WP.23
ENGLISH ONLY

**UNITED NATIONS STATISTICAL COMMISSION and
ECONOMIC COMMISSION FOR EUROPE**

CONFERENCE OF EUROPEAN STATISTICIANS

Work Session on Statistical Data Editing
(Ljubljana, Slovenia, 9-11 May 2011)

Topic (iv): Micro editing—methods and software

SOFTWARE FOR MULTIVARIATE OUTLIER DETECTION IN SURVEY DATA

Key Invited Paper

Prepared by Valentin Todorov (UNIDO), Matthias Templ (Statistics Austria) and
Peter Filzmoser (Vienna University of Technology)

I. INTRODUCTION

1. The multivariate aspect of the data collected in surveys makes the task of outlier identification particularly challenging. The outliers can be completely hidden in one or two dimensional views of the data. This underlines that univariate outlier detection methods are useless, although they are usually favored because of their simplicity. In a multivariate set-up the outlyingness of the observations can be measured by the Mahalanobis distance which is based on location and scatter estimates of the data set. In order to avoid the masking effect, robust estimates of these parameters are called for, even more, they must possess a positive breakdown point. In contrast to univariate outliers, multivariate outliers are not necessarily extreme along a single coordinate. They could deviate from the multivariate structure formed by the majority of the observations. To illustrate this we will consider the well-known `bushfire` data set which was used by Campbell [1989] to locate bushfire scars and was studied in detail by Maronna and Yohai [1995]. It is available from the R package `robustbase` and is a complete data set consisting of 38 observations in 5 dimensions. In the left panel of Figure 1 a scatter-plot of the variables `V2` and `V3` is shown which reveals most of the outliers - the two clusters 33-38 and 7-11. The estimated central location of each variable is indicated by dashed-dotted lines and their intersection represents the multivariate centroid of the data. The dotted lines are at the 2nd and 98th empirical percentiles for the individual variables. A univariate outlier detection would declare as candidates the observations falling outside the rectangle visualized with bold dotted lines. Such a procedure would ignore the elliptical shape of the bivariate data. The bivariate data structure can be visualized by Mahalanobis distances, which depend on the center and the covariance (see Equation (1) below). Certain quantiles (e.g. 0.25, 0.50, 0.75 and 0.98) will result in tolerance ellipses of the corresponding size. It is, however, crucial how location and covariance are estimated for this purpose. Both the univariate and the multivariate procedures illustrated in the left panel of Figure 1 are based on classical estimates of location and covariance and therefore they fail to identify the outliers in the data. The right panel of Figure 1 shows the same picture but robust estimates of location and covariance are used (here we used the MCD estimator, see below). All outliers lie clearly outside the ellipse corresponding to the 0.98th quantile.

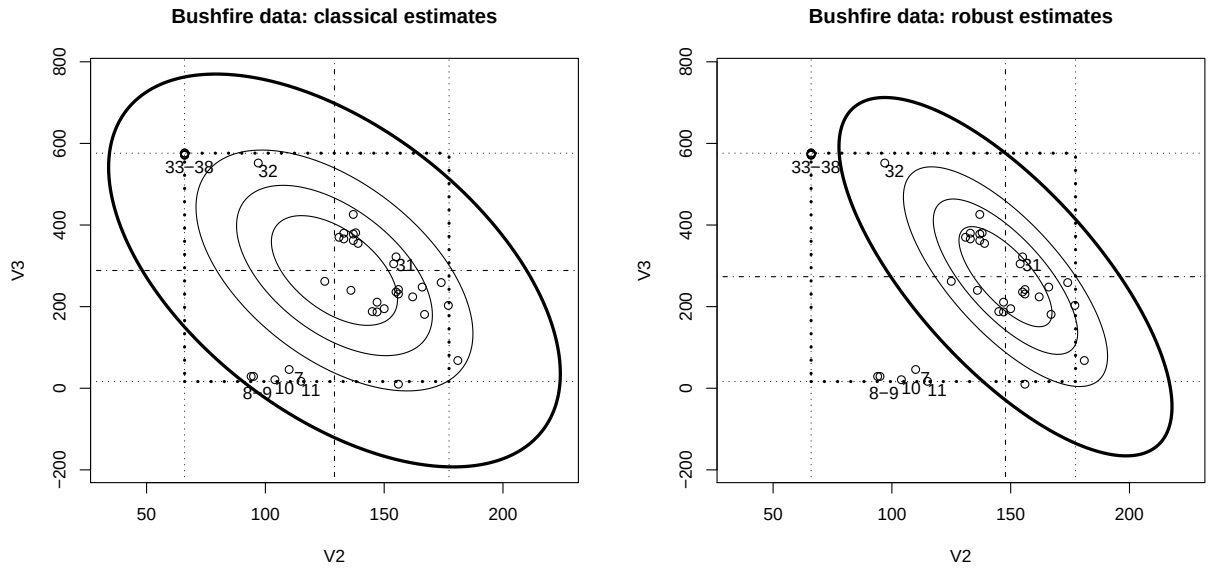

FIGURE 1. Example of multivariate outliers: the variables `V2` and `V3` of the bushfire data set. Dashed-dotted lines mark the centers of the variables, and ellipses represent the 0.25, 0.50, 0.75 and 0.98 quantiles of the Mahalanobis distances. The bold dotted lines are the 2nd and 98th empirical percentiles of the individual variables. In the left panel the sample mean and covariance matrix are used while the right panel is based on a robust alternative (Minimum Covariance Determinant).

2. Further challenges in survey data can be missing values and semi-continuous variables, to name some of them which are probably the main reasons for not applying the well known high-breakdown point estimators. A review of the available multivariate outlier detection methods which can cope with incomplete data was presented in Todorov et al. [2011]. In a simulation study, where a subset of the Austrian Structural Business Statistics was simulated, the authors compared several approaches. Robust methods based on the Minimum Covariance Determinant (MCD) estimator, S-estimators and OGK-estimator as well as BACON-BEM were shown to provide the best results in finding the outliers. The routine use of robust methods in a wide area of application domains including analysis of survey data is unthinkable without the computational power of today's personal computers and the availability of ready to use implementations of the algorithms. The freely available statistical software R has gained importance not only in the academic science but also in many other applied fields. The R repository CRAN includes the latest developments of statistical methods in the form of documented functions and example data sets, packaged in a unified way and ready to be installed and used. We will consider the package VIM for exploring the mechanism generating the missing values is with innovative visualization tools [see Templ et al., 2009] and the package `rrcovNA` [Todorov, 2011] providing a computational platform for robust multivariate analysis in R with incomplete data. The multivariate outlier detection methods will be illustrated on well known literature data sets.

