Journal of Statistical Software

April 2013, Volume 53, Issue 3. *http://www.jstatsoft.org/*

Fast and Robust Bootstrap for Multivariate Inference: The R Package FRB

Stefan Van Aelst
Ghent University

Gert Willems
Ghent University

Abstract

We present the **FRB** package for R, which implements the fast and robust bootstrap. This method constitutes an alternative to ordinary bootstrap or asymptotic inference procedures when using robust estimators such as S-, MM- or GS-estimators. The package considers three multivariate settings: principal components analysis, Hotelling tests and multivariate regression. It provides both the robust point estimates and uncertainty measures based on the fast and robust bootstrap. In this paper we give some background on the method, discuss the implementation and provide various examples.

Keywords: robustness, multivariate regression, principal components analysis, Hotelling tests, outliers.

1. Introduction

In this paper we introduce the **FRB** package for R (R Core Team 2012) which implements robust multivariate data analysis with inference based on the fast and robust bootstrap (FRB) method of Salibian-Barrera and Zamar (2002). The package is available from the Comprehensive R Archive Network (CRAN) at `http://CRAN.R-project.org/package=FRB`. Three multivariate settings are considered: 1. principal components analysis; 2. hypothesis testing for multivariate means; and 3. multivariate linear regression. The settings have in common that the classical analysis methods are easily robustified by the use of multivariate robust estimates of the type of S-, MM- or GS-estimates. These specific estimates allow the FRB to be applied in order to extend the robust point estimates with accompanying standard errors, confidence intervals or p values. We first give a short overview of the three concerned settings.

1.1. Principal components analysis (PCA)

Suppose we have a sample $\mathcal{X}_n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset I\!R^p$ from an unknown p-variate distribution G with center $\boldsymbol{\mu}$ and scatter matrix $\boldsymbol{\Sigma}$. Principal components analysis aims to explain the covariance structure of G. It can be used to reduce the dimension of the data without too much loss of information, by projecting the observations onto a small number of principal components which are linear combinations of the original p variables. On the population level the principal components are given by the eigenvectors of $\boldsymbol{\Sigma}$. In classical PCA, the components are estimated by the eigenvectors of the sample covariance or shape matrix. The corresponding eigenvalues measure the amount of variance explained by the components.

1.2. Hotelling T^2 tests

Consider again a sample $\mathcal{X}_n = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset I\!R^p$ from a p-variate distribution G with center $\boldsymbol{\mu}$ and scatter matrix $\boldsymbol{\Sigma}$. The one-sample Hotelling test is the standard tool for inference about the center $\boldsymbol{\mu}$. With $\overline{X}_n$ and S_n denoting the sample mean and sample covariance matrix, the hypothesis $H_0 : \boldsymbol{\mu} = \boldsymbol{\mu}_0$ is tested via the Hotelling T^2 statistic:

$$T^2 = n(\overline{X}_n - \boldsymbol{\mu}_0)^\top S_n^{-1}(\overline{X}_n - \boldsymbol{\mu}_0).$$

Now, consider two samples $\mathcal{X}_{n_1}^{(1)}$ and $\mathcal{X}_{n_2}^{(2)}$ from p-variate distributions G_1 and G_2 with respective centers $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ and common scatter matrix $\boldsymbol{\Sigma}$. Let $\overline{X}_{n_j}^{(j)}$ denote the sample mean of the j-th sample and let S_n^p be the pooled sample covariance matrix. Then, the two-sample Hotelling statistic

$$T^2 = \frac{n_1 n_2}{n_1 + n_2}(\overline{X}_{n_1}^{(1)} - \overline{X}_{n_2}^{(2)})^\top S_n^{p-1}(\overline{X}_{n_1}^{(1)} - \overline{X}_{n_2}^{(2)}),$$

can test the hypothesis $H_0 : \boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$. In both the one- and two-sample case, and under the assumption of underlying normality, the null distribution of the T^2 statistic is a multiple of an F-distribution.

1.3. Multivariate linear regression

Consider a sample of the form $\mathcal{Z}_n = \{(\mathbf{y}_1^\top, \mathbf{x}_1^\top)^\top, \ldots, (\mathbf{y}_n^\top, \mathbf{x}_n^\top)^\top\} \subset I\!R^{q+p}$. The multivariate linear regression model is given by

$$\mathbf{y}_i = \boldsymbol{\mathcal{B}}^\top \mathbf{x}_i + \boldsymbol{\epsilon}_i, \quad i = 1, \ldots, n\,, \tag{1}$$

where $\boldsymbol{\mathcal{B}}$ is the $p \times q$ matrix of regression coefficients. It is assumed that the q-variate error vectors $\boldsymbol{\epsilon}_i$ are independent and identically distributed with zero center and scatter matrix $\boldsymbol{\Sigma}_\epsilon$. The interest usually lies in the coefficient matrix $\boldsymbol{\mathcal{B}}$ which classically is estimated through least squares.

It is well known that sample means, sample covariances and least squares estimates in linear regression are very sensitive to outliers in the data. Hence, statistical inference based on such estimates can be severely distorted, even by a few atypical observations. In the last few decades a considerable number of alternative multivariate estimators have been proposed, which were designed to be robust against outliers (see e.g., Maronna and Yohai 1998; Maronna, Martin, and Yohai 2006; Hubert, Rousseeuw, and Van Aelst 2008). A primary measure of robustness is the breakdown point of the estimator. It is roughly defined as the minimum fraction of observations in the data that would need to be replaced by arbitrary values in order to arbitrarily change the original estimate. Intuitively, it reveals the maximum fraction of outliers that the estimator can withstand. The classical estimators mentioned above have a breakdown point of 0% (asymptotically), whereas so-called high-breakdown robust estimators can go up to 50%.

Among the range of available robust estimators, we here focus on the class of S-estimators (Rousseeuw and Yohai 1984; Davies 1987) and the related classes of MM-estimators (Tatsuoka and Tyler 2000) and GS-estimators (Croux, Rousseeuw, and Hössjer 1994; Roelant, Van Aelst, and Croux 2009). These classes of estimators succeed in combining a high degree of robustness with relatively good efficiency properties. The second main reason to opt for these estimators is that they fall into a framework that allows the application of the FRB method.

Indeed, when aiming beyond point estimation, that is, when one is also interested in standard errors, confidence intervals and hypothesis tests, the use of robust estimators poses difficulties. While the least-squares normal-theory is well established, the available theory for robust estimators is limited to asymptotic results, often requiring quite stringent assumptions. Resampling methods such as bootstrap provide an interesting alternative, but are hampered by two specific problems. The first of these (and often the most serious) is the computational complexity of robust estimators. All affine equivariant high-breakdown estimators require time-consuming algorithms to find a sufficiently accurate approximation (exact computation is usually even not feasible). Hence, resampling is often not practical, especially for large data sets. The second problem is the instability of the bootstrap in case of outliers: due to possible outlier propagation in the resamples, there is no guarantee that inference based on the bootstrap is as robust as the estimate in the original sample itself.

The FRB method was first introduced in the context of univariate regression MM-estimators by Salibian-Barrera and Zamar (2002), and later generalized to multivariate settings by Van Aelst and Willems (2005) and Salibian-Barrera, Van Aelst, and Willems (2006). It is based on the fact that S-, MM- or GS-estimators can be represented by smooth fixed point equations which allows us to calculate only a fast approximation of the estimates in each bootstrap sample. Hence, the computation is much easier than for ordinary bootstrap. Furthermore, stability is ensured since observations that were downweighted in the original sample will automatically be downweighted in the bootstrap samples as well.

The primary aim of the **FRB** package is to provide a software implementation of the FRB method. The package then enables robust multivariate data analysis that includes uncertainty measures such as standard errors, confidence limits and p values. Note however that in the FRB method the bootstrap component is tightly linked to the chosen point estimator. Therefore the functions in the **FRB** package perform both the computation of the point estimates and the subsequent actual FRB procedure. The **FRB** package thus offers the FRB method instead of the ordinary bootstrap method, but it can only be applied to the point estimates that are made available in the package. In this sense its concept differs from e.g., the well-known **boot** package (Canty and Ripley 2013), which provides only the bootstrap component and can be applied to any estimator of choice (implemented in R).

S and MM-estimators for multivariate location and scatter are already implemented in the R package **rrcov** (Todorov and Filzmoser 2009). We take advantage of this fast implementation, based on C code, to obtain these point estimates in the **FRB** package. The **rrcov** package also offers PCA based on robust covariance estimates, but does not go beyond point estimation. Univariate regression MM-estimators are available in the R package **robustbase** (Rousseeuw *et al.* 2012) and have accompanying standard errors and p values based on its asymptotic distribution. The multivariate analogue of MM-regression is not publicly available in R and, to the best of our knowledge, neither are any other robust multivariate regression methods. We may therefore summarize the contributions of the package **FRB** as follows:

– the package makes available the FRB method for robust inference.

– the package is the first to make available robust multivariate regression methods (specifically S-, MM- and GS-estimates).

While the point estimates for multivariate regression are part of the package, its name **FRB** reflects the essential component of the robust inference provided by the package.

The rest of the paper is organized as follows. Section 2 discusses multivariate S-, MM- and GS-estimators and describes how they can be used to obtain robust data analysis procedures. Section 3 explains the FRB method. Section 4 then covers the implementation of the estimates and the FRB method and gives an overview of the package **FRB**. Section 5 illustrates the use of the package through various examples, while Section 6 contains some concluding remarks.

2. Robust analysis based on S-, MM- or GS-estimators

2.1. Definitions

In this paper, ρ_0 and $\rho_1 : [0, \infty[\to [0, \infty[$ are so-called ρ-functions, which satisfy:

(R1) ρ is symmetric, twice continuously differentiable and $\rho(0) = 0$.

(R2) ρ is strictly increasing on $[0, c]$ and constant on $[c, \infty)$ for some finite constant c.

Rousseeuw and Yohai (1984) introduced S-estimators in univariate regression. One-sample multivariate S-estimators for location and scatter were later investigated by Davies (1987) and Lopuhaä (1989). Two-sample S-estimators, for robustly estimating two location vectors and a common covariance matrix, were considered by He and Fung (2000). A generalization to multivariate regression was considered in Van Aelst and Willems (2005). Note that the multivariate regression model encompasses the one- and two-sample location-covariance models (as well as the univariate regression model) as special cases. Nevertheless, we use separate definitions here for ease of understanding and because the applications are different for these special settings. The S-estimators can be defined as follows.

