Least Trimmed Squares Regression with Missing Values and Cellwise Outliers

Jakob Raymaekers

U of Antwerp, Belgium

Peter Rousseeuw

KU Leuven, Belgium

What are outliers?

An **outlier** is a piece of data that does not follow the pattern suggested by the majority of the data set.

Outliers may be (a) **undesirable** errors which can hurt our data analysis, or (b) **valuable** nuggets of unexpected information!

We want to identify them. This has many names:

- outlier detection (Statistics,...)
- exception mining (Data Mining,...)
- anomaly detection (Machine Learning, Data Science,...)
- ...

Robust statistics aims to fit the majority of the data first, and then to detect the outliers by their deviation (distance, residual,...) from that fit.

Types of outliers

We want to analyze a data matrix $\boldsymbol{X}$ with n rows (cases) and $p > 1$ columns (variables). But the data may contain outliers.

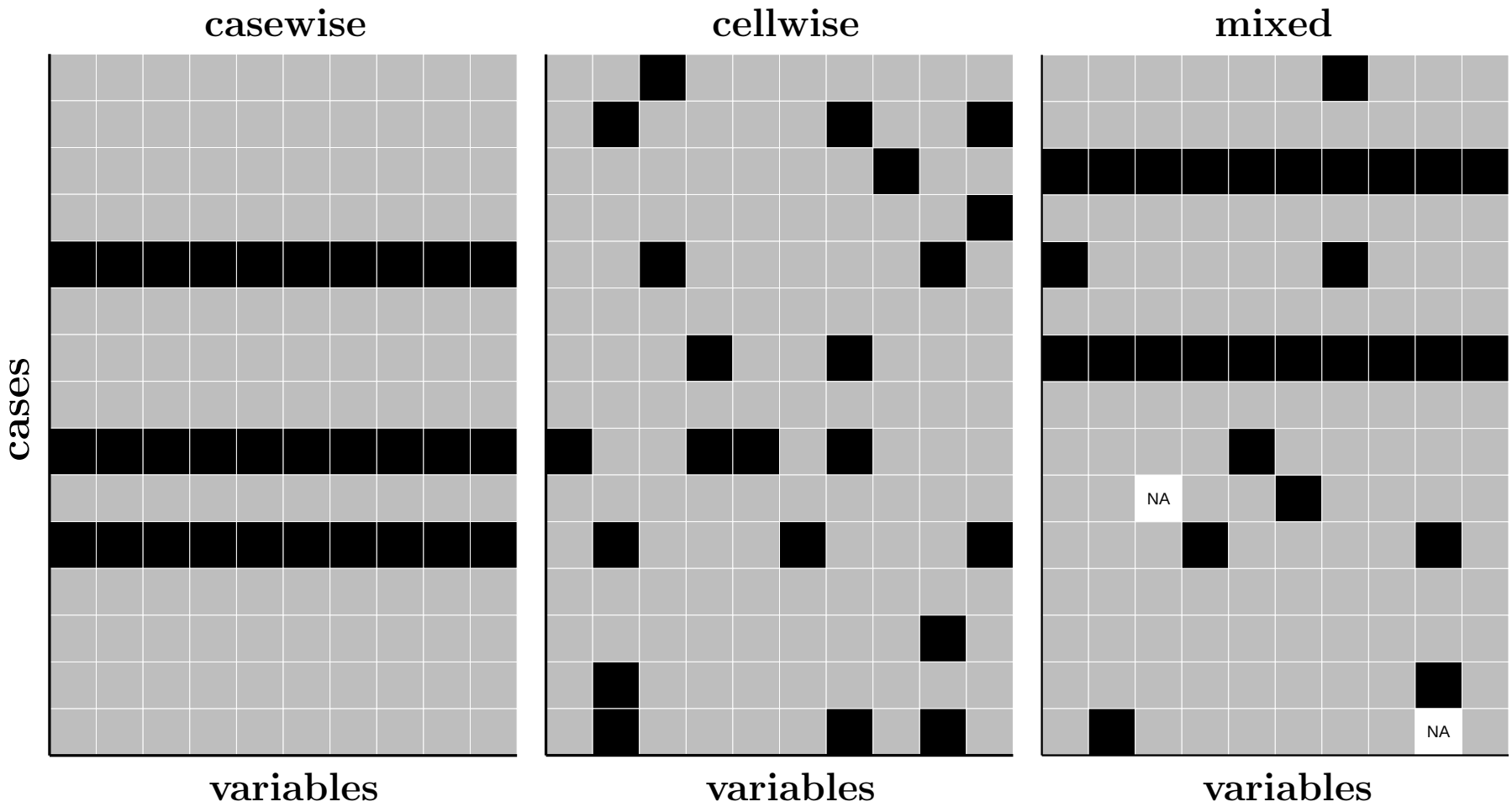
The usual **casewise contamination model** of Tukey (1960), Huber (1964),... assumes that some **rows** $\boldsymbol{x}_i^\top$ have been replaced by arbitrary rows.

Such outlying rows may be cases belonging to a different population.
Many robust and equivariant estimators of covariance, regression, PCA,... were devised for this situation. They downweight outlying rows.
These methods all require at least 50% of clean rows.

The **cellwise contamination model** of Alqallaf, Van Aelst, Yohai and Zamar (2009 AOS) assumes that some **cells** x_{ij} have been replaced.

In that case downweighting/dropping an entire row loses a lot of information.
For high p it is even possible that *every* row contains an outlying cell!