3. The rest of the paper is organized as follows. Section II facilitates the quick start by an example session giving a brief overview of the package `rrcovNA`. In Section III the general outlier detection framework is presented, the available algorithms are briefly described and their applicability to incomplete data is discussed. Section IV presents the design approach and implementation details of the package `rrcovNA` and in Section V the visualization and diagnostic tools of the package are presented.

Section VI presents the availability of the software and Section VII concludes with discussion and outline of the future work.

II. EXAMPLE SESSION

4. In this section we will introduce the base functionalities provided in the package for analysis of incomplete data `rrcovNA` by an example session. First of all we have to load the package `rrcovNA` which will cause all necessary packages to be loaded too. The framework includes example data sets but here we will load only those which will be used throughout the following examples. For the rest of the paper it will be assumed that the package has been loaded already.

```
R> library("rrcovNA")
```

```
Scalable Robust Estimators with High Breakdown Point (version 1.1-00)
Scalable Robust Estimators with High Breakdown Point for
Incomplete Data (version 0.4-00)
```

```
R> data("bush10")
```

5. Most of the multivariate statistical methods are based on estimates of multivariate location and covariance, therefore these estimates play a central role in the framework. We will start with computing the robust *minimum covariance determinant* estimate for the data set `bush10` included in the package `rrcovNA`. After computing its robust location and covariance matrix using the MCD method implemented in the function `CovNAMcd()` we can print the results by calling the default `show()` method on the returned object `mcd`.Additional summary information can be displayed by the `summary()` method. The standard output contains the robust estimates of location and covariance. The summary output (not shown here) contains additionally the eigenvalues of the covariance matrix and the robust distances of the data items (Mahalanobis type distances computed with the robust location and covariance instead of the sample ones).

```
R> mcd <- CovNAMcd(bush10)
R> mcd
```

```
Call:
CovNAMcd(x = bush10)
-> Method:  Minimum Covariance Determinant Estimator for incomplete data.

Robust Estimate of Location:
   V1     V2     V3     V4     V5
109.5  149.5  257.9  215.0  276.9

Robust Estimate of Covariance:
    V1       V2       V3       V4       V5
V1    697.6    489.3  -3305.1   -671.4   -550.5
V2    489.3    424.5  -1889.0   -333.5   -289.5
V3  -3305.1  -1889.0  18930.9   4354.2   3456.4
V4   -671.4   -333.5   4354.2   1100.1    856.0
V5   -550.5   -289.5   3456.4    856.0    671.7
```

```
R> summary(mcd)
```

```
R> plot(mcd)
```

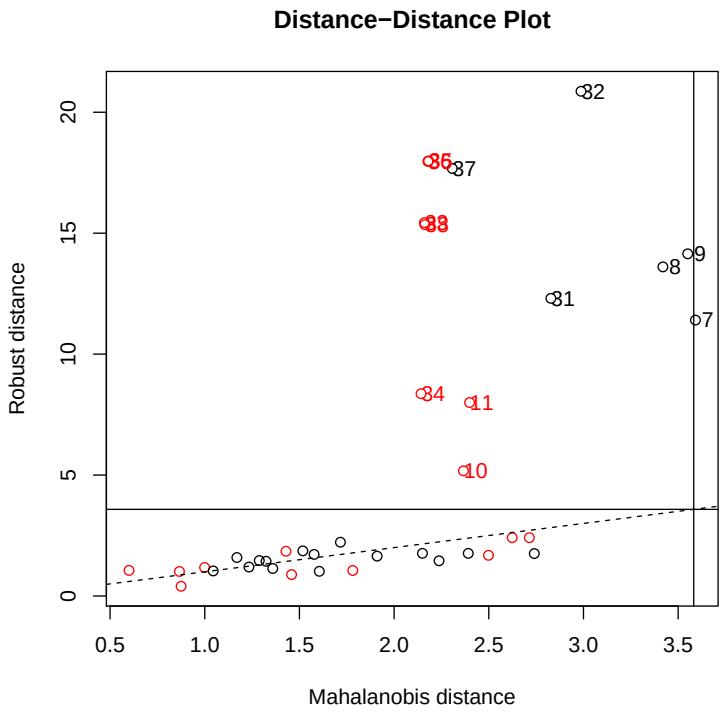

FIGURE 2. Example plot of the robust against classical distances for the modified `bushfire` data set (including missing values).

6. Now we will show one of the available plots by calling the `plot()` method—in Figure 2 the Distance-Distance plot introduced by Rousseeuw and van Zomeren [1990] is presented, which plots the robust distances versus the classical Mahalanobis distances and allows to classify the observations and identify the potential outliers. The observations containing missing values are shown in a different color. The description of this plot as well as examples of more graphical displays based on the covariance structure will be shown in Section V. Apart from the demonstrated MCD method the package provides many other robust estimators of multivariate location and covariance for incomplete data. It is important to note that one will get the output and the graphs in the same format, whatever estimation method was used. For example the following code lines will compute the S estimates for the same data set and provide the standard and extended output (not shown here).

```
R> est <- CovNASest(bush10, method = "bisquare")
R> est
R> summary(est)
```

Nevertheless, the variety of methods could pose a serious hurdle for the novice and could be quite tedious even for the experienced user. Therefore a shortcut is provided too—the function `CovNARobust()` can be called with a parameter set specifying any of the available estimation methods, but if this parameter set is omitted the function will decide on the basis of the data size which method to use. As we see in the example below, in this case it selects the Stahel-Donoho estimates. For details and further examples see Section IV.

```
R> est <- CovNARobust(bush10)
R> est

Call:
CovSde(x = x, control = obj)
```

```
-> Method:  Stahel-Donoho estimator

Robust Estimate of Location:
   V1     V2     V3     V4     V5
103.7  147.1  292.9  223.1  283.3

Robust Estimate of Covariance:
    V1       V2       V3       V4       V5
V1   905.2    749.9  -4660.2  -1206.0   -951.9
V2   749.9    708.1  -3482.7   -882.0   -692.6
V3 -4660.2  -3482.7  27046.7   7201.0   5719.6
V4 -1206.0   -882.0   7201.0   1966.0   1552.6
V5  -951.9   -692.6   5719.6   1552.6   1235.7
```

