# Robust covariance estimation for financial applications

**Tim Verdonck**, Mia Hubert, Peter Rousseeuw

*Department of Mathematics*
*K.U.Leuven*

August 30 2011

# Contents

# Introduction Robust Statistics

Real data often contain outliers.
Most classical methods are highly influenced by these outliers.

## What is robust statistics?

Robust statistical methods try to fit the model imposed by the **majority** of the data. They aim to find a 'robust' fit, which is similar to the fit we would have found without outliers (observations deviating from robust fit). This also allows for **outlier detection**.

Robust estimate applied on all observations is comparable with the classical estimate applied on the outlier-free data set.

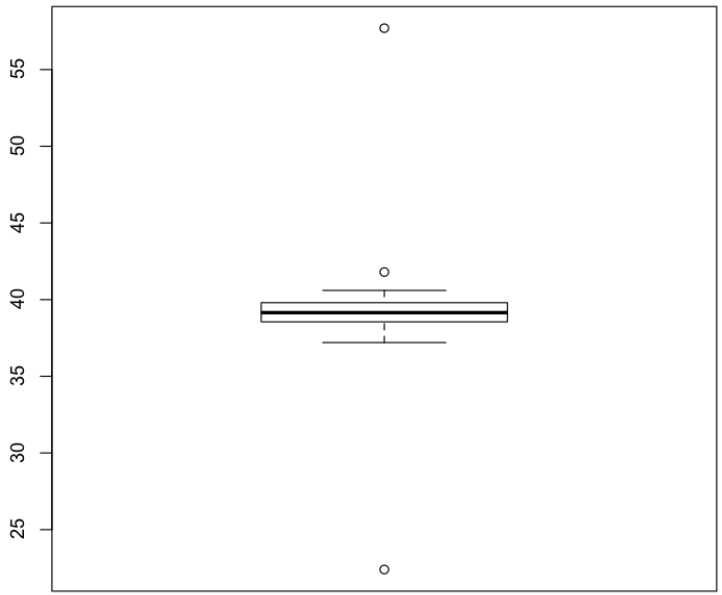
## Robust estimator

A good robust estimator combines high robustness with high efficiency.

- Robustness: being less influenced by outliers.
- Efficiency: being precise at uncontaminated data.

# Univariate Scale Estimation: Wages data set

6000 households with male head earning less than USD 15000 annually in 1966. Classified into 39 demographic groups (we concentrate on variable AGE).




- Standard Deviation (SD): $\sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x})^2} = 4.91$
- Interquartile Range (IQR): $0.74(x_{(\lfloor 0.75n \rfloor)} - x_{(\lfloor 0.25n \rfloor)}) = 0.91$
- Median Absolute Deviation (MAD): $1.48\,\mathrm{med}_i\,|x_i - \mathrm{med}_j\,x_j| = 0.96$

# Measures of robustness

## Breakdown Point

The breakdown point of a scale estimator $S$ is the smallest fraction of observations to be contaminated such that $S \uparrow \infty$ or $S \downarrow 0$.

| Scale estimator | Breakdown point |
|---|---|
| SD | $\frac{1}{n} \approx 0$ |
| IQR | 25% |
| MAD | 50% |

Note that when the breakdown value of an estimator is $\varepsilon$, this does not imply that a proportion of less than $\varepsilon$ does not affect the estimator at all.

# Measures of robustness

A specific type of contamination is point contamination

$$F_{\varepsilon,y} = (1-\varepsilon)F + \varepsilon\Delta_y$$

with $\Delta_y$ Dirac measure at $y$.

## Influence Function (Hampel, 1986)

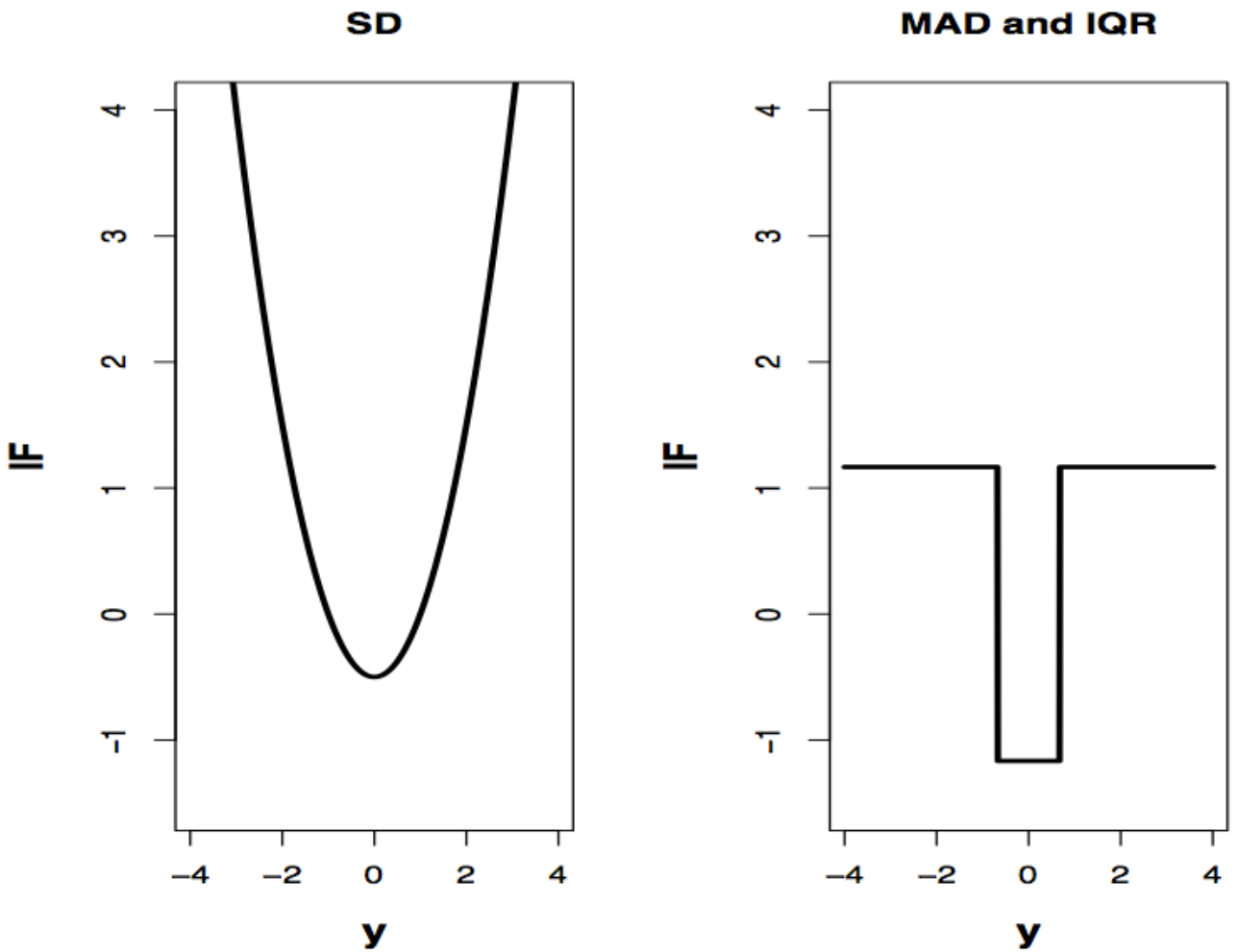
The influence function measures how $T(F)$ changes when contamination is added in $y$

$$IF(y; T, F) = \lim_{\varepsilon \to 0} \frac{T(F_{\varepsilon,y}) - T(F)}{\varepsilon}$$

where $T(.)$ is functional version of the estimator.

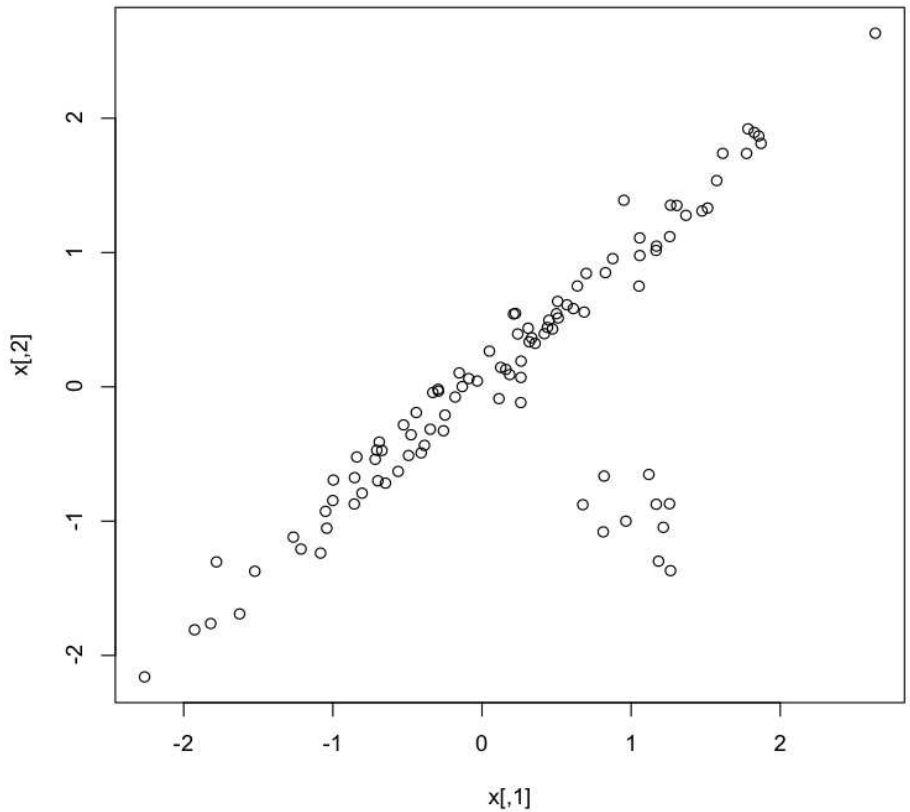
- IF is a local measure of robustness, whereas breakdown point is a global measure.
- We prefer estimators that have a bounded IF.

