frac{1}{n}\sum_{i=1}^{n} r_i^2(\theta)$ or equivalently
- **LS-estimator:** $\begin{cases} \min_{\theta} s(r_1(\theta), ..., r_n(\theta)) \\ \text{s.t.} \frac{1}{n}\sum_{i=1}^{n}\left(\frac{Y_i - X_i^t\theta}{s}\right)^2 = 1 \end{cases}$

Robust estimators

S-estimator of regression

The square function in LS awards excessive importance to outliers. To increase robustness, another function $\rho_0(\cdot)$ (even, non decreasing for positive values, less increasing than the square with a minimum at zero) should be preferred

- **LS-estimator:** $\begin{cases} \min\limits_{\theta} s(r_1(\theta), ..., r_n(\theta)) \\ \text{s.t.} \frac{1}{n} \sum\limits_{i=1}^{n} \left(\frac{Y_i - X_i^t \theta}{s} \right)^2 = 1 \end{cases}$

Remark: for a thorough description of the robust M, S, MM, MS and SD estimators presented in this talk, we advice to refer to: Maronna, R., Martin, D.R. and Yohai, V.J. (2006). "Robust Statistics: Theory and Methods", Wiley.

Ref: Rousseeuw, P. and Yohai, V. (1984), "Robust Regression by Means of S-estimators" in Robust and nonlinear time series analysis, pages 256–272.

Robust estimators

S-estimator of regression

The square function in LS awards excessive importance to outliers. To increase robustness, another function $\rho_0(\cdot)$ (even, non decreasing for positive values, less increasing than the square with a minimum at zero) should be preferred

- **LS-estimator:** $\begin{cases} \min\limits_{\theta} s(r_1(\theta), ..., r_n(\theta)) \\ \text{s.t.} \frac{1}{n} \sum\limits_{i=1}^{n} \left(\frac{Y_i - X_i^t \theta}{s} \right)^2 = 1 \end{cases}$

- **S-estimator:** $\begin{cases} \min\limits_{\theta} s(r_1(\theta), ..., r_n(\theta)) \\ \text{s.t.} \frac{1}{n} \sum\limits_{i=1}^{n} \rho_0 \left(\frac{Y_i - X_i^t \theta}{s} \right) = \delta \end{cases}$

  where $\delta = E\left[\rho_0\left(u\right)\right]$ with $u \backsim N(0, 1)$

Robust estimators

Tukey Biweight Function

Several ρ_0 functions can be used. We chose Tukey's Biweight function here defined as

$$\rho_0(u) = \begin{cases} \frac{c^2}{6}\left(1 - \left[1 - \left(\frac{u}{c}\right)^2\right]^3\right) \text{ if } |u| \leq c \\ \frac{c^2}{6} \text{ if } |u| > c \end{cases} .$$

There is a **trade-off** between **robustness** and Gaussian **efficiency**

- $c = 1.56$ leads to a 50% BDP and an efficiency of 28%

Robust estimators

Tukey Biweight Function

Several ρ_0 functions can be used. We chose Tukey's Biweight function here defined as

$$\rho_0(u) = \begin{cases} \frac{c^2}{6}\left(1-\left[1-\left(\frac{u}{c}\right)^2\right]^3\right) \text{ if } |u| \leq c \\ \frac{c^2}{6} \text{ if } |u| > c \end{cases}.$$

There is a **trade-off** between **robustness** and Gaussian **efficiency**

- $c = 1.56$ leads to a 50% BDP and an efficiency of 28%
- $c = 3.42$ leads to a 20% BDP and an efficiency of 85%

Robust estimators

Tukey Biweight Function

Several ρ_0 functions can be used. We chose Tukey's Biweight function here defined as

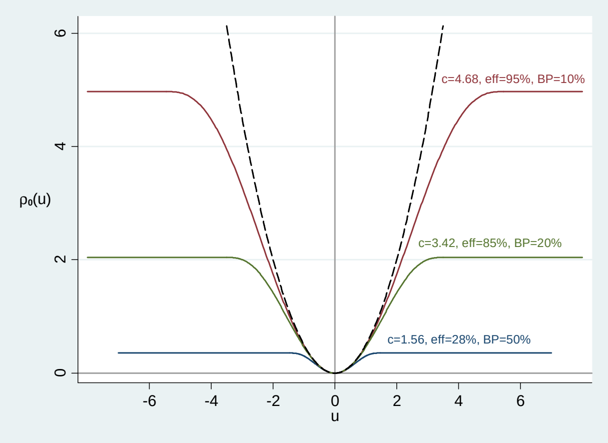
$$\rho_0(u) = \begin{cases} \frac{c^2}{6}\left(1 - \left[1 - \left(\frac{u}{c}\right)^2\right]^3\right) \text{ if } |u| \leq c \\ \frac{c^2}{6} \text{ if } |u| > c \end{cases}.$$