III. FUNCTIONS FOR OUTLIER DETECTION

7. *General principles.* A general framework for multivariate outlier identification in a p-dimensional data set $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ is to compute some measure of the distance of a particular data point from the center of the data and declare as outliers those points which are too far away from the center. Usually, as a measure of "outlyingness" for a data point $\boldsymbol{x}_i, i = 1, \ldots, n$, a robust version of the (squared) Mahalanobis distance RD_i^2 is used, computed relative to high breakdown point robust estimates of location $\boldsymbol{T}$ and covariance $\boldsymbol{C}$ of the data set $\boldsymbol{X}$:

$$RD_i^2 = (\boldsymbol{x}_i - \boldsymbol{T})^t \boldsymbol{C}^{-1} (\boldsymbol{x}_i - \boldsymbol{T}) \tag{1}$$

The most common estimators of multivariate location and scatter are the sample mean $\bar{\boldsymbol{x}}$ and the sample covariance matrix $\boldsymbol{S}$, i.e. the corresponding ML estimates (when the data follow a normal distribution). These estimates are optimal if the data come from a multivariate normal distribution but are extremely sensitive to the presence of even a few outliers in the data. In the last several decades much effort was devoted to the development of affine equivariant estimators possessing a high breakdown point. The most widely used estimators of this type are the Minimum Covariance Determinant (MCD) estimator and the Minimum Volume Ellipsoid (MVE) estimator, S-estimators and the Stahel-Donoho estimator. These estimators can be configured in such a way as to achieve the theoretically maximal possible breakdown point of 50% which gives them the ability to detect outliers even if their number is as much as almost half of the sample size. If we give up the requirement for affine equivariance, estimators like the orthogonalized Gnanadesikan-Kettenring (OGK) estimator are available and the reward is an extreme gain in speed. For definitions, algorithms and references to the original papers it is suitable to use Maronna et al. [2006]. Most of these methods are implemented in the R statistical environment [R Development Core Team, 2009] and are available in the object-oriented framework for robust multivariate analysis [Todorov and Filzmoser, 2009].

After having found reliable estimates for the location and covariance matrix of the data set, the second issue is to determine how large the robust distances should be in order to declare a point an outlier. The usual cutoff value is a quantile of the χ^2 distribution, like $D_0 = \chi^2_p(0.975)$. The reason is that if $\boldsymbol{X}$ follows a multivariate normal distribution, the squared Mahalanobis distances based on the sample mean $\bar{\boldsymbol{x}}$ and sample covariance matrix $\boldsymbol{S}$ follow χ^2_p distribution. This will no more be valid if robust estimators are applied and/or if the data have other than multivariate normal distribution. In Maronna and Zamar [2002] it was proposed to use a transformation of the cutoff value which should

help the distribution of the squared robust distances RD_i^2 to resemble χ^2 for non-normal original data:

$$D_0 = \frac{\chi_p^2(0.975) med(RD_1^2, \ldots, RD_n^2)}{\chi_p^2(0.5)}. \tag{2}$$

A drawback of all so far considered methods is that they work only with complete data which is not a usual case when dealing with sample surveys. In the next subsections we describe and introduce methods that are able to cope with missing values.

8. *Robustifying the EM algorithm.* Little and Smith [1987] were the first to propose a robust estimator for incomplete data by replacing the MLE in the M-step of the EM algorithm [see Dempster et al., 1977] by an estimator belonging to the general class of M-estimates and called this procedure ER-estimator. They suggested to use as a starting point for the ER algorithm ML estimation where the missing values were replaced by the median of the corresponding observed data. Unfortunately, the breakdown point of this estimator, as of all general M-estimates cannot be higher than $1/(p+1)$ [see for example Maronna et al., 2006, p. 186] which renders it unusable for the purpose of outlier detection.

9. *Normal imputation followed by high-BP estimation.* A straightforward strategy for adapting estimators of location and covariance to work with missing data is to perform one preliminary step of imputation and then run any of the above described algorithms, like for example MCD, OGK, S and Stahel-Donoho (SDE) on the complete data. Many different methods for imputation have been developed over the last few decades and here we will consider a likelihood-based approach such as the before mentioned expectation maximization (EM) imputation method [Dempster et al., 1977] assuming the underlying model for the observed data is Gaussian. This method is able to deal with MCAR and MAR missing values mechanism. The next step after estimating reliably the location $\boldsymbol{T}$ and covariance matrix $\boldsymbol{C}$ is to compute the robust distances from the incomplete data. For this purpose we have to adapt Equation (1) to use only the observed values in each observation $\boldsymbol{x}_i$ and then to scale up the obtained distance. We rearrange the variables if necessary and partition the observation $\boldsymbol{x}_i$ into $\boldsymbol{x}_i = (\boldsymbol{x}_{oi}, \boldsymbol{x}_{mi})$ where $\boldsymbol{x}_{oi}$ denotes the observed part and $\boldsymbol{x}_{mi}$ - the missing part of the observation. Similarly, the location and covariance estimates are partitioned, so that we have $\boldsymbol{T}_{oi}$ and $\boldsymbol{C}_{oi}$ as the parts of $\boldsymbol{T}$ and $\boldsymbol{C}$ which correspond to the observed part of $\boldsymbol{x}_i$. Then

$$RD_{oi}^2 = (\boldsymbol{x}_{oi} - \boldsymbol{T}_{oi})^t \boldsymbol{C}_{oi}^{-1} (\boldsymbol{x}_{oi} - \boldsymbol{T}_{oi}) \tag{3}$$

is the squared robust distance computed only from the observed part of $\boldsymbol{x}_i$. If $\boldsymbol{x}_i$ is uncontaminated, follow a multivariate normal distribution, and if the missing values are missing at random, then the squared robust distance given by Equation (3) is asymptotically distributed as $\chi_{p_i}^2$ where p_i is the number of observed variables in $\boldsymbol{x}_i$ [see Little and Smith, 1987].

The MCD estimator is not very efficient at normal models, especially if h is selected so that maximal breakdown point (BP) is achieved [Croux and Haesbroeck, 1999], and the same is valid for the OGK estimator [Maronna et al., 2006, p. 193, 207]. To overcome the low efficiency of these estimators, a reweighted version can be used. For this purpose a weight w_i is assigned to each observation $\boldsymbol{x}_i$, defined as $w_i = 1$ if $RD_{oi}^2 \leq \chi_{p_i,0.975}^2$ and $w_i = 0$ otherwise, relative to the raw estimates $(\boldsymbol{T}, \boldsymbol{C})$ and using Equation (3). Then the reweighted estimates are computed as

$$\begin{aligned} \boldsymbol{T}_R &= \frac{1}{\nu} \sum_{i=1}^{n} w_i \boldsymbol{x}_i, \\ \boldsymbol{C}_R &= \frac{1}{\nu - 1} \sum_{i=1}^{n} w_i (\boldsymbol{x}_i - \boldsymbol{T}_R)(\boldsymbol{x}_i - \boldsymbol{T}_R)^t, \end{aligned} \tag{4}$$

where ν is the sum of the weights, $\nu = \sum_{i=1}^{n} w_i$. Since the underlying data matrix is incomplete, the EM algorithm is used to compute $\boldsymbol{T}_R$ and $\boldsymbol{C}_R$. These reweighted estimates $(\boldsymbol{T}_R, \boldsymbol{C}_R)$ which have the same breakdown point as the initial (raw) estimates but better statistical efficiency are computed and used by default for the methods MCD and OGK.