# Influence Function (Hampel, 1986)



# Multivariate Location and Scatter

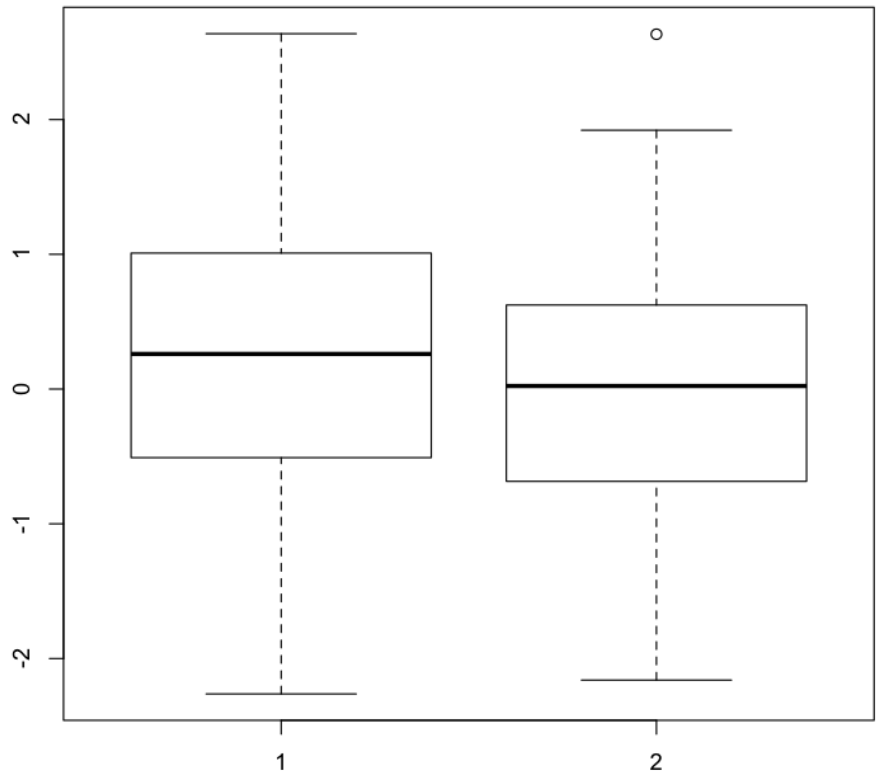
Scatterplot of bivariate data ($\rho = 0.990$)




- $\hat{\rho} = 0.779$
- $\hat{\rho}_{\mathrm{MCD}} = 0.987$.

# Boxplot of the marginals

In the multivariate setting, outliers can not just be detected by applying outlier detection rules on each variable separately.




Only by correctly estimating the covariance structure, we can detect the outliers.

# Classical Estimator

Data: $X_n = \mathbf{x}_1, \ldots, \mathbf{x}_n$ with $\mathbf{x}_i \in \mathbb{R}^p$.
Model: $X_i \sim N_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$.
More general we can assume that the data are generated from an elliptical distribution, i.e. a distribution whose density contours are ellipses.

The classical estimators for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are the empirical mean and covariance matrix

$$\begin{aligned} \overline{\mathbf{x}} &= \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i \\ \mathbf{S}_n &= \frac{1}{n-1}\sum_{i=1}^{n}(\mathbf{x}_i - \overline{\mathbf{x}})(\mathbf{x}_i - \overline{\mathbf{x}})'. \end{aligned}$$

Both are highly sensitive to outliers

- zero breakdown value
- unbounded IF.

# Tolerance Ellipsoid

Boundary contains **x**-values with constant **Mahalanobis distance** to mean.

$$MD_i = \sqrt{(\mathbf{x}_i - \overline{\mathbf{x}})' S_n^{-1} (\mathbf{x}_i - \overline{\mathbf{x}})}$$

## Classical Tolerance Ellipsoid

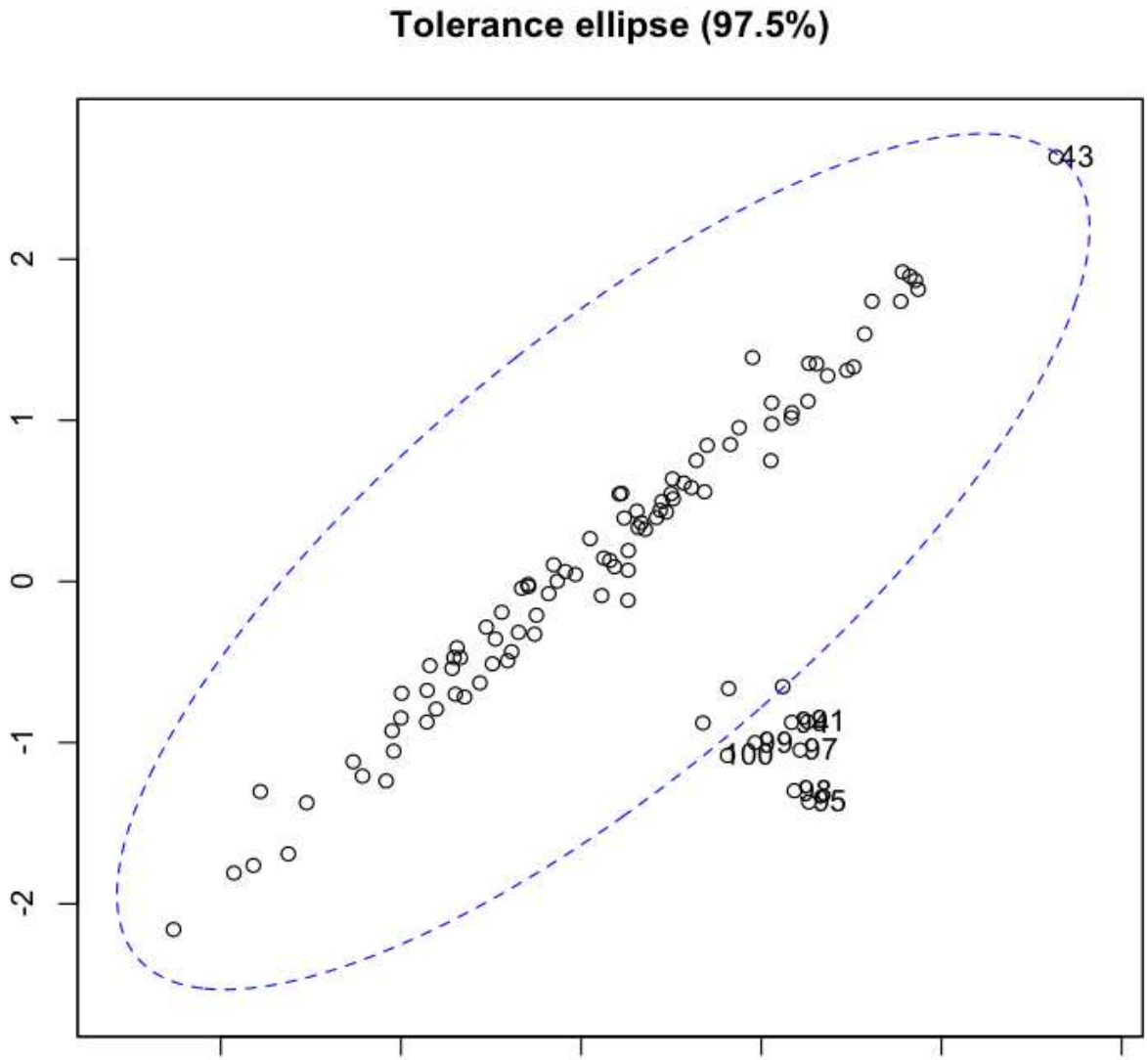
$$\{\mathbf{x} | MD(\mathbf{x}) \leq \sqrt{\chi^2_{p,0.975}}\}$$

with $\chi^2_{p,0.975}$ the 97.5% quantile of the $\chi^2$ distribution with $p$ d.f.

We expect (at large samples) that 97.5% of the observations belong to this ellipsoid. We can flag observation $\mathbf{x}_i$ as an outlier if it does not belong to the tolerance ellipsoid.