One-sample S-estimators. *Consider a sample $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subset I\!R^p$. Then, for a chosen function ρ_0, S-estimates of location and scatter $(\hat{\boldsymbol{\mu}}_n, \hat{\boldsymbol{\Sigma}}_n)$ minimize $|\mathbf{C}|$ subject to*

$$\frac{1}{n}\sum_{i=1}^{n} \rho_0\left([(\mathbf{x}_i - \mathbf{T})^\top \mathbf{C}^{-1}(\mathbf{x}_i - \mathbf{T})]^{\frac{1}{2}}\right) = b_0, \tag{2}$$

among all $\mathbf{T} \in I\!R^p$ and $\mathbf{C} \in PDS(p)$.

Two-sample S-estimators. *Consider two p-variate samples $\{\mathbf{x}_{11}, \ldots, \mathbf{x}_{1n_1}\}$ and $\{\mathbf{x}_{21}, \ldots, \mathbf{x}_{2n_2}\}$. S-estimates for the location vectors and common scatter matrix ($\hat{\boldsymbol{\mu}}_{1,n}$, $\hat{\boldsymbol{\mu}}_{2,n}$, $\hat{\boldsymbol{\Sigma}}_n$) minimize $|\mathbf{C}|$ subject to*

$$\frac{1}{n}\sum_{j=1}^{2}\sum_{i=1}^{n_j}\rho_0\left([(\mathbf{x}_{ji}-\mathbf{T}_j)^\top\mathbf{C}^{-1}(\mathbf{x}_{ji}-\mathbf{T}_j)]^{\frac{1}{2}}\right)=b_0, \tag{3}$$

among all $\mathbf{T}_1, \mathbf{T}_2 \in \mathbb{R}^p$ and $\mathbf{C} \in PDS(p)$.

Multivariate regression S-estimators. *Consider a sample $\{(\mathbf{y}_1^\top, \mathbf{x}_1^\top)^\top, \ldots, (\mathbf{y}_n^\top, \mathbf{x}_n^\top)^\top\} \subset I\!R^{q+p}$ and the linear regression model of (1). Then, the S-estimates for the regression coefficients and error scatter matrix ($\hat{\boldsymbol{\mathcal{B}}}_n$, $\hat{\boldsymbol{\Sigma}}_n$) minimize $|\mathbf{C}|$ subject to*

$$\frac{1}{n}\sum_{i=1}^{n}\rho_0\left([(\mathbf{y}_i-\mathbf{B}^\top\mathbf{x}_i)^\top\mathbf{C}^{-1}(\mathbf{y}_i-\mathbf{B}^\top\mathbf{x}_i)]^{\frac{1}{2}}\right)=b_0, \tag{4}$$

among all $\mathbf{B} \in I\!R^{p\times q}$ and $\mathbf{C} \in PDS(q)$.

Here, PDS(p) denotes the set of positive definite symmetric $p \times p$ matrices and by $|\mathbf{C}|$ we denote the determinant of the square matrix $\mathbf{C}$. The constant b_0 is usually chosen such that $b_0 = \mathsf{E}_\Phi[\rho_0(\|\mathbf{x}\|)]$, which ensures consistency at the normal model. In this paper and in the **FRB** package we use Tukey biweight ρ-functions, given by $\rho(t) = \min(t^2/2 - t^4/(2c^2) + t^6/(6c^4), c^2/6)$. The constant c can then be tuned to achieve any given degree of robustness, in terms of breakdown point (between 0% and 50%).

However, tuning ρ_0 involves a compromise since a higher degree of robustness corresponds to a lower Gaussian efficiency for the S-estimator. This trade-off can be avoided by computing a more efficient M-estimator as a follow-up step for the S-estimator, in which the robustly estimated S-scale is kept fixed. The resulting estimators are called multivariate MM-estimators, as introduced by Tatsuoka and Tyler (2000) for the one-sample location-covariance setting. The definition is straightforwardly generalized to the two-sample or the regression context as explained below (see also Van Aelst and Willems 2011).

In the following, let $\hat{\sigma}_n := |\hat{\boldsymbol{\Sigma}}_n|^{1/(2p)}$ denote the S-scale corresponding to the S-estimates defined above. Then, based on a function ρ_1 which typically differs from ρ_0 by having a larger tuning constant c, multivariate MM-estimators are defined as follows.

One-sample MM-estimators. *Given the S-scale $\hat{\sigma}_n$, the MM-estimates for location and shape ($\tilde{\boldsymbol{\mu}}_n$, $\tilde{\boldsymbol{\Gamma}}_n$) minimize*

$$\frac{1}{n}\sum_{i=1}^{n}\rho_1\left([(\mathbf{x}_i-\mathbf{T})^\top\mathbf{G}^{-1}(\mathbf{x}_i-\mathbf{T})]^{\frac{1}{2}}/\hat{\sigma}_n\right),$$

among all $\mathbf{T} \in I\!R^p$ and $\mathbf{G} \in PDS(p)$ for which $|\mathbf{G}|$=1.

Two-sample MM-estimators. *Given the S-scale $\hat{\sigma}_n$, MM-estimates for the location vectors and common shape matrix ($\tilde{\boldsymbol{\mu}}_{1,n}$, $\tilde{\boldsymbol{\mu}}_{2,n}, \tilde{\boldsymbol{\Gamma}}_n$) minimize*

$$\frac{1}{n}\sum_{j=1}^{2}\sum_{i=1}^{n_j}\rho_1\left([(\mathbf{x}_{ji}-\mathbf{T}_j)^\top\mathbf{G}^{-1}(\mathbf{x}_{ji}-\mathbf{T}_j)]^{\frac{1}{2}}/\hat{\sigma}_n\right),$$

among all $\mathbf{T}_1, \mathbf{T}_2 \in \mathbb{R}^p$ and $\mathbf{G} \in PDS(p)$ with $|\mathbf{G}|$=1.

Multivariate regression MM-estimators. *Given the S-scale $\hat{\sigma}_n$, the MM-estimates for the coefficients and error shape matrix ($\tilde{\boldsymbol{\mathcal{B}}}_n, \tilde{\boldsymbol{\Gamma}}_n$) minimize*

$$\frac{1}{n}\sum_{i=1}^{n}\rho_0\left([(\mathbf{y}_i-\mathbf{B}^\top\mathbf{x}_i)^\top\mathbf{G}^{-1}(\mathbf{y}_i-\mathbf{B}^\top\mathbf{x}_i)]^{\frac{1}{2}}/\hat{\sigma}_n\right),$$

among all $\mathbf{B} \in I\!R^{p\times q}$ and $\mathbf{G} \in PDS(q)$ with $|\mathbf{G}|$=1.

The MM-estimate for the scatter matrix is then defined as $\tilde{\boldsymbol{\Sigma}}_n = \hat{\sigma}_n^2\tilde{\boldsymbol{\Gamma}}_n$. The function ρ_1 can be tuned to achieve any given efficiency for the MM-estimates without affecting the breakdown point of the estimates, which is the same as that of the initial S-estimates and thus determined only by ρ_0. In practice, one usually chooses the constant c_0 in the (biweight) ρ_0-function for the S-scale that yields the maximal breakdown value of 50%, while $c_1(> c_0)$ in ρ_1 is tuned to additionally achieve 95% Gaussian efficiency. There is a limited cost associated with the M-step, in the sense that MM-estimates of location or regression have a higher maximum bias than S-estimates, if indeed $c_1 > c_0$ (see Berrendero, Mendes, and Tyler 2007). Salibian-Barrera *et al.* (2006) give some efficiency comparisons between S- and MM-estimates.

Generalized S-estimators (GS) were introduced by Croux *et al.* (1994) in univariate regression and are generally more efficient than regular S-estimators. They minimize a robust scale of the differences between residuals rather than of the residuals themselves. By using differences of residuals, the estimation procedure is “intercept-free” and consequently cannot be used to estimate the intercept or location vectors in general. However, the intercept can easily be estimated e.g., by an additional M-step, in which the GS-regression slope coefficients and error scatter are kept fixed. (Roelant *et al.* 2009) introduced GS-estimators for multivariate regression which is the setting that we consider here as well.

Multivariate regression GS-estimators. *Consider again a sample $\{(\mathbf{y}_1^\top, \mathbf{x}_1^\top)^\top, \ldots, (\mathbf{y}_n^\top, \mathbf{x}_n^\top)^\top\} \subset I\!R^{q+p}$ and the linear regression model (1). Suppose $\mathbf{x}_i^\top = (1, \mathbf{u}_i^\top)$, i.e., an intercept term is included. Then, the GS-estimates for the slope coefficients and error covariance ($\hat{\boldsymbol{\mathcal{B}}}_n^s, \hat{\boldsymbol{\Sigma}}_n$) minimize $|\mathbf{C}|$ subject to*

$$\binom{n}{2}^{-1}\sum_{i<j}\rho_0\left([(\mathbf{r}_i-\mathbf{r}_j)^\top\mathbf{C}^{-1}(\mathbf{r}_i-\mathbf{r}_j)]^{\frac{1}{2}}\right) = b_0,$$

with $\mathbf{r}_i = \mathbf{y}_i - \mathbf{B}^\top\mathbf{u}_i$, among all $\mathbf{B} \in I\!R^{(p-1)\times q}$ and $\mathbf{C} \in PDS(q)$.

Computing the S-, MM- or GS-estimates requires computationally demanding approximative algorithms. Details about their implementation in the **FRB** package are given in Section 4.1 below.

2.2. Robust analysis

In order to obtain statistical procedures which are more resistant to outliers, it generally suffices to replace the classical estimates by robust estimates such as the ones described above, as briefly explained in this section.

Robust principal components analysis

Robust estimates of the population principal components are obtained by taking the eigenvectors of the robust scatter estimates $\hat{\mathbf{\Sigma}}_n$ or $\tilde{\mathbf{\Sigma}}_n$, instead of those of the sample covariance matrix. This is the plug-in approach to robust PCA, as applied in Salibian-Barrera *et al.* (2006) (and see references therein for other authors).

Robust Hotelling T^2 test

Replacing the sample mean(s) and sample covariance in the Hotelling T^2 statistic by the robust one- or two-sample location and covariance estimates (which is again the plug-in approach), yields a robust test statistic. For example, the one-sample robust T^2 based on S-estimates is given by

$$T_R^2 = n(\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_0)^\top \hat{\mathbf{\Sigma}}_n^{-1} (\hat{\boldsymbol{\mu}}_n - \boldsymbol{\mu}_0), \tag{5}$$

while the two-sample version is

$$T_R^2 = \frac{n_1 n_2}{n_1 + n_2} (\hat{\boldsymbol{\mu}}_{1,n} - \hat{\boldsymbol{\mu}}_{2,n})^\top \hat{\mathbf{\Sigma}}_n^{-1} (\hat{\boldsymbol{\mu}}_{1,n} - \hat{\boldsymbol{\mu}}_{2,n}). \tag{6}$$

For the two-sample test, one can either use the two-sample estimates discussed above, or one can consider one-sample estimates in each group separately and *pool* the covariance estimates. In this paper we focus on the first approach (although the **FRB** package provides both options). The difficulty in any case is to obtain an estimate of the null distribution since the classical F-distribution does not hold anymore. However, bootstrapping can be used as explained in Roelant, Van Aelst, and Willems (2008).

