Noname manuscript No.
(will be inserted by the editor)

The shooting S-estimator for robust regression

**Viktoria Öllerer · Andreas Alfons ·
Christophe Croux**

Received: date / Accepted: date

Abstract To perform multiple regression, the least squares estimator is commonly used. However, this estimator is not robust to outliers. Therefore, robust methods such as S-estimation have been proposed. These estimators flag any observation with a large residual as an outlier and downweight it in the further procedure. However, a large residual may be caused by an outlier in only one single predictor variable, and downweighting the complete observation results in a loss of information. Therefore, we propose the *shooting S-estimator*, a regression estimator that is especially designed for situations where a large number of observations suffer from contamination in a small number of predictor variables. The shooting S-estimator combines the ideas of the coordinate descent algorithm with simple S-regression, which makes it robust against componentwise contamination, at the cost of failing the regression equivariance property.

Keywords cellwise outliers · componentwise contamination · shooting algorithm · coordinate descent algorithm · regression S-estimation

1 Introduction

In robust statistics it is generally assumed that the majority of observations is totally free of contamination. Any observation that deviates from the model is

V. Öllerer
Faculty of Economics and Business, KU Leuven, Naamsestraat 69, 3000 Leuven, Belgium
E-mail: viktoria.oellerer@kuleuven.be

A. Alfons
Erasmus School of Economics, Erasmus University Rotterdam, PO Box 1738, 3000 DR Rotterdam, The Netherlands
E-mail: alfons@ese.eur.nl

C. Croux
Faculty of Economics and Business, KU Leuven, Naamsestraat 69, 3000 Leuven, Belgium
E-mail: christophe.croux@kuleuven.be

as a whole flagged as an outlier, even if only one component of the observation is contaminated. In case only a small number of predictor variables cause the deviation from the model, a lot of information is lost through downweighting the whole observation. Therefore, it seems more appropriate to not consider whole observations as outliers but only those components that really deviate from the model. This is especially useful if the majority of observations is contaminated in only a small number of variables. Imagine, for example, a regression setting where in every observation one single predictor variable is contaminated. Here the usual robust methods break down, as there is not one single clean observation. But the majority of the cells of the design matrix is still clean and thus the majority of the data is still clean. In this setting, it is more suitable to use techniques developed for cellwise contamination (componentwise contamination) rather than those developed for rowwise contamination.

Alqallaf et al (2009) extend the rowwise contamination model to also cover cellwise contamination. They define the influence function and the breakdown point in this setting and derive them for some multivariate location estimators, showing that these cannot cope with cellwise contamination. For principal component analysis, Van Aelst et al (2010) develop a method based on pairwise correlation that can deal with cellwise contamination. The same authors propose versions of the Stahel-Donoho estimator based on Huberized outlyingness (see Van Aelst et al, 2012) and cellwise weights (see Van Aelst et al, 2011).

In this paper we derive a regression estimator, called the *shooting S-estimator*, that can cope with cellwise contamination. It combines the ideas of the coordinate descent algorithm ('shooting algorithm') (see Friedman et al, 2007; Fu, 1998) with simple regression S-estimation (see Maronna et al, 2006). In Section 2, we introduce the estimator. An algorithm is proposed in Section 3. We show simulation results in Section 4 where we compare the shooting S-estimator to the least squares estimator and the robust S- and MM-estimators. Real data examples are presented in Section 5 and Section 6 concludes.

2 Motivation

Our *shooting S-estimator* uses the idea of the coordinate descent algorithm (see Friedman et al, 2007), also called shooting algorithm (Fu, 1998). Originally, this method performs, variable by variable, simple lasso regression. Tseng (2001) showed that by iteratively looping through all variables, it converges to the lasso estimate for any starting value. However, it is well known that the lasso estimate is not robust (see e.g. Alfons et al, 2013). In the shooting S-estimator, we achieve robustness by replacing the lasso estimation with unpenalized S-estimation (see Maronna et al, 2006). In contrast to ordinary S-regression, the coordinate-wise approach of the coordinate descent algorithm allows us to weight all components of an observation differently.

The lasso estimate is defined as

$$\hat{\boldsymbol{\beta}}_{Lasso} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\arg\min}\ \frac{1}{n}\sum_{i=1}^{n}(y_i - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + 2\lambda\sum_{j=1}^{p}|\beta_j|.$$

In the coordinate descent algorithm, to update the estimate of the lasso coefficient $\hat{\beta}_j$ $(j = 1, \dots, p)$, all other coefficients are kept fixed at $\hat{\beta}_k$ $(k \neq j)$

$$\begin{aligned}\hat{\beta}_{j,Lasso} &= \underset{\beta_j\in\mathbb{R}}{\arg\min}\ \frac{1}{n}\sum_{i=1}^{n}((y_i - \sum_{k\neq j} x_{ik}\hat{\beta}_k) - x_{ij}\beta_j)^2 + 2\lambda\sum_{k\neq j}|\hat{\beta}_k| + 2\lambda|\beta_j| \\ &= \underset{\beta_j\in\mathbb{R}}{\operatorname{argmin}}\ \frac{1}{n}\sum_{i=1}^{n}((y_i - \sum_{k\neq j} x_{ik}\hat{\beta}_k) - x_{ij}\beta_j)^2 + 2\lambda|\beta_j|. \qquad (1)\end{aligned}$$

This can be seen as simple lasso regression where the new response

$$y_i^{(j)} = y_i - \sum_{k\neq j} x_{ik}\hat{\beta}_k, \quad i = 1, \dots, n, \qquad (2)$$

is regressed on x_{ij}, for a fixed value of j.