# Tolerance Ellipsoid

Tolerance Ellipsoid for example



# Robust Estimator

## Minimum Covariance Determinant Estimator (MCD)

- Estimator of multivariate location and scatter [Rousseeuw, 1984].
- Raw MCD estimator:
  - Choose $h$ between $\lfloor (n+p+1)/2 \rfloor$ and $n$.
  - Find $h < n$ observations whose classical covariance matrix has lowest determinant.

$$H_0 = \underset{H}{\operatorname{argmin}} \det\left(\operatorname{cov}(\mathbf{x}_i | i \in H)\right)$$

  - $\hat{\mu}_0$ is mean of those $h$ observations.

$$\hat{\mu}_0 = \frac{1}{n} \sum_{i \in H_0} \mathbf{x}_i.$$

  - $\hat{\Sigma}_0$ is covariance matrix of those $h$ observations (multiplied by consistency factor).

$$\hat{\Sigma}_0 = c_0 \operatorname{cov}(\mathbf{x}_i | i \in H_0)$$

# Robust Estimator

## Minimum Covariance Determinant Estimator (MCD)

- Estimator of multivariate location and scatter [Rousseeuw, 1984].
- Raw MCD estimator.
- Reweighted MCD estimator:
  - Compute initial **robust distances**

$$d_i = D(\mathbf{x}_i, \hat{\mu}_0, \hat{\Sigma}_0) = \sqrt{(x_i - \hat{\mu}_0)' \hat{\Sigma}_0^{-1} (x_i - \hat{\mu}_0)}.$$

  - Assign weights $w_i = 0$ if $d_i > \sqrt{\chi^2_{p,0.975}}$, else $w_i = 1$.
  - Compute reweighted mean and covariance matrix:

$$\begin{aligned} \hat{\boldsymbol{\mu}}_{\mathrm{MCD}} &= \frac{\sum_{i=1}^n w_i \mathbf{x}_i}{\sum_{i=1}^n w_i} \\ \hat{\boldsymbol{\Sigma}}_{\mathrm{MCD}} &= c_1 \left( \sum_{i=1}^n w_i (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_{\mathrm{MCD}})(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_{\mathrm{MCD}})' \right) \left( \sum_{i=1}^n w_i \right)^{-1}. \end{aligned}$$

  - Compute final robust distances and assign new weights $w_i$.

# Outlier detection

For outlier detection, recompute the **robust distances** (based on MCD).

$$RD_i = \sqrt{(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_{MCD})' \hat{\boldsymbol{\Sigma}}_{MCD}^{-1} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_{MCD})}$$

Flag observation $\mathbf{x}_i$ as outlier if $RD_i > \sqrt{\chi^2_{p,0.975}}$.

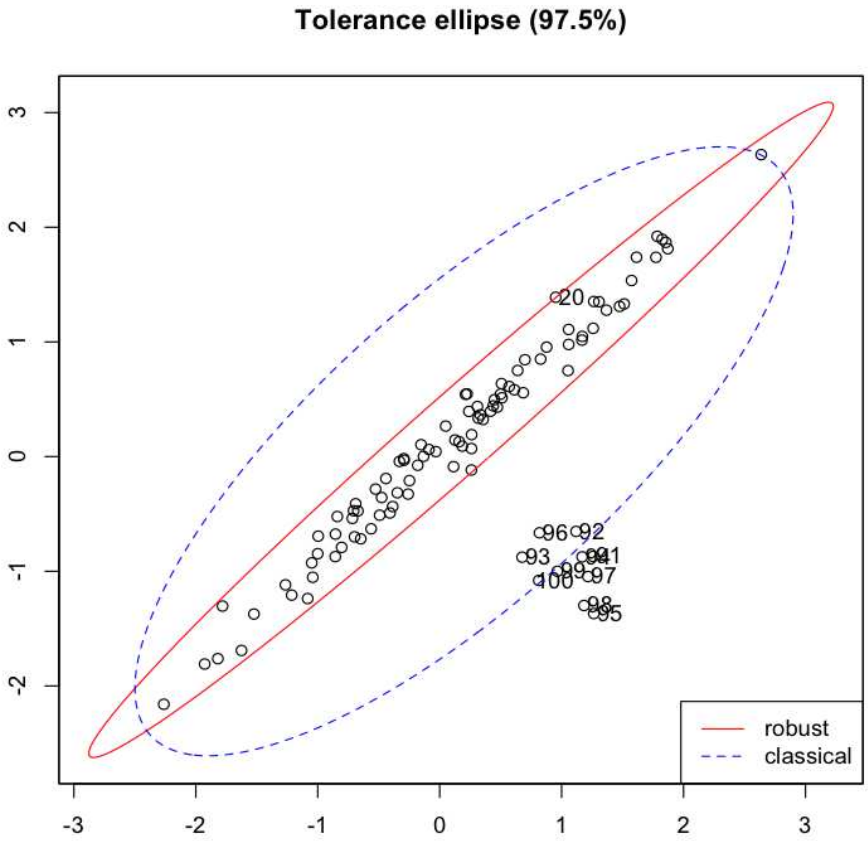
This is equivalent with flagging the observations that do not belong to the robust tolerance ellipsoid.

**Robust tolerance ellipsoid**

$$\{\mathbf{x} | RD(\mathbf{x}) \leq \sqrt{\chi^2_{p,0.975}}\}$$

# Outlier detection

Robust Tolerance Ellipsoid (based on MCD) for example



# Properties of the MCD

- Robust
  - breakdown point from 0 to 50%
  - bounded influence function [Croux and Haesbroeck, 1999] .
- Positive definite
- Affine equivariant
  - given $\mathbf{X}$, the MCD estimates satisfy

$$\begin{aligned} \hat{\mu}(\mathbf{XA} + \mathbf{1}_n\mathbf{v}') &= \hat{\mu}(\mathbf{X})\mathbf{A} + \mathbf{v} \\ \hat{\mathbf{\Sigma}}(\mathbf{XA} + \mathbf{1}_n\mathbf{v}') &= \mathbf{A}'\hat{\mathbf{\Sigma}}(\mathbf{X})\mathbf{A}. \end{aligned}$$

    for all nonsingular matrices $\mathbf{A}$ and all constant vectors $\mathbf{v}$.

  $\Rightarrow$ data may be rotated, translated or rescaled without affecting the outlier detection diagnostics.
- Not very efficient: improved by reweighting step.
- Computation: FAST-MCD algorithm [Rousseeuw and Van Driessen, 1999].

# FAST-MCD algorithm

Computation of the raw estimates for $n \leqslant 600$:

- For $m = 1$ to 500:
  - Draw random subsets of size $p + 1$.
  - Apply two C-steps:
    - Compute robust distances

$$d_i = D(\mathbf{x}_i, \hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}) = \sqrt{(\mathbf{x}_i - \hat{\boldsymbol{\mu}})' \hat{\boldsymbol{\Sigma}}^{-1} (\mathbf{x}_i - \hat{\boldsymbol{\mu}})}.$$

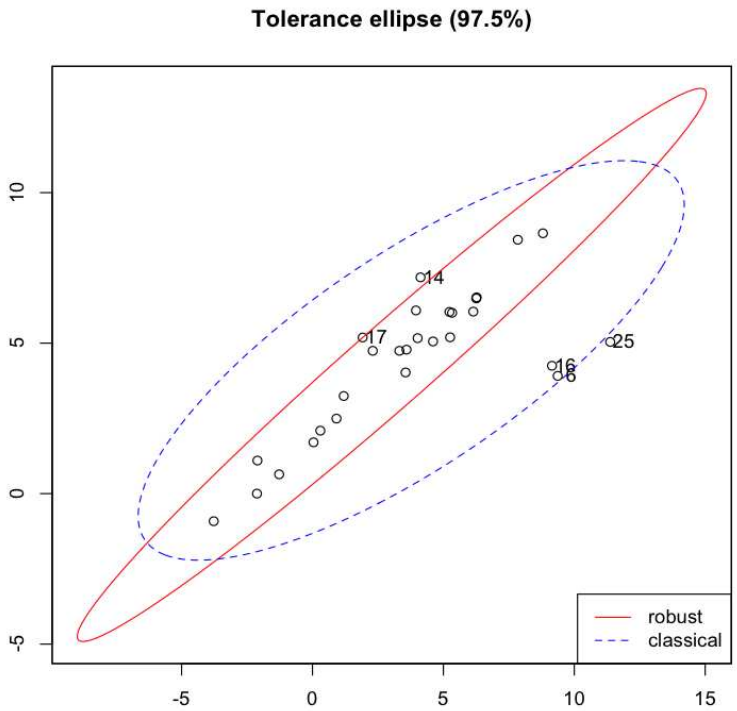
    - Take $h$ observations with smallest robust distance.
    - Compute mean and covariance matrix of this $h$-subset.
- Retain 10 $h$-subsets with lowest covariance determinant.
- Apply C-steps on these 10 subsets until convergence.
- Retain the $h$-subset with lowest covariance determinant.

# FASTMCD algorithm

- A C-step will always decrease the determinant of the covariance matrix.
- As there are only a finite number of $h$-subsets, convergence to a (local) minimum is guaranteed.
- The algorithm is not guaranteed to yield the global minimum. The fixed number of initial $p+1$-subsets (500) is a compromise between robustness and computation time.
- Implementations of FASTMCD algorithm widely available.
  - R: in the packages robustbase and rrcov
  - Matlab: in LIBRA toolbox and PLS toolbox of Eigenvector Research.
  - SAS: in PROC ROBUSTREG
  - S-plus: built-in function *cov.mcd*.