Robust multivariate regression

Robust regression analysis is immediately obtained by using S-, MM-, or GS-estimates instead of the least squares estimates. S- and MM-estimates are nowadays quite routinely used for robust univariate linear regression, see e.g., the R package **robustbase** (Rousseeuw *et al.* 2012). The developments for the multivariate regression setting have only received attention in recent years, see e.g., Roelant *et al.* (2009) and references therein.

In each of these settings, both classical bootstrap and asymptotic inference have serious disadvantages, as explained above. Therefore, the option to perform FRB is most welcome. In the next section this method is reviewed.

3. Fast and robust bootstrap

The fast and robust bootstrap method was introduced by Salibian-Barrera and Zamar (2002). The idea is to draw bootstrap samples as usual, but instead of computing the actual (algorithm) estimator in each bootstrap sample, a fast approximation is computed based on the estimating equations of the estimator. For example, the following system of fixed-point equations holds for the multivariate regression S-estimator:

$$\hat{\boldsymbol{\mathcal{B}}}_n = \left(\sum_{i=1}^{n} \frac{\rho_0'(d_i)}{d_i}\mathbf{x}_i\mathbf{x}_i^\top\right)^{-1} \sum_{i=1}^{n} \frac{\rho_0'(d_i)}{d_i}\mathbf{x}_i\mathbf{y}_i, \tag{7}$$

$$\hat{\mathbf{\Sigma}}_n = \frac{1}{nb}\sum_{i=1}^{n} p\frac{\rho_0'(d_i)}{d_i}(\mathbf{y}_i - \hat{\boldsymbol{\mathcal{B}}}_n^\top\mathbf{x}_i)(\mathbf{y}_i - \hat{\boldsymbol{\mathcal{B}}}_n^\top\mathbf{x}_i)^\top + \left(\sum_{i=1}^{n} s_i\right)\hat{\mathbf{\Sigma}}_n, \tag{8}$$

where $d_i = [(\mathbf{y}_i - \hat{\boldsymbol{\mathcal{B}}}_n^\top\mathbf{x}_i)^\top \hat{\mathbf{\Sigma}}_n^{-1}(\mathbf{y}_i - \hat{\boldsymbol{\mathcal{B}}}_n^\top\mathbf{x}_i)]^{1/2}$, $s_i = \rho_0(d_i) - \rho_0'(d_i)d_i$.
In general, let $\hat{\mathbf{\Theta}}_n \in I\!R^m$ contain all estimates in vectorized form, for example in case of multivariate regression S-estimates $\hat{\mathbf{\Theta}}_n = (\mathrm{vec}(\hat{\boldsymbol{\mathcal{B}}}_n)^\top\ \mathrm{vec}(\hat{\mathbf{\Sigma}}_n)^\top)^\top$. Suppose further that $\hat{\mathbf{\Theta}}_n$ can be represented as a solution of fixed-point equations as

$$\hat{\mathbf{\Theta}}_n = \mathbf{g}_n(\hat{\mathbf{\Theta}}_n), \tag{9}$$

where the function $\mathbf{g}_n : I\!R^m \to I\!R^m$ depends on the sample. Given a bootstrap sample, randomly drawn with replacement from the original sample, the recalculated estimates $\hat{\mathbf{\Theta}}_n^*$ then solve

$$\hat{\mathbf{\Theta}}_n^* = \mathbf{g}_n^*(\hat{\mathbf{\Theta}}_n^*), \tag{10}$$

where the function $\mathbf{g}_n^*$ now depends on the bootstrap sample. As explained above, calculating the robust estimates $\hat{\mathbf{\Theta}}_n^*$ for every bootstrap sample can be a computationally expensive task. Moreover, even though we may assume that the solution to (9) was resistant to outliers, this does not guarantee that we will obtain an equally resistant solution to (10) as $\mathbf{g}_n^*$ is potentially more severely affected by outliers than $\mathbf{g}_n$ is. Instead of $\hat{\mathbf{\Theta}}_n^*$, however, we can easily calculate

$$\hat{\mathbf{\Theta}}_n^{1*} := \mathbf{g}_n^*(\hat{\mathbf{\Theta}}_n) \tag{11}$$

which can be viewed as a one-step approximation of $\hat{\mathbf{\Theta}}_n^*$ starting from the initial value $\hat{\mathbf{\Theta}}_n$. It can be shown that, under certain conditions, the distribution of $\hat{\mathbf{\Theta}}_n^*$ consistently estimates the sampling distribution of $\hat{\mathbf{\Theta}}_n$. It is intuitively clear, however, that the distribution of $\hat{\mathbf{\Theta}}_n^{1*}$ does not have this property in general. Indeed, the recalculated $\hat{\mathbf{\Theta}}_n^{1*}$ typically underestimate the actual variability of $\hat{\mathbf{\Theta}}_n$, mainly because every bootstrap sample uses the same initial value in the one-step approximation. To remedy this, a linear correction can be applied as follows. Using the smoothness of $\mathbf{g}_n$, we can apply a Taylor expansion about $\hat{\mathbf{\Theta}}_n$'s limiting value $\mathbf{\Theta}$,

$$\hat{\mathbf{\Theta}}_n = \mathbf{g}_n(\mathbf{\Theta}) + \nabla\mathbf{g}_n(\mathbf{\Theta})(\hat{\mathbf{\Theta}}_n - \mathbf{\Theta}) + R_n, \tag{12}$$

where R_n is the remainder term and $\nabla\mathbf{g}_n(.) \in I\!R^{m\times m}$ is the matrix of partial derivatives. The remainder term is typically of order $O_p(n^{-1})$, and then equation (12) can be rewritten as

$$\sqrt{n}(\hat{\mathbf{\Theta}}_n - \mathbf{\Theta}) = [\mathbf{I} - \nabla\mathbf{g}_n(\mathbf{\Theta})]^{-1}\sqrt{n}(\mathbf{g}_n(\mathbf{\Theta}) - \mathbf{\Theta}) + O_p(n^{-1/2}), \tag{13}$$

Similarly, for the bootstrap estimates we obtain

$$\sqrt{n}(\hat{\mathbf{\Theta}}_n^* - \hat{\mathbf{\Theta}}_n) = [\mathbf{I} - \nabla \mathbf{g}_n(\hat{\mathbf{\Theta}}_n)]^{-1}\sqrt{n}(\mathbf{g}_n^*(\hat{\mathbf{\Theta}}_n) - \hat{\mathbf{\Theta}}_n) + O_p(n^{-1/2}). \tag{14}$$

When $\mathbf{g}_n(.)$ is essentially a smooth function of means, as in (7)-(8) for example, it is straightforward to show that under certain regularity conditions

$$[\mathbf{I} - \nabla \mathbf{g}_n^*(\hat{\mathbf{\Theta}}_n)]^{-1}\sqrt{n}(\mathbf{g}_n^*(\hat{\mathbf{\Theta}}_n) - \hat{\mathbf{\Theta}}_n) = [\mathbf{I} - \nabla \mathbf{g}_n(\mathbf{\Theta})]^{-1}\sqrt{n}(\mathbf{g}_n(\mathbf{\Theta}) - \mathbf{\Theta}) + o_p(1), \tag{15}$$

where the left side should be considered as conditionally on the sample. Now, define the linearly corrected version of the one-step approximation (11) as

$$\hat{\mathbf{\Theta}}_n^{R*} := \hat{\mathbf{\Theta}}_n + [\mathbf{I} - \nabla \mathbf{g}_n(\hat{\mathbf{\Theta}}_n)]^{-1}(\hat{\mathbf{\Theta}}_n^{1*} - \hat{\mathbf{\Theta}}_n). \tag{16}$$

If (14) indeed holds, then $\hat{\mathbf{\Theta}}_n^{R*}$ will be estimating the same limiting distribution as the actual bootstrap calculations $\hat{\mathbf{\Theta}}_n^*$, and by (13) and (15) both will be consistently estimating the limiting distribution of $\hat{\mathbf{\Theta}}_n$ as desired. For one-sample S- or MM-estimates, a formal consistency proof under relatively mild conditions can be found in Salibian-Barrera *et al.* (2006), while Roelant *et al.* (2009) prove consistency for multivariate regression GS-estimates.

Clearly, $\hat{\mathbf{\Theta}}_n^{R*}$ is much faster to calculate than $\hat{\mathbf{\Theta}}_n^*$. It is also more resistant against a possibly large number of outlier recurrences in the bootstrap sample, as can be seen by noting that the estimating equations typically involve weighted least squares estimates (or means) and covariances. The weights will be small or even zero for observations detected as outliers. The approximation does not recompute the weights but instead gives each observation its original weight, such that $\hat{\mathbf{\Theta}}_n^{R*}$ is essentially equally robust as $\hat{\mathbf{\Theta}}_n$.

The principle outlined above can be applied to multivariate S-, MM- and GS-estimators. Application possibilities for bootstrap methods in general are extremely wide, as almost any conceivable statistical estimate can be "bootstrapped" to assess its uncertainty, see e.g., Davison and Hinkley (1997). The method is obviously most useful when the concerned estimate has limited distributional theory, which is the case for robust estimates in general. For the robust estimates considered here, the FRB can do anything that classical bootstrap could do, only much faster and with less instability problems. In Section 4.3 below we list the specific applications that are available in the **FRB** package.

4. Implementation

4.1. Point estimates

Computing S- and GS-estimates is far from trivial. Exact computation is practically infeasible because it involves optimizing non-convex objective functions with possibly many local minima, which on top require solving a non-linear equation to evaluate them. Since MM-estimates need initial S-estimates, they too are obviously computationally demanding. In practice this type of estimates is usually approximated by algorithms which combine random subsampling with iterations of reweighted least squares (RWLS). Salibian-Barrera *et al.* (2006) in the context of univariate S-regression presented a very effective version of such an algorithm, called the *fast-S* algorithm. The algorithm involves the following steps:

1. Randomly select N *elementary* subsamples on which to compute the least squares solution, such that we have N different starting candidates for the S-estimates.

2. Locally improve each of the N candidates through k steps of RWLS.

3. Evaluate the objective function on each of the N candidates and retain the t best ones.

4. Further locally improve each of these t candidates until convergence and retain the best one.

Typical values for the parameters here would be $N = 500$, $k = 2$ and $t = 5$, although the optimality of these numbers depends on the data (see e.g., Salibian-Barrera, Willems, and Zamar 2008). In particular, N would ideally increase exponentially with the dimension of the data in order to preserve high robustness against outliers.

