Computational Statistics and Data Analysis 147 (2020) 106944

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Cellwise robust M regression

P. Filzmoser [a,*], S. Höppner [b], I. Ortner [c], S. Serneels [d], T. Verdonck [e]

[a] Institute of Statistics and Mathematical Methods in Economics, TU Wien, Wiedner Hauptstraße 8-10, 1040 Vienna, Austria
[b] Department of Mathematics, KU Leuven, Leuven, Belgium
[c] Applied Statistics GmbH, Vienna, Austria
[d] Aspen Technology, Bedford, Massachusetts, MA01730, USA
[e] Department of Mathematics, University of Antwerp, Antwerp, Belgium

ARTICLE INFO

Article history:
Received 4 December 2019
Received in revised form 11 February 2020
Accepted 1 March 2020
Available online 4 March 2020

Keywords:
Cellwise robust statistics
Cellwise robust M regression
Cellwise outliers
Detecting deviating cells
Linear regression

ABSTRACT

The cellwise robust M regression estimator is introduced as the first estimator of its kind that intrinsically yields both a map of cellwise outliers consistent with the linear model, and a vector of regression coefficients that is robust against vertical outliers and leverage points. As a by-product, the method yields a weighted and imputed data set that contains estimates of what the values in cellwise outliers would need to amount to if they had fit the model. The method is illustrated to be equally robust as its casewise counterpart, MM regression. The cellwise regression method discards less information than any casewise robust estimator. Therefore, predictive power can be expected to be at least as good as casewise alternatives. These results are corroborated in a simulation study. Moreover, while the simulations show that predictive performance is at least on par with casewise methods if not better, an application to a data set consisting of compositions of Swiss nutrients, shows that in individual cases, CRM can achieve a much higher predictive accuracy compared to MM regression.

1. Introduction

Linear regression is one of the most frequently studied problems in the statistical sciences. It is well known that the least squares estimator fulfills several optimality criteria under normal distribution assumptions, a result that goes all the way back to Gauß (Gauss, 1826). Likewise, it is well known that the least squares estimator is not optimal when data deviate from these assumptions. A lot of attention has been attributed to developing methods that still yield sensible regression parameters in the presence of casewise deviations. Such casewise deviations may originate from a fraction ϵ of the data having been generated from a different distribution (outliers), or the data satisfying the linear model with a non-normal error term, such as a Cauchy or Student's t. In these cases, robust linear regression methods generally outperform their least squares counterpart. Many different approaches to casewise robust regression have been proposed, a good overview of which can be found in reference works (Huber and Ronchetti, 2009; Maronna et al., 2006, 2019; Rousseeuw and Leroy, 1987).

In the bulk of the literature on robust statistics, robustness is considered to be robustness against entire cases that do not satisfy model assumptions. For a univariate predictor $\mathbf{x} = (x_1, \ldots, x_n)^T$, this approach is plausible because it corresponds to individual elements x_i either fitting the assumptions or not. Conversely, assuming that outliers are

* Corresponding author.
E-mail addresses: P.Filzmoser@tuwien.ac.at (P. Filzmoser), sebastiaan.hoppner@kuleuven.be (S. Höppner), irene.ortner@applied-statistics.at (I. Ortner), Sven.Serneels@aspentech.com (S. Serneels), Tim.Verdonck@uantwerpen.be (T. Verdonck).

https://doi.org/10.1016/j.csda.2020.106944
0167-9473/

complete observations of a multivariate predictor, thus multivariate observations where each entry in the observation vector is considered as an outlier, may not correspond to reality. In real life, the predictor matrix often consists of single predictors that are measurements of different physical entities, which need not generate outliers simultaneously. Imagine, for example, each column being a sensor in a manufacturing plant. Whereas it is viable to assume multivariate interplay between these sensors to be present under normal operating conditions, each of these sensors may break down independently and therefore, generate outliers individually. Another example would be gene expression in microarray data, and there are many more. Discarding whole cases in these (and other) practical situations can cause a significant loss of information in the estimation procedure, which just like harsh downweighting of entire outliers, can be surmised to increase estimation variance.

In the light of the above, to make maximal use of the non-contaminated portion of the data, in practice it is often preferable to detect outliers on a cellwise basis instead of casewise. This means that single entries (cells) in the data matrix are considered as potential outliers, and not necessarily a whole row (observation). Up to today, this usually implies that outlier detection is done as a separate step *before* the remainder of the analysis. However, any outlier is only outlying with respect to a model and therefore, such a preliminary outlier detection c.q. correction step may distort the data in a way that is inconsistent with the model. There is a large gap yet to be covered in method development on cellwise robust techniques: methods that allow to detect and correct for deviating cells in a single model consistent way. Cellwise robust regression is still a nascent field of research. In this paper, a new cellwise robust M regression estimator (CRM) is proposed. In one run, it allows to estimate regression coefficients that are robust against cellwise and casewise outliers, while also providing a map of the deviating cells. The option to construct the estimator as a cellwise robust M regression as opposed to alternative paths, such as MCD regression (Rousseeuw, 1984), comes from the observation that robust M regression estimators have proven to yield a very good trade-off between efficiency and robustness in simulations and applications in fields as diverse as quantitative structure–property relationships (QSPR) (Serneels et al., 2006), gravimetry (Hu et al., 2017), finance (Guerard, 2016), chemometrics (Hoffmann et al., 2015), analytical chemistry with applications to e.g. analysis of archæological glass (Serneels et al., 2005) and meteorite samples (Hoffmann et al., 2016), as well as estimation of shaping coefficients for futures trading in the electricity markets (Leoni et al., 2018). Note though, that S-regression has also proven a valid path in this context (Öllerer et al., 2016).

Motivated by this assumption, in this manuscript the *cellwise robust M* (CRM) regression estimator is introduced. It consists of an iteratively reweighted least squares procedure, starting with weights derived from highly robust estimates, that both compensate for casewise vertical outliers and leverage points. Within each iteration, the SPADIMO (Debruyne et al., 2019) procedure is applied, detecting the cells that contribute most to outlyingness. The re-weighting scheme is then adapted to only downweight outlying cells. The resulting method thereby can deliver a highly robust estimate of regression coefficients (and intercept), and in a model consistent way, yield cellwise outlier detection. Because not as much information in the data is discarded, the method should be more efficient than a casewise robust estimator.

The article is organized as follows. In Section 2, the CRM algorithm is described in detail. Section 3 presents a simulation study comparing CRM to different approaches in terms of efficiency, as well as in terms of its capability to detect and downweight the correct set of outlying cells. In Section 4, the method is applied to a compelling example. Finally, Section 5 concludes.

2. The CRM algorithm

2.1. Background

The target of this section is to propose an estimator for the linear model that is robust against cellwise outliers, and as a by-product, yields a map of the detected outlying cells.

Let $\mathbf{X} \in \mathbb{R}^{n\times p}$ be a predictor matrix consisting of n cases of p predictor variables (or, if an intercept is considered, $p-1$ predictors, and the first column with ones for the intercept) and let $\boldsymbol{\beta} \in \mathbb{R}^p$ be a fixed, true vector of regression coefficients. Then, in the linear model, n cases of a univariate dependent variable $\mathbf{y} \in \mathbb{R}^n$ relate to the predictors as

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{1}$$

where the entries of $\boldsymbol{\varepsilon}$ are independent and identically distributed (i.i.d.) and where $E(\boldsymbol{\varepsilon}) = \mathbf{0}$ and $Cov(\boldsymbol{\varepsilon}) = \sigma^2\mathbf{I}_n$. As signaled before, while the least squares estimator may be optimal under normality assumptions, robust regression methods should be the estimators of choice when outliers are expected to be in the data. Several classes of robust regression estimators exist (see, e.g., Rousseeuw and Leroy, 1987; Maronna et al., 2006). Performance analysis of robust estimators has several facets, but most prominently it comes down to analyzing how well the estimators perform in trading off robustness for statistical efficiency. Estimators that can resist a high fraction of outliers in the data, tend to have a higher variance than the corresponding maximum likelihood estimator, but that loss in efficiency need not be dramatic. Along other classes of methods, MM estimators (Yohai, 1987) are known to perform well in terms of the robustness–efficiency trade-off and have, for that reason, been incorporated into mainstay implementations of robust regression, such as the function lmrob() in the R package robustbase (Maechler et al., 2018).