For the shooting S-estimator, we want to make sure that the new response $\tilde{y}_i^{(j)}$, to be defined below, is not influenced by outliers in the cells x_{ik}. Therefore, we first define regression weights

$$w_{ik} = w(\frac{|\tilde{y}_i^{(k)} - x_{ik}\hat{\beta}_k|}{\hat{\sigma}_k}) \qquad (3)$$

where the argument of the weighting function $w(\cdot)$ is the residual of regressing $\tilde{y}_i^{(k)}$ on x_{ik}, scaled by a robust residual scale $\hat{\sigma}_k$. Thus, w_{ik} determines the 'outlyingness' of the cell x_{ik} in the regression $\tilde{y}_i^{(k)}$ on x_{ik}. The weighting function should be non-increasing on the positive numbers and take values in the interval $[0, 1]$. Our preferred option - for reasons of simplicity - is hard rejection, where $w(r) = 1$ if $r \leq c$ and 0 otherwise. Choosing the cut-off value $c = 3$, less than 0.3% of clean observations are expected to be flagged as outliers in the regression model with normal errors. Of course, other choices for the weight function are possible.

The new response is defined as

$$\tilde{y}_i^{(j)} = y_i - \sum_{k\neq j} \tilde{x}_{ik}\hat{\beta}_k \quad \text{with } \tilde{x}_{ik} = w_{ik}x_{ik} + (1 - w_{ik})\hat{x}_{ik} \qquad (4)$$

The difference with (2) is that in the computation of the new response the values x_{ik} are replaced by a convex combination $\tilde{x}_{ik}$ of the observed value x_{ik} and of a 'corrected' value $\hat{x}_{ik}$. As we know $\tilde{y}_i^{(k)}$ and $\hat{\beta}_k$, this 'corrected' value $\hat{x}_{ik}$ is computed through calibration (Brown, 1982):

$$\hat{x}_{ik} = \frac{\tilde{y}_i^{(k)}}{\hat{\beta}_k}. \qquad (5)$$

(To avoid computational problems, we set $\hat{x}_{ik} = 0$ in case $|\hat{\beta}_k|$ is small.) The $\tilde{x}_{ik}$ can be interpreted as a cleaned version of the cell value x_{ik} in the design matrix. If an observation is flagged as an outlier and gets a zero weight, the $\tilde{x}_{ik}$ equals the 'corrected' value $\hat{x}_{ik}$. If an observation is declared as clean and gets a weight of one, the cleaned version equals the observed value. Note that $\hat{x}_{ik}$ and w_{ik} depend on $\hat{\beta}_k$, for $k \neq j$.

To compute the regression estimate $\hat{\beta}_j$, we use instead of the lasso as in (1), the robust unpenalized simple S-regression estimator. This leads us to the *shooting S-estimator*, which is defined variablewise conditional on knowing the other estimates $\hat{\beta}_k$ with $k \neq j$,

$$\hat{\beta}_j = \operatorname*{arg\,min}_{\beta \in \mathbb{R}} \hat{\sigma}_j(\beta) \tag{6}$$

with $\hat{\sigma}_j(\beta)$ defined as solution s of the equation

$$\frac{1}{n}\sum_{i=1}^{n} \rho\left(\frac{\tilde{y}_i^{(j)} - x_{ij}\beta}{s}\right) = \delta. \tag{7}$$

Hence, $\hat{\sigma}_j(\hat{\beta}_j)$ is an M-estimator of scale computed from the residuals. Here δ equals the expected value of the ρ-function at the normal distribution, i.e. $\delta = \mathbb{E}[\rho(Z)]$ with $Z \sim \mathcal{N}(0,1)$. It is chosen such that the breakdown point of the estimator is not too low, while its efficiency is high enough. A higher value of δ implies a higher breakdown point, but a lower efficiency (see e.g. Rousseeuw and Leroy, 1987, Chapter 3.4).

As a ρ-function we will use either Tukey's biweight

$$\rho_{BI}(z) = \begin{cases} \frac{k_{BI}^2}{6}(1-(1-(\frac{z}{k_{BI}})^2)^3) & \text{if } |z| \leq k_{BI} \\ \frac{k_{BI}^2}{6} & \text{if } |z| > k_{BI}, \end{cases} \tag{8}$$

or the skipped Huber

$$\rho_{skH}(z) = \begin{cases} \frac{1}{2}z^2 & \text{if } |z| \leq k_{skH} \\ \frac{k_{skH}^2}{2} & \text{if } |z| > k_{skH}. \end{cases} \tag{9}$$

These two ρ-functions are quite different in nature. The skipped Huber loss is quadratic in a central region $[-k_{skH}, k_{skH}]$ and constant outside this interval. Thus, skipped Huber is a skipped version of the quadratic loss. In contrast, the biweight loss is designed to be smooth while still bounding the effect of extreme values. Apart from those two loss functions any ρ-function (see Maronna et al, 2006, p31, Def 2.1) could be used as well.

The shooting S-estimator fulfills some natural equivariance properties. Assume a regression model with intercept. The estimator is computed using the coordinate descent algorithm with starting value described in Section 3. If a constant a is added to an explanatory variable, the corresponding estimate of the slope coefficient $\hat{\beta}_j$ stays unchanged, while the intercept shifts by $a\hat{\beta}_j$.

If a constant a is added to the response, none of the estimated coefficients changes and the intercept shifts by a. These properties can be shown using Equations (3), (4), (5) and (6), as well as the properties of the proposed initial estimator. If a multiple γ of an explanatory variable is added to the response, we would like the corresponding slope coefficient to become $\gamma + \hat{\beta}_j$. This type of regression equivariance is fulfilled if the starting value has this property. Since the proposed starting value uses Huberized values for the predictor variables, this property does not fully hold, although one could say that for the converged estimator it 'practically' holds.

