Session 4: Robust regression
Winter course, CMStatistics 2016

Mia Hubert, Peter Rousseeuw, Stefan Van Aelst

Department of Mathematics
KU Leuven, Belgium

December 6–7, 2016

Outline of the course

- 1. General notions of robustness
- 2. Robustness for univariate data
- 3. Robust multivariate methods
- 4. Robust regression
- 5. Robust principal component analysis
- 6. Inference
- 7. Multivariate and functional depth
- 8. High dimensional data and sparsity
- 9. Cellwise outliers

Linear regression: Outline

The linear regression model

The linear regression model says:

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{i1} + \ldots + \beta_p x_{ip} + \varepsilon_i \\ &= \boldsymbol{x}_i' \boldsymbol{\beta} + \varepsilon_i \end{aligned}$$

with i.i.d. errors $\varepsilon_i \sim N(0, \sigma^2)$, $\boldsymbol{x}_i = (1, x_{i1}, \ldots, x_{ip})'$ and $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)'$.

Denote the $n \times (p+1)$ matrix containing the predictors $\boldsymbol{x}_i$ as $X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)'$, the vector of responses $\boldsymbol{y} = (y_1, \ldots, y_n)'$ and the error vector $\boldsymbol{\varepsilon} = (\varepsilon_1, \ldots, \varepsilon_n)'$. Then:

$$\boxed{\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}}$$

Any regression estimate $\hat{\boldsymbol{\beta}}$ yields fitted values $\hat{\boldsymbol{y}} = X\hat{\boldsymbol{\beta}}$ and residuals

$$r_i = r_i(\hat{\boldsymbol{\beta}}) = y_i - \hat{y}_i \quad .$$

The least squares estimator

Least squares estimator

$$\hat{\boldsymbol{\beta}}_{LS} = \underset{\boldsymbol{\beta}}{\text{argmin}} \sum_{i=1}^{n} r_i^2(\boldsymbol{\beta})$$

If X has full rank, then

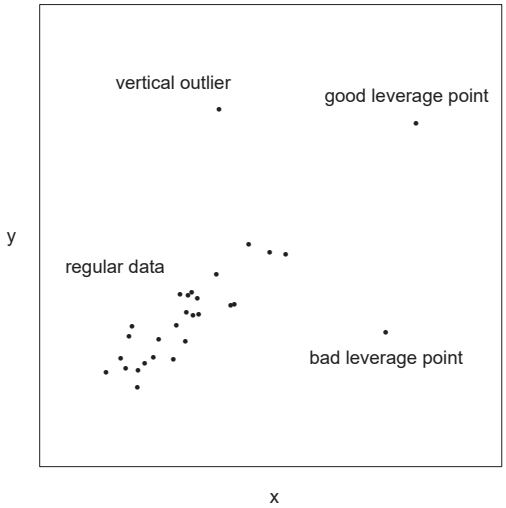
$$\boxed{\hat{\boldsymbol{\beta}}_{LS} = (X'X)^{-1}X'\boldsymbol{y}}$$

The usual unbiased estimator of the error variance is

$$\boxed{\hat{\sigma}^2_{LS} = \frac{1}{n-p-1}\sum_{i=1}^{n} r_i^2(\hat{\boldsymbol{\beta}}_{LS})}$$

Outliers in regression

Different types of outliers:

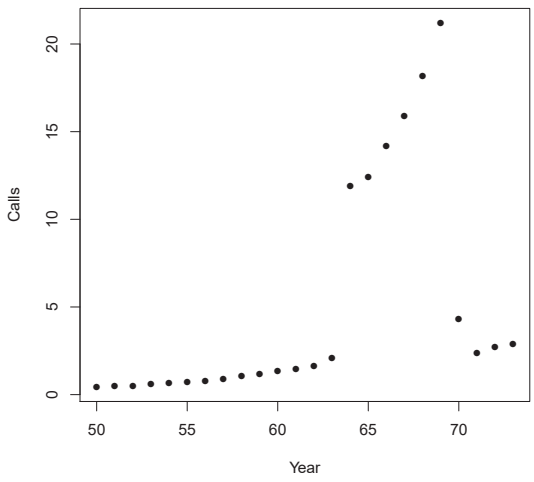


Outliers in regression

1. **regular observations**: internal $\boldsymbol{x}_i$ and well-fitting y_i
2. **vertical outliers**: internal $\boldsymbol{x}_i$ and non-fitting y_i
3. **good leverage points**: outlying $\boldsymbol{x}_i$ and well-fitting y_i
4. **bad leverage points**: outlying $\boldsymbol{x}_i$ and non-fitting y_i

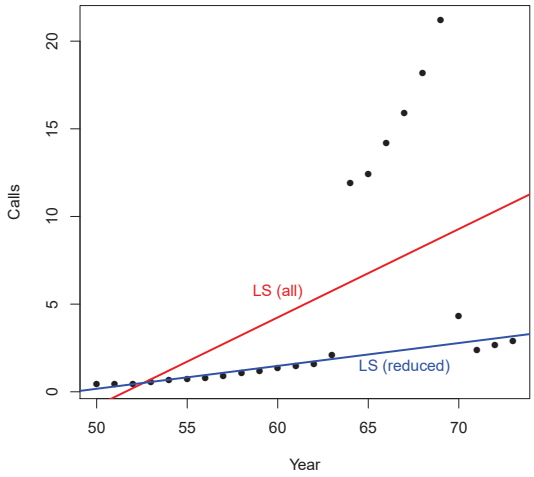
Effect of vertical outliers

Example: **Telephone** data set, which contains the *number* of international telephone calls (in tens of millions) from Belgium in the years 1950-1973.



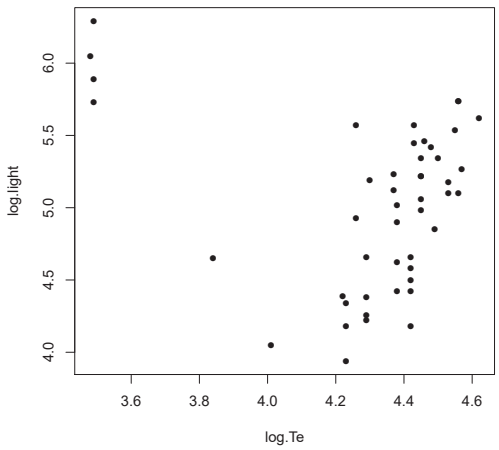
Effect of vertical outliers

LS fit with and without the outliers:



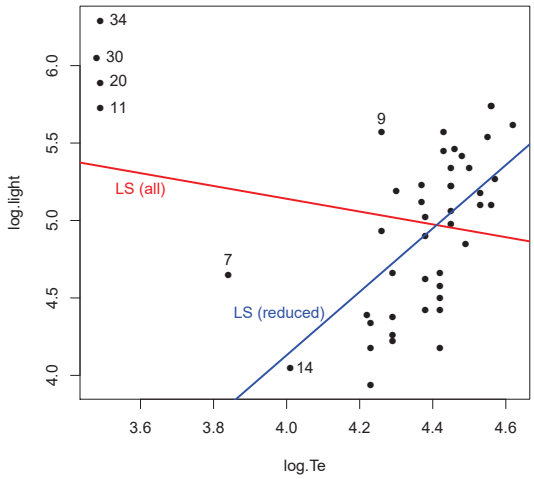
Effect of bad leverage points

Example: **Stars** data set: Hertzsprung-Russell diagram of the star cluster CYG OB1 (47 stars). Here X is the logarithm of a star's surface temperature, and Y is the logarithm of its light intensity.