# Example: Animal set

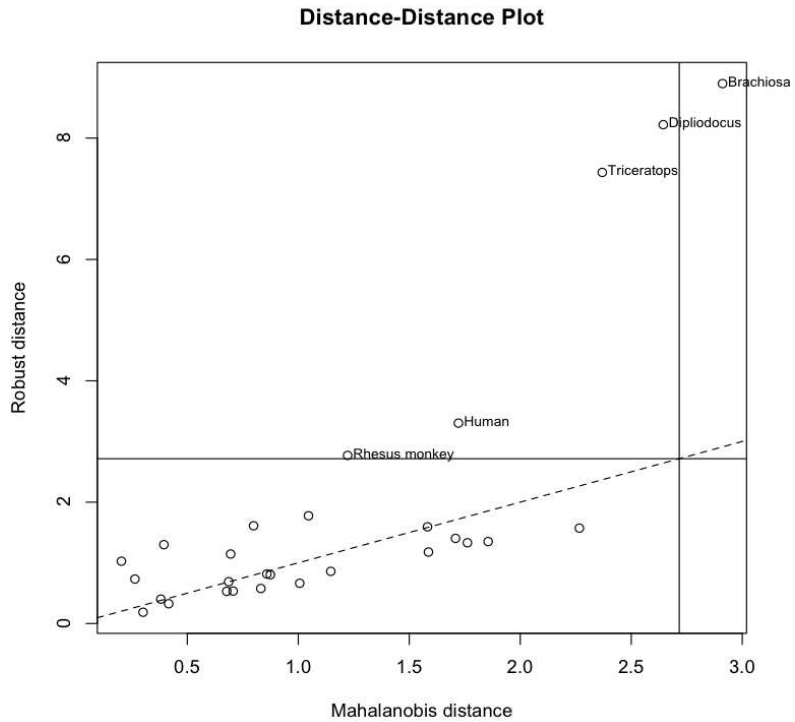
Logarithm of body and brain weight for 28 animals.




Outlier detection based on MCD correctly indicates the outliers.

# Example: Animal set

In dimension $p > 2$, a scatterplot and tolerance ellipsoid can not be drawn. To expose the differences between a classical and a robust analysis, a distance-distance plot can be made




Outlier detection based on MCD correctly indicates the outliers.

# DetMCD algorithm

Deterministic algorithm for MCD [Hubert, Rousseeuw and Verdonck, 2010].

- Idea:
  - Compute several 'robust' $h$-subsets, based on
    - robust transformations of variables
    - robust estimators of multivariate location and scatter.
  - Apply C-steps until convergence.

# Computation of DetMCD

- Standardize $\mathbf{X}$ by subtracting median and dividing by $Q_n$.
  - Location and scale equivariant.
  - Standardized data: $\mathbf{Z}$ with rows $\mathbf{z}_i'$ and columns $Z_j$.
- Obtain estimate $\mathbf{S}$ for covariance/correlation matrix of $\mathbf{Z}$.
- To overcome lack of positive definiteness:
  1. Compute eigenvectors $\mathbf{P}$ of $\mathbf{S}$ and define $\mathbf{B} = \mathbf{ZP}$.
  2. $\hat{\mathbf{\Sigma}}(\mathbf{Z}) = \mathbf{PLP}'$ with $\mathbf{L} = \operatorname{diag}\left(Q_n(B_1)^2, \ldots, Q_n(B_p)^2\right)$.
- Estimation of the center: $\hat{\boldsymbol{\mu}}(\mathbf{Z}) = \left(\operatorname{med}(\mathbf{Z}\hat{\mathbf{\Sigma}}^{-\frac{1}{2}})\right)\hat{\mathbf{\Sigma}}^{\frac{1}{2}}$.
- Compute statistical distances

$$d_i = D(\mathbf{z}_i, \hat{\boldsymbol{\mu}}(\mathbf{Z}), \hat{\mathbf{\Sigma}}(\mathbf{Z})).$$

- Initial $h$-subset: $h$ observations with smallest distance.
- Apply C-steps until convergence.

# Construct preliminary estimates **S**

1. Take hyperbolic tangent of the standardized data.

$$Y_j = \tanh(Z_j) \qquad \forall j = 1, \ldots, p.$$

Take Pearson correlation matrix of $\mathbf{Y}$

$$\mathbf{S}_1 = \text{corr}(\mathbf{Y}).$$

2. Spearman correlation matrix.

$$\mathbf{S}_2 = \text{corr}(\mathbf{R})$$

where $R_j$ is the rank of $Z_j$.

3. Compute Tukey normal scores $T_j$ from the ranks $R_j$:

$$T_j = \Phi^{-1}\left(\frac{R_j - \frac{1}{3}}{n + \frac{1}{3}}\right)$$

where $\Phi(.)$ is normal cdf

$$\mathbf{S}_3 = \text{corr}(\mathbf{T}).$$

# Construct preliminary estimates **S**

4. Related to spatial sign covariance matrix [Visuri et al., 2000] .
   Define $\mathbf{k}_i = \frac{\mathbf{z}_i}{\|\mathbf{z}_i\|}$ and let

$$\mathbf{S}_4 = \frac{1}{n}\sum_{i=1}^{n}\mathbf{k}_i\mathbf{k}_i'$$

5. We take first step of BACON algorithm [Billor et al., 2000 ] .
   Consider $\lceil n/2 \rceil$ standardized observations $\mathbf{z}_i$ with smallest norm, and compute their mean and covariance matrix.
6. Obtained from the raw OGK estimator for scatter.
   [Maronna and Zamar, 2002 ]

# Simulation study

Compare DetMCD with FASTMCD on artificial data.

- Different small and moderate data sets
  - $n = 100$ and $p = 2$
  - $n = 100$ and $p = 5$
  - $n = 200$ and $p = 10$
  - $n = 400$ and $p = 40$
  - $n = 600$ and $p = 60$.
- Also consider correlated data [Maronna and Zamar, 2002 ] .
- Different contamination models
  - $\varepsilon = 0, 10, 20, 30$ and $40\%$.
- Different types of contamination
  - point, cluster and radial contamination.

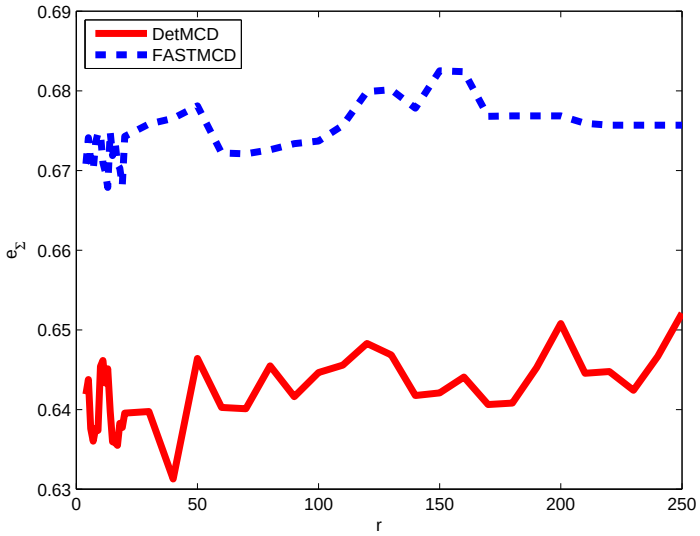
# Simulation study

Compare DetMCD with FASTMCD on artificial data.

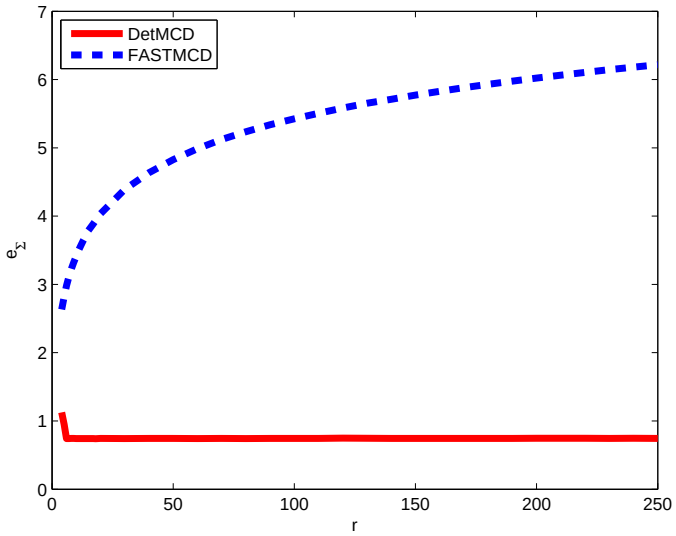
- Measures of performance
  - The objective function of the raw scatter estimator, $\mathrm{OBJ} = \det \hat{\mathbf{\Sigma}}_{\mathrm{raw}}(\mathbf{Y})$.
  - An error measure of the location estimator, given by $e_\mu = ||\hat{\boldsymbol{\mu}}(\mathbf{Y})||^2$.
  - An error measure of the scatter estimate, defined as the logarithm of its condition number: $e_\Sigma = \log_{10}(\mathrm{cond}(\hat{\mathbf{\Sigma}}(\mathbf{Y})))$.
  - The computation time $t$ (in seconds).