3 Algorithm

To compute the shooting S-estimate described in Section 2, we use an iterative procedure similar to the coordinate descent algorithm. We first describe the iteration steps, and afterwards the determination of initial values (see Algorithm 1 for details). We assume that the model contains an intercept, denoted by α.

Loop. In each step of the coordinate descent loop (with fixed j), we calculate the $\tilde{y}_i^{(j)}$ by (4) and (5) and then compute the simple regression S-estimate of the $\tilde{y}_i^{(j)}$ on the x_{ij}. To do this, we use the iteratively reweighted least squares (IRLS) algorithm recommended by Maronna et al (2006). It consists of another iterative algorithm. In each iteration, called an I-step, a weighted least squares estimate of β_j is calculated and subsequently, a new value of the M-estimator of scale $\hat{\sigma}_j(\hat{\beta}_j)$ is computed by searching a fixed point of a recursive version of (7), $f(s) = 1/(n\delta)\sum_{i=1}^{n} \rho((\tilde{y}_i^{(j)} - x_{ij}\hat{\beta}_j)/s)s = s$.

Although convergence of the coordinate descent loop is not assured, we have observed it empirically in all our simulations studies.

Initial values. We first Huberize the predictor values, and get 'approximately clean' predictors $\tilde{x}_{ij}^{0}$. Then we use the MM-estimator to get initial coefficients $\hat{\beta}_j^{(0)}$, with the *linear quadratic quadratic (lqq)* ρ-function (Koller and Stahel, 2011) and tuning constants set for 50% breakdown point and 95% efficiency.

Algorithm 1 gives the details. The code of the algorithm is available on the homepage of the first author.

4 Simulations

To evaluate the shooting S-estimator, we compare it to the classical least squares estimator (LS), the ordinary S-estimator and the MM-estimator (see Maronna et al, 2006). The shooting S-estimator is computed as in Algorithm 1 once with the biweight ρ-function (8) and once with the skipped Huber ρ-function (9). We choose $k_{BI} = 3.420$ and $k_{skH} = 2.177$. This corresponds to

Algorithm 1 ***Computation of the shooting S-estimate for a regression model with constant term***

```
# Initialization
– $L := 0$ # Number of steps in coordinate descent loop
– $\tilde{x}_{ij}^{(0)} = \max(\mathrm{med}_i(x_{ij}) - 2\,\mathrm{mad}_i(x_{ij}), \min(x_{ij}, \mathrm{med}_i(x_{ij}) + 2\,\mathrm{mad}_i(x_{ij})))$
– Compute the slopes $\hat{\boldsymbol{\beta}}^{(0)}$, the intercept $\hat{\alpha}$ and the residual scale $\hat{s}$ from the MM-regression of $y_i$ on the Huberized predictors $\tilde{x}_{ij}^{(0)}$ using the lqq $\rho$-function
– $\hat{\alpha}_j^{(0)} := \hat{\alpha}, \quad j = 1, \ldots, p$
– $s_j^{(0)} := \hat{s}, \quad j = 1, \ldots, p$

      # Coordinate descent loop
      ◇ $L := L + 1$
      ◇ For $j = 1, \ldots, p$ # Index of the variable used in regression step

          # Regression step
          ▪ $\tilde{y}_i^{(j)} := y_i - \sum_{k<j} \tilde{x}_{ik}^{(L)} \hat{\beta}_k^{(L)} - \sum_{k>j} \tilde{x}_{ik}^{(L-1)} \hat{\beta}_k^{(L-1)}, \qquad i = 1, \ldots, n$
          ▪ $r := 0$ # Number of I-steps
          ▪ $res_i^{(L,0)} := \tilde{y}_i^{(j)} - x_{ij}\hat{\beta}_j^{(L-1)} - \mathrm{med}_i(\tilde{y}_i^{(j)} - x_{ij}\hat{\beta}_j^{(L-1)}),$
                                                            $i = 1, \ldots, n$
          ▪ $\omega_{ij}^{(L,0)} := \rho'(res_i^{(L,0)}/s_j^{(L-1)})/(res_i^{(L,0)}/s_j^{(L-1)}), \qquad i = 1, \ldots, n$

              # I-steps
              ○ $r := r + 1$
              ○ Compute the slope $\hat{\beta}_j^{(L,r)}$ and the intercept $\hat{\alpha}_j^{(L,r)}$ from the weighted least squares regression of $\tilde{y}_i^{(j)}$ on $x_{ij}$ with weights $\omega_{ij}^{(L,r-1)}$ # $j$ is fixed
              ○ $res_i^{(L,r)} := \tilde{y}_i^{(j)} - x_{ij}\hat{\beta}_j^{(L,r)} - \hat{\alpha}_j^{(L,r)}, \quad i = 1, \ldots, n$
              ○ $\ell := 0$ # Number of M-steps to compute scale
              ○ $s_0 = \begin{cases} \mathrm{med}_i\, |res_i^{(L,r)}| \cdot 1.4826 & \text{if } r = 1 \\ s_j^{(L,r-1)} & \text{if } r > 1 \end{cases}$

                  # M-step
                  ▲ $\ell := \ell + 1$
                  ▲ $s_\ell := \sqrt{\frac{s_{\ell-1}^2}{\delta \cdot n} \sum_{i=1}^n \rho(\frac{res_i^{(L,r)}}{s_{\ell-1}})}$
                  ▲ Repeat M-step until $|\frac{s_\ell}{s_{\ell-1}} - 1| < \epsilon_1 = 10^{-6}$

              ○ $s^{(L,r)} := s_\ell$
              ○ $\omega_{ij}^{(L,r)} := \rho'(res_i^{(L,r)}/s^{(L,r)})/(res_i^{(L,r)}/s^{(L,r)}),$
                                                            $i = 1, \ldots, n$
              ○ Repeat I-step until $\max_i |res_i^{(L,r)} - res_i^{(L,r-1)}| < \epsilon_2$
                # $\epsilon_2 = 10^{-6}\mathrm{mad}_i\, y_i$

          ▪ $\hat{\beta}_j^{(L)} := \hat{\beta}_j^{(L,r)}$
          ▪ $\hat{\alpha}_j^{(L)} := \hat{\alpha}_j^{(L,r)}$
          ▪ $s_j^{(L)} := s^{(L,r)}$
          ▪ $res_i^{(L)} := res_i^{(L,r)} \qquad i = 1, \ldots, n$
          ▪ $\hat{x}_{ij}^{(L)} := \begin{cases} (\tilde{y}_i^{(j)} - \hat{\alpha}_j^{(L)})/\hat{\beta}_j^{(L)} & \text{if } |\hat{\beta}_j^{(L)}| \geq \epsilon_3 \\ \mathrm{med}_i\, x_{ij} & \text{otherwise} \end{cases} \qquad i = 1, \ldots, n$
            # $\epsilon_3 = 10^{-4}(\mathrm{mad}_i\, y_i)/(\mathrm{mad}_i\, x_{ij})$
          ▪ $w_{ij}^{(L)} := w(res_i^{(L)}/s_j^{(L)}) \qquad i = 1, \ldots, n$
          ▪ $\tilde{x}_{ij}^{(L)} := w_{ij}^{(L)} x_{ij} + (1 - w_{ij}^{(L)})\hat{x}_{ij}^{(L)}, \quad i = 1, \ldots, n$

      ◇ # End for-loop
      ◇ Repeat coordinate descent loop until $\sum_{j=1}^p |s_j^{(L)} - s_j^{(L-1)}| < \epsilon_4$
        # $\epsilon_4 = 10^{-2}\,\mathrm{mad}_i\, y_i$
– $\hat{\beta}_j := \hat{\beta}_j^{(L)}$
– $\hat{\alpha} := \mathrm{med}_i(y_i - \sum_{j=1}^p \tilde{x}_{ij}^{(L)} \hat{\beta}_j^{(L)})$
```