Types of outliers



We can fit a regression model

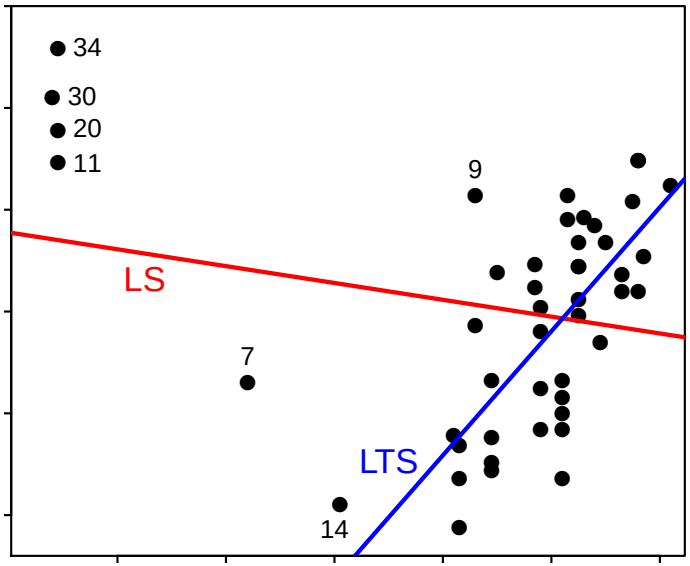
$$y_i = \alpha + \boldsymbol{x}_i^\top \boldsymbol{\beta} + \text{noise}_i$$

by Least Trimmed Squares (**LTS**) regression (Rousseeuw 1984): for $h \geqslant n/2$ put

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

with $(r^2(\alpha, \boldsymbol{\beta}))_{(i)}$ the ith smallest squared residual (first squared, then sorted).

LTS regression is robust to casewise outliers.



In higher dimensions: **detect** casewise outliers by large absolute LTS residuals.

Problem: this casewise LTS may be unable to withstand cellwise outliers, when they pollute over half of the cases.

This can easily happen. For instance, 5% of randomly placed outlying cells in a dataset with 14 variables will contaminate over 50% of the cases.

We are looking for a cellwise robust version of LTS.

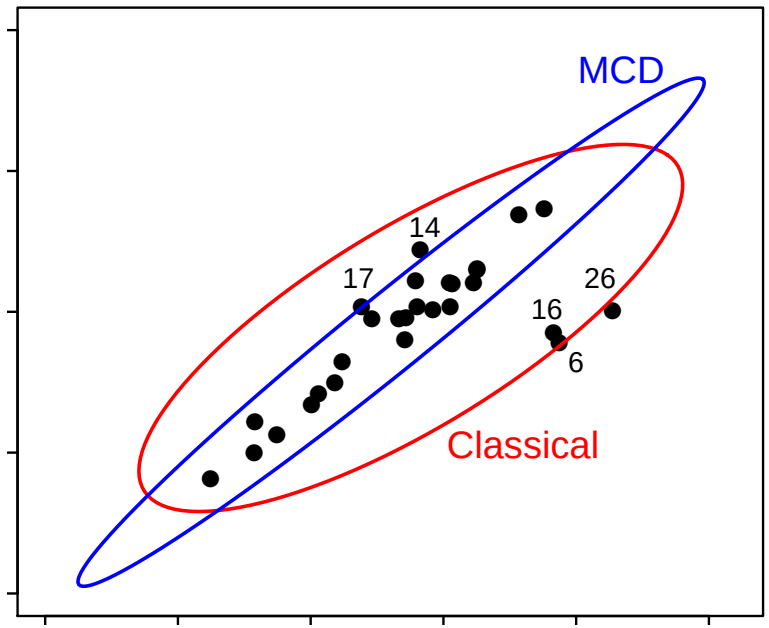
For the p-variate regressors $\boldsymbol{x}_i$ that make up an $n \times p$ matrix $\boldsymbol{X}$, we need a cellwise robust covariance estimator that is able to

- provide **estimates** $\widehat{\boldsymbol{\mu}}$ and $\widehat{\boldsymbol{\Sigma}}$ of the true center and covariance matrix
- **detect** the outlying cells x_{ij}
- **impute** them by inlying cells $\widehat{x}_{ij}$ that follow the estimated structure.

We search for a cellwise robust covariance estimator that generalizes the Minimum Covariance Determinant (**MCD**) method (Rousseeuw 1985).

The casewise MCD looks for subset of h points whose sample covariance matrix has the lowest determinant. It then puts $\widehat{\boldsymbol{\mu}} :=$ mean of that subset, and $\widehat{\boldsymbol{\Sigma}}$ a multiple of its covariance matrix.

The MCD is robust to casewise outliers.
The first practical algorithm was FastMCD (Rousseeuw–Van Driessen 1999).



The MCD can be formulated as a maximum trimmed likelihood estimator!

The cellwise MCD

When constructing a cellwise MCD, h can no longer be the number of cases to include. Instead, h is the minimal number of unflagged cells per column. We create the binary $n \times p$ matrix $\boldsymbol{W}$ with entries w_{ij} that are 0 for flagged cells x_{ij} and 1 otherwise. In the missing value setting, this yields the observed negative log likelihood of case i given by

$$\ln |\boldsymbol{\Sigma}^{(\boldsymbol{W}_{i\cdot})}| + dim(\boldsymbol{W}_{i\cdot}) \ln(2\pi) + \mathrm{MD}^2(\boldsymbol{x}_i, \boldsymbol{W}_{i\cdot}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$$

in which

$$\mathrm{MD}(\boldsymbol{x}_i, \boldsymbol{W}_{i\cdot}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sqrt{\left(\boldsymbol{x}_i^{(\boldsymbol{W}_{i\cdot})} - \boldsymbol{\mu}^{(\boldsymbol{W}_{i\cdot})}\right)^\top \left(\boldsymbol{\Sigma}^{(\boldsymbol{W}_{i\cdot})}\right)^{-1} \left(\boldsymbol{x}_i^{(\boldsymbol{W}_{i\cdot})} - \boldsymbol{\mu}^{(\boldsymbol{W}_{i\cdot})}\right)}$$

is the partial Mahalanobis distance. Here

- $\boldsymbol{x}_i^{(\boldsymbol{W}_{i\cdot})}$ is the vector with only the entries for which $w_{ij} = 1$
- $\boldsymbol{\Sigma}^{(\boldsymbol{W}_{i\cdot})}$ is the submatrix of $\boldsymbol{\Sigma}$ for the variables j with $w_{ij} = 1$.
- $dim(\boldsymbol{W}_{i\cdot})$ is the dimension of $\boldsymbol{x}_i^{(\boldsymbol{W}_{i\cdot})}$ = number of unflagged entries of $\boldsymbol{x}_i$.