To understand how MM estimators work, let us revert to least squares. By definition, least squares minimizes a loss function of squared residuals. Consider a given estimator $\hat{\boldsymbol{\beta}}$ of $\boldsymbol{\beta}$. Then the ith regression residual is defined as

$r_i(\hat{\boldsymbol{\beta}}) = y_i - \mathbf{x}_i^T\hat{\boldsymbol{\beta}}$, where $\mathbf{y} = (y_1, \ldots, y_n)^T$ and $\mathbf{x}_i$ represents the ith row of the data matrix $\mathbf{X}$, for $i = 1, \ldots, n$. The least squares estimator of the regression coefficients is given by the minimization problem

$$\hat{\boldsymbol{\beta}}_{\mathrm{LS}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_i r_i(\boldsymbol{\beta})^2. \tag{2}$$

A more general definition is

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \sum_i \rho\left(\frac{r_i(\boldsymbol{\beta})}{\hat{\sigma}}\right), \tag{3}$$

where $\hat{\sigma}$ is a robust scale estimator of the residuals, and $\rho(r)$ is a function that is approximately quadratic for small (absolute) r, but increases more slowly than r^2 for larger values of r. Moreover, the inclusion of $\hat{\sigma}$ allows to get the same result if the response is rescaled. This definition (3) is referred to as the class of M-estimators (Huber and Ronchetti, 2009). Not all choices of ρ have practical relevance, but when diligently chosen, M-estimators can have a bounded influence regarding deviating, or even erratic values in the response, and therefore they can be robust against vertical outliers. The robustness properties of the resulting estimator derive back to ρ, or more precisely, to its derivative $\psi = \rho'$. More particularly, common practicable choices leading to robust estimators for ψ, such as the Huber, Hampel redescending or Hampel–Rousseeuw hyperbolic tangent functions (Hampel et al., 1986), depend on a set of parameters. The resulting robustness properties then become a function of the corresponding parameters. In this paper, the Hampel redescending function is chosen, with a similar motivation as to why the MM estimation path is pursued. The Hampel function, in its reweighting representation, is given by:

$$w_H(r) = \begin{cases} 1 & |r| \le Q_1 \\ \frac{Q_1}{|r|} & Q_1 < |r| \le Q_2 \\ \frac{Q_3 - r}{Q_3 - Q_2}\frac{Q_1}{|r|} & \text{if} \quad Q_2 < |r| \le Q_3 \\ 0 & Q_3 < |r| \end{cases}. \tag{4}$$

It depends on a set of three parameters Q_1, Q_2 and Q_3. When applied to regression residuals, which for standardized data can be assumed to be standard normally distributed, sensible values for the parameters are the 0.95, 0.975 and 0.999 quantiles of the standard normal distribution.

It can be shown that in practice, calculating M-estimators directly through optimizing (3) is equivalent to running an iteratively re-weighted least squares (IRLS) procedure (Green, 1984). However, the resulting estimator will in general only be robust against outliers in the response, i.e. vertical outliers. In order to achieve robustness also against outliers in the explanatory variables (leverage points), it is important to select a robust starting estimator $\hat{\boldsymbol{\beta}}_0$. This can be done by taking a highly robust but inefficient S-estimator (Rousseeuw and Yohai, 1984), combined with a robust M-scale estimator $\hat{\sigma}$ (Huber and Ronchetti, 2009). The resulting robust *MM* estimator inherits the 50% breakdown point of the S-estimator, and has tunable efficiency (see Maronna et al., 2006, for more details).

In this paper, robust MM estimators are now being generalized to cellwise robustness. In order to achieve this, the IRLS procedure will need a way to be able to detect which cells are outlying. Exactly for this purpose, the method of Sparse Directions of Maximal Outlyingness (SPADIMO) (Debruyne et al., 2019) has recently been developed. SPADIMO is a method that identifies which variables contribute most to a case being detected as an outlier. By incorporating SPADIMO into the IRLS reweighting scheme, and only downweighting cells flagged by SPADIMO, the method will be cellwise robust.

2.2. The algorithm

The overarching algorithm described in this section can be seen as a way to convey cellwise robustness properties to any given robust regression method. However, robust regression methods being significantly different by construction, many details in the algorithm need to be adapted to the specific regression method. As outlined before, we have opted to develop the algorithm specifically as a cellwise extension to robust MM regression, inspired by the good robustness–efficiency tradeoff which have been shown in theory, simulation and practical applications (Maronna et al., 2006).

MM regression estimators consist of two steps: at first, a highly robust initial estimate is calculated, which conveys its high breakdown point to the entire procedure. Then, the highly robust initial estimate is used as a plug-in estimator for an M-estimator, which in practice means that the initial estimate is used as a starting point for an algorithm to achieve higher efficiency by iterative reweighting. How this concept can be used to obtain an efficient and highly robust cellwise regression method is presented below in detail, and an overview of the essential steps is given in Algorithm 1.

Initial outlier detection. Prior to starting the algorithm, the data should be centered and scaled. Since the data may still contain both cellwise and casewise outliers at that point, the preprocessing should be done robustly, with estimators that have a 50% asymptotic breakdown point. Good choices for centering and scaling the data robustly would be the L_1 median and the Q_n scale estimator (Rousseeuw and Croux, 1993), respectively, but viable alternatives to these choices exist. Several algorithms to compute the L_1 median are available, and a good comparison is given in Fritz et al. (2012).

In the spirit of MM estimation, an initial, highly robust regression estimator is used to identify suspected casewise outliers. Here we use the MM estimator as starting point, but a good alternative would be the LTS estimator, which is also known to be an efficient and robust regression estimator (Rousseeuw and Van Driessen, 2006). Based on the robust MM regression estimator, observations are flagged as casewise outliers if their absolute standardized residuals exceed the 95% quantile of the standard normal distribution. For each of the casewise outliers, it now has to be determined if they truly are casewise outliers, or if there is a subset of cells in them which make them outlying. In order to investigate which variables contribute most to the outlyingness of the casewise outliers, the SPADIMO (Debruyne et al., 2019) algorithm is applied. For those cases that contain cellwise outliers, outlying variables are imputed as if they were missing cells (see Algorithm 2). To impute the values in the outlying cells, the two nearest neighbors are detected based on the *clean* cells in the case. This means that, when a set $\mathcal{C}$ of $q < p$ variables have been detected as cellwise outliers in the case, the two nearest neighbors are determined in the $\mathbb{R}^{p-q}$ variate space spanned by the variables in $\{1, \ldots, p\}\backslash\mathcal{C}$. The nearest neighbor search is only carried out in the subset of cases that are not outlying, i.e. have case weights equal to one. The cellwise outliers in the case under consideration are now being imputed with the corresponding column means of the two nearest neighbors. This imputation procedure generates modified cases with smaller residuals, which increases their case weights and by consequence, the valuable information in the non-outlying variables contributes to the model. This imputation step is not only a part of the algorithm initiation, but will also take place in each IRLS step. As such, the initial outlier detection step yields a first value of estimates for the regression coefficients $\hat{\boldsymbol{\beta}}$, as well as a first set of weighted and imputed data $\mathbf{X}_\omega$ and $\mathbf{y}_\omega$ for the explanatory variables and the response, respectively.

Iterative modeling and outlier detection. Once an initial highly cellwise robust estimate of the regression coefficients has been obtained, a more efficient estimate can be found by using this estimate as a starting value into an iteratively reweighted least squares (IRLS) routine. Starting from the initial $\hat{\boldsymbol{\beta}}$, $\mathbf{X}_\omega$ and $\mathbf{y}_\omega$, the first IRLS update will consist of least squares regression estimates based on the weighted data $\mathbf{X}_\omega$ and $\mathbf{y}_\omega$. In each step, the residuals are calculated, and casewise outliers are detected based on the magnitude of the residuals. For those cases flagged as outliers, variables contributing to outlyingness are found by SPADIMO, and for the cases that are not entirely outlying, the outlying cells are imputed as before. Now the residuals can be recalculated based on the newly imputed data and a new set of weights is computed. The procedure continues until the estimated regression coefficients stabilize.

