

Robustness under missing data: a comparison with special attention to inference

Carole Baum^{1*}, Holger Cevallos-Valdiviezo², Arnout Van Messem¹

¹University of Liège, Belgium

²Escuela Superior Politécnica del Litoral (ESPOL), Ecuador

COMPSTAT 2023 - August 22-25, 2023

Problem statement

Goal: Evaluate different robust regression methods in inference when missing data are present

Procedure: Simulations

Framework: Let $X \in \mathbb{R}^{n \times p}$ be the design matrix containing n observations and p predictor variables. Let $\beta \in \mathbb{R}^p$ be the vector of regression coefficients.

The dependent variable $y \in \mathbb{R}^n$ is defined according to the linear model as

$$y = X\beta + \varepsilon$$

where ε is a vector from \mathbb{R}^n with entries independent and identically distributed with $\mathbb{E}[\varepsilon] = 0$ and $\text{Var}[\varepsilon] = \sigma^2$.

Comparison of robust linear regressions in inference

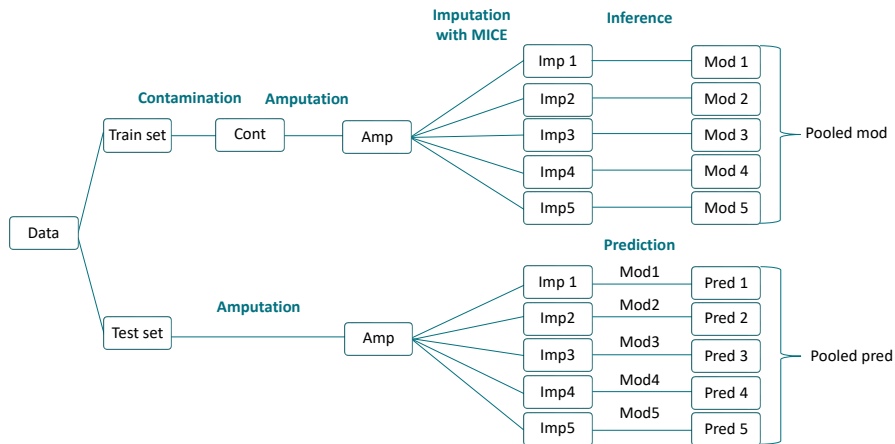
Comparison criteria:

- Bias of the regression coefficients
- Standard error of the regression coefficients
- Mean Squared Prediction Error

Simulation procedure

- 1 *Generate* the starting train and test sets
- 2 *Contaminate* the train set
- 3 *Ampute* the train set and the test set by deleting a selection of values
- 4 *Impute* the *train set* using MICE to obtain 5 complete imputed train sets
- 5 *Impute* the *test set* using the same models as created in the previous step to obtain 5 complete imputed test sets
- 6 *Fit a linear model* on the 5 train sets to get 5 models and pool the results → inference performance
- 7 *Predict* the *response* variable for the 5 test sets with the 5 fitted models from the previous step and pool the results → prediction performance

Overview of the simulation procedure



Generation of the train and test sets

The simulation setup follows the one proposed by Öllerer, Alfons and Croux [10]:

- $p = 15$ variables; $n = 300$ observations for the train set and $n = 100$ for the test set
- Regression coefficients: $\beta_j = j/p$ for $j = 1, \dots, p$
- Correlation:
 - Independent case: $\Sigma = I_p$ the covariance matrix, $\sigma^2 = 0.5^2$ the error variance
 - Dependent case: $\Sigma_{ij} = 0.5^{|i-j|}$, $\sigma^2 = 0.81^2$
- Generate X according to $\mathcal{N}_p(0, \Sigma)$
- Generate ε according to $\mathcal{N}(0, \sigma^2)$
- Define $y_i = x_i' \beta + \varepsilon_i$ for $i = 1, \dots, n$

Contamination of the train set

● Cellwise contamination

- Dense cluster: $x_{ij}^{cont} \sim \mathcal{N}(50, 1)$
- Dispersed outliers: $x_{ij}^{cont} \sim \mathcal{N}(0, 100^2)$
- Wide cluster: $x_{ij}^{cont} \sim \mathcal{N}(50, 10^2)$