The cellMCD method of Raymaekers and Rousseeuw (2024) minimizes

$$\sum_{i=1}^{n}\left(\ln|\boldsymbol{\Sigma}^{(\boldsymbol{W}_{i\cdot})}| + dim(\boldsymbol{W}_{i\cdot})\ln(2\pi) + \mathrm{MD}^2(\boldsymbol{x}_i, \boldsymbol{W}_{i\cdot}, \boldsymbol{\mu}, \boldsymbol{\Sigma})\right) + \sum_{j=1}^{p} q_j ||\mathbf{1}_p - \boldsymbol{W}_{\cdot j}||_0$$

under the constraints $||\boldsymbol{W}_{\cdot j}||_0 \geqslant h$ for all $j = 1, \ldots, p$

and $\lambda_p(\boldsymbol{\Sigma}) \geqslant a > 0$

over all $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and $\boldsymbol{W}$. For a fixed $\boldsymbol{W}$, the EM algorithm minimizes

$$\sum_{i=1}^{n}\left(\ln|\boldsymbol{\Sigma}^{(\boldsymbol{W}_{i\cdot})}| + dim(\boldsymbol{W}_{i\cdot})\ln(2\pi) + \mathrm{MD}^2(\boldsymbol{x}_i, \boldsymbol{W}_{i\cdot}, \boldsymbol{\mu}, \boldsymbol{\Sigma})\right).$$

In the penalty term, $||\mathbf{1}_p - \boldsymbol{W}_{\cdot j}||_0 =$ number of flagged cells in column j.
The constants q_j stem from a cutoff probability, set to 0.99 by default.
This term penalizes flagging more cells than needed, thus helping efficiency.

The first constraint $||\boldsymbol{W}_{\cdot j}||_0 \geqslant h$ ensures at least h unflagged cells per column.

The second constraint $\lambda_p(\boldsymbol{\Sigma}) \geqslant a > 0$ ensures that $\boldsymbol{\Sigma}$ is nonsingular, which is required to compute Mahalanobis distances.

One obstacle is that some regressors $\boldsymbol{X}_{.j}$ and/or the response y may be skewed. We can transform them to symmetry, but what if the linear relation

$$y_i = \alpha + \boldsymbol{x}_i^\top \boldsymbol{\beta} + \text{noise}_i \qquad \text{for} \quad i = 1, \ldots, n$$

holds for the original skewed variables? We can then **symmetrize** the data:

$$(y_k - y_i) = (\boldsymbol{x}_k - \boldsymbol{x}_i)^\top \boldsymbol{\beta} + (\text{noise}_k - \text{noise}_i) \qquad \text{for} \quad i \neq k \text{ in } \{1, \ldots, n\}.$$

The intercept α has canceled, but can easily be estimated afterward.
The new residual variance is $2\sigma^2$.

We also estimate the covariance of $\boldsymbol{X}$ from the symmetrized

$$(\boldsymbol{x}_k - \boldsymbol{x}_i) \qquad \text{for} \quad i \neq k \text{ in } \{1, \ldots, n\}$$

with fixed center $\mathbf{0}$, and divide the resulting $\widehat{\boldsymbol{\Sigma}}$ by 2.

We will denote the symmetrized variables as

$$\widetilde{y}_\ell = y_k - y_i \quad \text{and} \quad \widetilde{\boldsymbol{x}}_\ell = \boldsymbol{x}_k - \boldsymbol{x}_i \qquad \text{for} \quad \ell = 1, \ldots, n(n-1).$$

The proposed **cellLTS** algorithm takes the following steps:

- symmetrize $\boldsymbol{X}$ to $\mathrm{Sym}(\boldsymbol{X})$
- robustly standardize the columns of $\mathrm{Sym}(\boldsymbol{X})$ and $\boldsymbol{X}$
- compute $\widehat{\boldsymbol{\Sigma}}(\mathrm{Sym}(\boldsymbol{X}))$ by cellMCD with fixed center $\mathbf{0}$
- put $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{X}} := \frac{1}{2}\widehat{\boldsymbol{\Sigma}}(\mathrm{Sym}(\boldsymbol{X}))$ and compute $\widehat{\boldsymbol{\mu}}_{\boldsymbol{X}}$ coordinatewise
- compute $\boldsymbol{W}_{\boldsymbol{X}}$ from $\boldsymbol{X}$, keeping $(\widehat{\boldsymbol{\mu}}_{\boldsymbol{X}}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{X}})$ fixed
- impute the cells x_{ij} with $(\boldsymbol{W}_{\boldsymbol{X}})_{ij} = 0$ by $\widehat{x}_{ij}$, yielding $\widehat{\boldsymbol{X}}$
- symmetrize $\widehat{\boldsymbol{X}}$ to $\widetilde{\boldsymbol{X}}$, y to $\widetilde{y}$, and standardize $\widetilde{y}$ and y
- estimate $\boldsymbol{\beta}$ with a new ridge version of LTS, without intercept, given by

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}(\widetilde{H})}{\mathrm{argmin}} \left(\sum_{\ell \in \widetilde{H}} (\widetilde{y}_\ell - \widetilde{\boldsymbol{x}}_\ell^\top \boldsymbol{\beta})^2 + \lambda ||\boldsymbol{\beta}||_2^2 \right)$$

  with $\widetilde{H}$ a subset of $\{1, \ldots, n(n-1)\}$ of size $\widetilde{h} = h(h-1)$
- compute $\widehat{\alpha}$ as the robust location of $y - \widehat{\boldsymbol{X}}\widehat{\boldsymbol{\beta}}$
- undo the above standardizations to obtain the final estimates.