The complete algorithm is as follows:

- Apply robust regression (e.g. MM regression) on the original observations $\mathbf{x}_i$ and y_i, for $i = 1, \ldots, n$, to obtain the initial estimator $\hat{\boldsymbol{\beta}}$.
- Run Algorithm 1, starting with the original observations and the initial regression estimator.
- Run Algorithm 1, starting with the resulting weighted and imputed data $\mathbf{X}_\omega$ and $\mathbf{y}_\omega$ from the previous point, and with the least squares estimator $\hat{\boldsymbol{\beta}}$ from a regression using these data.
- Run Algorithm 1 with the weighted data as a result from the previous point, and the corresponding least squares estimator, and repeat until the mean absolute difference of the subsequent regression estimates is smaller than a tolerance bound (e.g. 0.01).

The R code implementations of CRM and SPADIMO are available in the R package crmReg at github.com/SebastiaanHoppner/CRM.

Algorithm 1: CRM Iteratively reweighted least squares algorithm.

1. Calculate residuals based on the estimator $\hat{\boldsymbol{\beta}}$:
$$r_i = y_i - \mathbf{x}_i^T \hat{\boldsymbol{\beta}} \quad \text{for } i \in \{1, \ldots, n\}$$
2. Detect outliers as cases that satisfy
$$\frac{|r_i|}{c\,\mathrm{med}_j|r_j|} > z_{0.95},$$
where $c = 1.4826$ for consistency of the MAD, and $z_{0.95}$ is the 0.95 quantile of the standard normal distribution.
3. For each outlying case:
   - Apply SPADIMO and obtain outlying variables.
   - If not all variables contribute to outlyingness: Impute values in outlying variables as in Algorithm 2.
   - Denote the newly imputed data matrix by $\tilde{\mathbf{X}}$.
4. Update residuals
$$\tilde{r}_i = y_i - \tilde{\mathbf{x}}_i^T \hat{\boldsymbol{\beta}} \quad \text{for } i = 1, \ldots, n.$$
5. Calculate case weights by the Hampel weight function (4),
$$\omega_i = w_H\left(\frac{|\tilde{r}_i|}{\mathrm{cmed}_j|\tilde{r}_j|}\right)$$
with c as in Step 2.
6. Let $\boldsymbol{\Omega} = \mathrm{Diag}(\sqrt{\omega_1}, \ldots, \sqrt{\omega_n})$ be a diagonal matrix with the case weights as diagonal elements.

Update the (imputed) data as
$\mathbf{X}_\omega = \boldsymbol{\Omega}\tilde{\mathbf{X}}$ and $\mathbf{y}_\omega = \boldsymbol{\Omega}\mathbf{y}$.

Algorithm 2: CRM imputation algorithm.

1. Let i be the index of an outlying case $\mathbf{x}_i$.
2. Let $\mathcal{C}$ be the set of $q < p$ variables detected as cellwise outliers in $\mathbf{x}_i$.
3. Detect the two nearest neighbors $\mathbf{x}_{k_1}$ and $\mathbf{x}_{k_2}$ of the outlier $\mathbf{x}_i$ in the subspace $\{1, \ldots, p\}\backslash\mathcal{C}$ and only among observations $\mathbf{x}_j$ with $\omega_j = 1$.
4. Impute outlying cells $\tilde{x}_{iq} = (x_{k_1 q} + x_{k_2 q})/2$ with $q \in \mathcal{C}$.

3. Simulation study

Cellwise robust estimation is a fairly recent development in the statistical sciences. Up to our knowledge, there is no report of a cellwise robust M-type regression estimator. With an emphasis on cellwise outlier detection, the *Detecting Deviating Data Cells* (DDC) method has been proposed (Rousseeuw and Vanden Bossche, 2018). At this point, it is noted that DDC has been designed with the purpose to yield reliable cellwise outlier detection, *even when* >50% *of the cases contain outlying cells*. The CRM method proposed here will not be robust against contamination of more than half of the data. While it does not offer the latter advantage, it does yield model consistent cell weights in combination with an increased statistical efficiency when compared to casewise robust regression methods.

In this simulation study, the performance of CRM applied to the robust coefficient estimator of an MM regression is compared to conventional MM regression, MM regression combined with DDC, ordinary least squares (OLS) regression and OLS regression combined with DDC. The simulation study establishes that CRM, as a method that intertwines cellwise robustness properties with estimating regression coefficients for the linear model in a model consistent way, significantly outperforms application of a model agnostic detection method for cellwise outliers (DDC), followed by either a classical or robust regression estimator.

3.1. Simulation setting

The data for the simulation study are generated from a p-dimensional multivariate normal distribution with center $\boldsymbol{\mu} = (0, \ldots, 0)^T$ and covariance matrix $\boldsymbol{\Sigma}$. The covariance matrix is a matrix of zeros, with ones in the diagonal and 0.5 in the first off-diagonal, so $\boldsymbol{\Sigma}_{i,i} = 1$ for $i = 1, \ldots, p$, $\boldsymbol{\Sigma}_{j,j+1} = \boldsymbol{\Sigma}_{j+1,j} = 0.5$ for $j = 1, \ldots, p-1$ and $\boldsymbol{\Sigma}$ is zero elsewhere. The number of variables is set to $p = 50$ and $n = 400$ cases are generated, resulting in the data matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$.

Let $\boldsymbol{\beta}$ be a vector of length p of random values from a standard normal distribution, normalized to length 10 and the intercept $\beta_0 = 10$. The error term $\boldsymbol{\epsilon}$ is a vector of length n of random values from a normal distribution with mean 0 and standard deviation 0.5.

Then, the response is generated for clean data as follows:

$$\mathbf{y} = \mathbf{1}_n \beta_0 + \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{5}$$

so the clean data consists of $(\mathbf{y}, \mathbf{X})$ and the regression coefficients are $(\beta_0, \boldsymbol{\beta})$. A pairwise scatterplot of the response variable $\mathbf{y}$ and the first four predictor variables in $\mathbf{X}$ is shown in Fig. 1.

3.2. Adding contamination

Contamination is added to the data matrix $\mathbf{X}$, and the contaminated matrix is denoted as $\mathbf{X}^c$. For the contamination we randomly select a fraction of $r = 5\%$ of the observations in $\mathbf{X}$, so $r \cdot n = 20$ rows of $\mathbf{X}$ are randomly selected. These 20 observations will be contaminated and are called casewise outliers. Here, the fraction of contamination is fixed; note that in Section 3.7, the effect of the fraction of contamination will be investigated. Let $I^c \subset \{1, \ldots, n\}$ denote the random subset of 20 selected case indices. To generate cellwise outliers, for each selected case $i \in I^c$, $\check{r} = 10\%$ of the predictor variables are randomly picked. So for each casewise outlier, $\check{r} \cdot p = 5$ randomly selected cells will be contaminated. For each $i \in I^c$, let $J_i^c \subset \{1, \ldots, p\}$ denote the subset of 5 selected variable indices. The total number of contaminated cells in $\mathbf{X}^c$ will be $r \cdot \check{r} \cdot n \cdot p = 100$.

Cellwise contamination in variable j is achieved by adding to its mean value $\bar{x}_j$, $k = 6$ times the standard deviation s_j of variable j plus a random value e of the standard normal distribution. The contaminated matrix is $\mathbf{X}^c$ with

$$x_{ij}^c = \bar{x}_j + k s_j + e = \bar{x}_j + k \sqrt{\frac{1}{n-1} \sum_{l=1}^{n} (x_{lj} - \bar{x}_j)^2} + e$$

for all $i \in I^c$ and $j \in J_i^c$. The contaminated data consists of $(\mathbf{y}, \mathbf{X}^c)$ where a casewise outlier is considered an observation which has contaminated cells. Fig. 2 shows a pairwise scatterplot of the response variable $\mathbf{y}$ and the first four predictor variables in $\mathbf{X}^c$. The casewise outliers are in red and the uncontaminated cases are in blue.