There is a **trade-off** between **robustness** and Gaussian **efficiency**

- $c = 1.56$ leads to a 50% BDP and an efficiency of 28%
- $c = 3.42$ leads to a 20% BDP and an efficiency of 85%
- $c = 4.68$ leads to a 10% BDP and an efficiency of 95%

Robust estimators

Tukey Biweight Function



Robust estimators

MM-estimators (Yohai,1987)

Fit an S-estimator of regression with 50% BDP and estimate the scale parameter

$$\hat{\sigma}_S = s(r_1(\hat{\theta}_S), \ldots, r_n(\hat{\theta}_S)).$$

Take another function $\rho \geq \rho_0$ and estimate:

$$\hat{\theta}_{MM} = \arg\min_{\theta} \sum_{i=1}^{n} \rho\left(\frac{r_i(\theta)}{\hat{\sigma}_S}\right)$$

The BDP is set by ρ_0 and the efficiency by ρ.

Ref: Yohai., V, J, (1987) "High Breakdown-Point and High Efficiency Robust Estimates for Regression." Ann. Statist. 15 (2) 642 - 656.

P-subset

Subsampling algorithms to approach the best solution

Exact formulas do not exist to estimate these models and subsampling algorithms are needed:

1. Consider enough subsets of **p**-points to be sure that at least one does not contain outliers.

P-subset

Subsampling algorithms to approach the best solution

Exact formulas do not exist to estimate these models and subsampling algorithms are needed:

1. Consider enough subsets of **p**-points to be sure that at least one does not contain outliers.
2. For each subset fit the hyperplane connecting all points and use it as a first guess of the robust estimated hyperplane.

P-subset

Subsampling algorithms to approach the best solution

Exact formulas do not exist to estimate these models and subsampling algorithms are needed:

1. Consider enough subsets of **p**-points to be sure that at least one does not contain outliers.
2. For each subset fit the hyperplane connecting all points and use it as a first guess of the robust estimated hyperplane.
3. Do some fine tuning using iteratively reweighted least squares based on the residuals estimated in (3) to get closer to the global solution

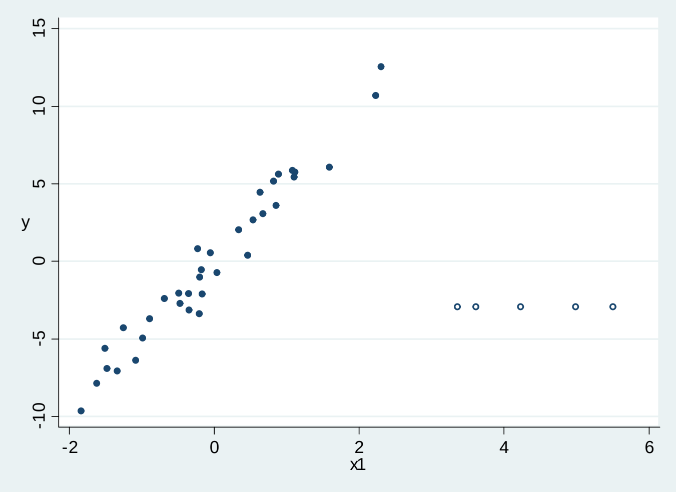
P-subset

Subsampling algorithms to approach the best solution

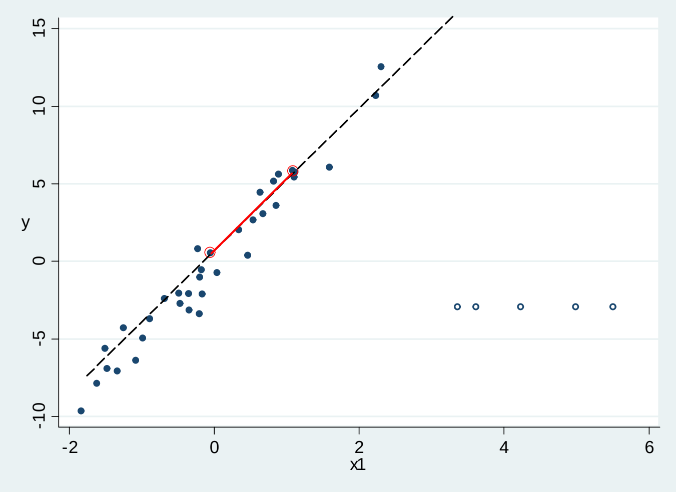
Exact formulas do not exist to estimate these models and subsampling algorithms are needed:

1. Consider enough subsets of **p**-points to be sure that at least one does not contain outliers.
2. For each subset fit the hyperplane connecting all points and use it as a first guess of the robust estimated hyperplane.
3. Do some fine tuning using iteratively reweighted least squares based on the residuals estimated in (3) to get closer to the global solution
4. Keep the result associated to the refined estimator associated with the smallest (robust) aggregate error.

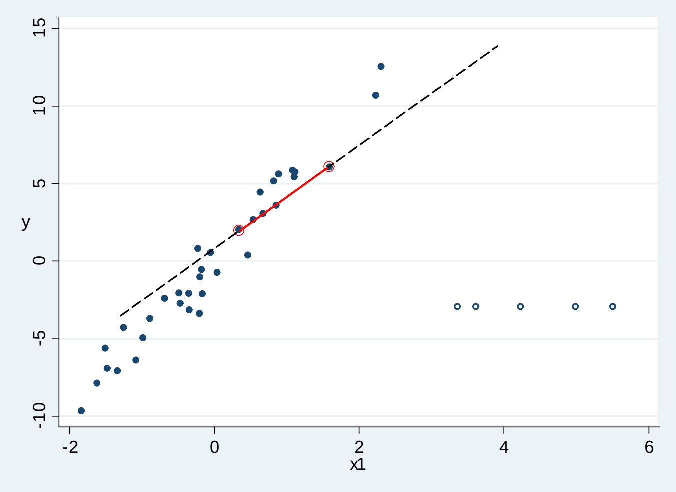
Scatter diagram



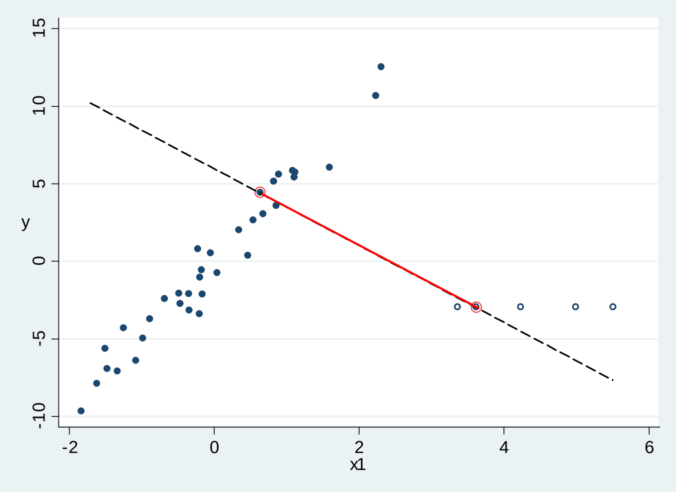
First subset



Second subset



Third subset



Problematic when several dummies are present

It is very likely to observe perfectly collinear subsamples.

id	y	x1	d1	d2	d3
1	0.114251	0.694536	0	0	0
2	0.934258	0.029458	1	1	0
3	0.565081	0.247579	0	0	0
4	0.876498	0.915357	0	0	0
5	0.710484	0.656413	0	0	0
6	0.856098	0.93658	0	0	1
7	0.521096	0.085324	1	1	0

Problem

If there are five independent explanatory dummy variables that, for example, take value 1 with probability 0.1, the likelihood of selecting a non-collinear sample of size 5 is only 1.1%

The MS-estimator is a first solution

Consider regression model

$$y = \underbrace{X_1}_{dummies} \theta_1 + \underbrace{X_2}_{continuous} \theta_2 + \varepsilon$$

- If θ_2 were known, then θ_1 could be robustly estimated using a monotonic M-estimator (no leverage points)

Ref: Maronna, R. A., and Yohai, V. J. (2000). "Robust regression with both continuous and categorical predictors". Journal of Statistical Planning and Inference 89, 197–214.