Generalization of the fast-S algorithm to multivariate settings is straightforward. The algorithm is also easily adapted to compute GS-estimates. Finally, once S-estimates are available, the second part in the MM-estimation requires only one iteration of RWLS. For multivariate location and scatter the **rrcov** package (Todorov and Filzmoser 2009) provides implementations of the S and MM-estimators using the fast-S algorithm. Starting from **rrcov** version 1.3-01 the output of these functions provides all information that is needed to perform the bootstrap part of the FRB method corresponding to the point estimates. However, for the two-sample and multivariate regression settings, implementations of these estimators are not yet available. In the **FRB** package, the functions

`Sest_twosample()` and `Sest_multireg()`

implement the fast-S algorithm in these two settings. They are also respectively called by the functions

`MMest_twosample()` and `MMest_multireg()`,

to compute the S-part of the corresponding MM-estimates. These functions additionally do the final iteratively RWLS part of these estimates. Finally, the function

`GSest_multireg()`

performs a version of fast-S which is adapted to GS-estimates. In these functions the parameters N, k and t default to the values given above, but can be changed by the user. The tuning of the ρ-functions in the estimates is by default set to obtain a 50% breakdown estimator in all cases. The second ρ-function in case of the MM-estimates is chosen to additionally yield 95% Gaussian efficiency. These settings can all be changed by the user to any sensible values.

4.2. Bootstrap distribution estimate

The implementation of the FRB procedure is quite straightforward:

1. Based on the original sample, compute the gradient $\nabla \mathbf{g}_n(\hat{\mathbf{\Theta}}_n)$ and the corresponding correction matrix as given in (16).

2. Draw R bootstrap samples as usual.

3. For each bootstrap sample, compute the one-step approximation given by (11) and multiply by the correction matrix to obtain the bootstrap recalculations of the estimates.

Note that the correction matrix only needs to be computed once. In the package, the FRB in the various settings is performed by the functions

`Sboot_loccov()`, `Sboot_twosample()` and `Sboot_multireg()`

for the S-estimates, by the functions

`MMboot_loccov()`, `MMboot_twosample()` and `MMboot_multireg()`

for the MM-estimates, and by the function

`GSboot_multireg()`

for the GS-estimates. These "`.boot_...`" functions require the result from the respective "`.est_...`" functions as input parameter. Ideally the functions return R recalculated S-, MM- or GS-estimates. However, a few problems may sometimes occur with the approximation (16), leading to less than R usable recalculations:

- in the regression setting: the number of distinct observations with non-zero weight drawn into the bootstrap sample, needs to be sufficient to enable computation of the weighted least squares approximation (11); this is not guaranteed, especially in small samples, although failure is generally rare.

- in the PCA and Hotelling setting: due to the linear correction in (16) the FRB approximations to the scatter matrix estimates may lack positive definiteness, leading to awkward results when used to compute e.g., eigenvalues; the frequency of occurrence depends on the dimension and sample size.

If one of these problems occurs, the bootstrap sample is omitted from further calculations. An exception is made in the PCA setting in the rare event that more than 75% of the samples have failed to produce a positive definite scatter matrix. In that case, the `make.positive.definite` function from the **corpcor** package (Schäfer, Opgen-Rhein, Zuber, Ahdesmäki, Silva, and Strimmer 2012) is used in an attempt to rescue the bootstrap samples. If the attempt is succesful (which it often is but not guaranteed), the bootstrap sample is used and a warning is produced.

4.3. Bootstrap applications

Once we have the FRB-based estimate of the sampling distribution of the S-, MM- or GS-estimates, we use it to derive several measures of uncertainty. Note that the FRB-estimate can be applied in exactly the same way as one would apply the ordinary bootstrap estimate (which again would be much more time-consuming and less robust). Here we give an overview of the bootstrap applications that are available in the **FRB** package for the respective settings. Examples and illustrations are presented in Section 5 below.

Principal components analysis

Salibian-Barrera *et al.* (2006), while examining the performance of the FRB in robust PCA, considered four ways in which the bootstrap can be useful. We follow their approach in our implementation and provide the following uncertainty measures (in the following let $\hat{\boldsymbol{\Sigma}}_n$ denote either S- or MM-estimates of scatter):

- Standard errors and confidence limits for the eigenvalues of $\hat{\boldsymbol{\Sigma}}_n$, which estimate the eigenvalues of the population scatter matrix $\boldsymbol{\Sigma}$. The eigenvalues can also be seen as estimates of the variance in the respective principal components.
- Standard errors and confidence limits for $\widehat{p}_k = (\sum_{i=1}^{k} \widehat{\lambda}_i)/(\sum_{i=1}^{p} \widehat{\lambda}_i)$, where $\widehat{\lambda}_i; i = 1, \ldots, p$ are the ordered eigenvalues of $\hat{\boldsymbol{\Sigma}}_n$. The statistic $\widehat{p}_k$ estimates the percentage of variance explained by the first k robust principal components ($k = 1, \ldots, p-1$) and is often used to decide how many principal components are retained for further analysis. Confidence limits for $\widehat{p}_k$ can give additional information on which to base such a decision.
- A distribution estimate of the angles between the eigenvectors of $\hat{\boldsymbol{\Sigma}}_n$ and those of $\boldsymbol{\Sigma}$. The estimate is given by the empirical distribution of the angles between the bootstrap recalculated eigenvectors of $\hat{\boldsymbol{\Sigma}}_n^{R*}$ and the eigenvectors of $\hat{\boldsymbol{\Sigma}}_n$. For example, an eigenvector of $\hat{\boldsymbol{\Sigma}}_n$ which is relatively aligned with its recalculations based on $\hat{\boldsymbol{\Sigma}}_n^{R*}$, can be considered an accurate estimate of the corresponding eigenvector of $\boldsymbol{\Sigma}$.
- Standard errors and confidence limits for the loadings of the robust principal components, which are the coefficients of the normalized eigenvectors of $\hat{\boldsymbol{\Sigma}}_n$ and which estimate the loadings of the population level principal components.

Hotelling T^2 tests

In general, using the bootstrap to perform hypothesis tests requires some care because one needs to ensure that the resampling is done under conditions that agree with the null hypothesis. However, when a test statistic is pivotal, it suffices to draw ordinary bootstrap samples (that is, with replacement from the original sample). The pivotal assumption is usually reasonable for test statistics such as z- and t-type statistics and their multivariate T^2 variants, see e.g., Davison and Hinkley (1997, p. 139). Hence, following Roelant *et al.* (2008) we may assume that the distribution of the robust T_R^2 in (5) does not depend on the true value of $\boldsymbol{\mu}_0$. Therefore, the null distribution of T_R^2 can be approximated by the distribution of

$$T_R^{2*} = n(\hat{\boldsymbol{\mu}}_n^{R*} - \hat{\boldsymbol{\mu}}_n)^\top \hat{\boldsymbol{\Sigma}}_n^{R*,-1} (\hat{\boldsymbol{\mu}}_n^{R*} - \hat{\boldsymbol{\mu}}_n),$$

where $(\hat{\boldsymbol{\mu}}_n^{R*}, \hat{\boldsymbol{\Sigma}}_n^{R*})$ are the FRB approximations for the location and covariance S-estimates in the bootstrap sample. Similarly, the null distribution of the robust two-sample test statistic (6) can be approximated by the distribution of

$$T_R^{2*} = \frac{n_1 n_2}{n_1 + n_2}((\hat{\boldsymbol{\mu}}_{1,n}^{R*} - \hat{\boldsymbol{\mu}}_{2,n}^{R*}) - (\hat{\boldsymbol{\mu}}_{1,n} - \hat{\boldsymbol{\mu}}_{2,n}))^\top \hat{\boldsymbol{\Sigma}}_n^{R*,-1} ((\hat{\boldsymbol{\mu}}_{1,n}^{R*} - \hat{\boldsymbol{\mu}}_{2,n}^{R*}) - (\hat{\boldsymbol{\mu}}_{1,n} - \hat{\boldsymbol{\mu}}_{2,n})),$$

with analogous notation. For both tests, the 5% critical value is then the 95% empirical quantile of the R recalculated T_R^{2*} statistics, where e.g., $R = 1000$. Rather than a single critical value, we consider two main bootstrap results for the Hotelling tests:

- An estimated bootstrap p value, naturally obtained as the fraction of the recalculated T_R^{2*} statistics which exceed the value of the original T_R^2.
- Simultaneous confidence intervals for the components of the true location vector (or difference between the two location vectors), based on the empirical quantiles of T_R^{2*}, similarly to the classical T^2-based intervals (Johnson and Wichern 1988, p. 239).

Multivariate linear regression

In the regression model (1), the bootstrap is mainly used to provide an uncertainty measure on the estimates of the coefficients in $\boldsymbol{\mathcal{B}}$. Specifically, we use the FRB to provide the following inference:

- Standard errors and confidence limits for the coefficient estimates.
- Bootstrap p values corresponding to each coefficient, obtained by exploiting the duality between confidence intervals and hypothesis tests. The p value is defined as the probability p^* such that $1 - p^*$ is the smallest coverage level for which the confidence interval would include zero.

Van Aelst and Willems (2005) and Roelant *et al.* (2009) investigated the performance of FRB-based confidence intervals for $\boldsymbol{\mathcal{B}}$, respectively in case of S- and GS-estimates. Hypothesis tests that concern more than one parameter, such as for comparing nested models, require adapted resampling schemes. Salibian-Barrera (2005) investigated such tests in the univariate case. However, they have not yet been considered in the multivariate setting and thus are not available in the **FRB** package.

Confidence interval methods

In both the PCA and the regression setting we consider bootstrap confidence intervals. There are several well-known methods to compute such intervals from the bootstrap result. We implemented two of these: bias-corrected and accelerated (BCa) intervals, as considered by all FRB references mentioned above, and so-called basic bootstrap intervals (see e.g., Davison and Hinkley 1997, p. 204 and p. 29 respectively).

BCa intervals are defined as follows. Suppose $\hat\theta_n$ is estimating the scalar parameter θ. Let $\hat\theta_n^{*(\alpha)}$ be the 100α-th percentile of the R bootstrap estimates $\hat\theta_n^{*,1}, \ldots, \hat\theta_n^{*,R}$ for that parameter. Then, the BCa interval for θ with coverage $1 - 2\alpha$ is given by

$$(\hat\theta_n^{*(\alpha_1)},\ \hat\theta_n^{*(\alpha_2)}),$$

where

$$\begin{aligned} \alpha_1 &= \Phi\left(w + \frac{w + z_\alpha}{1 - a(w + z_\alpha)}\right), \\ \alpha_2 &= \Phi\left(w + \frac{w + z_{1-\alpha}}{1 - a(w + z_{1-\alpha})}\right). \end{aligned}$$

Here Φ is the standard normal cumulative distribution function, z_α is the 100α-th percentile of the standard normal distribution, w is the bias-correction parameter and a the acceleration factor. Both w and a need to be estimated from the data. Estimation of w is straightforward, while for a we use empirical influences computed from the theoretical influence function assuming normality (see Davison and Hinkley 1997, p. 209).