Effect of bad leverage points

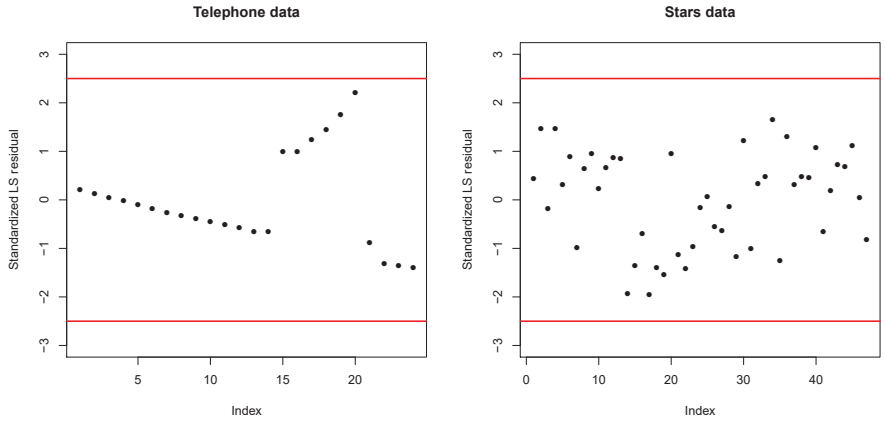
LS fit with and without the giant stars:



Classical outlier diagnostics

Standardized residuals

This residual plot shows the **standardized LS residuals** $\frac{r_i(\hat{\boldsymbol{\beta}}_{LS})}{\hat{\sigma}_{LS}}$



Studentized residuals

Residual plot of the **studentized LS residuals** given by:

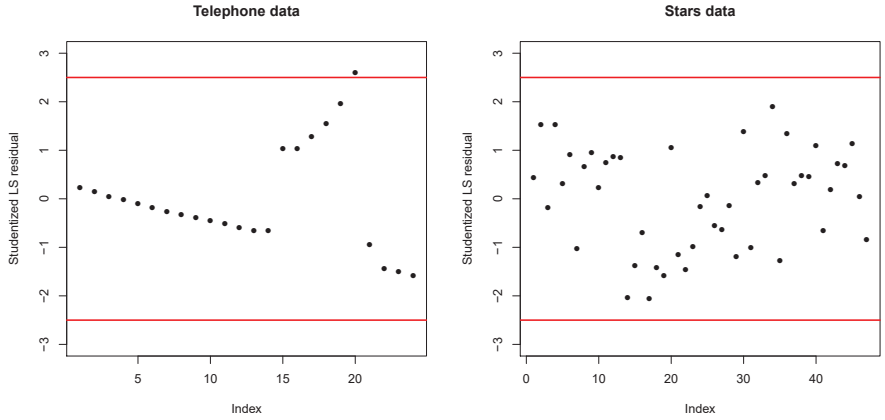
1. remove observation $(\boldsymbol{x}_i, y_i)$ from the data set
2. compute $\hat{\boldsymbol{\beta}}_{LS}^{(i)}$ on the remaining data
3. compute the fitted value of y_i given by $\hat{y}_i^{(i)} = \boldsymbol{x}_i' \hat{\boldsymbol{\beta}}_{LS}^{(i)}$
4. compute the "deleted residual":

$$d_i = y_i - \hat{y}_i^{(i)}$$

5. the studentized residuals are $r_i^* = d_i / s(d_j)$ where $s(d_i)$ is the standard deviation of all d_j .

The studentized residuals can be computed without refitting the model each time an observation is deleted.

Studentized residuals



Hat matrix

The hat matrix

$$H = X(X'X)^{-1}X'$$

transforms the observed response vector $\boldsymbol{y}$ into its LS estimate:

$$\boxed{\hat{\boldsymbol{y}} = H\boldsymbol{y}}$$

or equivalently

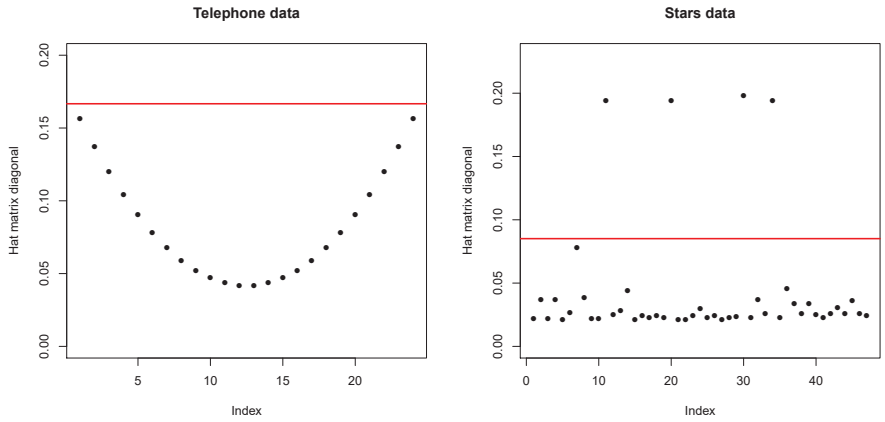
$$\hat{y}_i = h_{i1}y_1 + h_{i2}y_2 + \ldots + h_{in}y_n \ .$$

The element h_{ij} of H thus measures the effect of the jth observation on $\hat{y}_i$, and the **diagonal element** h_{ii} the effect of the ith observation on its own prediction.

Since it holds that average$(h_{ii}) = (p+1)/n$ and $0 \leqslant h_{ii} \leqslant 1$, it is sometimes suggested to call observation i a leverage point iff

$$h_{ii} > \frac{2(p+1)}{n} \ .$$

Hat matrix



Hat matrix

It can be shown that there is a one-to-one correspondence between the squared Mahalanobis distance for object i and its h_{ii}:

$$h_{ii} = \frac{1}{n-1}\mathsf{MD}_i^2 + \frac{1}{n}$$

with

$$\mathsf{MD}_i = \mathsf{MD}(\boldsymbol{x}_i) = \sqrt{(\boldsymbol{x}_i - \bar{\boldsymbol{x}}_n)' S_n^{-1} (\boldsymbol{x}_i - \bar{\boldsymbol{x}}_n)} \ .$$

From this expression we see that h_{ii} measures the distance of $\boldsymbol{x}_i$ to the center of the data points in the $\boldsymbol{x}$-space.
On the other hand, it shows that the h_{ii} diagnostic is based on nonrobust estimates! Indeed, it often masks outlying $\boldsymbol{x}_i$.