10. *Robust sequential imputation followed by high-BP estimation.* Since we assume that outliers are present in the data we could expect an improvement of the performance of the previously described methods if the non-robust Gaussian imputation is substituted by a robust imputation technique that can handle simultaneously missing and outlying values. One such method was proposed by Vanden Branden and Verboven [2009] (`RSEQ`), extending the sequential imputation technique (`SEQimpute`) of Verboven et al. [2007] by robustifying some of its crucial steps. `SEQimpute` starts from a complete subset of the data set $\boldsymbol{X}_c$ and estimates sequentially the missing values in an incomplete observation, say $\boldsymbol{x}^*$, by minimizing the determinant of the covariance of the augmented data matrix $\boldsymbol{X}^* = [\boldsymbol{X}_c; (\boldsymbol{x}^*)^t]$. Since `SEQimpute` uses the sample mean and covariance matrix it will be vulnerable to the influence of outliers and it is improved by plugging in robust estimators of location and scatter. One possible solution is to use the outlyingness measure as proposed by Stahel [1981] and Donoho [1982] and successfully used for outlier identification in Hubert et al. [2005]. We can compute the outlyingness measure for the complete observations only but once an incomplete observation is imputed (sequentially) we could compute the outlyingness measure for it too and use it to decide if this observation is an outlier or not. If the outlyingness measure does not exceed a predefined threshold the observation is included in the further steps of the algorithm. After obtaining a complete data set we proceed by applying a high breakdown point estimation method in the same way as described in the previous section.

11. *Robust Principal Components for incomplete data.* Principal component analysis (PCA) is a widely used technique for dimension reduction achieved by finding a smaller number k of linear combinations of the originally observed p variables and retaining most of the variability of the data. These new variables, referred to as *principal components* are uncorrelated with each other and account for decreasing amount of the total variance, i.e. the first principal component explains the maximum variance in the data, the second principal component explains the maximum variance in the data that has not been explained by the first principal component and so on. Dimension reduction by PCA is mainly used for visualization of multivariate data by scatter plots (in a lower dimensional space) or transformation of highly correlated variables into a smaller set of uncorrelated variables which can be used by other methods (e.g. multiple or multivariate regression). The classical approach to PCA measures the variability through the empirical variance and is essentially based on computation of eigenvalues and eigenvectors of the sample covariance or correlation matrix. Therefore the results may be extremely sensitive to the presence of even a few atypical observations in the data. The outliers could artificially increase the variance in an otherwise uninformative direction and this direction will be determined as a PC direction. PCA was probably the first multivariate technique subjected to robustification, either by simply computing the eigenvalues and eigenvectors of a robust estimate of the covariance matrix or directly by estimating each principal component in a robust manner. Different approaches to robust PCA are presented in Todorov and Filzmoser [2009] and examples are given how these robust analysis can be carried out in R. Details about the methods and algorithms can be found in the corresponding references.
Projecting the data into a lower dimensional space one could obtain an estimate of location and covariance matrix and then use them for outlier detection as described in the beginning of this section or alternatively one could directly compute the Mahalanobis distances of the project observations to the projection of the center of the data [see for example Filzmoser et al., 2008].
If the data are incomplete as it is usual in business surveys the standard classical or robust PCA cannot be applied. Walczak and Massart [2001] proposed to use the EM approach for dealing with

missing data in PCA and Serneels and Verdonck [2008] extended it to robust PCA. Most of the known methods for robust PCA are implemented in the package `rrcov` [see Todorov and Filzmoser, 2009] and the corresponding versions for dealing with incomplete data can be found in the package `rrcovNA`. More details about the implementation and examples will be given in Section IV.

12. *Handling of semi-continuous variables.* Often in establishment and other surveys variables occur, which have valid values either in a given interval or are zero. These variables must be treated in the same way as regular variables, except that a value of zero is also accepted. Of course there could be a minimum bigger than zero on the variable and the number of zero valued observations could be larger then half of the total number. Such variables are called semi-continuous variables and it is obvious that none of the methods discussed so far can handle such type of variables. Recently Meraner [2010] proposed a modification for the OGK algorithm which can handle semi-continuous variables. This approach takes advantage of the pairwise character of the algorithm which allows to "skip" the zeros in the actual computation of robust location and covariance matrix estimates and then use them for outlier detection.

IV. OBJECT MODEL AND IMPLEMENTATION DETAILS

13. The object model for the S4 classes and methods implementing the different multivariate location and scatter estimators for incomplete data follows the proposed class hierarchy given in Todorov and Filzmoser [2009]. The abstract class `CovNA` serves as a base class for deriving all classes representing classical and robust location and scatter estimation methods. It defines the common slots and the corresponding accessor methods, provides implementation for the general methods like `show()`, `plot()` and `summary()`. The slots of `CovNA` hold some input or default parameters as well as the results of the computations: the location, the covariance matrix and the distances. The `show()` method presents brief results of the computations and the `summary()` method returns an object of class `SummaryCovNA` which has its own `show()` method. These slots and methods are defined and documented only once in this base class and can be used by all derived classes. Whenever new data (slots) or functionality (methods) are necessary, they can be defined or redefined in the particular class.

14. The classical location and scatter estimates for incomplete data are represented by the class `CovNAClassic` which inherits directly from `CovNA` (and uses all slots and methods defined there). The function `CovNAClassic()` serves as a constructor (generating function) of the class. It can be called by providing a data frame or matrix. As already demonstrated in Section II the methods `show()` and `summary()` present the results of the computations. The `plot()` method draws different diagnostic plots which are shown in one of the next sections. The accessor functions like `getCenter()`, `getCov()`, etc. are used to access the corresponding slots. Another abstract class, `CovNARobust` is derived from `CovNA`, which serves as a base class for all robust location and scatter estimators. The classes representing robust estimators like `CovNAMcd`, `CovNASest`, etc. are derived from `CovNARobust` and provide implementation for the corresponding methods. Each of the constructor functions `CovNAMcd()`, `CovNAOgk()` and `CovNASest()` performs the necessary computations and returns an object of the class containing the results. Similarly as the `CovNAClassic()` function, these functions can be called either with a data frame or a numeric matrix.