Section 3.6 will also contain simulation results where k is varied within a certain range.

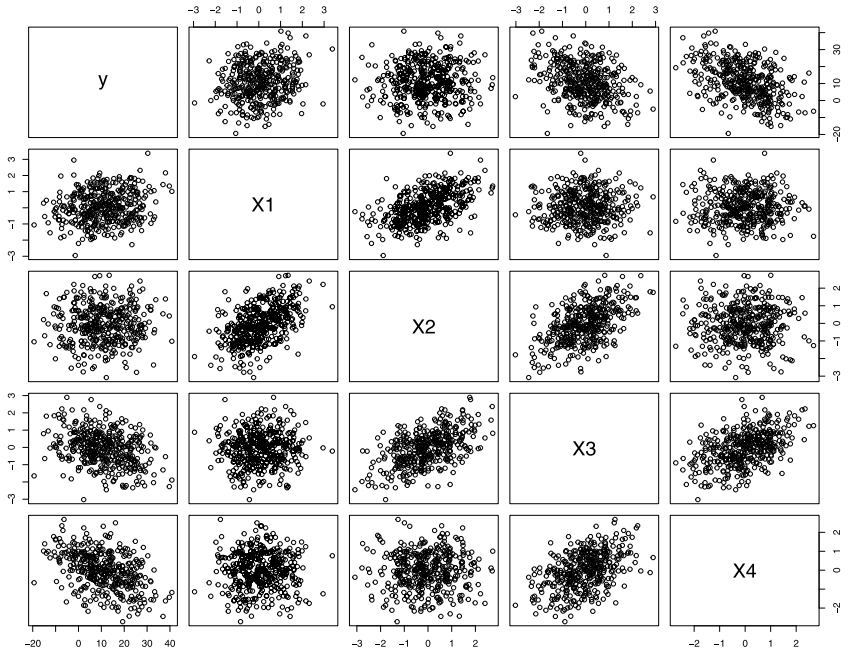

Fig. 1. Pairwise scatterplot of the response variable **y** and the first four predictor variables in **X**.

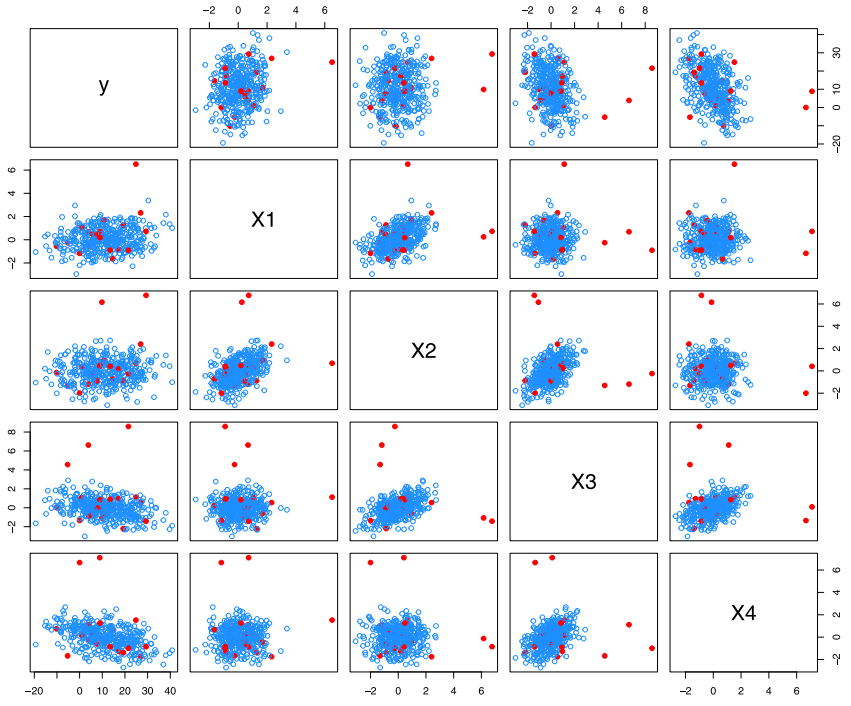

Fig. 2. Pairwise scatterplot of the response variable **y** and the first four predictor variables in $\mathbf{X}^c$. The casewise outliers are in red and the uncontaminated cases are in blue. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

3.3. Regression methods

The following linear regression methods have been fit to the contaminated data $(\mathbf{y}, \mathbf{X}^c)$: CRM (with MM regression as a starting estimate), simple MM regression and OLS regression. The following parameter settings for the CRM regression

estimation were used: the maximal number of iterations was set to 100, the relative tolerance for converging the regression coefficients was set to 0.01, the outlyingness factor for SPADIMO was set to 1.5; SPADIMO sparsity was allowed to vary as $\eta \in \{0.1, 0.2, \ldots, 0.9\}$ (Debruyne et al., 2019). The authors also suggest these settings as the default values in the R implementation.

For the methods that are sequential combinations of DDC with regression methods, the workflow goes as follows. At first, the Detect Deviating Cells (DDC) method is applied to the contaminated data matrix $\mathbf{X}^c$, which returns a DDC-imputed matrix, further denoted $\mathbf{X}^{DDC}$. Then, MM and OLS regression are fit to $(\mathbf{y}, \mathbf{X}^{DDC})$.

3.4. Evaluation

Four different criteria are assessed to evaluate the relative performance of the three approaches.

At first, the most obvious performance criterion for any regression method is predictive performance on independent test data. To evaluate the prediction performance, the mean squared error of prediction (MSEP) is calculated over the set of uncontaminated cases:

$$\text{MSEP} = \frac{1}{n_{clean}} \sum_{i \in I} (\hat{y}_i - y_i)^2 \tag{6}$$

where I contains the indices of clean, uncontaminated cases (I is the complementary set of I^c) and n_{clean} is the number of uncontaminated cases.

Secondly, it is interesting to know how much the individual regression coefficients, estimated by the three methods, deviate from the truth. To assess bias for the individual regression coefficients, the mean absolute error (MAE)

$$\text{MAE} = \frac{1}{p} \sum_{j=1}^{p} |\hat{\beta}_j - \beta_j| \tag{7}$$

is reported.

CRM and DDC each generate an imputed matrix of $\mathbf{X}^c$, denoted as $\mathbf{X}^{imp}$. In the case of DDC, $\mathbf{X}^{imp} = \mathbf{X}^{DDC}$. It is very informative to compare how close to the true values each of these methods comes when imputing the cellwise outliers. We report the performance of each imputed matrix $\mathbf{X}^{imp}$ as the root mean squared error of imputation (RMSEI) between the simulated uncontaminated matrix $\mathbf{X}$ and imputation of CRM and DDC:

$$\text{RMSEI}(\mathbf{X}^{imp}, \mathbf{X}) = \sqrt{\frac{1}{np} \sum_{i=1}^{n} \sum_{j=1}^{p} (x_{ij}^{imp} - x_{ij})^2} \tag{8}$$

Finally, it is an interesting question to investigate the quality of identification of cellwise outliers by CRM. The latter is reported as:

- the *recall* (also called hit rate or true positive rate): of all cellwise outliers, how many have actually been detected as outliers (and therefore imputed) by CRM
- the precision: of all cells that were flagged as outliers by CRM, how many actually were cellwise outliers

3.5. Results

In what follows, the simulation results will be shown, illustrating the results according to the four evaluation criteria described in the previous section. Each result reported is the aggregate across hundred repeats. These aggregates are illustrated as boxplots in Figs. 3–5. The average result for each method is printed at the bottom and the best result is shown in bold.

From Fig. 3, one can derive that DDC based imputation as a preprocessing step to regression methods does not improve predictive performance. In fact, just applying least squares regression without DDC preprocessing predicts better. However, due to the presence of outliers, both robust regression methods, CRM and MM, clearly outperform least squares. In terms of predictive power, Fig. 3 is reassuring in the sense that CRM performs equally well as the robust benchmark method, MM regression.