  Each of these performance measures should be as close to 0 as possible.

# Simulation study




(a)




(b)

Figuur: The error of the scatter estimate for different values of $r$ when $n = 400$, $p = 40$ for (a) 10% and (b) 40% cluster contamination.

Value of $r$ determines distance between outliers and main center.

# Simulation results for clean data

| | A | | B | | C | | D | | E | |
|---|---|---|---|---|---|---|---|---|---|---|
| | DetMCD | OGK | DetMCD | OGK | DetMCD | OGK | DetMCD | OGK | DetMCD | OGK |
| OBJ | 0.088 | 0.086 | 0.031 | 0.030 | 0.009 | 0.009 | 1e-5 | 1e-5 | 4.35e-7 | 8.68e-7 |
| $e_\mu$ | 0.028 | 0.031 | 0.065 | 0.073 | 0.060 | 0.063 | 0.124 | 0.132 | 0.1250 | 0.1285 |
| $e_\Sigma$ | 0.175 | 0.202 | 0.390 | 0.460 | 0.393 | 0.418 | 0.636 | 0.668 | 0.6424 | 0.6576 |
| $t$ | 0.019 | 0.498 | 0.029 | 0.581 | 0.096 | 0.868 | 1.775 | 4.349 | 5.7487 | 8.7541 |

| | | Point (10%) | | Cluster (10%) | | Radial (10%) | |
|---|---|---|---|---|---|---|---|
| A | OBJ | 0.120 / 0.120 | 0.117 / 0.117 | 0.119 / 0.120 | 0.117 / 0.117 | 0.119 | 0.117 |
| | $e_\mu$ | 0.027 / 0.028 | 0.028 / 0.029 | 0.027 / 0.027 | 0.028 / 0.028 | 0.027 | 0.029 |
| | $e_\Sigma$ | 0.156 / 0.158 | 0.171 / 0.172 | 0.157 / 0.157 | 0.171 / 0.171 | 0.161 | 0.177 |
| | $t$ | 0.018 / 0.019 | 0.483 / 0.482 | 0.018 / 0.018 | 0.482 / 0.482 | 0.018 | 0.496 |
| B | OBJ | 0.047 / 0.047 | 0.045 / 0.045 | 0.047 / 0.047 | 0.045 / 0.045 | 0.047 | 0.045 |
| | $e_\mu$ | 0.068 / 0.068 | 0.074 / 0.074 | 0.068 / 0.068 | 0.074 / 0.074 | 0.067 | 0.074 |
| | $e_\Sigma$ | 0.383 / 0.383 | 0.425 / 0.425 | 0.382 / 0.383 | 0.426 / 0.426 | 0.379 | 0.425 |
| | $t$ | 0.028 / 0.028 | 0.556 / 0.555 | 0.028 / 0.028 | 0.557 / 0.557 | 0.028 | 0.579 |
| C | OBJ | 0.014 / 0.015 | 0.014 / 0.013 | 0.015 / 0.015 | 0.014 / 0.014 | 0.015 | 0.014 |
| | $e_\mu$ | 0.064 / 0.063 | 0.065 / 0.855 | 0.063 / 0.064 | 0.065 / 0.065 | 0.063 | 0.066 |
| | $e_\Sigma$ | 0.399 / 0.398 | 0.415 / 1.037 | 0.398 / 0.398 | 0.415 / 0.415 | 0.397 | 0.414 |
| | $t$ | 0.092 / 0.092 | 0.823 / 0.825 | 0.093 / 0.093 | 0.828 / 0.828 | 0.092 | 0.861 |
| D | OBJ | 3e-05 / 3e-05 | 5e-05 / 3e-05 | 4e-05 / 4e-05 | 5e-05 / 5e-05 | 4e-05 | 5e-05 |
| | $e_\mu$ | 0.131 / 0.130 | 0.135 / 175 | 0.131 / 0.130 | 0.135 / 0.135 | 0.129 | 0.136 |
| | $e_\Sigma$ | 0.651 / 0.650 | 0.672 / 4.639 | 0.651 / 0.651 | 0.672/ 0.673 | 0.645 | 0.670 |
| | $t$ | 1.694 / 1.710 | 4.395 / 4.305 | 1.715 / 1.717 | 4.362 / 4.344 | 1.739 | 4.336 |
| E | OBJ | 1e-06 / 2e-06 | 5e-10 / 6e-07 | 1e-06 / 1e-06 | 2e-06 / 2e-06 | 1e-06 | 2e-06 |
| | $e_\mu$ | 0.288 / 0.134 | 51.5 / 65317 | 0.134 / 0.134 | 0.134 / 0.134 | 0.135 | 0.136 |
| | $e_\Sigma$ | 0.666 / 0.661 | 3.098 / 6.201 | 0.660 / 0.660 | 0.663 / 0.663 | 0.660 | 0.669 |
| | $t$ | 5.527 / 5.527 | 8.530 / 8.769 | 5.649 / 5.644 | 8.773 / 8.758 | 5.703 | 8.617 |

| | | Point (40%) | | Cluster (40%) | | Radial (40%) | |
|---|---|---|---|---|---|---|---|
| A | OBJ | 0.018 / 0.436 | 0.010 / 0.165 | 0.436 / 0.436 | 0.433 / 0.433 | 0.435 | 0.433 |
| | $e_\mu$ | 13.79 / 0.033 | 15.24 / 272.0 | 0.033 / 0.033 | 0.033 / 0.033 | 0.095 | 0.091 |
| | $e_\Sigma$ | 2.615 / 0.144 | 2.870 / 4.102 | 0.144 / 0.144 | 0.144 / 0.144 | 0.352 | 0.361 |
| | $t$ | 0.019 / 0.017 | 0.483 / 0.483 | 0.017 / 0.017 | 0.482 / 0.482 | 0.016 | 0.495 |
| B | OBJ | 1e-04 / 0.313 | 3e-05 / 0.053 | 0.371 / 0.312 | 0.309 / 0.309 | 0.313 | 0.309 |
| | $e_\mu$ | 79.0 / 0.084 | 96.8 / 2e+05 | 1.206 / 0.084 | 0.134 / 0.085 | 0.086 | 0.086 |
| | $e_\Sigma$ | 3.46 / 0.391 | 4.58 / 7.84 | 0.465 / 0.391 | 0.395 / 0.392 | 0.398 | 0.400 |
| | $t$ | 0.027 / 0.027 | 0.550 / 0.553 | 0.030 / 0.027 | 0.553 / 0.554 | 0.027 | 0.577 |
| C | OBJ | 3e-04 / 0.168 | 4e-09 / 6e-06 | 0.168/ 0.168 | 110 / 1404 | 0.168 | 0.166 |
| | $e_\mu$ | 160 / 0.084 | 187 / 3+05 | 0.084 / 0.084 | 7111 / 90886 | 0.084 | 0.084 |
| | $e_\Sigma$ | 3.58 / 0.441 | 4.20 / 7.43 | 0.441 / 0.441 | 4.089 / 5.127 | 0.440 | 0.442 |
| | $t$ | 0.088 / 0.088 | 0.804 / 0.809 | 0.093 / 0.093 | 0.824 / 0.830 | 0.089 | 0.850 |
| D | OBJ | 5e-33 / 0.004 | 2e-32/ 1e-29 | 0.004 / 0.004 | 0.003/ 12.2 | 0.004 | 0.004 |
| | $e_\mu$ | 766 / 0.171 | 760 / 1e+06 | 15.7 / 0.171 | 99.76 / 4e+05 | 0.172 | 0.174 |
| | $e_\Sigma$ | 4.57 / 0.734 | 5.06 / 8.13 | 1.03 / 0.733 | 2.62 / 6.21 | 0.735 | 0.737 |
| | $t$ | 1.64 / 1.64 | 4.00 / 4.18 | 1.76 / 1.78 | 4.34 / 4.33 | 1.72 | 4.23 |
| E | OBJ | 5-49 / 5e-04 | 6e-49 / 8e-46 | 1e-04 / 4e-04 | 1e-04 / 0.819 | 4e-04 | 4e-04 |
| | $e_\mu$ | 1152 / 0.172 | 1142 / 2e+06 | 75.4 / 0.172 | 84.7 / 6e+05 | 0.171 | 0.171 |
| | $e_\Sigma$ | 4.72 / 0.744 | 4.88 / 8.14 | 2.43 / 0.742 | 2.53 / 6.37 | 0.739 | 0.740 |
| | $t$ | 5.33 / 5.32 | 7.13 / 7.39 | 5.91 / 5.77 | 8.70 / 8.76 | 5.59 | 8.43 |