The basic bootstrap interval on the other hand, with coverage $1 - 2\alpha$, is given by

$$(2\hat{\theta}_n - \hat{\theta}_n^{*(1-\alpha)},\ 2\hat{\theta}_n - \hat{\theta}_n^{*(\alpha)}).$$

These intervals have been formally shown to be inferior to BCa intervals regarding accuracy. However, they may be more appealing because of their simpler definition and calculation.

4.4. R functions overview

The main functions in the **FRB** package are listed in Table 1. These functions can be called with a formula interface or by providing a dataframe(s) or matrix (matrices). They process the results from the FRB as described in Section 4.3 and produce objects which have print, plot and summary methods.

Figure 1 displays the functional structure of the package, with E standing for either S, MM or GS. Here, a solid arrow from `foo1` to `foo2` indicates that `foo2` is called by `foo1`, while a dashed arrow would mean that `foo1` requires the result of `foo2`. Note for example that

PCA	Hotelling	Regression
`FRBpcaS()`	`FRBhotellingS()`	`FRBmultiregS()`
`FRBpcaMM()`	`FRBhotellingMM()`	`FRBmultiregMM()`
		`FRBmultiregGS()`

Table 1: Main functions in package **FRB**.

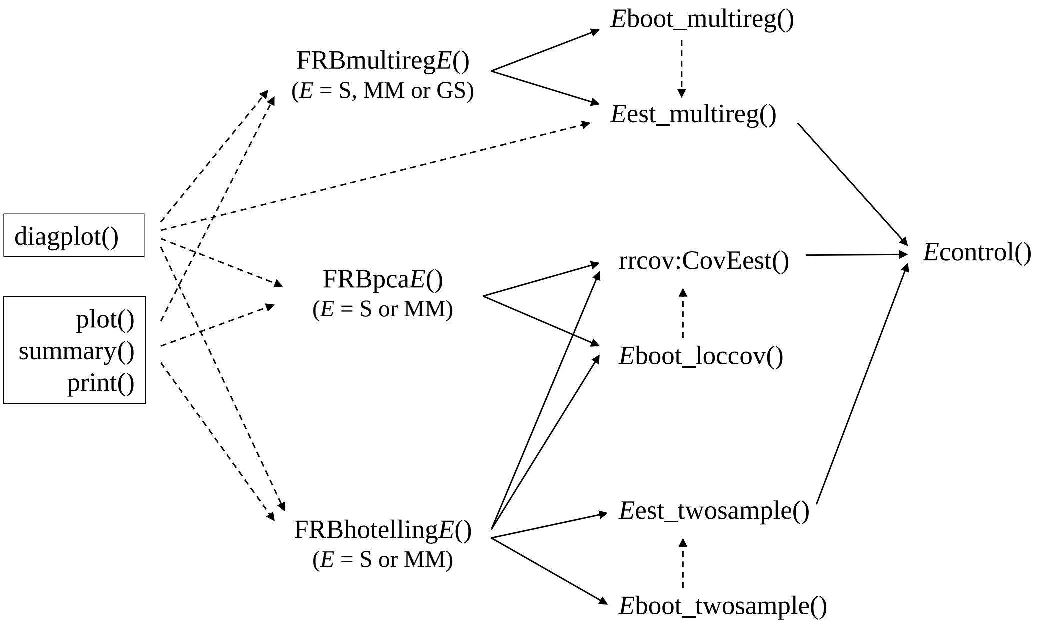

Figure 1: Structure of the **FRB** package.

the two-sample location-scatter functions are called only in the (two-sample) Hotelling test procedure, while the respective one-sample functions are used both for (one-sample) Hotelling tests and for PCA.

The function `Scontrol()` allows to change the default settings for the fast-S algorithm. It can directly be used in the input of the main functions such as `FRBpcaS()`. For the GS-estimates the `GScontrol()` function acts analogously. For MM-estimates, the `MMcontrol()` function sets the fast-S parameters for the S-part and additionally allows e.g., to change the Gaussian efficiency, which defaults to 95%.

The main functions listed above return objects of respective classes

`FRBpca`, `FRBhot` and `FRBmultireg`.

For each of these classes the following methods exist:

`plot()`, `summary()` and `print()`.

These will be illustrated in the next section, but we first give an overview. The `plot` method acts as follows:

- `plot.FRBpca()`: The function produces graphs depicting the FRB inference results, essentially as listed in Section 4.3, or in particular:
  1. FRB confidence intervals for the eigenvalues or the variances explained by the components.
  2. FRB confidence intervals for the cumulative percentage of variance explained by the components.
  3. Histograms of the angles between the FRB recalculated components and the original components.
  4. FRB confidence intervals for the loadings of the principal components.
- `plot.FRBhot()`: The function produces graphs depicting:
  1. The histogram of the FRB recalculated test statistics.
  2. FRB simultaneous confidence intervals for the components of the location or difference between locations.
- `plot.FRBmultireg()`: The function depicts histograms for each of the FRB recalculated coefficient estimates with indication of the corresponding confidence intervals.

Moreover, the `diagplot()` method is available for outlier detection purposes. The plot is based on robust (Mahalanobis-type) distances of the observations. It is thus not related to the FRB, but solely uses the point estimates of the original sample. Therefore, for the multivariate regression methods provided by the **FRB** package, the `diagplot` function can also be applied directly on the point estimates. In particular, the following diagnostic plots are presented:

- `diagplot.FRBpca()`: Based on the location and covariance estimates, robust distances d_i are computed for each observation. E.g., for S-estimates $d_i = [(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_n)^\top \hat{\boldsymbol{\Sigma}}_n^{-1} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_n)]^{1/2}$. These are plotted either versus the observation index, or versus a measure of the overall empirical influence that the observation would have on the classical principal components. The latter demands some additional computation time in order to obtain a simulation-based cutoff value for the empirical influences (see Pison and Van Aelst 2004, for details).

- `diagplot.FRBhot()`: Based on the one- or two-sample location and covariance estimates, robust distances are computed for each observation and are plotted against their index (separately for each sample in the two-sample case).

- `diagplot.FRBmultireg()`: Based on the estimates for the regression coefficients and the error covariance matrix, robust distances d_i are computed for each residual. E.g., for S-estimates $d_i = [(\mathbf{y}_i - \hat{\boldsymbol{\mathcal{B}}}_n^\top \mathbf{x}_i)^\top \hat{\boldsymbol{\Sigma}}_n^{-1} (\mathbf{y}_i - \hat{\boldsymbol{\mathcal{B}}}_n^\top \mathbf{x}_i)]^{1/2}$. These are again plotted either versus the observation index, or versus the robust distance of the observation in the covariate space. This last option is the typical diagnostic plot as introduced by Rousseeuw and Van Zomeren (1990) in univariate regression and by Rousseeuw, Van Aelst, Van Driessen, and Agulló (2004) in multivariate regression. The plot makes a distinction between three types of outliers: so-called bad leverage, good leverage or vertical outliers. It demands additional computation time since robust estimates for the location and scatter in the covariate space are required. For these additional computations the function uses the same type of estimate as that was used to produce the `FRBmultireg` object, with the same breakdown point and control settings.

The `summary` and `print` methods give formatted output that resembles well the output of the R functions for the corresponding classical methods. this should make the output easy to understand, even by non-experts in robustness, and facilitates comparison with the output of the classical method. For a large part the formatted output provides the numerical counterpart of the graphical representation produced by the `plot` method. However, it sometimes provides additional information. For example the function `summary.FRBmultireg()` lists p values for each regression coefficient, as explained in Section 4.3.

5. Examples

We now illustrate the use of the **FRB** package through some examples (a few more examples are available in the documentation of the package). Throughout this section we focus on MM-estimates. The use of S- or GS-estimates would obviously be similar.

5.1. Principal components analysis

Our first example concerns the Swiss Bank Notes data (Flury and Riedwyl 1988), which consists of $p = 6$ measurements on 100 real and 100 forged Swiss 1000 francs bills. We here consider only the forged bills. These data are available in the package through

```
R> data("ForgedBankNotes")
```

Suppose we are interested in the covariance structure of the data. To guard against the influence of possible outliers, robust PCA is advisable. PCA based on MM-estimates with FRB inference is obtained via

```
R> res <- FRBpcaMM(ForgedBankNotes, R = 999, conf = 0.95)
```

or alternatively, by using the formula interface

```
R> res <- FRBpcaMM(~ ., data = ForgedBankNotes, R = 999, conf = 0.95)
```

Note that we have specified that the number of bootstrap samples should be $R = 999$ and the confidence intervals should have nominal coverage of 95%. These are also the default settings for the FRB functions in the package. The `summary` method presents an overview of the results:

```
R> summary(res)
```

```
PCA based on multivariate MM-estimates (bdp = 0.5, eff = 0.95)

Eigenvalues:
                  PC1  PC2   PC3   PC4   PC5   PC6
      estimates 10.10 1.92 1.051 0.502 0.412 0.238
  BCa 95% lower  7.44 1.27 0.798 0.365 0.327 0.190
  BCa 95% upper 12.66 2.61 1.399 0.606 0.586 0.350

Principal components loadings:
             PC1      PC2    PC3     PC4    PC5       PC6
Length   -0.0710  0.36087  0.219  0.8519 -0.131 -2.72e-01
Left      0.0267  0.42366  0.256 -0.0175 -0.257  8.30e-01
Right    -0.0197  0.53517  0.205 -0.5015 -0.434 -4.81e-01
Bottom    0.8160 -0.00833  0.485 -0.0309  0.304 -7.80e-02
Top      -0.5651 -0.15725  0.725 -0.1213  0.339 -2.24e-02
Diagonal -0.0930  0.61570 -0.291 -0.0828  0.722 -1.57e-05

Average angle between PC and its bootstrapped versions:
   PC1   PC2   PC3   PC4   PC5   PC6
 0.086 0.247 0.340 0.641 0.808 0.402
(in [0 - pi/2], cf. aligned - perpendicular)

Percentage of variance explained by first k components:
     Est. (BCa 95% lower   upper)
k=1  71.0           63.9     76.9
k=2  84.5           78.7     87.7
k=3  91.9           88.1     93.9
k=4  95.4           93.0     96.3
k=5  98.3           97.5     98.7
k=6 100.0          100.0    100.0
```