● Rowwise contamination

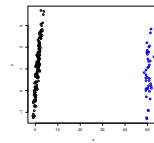
- Dense cluster: $x_i^{cont} \sim \mathcal{N}_p(50, \Sigma)$
- Dispersed outliers: $x_i^{cont} \sim \mathcal{N}_p(0, 100^2 \Sigma)$
- Wide cluster: $x_i^{cont} \sim \mathcal{N}_p(50, 10^2 \Sigma)$

● Response contamination

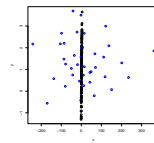
- Vertical outliers: $\varepsilon_i^{cont} \sim \mathcal{N}(50, \sigma^2)$

→ Percentage of contamination :
0%, 5% or 10%.

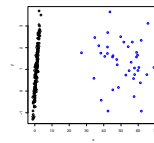
Dense cluster



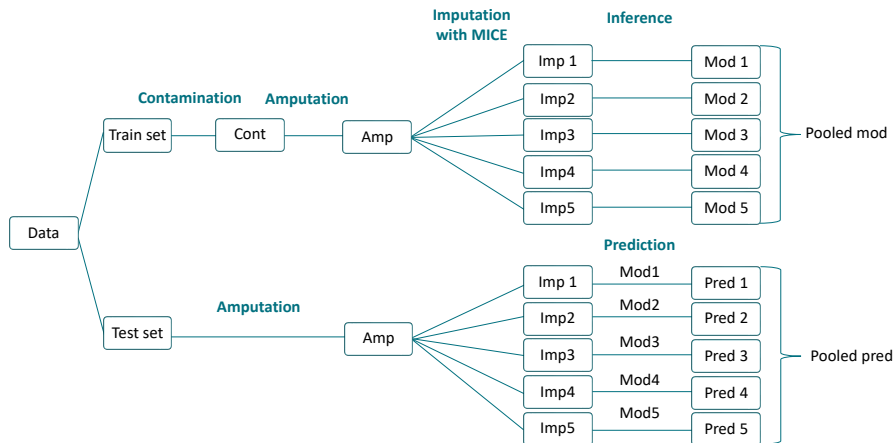
Dispersed outliers



Wide cluster



Overview of the simulation procedure



- **S-estimators** [6]

$$\hat{\beta}_S = \arg \min_{\beta \in \mathbb{R}^p} s(r_1(\beta), \dots, r_n(\beta))$$

where $s(r_1, \dots, r_n)$ is a solution of $\frac{1}{n} \sum_{i=1}^n \rho(r_i/s) = \delta$, $\delta = \mathbb{E}_\phi[\rho]$ and (r_1, \dots, r_n) are the residuals.

- **MM-estimators** [9]

$\hat{\beta}_{MM}$ is any solution of

$$\sum_{i=1}^n \rho' \left(\frac{r_i(\beta)}{s(r_1(\beta_S), \dots, r_n(\beta_S))} \right) x_i = 0$$

which verifies $S(\beta_{MM}) \leq S(\beta_S)$ where $S(\beta) = \sum_{i=1}^n \rho \left(\frac{r_i(\beta)}{s(r_1(\beta_S), \dots, r_n(\beta_S))} \right)$

- **Least trimmed squares** [4, 5]

For a fixed h with $\lceil \frac{n+p+2}{2} \rceil \leq h \leq n$,

$$\hat{\beta}_{LTS} = \operatorname{argmin}_{\hat{\beta}} \sum_{i=1}^h (r^2)_{i:n}$$

where $(r^2)_{1:n} \leq \dots \leq (r^2)_{n:n}$ are the ordered squared residuals

• Shooting-S [10]