Cook's distance

Cook's distance D_i measures the influence of the ith case on all n fitted values:

$$D_i = \frac{(\hat{\boldsymbol{y}} - \hat{\boldsymbol{y}}^{(i)})'(\hat{\boldsymbol{y}} - \hat{\boldsymbol{y}}^{(i)})}{(p+1)\hat{\sigma}^2_{LS}} \quad .$$

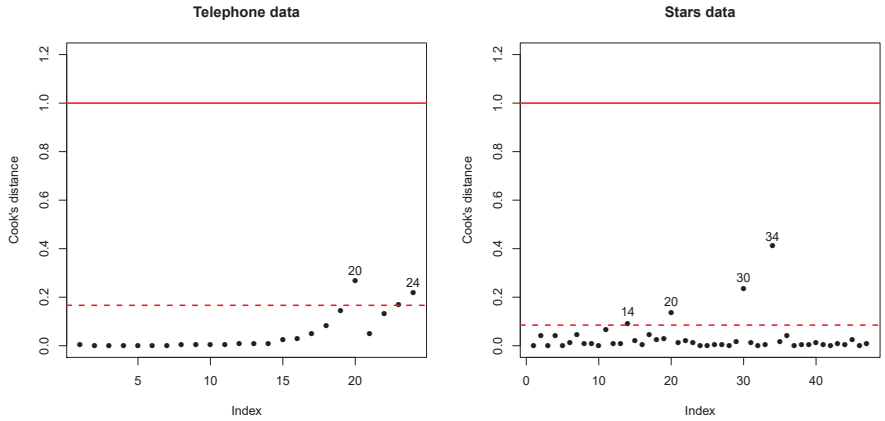
It is also equivalent to

$$D_i = \frac{(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}^{(i)})'(X'X)(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}^{(i)})}{(p+1)\hat{\sigma}^2_{LS}} \quad .$$

In this sense D_i measures the influence of the ith case on the regression coefficients.

Often the cutoff values 1 or $4/n$ are suggested.

Cook's distance



Conclusion

- Also other single-case diagnostics exist, e.g. DFFITS and DFBETAS, and are available in software.
- They all rely on deleting one case and measuring the effect on the fitted values or the regression estimates.
- They may not adequately detect outliers when there are several!

Regression M-estimators

Equivariance

When looking for alternative estimators, we (usually) look for estimators $\hat{\boldsymbol{\beta}}$ that are:

1. **regression equivariant**: $\hat{\boldsymbol{\beta}}(X, \boldsymbol{y} + X\boldsymbol{\gamma}) = \hat{\boldsymbol{\beta}}(X, \boldsymbol{y}) + \boldsymbol{\gamma}$ for all $\boldsymbol{\gamma} \in \mathbb{R}^{p+1}$
2. **scale equivariant**: $\hat{\boldsymbol{\beta}}(X, \lambda\boldsymbol{y}) = \lambda\hat{\boldsymbol{\beta}}(X, \boldsymbol{y})$ for all $\lambda \in \mathbb{R}$
3. **affine equivariant**: $\hat{\boldsymbol{\beta}}(XA, \boldsymbol{y}) = A^{-1}\hat{\boldsymbol{\beta}}(X, \boldsymbol{y})$ for all nonsingular $(p+1) \times (p+1)$ matrices A.

Equivariance

A regression estimator is said to break down if $\|\hat{\boldsymbol{\beta}}\|$ becomes arbitrary large.

All regression equivariant estimators satisfy:

$$\varepsilon_n^*(\hat{\boldsymbol{\beta}}) \leqslant \frac{1}{n}\left[\frac{n-p-1}{2}\right] \quad .$$

Regression M-estimators

Regression M-estimator

$$\hat{\boldsymbol{\beta}}_M = \underset{\boldsymbol{\beta}}{\text{argmin}} \frac{1}{n} \sum_{i=1}^{n} \rho \left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}} \right)$$

with $\hat{\sigma}$ a preliminary scale estimate.

For $\psi = \rho'$ it follows that

$$\sum_{i=1}^{n} \psi \left(\frac{r_i(\hat{\boldsymbol{\beta}}_M)}{\hat{\sigma}} \right) \boldsymbol{x}_i = \mathbf{0} \ . \tag{1}$$

Regression M-estimators

- For $\rho(t) = t^2$ we obtain $\hat{\boldsymbol{\beta}}_M = \hat{\boldsymbol{\beta}}_{LS}$.
- For $\rho(t) = |t|$ we obtain

$$\hat{\boldsymbol{\beta}}_{L^1} = \underset{}{\text{argmin}} \sum_{i=1}^{n} |r_i(\boldsymbol{\beta})|$$

- Initial scale estimate:

$$\hat{\sigma} = \frac{1}{0.675} \underset{i}{\text{med}} \, |r_i(\hat{\boldsymbol{\beta}}_{L^1})|$$

- A monotone M-estimator, i.e. with a weakly increasing ψ function, has a unique solution.
- A redescending M-estimator can have multiple solutions. It thus requires a robust starting point.

Computation of regression M-estimators

With $W(x) = \psi(x)/x$ and $w_i = W(r_i(\hat{\boldsymbol{\beta}}_M)/\hat{\sigma})$, the M-estimator equation (1) can be rewritten as:

$$\sum_{i=1}^{n} w_i r_i(\hat{\boldsymbol{\beta}}_M)\boldsymbol{x}_i = \mathbf{0} \ .$$

This suggests an iterative reweighted least squares (IRLS) procedure:

- Compute $\hat{\boldsymbol{\beta}}_{L^1}$ as initial fit with corresponding scale $\hat{\sigma}$
- For $k = 0, 1, \ldots$ do:
  - compute $r_{i,k} = r_i(\hat{\boldsymbol{\beta}}_k)$ and $w_{i,k} = W(r_{i,k}/\hat{\sigma})$
  - compute $\hat{\boldsymbol{\beta}}_{k+1}$ by solving

$$\sum_{i=1}^{n} w_{i,k} r_i(\hat{\boldsymbol{\beta}}_{k+1})\boldsymbol{x}_i = \mathbf{0} \ .$$

    This is the LS fit to the points $(\sqrt{w_{i,k}}\boldsymbol{x}_i, \sqrt{w_{i,k}}y_i)$.
- Stop when $\max_i(|r_{i,k} - r_{i,k+1}|/\hat{\sigma}) < \varepsilon$.

Efficiency and nonrobustness of regression M-estimators

- M-estimators are asymptotically normal with an asymptotic covariance matrix which depends on the design matrix through $X'X$ and on the ψ-function.
  - for Huber and bisquare, the efficiency depends on the tuning constants b and c
  - inference can be based on the estimated asymptotic covariance matrix.
- The IF of regression M-estimators is unbounded.
  - If ψ is monotone, the IF tends to infinity at all y when $\boldsymbol{x}$ tends to infinity.
  - If ψ is redescending, the IF will only be unbounded for good leverage points.
- As the weights $w_i = W(r_i(\hat{\boldsymbol{\beta}}_M)/\hat{\sigma})$ only penalize large residuals, but not leverage points,

$$\varepsilon^*(\hat{\boldsymbol{\beta}}_M) = 0\% \ .$$

  Breakdown only occurs when there are leverage points. In fixed designs, regression M-estimators can attain a positive breakdown value.
- To obtain a high-breakdown estimator, one should start from a high-breakdown initial estimate (instead of $\hat{\boldsymbol{\beta}}_{L^1}$).
  Some high-breakdown estimators will be discussed below.

The LTS estimator

The LTS estimator

Least trimmed squares estimator (Rousseeuw, 1984)

For fixed h, with $[n+p+2]/2 \leqslant h \leqslant n$,

1. select the h-subset with smallest LS scale of the residuals;
2. the regression estimate $\hat{\boldsymbol{\beta}}_0$ is the LS fit to those h observations;
3. the scale estimate $\hat{\sigma}_0$ is the corresponding LS scale (multiplied by a consistency factor).