Now we have cellwise robust estimates $\widehat{\boldsymbol{\mu}}_{\boldsymbol{X}}$, $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{X}}$, $\widehat{\alpha}$, and $\widehat{\boldsymbol{\beta}}$.
But what about prediction? **In-sample prediction** is easy, because we already have the imputed $\widehat{\boldsymbol{x}}_i$ with clean cells, so we can put

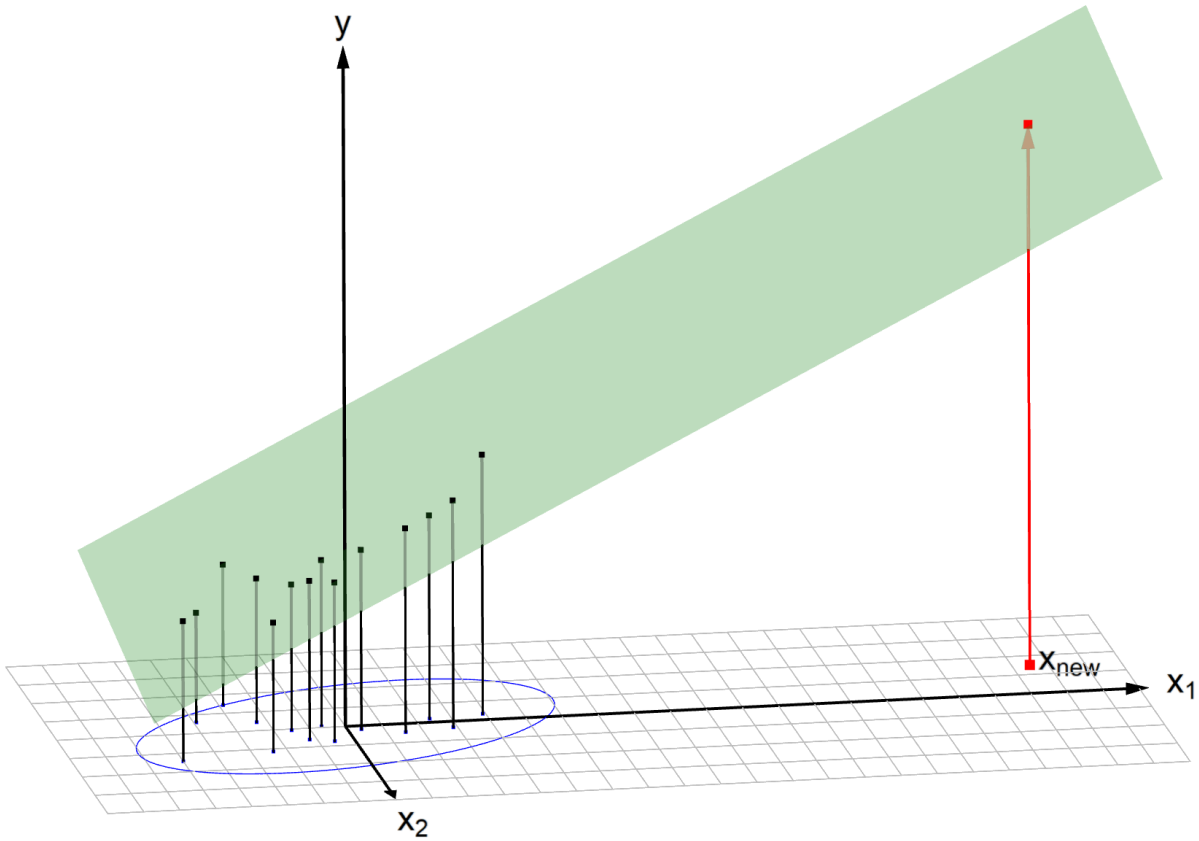
$$\widehat{y}_i := \widehat{\alpha} + \widehat{\boldsymbol{x}}_i \widehat{\boldsymbol{\beta}} \ .$$

Now we want **out-of-sample prediction** at a **new** $\boldsymbol{x}$.
This $\boldsymbol{x}$ can itself have outlying cells and/or NA's... Naively we get:

Now we have cellwise robust estimates $\widehat{\boldsymbol{\mu}}_{\boldsymbol{X}}$, $\widehat{\boldsymbol{\Sigma}}_{\boldsymbol{X}}$, $\widehat{\alpha}$, and $\widehat{\boldsymbol{\beta}}$.
But what about prediction? **In-sample prediction** is easy, because we already have the imputed $\widehat{\boldsymbol{x}}_i$ with clean cells, so we can put

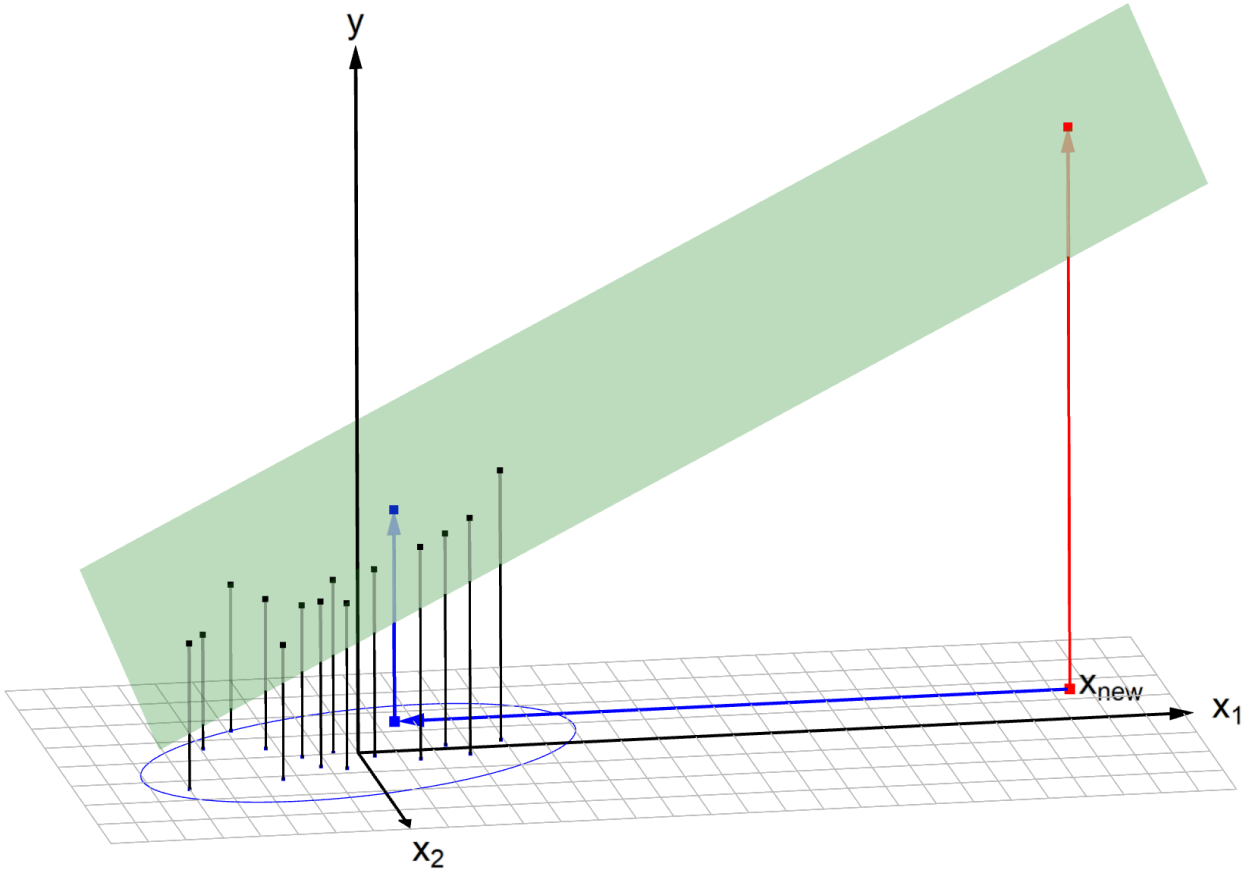
$$\widehat{y}_i := \widehat{\alpha} + \widehat{\boldsymbol{x}}_i \widehat{\boldsymbol{\beta}} \ .$$