Combine the coordinate descent algorithm with regression S-estimation to obtain the following objective function, $\forall j \in \{1, \dots, p\}$:

$$\hat{\beta}_j = \arg \min_{\beta \in \mathbb{R}} \hat{\sigma}_j(\beta)$$

$$\text{with } \frac{1}{n} \sum_{i=1}^n \rho \left(\frac{\tilde{y}_i^{(j)} - x_{ij}\beta}{\hat{\sigma}_j(\beta)} \right) = \delta$$

$$\text{and } w_{ik} = w \left(\frac{|\tilde{y}_i^{(j)} - x_{ik}\hat{\beta}_k|}{\hat{\sigma}_k(\hat{\beta}_k)} \right)$$

where

$$\tilde{y}_i^{(j)} = y_i - \sum_{k \neq j} (w_{ik}x_{ik} + (1 - w_{ik})\hat{x}_{ik})\hat{\beta}_k$$

$$\text{and } \hat{x}_{ik} = \frac{\tilde{y}_i^{(k)}}{\hat{\beta}_k}$$

• Cellwise robust M regression (CRM) [2]

- 1 Center and scale the data
- 2 Use a starting robust regression estimator (MM) to flag observations as casewise outliers if their absolute standard residuals exceed $z_{0.95}$. Apply SPADIMO to separate cellwise and casewise outliers. Cellwise outliers are deleted and then imputed
- 3 Apply an IRLS procedure to improve efficiency of the estimates

- S-estimators
- MM-estimators
- Least trimmed squares
- Shooting-S estimator
- Cellwise robust M regression
- Linear regression (classical)

But How to compute the standard deviation of the regression parameters ?

- For LTS and the classical regression : given in the output
- Other methods: use bootstrap samples \Rightarrow Fast and robust bootstrap [7]
 - Already computed and implemented in R for S and MM-estimators
 - Need to be adapted for CRM and Shooting-S

Fast and robust bootstrap (FRB) [7]

Let $\hat{\theta}_n \in \mathbb{R}^p$ be the robust parameter estimates of interest and Z_n be a sample. FRB can be used if $\hat{\theta}_n$ can be represented as a solution of fixed-point equations :

$$\hat{\theta}_n = g_n(\hat{\theta}_n)$$

where g_n generally depends on the sample Z_n .

Given a bootstrap sample Z_n^* , the recalculated estimates $\hat{\theta}_n^*$ then solves

$$\hat{\theta}_n^* = g_n^*(\hat{\theta}_n^*).$$

Instead of computing $\hat{\theta}_n^*$, we can compute the approximation

$$\hat{\theta}_n^{1*} := g_n^*(\hat{\theta}_n).$$

But $\hat{\theta}_n^{1*}$ underestimate the variability of $\hat{\theta}_n$. To remedy this, they define a linearly corrected version of $\hat{\theta}_n^{1*}$:

$$\hat{\theta}_n^{R*} := \hat{\theta}_n + [I + \nabla g_n(\hat{\theta}_n)]^{-1}(\hat{\theta}_n^{1*} - \hat{\theta}_n)$$

FRB applied to shooting-S

Note $\tilde{y}_i^{(j)} = y_i - \sum_{k \neq j} (w_{ik} x_{ik} + (1 - w_{ik}) \hat{x}_{ik}) \hat{\beta}_k$, the objective function of shooting-S, $\forall j \in \{1, \dots, p\}$, can be written as

$$\frac{1}{n} \sum_{i=1}^n \rho \left(\frac{\tilde{y}_i^{(j)} - x_{ij} \hat{\beta}_j}{\hat{\sigma}_j(\hat{\beta}_j)} \right) = \delta \quad (1)$$

$$\frac{1}{n} \sum_{i=1}^n \rho' \left(\frac{\tilde{y}_i^{(j)} - x_{ij} \hat{\beta}_j}{\hat{\sigma}_j(\hat{\beta}_j)} \right) x_{ij} = 0 \quad (2)$$

Equations (1) and (2) can be rewritten as

$$\hat{\beta}_j = \left(\sum_{i=1}^n z_{ij} x_{ij}^2 \right)^{-1} \sum_{i=1}^n z_{ij} x_{ij} \tilde{y}_i^{(j)} \quad (3)$$

$$\hat{\sigma}_j = \sum_{i=1}^n v_{ij} (\tilde{y}_i^{(j)} - x_{ij} \hat{\beta}_j) \quad (4)$$

where $z_{ik} = \frac{\rho'(r_{ik}/\hat{\sigma}_k)}{r_{ik}}$, $v_{ik} = \frac{\hat{\sigma}_k}{n\delta} \frac{\rho(r_{ik}/\hat{\sigma}_k)}{r_{ik}}$ and $r_{ik} = \tilde{y}_i^{(j)} - x_{ij} \hat{\beta}_j$.