The MS-estimator is a first solution

Consider regression model

$$y = \underbrace{X_1}_{dummies} \theta_1 + \underbrace{X_2}_{continuous} \theta_2 + \varepsilon$$

- If θ_2 were known, then θ_1 could be robustly estimated using a monotonic M-estimator (no leverage points)
- If θ_1 were known, then θ_2 should be estimated using an S-estimator. The subsampling algorithm **would not** generate collinear subsamples as only continuous variables would be present.

The MS-estimator is a first solution

Consider regression model

$$y = \underbrace{X_1}_{dummies} \theta_1 + \underbrace{X_2}_{continuous} \theta_2 + \varepsilon$$

- If θ_2 were known, then θ_1 could be robustly estimated using a monotonic M-estimator (no leverage points)
- If θ_1 were known, then θ_2 should be estimated using an S-estimator. The subsampling algorithm **would not** generate collinear subsamples as only continuous variables would be present.

Alternate

$$\begin{cases} \hat{\theta}_1^{MS} = \arg\min_{\theta_1} \sum_{i=1}^{n} \rho\left([y_i - X_{2i}\hat{\theta}_2] - X_{1i}\theta_1\right) \\ \hat{\theta}_2^{MS} = \arg\min_{\theta_2} \hat{\sigma}^S\left([y - X_1\hat{\theta}_1] - X_2\theta_2\right) \end{cases}$$

The SD-estimator is a second solution solution

Consider regression model

$$y = \underbrace{X_1}_{dummies} \theta_1 + \underbrace{X_2}_{continuous} \theta_2 + \varepsilon$$

- To identify outliers matrix $M_{n \times q} = (y, X_2)$ is projected in "all" possible directions and dummies are partialled out on each projection using any monotonic M-estimator.

Ref:
Stahel, W. A. (1981). "Robust estimation: Infinitesimal optimality and covariance matrix estimators". Ph.D. thesis, ETH, Zurich
and
Donoho, D. L. (1982). "Breakdown properties of multivariate location estimators". Qualifying paper, Dept. Statistics, Harvard Univ.

The SD-estimator is a second solution solution

Consider regression model

$$y = \underbrace{X_1}_{dummies} \theta_1 + \underbrace{X_2}_{continuous} \theta_2 + \varepsilon$$

- To identify outliers matrix $M_{n \times q} = (y, X_2)$ is projected in "all" possible directions and dummies are partialled out on each projection using any monotonic M-estimator.
- The outlyingness of a given point is then defined as the maximum distance from the projection of the point to the center of the projected data cloud, i.e. $\delta_i = \max_{\|a\|=1} \frac{|\tilde{z}_i(a)|}{\hat{s}(\tilde{z}(a))}$.

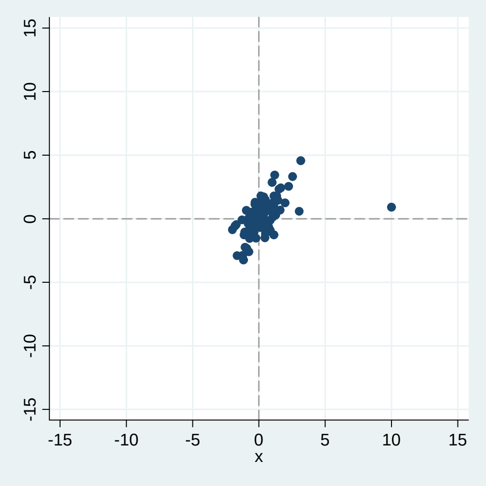
The SD-estimator is a second solution solution

Consider regression model

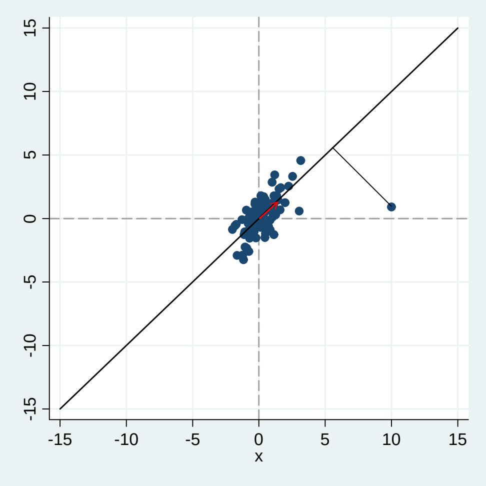
$$y = \underbrace{X_1}_{dummies} \theta_1 + \underbrace{X_2}_{continuous} \theta_2 + \varepsilon$$