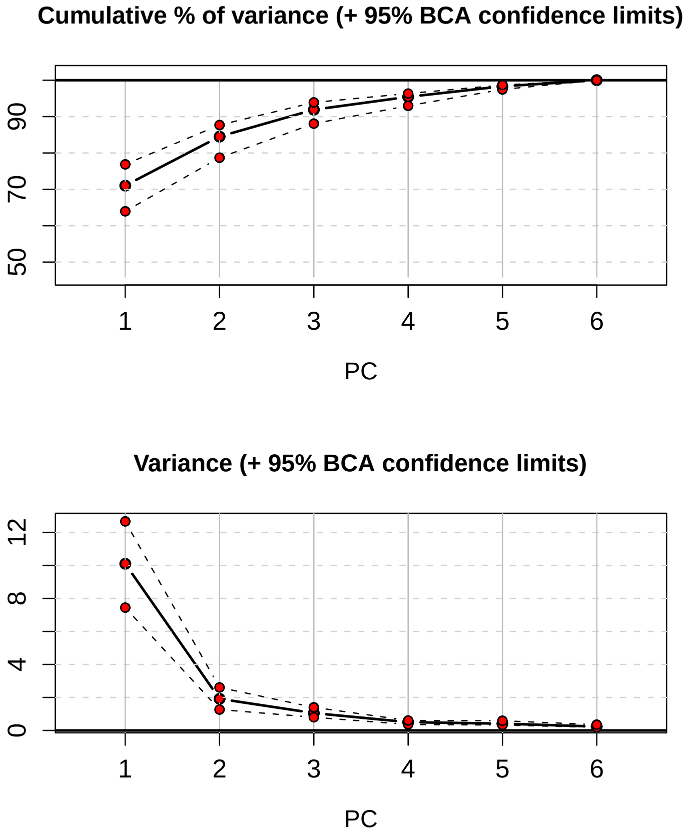

Figure 2: Bank notes data. Percentage of variance explained with FRB-based confidence intervals. Result of `plot` method on object of type `FRBpca`.

The output is intended to be self-explanatory. The confidence intervals shown are of the BCa type, which is the default in all applications. If interested, basic bootstrap intervals can be asked for instead by the command `summary(res, confmethod = "basic")`. The intervals for the loadings are not listed but are available graphically through the `plot` method, together with the graphical FRB results for the angles and the percentages of explained variance:

```
R> plot(res)
```

The result of the `plot` method consists of various pages of output and the user is prompted before starting each new page. Figure 2 shows the first page, which in the top panel displays the (cumulative) percentage of variance explained by each component and in the bottom panel the variances in absolute terms, which are the eigenvalues. The FRB-based confidence intervals are indicated by the dashed lines. Again, basic bootstrap instead of BCa intervals can be requested by specifying `plot(res, confmethod = "basic")`.

We see in this example that the first two principal components seem to explain more than 80% of the total variation. However, the lower confidence limit is actually below 80%. In general, when selecting the number of components to retain for further analysis on the basis of such percentages, it may be safer to consider the lower limits instead of the estimated percentages.

In Figure 3 we have the second output page of the `plot` method. It shows for each principal

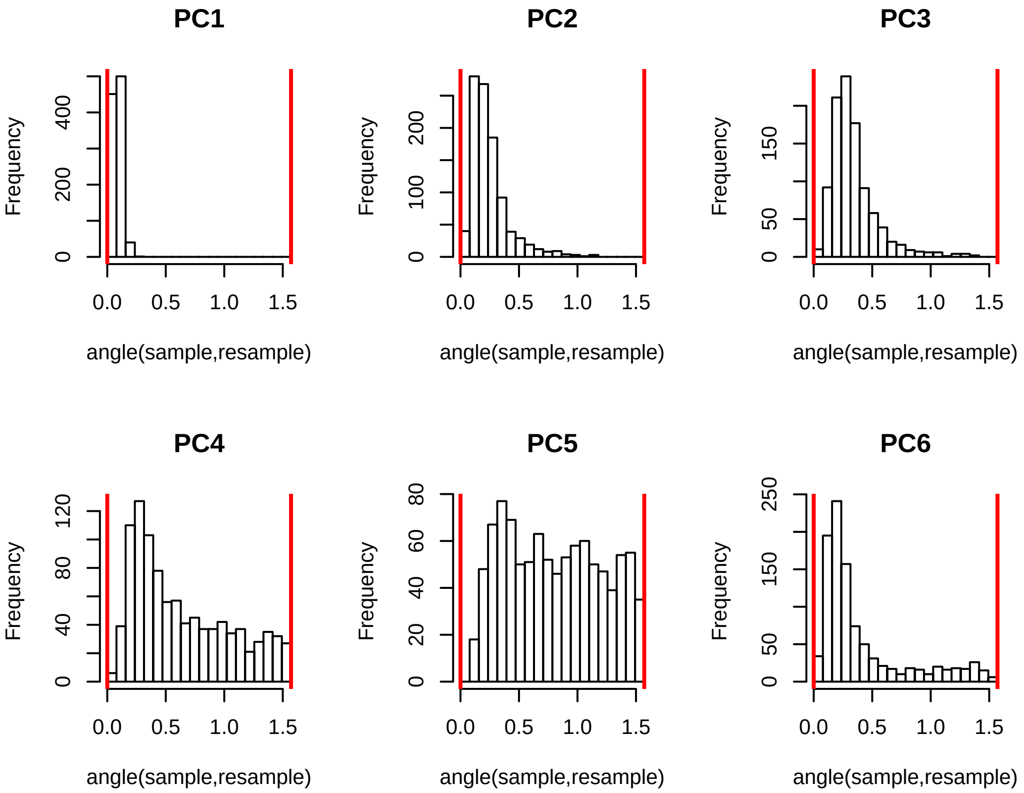

Figure 3: Bank notes data. Histograms of angles between original and bootstrap estimates of principal components. Result of `plot` method on object of type `FRBpca`.

component the histogram of the angles between the original component and the corresponding bootstrap components. The angle between two components is expressed by a value between 0 and $\pi/2$. These limits are indicated by thick vertical lines on the histograms. We see for example that the estimate of the first component is very much aligned with its bootstrap versions (most angles are close to zero) indicating low variability of that estimate. For the other components quite some more instability is observed.

The last type of graph presented by the `plot` method displays the loadings for a given principal component along with FRB-based confidence intervals. By default the first 5 principal components are shown, on separate pages. Figure 4 shows the graph corresponding to the first component. Next to the histogram of the angles, this yields another way of assessing the stability of the component. Note that the loadings are the coefficients of the normalized eigenvectors and hence lie within the interval $[-1, 1]$. We again conclude that the estimate of the first principal component should be quite accurate, since the confidence intervals are relatively small.

The three types of graphs produced by the `plot` method can also be requested through separate functions, called `plotFRBvars()`, `plotFRBangles()` and `plotFRBloadings()` respectively.

The purpose of using robust estimates instead of the classical ones is often twofold: (1) ensuring that the analysis is reasonably well protected against the influence of possible outliers,

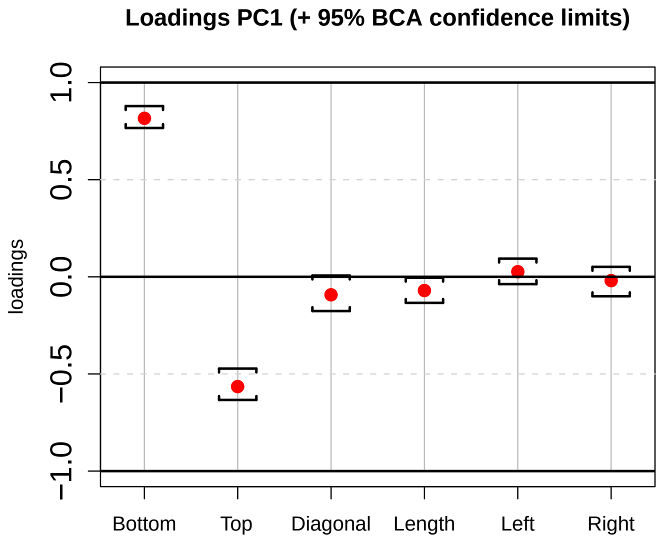

Figure 4: Bank notes data. Loadings of the first principal component with FRB-based confidence intervals. Result of `plot` method on object of type `FRBpca`.

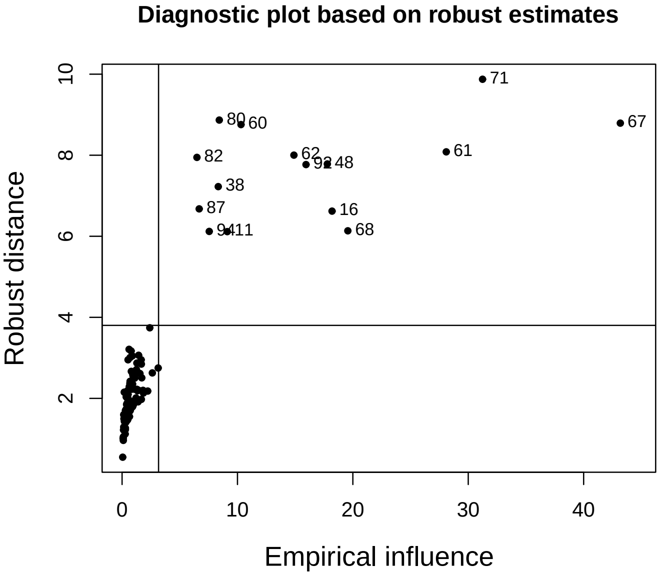

Figure 5: Bank notes data. Diagnostic plot. Result of `diagplot` method on object of type `FRBpca`.

and (2) detecting actual outliers in the data. For this last purpose, the package provides the `diagplot` method:

```
R> diagplot(res)
```

For objects of class `FRBpca` it draws by default the diagnostic plot of Pison and Van Aelst (2004), as explained in Section 4.4. The result is shown in Figure 5. The simpler plot of robust distances versus the index can be obtained through `diagplot(res, EIF = FALSE)`.

The horizontal cutoff line, here and in all other diagnostic plots, is drawn at the square root of the 97.5% quantile of the χ_p^2 distribution. We notice a total of 15 observations for which the robust distance clearly exceeds the cutoff and hence these should be regarded as outliers. The points also have a large empirical influence measure, which means that they would heavily influence the *classical* PCA analysis. Note that they did not have much influence on the robust PCA analysis because large robust distances by definition correspond to small weights in the robust estimates.