Cellwise robust M regression:

- 1 Center and scale the data
- 2 Use a starting robust regression estimator (MM) to flag observations as casewise outliers if their absolute standard residuals exceed $z_{0.95}$.
Apply SPADIMO to separate cellwise and casewise outliers. Cellwise outliers are deleted and then imputed
- 3 Apply an IRLS procedure to improve efficiency of the estimates

Output of the method:

- $\hat{\beta}$: The regression coefficients
- \tilde{X} : The imputed design matrix
- Ω : A diagonal matrix containing the weights on the diagonal

Bootstrap procedure:

- 1 Compute the clean and weighted sample : $y_{\omega} = \Omega y$ and $X_{\omega} = \Omega \tilde{X}$
- 2 Draw bootstrap samples $(X_{\omega}^*, y_{\omega}^*)$ from (X_{ω}, y_{ω})
- 3 Bootstrap regression estimates are obtained by fitting a linear regression on that clean bootstrap sample:

$$y_{\omega}^* = X_{\omega}^* \hat{\beta}^*$$

Preliminary results

Simulation settings:

- Independent case
- Percentage of missing values: 10%
- Imputation method: MM-regression

Prediction error:

	Cellwise - Dense cluster			Rowwise - Dense cluster			Response		
	0%	5%	10%	0%	5%	10%	0%	50%	10%
Lin. Reg	0.67	31.48	95.40	0.67	8.41	1.97	0.66	12.53	35.44
MM-Reg	10.74	1013	2763	10.69	7431	15856	10.78	10.81	10.71
S-Reg	0.63	26.83	74.55	0.64	7.70	2.06	0.62	0.6	0.58
LTS	10.52	1025	2779	10.81	7404	15825	10.72	10.84	10.82
Shooting-S	0.62	21.82	77.27	0.62	7.96	1.89	0.62	0.61	0.63
CRM	0.65	25.83	75.67	0.63	7.75	2.20	0.59	0.62	0.62

Preliminary results

Simulation settings:

- Independent case
- Percentage of missing values: 10%
- Imputation method: MM-regression

Prediction error:

	Cellwise - Dense cluster			Rowwise - Dense cluster			Response		
	0%	5%	10%	0%	5%	10%	0%	50%	10%
Lin. Reg	0.67	31.48	95.40	0.67	8.41	1.97	0.66	12.53	35.44
MM-Reg	10.74	1013	2763	10.69	7431	15856	10.78	10.81	10.71
S-Reg	0.63	26.83	74.55	0.64	7.70	2.06	0.62	0.6	0.58
LTS	10.52	1025	2779	10.81	7404	15825	10.72	10.84	10.82
Shooting-S	0.62	21.82	77.27	0.62	7.96	1.89	0.62	0.61	0.63
CRM	0.65	25.83	75.67	0.63	7.75	2.20	0.59	0.62	0.62

Preliminary results

Simulation settings:

- Independent case
- Percentage of missing values: 10%
- Imputation method: MM-regression

Prediction error:

	Cellwise - Dense cluster			Rowwise - Dense cluster			Response		
	0%	5%	10%	0%	5%	10%	0%	50%	10%
Lin. Reg	0.67	31.48	95.40	0.67	8.41	1.97	0.66	12.53	35.44
MM-Reg	10.74	1013	2763	10.69	7431	15856	10.78	10.81	10.71
S-Reg	0.63	26.83	74.55	0.64	7.70	2.06	0.62	0.6	0.58
LTS	10.52	1025	2779	10.81	7404	15825	10.72	10.84	10.82
Shooting-S	0.62	21.82	77.27	0.62	7.96	1.89	0.62	0.61	0.63
CRM	0.65	25.83	75.67	0.63	7.75	2.20	0.59	0.62	0.62

Preliminary results

Simulation settings:

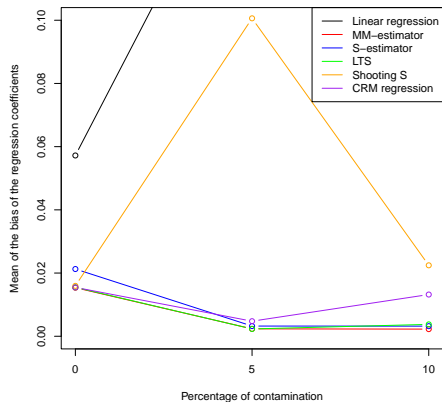
- Independent case
- Percentage of missing values: 10%
- Imputation method: MM-regression

Prediction error:

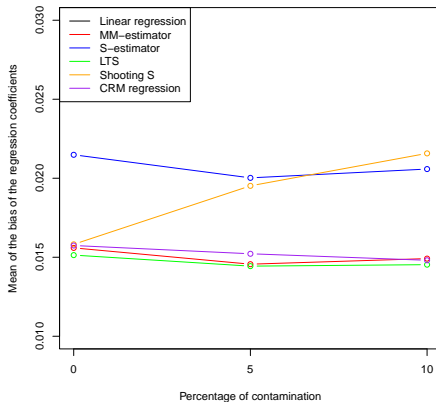
	Cellwise - Dense cluster			Rowwise - Dense cluster			Response		
	0%	5%	10%	0%	5%	10%	0%	50%	10%
Lin. Reg	0.67	31.48	95.40	0.67	8.41	1.97	0.66	12.53	35.44
MM-Reg	10.74	1013	2763	10.69	7431	15856	10.78	10.81	10.71
S-Reg	0.63	26.83	74.55	0.64	7.70	2.06	0.62	0.6	0.58
LTS	10.52	1025	2779	10.81	7404	15825	10.72	10.84	10.82
Shooting-S	0.62	21.82	77.27	0.62	7.96	1.89	0.62	0.61	0.63
CRM	0.65	25.83	75.67	0.63	7.75	2.20	0.59	0.62	0.62

Preliminary results

Cellwise contamination – Dense cluster



Rowwise contamination – Dense cluster



References

- [1] Michiel Debruyne, Sebastiaan Höppner, Sven Serneels, and Tim Verdonck. Outlyingness: Which variables contribute most? *Statistics and Computing*, 29(4):707–723, 2019.
- [2] Peter Filzmoser, Sebastiaan Höppner, Irene Ortner, Sven Serneels, and Tim Verdonck. Cellwise robust M regression. *Computational Statistics & Data Analysis*, 147:106944, 2020.
- [3] Roderick J. A. Little and Donald B. Rubin. *Statistical analysis with missing data*. Wiley series in probability and statistics. John Wiley, New York, NY, 2nd edition edition, 2002.
- [4] Peter Rousseeuw. Least Median of Squares Regression. *Journal of the American Statistical Association*, 79(388):871–880, 1984.
- [5] Peter Rousseeuw and Annick Leroy. *Robust Regression and Outlier Detection*. Wiley Series in Probability and Statistics. Wiley, 1987.

- [6] Peter Rousseeuw and Victor Yohai. Robust Regression by Means of S-Estimators. In Jürgen Franke, Wolfgang Härdle, and Douglas Martin, editors, *Robust and Nonlinear Time Series Analysis*, pages 256–272, New York, NY, 1984. Springer US.
- [7] Matías Salibián-Barrera, Stefan Van Aelst, and Gert Willems. Fast and robust bootstrap. *Statistical Methods and Applications*, 17:41–71, 2008.
- [8] Rianne Margaretha Schouten, Peter Lugtig, and Gerko Vink. Generating missing values for simulation purposes: a multivariate amputation procedure. *Journal of Statistical Computation and Simulation*, 88(15):2909–2930, 2018.
- [9] Victor Yohai. High Breakdown-Point and High Efficiency Robust Estimates for Regression. *The Annals of statistics*, 15(2):642–656, 1987.
- [10] Viktoria Öllerer, Aneas Alfons, and Christophe Croux. The shooting S-estimator for robust regression. *Computational statistics*, 31(3):829–844, 2016.