This definition is equivalent to

$$\hat{\boldsymbol{\beta}}_0 = \underset{\hat{\boldsymbol{\beta}}}{\text{argmin}} \sum_{i=1}^{h} (r^2(\hat{\boldsymbol{\beta}}))_{(i)}$$

with $r^2_{(i)}$ the ith smallest squared residual.

The LTS estimator does not try to make *all* the residuals as small as possible, but only the 'majority', where the size of the majority is given by $\alpha = h/n$.

Robustness of the LTS

- Influence function bounded in y, but unbounded at good leverage points $\boldsymbol{x}$.
- At samples *in general position*

$$\varepsilon_n^*(\hat{\boldsymbol{\beta}}_0) = \frac{(n-h+1)}{n} \quad .$$

The maximal breakdown value is achieved by taking $h = [n+p+2]/2$. Typical choices are $\alpha = h/n = 0.5$ or $\alpha = 0.75$, yielding a breakdown value of 50% and 25% respectively.

General position

A regression data set with $p+1$ covariates is in general position if any $p+1$ observations give a unique determination of $\hat{\boldsymbol{\beta}}$.

For example, no 2 observations lie on a vertical line, no 3 lie on a vertical plane, etc.

The univariate special case

The special case where $p = 0$ and x_i is the constant 1 (model with intercept only) corresponds to estimating the location and scale of the univariate sample $\boldsymbol{y} = \{y_1, \ldots, y_n\}$. In that case LTS

1. selects the h-subset of $\boldsymbol{y}$ with the smallest standard deviation;
2. the location estimator $\hat{\mu}_0 = \hat{\beta}_0$ is the mean of that h-subset;
3. the scale estimate $\hat{\sigma}_0$ is that standard deviation (multiplied by a consistency factor).

This makes it the same as the univariate special case of the MCD estimator. It can equivalently be defined as

$$\hat{\mu}_0 = \underset{\mu}{\text{argmin}} \sum_{i=1}^{h} (y-\mu)^2_{(i)}$$

where $(y-\mu)^2_{(i)}$ is the i-th order statistic of the set $\{(y_i-\mu)^2; i = 1, \ldots, n\}$. The univariate LTS location can be computed in $O(n\log(n))$ time, and is sometimes used to compute the intercept of LTS regression.

Efficiency of the LTS

The LTS is asympotically normal, but it has a low efficiency. The efficiency increases with α.

For example, for $\alpha = 0.5$ the asymptotic efficiency relative to the LS estimator is only 7%. For $\alpha = 0.75$ the relative efficiency is 27.6%.

The efficiency of the LTS can easily be increased by applying a reweighting step.

Reweighted LTS

First compute the standardized residuals $\frac{r_i(\hat{\boldsymbol{\beta}}_0)}{\hat{\sigma}_0}$.
Then apply the weight function

$$w_i = \begin{cases} 1 & \text{if } \left|\frac{r_i(\hat{\boldsymbol{\beta}}_0)}{\hat{\sigma}_0}\right| \leqslant 2.5 \\ 0 & \text{otherwise.} \end{cases}$$

Finally, the LS fit is computed on the observations with non-zero weight, resulting in $\hat{\boldsymbol{\beta}}_{LTS}$ and $\hat{\sigma}_{LTS}$.

This also yields t-values, p-values etc. that can be used for inference.

FAST-LTS

The FAST-LTS algorithm is similar to FAST-MCD:

1. For $m = 1$ to 500:
   - Draw a random subset of size $p + 1$ and compute the LS fit of this subset.
   - Apply two C-steps:
     1. Compute residuals r_i based on the most recent LS fit.
     2. Take the h observations with smallest absolute residual.
     3. Compute the LS fit of this h-subset.
2. Retain the 10 h-subsets with smallest scale estimate.
3. Apply C-steps to these subsets until convergence.
4. Retain the h-subset with smallest scale estimate.

Examples

Example: telephone data.

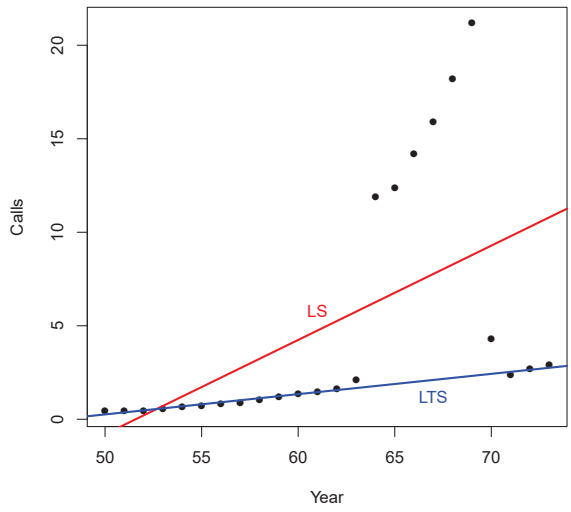
The LS estimates were:

$$\hat{\boldsymbol{\beta}}_{LS} = (-26.0059 \;\; 0.5041)'$$
$$\hat{\sigma}_{LS} = 5.6223$$

The reweighted LTS estimates (with $\alpha = 0.5$) are:

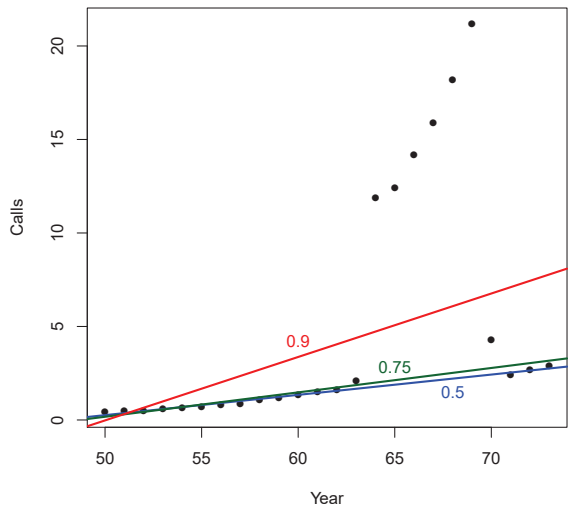
$$\hat{\boldsymbol{\beta}}_{LTS} = (-5.1645 \;\; 0.1085)'$$
$$\hat{\sigma}_{LTS} = 0.1872$$

Examples



Examples

The effect of increasing α:



Examples

Example: stars data.