Now we want **out-of-sample prediction** at a **new** $\boldsymbol{x}$.
This $\boldsymbol{x}$ can itself have outlying cells and/or NA's... Naively we get:



So we first need an imputed $\widehat{\boldsymbol{x}}$ with clean cells.
We first flag outlying cells of $\boldsymbol{x}$ by iteratively optimizing the cellMCD objective while keeping $(\widehat{\boldsymbol{\mu}}_{\boldsymbol{X}}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{X}})$ fixed.
Then we apply the E-step of EM to $\boldsymbol{x}$, yielding $\widehat{\boldsymbol{x}}$.
Finally, the out-of-sample prediction is given by $\widehat{y} := \widehat{\alpha} + \widehat{\boldsymbol{x}}^\top \widehat{\boldsymbol{\beta}}$.

So we first need an imputed $\widehat{\boldsymbol{x}}$ with clean cells.
We first flag outlying cells of $\boldsymbol{x}$ by iteratively optimizing the cellMCD objective while keeping $(\widehat{\boldsymbol{\mu}}_{\boldsymbol{X}}, \widehat{\boldsymbol{\Sigma}}_{\boldsymbol{X}})$ fixed.
Then we apply the E-step of EM to $\boldsymbol{x}$, yielding $\widehat{\boldsymbol{x}}$.
Finally, the out-of-sample prediction is given by $\widehat{y} := \widehat{\alpha} + \widehat{\boldsymbol{x}}^\top \widehat{\boldsymbol{\beta}}$.



CellLTS is the only cellwise robust regression method with this feature.

Breakdown value of cellLTS

We write the model as $\boldsymbol{y} = [\mathbf{1}_n \, ; \boldsymbol{X}] \, \boldsymbol{\gamma} +$ noise where $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$ and $\boldsymbol{\gamma} = (\alpha, \beta_1, \ldots, \beta_p)^\top$. Also put $\boldsymbol{Z} := [\boldsymbol{X} \, ; \, \boldsymbol{y}]$. Replacing at most m cells in each column of $\boldsymbol{Z}$ by arbitrary values yields a contaminated matrix denoted as $\boldsymbol{Z}^m$. The *cellwise finite-sample breakdown value* of $\widehat{\boldsymbol{\gamma}}$ at $\boldsymbol{Z}$ is

$$\varepsilon_n^{\text{cell}}(\widehat{\boldsymbol{\gamma}}, \boldsymbol{Z}) := \min \left\{ \frac{m}{n} : \sup_{\boldsymbol{Z}^m} ||\widehat{\boldsymbol{\gamma}}(\boldsymbol{Z}^m) - \widehat{\boldsymbol{\gamma}}(\boldsymbol{Z})|| = \infty \right\}.$$

Proposition. *If $\boldsymbol{X}$ is in general position and $h \geqslant [(n+p+1)/2]$, then the breakdown value of cellLTS with $\widetilde{h} = h(h-1)$ is $\varepsilon_n^*(\widehat{\boldsymbol{\gamma}}, \boldsymbol{Z}) = (n-h+1)/n$.*

This is the first breakdown result for a cellwise robust regression method.
We are also robust to outlying cases:

Corollary. *The casewise breakdown value of cellLTS equals $\varepsilon_n^{cell}(\widehat{\boldsymbol{\gamma}}, \boldsymbol{Z})$.*

This is the same as for the casewise LTS, and can be made as high as 50%.

Simulation

We generate $p = 20$ clean exponential variables $\boldsymbol{X}_{\cdot j}$ with covariance matrix $\boldsymbol{\Sigma}_{\text{A09}}$ given by $(\boldsymbol{\Sigma}_{\text{A09}})_{ij} = (-0.9)^{|i-j|}$.

We set the true coefficients equal to $\boldsymbol{\beta} = [p \quad p-1 \quad \cdots \quad 2 \quad 1]$ and compute the clean response $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} +$ noise .