a breakdown point of 20% in the simple regressions. Our choice seems to be a good trade-off between robustness and efficiency. In practice, the breakdown point needs to be increased if the data at hand is more severely contaminated than in this simulation setting. For the computation of the ordinary S-estimate, we use the biweight loss function and set again $k_{BI} = 3.420$. The MM-estimator is computed with the standard settings of 50% breakdown point and an efficiency of 95% at the normal model, using the biweight loss function. We stick here to the high breakdown point of 50%, as MM can achieve high efficiency and a high breakdown point simultaneously. Thus, lowering the breakdown point would not increase the efficiency of the MM-estimator.

For the simulation setup we take $n = 100$ and $p = 15$. The regression coefficients $\boldsymbol{\beta}$ are taken equally spaced over the interval [0,1], i.e. $\beta_j = j/p$ for $j = 1, \dots, p$. The predictors $\mathbf{x}_i$ and errors e_i are independent and identically normally distributed with mean 0 for $i = 1, \dots, n$. We choose two different sampling schemes, one with uncorrelated and one with correlated predictors. For the first one, we use the identity matrix as a covariance matrix for the predictors. The error variance is $\sigma^2 = 0.5^2$. In the correlated setting we choose the predictor covariance matrix Σ with $\Sigma_{ij} = 0.5^{|i-j|}$ and the error variance $\sigma^2 = 0.81^2$. By this the signal-to-noise ratio[1] is the same in both settings. The response variable is then created as $y_i = \mathbf{x}_i'\boldsymbol{\beta} + e_i$ for $i = 1, \dots, n$.

To every generated data set, we add 1%, 2%, 5% and 10% of cellwise contamination. The cells x_{ij} that we contaminate are chosen randomly from the design matrix X. So every cell of our data set is equally likely to be contaminated. Three different contamination settings are used: a dense cluster $x_{ij}^{cont} \sim \mathcal{N}(50, 1)$, scattered outliers $x_{ij}^{cont} \sim \mathcal{N}(0, 100^2)$ and a wide cluster $x_{ij}^{cont} \sim \mathcal{N}(50, 10^2)$. We only contaminate the x-values and not the y-values, which creates bad leverage points. For comparison, we also construct classical contamination settings where we choose whole rows for contamination instead of cells. For these we choose the three contaminations $\mathbf{x}_i^{cont} \sim \mathcal{N}(\mathbf{50}, \Sigma)$, $\mathbf{x}_i^{cont} \sim \mathcal{N}(\mathbf{0}, 100^2 \cdot \Sigma)$ and $\mathbf{x}_i^{cont} \sim \mathcal{N}(\mathbf{50}, 10^2 \cdot \Sigma)$. Additionally, we also want to demonstrate that the shooting S-algorithm can deal with contamination in the response. From the clean data set, we select 1%, 2%, 5% and 10% of observations and generate their error terms as $e_{cont} \sim \mathcal{N}(50, \sigma^2)$ to create vertical outliers.

To compare the different estimators, we apply them to $R = 1000$ generated data sets. For each data set, we compute the mean squared error (MSE)

$$\text{MSE}(\hat{\boldsymbol{\beta}}) = \frac{1}{p} \sum_{j=1}^{p} \frac{1}{R} \sum_{r=1}^{R} (\hat{\beta}_j^{(r)} - \beta_j)^2.$$

Additionally, also the bias or the median squared error could be used as evaluation methods. We omit them as they are in line with the MSE.

The simulation results for cellwise contamination are displayed in Tables 1 and 2 for uncorrelated and correlated predictors, respectively. Table 3 gives the

[1] The signal-to-noise ratio equals $\frac{\sqrt{\boldsymbol{\beta}' \Sigma \boldsymbol{\beta}}}{\sigma}$.