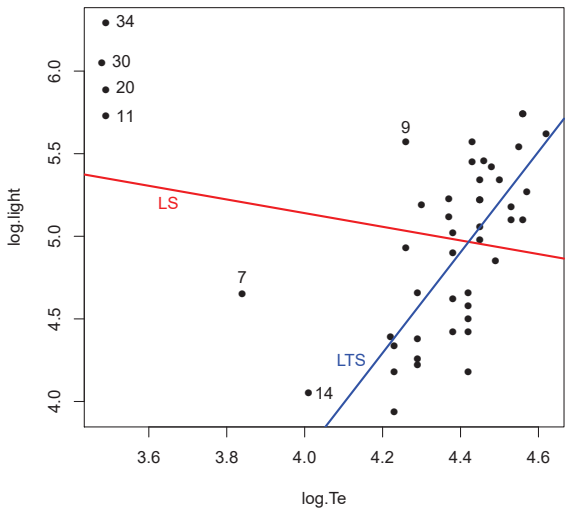
The LS estimates were:

$$\hat{\boldsymbol{\beta}}_{LS} = (6.7935 \quad -0.4133)'$$
$$\hat{\sigma}_{LS} = 0.5646$$

The reweighted LTS estimates (with $\alpha = 0.5$) are:

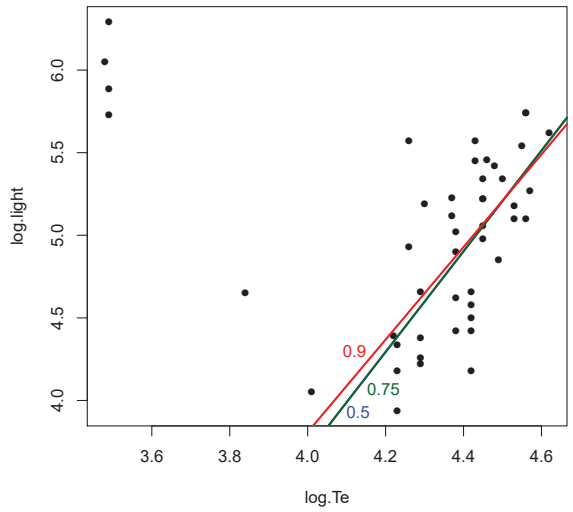
$$\hat{\boldsymbol{\beta}}_{LTS} = (-8.500 \;\; 3.046)'$$
$$\hat{\sigma}_{LTS} = 0.4562$$

Examples



Examples

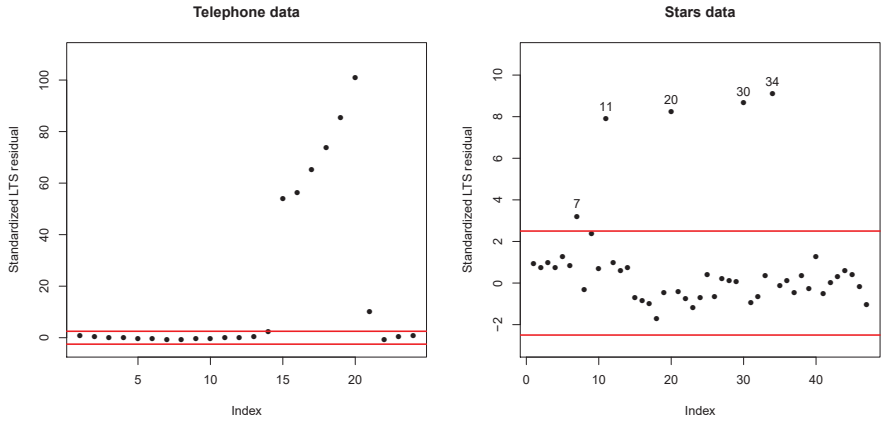
The effect of increasing α:



Outlier detection

Outlier detection

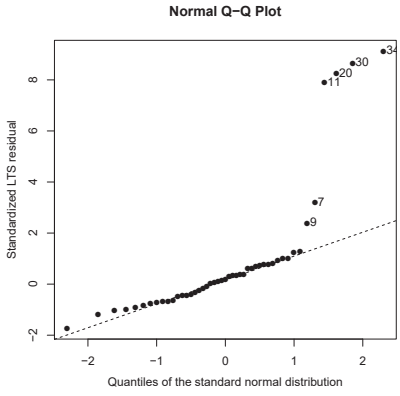
Flag observation $(\boldsymbol{x}_i, y_i)$ as an outlier if $\left|\frac{r_i(\hat{\boldsymbol{\beta}}_{LTS})}{\hat{\sigma}_{LTS}}\right| > 2.5$:



Outlier detection

Note that it is always important to check the model assumptions (for the majority of the data). Residual plots (e.g. residuals versus fitted values) allow to check the independence of the errors and the constancy of the error variance.

Additionally, a normal Q-Q plot of the residuals is helpful to check the normality of the errors. For the stars example:



Outlier map

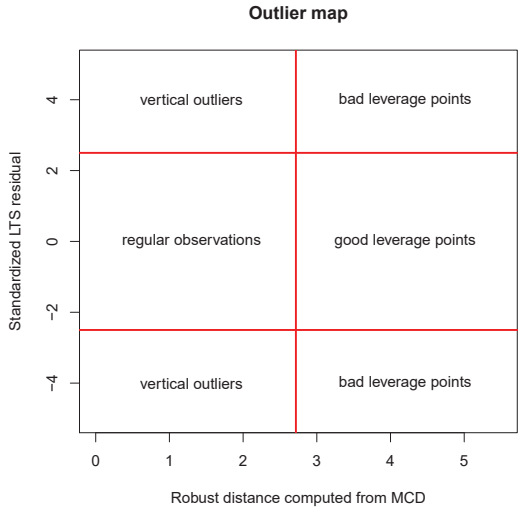
- A residual plot does not distinguish between vertical outliers and bad leverage points.
- As bad leverage points have outlying $\boldsymbol{x}_i$ we can detect them by computing the **robust distances** of all our $\boldsymbol{x}_i$:
  - Consider the components $(x_{i1}, \ldots, x_{ip})$ only (without the intercept).
  - Compute the MCD location and scatter from these points, and the corresponding robust distances.

Outlier map (Rousseeuw and Van Zomeren, 1990)

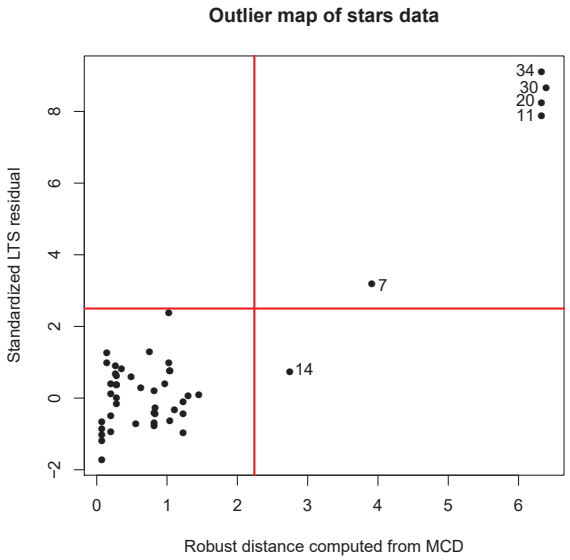
An outlier map (diagnostic plot) plots standardized robust residuals versus robust distances, together with cutoff values for the residuals (± 2.5) and for the robust distances ($\sqrt{\chi^2_{p,0.975}}$).