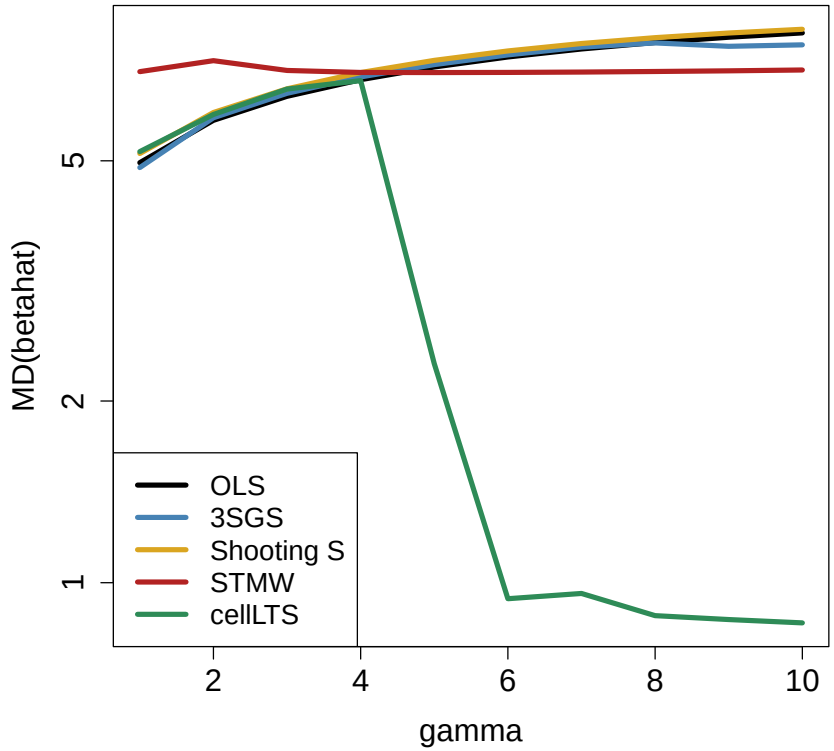
To create cellwise outliers, we start by drawing a fraction $\varepsilon = 20\%$ of random cells in each column of the clean data matrix $\boldsymbol{Z}$, including the response variable $\boldsymbol{y}$. Each contaminated cell is then put at a value $\gamma = 1, \ldots, 10$.

We compare cellLTS with:

- **OLS**
- **3SGS**: the three-step generalized S of Leung et al. (2016)
- **Shooting S**: from Bottmer et al. (2022)
- **STMW**: from Su, Tarr, Muller and Wang (2024).

To measure the accuracy of $\widehat{\beta}$ we compute the distance

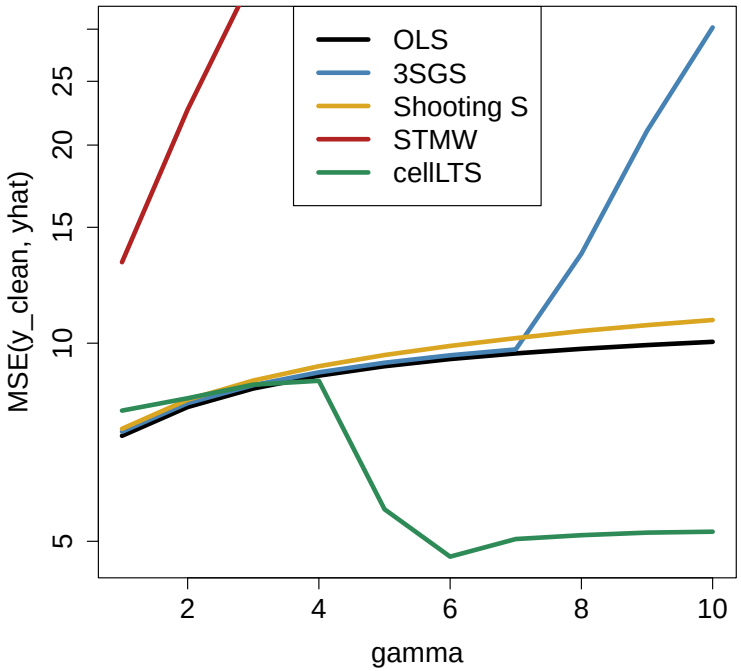
$$\text{MD} = \sqrt{(\widehat{\beta} - \beta)^\top \boldsymbol{\Sigma_X} (\widehat{\beta} - \beta)}\,.$$



The average(MD) grows with γ for 3SGS, OLS and Shooting S. STMW is more stable but high. The MD of cellLTS grows for small γ, when the contaminated cells are not far enough to be truly outlying, and then comes down.

We also generate clean out-of-sample regressor matrices and their clean responses $\boldsymbol{y}^*$, after which we contaminate the regressor matrices $\boldsymbol{X}^*$ as in training. We then compute the out-of sample predictions $\widehat{\boldsymbol{y}}^*$ and

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i^* - y_i^*)^2 \ .$$

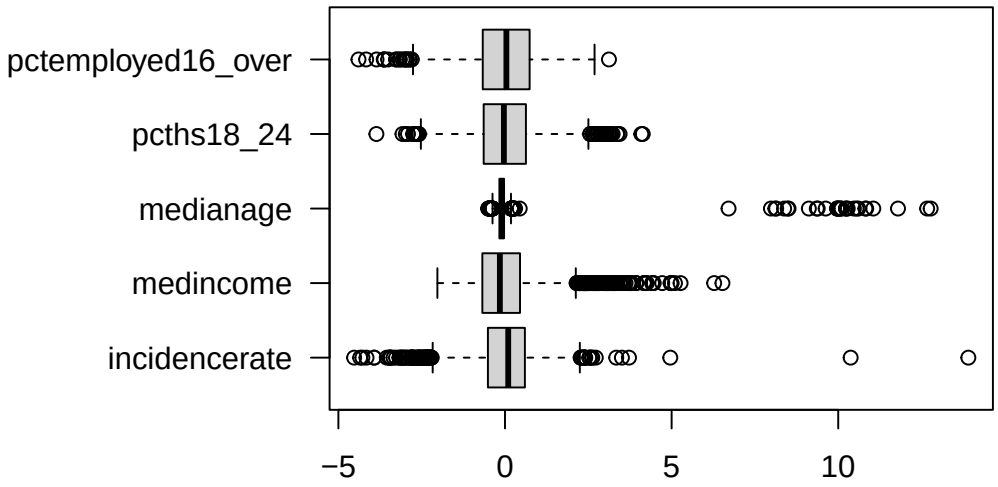


Now STMW is the most affected, followed by 3SGS that breaks away from OLS and Shooting S. CellLTS continues to perform well, because it robustly imputes outlying cells in the out-of-sample $\boldsymbol{X}^*$ before computing its prediction $\widehat{\boldsymbol{y}}^*$.