- To identify outliers matrix $M_{n \times q} = (y, X_2)$ is projected in "all" possible directions and dummies are partialled out on each projection using any monotonic M-estimator.
- The outlyingness of a given point is then defined as the maximum distance from the projection of the point to the center of the projected data cloud, i.e. $\delta_i = \max_{\|a\|=1} \frac{|\tilde{z}_i(a)|}{\hat{s}(\tilde{z}(a))}$.
- Outlyingness distance δ_i is distributed as $\sqrt{\chi^2_q}$. We can therefore define an individual as being an outlier if δ_i is larger than a chosen quantile of $\sqrt{\chi^2_q}$.

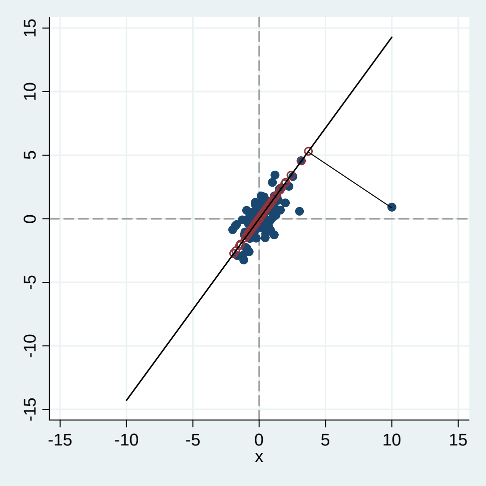
The SD-estimator: a graphical explanation



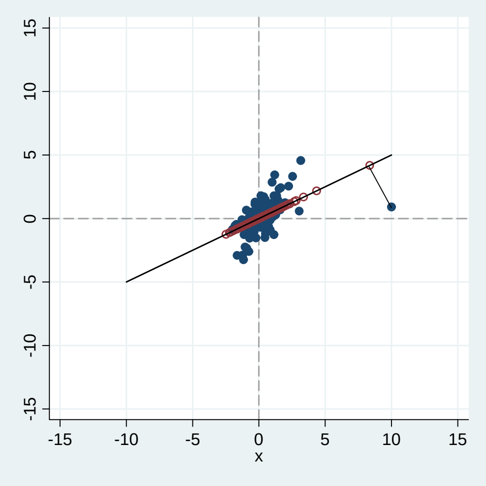
The SD-estimator: a graphical explanation



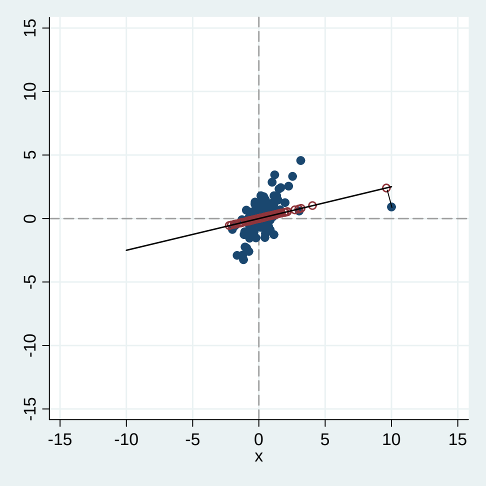
The SD-estimator: a graphical explanation



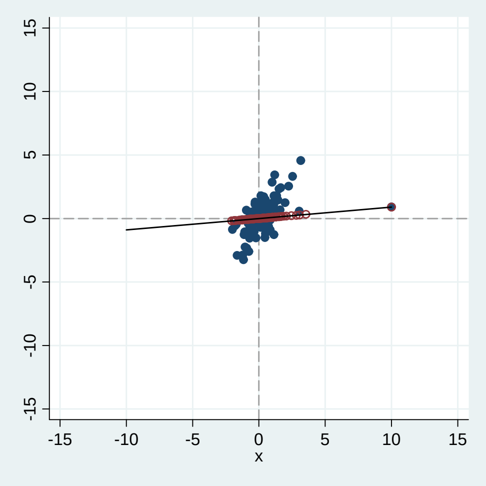
The SD-estimator: a graphical explanation