Outlier map

This allows to classify all data points in a regression:

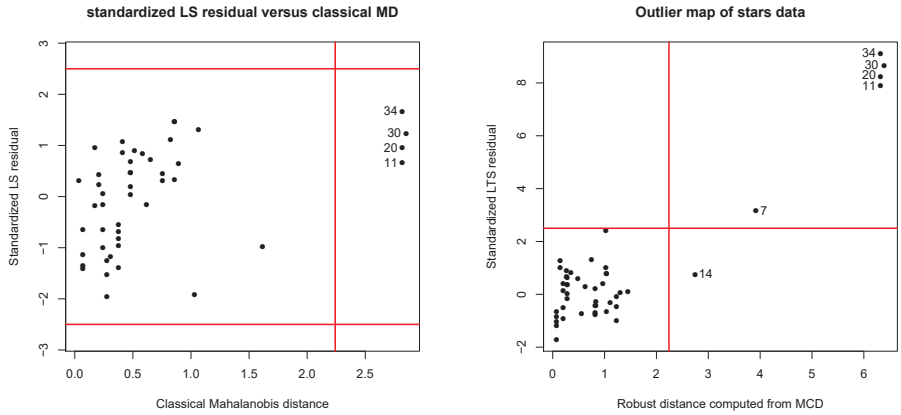


Outlier map: stars data

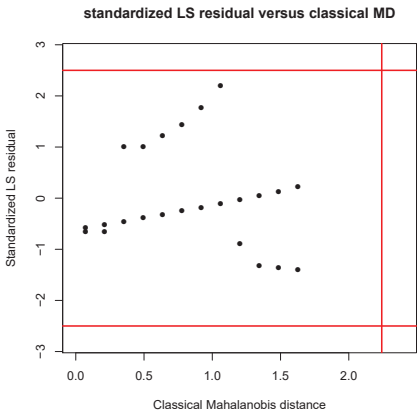


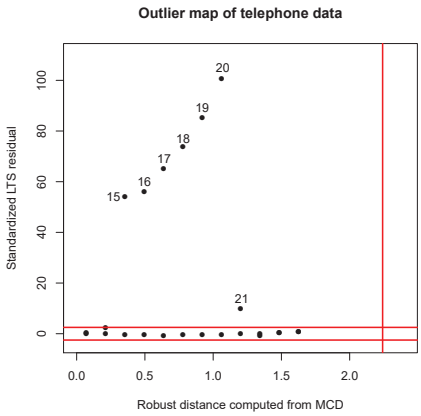
Outlier map: stars data

Left panel is corresponding plot based on classical estimates:



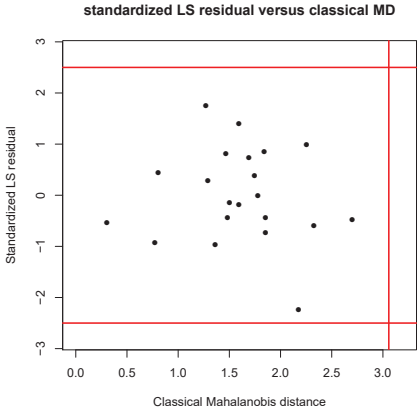
Outlier map: telephone data

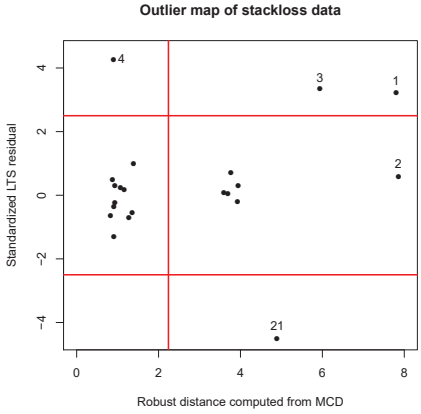




Stackloss data ($n = 21$, $p = 3$)

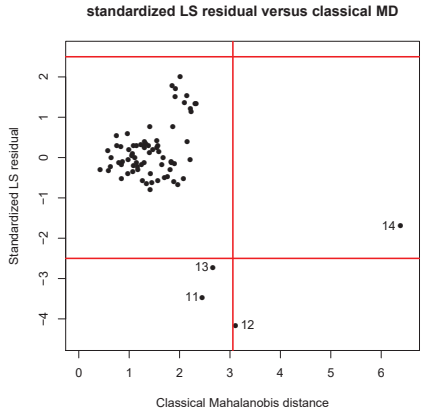
Operational data of a plant for the oxidation of ammonia to nitric acid.
For $p > 1$ we cannot rely on visual inspection!

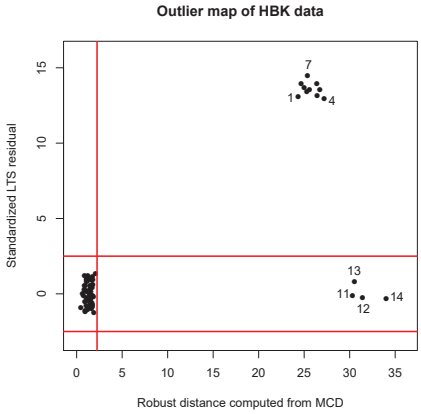




Hawkins-Bradu-Kass data

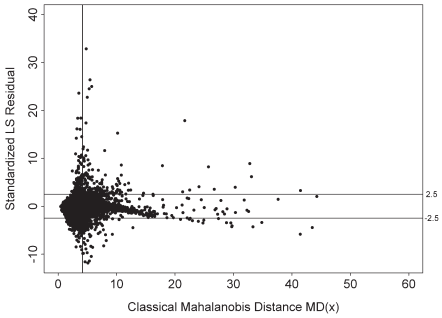
Cases 1–10 are bad leverage points, cases 11–14 are good leverage points, and the remainder is regular ($n = 75$, $p = 3$).

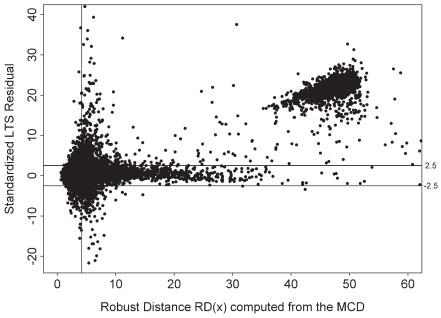




Digital sky survey

Dataset of $n = 56744$ stars (not galaxies), $p = 8$.





The bad leverage points turned out to be giant stars.

Software

Implementations of the FAST-LTS algorithm are widely available:

- R: as the function `ltsReg` in the package *robustbase*
- SAS/IML Version 7+ and SAS Version 9+ : in PROC ROBUSTREG
- Matlab: as the function `ltsregres` in the toolbox LIBRA (wis.kuleuven.be/stat/robust), and the PLS toolbox of Eigenvector Research (www.eigenvector.com)

Note that some functions use $\alpha = 0.5$ as default value, yielding a breakdown value of 50%, whereas other implementations use $\alpha = 0.75$.

Regression S-estimators

Regression S-estimators

The LS estimator can be rewritten as:

$$\hat{\boldsymbol{\beta}}_{LS} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, \hat{\sigma}(\boldsymbol{\beta})$$

with $\hat{\sigma}(\boldsymbol{\beta}) = \sqrt{\frac{1}{n-p-1}\sum_{i=1}^{n} r_i^2(\boldsymbol{\beta})}$.

The LTS estimator can be rewritten as:

$$\hat{\boldsymbol{\beta}}_{LTS} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, \hat{\sigma}(\boldsymbol{\beta})$$

with $\hat{\sigma}(\boldsymbol{\beta}) = \sqrt{\frac{1}{h-p-1}\sum_{i=1}^{h} (r^2(\boldsymbol{\beta}))_{(i)}}$.