5.2. Hotelling test

Let us now consider the same data to illustrate the one-sample robust Hotelling test. We apply the test to demonstrate again the impact of the 15 outliers in this data set. That is, we formally test whether the robust test rejects the empirical (classical) mean of the data as the hypothesized true location. If so, we can conclude that the outliers severely influenced the empirical mean. We proceed as follows:

```
R> samplemean <- apply(ForgedBankNotes, 2, mean)
R> res <- FRBhotellingMM(ForgedBankNotes, mu0 = samplemean)
```

or using the formula interface

```
R> res <- FRBhotellingMM(
+    cbind(Length, Left, Right, Bottom, Top, Diagonal) ~ 1,
+    data = ForgedBankNotes, mu0 = samplemean)
```

Note that multivariate responses in a formula in R should be combined through `cbind`. An overview of the results is obtained by

```
R> summary(res)
```

which produces the following output:

```
One sample Hotelling test based on multivariate MM-estimates
(bdp = 0.5, eff = 0.95)

data:  ForgedBankNotes
T^2_R =  128.68
p-value =  0
Alternative hypothesis : true mean vector is not equal to
(214.823 130.3 130.193 10.53 11.133 139.45)

 95 % simultaneous confidence intervals for components of mean :
            Length   Left  Right Bottom    Top Diagonal
Lower bound 214.65 130.15 130.04 10.468 10.809   139.47
Upper bound 214.91 130.38 130.32 11.251 11.394   139.78

Sample estimates :
   location:
```

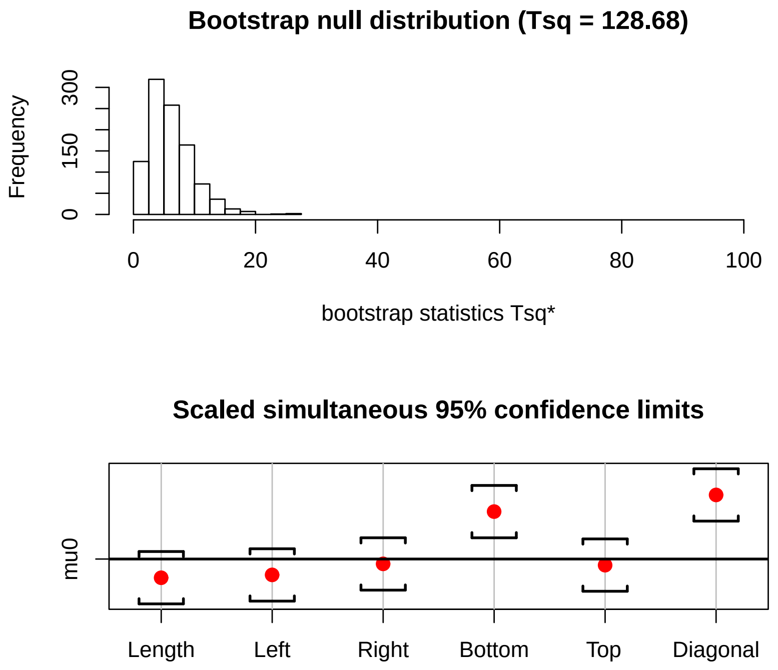

Figure 6: Bank notes data. Histogram of FRB test statistic (top) and scaled FRB-based simultaneous confidence intervals (bottom). Result of `plot` method on object of type `FRBhot`.

```
               Length   Left  Right Bottom    Top Diagonal
MM-loc vector 214.78 130.27 130.18  10.86 11.101   139.63

   covariance:
           Length    Left   Right  Bottom     Top Diagonal
Length     0.1010  0.0390  0.0368 -0.0671  0.0539   0.0489
Left       0.0390  0.0837  0.0597  0.0422 -0.0172   0.0441
Right      0.0368  0.0597  0.1163 -0.0136  0.0112   0.0651
Bottom    -0.0671  0.0422 -0.0136  0.9509 -0.5831  -0.1155
Top        0.0539 -0.0172  0.0112 -0.5831  0.5323   0.0313
Diagonal   0.0489  0.0441  0.0651 -0.1155  0.0313   0.1512
```

We see that $T_R^2 = 128.68$ with an FRB-based p value of 0, implying that all of the bootstrap T_R^{2*} values are smaller than 128.68. Note that by default the number of bootstrap samples is $R = 999$. In order to learn more about the difference between the robust location estimate and the hypothesized location vector (in this case the non-robust location estimate), one can consider the simultaneous confidence intervals displayed in the output. From these intervals we notice that the difference mainly lies in the variables `Bottom` and `Diagonal`. Interpreting these intervals should be somewhat easier in the graphical representation obtained via

```
R> plot(res)
```

for which the result is shown in Figure 6. The top panel, first, shows the histogram of the T_R^{2*} values, which gives a somewhat better idea of the magnitude of the T_R^2 value (if the value of

T_R^2 in the original sample would be below 100, it would be superimposed on the histogram). We may conclude that 128.68 is in fact huge.

The bottom panel then displays the confidence interval for each variable. In particular, each variable is scaled such that the interval has unit length and then centered around the hypothesized location vector `mu0`. In this way, all intervals can immediately be compared with regard to the significance of the difference between `mu0` (horizontal line) and the robust location estimate (bullets). As expected, we see that `Diagonal` exhibits the largest difference, followed by the `Bottom` variable. The least significant differences are found for variables `Right` and `Top`. Note that these confidence intervals are generally conservative in the sense that the simultaneous confidence level holds for all linear combinations of the location components and here only p of these are considered.

Further inspection of the data would indeed reveal that the 15 outliers have particularly deviating values in the variables `Bottom` and `Diagonal`. For example, these 15 bank notes tend to have a shorter diagonal than the rest of the notes. Note that the 15 outliers would show up again in the diagnostic plot obtained through `diagplot(res)`.

For an illustration of the two-sample Hotelling test, we turn to the Hemophilia data (Habemma, Hermans, and Van den Broek 1974), which are also available in the package:

```
R> data("hemophilia")
```

This data set contains two measurements on 75 women, belonging to two groups: $n_1 = 30$ of them are non-carriers and $n_2 = 45$ are known hemophilia A carriers. We would like to robustly test whether the difference between the two group means is significant. The MM-based test is obtained as follows:

```
R> grp <- as.factor(hemophilia[, 3])
R> x <- hemophilia[which(grp == levels(grp)[1]), 1:2]
R> y <- hemophilia[which(grp == levels(grp)[2]), 1:2]
R> res <- FRBhotellingMM(x, y)
```

Equivalently, using the formula interface, we have

```
R> res <- FRBhotellingMM(cbind(AHFactivity, AHFantigen) ~ gr,
+    data = hemophilia)
```

The short output, via the `print` method, is

```
R> res

        Two sample Hotelling test based on multivariate MM-estimates (bdp =
        0.5, eff = 0.95) (common covariance estimated by He and Fung method)

data:  x and y
T^2_R = 79.0532, p-value < 2.2e-16
alternative hypothesis: true difference in mean vectors is not equal to (0,0)

sample estimates:
```

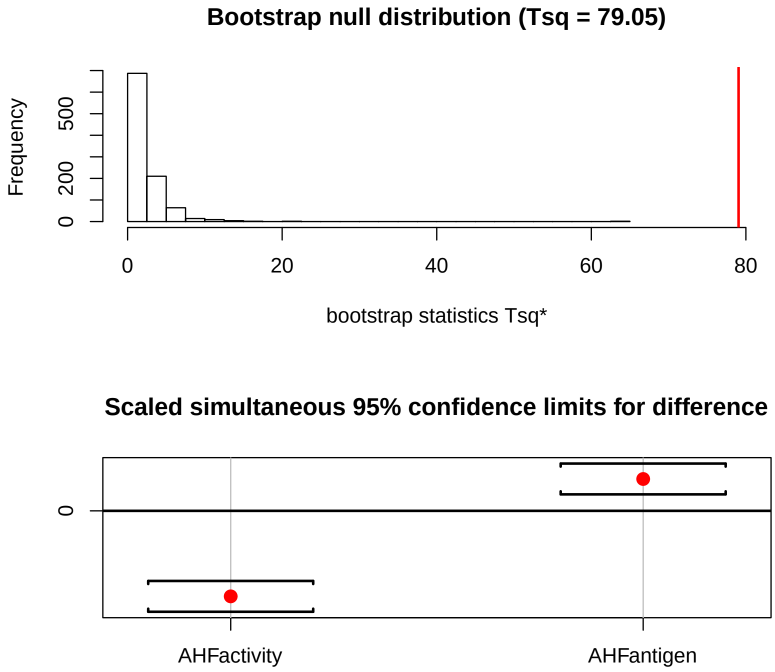

Figure 7: Hemophilia data. Histogram of FRB test statistic (top) and scaled FRB-based simultaneous confidence intervals (bottom). Result of `plot` method on object of type `FRBhot`.

```
                   AHFactivity AHFantigen
MM-loc x-vector         -0.305     -0.006
MM-loc y-vector         -0.128     -0.071
```

We find an extremely small p value, such that the difference is highly significant. More details are available via the `summary` method or graphically via

```
R> plot(res)
```

the result of which can be seen in Figure 7. It is clear from the top panel that $T_R^2 = 79.0532$, indicated by the thick vertical line, represents quite a large value. In the bottom panel the scaled simultaneous confidence limits are shown, which reveal that the means differ significantly in both variables, although the difference is relatively larger for the `AHFactivity` component.

Wondering whether the data contain any outliers, we examine the robust distances through the `diagplot` method:

```
R> diagplot(res)
```

The diagnostic plot is shown in Figure 8 and suggests that the data set is more or less outlier-free. Note that the robust distances are plotted versus their index within the group. The dashed line separates the two groups.

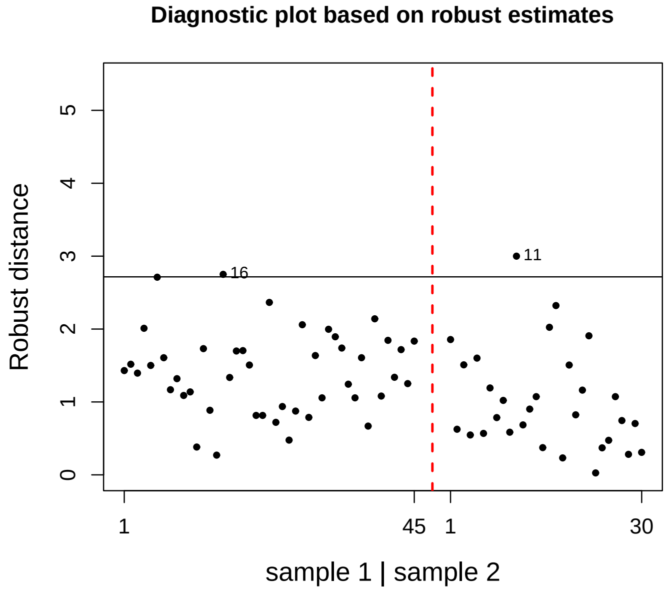

Figure 8: Hemophilia data. Diagnostic plot. Result of `diagplot` method on object of type `FRBhot`.