A. GENERALIZED ESTIMATION FUNCTION

15. The provided variety of estimation methods for incomplete data, each of them with different parameters as well as the object models described earlier in this section can be overwhelming for the user, especially for the novice who does not care much about the technical implementation of the framework. Therefore one function is provided which gives a quick access to the robust estimates of location and covariance matrix for incomplete data. The class `CovNARobust` is abstract (defined as *VIRTUAL*) and no objects of it can be created but any of the classes derived from `CovNARobust`, such as `CovNAMcd` or `CovNAOgk`, can act as an object of class `CovNARobust`. The function `CovNARobust()` which is technically not a constructor function can return an object of any of the classes derived from `CovNARobust` according to the user request. This request can be specified in one of three forms:

- If only a data frame or matrix is provided and the control parameter is omitted, the function decides which estimate to apply according to the size of the problem at hand. If there are less than 1000 observations and less than 10 variables or less than 5000 observations and less than 5 variables, Stahel-Donoho estimator will be used. Otherwise, if there are less than 50000 observations, either bisquare S estimates (in case of less than 10 variables) or Rocke type S estimates (for 10 to 20 variables) will be used. In both cases the S iteration starts at the initial MVE estimate. And finally, if there are more than 50000 observations and/or more than 20 variables the Orthogonalized Quadrant Correlation estimator (function `CovNAOgk()` with the corresponding parameters) is used. This is illustrated by the following example:

```
R> genNAData <- function(n, ncol) {
+     x <- rnorm(n)
+     x[sample(1:n, size = 0.1 * n)] <- NA
+     matrix(x, ncol = ncol)
+ }
R> getMeth(CovNARobust(genNAData(n = 40, ncol = 2)))
[1] "Stahel-Donoho estimator"
R> getMeth(CovNARobust(genNAData(n = 1600, ncol = 8)))
[1] "Stahel-Donoho estimator"
R> getMeth(CovNARobust(genNAData(n = 20000, ncol = 10)))
[1] "S-estimates: Rocke type"
R> getMeth(CovNARobust(genNAData(n = 2e+05, ncol = 2)))
[1] "Orthogonalized Gnanadesikan-Kettenring Estimator"
```

- The simplest way to choose an estimator is to provide a character string with the name of the estimator—one of `"mcd"`, `"ogk"`, `"s-fast"`, `"s-rocke"`, etc.

```
R> getMeth(CovNARobust(matrix(rnorm(40), ncol = 2), control = "rocke"))
[1] "S-estimates: Rocke type"
```

- If it is necessary to specify also some estimation parameters, the user can create a control object (derived from `CovControl`) and pass it to the function together with the data. For example to compute the OGK estimator using the median absolute deviation (MAD) as a scale estimate and the quadrant correlation (QC) as a pairwise correlation estimate we create a control object `ctrl` passing the parameters `s_mad` and `s_qc` to the constructor function and then call `CovNARobust` with this object.

```
R> data("toxicity")
R> ctrl <- CovControlOgk(smrob = "s_mad", svrob = "qc")
R> est <- CovNARobust(toxicity, ctrl)
```

For more details see the description of the function `CovRobust()` for complete data in Todorov and Filzmoser [2009].

B. ROBUST PCA FOR INCOMPLETE DATA

16. The object model for the S4 classes and methods implementing the principal component analysis methods follows the proposed class hierarchy given in Todorov and Filzmoser [2009] but for simplicity the number of classes is reduced and the different estimation methods are specified by a parameter. The abstract class `PcaNA` (derived from `Pca` in package `rrcov`) serves as a base class for deriving all classes representing classical and robust principal components analysis methods. It defines the common slots and the corresponding accessor methods, provides implementation for the general methods like `show()`, `plot()`, `summary()` and `predict()`. The slots of `PcaNA` hold some input or default parameters like the requested number of components as well as the results of the computations: the eigenvalues, the loadings and the scores. The `show()` method presents brief results of the computations, and the `predict()` method projects the original or new data to the space spanned by the principal components. It can be used either with new observations or with the scores (if no new data are provided). The `summary()` method returns an object of class `SummaryPca` which has its own `show()` method. As in the other sections of the package these slots and methods are defined and documented only once in this base class and can be used by all derived classes. Whenever new information (slots) or functionality (methods) are necessary, they can be defined or redefined in the particular class.

17. Classical principal component analysis for incomplete data is represented by the class `PcaNA` withe `method="class"` which inherits directly from `Pca` (and uses all slots and methods defined there). The function `PcaNA()` serves as a constructor (generating function) of the class. It can be called either by providing a data frame or matrix or a formula with no response variable, referring only to numeric variables. Let us consider the following simple example with the data set `bush10` containing missing values. The code line

```
R> PcaNA(bush10, method="class")
```

can be rewritten as (and is equivalent to) the following code line using the formula interface

```
R> PcaNA(~ ., data = bush10, method="class")
```

The function `PcaNA()` with `method="class"` performs the standard principal components analysis and returns an object of the class `PcaNA`.

```
R> pca <- PcaNA(~., data = bush10, method = "class")
R> pca
Call:
PcaNA(formula = ~., data = bush10, method = "class")

Standard deviations:
[1] 163.679611  27.335832  16.573119   8.417495   1.502502

Loadings:
           PC1         PC2        PC3        PC4          PC5
V1 -0.02659665  0.40918484  0.4023615  0.8184958 -0.005503999
V2 -0.01525114  0.90453802 -0.2930261 -0.3087768 -0.019260616
```

```
V3  0.90576986 -0.02651191 -0.3610926  0.2202021  0.001101088
V4  0.32660506  0.07626469  0.6095852 -0.3314624 -0.637221583
V5  0.26827246  0.08865405  0.5002589 -0.2763426  0.770419481
```

```
R> summary(pca)
```

```
Call:
PcaNA(formula = ~., data = bush10, method = "class")
Importance of components:
                            PC1     PC2      PC3     PC4     PC5
Standard deviation     163.6796 27.3358 16.57312 8.41749 1.50250
Proportion of Variance   0.9607  0.0268  0.00985 0.00254 0.00008
Cumulative Proportion    0.9607  0.9875  0.99738 0.99992 1.00000
```

```
R> plot(pca)
```