Regarding bias in the regression coefficients, Fig. 4 shows that, not unexpectedly, the OLS regression coefficients are most biased in the presence of cellwise outliers. However, again applying DDC as a preprocessing step prior to either OLS or MM regression, performs much worse than casewise or cellwise robust M regression. The MAE for CRM regression is slightly smaller than that of plain MM regression, and thus CRM has a slightly higher statistical efficiency than its casewise counterpart.

The subplots in Fig. 5 illustrate CRM's performance at imputing the true values for the cellwise outliers. In this respect, CRM beats DDC by a wide margin (left subplot). This is in line with CRM having a better predictive performance and a lower bias in the regression coefficients. Essentially, this tells us that much of CRM's superior performance shown in Figs. 3 and 4 can be attributed to it yielding a much more truthful intermediate imputation. Finally, the right panel in

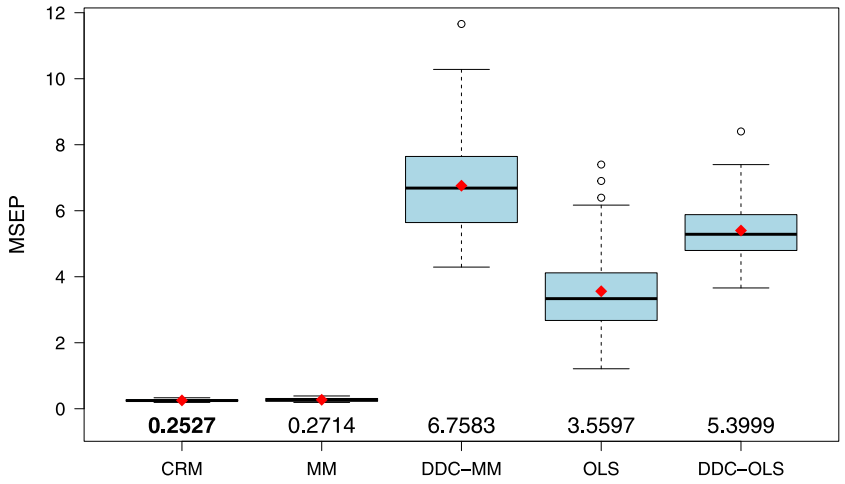

Fig. 3. Boxplot of MSEP for each of the regression methods. The average result for each method is printed at the bottom where the best result is shown in bold.

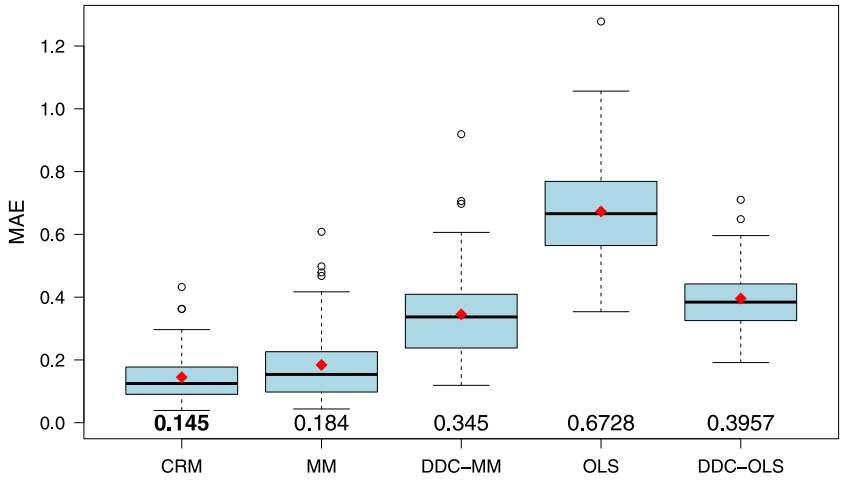

Fig. 4. Boxplot of MAE for each of the regression methods.

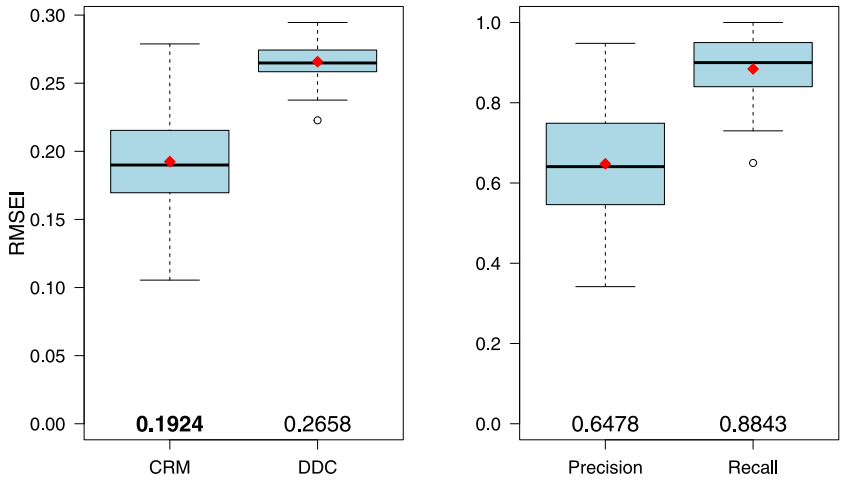

Fig. 5. (Left) Boxplot of RMSEI for CRM and DDC. (Right) Precision and recall of detected cellwise outliers by CRM.

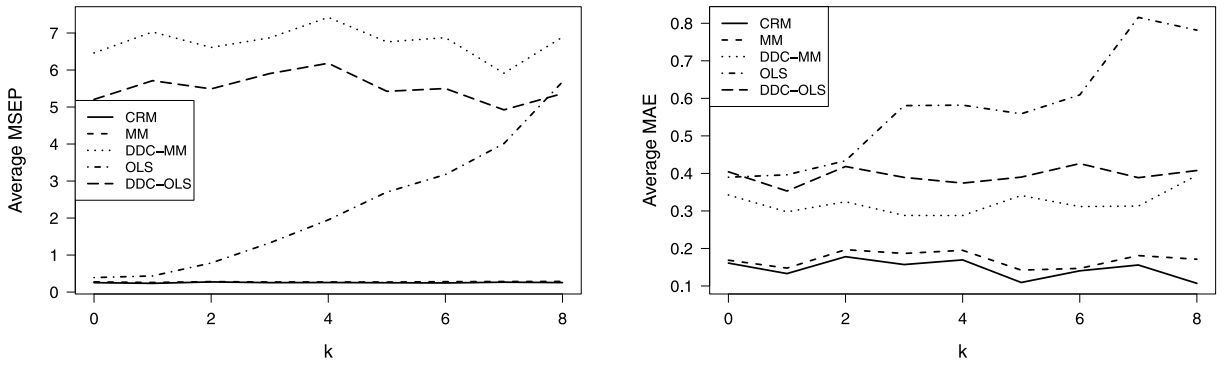

Fig. 6. Average MSEP (left) and MAE (right) for each of the regression methods for different values of k, controlling the magnitude of the outlyingness.

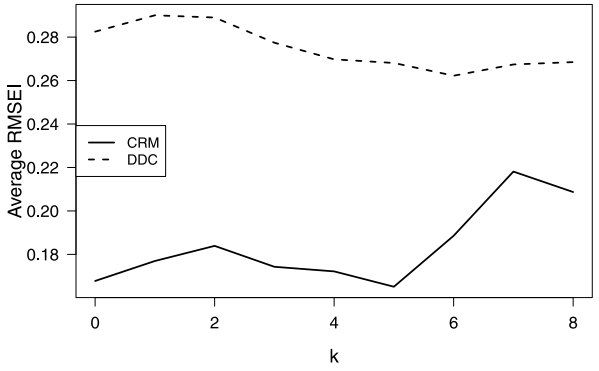

Fig. 7. Average RMSEI for CRM and DDC for different values of k, controlling the magnitude of the outlyingness.

Fig. 5 shows that CRM imputation does well detecting and imputing cells that are cellwise outliers in reality, but it may sometimes impute a few too many.

On average it took 11.5 s to execute the CRM algorithm across the hundred repeats. The execution times were measured on an Intel core i5 with 2.7 GHz and 8 GB RAM.