Table 1 $n \cdot MSE$ of different estimators for cellwise contamination for all three contamination settings with $n = 100$, $p = 15$ and uncorrelated predictors

		$\epsilon = 0$	$\epsilon = 0.01$	$\epsilon = 0.02$	$\epsilon = 0.05$	$\epsilon = 0.1$
$x_{ij}^{cont} \sim \mathcal{N}(50, 1)$	LS	0.30	23.96	31.86	35.97	36.46
	S	0.36	1.08	12.55	32.44	36.36
	MM	0.33	0.48	0.88	18.35	34.52
	shooting S + BI	0.43	0.62	0.84	1.72	5.37
	shooting S + skH	0.55	0.65	0.80	2.02	5.61
$x_{ij}^{cont} \sim \mathcal{N}(0, 100^2)$	LS	0.30	22.41	30.82	36.16	36.67
	S	0.36	0.99	11.15	31.21	36.54
	MM	0.33	0.50	0.99	16.76	33.63
	shooting S + BI	0.43	0.62	0.86	2.00	8.94
	shooting S + skH	0.55	0.66	0.81	2.21	7.48
$x_{ij}^{cont} \sim \mathcal{N}(50, 10^2)$	LS	0.30	23.81	31.74	35.97	36.47
	S	0.36	1.08	12.69	32.49	36.37
	MM	0.33	0.49	0.92	18.50	34.41
	shooting S + BI	0.43	0.62	0.84	1.76	5.72
	shooting S + skH	0.55	0.65	0.81	2.05	5.83

results for rowwise contamination in the data set with correlated predictors. Table 4 illustrates the behavior of the estimators in presence of vertical outliers for correlated predictors. The standard errors around the reported results are smaller than 4% of the reported numbers in all tables. We omit the results for rowwise contamination and vertical outliers for uncorrelated predictors as they are comparable to the ones in the correlated case.

For uncorrelated predictors, Table 1 demonstrates the need of a new method that can deal with cellwise contamination. As well known, LS breaks down for any amount of contamination. But also the robust MM- and S-estimator have problems with larger amounts of cellwise contamination. As 2% of cellwise contamination corresponds in this setting to about $20 - 30\%$ of rowwise contamination[2], the ordinary S-estimator already breaks down. As we have chosen a breakdown point of 50% for MM, it can deal with slightly higher contamination. But for about 5% of cellwise contamination it also breaks down. In contrast, the shooting S-estimators can deal with much higher levels of cellwise contamination. They are reliable even for up to 10% of cellwise contamination, or around 80% of rowwise contamination. The two shooting S-estimators perform comparably in this setting.

Table 2 confirms for correlated predictors what is shown in Table 1 for uncorrelated ones. The only major difference is that for correlated predictors the shooting S-estimators already outperform the MM-estimator for 1% of cellwise contamination, even though the MM-estimator does not break down yet in this case.

[2] The expected value of the number of contaminated rows is $n(1 - (1 - \epsilon)^p)$ for a cellwise contamination level ϵ.

Table 2 $n \cdot MSE$ of different estimators for cellwise contamination for all three contamination settings with $n = 100$, $p = 15$ and predictors with correlation matrix Σ

		$\epsilon = 0$	$\epsilon = 0.01$	$\epsilon = 0.02$	$\epsilon = 0.05$	$\epsilon = 0.1$
$x_{ij}^{cont} \sim \mathcal{N}(50, 1)$	LS	1.28	35.28	39.47	35.46	36.02
	S	1.53	6.10	21.57	45.55	40.45
	MM	1.39	2.82	5.58	26.88	46.73
	shooting S + BI	1.70	2.28	2.84	3.55	6.20
	shooting S + skH	2.00	2.26	2.44	3.66	6.07
$x_{ij}^{cont} \sim \mathcal{N}(0, 100^2)$	LS	1.28	32.66	38.84	36.16	36.54
	S	1.53	5.60	19.17	43.82	40.40
	MM	1.39	2.93	5.66	25.74	44.23
	shooting S + BI	1.70	2.32	2.95	4.25	10.03
	shooting S + skH	2.00	2.29	2.53	4.04	7.74
$x_{ij}^{cont} \sim \mathcal{N}(50, 10^2)$	LS	1.28	34.98	39.24	35.47	36.04
	S	1.53	6.15	21.62	45.52	40.38
	MM	1.39	2.90	5.55	27.17	46.51
	shooting S + BI	1.70	2.28	2.86	3.63	6.67
	shooting S + skH	2.00	2.25	2.45	3.69	6.23

Table 3 $n \cdot MSE$ of different estimators for rowwise contamination for all three contamination settings with $n = 100$, $p = 15$ and predictors with correlation matrix Σ

		$\epsilon = 0$	$\epsilon = 0.01$	$\epsilon = 0.02$	$\epsilon = 0.05$	$\epsilon = 0.1$
$\mathbf{x}_i^{cont} \sim \mathcal{N}(\mathbf{50}, \Sigma)$	LS	1.28	53.62	53.96	54.72	54.80
	S	1.53	1.51	1.50	1.48	1.48
	MM	1.39	1.40	1.41	1.44	1.50
	shooting S + BI	1.70	1.66	1.63	1.63	1.66
	shooting S + skH	2.00	1.91	1.88	1.77	1.78
$\mathbf{x}_i^{cont} \sim \mathcal{N}(\mathbf{0}, 100^2\Sigma)$	LS	1.28	12.58	22.82	44.10	56.41
	S	1.53	1.57	1.58	1.75	2.27
	MM	1.39	1.43	1.47	1.58	1.79
	shooting S + BI	1.70	1.69	1.73	1.98	3.04
	shooting S + skH	2.00	1.95	1.94	1.98	2.48
$\mathbf{x}_i^{cont} \sim \mathcal{N}(\mathbf{50}, 10^2\Sigma)$	LS	1.28	56.43	56.28	55.85	49.89
	S	1.53	1.51	1.50	1.48	1.48
	MM	1.39	1.40	1.41	1.44	1.50
	shooting S + BI	1.70	1.66	1.65	1.67	1.76
	shooting S + skH	2.00	1.91	1.88	1.79	1.86