The `show()` method displays the standard deviations of the resulting principal components, the loadings and the original call. The `summary()` method presents the importance of the calculated components. The `plot()` draws a PCA diagnostic plot which is shown and described later. The accessor functions like `getLoadings()`, `getEigenvalues()`, etc. are used to access the corresponding slots, and `predict()` is used to rotate the original or new data to the space of the principle components. The robust PCA methods are performed by supplying the corresponding parameter to the function `PcaNA()` and correspond to the complete data methods `PcaHubert`, `PcaLocantore`, etc. derived from `PcaRobust` in `rrcov`. The constructor function `PcaNA()` with the corresponding paramater `method=c("locantore", "hubert", "grid", "proj", "class", "cov")` performs the necessary computations and returns an object of the class containing the results. In the following example the same data are analyzed using a projection pursuit method.

```
R> rpca <- PcaNA(~., data = bush10, method = "grid", k = 3)
R> rpca
```

```
Call:
PcaNA(formula = ~., data = bush10, method = "grid", k = 3)

Standard deviations:
[1] 134.503708  24.947475   4.794661

Loadings:
           PC1        PC2        PC3
V1 -0.01248133  0.5058076  0.1470629
V2 -0.12643309  0.7822248  0.3153328
V3 -0.87902225 -0.2551691  0.3964164
V4 -0.35099688  0.1952972 -0.6683504
V5 -0.29661417  0.1703843 -0.5244993
```

V. VISUALIZATION OF THE RESULTS

18. The default plot accessed through the method `plot()` of class `CovNARobust` is the Distance-Distance plot introduced by Rousseeuw and van Zomeren [1990]. An example of this graph, which plots the robust distances versus the classical Mahalanobis distances is shown in Figure 2. The dashed line represents the points for which the robust and classical distances are equal. The horizontal and vertical lines are drawn at values $x = y = \sqrt{\chi^2_{p,0.975}}$. Points beyond these lines can be considered as

outliers and are identified by their labels. All observations which have at list one missing value are shown in red.

19. The other available plots are accessible either interactively or through the `which` parameter of the `plot()` method. Figure 3 shows the pairwise correlations (`which="pairs"`) computed classically as the sample correlation coefficients (excluding the pairwise missing values) and computed robustly by applying the Minimum Covariance Determinant (MCD) method for incomplete data. In the upper triangle the corresponding ellipses are shown representing bivariate normal density contours with zero mean, unit variance together with a bivariate scatter plot of the data. The observations which have a missing value in any of the coordinates are projected on the axis and are shown in red. The lower triangle presents classical and robust correlation coefficients. A large positive or negative correlation is represented by an elongated ellipse with major axis oriented along the ±45 degree direction while near to zero correlation is represented by almost circular ellipse. The differences between the classical and robust estimates are easily seen visually.

```
R> mcd <- CovNAMcd(bush10)
R> plot(mcd, which = "pairs")
```

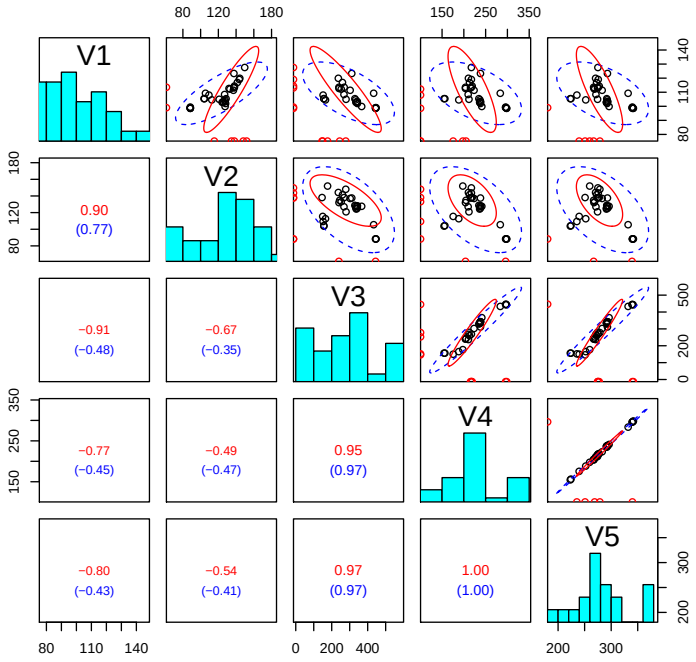

FIGURE 3. Classical and robust correlations and scatter plot matrix with tolerance ellipses.

20. The left panel of Figure 4 shows an example of the distance plot in which robust and classical Mahalanobis distances are shown in parallel panels. The outliers have large robust distances and are identified by their labels. The right panel of Figure 4 shows a Quantile-Quantile comparison plot of the robust and the classical Mahalanobis distances versus the square root of the quantiles of the chi-squared distribution.

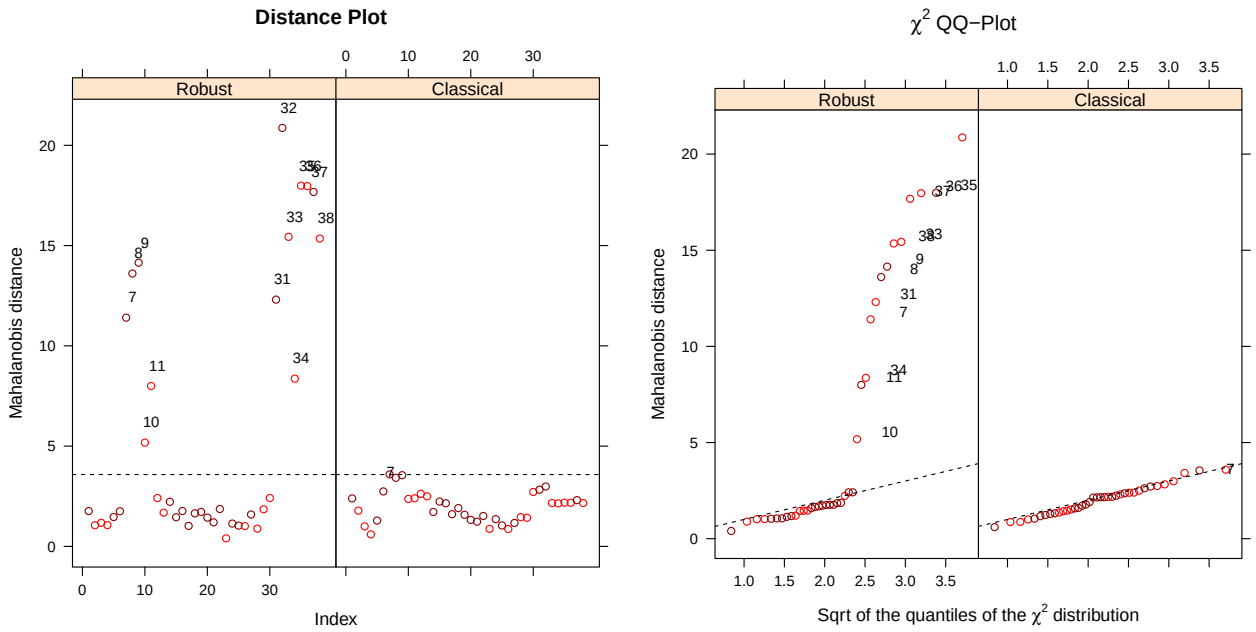

FIGURE 4. Distance plot and Chi-square Q-Q plot of the robust and classical distances.