3.6. Simulation results for varying magnitude of contamination

The way how the data cells were contaminated was described in Section 3.2, where the magnitude of the contamination was controlled by the parameter k as a multiple of the standard deviation of the respective variable. In this section we report simulation results when this parameter is varied as $k \in \{0, 1, 2, \ldots, 8\}$, and thus we focus on the effect where the cells are getting more and more extreme, starting from the variable average. Figs. 6–8 present the average results across 10 simulation replications.

The results in Fig. 6 indicate that already the situation $k = 0$, where the cells are replaced by the column means, creates outliers which are identified by CRM and MM regression, as well as by DDC. There is clearly a strong effect on the prediction error (MSEP) and on the quality of the parameter estimates (MAE) of the non-robust OLS estimator if the contamination gets more extreme. All other estimators do not show a clear effect when k is varied. However, Figs. 7 and 8 provide more insight: increasing k leads to higher precision and recall for CRM. In other words, CRM can only correctly identify the outlying cells (and not declare others as outliers) if they are outlying clearly enough. With $k = 0$, precision and recall are still very low. On the other hand, the resulting error from imputation (RMSEI) for CRM is in the same range for k varying from 0 to 6, and only for bigger k it increases slightly, maybe due to an overfit effect, but is still clearly smaller than that of DDC.

3.7. Simulation results for breakdown

In the simulation settings considered so far, the fraction of contamination has been fixed at 5% (see Section 3.2). It will now gradually be increased, while keeping all other parameters unchanged (here, $k = 6$ is chosen again). Thus, more and more out of the 400 observations are contaminated, where in each observation 10% of the cells are randomly being picked for contamination. Fig. 9 shows the average of the results for MAE from all previously considered regression methods,

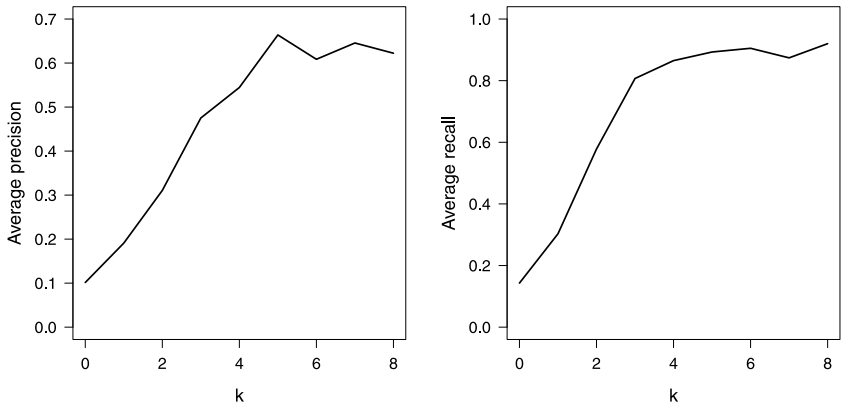

Fig. 8. Average precision (left) and average recall (right) of detected cellwise outliers by CRM for different values of k, controlling the magnitude of the outlyingness.

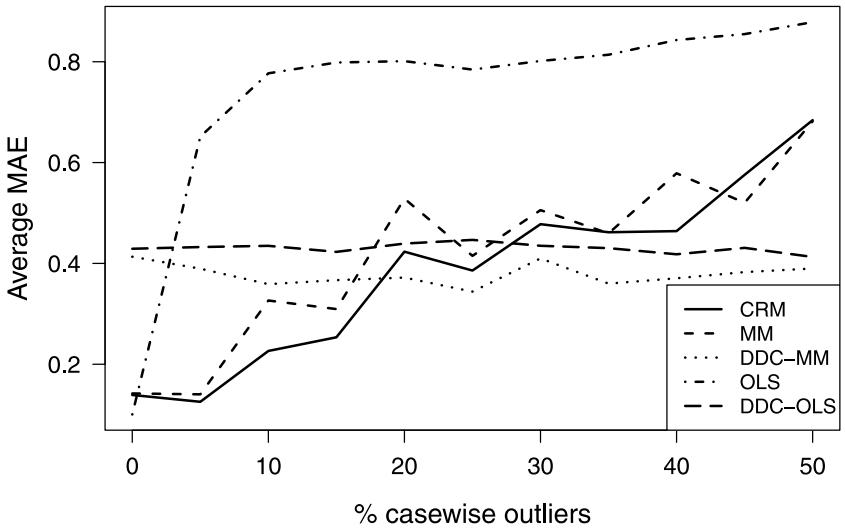

Fig. 9. Breakdown behavior of the different regression methods: the fraction of contamination (10% in each observation) is increased from 0 to 50%.

starting from no contamination, up to 50% (in steps of 5%). This reveals the breakdown behavior of the methods. The OLS estimator is the most efficient when no contamination is present, but its bias increases quickly when more and more contamination is added. The DDC-based methods are almost not affected by the amount of contamination, because DDC is highly robust, but they lead to a comparably large bias for small percentages of contamination. MM-regression and CRM behave very similarly, but CRM seems to have a slightly smaller bias. Finally, note that the robustness behavior of both methods depends on tuning parameters; in case of CRM this depends on the parameters of the Hampel function, which could be adjusted if necessary.

4. Real data example

The target of this analysis is to have a predictive model for cholesterol based on the nutrients contained in individual products. The data were taken from the Swiss nutrition data base 2015 (Nährwerttabelle, 2015). The original data set consists of nutrients on more than 40 components and 965 generic food products. We will focus on the first 193 products which do not contain any missing values and consider the variables in Table 1 where `cholesterol` is the response variable. Since all of these 6 variables are skewed right, they were logarithmically transformed first.

These data are a good example of data where one would expect the cellwise robust estimation technique to outperform the casewise one. While it is variable to assume a multivariate interplay between these variables in real life systems, these variables are measured independently. Moreover, biological effects can generate deviating behavior independently. It is therefore plausible to assume the necessity for multiple regression and corresponding multivariate effects taking place between the inputs, yet the mechanisms that generate outliers can be assumed to be largely independent.

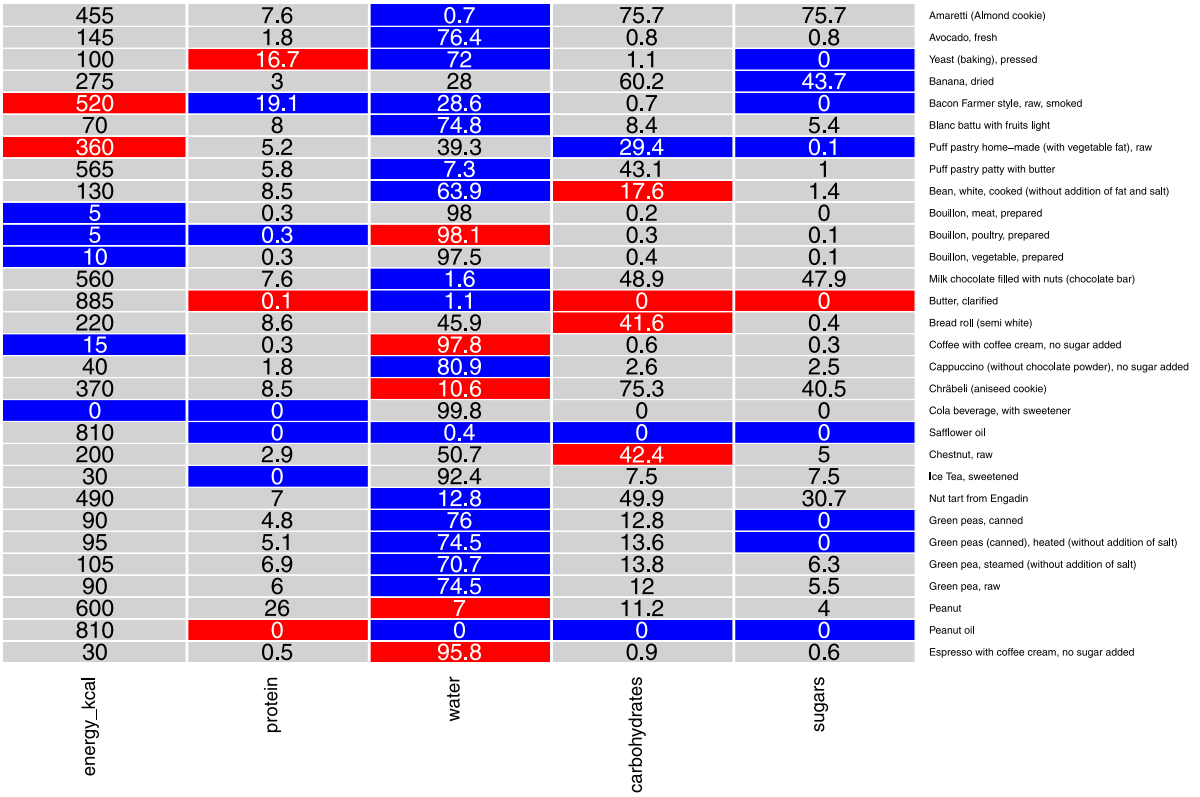

Fig. 10. Heatmap of outliers detected by CRM in the nutrients data. The red and blue boxes indicate the contaminated cells. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Variables of the nutrients data.