# Properties of DetMCD

Advantages

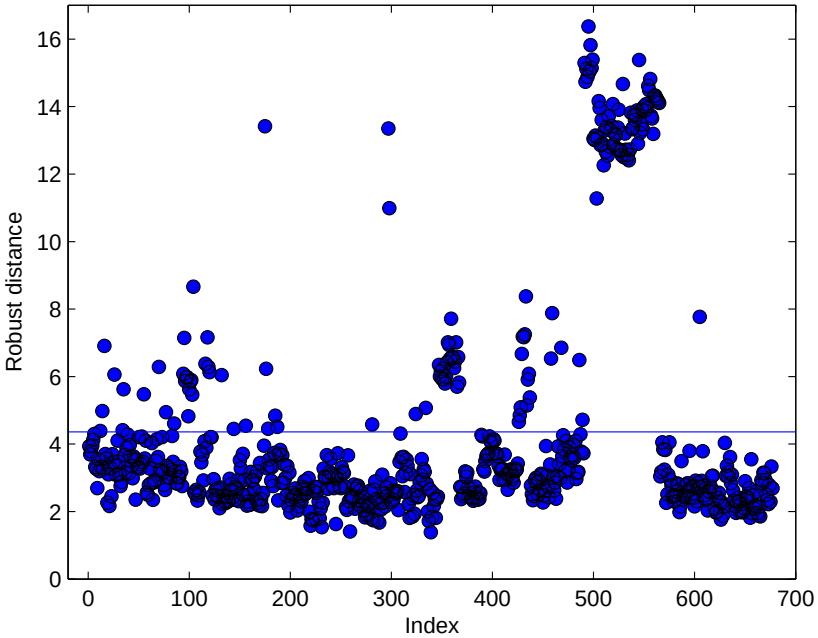
- Very fast
  - DetMCD: typically 3/4 C-steps needed to converge, hence 21 C-steps in total.
  - FASTMCD uses 1000 C-steps.
- Fully deterministic
- Permutation invariant
- Easy to compute DetMCD for different values of $h$
  - The initial subsets are independent of $h$.

Disadvantages

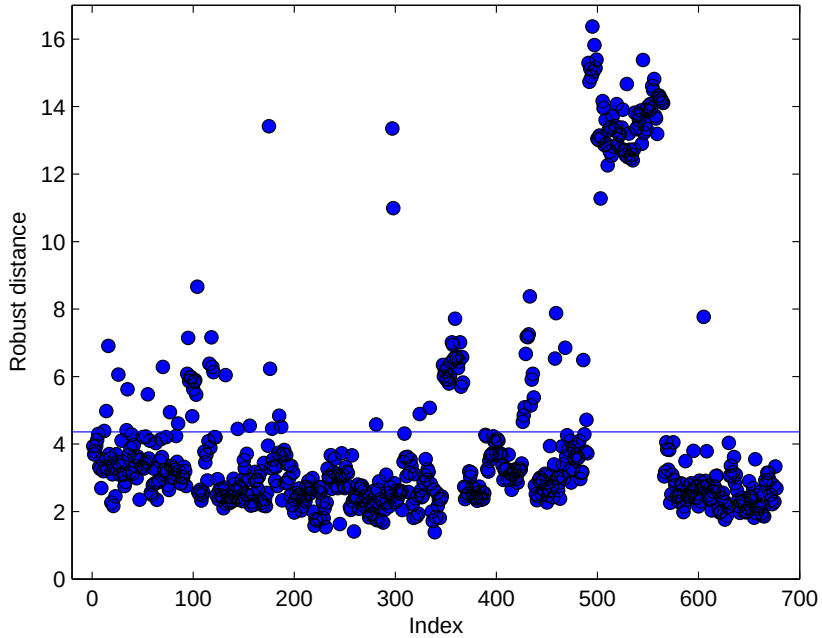
- Not fully affine equivariant

# Example: Philips data

Engineers measured 9 characteristics for 667 diaphragm parts for television sets.




(a)




(b)

Figuur: Robust distances of the Philips data with (a) DetMCD and (b) FASTMCD.

# Example: Philips data

Engineers measured 9 characteristics for 667 diaphragm parts for television sets.

- Estimates for location and scatter almost identical.
  - $d_\mu = ||\hat{\boldsymbol{\mu}}_{\text{MCD}} - \hat{\boldsymbol{\mu}}_{\text{DetMCD}}|| = 0.0000$
  - $d_\Sigma = \text{cond}\left(\hat{\boldsymbol{\Sigma}}_{\text{MCD}}^{-\frac{1}{2}} \hat{\boldsymbol{\Sigma}}_{\text{DetMCD}} (\hat{\boldsymbol{\Sigma}}_{\text{MCD}}^{-\frac{1}{2}})'\right) = 1.0000.$
- Objective functions almost the same
  - $\frac{\text{OBJ}_{\text{MCD}}}{\text{OBJ}_{\text{DetMCD}}} = 0.9992.$
- Optimal $h$-subsets only differed in 1 observation.
- Computation time
  - DetMCD: 0.2676s
  - FASTMCD: 1.0211s

# Applications of MCD

MCD has been applied in numerous research fields, such as

- Finance
- Medicine
- Quality control
- Image analysis
- Chemistry

MCD has also been used as a basis to develop robust and computationally efficient multivariate techniques, such as

- Principal Component Analysis (PCA)
- Classification
- Factor Analysis
- Multivariate Regression

# Principal Component Analysis (PCA)

PCA summarizes information in data into few principal components (PCs)

- Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ be the data ($n$ cases and $p$ variables).
- PCs $\mathbf{t}_i$ are defined as linear combinations of the data

$$\mathbf{t}_i = \mathbf{X}\mathbf{p}_i$$

- where

$$\mathbf{p}_i = \underset{\mathbf{a}}{\operatorname{argmax}} \{\operatorname{var}(\mathbf{X}\mathbf{a})\}$$

  under the constraint that

$$\|\mathbf{p}_i\| = 1 \quad \text{and} \quad \operatorname{cov}\left(\mathbf{X}\mathbf{p}_i, \mathbf{X}\mathbf{p}_j\right) = 0 \quad \text{for } j < i$$

- The PCs are uncorrelated and ordered so that the first few retain most of the variation present in all of the original variables.
- From Lagrange multiplier method: PCs can be computed as eigenvectors of the variance-covariance matrix $\mathbf{\Sigma}$.
- Variance and variance-covariance matrix are sensitive to outliers
  $\Rightarrow$ PCA is a non-robust method.

# PCA

- Example: *Chinchilla data*
  - 50 Chinchillas for auditory research
  - 3 measurements (cm): length tail, length whisker and length ear
  - data is standardized.

- Visualization.

# PCA

- Example: *Chinchilla data*
- Measurements for 10 more Chinchillas from USA
  $\rightarrow$ added to the data.
- Visualization.

# PCA

- Example: *Chinchilla data*
- Measurements for 10 more Chinchillas from USA
  $\rightarrow$ added to the data.
- Solution: Robust PCA when data contains outliers.
- Visualization.

# PCA

- Example: *Chinchilla data*
- Measurements for 10 more Chinchillas from USA
  $\rightarrow$ added to the data.
- Solution: Robust PCA when data contains outliers
- Reason for outliers $\rightarrow$ wrong Chinchillas from USA.

# Robust covariance matrix

- PCA corresponds to a spectral decomposition of the variance-covariance matrix as $\mathbf{\Sigma} = \mathbf{P\Lambda P}'$
  - $\mathbf{P}$ contains as columns the eigenvectors $\mathbf{p}_j$ of $\mathbf{\Sigma}$.
  - $\mathbf{\Lambda}$ is a diagonal matrix where the diagonal elements $\lambda_{jj}$ are the eigenvalues of $\mathbf{\Sigma}$ corresponding to $\mathbf{p}_j$.
- Simple idea: Compute principal components as eigenvectors of a robust covariance matrix (a robust estimate of $\mathbf{\Sigma}$).
- Robust scatter estimators can not be computed or have bad statistical properties in high dimensions.

# ROBPCA: a hybrid method (Hubert et al.)

- Use PP to find directions which are most outlying.
- Stahel-Donoho Outlyingness (SDO) is defined as

$$r(\mathbf{x}_i, \mathbf{X}) = \sup_{\mathbf{v} \in \mathbb{R}^p} \left\| \frac{\mathbf{v}'\mathbf{x}_i - M(\mathbf{v}'\mathbf{X})}{S(\mathbf{v}'\mathbf{X})} \right\|$$

- $\mathbf{x}_i$: rows of $\mathbf{X}$
- $M$: estimator of location (univariate MCD).
- $S$: estimator of scale (univariate MCD).
- $\mathbf{v}$: a $p$ variate direction.