The next plot shown in Figure 5 presents a scatter plot of the data on which the 97.5% robust and classical confidence ellipses are superimposed. The observations with distances larger than $\sqrt{\chi^2_{p,0.975}}$ are identified by their subscript. In the right panel of Figure 5 a `screeplot` of the `ces` data set is shown, presenting the robust and classical eigenvalues.

```
R> data("bush10")
R> data("ces")
R> X <- bush10[, c(2, 3)]
R> usr <- par(mfrow = c(1, 2))
R> plot(CovNAMcd(X), which = "tolEllipsePlot", classic = TRUE)
R> plot(CovNAMcd(ces), which = "screeplot", classic = TRUE)
R> par(usr)
```

In the context of PCA Hubert et al. [2005] defined a *diagnostic plot* or *outlier map* which helps to distinguish between the regular observations and the different types of outliers. The diagnostic plot is based on the *score distances* and *orthogonal distances* computed for each observation. The *score distance* is defined by

$$SD_i = \sqrt{\sum_{j=1}^{k} \frac{t_{ij}^2}{l_j}}, \qquad i = 1, \ldots, n, \tag{5}$$

where t_{ij} are the elements of the score matrix $\boldsymbol{T}$. It measures the distance of each observation to the subspace spanned by the first k principal components. The *orthogonal distance* is defined by

$$OD_i = ||\boldsymbol{x}_i - \boldsymbol{m} - \boldsymbol{P}\boldsymbol{t}_i||, \qquad i = 1, \ldots, n \tag{6}$$

where $\boldsymbol{t}_i$ is the ith row of the score matrix $\boldsymbol{T}$. This measure corresponds to the distance of the projection of each observation into the space spanned by the first k principal components. The *diagnostic plot* shows the score versus the orthogonal distance, and indicates with a horizontal and vertical line the

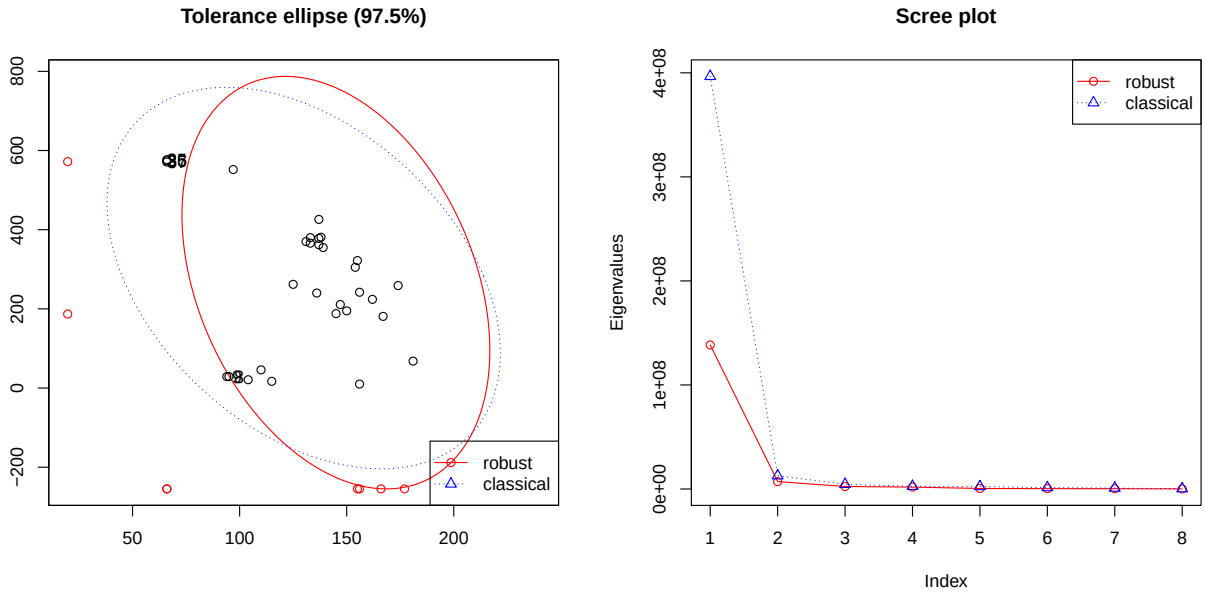

FIGURE 5. Robust and classical tolerance ellipse for two selected variables (V2 and V3) of the modified `bushfire` data data and robust and classical screeplot for the Consumer Expenditure Survey data (`ces` data set).

cut-off values that allow to distinguish regular observations from the two types of outliers [for details, see Hubert et al., 2005]. An example of the classical and robust diagnostic plot for the `bush10` data set is shown in Figure 6.

```
R> usr <- par(mfrow = c(1, 2))
R> plot(PcaNA(bush10, k = 3, method = "class"))
R> plot(PcaNA(bush10, k = 3, method = "locantore"))
R> par(usr)
```

VI. SOFTWARE AVAILABILITY

21. The algorithms discussed in this paper are available in the R package `rrcovNA` which in turn uses the packages `robustbase`, `rrcov` and `mvoutlier`. These packages are available from the Comprehensive R Archive Network (CRAN) at http://CRAN.R-project.org under the GNU General Public License. The three algorithms from the EUREDIT project (`TRC`, `EA` and `BEM`), were kindly provided by the authors and will be included in a later version of `rrcovNA`.