For rowwise contamination the situation is different (see Table 3). Here, as known, MM and S-estimation give excellent results. The shooting S-estimators give only slightly higher values of MSE compared to the ordinary S-estimator, indicating that the shooting S-estimators can cope with rowwise contamination as well. Nevertheless, as the shooting S-estimator has been developed for cellwise contamination, we do not advise its usage if there is only rowwise contamination present.

Table 4 $n \cdot MSE$ of different estimators for vertical outliers with $n = 100$, $p = 15$ and predictors with correlation matrix Σ

	$\epsilon = 0$	$\epsilon = 0.01$	$\epsilon = 0.02$	$\epsilon = 0.05$	$\epsilon = 0.1$
LS	1.28	51.40	97.69	234.23	438.65
S	1.53	1.51	1.50	1.48	1.48
MM	1.39	1.40	1.41	1.44	1.50
shooting S + BI	1.70	1.73	1.77	1.88	2.13
shooting S + skH	2.00	2.03	2.01	2.10	2.14

The shooting S-estimator can also cope with vertical outliers (see Table 4). It gives good results for all levels of contamination used here, although its MSE is slightly higher than for the S- and MM-estimators. The reason for the good performance of the shooting S-estimator is that the contamination in the response is present in the computation of each single coefficient $\hat{\beta}_j$. Robustness of the 'regression step' leads to small weights w_{ij} for all j.

We may conclude that the shooting S-estimator is the only considered regression estimator that can deal with cellwise contamination above 2% in our simulation setting. The estimator also gives good results in presence of vertical outliers. If there are no outliers, there is a slight loss in efficiency compared to the other robust estimators. In a rowwise contamination setting, we advise the use of the usual S- and MM-estimator.

5 Real Data

We evaluate the performance of the shooting S-estimator on real data sets and compare it to the LS, S- and MM-estimators. For all estimators, the tuning parameters are chosen as in Section 4. We choose the three data sets `Cars93`, `Auto` and `Boston`. When applying the shooting S-estimator, we declare a component of an observation, hence a cell in the data matrix, as an outlier if it gets a robustness weight below 0.5. If all components of an observation are flagged as outliers, we say that the whole observation is outlying.

The `Cars93` data, a selection of 1993 model cars, are included in the `R` package `MASS`. Omitting not fully observed data points, we are left with $n = 82$ observations. We fit the following model with $p = 14$ predictor variables of the `Cars93` data (for the definition of the variables, see Table 6 in the appendix)

$$\begin{aligned} PRICE =&\beta_0 + \beta_1 MPG.C + \beta_2 MPG.H + \beta_3 ENG.SIZE + \beta_4 HP \\ &+ \beta_5 RPM + \beta_6 REV.MILE + \beta_7 FUEL.TANK + \beta_8 LENGTH \\ &+ \beta_9 WHEELBASE + \beta_{10} WIDTH + \beta_{11} TURN + \beta_{12} REAR.SEAT \\ &+ \beta_{13} LUGGAGE + \beta_{14} WEIGHT + error. \end{aligned}$$

The shooting S-estimator using a biweight loss downweights seven observations as a whole and detects outlying cells for another 19 observations. In contrast, the MM-estimator downweights the observations corresponding to these outlying cells as a whole, thereby loosing information. This information loss is especially visible when looking, for example, at observation 46, which receives the weight 0 by the MM-estimator, while the shooting S-estimator with biweight loss assigns a weight of about 1 to all components except the first component, which receives weight 0.

The `Auto` data set is included in Stata and can be downloaded from `http://www.stata-press.com/data/r13/auto.dta` (see StataCorp, 2013). It consists of $n = 74$ fully observed sales of vintage 1978 automobiles in the United States (see Table 7 in the appendix). We fit the following model with $p = 8$ predictor variables

$$\begin{aligned} PRICE =& \beta_0 + \beta_1 MPG + \beta_2 HEADROOM + \beta_3 TRUNK + \beta_4 WEIGHT \\ &+ \beta_5 LENGTH + \beta_6 TURN + \beta_7 DISPLACE + \beta_8 GEAR + error. \end{aligned}$$

The shooting S-estimator with biweight loss downweights five observation as a whole and flags cells of another 17 observations as outliers. For instance, observations 12 ('Chevrolet Cavalier') and 13 ('Chevrolet Corsica') receive a weight of zero by MM and the ordinary S, while the shooting S-estimator using a biweight loss finds out that only component 2, the headroom, is outlying. Again, we conclude that the shooting S-estimator uses more information from the data than the MM-estimator or the ordinary S-estimator.

The third data set, the `Boston` housing data, originates from Harrison and Rubinfeld (1978) and has been extensively analyzed in the robust statistics literature. The data are available in the R package `mlbench` and contain various characteristics of houses, demographics, air pollution and geographical details on $n = 506$ census tracts in and nearby Boston. Table 8 in the appendix gives an overview of the definition of the variables ($p = 9$) in the model

$$\begin{aligned} \log(MEDV) =& \beta_0 + \beta_1 CRIM + \beta_2 NOX^2 + \beta_3 RM^2 + \beta_4 AGE + \beta_5 \log(DIS) \\ &+ \beta_6 TAX + \beta_7 PTRATIO + \beta_8 B + \beta_9 \log(LSTAT) + error. \end{aligned}$$

Belsley et al (1980) discovered outlying behavior of census tracts lying in central area of Boston, concentrated in three neighborhoods. Applying the shooting S-estimator using a biweight loss to the full data set, we get similar results. The shooting S-estimator declares the observations from the neighborhoods Back Bay ($365 - 370$), Beacon Hill (371- 373) and South Boston (394-406) as cellwise contaminated, with mainly the components corresponding to the variables `RM` and `AGE` indicated as outlying. The MM-estimator and the ordinary S-estimator downweight as a whole the observations of the neighborhoods Back Bay and Beacon Hill and half of the observations of South Boston, resulting in a loss of information.