# ROBPCA: a hybrid method (Hubert et al.)

- Use PP to find directions which are most outlying.
- Stahel-Donoho Outlyingness (SDO) is defined as

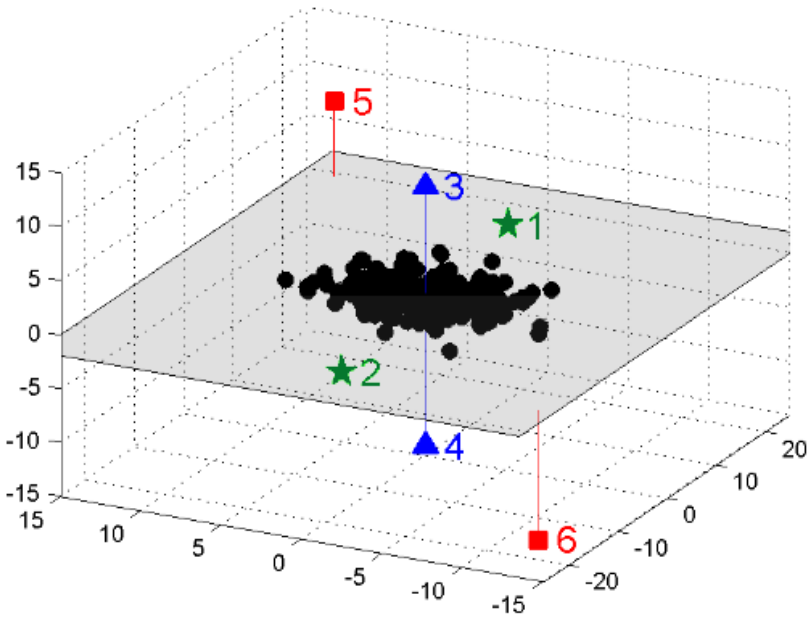
$$r(\mathbf{x}_i, \mathbf{X}) = \sup_{\mathbf{v} \in \mathbb{R}^p} \left\| \frac{\mathbf{v}'\mathbf{x}_i - M(\mathbf{v}'\mathbf{X})}{S(\mathbf{v}'\mathbf{X})} \right\|$$

- Apply classical PCA on $h$ data points with smallest $\mathrm{SDO}$ and retain $k$ components.
- Obtain improved robust subspace estimate as subspace spanned by $k$ dominant eigenvectors of covariance matrix of all points with $\mathrm{OD}_i = \|x_i - \hat{x}_i\| < c_h$ ($c_h$ cutoff value based on robust measure of location and scale of the $\mathrm{OD}_i^{2/3}$). Project data points on this subspace.
- Apply MCD covariance estimator in subspace: mean and covariance of the $h$ points with smallest $\mathrm{RD}_i$.
- Final PCs are eigenvectors of this robust covariance matrix.
- Robustness properties are inherited from MCD.

# Outliers

Outlier is observation that does not obey the pattern of the majority of the data.

- Different kind of outliers.




- 1,2: good leverage points
  3,4: orthogonal outliers
  5,6: bad leverage points

# Outlier Map

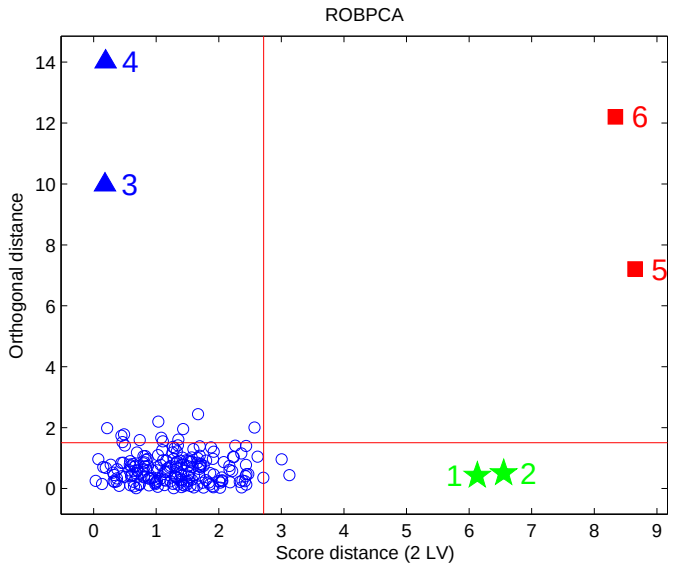
- Displays orthogonal distance $OD_{i,k}$ vs score distance $SD_{i,k}$

$$\begin{aligned} OD_{i,k} &= \|x_i - \hat{\mu}_x - P_{p,k} t_i\| = \|x_i - \hat{x}_i\| \\ SD_{i,k} &= \sqrt{t_i' (L_{k,k})^{-1} t_i} \end{aligned}$$

- Here $\hat{\mu}_x$, $P$, $t_i$ and $L$ represent resp. the robust center of the data, the robust loading matrix, the robust scores and a diagonal matrix with as entries the $k$ largest robust eigenvalues as a result of a robust PCA method.
- Cut-off value to determine outliers for each distance.

# Outlier Map

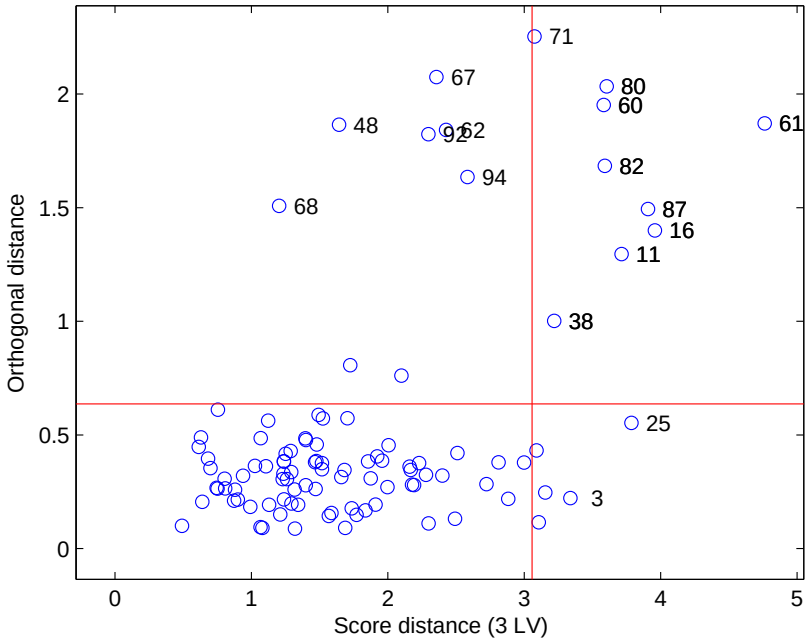
- Example: 3-dimensional data
  (generated from multivariate normal distribution)
  projected on a robust 2-dimensional PCA-subspace.

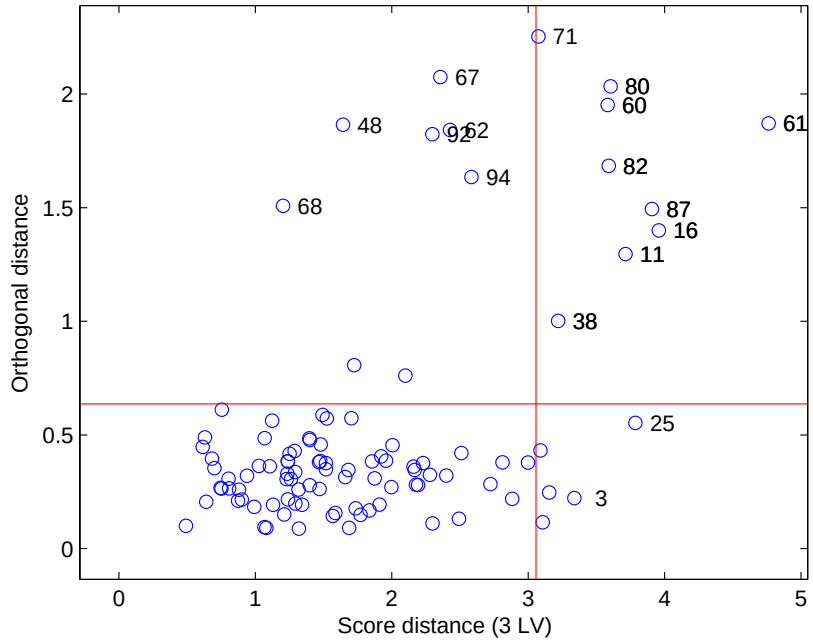



1. 1 & 2: Good leverage points (outlying SD, regular OD)
2. 3 & 4: Orhogonal outliers (outlying OD, regular SD)
3. 5 & 6: Bad leverage points (outlying OD, outlying SD)

# Example 1: Swiss bank notes ($n = 100$ and $p = 6$)

- Highly correlated variables and outliers ⇒ Robust PCA method.
- Missing values in the data ⇒ Methodology of Serneels and Verdonck (2008).
- 3 PCs explained 92% of the variance.




- Time: FASTMCD took 197s, whereas DetMCD needed 10s.