VII. CONCLUSIONS AND OUTLOOK

22. In this paper we presented several algorithms for for identifying outliers in data sets including missing values as well as their implementation in the R package `rrcovNA`. While a previous article investigated two important aspects: the computation time and the accuracy of the outlier detection method here we focused on the practical application of the methods and the available visualization

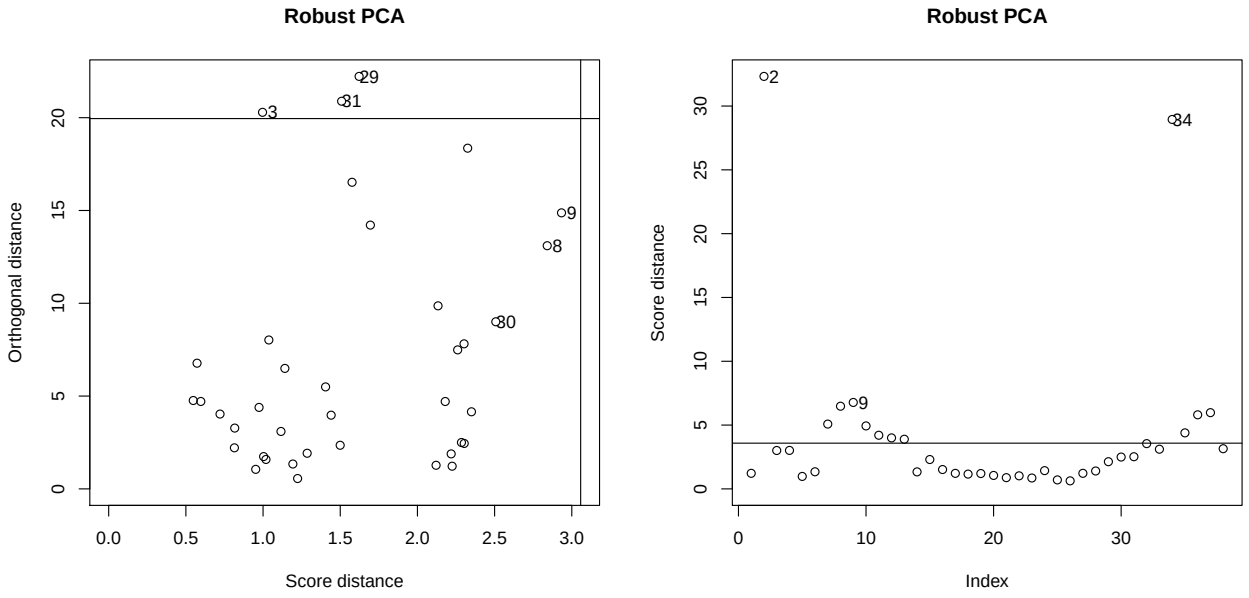

FIGURE 6. Classical and robust diagnostic plot for the `bush10` data with $k = 3$.

and diagnostic tools. The `doc` subdirectory of the package contains a vignette (user guide in PDF format), which presents much of the material in this article in greater detail.

23. First experiments with a real live example from the African Investor Survey 2009 in the Manufacturing Sector conducted by the United Nations Industrial Development Organization showed promising results and we intend to apply the methods implemented in package `rrcovNA` as the analysis of the survey data proceeds. As usual with business (and other) surveys the data contain many semi-continuous variables, which motivates further improvement of the methods that could handle this type of data distributions.

ACKNOWLEDGEMENTS

The views expressed herein are those of the authors and do not necessarily reflect the views of the United Nations Industrial Development Organization (UNIDO).

References

N. A. Campbell. Bushfire maping using NOAA AVHRR data. Technical report, CSIRO, 1989.

C. Croux and G. Haesbroeck. Influence function and efficiency of the minimum covariance determinant scatter matrix estimator. *Journal of Multivariate Analysis*, 71:161–190, 1999.

A. P. Dempster, M. N. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 39:1–22, 1977.

D. L. Donoho. Breakdown properties of multivariate location estimators. Technical report, Harvard University, Boston, 1982. URL http://www-stat.stanford.edu/~donoho/Reports/Oldies/

BPMLE.pdf.

Peter Filzmoser, Ricardo Maronna, and Mark Werner. Outlier identification in high dimensions. *Computational Statistics & Data Analysis*, 52(3):1694–1711, 2008.

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

R. J. A. Little and P. J. Smith. Editing and imputation for quantitative data. *Journal of the American Statistical Association*, 82:58–69, 1987.

R. A. Maronna and V. J. Yohai. The behavoiur of the Stahel-Donoho robust multivariate estimator. *Journal of the American Statistical Association*, 90:330–341, 1995.

R. A. Maronna and R. H. Zamar. Robust estimation of location and dispersion for high-dimensional datasets. *Technometrics*, 44:307–317, 2002.

R. A. Maronna, D. Martin, and V. Yohai. *Robust Statistics: Theory and Methods*. John Wiley & Sons, New York, 2006.

Angelika Meraner. Outlier detection for semi-continuous variables. Masters thesis, Vienna University of Technology, Vienna, 2010.

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2009. URL http://www.R-project.org/. ISBN 3-900051-07-0.

P. J. Rousseeuw and B. C. van Zomeren. Unmasking multivariate outliers and leverage points. *Journal of the American Statistical Association*, 85:633–651, 1990.

Sven Serneels and Tim Verdonck. Principal component analysis for data containing outliers and missing elements. *Computational Statistics & Data Analysis*, 52(3):1712–1727, January 2008. URL http://ideas.repec.org/a/eee/csdana/v52y2008i3p1712-1727.html.

W. A. Stahel. Robuste schätzungen: Infinitesimale optimalität und schätzungen von kovarianzmatrizen. Ph.d. thesis no. 6881, Swiss Federal Institute of Technology (ETH), Zürich, 1981. URL http://e-collection.ethbib.ethz.ch/view/eth:21890.

Matthias Templ, Andreas Alfons, and Peter Filzmoser. Visualization of missing values before imputation using the R-package VIM. *submitted for publication*, 2009.

Valentin Todorov. *rrcovNA: Scalable Robust Estimators with High Breakdown Point for incomplete data*, 2011. URL http://CRAN.R-project.org/package=rrcovNA. R package version 0.4-00.

Valentin Todorov and Peter Filzmoser. An object oriented framework for robust multivariate analysis. *Journal of Statistical Software*, 32(3):1–47, 2009. URL http://www.jstatsoft.org/v32/i03/.

Valentin Todorov, Matthias Templ, and Peter Filzmoser. Detection of multivariate outliers in business survey data with incomplete information. *Advances in Data Analysis and Classification*, 5:37–56, 2011.

Karlien Vanden Branden and Sabine Verboven. Robust data imputation. *Computational Biology and Chemistry*, 33(1):7–13, 2009.

Sabine Verboven, Karlien Vanden Branden, and Peter Goos. Sequential imputation for missing values. *Computational Biology and Chemistry*, 31(5-6):320–327, 2007.

B. Walczak and D.L. Massart. Dealing with missing data. Part I. *Chemometrics and Intellegent Laboratory Systems*, 58:15–27, 2001.