For each of the three data sets, we randomly choose 4/5th of the observations and compute all estimates on this training data set. This we repeat $R = 500$ times and we compare the estimates on the training data sets $\hat{\boldsymbol{\beta}}^{(r)}$, for $r = 1, \ldots, R$, to the one computed on the full data set $\hat{\boldsymbol{\beta}}^{full}$. Adjusting

Table 5 Average norm distance (AND) for five estimators computed on three data sets and their contaminated versions

	observed data			contaminated data		
	Auto	Cars93	Boston	Auto	Cars93	Boston
LS	0.388	0.141	0.024	1.320	0.325	0.273
S	0.459	0.172	0.021	0.697	0.240	0.223
MM	0.282	0.213	0.022	0.346	0.243	0.179
shooting S + BI	0.607	0.217	0.033	0.251	0.228	0.152
shooting S + skH	0.574	0.186	0.039	0.658	0.169	0.138

for the different scales of the explanatory variables, we get what we call the Average Norm Distance (AND)

$$\mathrm{AND}(\hat{\boldsymbol{\beta}}) = \frac{1}{R}\sum_{r=1}^{R}\sqrt{\frac{1}{p}\sum_{j=1}^{p}(\hat{\beta}_j^{(r)} - \hat{\beta}_j^{full})^2 \frac{\mathrm{mad}(x_{1j},\ldots,x_{nj})^2}{\mathrm{mad}(y_1,\ldots,y_n)^2}}. \qquad (10)$$

A low value of AND is desired. Table 5 shows the results for all considered estimators on the three data sets. As pointed out by a referee, the AND criterion is a version of the Jackknife estimator of the variance, and it reflects the efficiency. Therefore, the low value of AND for the LS estimator is no surprise. The AND for the S- and MM-estimator are close to those of LS, and sometimes even slightly better. The shooting S-estimators have somehow larger values of the AND, but the loss in efficiency remains limited.

To investigate the robustness of the estimators, we randomly choose 5% of the cells of the data set and replace them with $x_{ij}^{cont} \sim \mathcal{N}(\hat{\mu}_j + 10\hat{\sigma}_j, \hat{\sigma}_j^2)$ where $\hat{\mu}_j$ and $\hat{\sigma}_j$ denote the median and MAD of the jth column of the design matrix, respectively. This we repeat $R = 500$ times and we compute the average norm difference as in (10), where $\hat{\boldsymbol{\beta}}^{(r)}$ are the estimates from the contaminated data and $\hat{\boldsymbol{\beta}}^{full}$ is the estimate on the original data. Table 5 gives the results. Now the AND measures the robustness of the estimators, and the LS estimator clearly gives the worst results. The shooting S-estimators, and in particular when using a biweight loss, give the best results. They deal better with cellwise contamination than the ordinary S- and MM-estimator.

We did not use a prediction error criterion to assess the performance of the different estimators. If the level of cellwise contamination is moderate to high, we expect that most observations contain contaminated cells. When forecasting, outlying components of an observation get full influence (which is not the case in estimation). Assessing the prediction error by cross-validation is then not reliable anymore, since the validation set contains too many observations with contaminated components. Using a robust cross-validation criterion, as a trimmed mean squared prediction error does not solve this problem, as far too many observations used for validation may have outlying components. Prediction for cell-wise contaminated observations is left as a topic for future research.

6 Conclusion

In this paper, we introduce a regression estimator applicable for cellwise contamination. It combines the ideas of ordinary regression S-estimation with the coordinate descent algorithm. Thereby the shooting S-estimator is able to use different weights for different components of an observation. In our simulations, it can deal with cellwise contamination up to 10%.

Furthermore, the shooting S-estimator can also be used as a diagnostic tool. After computation of the shooting S-estimate, the entries of the weight matrix w_{ij} help to distinguish between clean data and outliers, and even between cellwise and rowwise contamination. While high weights indicate a clean cell, low weights indicate contamination. If all components of the same observation get low weights, this means that all components are contaminated or that it is a vertical outlier.

The efficiency of the shooting S-estimator can be improved by using a *shooting MM-estimator* instead. To obtain a shooting MM-estimator, the simple S-estimation step inside the algorithm needs to be replaced with a simple MM-estimation. In order to explore this idea, the simulations of Section 4 were repeated for a shooting MM-estimator, using simple MM-estimation with 20% breakdown point and 95% efficiency at the normal distribution. Preliminary results indicate that (i) the shooting MM-estimator gave generally lower values for mean squared error in the simulations of Section 4 than the shooting S-estimator; (ii) especially for clean data and small amounts of contamination, the improvement of the shooting MM-estimator over the shooting S-estimator was clearly visible; (iii) the shooting MM-estimator outperformed the ordinary MM-estimator in any setting where cellwise contamination was present. However, further development of the shooting MM-estimator is necessary and is left for future research.

Another idea worth considering is the application of an imputation method after performing shooting S-regression. Cells that are flagged as outliers can be set as missing. On the data set containing missing values, regression can be performed (see Little, 1992).