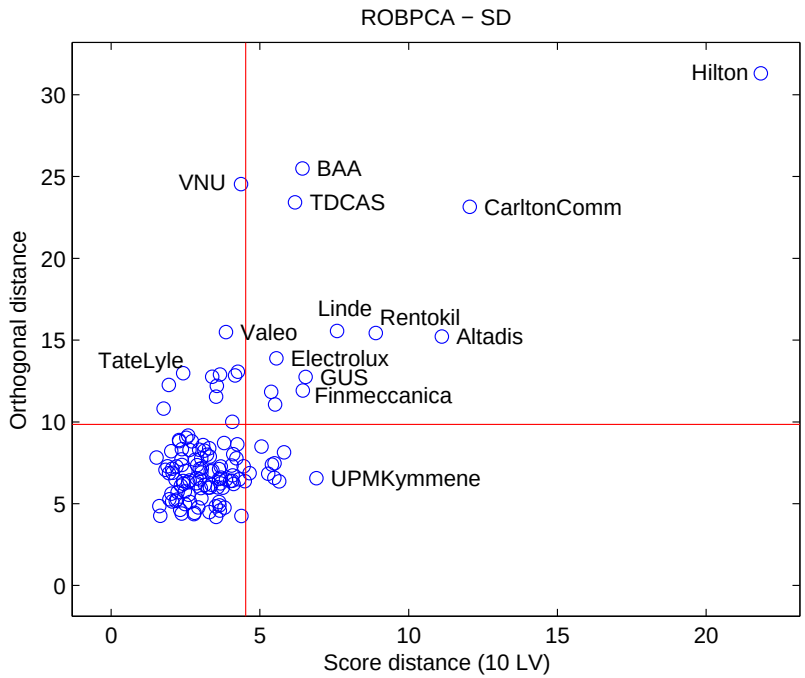
# Example 2: Credit Default Swap (CDS) from iTraxx Europe

Price of CDS of 125 companies over 58 weeks.

- take log ratios $\log(x_{i,j}/x_{i,j-1})$ for every $\mathbf{x}_i$ ($i$ is company and $j$ is week).
- Delete variables containing more than 63 zeroes.

$\Rightarrow$ 125 companies and 37 log ratios.

Applying ROBPCA



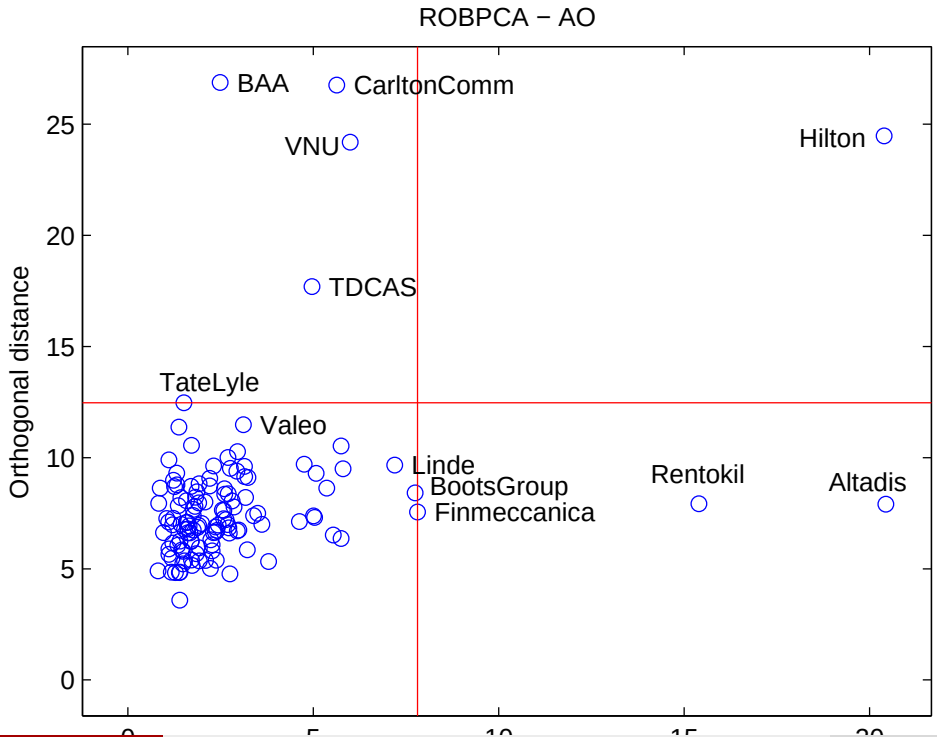
# Example 2: Credit Default Swap (CDS) from iTraxx Europe

Price of CDS of 125 companies over 58 weeks.

- take log ratios $\log(x_{i,j}/x_{i,j-1})$ for every $\mathbf{x}_i$ ($i$ is company and $j$ is week).
- Delete variables containing more than 63 zeroes.

$\Rightarrow$ 125 companies and 37 log ratios.

Cope with skewness of variables [Hubert et al., 2009] .



# Econometric application: Multivariate Time Series

Multivariate exponential smoothing is popular technique to forecast time series.

- Can not cope with outliers.

Robust version based on MCD [Croux et al., 2010] .

- Assume
  - $\mathbf{y}_1, \ldots, \mathbf{y}_T$: multivariate time series
  - $\hat{\mathbf{y}}_1, \ldots, \hat{\mathbf{y}}_{t-1}$: robust smoothed values of $\mathbf{y}_1, \ldots, \mathbf{y}_{t-1}$ are already computed.
- $\hat{\mathbf{y}}_t = \mathbf{\Lambda}\mathbf{y}_t^* + (\mathbf{I} - \mathbf{\Lambda})\hat{\mathbf{y}}_{t-1}$ where
  - $\mathbf{\Lambda}$ is smoothing matrix
  - $\mathbf{y}_t^*$ is cleaned version of $p$-dimensional vector $\mathbf{y}_t$.
- Forecast for $\mathbf{y}_{T+1}$ that can be made at time $T$

$$\hat{\mathbf{y}}_{T+1|T} = \hat{\mathbf{y}}_T = \mathbf{\Lambda} \sum_{k=0}^{T-1} (\mathbf{I} - \mathbf{\Lambda})^k \mathbf{y}_{T-k}.$$

# Econometric application: Multivariate Time Series

Multivariate exponential smoothing is popular technique to forecast time series.

- Can not cope with outliers.

Robust version based on MCD [Croux et al., 2010] .

- This multivariate cleaned series is calculated as

$$\mathbf{y}_t^* = \frac{\psi\left(\sqrt{\mathbf{r}'\hat{\mathbf{\Sigma}}_t^{-1}\mathbf{r}_t}\right)}{\sqrt{\mathbf{r}'\hat{\mathbf{\Sigma}}_t^{-1}\mathbf{r}_t}}\mathbf{r}_t + \hat{\mathbf{y}}_{t|t-1}$$

where

- $\mathbf{r}_t = \mathbf{y}_t - \hat{\mathbf{y}}_{t|t-1}$ denotes one-step-ahead forecast error
- $\psi$ is Huber $\psi$-function with clipping constant $\sqrt{\chi^2_{p,0.95}}$
- $\hat{\mathbf{\Sigma}}_t$ is estimated covariance matrix of one-step-ahead forecast error at time $t$

# Econometric application: Multivariate Time Series

Multivariate exponential smoothing is popular technique to forecast time series.

- Can not cope with outliers.

Robust version based on MCD [Croux et al., 2010] .

$\Rightarrow$ Robust version uses MCD in 2 different stages of algorithm

- starting values are obtained by MCD-based robust multivariate regression [Rousseeuw et al., 2010] .
- MCD is used as loss function to choose smoothing matrix $\mathbf{\Lambda}$ (iterative manner).

# Illustration: Housing Data (1968-1996)

Real bivariate time series of monthly data (housing starts and completions).

- Startup period of length 10 and complete series as training sample yields following smoothing matrix

$$\mathbf{\Lambda} = \begin{pmatrix} 0.68 & 0.04 \\ 0.04 & 0.62 \end{pmatrix}.$$

- Redoing this example with DetMCD gives exact same smoothing matrix.
- Time: FASTMCD took 1 hour and 52 minutes, whereas DetMCD only needed 22 minutes.
- Speed-up will become more important when considering higher-dimensional time series.

# Conclusions

- DetMCD is new algorithm which
  - is typically more robust than FASTMCD and needs even less time.
  - is deterministic in that it does not use any random subsets.
  - is permutation invariant and close to affine equivariant
  - allows to run the analysis for many values of $h$ without much additional computation.
- We illustrated DetMCD in contexts of PCA and time series analysis.
- Also many other methods that (in)directly rely on MCD may benefit from DetMCD approach, such as
  - robust canonical correlation
  - robust regression with continuous and categorical regressors
  - robust errors-in-variables regression
  - robust calibration
  - on-line applications or procedures that require MCD to be computed many times.

# Selected references



M. Hubert, P.J. Rousseeuw and T. Verdonck (2011).
A deterministic algorithm for the MCD.
Submitted.


M. Hubert, P.J. Rousseeuw and T. Verdonck (2009).
Robust PCA for skewed data and its outlier map.
*Computational Statistics and Data Analysis*, 53: 2264–2274.


S. Serneels and T. Verdonck (2008).
Principal component analysis for data containing outliers and missing elements.
*Computational Statistics and Data Analysis*, 52: 1712–1727.


P. J. Rousseeuw and K. Van Driessen (1999).
A Fast Algorithm for the Minimum Covariance Determinant Estimator.
*Technometrics*, 4:212–223. *Journal of the American Statistical Association*, 94(446): 434–445.


M. Hubert, P.J. Rousseeuw and K. Vanden Branden (2005).
ROBPCA: a new approach to robust principal component analysis.
*Technometrics*, 47: 64–79.


C. Croux, S. Gelper, K. Mahieu (2010).
Robust exponential smoothing of multivariate time series.
*Computational Statistics and Data Analysis*, 54: 2999-3006.