Real data example: US Cancer data

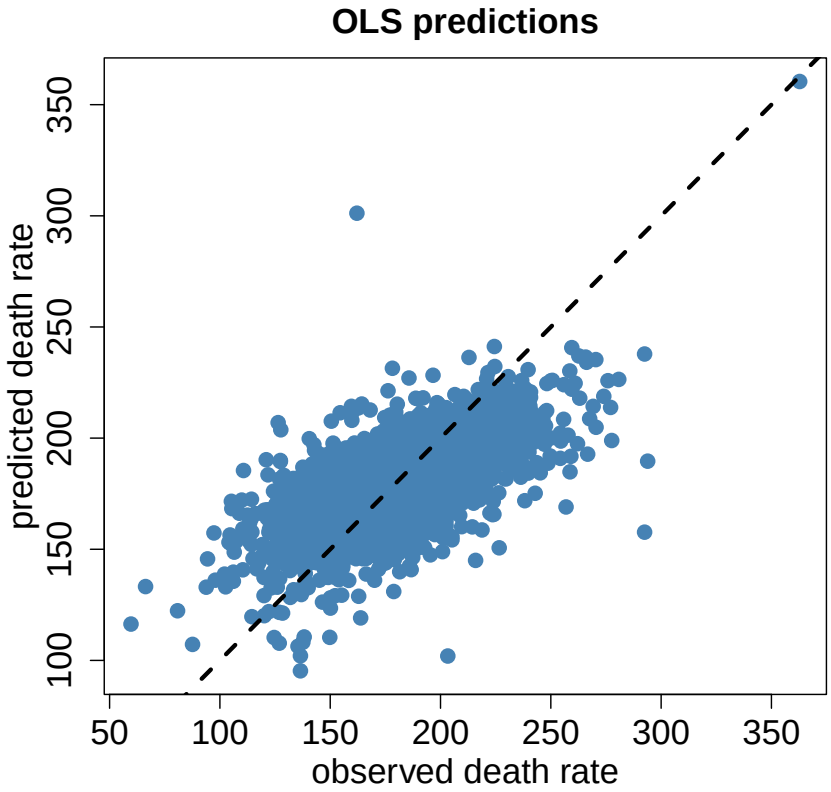
Demographics and cancer statistics for 3047 counties in the US, available at `https://www.kaggle.com/datasets/varunraskar/cancer-regression/data`.
Response y = death rate due to cancer (in cases per 100 000). Regressors:

feature	explanation
`incidencerate`	Incidence rate of cancer (per 100 000).
`medincome`	Median income in the county (in $1000).
`medianage`	Median age in the county.
`pcths18_24`	Percentage of residents aged 18-24 whose highest attained education is a high school diploma.
`pctemployed16_over`	Percentage of people aged 16+ who are employed.