We obtain a regression S-estimator when replacing $\hat{\sigma}(\boldsymbol{\beta})$ by an M-estimator of scale, applied to the residuals:

Regression S-estimators

Regression S-estimator (Rousseeuw and Yohai, 1984)

$$\hat{\boldsymbol{\beta}}_{S} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, \hat{\sigma}(\boldsymbol{\beta})$$

where $\hat{\sigma}(\boldsymbol{\beta})$ is given by

$$\frac{1}{n}\sum_{i=1}^{n} \rho\left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}(\boldsymbol{\beta})}\right) = \delta$$

with ρ a smooth bounded ρ-function.

The corresponding scale estimate $\hat{\sigma}_S$ satisfies

$$\frac{1}{n}\sum_{i=1}^{n} \rho\left(\frac{r_i(\hat{\boldsymbol{\beta}}_S)}{\hat{\sigma}_S}\right) = \delta \ .$$

Robustness and efficiency of regression S-estimators

- The influence function of S-estimators is unbounded.
- The breakdown value is again given by $\varepsilon^*(\hat{\boldsymbol{\beta}}_S) = \min\left(\frac{\delta}{\rho(\infty)}, 1 - \frac{\delta}{\rho(\infty)}\right)$ and can be as high as 50%.
- S-estimators satisfy the first-order conditions of M-estimators and thus are asymptotically normal.
- The efficiency of an S-estimator with 50% breakdown value cannot be higher than 33%.
- The efficiency of a bisquare S-estimator with 50% breakdown value is 29%.
- S-estimators can be computed with the FAST-S algorithm.

Regression MM-estimators

Regression MM-estimators

MM-estimators combine high breakdown value with high efficiency.

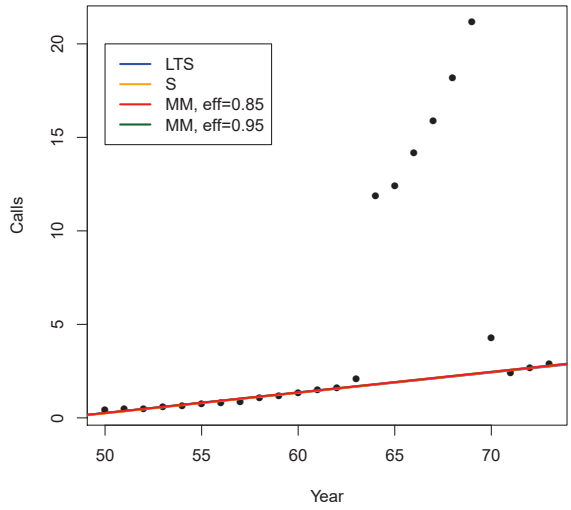
Regression MM-estimators (Yohai, 1987)

1. Compute an initial regression S-estimator $\hat{\boldsymbol{\beta}}_S$ with high breakdown value, and its corresponding scale estimate $\hat{\sigma}_S$.
2. Compute a regression M-estimator with fixed scale $\hat{\sigma}_S$ and initial estimate $\hat{\boldsymbol{\beta}}_S$ but now using a bounded ρ-function with high efficiency. This yields $\hat{\boldsymbol{\beta}}_{MM}$ and $\hat{\sigma}_{MM} = \hat{\sigma}_S$.

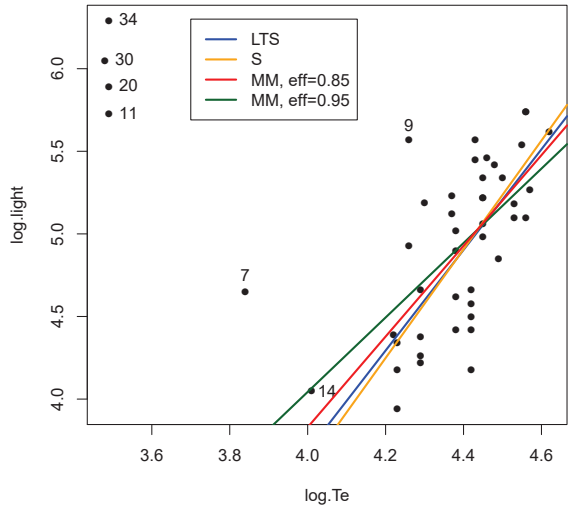
Robustness and efficiency of regression MM-estimators

- The influence function of MM-estimators is unbounded.
- The breakdown value of $\hat{\boldsymbol{\beta}}_{MM}$ is that of $\hat{\boldsymbol{\beta}}_S$.
- If $\hat{\boldsymbol{\beta}}_{MM}$ satisfies the M-equation (1), then it has the same asymptotic efficiency as the global minimum. It is thus not necessary to find the absolute minimum to ensure a high breakdown value and a high efficiency.
- The higher the efficiency, the higher the bias under contamination! An efficiency of 0.85 is often recommended. This corresponds with $c = 3.44$ for the bisquare ρ.
- Inference can be based on the asymptotic normality, in particular the estimated asymptotic covariance matrix.

Example: telephone data



Example: stars data



Example

```
> library(robustbase)
> stars.mm = lmrob(log.light ~ log.Te, data=starsCYG)
> summary(stars.mm)

Weighted Residuals:
     Min       1Q   Median       3Q      Max
-0.80959 -0.28838  0.00282  0.36668  3.39585

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -4.9694     3.4100  -1.457  0.15198
log.Te        2.2532     0.7691   2.930  0.00531 **
---
Signif. codes: *** 0.001   ** 0.01   * 0.05

Robust residual standard error: 0.4715
```


Example

```
Robustness weights:
 4 observations c(11,20,30,34) are outliers with |weight| = 0 ( < 0.0021);
 4 weights are ~= 1. The remaining 39 weights are summarized as
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.6533  0.9171  0.9593  0.9318  0.9848  0.9986

> stars.mm$weights
 [1] 0.9496 0.9239 0.9632 0.9239 0.9112 0.9416 0.6533 0.9986 0.6700 0.9759
[11] 0.0000 0.9229 0.9646 0.9999 0.9234 0.9324 0.8479 0.7494 0.9412 0.0000
[21] 0.9593 0.9090 0.8714 0.9641 0.9941 0.9560 0.9994 1.0000 0.9909 0.0000
[31] 0.9080 0.9828 0.9892 0.0000 0.9800 0.9868 0.9923 0.9892 0.9986 0.8765
[41] 0.9682 1.0000 0.9882 0.9675 0.9730 0.9976 0.9042

> stars.mm$init.S
$coef
[1] -9.570830  3.290361

$scale
[1] 0.4714579
```

Regression with categorical predictors

Regression with categorical predictors

When some of the predictors are categorical, the data are no longer in general position and consequently the estimators have a lower breakdown value.

Consider e.g. the situation with 1 continuous x_1 and 1 binary covariate x_2. This corresponds to fitting two parallel lines:

$$\begin{aligned} y_i &= \beta_0 + \beta_1 x_{i1} + \varepsilon_i && \text{if } x_2 = 0 \\ y_i &= \beta_0 + \beta_1 x_{i1} + \beta_2 + \varepsilon_i && \text{if } x_2 = 1 \end{aligned}$$

The model has three parameters. But if we take 3 data points they could all satisfy $x_2 = 0$, so they would not determine a unique fit $\hat{\boldsymbol{\beta}}$.