Admittedly, our shooting S-estimator has problems with cellwise good leverage points, which are observations with large values in some single cells that do follow the regression model. The shooting S-estimator tends to flag the contaminated cells of the good leverage points as outliers when computing the starting values of the algorithm. However, if the data contain rowwise good leverage points, thus large values for all cells of observations that do follow the model, the shooting S-estimator behaves comparable to the other estimators (LS, S, MM) in our experiments.

As the shooting S-algorithm deals with each variable separately, it can also be applied to data sets with a small sample size and even if $n < p$. A suitable ρ-function for this setting may be the linear quadratic quadratic (lqq) function of Koller and Stahel (2011), as it has been shown to have high efficiency also for small sample sizes. When using the lqq-function in the simulation setups in Section 4, the results are comparable to the ones with the other ρ-functions used there.

Finally, the shooting S-estimator can be extended to a *penalized shooting S-estimator*. To the simple S-estimation in every variable, a penalty term $J(\beta)$ can be added. Possible choices for the penalty term are $J(\beta) = |\beta|$ or $J(\beta) = \beta^2$. The penalized version of the shooting S-estimator could be very useful in high-dimensional settings.

Acknowledgements We gratefully acknowledge support from the GOA/12/014 project of the Research Fund KU Leuven. We thank the referees for their constructive comments, and in particular the third anonymous referee who corrected some flaws in the first version of the paper and who made many suggestions for improving the write up of the paper.

A APPENDIX - Description of Variables for Real Data Examples

Table 6 Variables of the `Cars93` data

Name	Description
$PRICE$	Midrange Price (in \$1,000)
$MPG.C$	City MPG (miles per US gallon by EPA rating)
$MPG.H$	Highway MPG (miles per US gallon by EPA rating)
$ENG.SIZE$	Engine displacement size in liters
HP	Maximum horsepower
RPM	Revolutions per minute at which maximum horsepower is achieved
$REV.MILE$	Number of revolutions of the engine needed for car to travel one mile in its highest gear
$FUEL.TANK$	Capacity of the fuel tank in US gallons
$LENGTH$	Length of the car in inches
$WHEELBASE$	Size of the wheelbase in inches
$WIDTH$	Width of the car in inches
$TURN$	U-turn space in feet
$REAR.SEAT$	Rear seat room in inches
$LUGGAGE$	Luggage capacity in cubic feet
$WEIGHT$	Weight of the car in pounds

Table 7 Variables of the `Auto` data

Name	Description
$PRICE$	Price in US-dollars
MPG	Milage
$HEADROOM$	Head room in inches
$TRUNK$	Trunk space in cubic feet
$WEIGHT$	Weight of the car in pounds
$LENGTH$	Length of the car in inches
$TURN$	U-turn space in feet
$DISPLACE$	Displacement in cubic inches
$GEAR$	Gear ratio

Table 8 Variables of the `Boston` data

Name	Description
$MEDV$	Median value of owner-occupied homes in USD 1000's
$CRIM$	Per capita crime rate by town
NOX	Nitric oxides concentration in parts per 10 million
RM	Average number of rooms per dwelling
AGE	Proportion of owner-occupied units built prior to 1940
DIS	Weighted distance to five Boston employment centres
TAX	Full-value property-tax rate per USD 10,000
$PTRATIO$	Pupil-Teacher ratio by town
B	Proportion of black population
$LSTAT$	Percentage of lower status population

References

Alfons A, Croux C, Gelper S (2013) Sparse least trimmed squares regression for analyzing high-dimensional large data sets. The Annals of Applied Statistics 7(1):226–248

Alqallaf F, Van Aelst S, Yohai V, Zamar R (2009) Propagation of outliers in multivariate data. The Annals of Statistics 37(1):311–331

Belsley D, Kuh E, Welsch R (1980) Regression Diagnostics: Identifying Influential Data and Source of Collinearity. John Wiley & Sons, New York

Brown P (1982) Multivariate calibration. Journal of the Royal Statistical Society, Series B 44(3):287–321

Friedman J, Hastie T, Hofling H, Tibshirani R (2007) Pathwise coordinate optimization. The Annals of Applied Statistics 1(2):302–332

Fu W (1998) Penalized regressions: The bridge versus the lasso. Journal of Computational and Graphical Statistics 7(3):397–416

Harrison D, Rubinfeld D (1978) Hedonic housing prices and the demand of clean air. Journal of Environmental Economics and Management 5(1):81–102

Koller M, Stahel W (2011) Sharpening Wald-type inference in robust regression for small samples. Computational Statistics and Data Analysis 55(8):2504–2515

Little R (1992) Regression with missing X's: A review. Journal of the American Statistical Association 87(420):1227–1237

Maronna R, Martin R, Yohai V (2006) Robust Statistics, 2nd edn. John Wiley & Sons, Hoboken, New Jersey

Rousseeuw P, Leroy A (1987) Robust Regression and Outlier Detection. John Wiley & Sons, Hoboken, New Jersey

StataCorp (2013) Stata: Release 13. Statistical Software. Stata Press, College Station, Texas

Tseng P (2001) Convergence of a block coordinate descent method for nondifferentiable minimization. Journal of Optimization Theory and Applications (3):475–494

Van Aelst S, Vandervieren E, Willems G (2010) Robust principal component analysis based on pairwise correlation estimators. In: Lechevallier Y, Saporta G (eds) COMPSTAT 2010: Proceedings in Computational Statistics, Physika-Verlag, Heidelberg, pp 1677–1684

Van Aelst S, Vandervieren E, Willems G (2011) Stahel-Donoho estimators with cellwise weights. Journal of Statistical Computation and Simulation 81(1):1–27

Van Aelst S, Vandervieren E, Willems G (2012) A Stahel-Donoho estimator based on huberized outlyingness. Computational Statistics and Data Analysis 56(3):531–542