Variable	Description
`cholesterol`	cholesterol in milligram per 100 g edible portion
`energy_kcal`	energy in kcal per 100 g edible portion
`protein`	protein in gram per 100 g edible portion
`water`	water in gram per 100 g edible portion
`carbohydrates`	carbohydrates in gram per 100 g edible portion
`sugars`	sugars in gram per 100 g edible portion

The estimates of the regression coefficients by CRM are given in Table 2. It took around 1 s to apply the CRM algorithm on this data set. CRM indicates 30 out of 193 food products as casewise outliers having at least one contaminated outlying cell. The results are plotted as a heatmap in Fig. 10, where the 30 outliers are represented as rows and each contaminated/outlying cell is shown as a colored box. The anomalous cells, whose values are deviating either upwards or downwards, are colored red or blue, respectively. It can be seen that some food products have a lot of anomalous nutrient values, whereas others only have an atypical value for a few cells. Fig. 11 contains the nutritional data of the 30 anomalous food products where the deviating cells have been imputed by CRM. The blue cells are replaced with larger values while the red cells are imputed with smaller values (according to CRM). A more saturated color refers to bigger differences between the imputed and the original data values.

A 10-fold cross validation is conducted on the nutrients data set, using CRM as well as OLS and MM regression, also in the version where outliers are first replaced with the DDC method (DDC–OLS and DDC-MM). Fig. 12 shows that CRM clearly outperforms MM regression as well as the DDC-MM method in terms of the 10% trimmed root mean squared error of prediction (RMSEP).

Note that trimming makes the evaluation measure robust against the outliers (Maronna et al., 2019). As Figs. 10 and 11 show, CRM only imputes the cellwise outliers, which often just amount to a single cell per case. This implies that CRM can process up to 80% more relevant information for the outlying cases when compared to MM regression, that downweights these entire cases, some of them even to zero. Because CRM retains more uncontaminated information, it allows to make more accurate predictions. There is, however, only a smaller difference between CRM and OLS or DDC–OLS.

Table 2
Estimated regression coefficients of the nutrients data by CRM.

Variable	Estimated coefficient
(Intercept)	−33.73173
log.energy_kcal	3.62970
log.protein	0.98341
log.water	3.78561
log.carbohydrates	0.05336
log.sugars	−0.10999

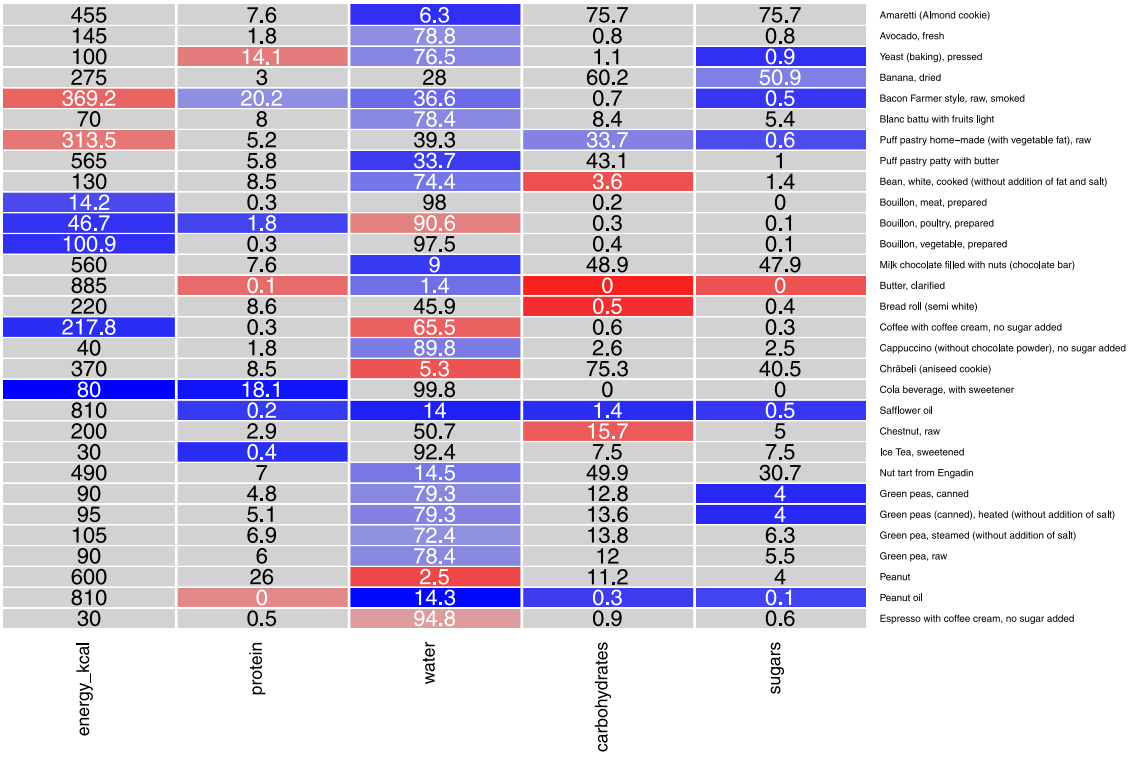

Fig. 11. Heatmap of outliers detected by CRM in the nutrients data. The red and blue boxes indicate the cells imputed by CRM, and higher saturation of the color refers to bigger differences to the original data. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

It seems that the outliers in the explanatory variables are not acting as bad leverage points, and correcting them by DDC even leads to a slightly poorer prediction performance for OLS regression. The imputation by CRM on the other hand seems much more appropriate. Even though the predictive performance is comparable to OLS, CRM has the advantage that it produces heatmaps of the outlying cells and the values that they have been imputed with, see Fig. 11. These can be of great value to the practitioner, since they allow to analyze which cells deviate per case and understand the outlier generating mechanism(s).

5. Conclusions and outlook

Cellwise robust M regression has been introduced as a regression method that is robust to vertical outliers and both cellwise and casewise and leverage points. Intrinsically, the method detects cells that are deviating *with respect to the linear model* and imputes them with more model consistent values. While CRM may not be the first method to detect deviating cells, it is the first to do so in a model consistent way for a linear model. This offers the practitioner a combined advantage of having a robust fit that reliably fits the majority of the data and producing a heatmap of suspected deviating cells, as well as model based imputations. Compared to casewise robust estimators, CRM will retain a larger fraction of uncontaminated data cells. Depending on the data, this fraction can be substantially larger. Therefore, the resulting fit is closer to the underlying model that generates the data, and the procedure is more efficient.