Moreover, the algorithms based on resampling $p + 1$ observations (LTS, S) will yield many singular subsets.

Finally, the robust distances based on MCD (or any other robust covariance matrix) are only appropriate when the majority of the data are roughly elliptical.

Model

The linear model with m categorical covariables is:

- response : continuous
  errors : i.i.d., common scale σ .
- explanatory variables : p continuous, m categorical ($c_1, \ldots, c_m$ levels).
- a single categorical variable ($m = 1$) would yield parallel hyperplanes.
- if $m > 1$ and $p = 0$ we get an m-way table.
- encode the categorical variables by binary dummies:
  set $q = \sum_{k=1}^{m} (c_k - 1)$ and write the model as

$$y_i = \theta_0 + \sum_{j=1}^{p} \theta_j x_{ij} + \sum_{l=1}^{q} \gamma_l I_{il} + e_i$$

Objective: find robust estimates of $(\theta_j)_{j=0,\ldots,p}$ and $(\gamma_l)_{l=1,\ldots,q}$.

The RDL^1 method

The RDL^1 method (Hubert and Rousseeuw, 1997) takes the following steps:

1. Detect outliers in the x-space of the continuous regressors by computing their reweighted MCD estimator and the robust distances $\mathrm{RD}(\boldsymbol{x}_i)$.
2. Assign to each observation the weight

$$w_i = \min\left\{1, \frac{p}{\mathrm{RD}^2(\boldsymbol{x}_i)}\right\} .$$

   Note that in the gaussian case $E[\mathrm{RD}^2(\boldsymbol{x}_i)] \approx E[\chi^2_p] = p$.
3. Compute the weighted L^1 estimator:

$$(\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\gamma}}) = \underset{(\boldsymbol{\theta},\boldsymbol{\gamma})}{\mathrm{argmin}} \sum_{i=1}^{n} w_i |r_i(\boldsymbol{\theta}, \boldsymbol{\gamma})| .$$

4. Compute the error scale estimate $\hat{\sigma} = 1.4826 * \mathrm{med}_i\, |r_i|$.

Example 1: Employment growth

response variable: rate of employment growth

continuous regressors:

- % of people engaged in production activities
- % of people engaged in higher services
- growths of these percentages

categorical regressors:

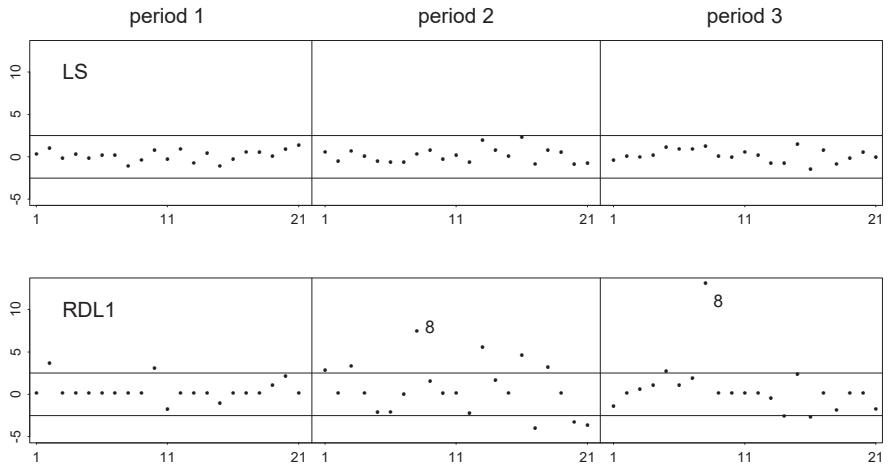
- region (21 regions around Hannover)
- time period (1979-1982, 1983-1988, 1989-1992)

$$\Downarrow$$

$$n = 21 * 3 = 63$$
$$p = 4, m = 2, q = 20 + 2 = 22$$

Example 1: Employment growth

Standardized residuals of classical and robust fits:



Example 2: Education expenditure data

Source: Chatterjee and Price (1991)

response variable: per capita expenditure on education in US states

continuous regressors:

- per capita income
- number of residents per thousand under 18 years of age
- number of residents per thousand living in urban areas

categorical regressors:

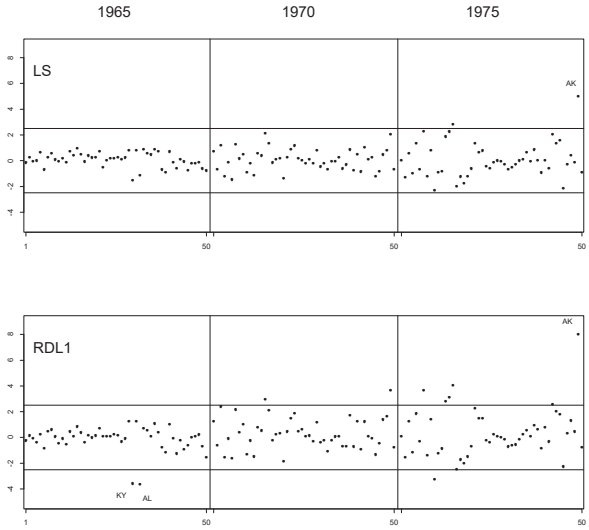
- region (NE, NC, S, W)
- time period (1965, 1970, 1975)

$$\Downarrow$$

$$n = 50 * 3 = 150$$
$$p = 3, m = 2, q = 3 + 2 = 5$$

Example 2: Education expenditure data

Standardized residuals of classical and robust fits:



Software

Software for robust regression: R

- FAST-LTS: the function `ltsReg` in the package *robustbase*. Default: $\alpha = 0.5$, yielding a breakdown value of 50%.
- FAST-S, MM: the function `lmrob` in package *robustbase* performs MM-estimation.

  Default settings:
  - bisquare S-estimator with 50% breakdown value as initial estimator. The results of the S-estimation can be retrieved from the `init.S` component of the output.
  - uses the FAST-S algorithm
  - computes bisquare M-estimator with 95% asymptotic efficiency.
  - robust x-distances are not computed.

  To change the default settings, use `lmrob.control`.

  Example:

```
stars.mm=lmrob(log.light~log.Te, data=starsCYG,
          control=lmrob.control(bb=0.25,compute.rd=TRUE))
```

Software for robust regression: R

- The function `ltsreg` in the MASS package uses an older (slower) implementation of the LTS estimator.
- the `lmRob` function within the package *robust* also performs MM-estimation.
  - when the predictors contain categorical variables it performs a robust procedure based on iterating S-estimation on the continuous predictors, and M-estimation on the categorical predictors.

Software for robust regression: Matlab and SAS

In Matlab:

- FAST-LTS: the function `ltsregres` in the toolbox LIBRA (wis.kuleuven.be/stat/robust), and the PLS toolbox of Eigenvector Research (www.eigenvector.com).
  Default: $\alpha = 0.75$, yielding a breakdown value of 25%.
- FAST-S and MM: in the FSDA toolbox of Riani, Perrotta and Torti.

In SAS:

- PROC ROBUSTREG performs LTS, M, S and MM-estimation.
- Robust distances are computed with the FAST-MCD algorithm.
- Default initial estimator: LTS with 25% breakdown value.
- the output gives leverage points (includes good and bad leverage points) and outliers (includes vertical outliers and bad leverage points).