The SD-estimator: a graphical explanation



The SD-estimator: a graphical explanation



Comparative advantages

- We programmed both estimators. They are available upon request; `robregms` and `sdmultiv`
- Both estimators can be used to fit distributed intercept models (such as LSDV)
- MS is more intuitive as it relies on IRWLS. SD is slightly more complicated theoretically.
- SD can be used to identify outliers in a wide variety of models since it does not rely on the dependent-explanatory relation (i.e. Logit, Heckman)
- SD can be used in multivariate analysis (i.e. calculate robust leverage taking into account dummies)

Computing time (5% of contamination in x1)

Model: $y = \sum_{j=1}^{5} \beta_j x_j + \sum_{k=1}^{K} \gamma_j d_j + \varepsilon$ for $K = 1, 11, 21, ..., 191$.

# Dummies	MS	SD	# Dummies	MS	SD
1	2.52	1.26	101	29.19	14.59
11	3.46	1.73	111	44.94	22.47
21	4.03	2.01	121	47.42	23.71
31	5.97	2.99	131	57.06	28.53
41	8.02	4.01	141	67.19	33.60
51	10.26	5.13	151	69.62	34.81
61	11.73	5.86	161	260.07	130.03
71	16.23	8.12	171	139.56	69.78
81	20.83	10.42	181	134.95	67.48
91	27.23	13.61	191	185.18	92.59

$N = 1000$

Creating a contaminated sample

```
clear
set obs 1000
drawnorm x1-x5 e
gen y=x1+x2+x3+x4+x5+e
forvalues i=1(1)5 {
gen d`i'=round(uniform())
replace y=y+d`i'
}
replace x1=10 in 1/100
robregms y x* d*
sdmultiv y x* d*, gen(a b)
reg y x* d* if a==0
reg y x* d*
```

MS-estimator

y	Coef.	Robust Std. Err.	z
x1	1.015749	.0847334	11.99
x2	.9840165	.0588595	16.72
x3	1.083979	.0527653	20.54
x4	1.052281	.0752983	13.97
x5	1.052403	.0676575	15.55
d1	1.124173	.1066948	10.54
d2	1.120287	.1195124	9.37
d3	1.011536	.1144117	8.84
d4	.7388712	.1095223	6.75
d5	1.124374	.1448934	7.76
_cons	-.0289221	.1153801	-0.25

Creating a contaminated sample

```
clear
set obs 1000
drawnorm x1-x5 e
gen y=x1+x2+x3+x4+x5+e
forvalues i=1(1)5 {
gen d`i'=round(uniform())
replace y=y+d`i'
}
replace x1=10 in 1/100
robregms y x* d*
sdmultiv y x* d*, gen(a b)
reg y x* d* if a==0
reg y x* d*
```

SD-estimator

y	Coef.	Std. Err.	t
x1	1.041384	.0362514	28.73
x2	1.04519	.0344185	30.37
x3	1.031552	.0345838	29.83
x4	1.066473	.0356224	29.94
x5	1.081784	.0346054	31.26
d1	1.061789	.0644784	16.47
d2	.9851284	.064193	15.35
d3	.9224563	.0643582	14.33
d4	.8450953	.0647661	13.05
d5	1.112151	.0643558	17.28
_cons	.0504445	.0767734	0.66

Creating a contaminated sample

```
clear
set obs 1000
drawnorm x1-x5 e
gen y=x1+x2+x3+x4+x5+e
forvalues i=1(1)5 {
gen d`i'=round(uniform())
replace y=y+d`i'
}
replace x1=10 in 1/100
robregms y x* d*
sdmultiv y x* d*, gen(a b)
reg y x* d* if a==0
reg y x* d*
```

LS-estimator

y	Coef.	Std. Err.	t
x1	.095752	.0137521	6.96
x2	1.041461	.0443404	23.49
x3	1.049491	.0440445	23.83
x4	.9723244	.0442183	21.99
x5	1.031377	.0436638	23.62
d1	.925825	.0866913	10.68
d2	1.018304	.0866161	11.76
d3	1.005827	.0867442	11.60
d4	.9866187	.0868825	11.36
d5	1.084528	.0867098	12.51
_cons	-.130481	.1048221	-1.24

Main points of the talk

- Robust models can cope with dummies
- Codes are relatively fast and stable
- SD opens the door to outlier identification in a very large variety of models
- SD can be used in many other contexts than regression analysis