A simulation study has shown that CRM can generally be assumed to perform on par with a casewise robust estimator in terms of predictive power; an example prediction cholesterol from other nutrients has shown, though, that real life

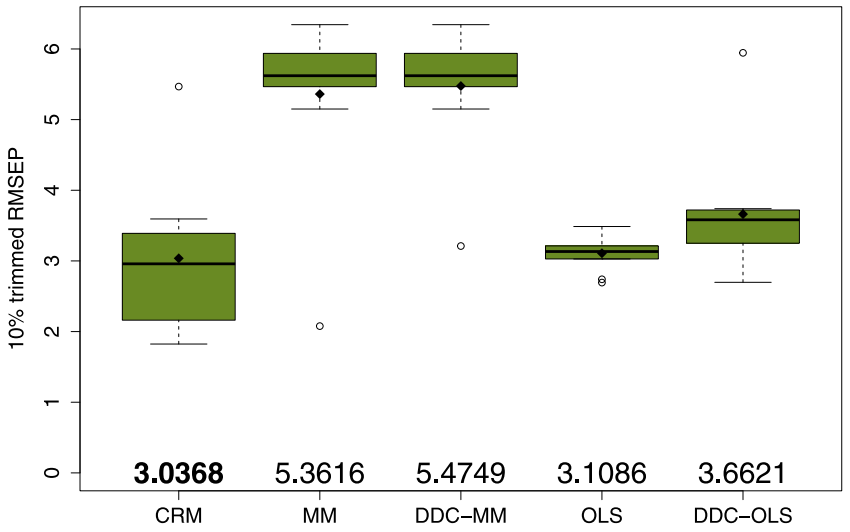

Fig. 12. Boxplot of 10% trimmed RMSEP values from 10-fold cross validation for each of the regression methods.

cases *can* be found for which CRM does outperform the casewise robust estimator. The simulation study has also shown that when a linear model can be assumed to have generated the data, detecting deviating cells and imputing them by CRM is a much better idea than using technique agnostic of the linear model, such as DDC, prior to regression. Also alternatives to DDC, for example kNN-based imputation methods, would suffer from the same problem that they ignore the underlying model. Finally, the simulation study has highlighted as well that CRM is slightly more efficient than MM at estimating individual regression coefficients.

On a casewise basis, there is widespread consensus that outliers are only outlying with respect to a model, and therefore need to be detected by robust estimators for the corresponding model. Few experts will assume that there exists a generic *data cleaning* estimator that can generically detect all outliers, regardless of which model is being estimated. This has led to over forty years of research on robust statistics, that has produced robust counterparts for virtually the entire arsenal of classical statistical estimators. However, on a cellwise basis, the few approaches in the literature, such as DDC, seem to be going for generically detecting cellwise outliers. DDC has practical merit: it can be applied to data sets that contain over 50% of casewise outliers, for which CRM would break down. That said, the majority of data sets containing cellwise outliers do not contain them in a majority of cases. In the latter situation, we argue that it is better to construct a model consistent cellwise robust estimator. The simulation study and example have corroborated this claim.

The introduction of CRM is a first step that opens the door to an entire class of model consistent cellwise robust estimators. Just like for casewise robust statistics, we would like to open up the field and see development starting on other cellwise robust statistics, such as cellwise robust principal component analysis, cellwise robust partial least squares, or cellwise robust canonical correlation analysis, just to name a few.

Another topic for further research would be inference. It has been shown that Wald type inference can be applied to MM-estimators (Koller and Stahel, 2011); these results have been incorporated into the R package `robustbase`. It will be a promising topic of research to extend these results to the cellwise setting. Note that while analytical results on inference for the CRM parameters are still to be generated, the practitioner can always resort to the robust bootstrap, which can be computed fast (Salibián-Barrera et al., 2008).

Acknowledgments

The authors are grateful to two anonymous reviewers. Their suggestions and remarks led to a substantially improved manuscript. This work was supported by the BNP Paribas Fortis Chair in Fraud Analytics and Internal Funds KU Leuven under Grant C16/15/068.

References

Debruyne, M., Höppner, S., Serneels, S., Verdonck, T., 2019. Outlyingness: which variables contribute most? Stat. Comput. 29 (4), 707–723.
Fritz, H., Filzmoser, P., Croux, C., 2012. A comparison of algorithms for the multivariate l1-median. Comput. Statist. 27 (3), 393–410.
Gauss, C.F., 1826. Theoria combinationis observationum erroribus minimis obnoxiae. Werke 4, 1–93.
Green, P.J., 1984. Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. J. R. Stat. Soc. Ser. B Stat. Methodol. 46 (2), 149–170.
Guerard, J.B., 2016. Investing in global markets: big data and applications of robust regression. Frontiers Appl. Math. Stat. 1, 14.

Hampel, F.R., Ronchetti, E.M., Rousseeuw, P.J., Stahel, W., 1986. Robust Statistics. The Approach Based on Influence Functions. Wiley and Sons, New York.
Hoffmann, I., Filzmoser, P., Serneels, S., Varmuza, K., 2016. Sparse and robust PLS for binary classification. J. Chemom. 30 (4), 153–162.
Hoffmann, I., Serneels, S., Filzmoser, P., Croux, C., 2015. Sparse partial robust M regression. Chemometr. Intell. Lab. Syst. 149, 50–59.
Hu, M., Zhang, W.M., Zhong, M., 2017. Robust regression and its application in absolute gravimeters. Rev. Sci. Instrum. 88 (5), 054501.
Huber, P.J., Ronchetti, E.M., 2009. Robust Statistics, second ed. John Wiley & Sons, Hoboken, NJ.
Koller, M., Stahel, W., 2011. Sharpening Wald-type inference in robust regression for small samples. Comput. Statist. Data Anal. 55 (8), 2504–2515.
Leoni, P., Segaert, P., Serneels, S., Verdonck, T., 2018. Multivariate constrained robust M-regression for shaping forward curves in electricity markets. J. Futures Mark. 38 (11), 1391–1406.
Maechler, M., Rousseeuw, P.J., Croux, C., Todorov, V., Ruckstuhl, A., Salibian-Barrera, M., Verbeke, T., Koller, M., Conceicao, E.L.T., Anna di Palma, M., 2018. Robustbase: basic robust statistics. R package version 0.93-3. URL http://robustbase.r-forge.r-project.org/.
Maronna, R., Martin, D., Yohai, V., 2006. Robust Statistics: Theory and Methods. John Wiley & Sons, Chichester.
Maronna, R.A., Martin, R.D., Yohai, V.J., Salibián-Barrera, M., 2019. Robust Statistics: Theory and Methods (with R). John Wiley & Sons, Chichester.
Nährwerttabelle, Infanger E. Schweizer, 2015. Schweizerische Gesellschaft für Ernährung. SGE, Bern. URL http://www.sge-ssn.ch/shop/produkt/schweizer-naehrwerttabelle/.
Öllerer, V., Alfons, A., Croux, C., 2016. The shooting S-estimator for robust regression. Comput. Statist. 31 (3), 829–844.
Rousseeuw, P.J., 1984. Least median of squares regression. J. Amer. Statist. Assoc. 79, 871–880.
Rousseeuw, P., Croux, C., 1993. Alternatives to the median absolute deviation. J. Amer. Statist. Assoc. 88 (424), 1273–1283.
Rousseeuw, P.J., Leroy, A.M., 1987. Robust Regression and Outlier Detection. Wiley and Sons, New York.
Rousseeuw, P.J., Van Driessen, K., 2006. Computing LTS regression for large data sets. Data Min. Knowl. Discov. 12, 29–45.
Rousseeuw, P.J., Vanden Bossche, W., 2018. Detecting deviating data cells. Technometrics 60 (2), 135–145.
Rousseeuw, P.J., Yohai, V., 1984. Robust regression by means of S-estimators. In: Robust and Nonlinear Time Series Analysis. Springer, pp. 256–272.
Salibián-Barrera, M., Van Aelst, S., Willems, G., 2008. Fast and robust bootstrap. Stat. Methods Appl. 17, 41–71.
Serneels, S., Croux, C., Filzmoser, P., Van Espen, P.J., 2005. Partial robust M-regression. Chemometr. Intell. Lab. Syst. 79 (1–2), 55–64.
Serneels, S., De Nolf, E., Van Espen, P.J., 2006. Spatial sign preprocessing: a simple way to impart moderate robustness to multivariate estimators. J. Chem. Inf. Model. 46 (3), 1402–1409.
Yohai, V.J., 1987. High breakdown-point and high efficiency estimates for regression. Ann. Statist. 15, 642–665.