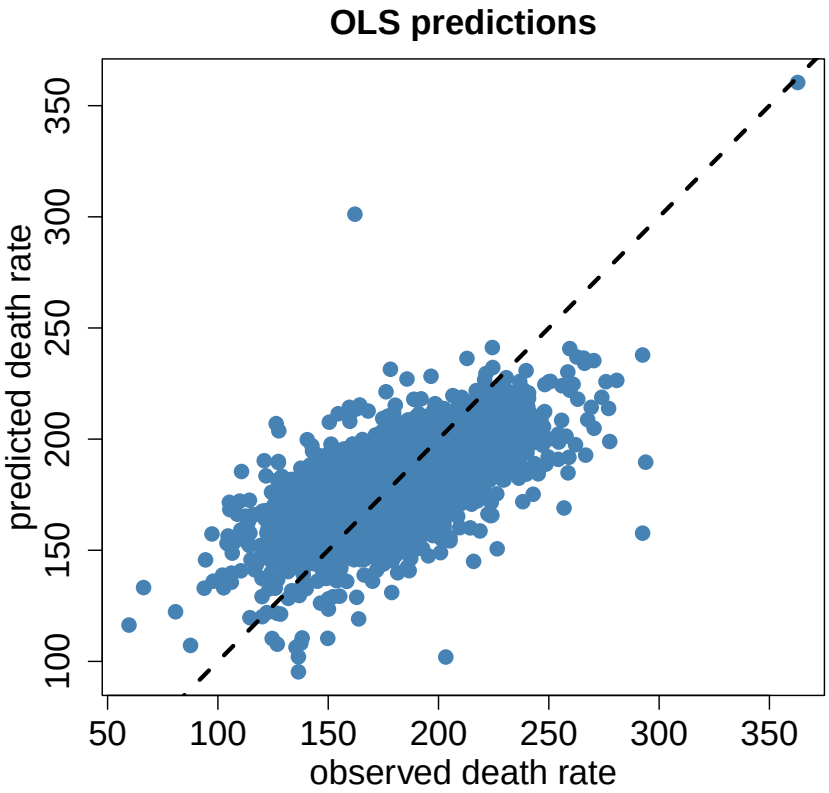


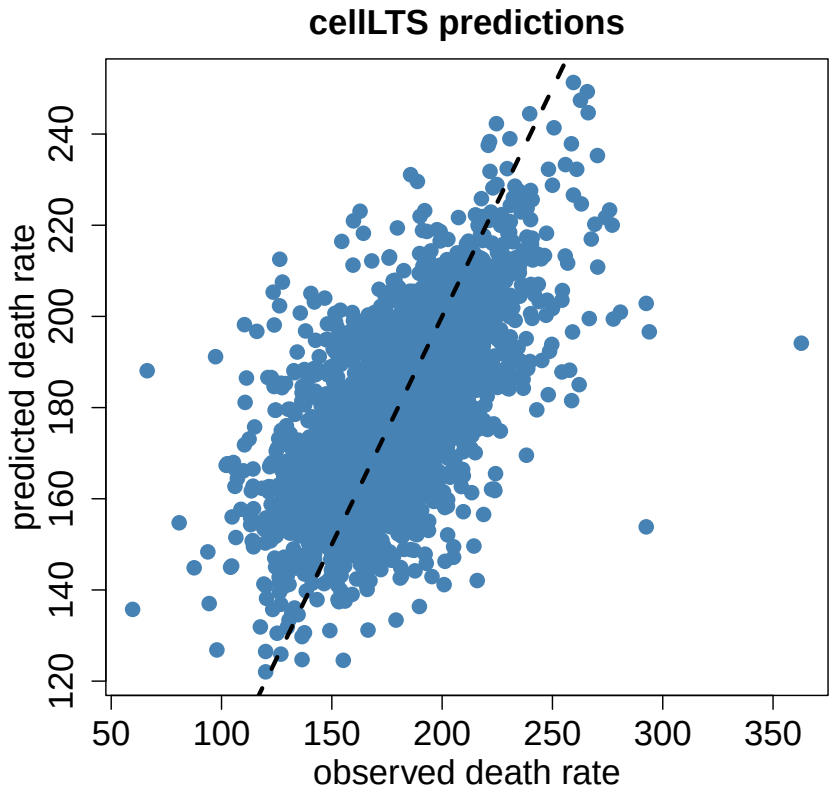
Boxplot of scaled regressors: extreme outliers in median age and incidence rate!

The outlying cells in median age create bad leverage points. The OLS coefficient of the median age is badly affected, yielding poor predictions:



The outlying cells in median age create bad leverage points. The OLS coefficient of the median age is badly affected, yielding poor predictions:

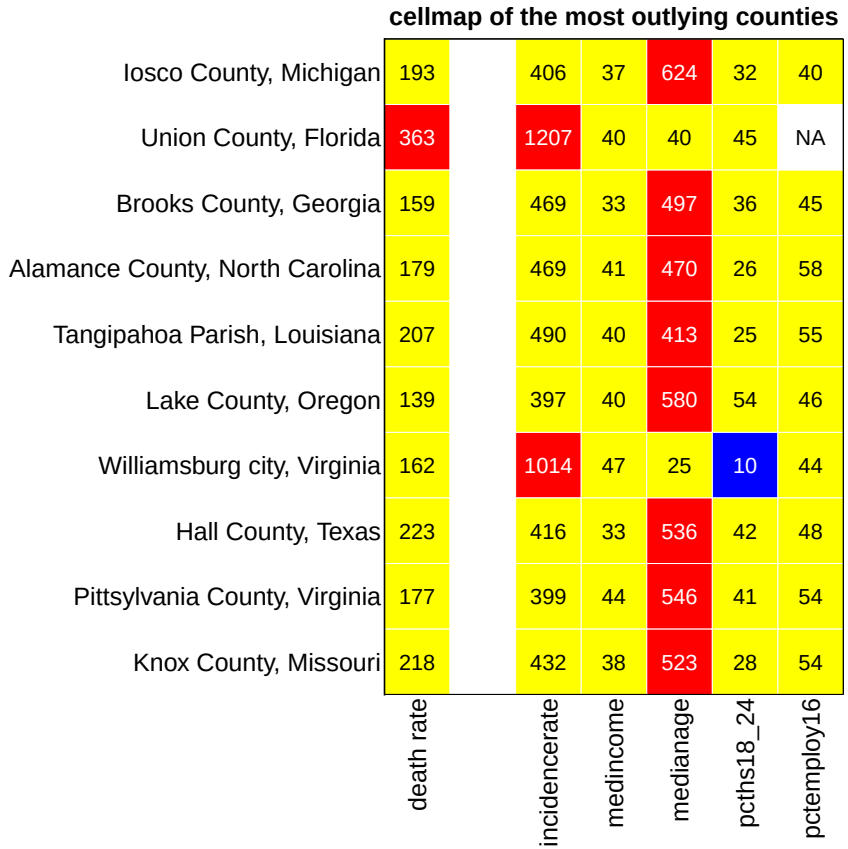




The cellLTS coefficients fit better.

Due to its new prediction mechanism, cellLTS doesn't produce outlying predictions in contaminated regressors.

Regression cellmap of the 10 counties with the highest $\sum_{j=1}^{p+1} \left| \frac{z_{ij} - \hat{z}_{ij}}{\hat{\sigma}_j} \right|$.
Has response in the left column, then the regressors:



- Median ages of 400+ years are definitely errors.
- Union County: huge death rate and incidence rate. They are correct.
- Williamsburg City: incidence rate is an error. Very few 18–24 year olds with only high school diploma! Is correct: College of William and Mary.

Summary

- cellLTS can withstand cellwise and casewise outliers.
- It can deal with skewed data and missing values.
- It is the first method designed to make robust out-of-sample predictions in contaminated datapoints.
- It has a high breakdown value.

Preprint: arXiv 2603.04632 .

The R code and an example script are at
`https://wis.kuleuven.be/statdatascience/code/cellreg_code_script.zip`.

Summary

- cellLTS can withstand cellwise and casewise outliers.
- It can deal with skewed data and missing values.
- It is the first method designed to make robust out-of-sample predictions in contaminated datapoints.
- It has a high breakdown value.

Preprint: arXiv 2603.04632 .

The R code and an example script are at
`https://wis.kuleuven.be/statdatascience/code/cellreg_code_script.zip`.

Thank you!

References

Bottmer, L., C. Croux, and I. Wilms (2022). Sparse regression for large data sets with outliers. *European Journal of Operational Research 297*(2), 782–794.

Leung, A., H. Zhang, and R. Zamar (2016). Robust regression estimation and inference in the presence of cellwise and casewise contamination. *Computational Statistics & Data Analysis 99*, 1–11.

Raymaekers, J. and P. Rousseeuw (2024). The Cellwise Minimum Covariance Determinant Estimator. *Journal of the American Statistical Association 119*, 2610–2621.

Rousseeuw, P. J. (1984). Least Median of Squares Regression. *Journal of the American Statistical Association 79*, 871–880.

Su, P., G. Tarr, S. Muller, and S. Wang (2024). CR-Lasso: Robust cellwise regularized sparse regression. *Computational Statistics and Data Analysis 197*(107971), 1–14.