5.3. Multivariate linear regression

Finally consider the School data from Charnes, Cooper, and Rhodes (1981) for an example of robust multivariate regression:

```
R> data("schooldata")
```

The data consist of $q = 3$ response variables (scores on three different tests) and $p = 5$ explanatory variables, all measured for 70 school sites. For multivariate MM-regression with FRB inference, we use the function `FRBmultiregMM()` as follows.

```
R> res <- FRBmultiregMM(cbind(reading, mathematics, selfesteem) ~ .,
+    data = schooldata, R = 999, conf = 0.95)
```

Alternatively to the formula interface it would also be possible to pass the **x** and **y** data matrices. Extended formatted output is again available through the `summary` method:

```
R> summary(res)

Multivariate regression based on MM-estimates (bdp = 0.5, eff = 0.95)

Call:
FRBmultiregMM(formula = cbind(reading, mathematics, selfesteem) ~     .,
data = schooldata, R = 999, conf = 0.95)

Response reading:
```

```
Residuals:
    Min      1Q  Median      3Q     Max
-13.826  -1.560   0.416   2.891  27.861

Coefficients:
            Estimate   Std.Error   p-value
(Intercept)   2.1957      1.0428   0.03035   *
education     0.1259      0.0769   0.07348   .
occupation    5.0490      1.3752   0.00145  **
visit        -0.0441      0.3967   0.83121
counseling   -0.7290      0.2011   0.00000 ***
teacher      -0.1677      0.1471   0.18875
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

p-values based on BCA method!


Response mathematics:

Residuals:
   Min     1Q Median     3Q    Max
-9.096 -2.078 -0.240  3.485 40.312

Coefficients:
            Estimate   Std.Error   p-value
(Intercept)   2.7546      1.0479   0.00693  **
education     0.0490      0.0813   0.46411
occupation    5.6821      1.3304   0.00000 ***
visit        -0.0162      0.3472   0.95402
counseling   -0.7422      0.2405   0.00649  **
teacher      -0.2384      0.1646   0.05695   .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

p-values based on BCA method!


Response selfesteem:

Residuals:
   Min     1Q Median     3Q    Max
-2.099 -0.585  0.135  0.883  4.925

Coefficients:
            Estimate   Std.Error   p-value
(Intercept)  0.27534      0.2746   0.37389
```

```
education   -0.01146      0.0232   0.65695
occupation   1.63797      0.3082   0.00000 ***
visit        0.24373      0.0862   0.00215  **
counseling   0.00646      0.0726   0.90406
teacher      0.03407      0.0361   0.26380
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

p-values based on BCA method!

Robust residual scale: 1.82

Error covariance matrix estimate:
            reading mathematics selfesteem
reading       10.56        9.84       2.12
mathematics    9.84       14.29       1.88
selfesteem     2.12        1.88       1.10
```

For each of the three responses, the output shows the values for the MM-estimates of the regression coefficients with the corresponding FRB standard error and p value for these coefficients. Significance is indicated by the usual codes. We conclude, for example, that the coefficient for `occupation` is significant for each of the three responses, and the same holds for `counseling` except for the response `selfesteem`.

The `plot` method provides a graphical representation of these results. Here we request the FRB results for all explanatory variables except the intercept (by specifying `expl = 2:6`).

```
R> plot(res, expl = 2:6)
```

If desired, the user can also specify which response variables to include, as would be useful in case q is large. Figure 9 shows the result of the above request. For each coefficient the bootstrap distribution is shown in a histogram and the confidence limits are superimposed. By default the BCa method is used for the intervals, but as before the `confmethod` argument can be used in both `summary` and `plot` to obtain the basic bootstrap intervals instead. The confidence level is as specified in the call to `FRBmultiregMM`(). The coefficients which are significantly different from zero on this specific level are indicated in the graph by the red color and a star in the corresponding title. For example, the second row of plots stands out because these are the coefficients corresponding to the predictor `occupation`, which is significant for all three responses.

For outlier diagnostics we again apply the `diagplot` method. By default, this function first computes the MM-estimates of location and covariance in the space of the explanatory variables, based on which it computes the robust distances in the explanatory space. It then plots the residual distances versus these *leverages* (see also Section 4.4). The additional computations are time-consuming and can be avoided by setting the argument `Xdist = FALSE`, in which case the residual distances would simply be plotted against the index of the observations. Here we choose the default option:

```
R> diagplot(res)
```

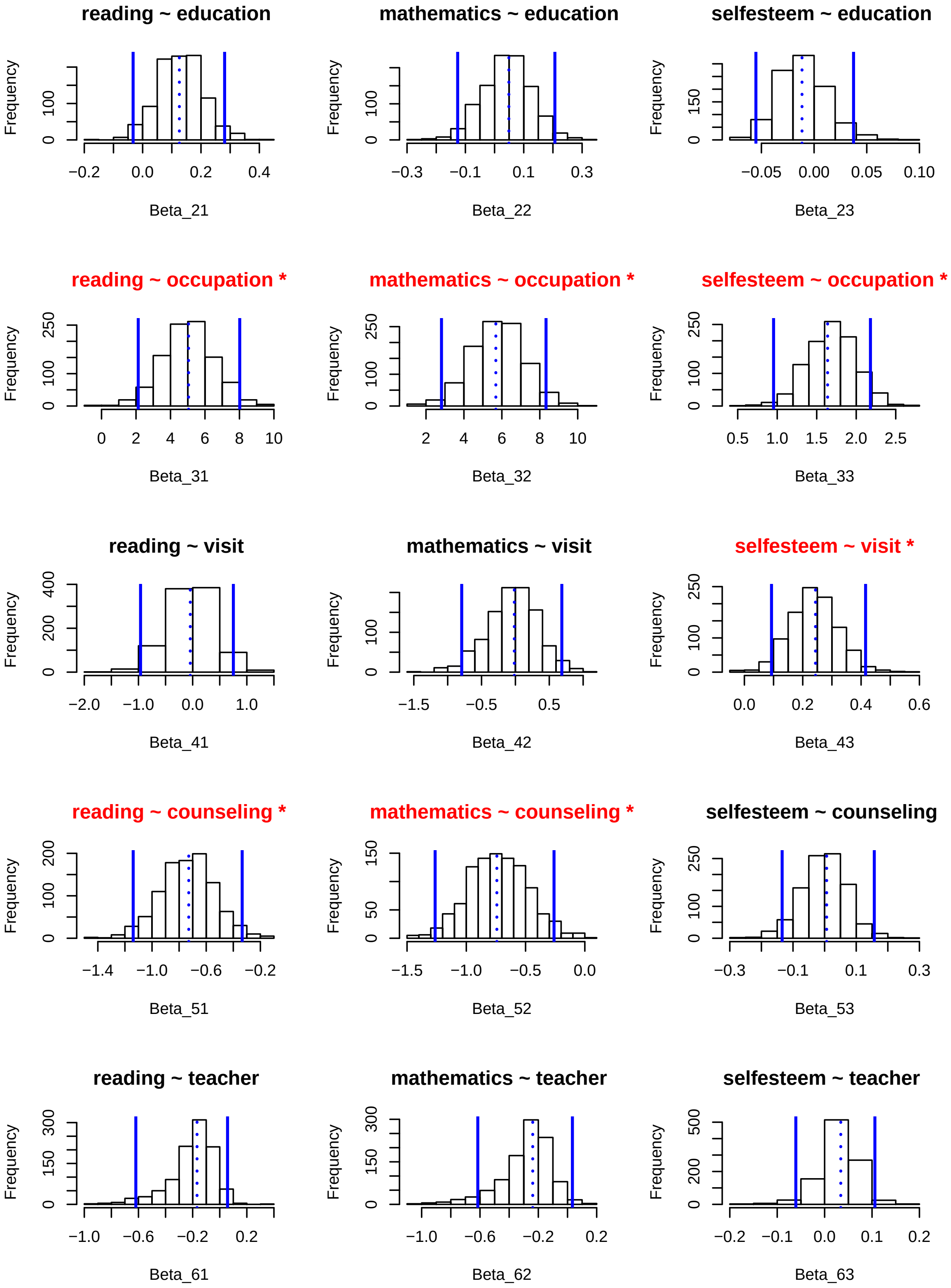

Figure 9: School data. Histograms of FRB coefficients with confidence intervals. Result of `plot` method on object of type `FRBmultireg`.

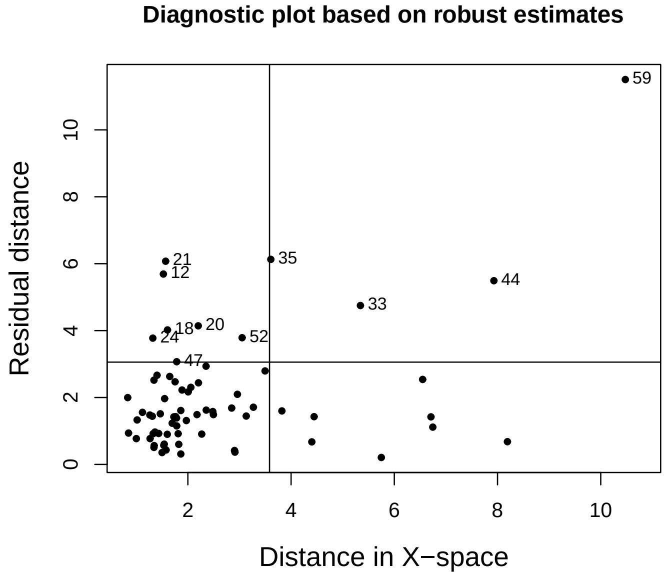

Figure 10: School data. Diagnostic plot. Result of `diagplot` method on object of type `FRBmultireg`.

Figure 10 contains the resulting diagnostic plot. It reveals at least one large outlier, observation 59, which can be categorized as a bad leverage point. Other observations which deserve some attention based on this plot are 12, 21, 33, 35 and 44. We conclude that a classical least squares analysis is likely to be overly influenced by a few outliers and especially by observation 59. Note that the outlier diagnostics can also be obtained without applying the FRB inference, but only based on the point estimates as follows:

```
R> res <- MMest_multireg(cbind(reading, mathematics, selfesteem) ~ .,
+    data = schooldata)
R> diagplot(res)
```

6. Conclusion

In this paper we provided some background on the fast and robust bootstrap method and we introduced the R package **FRB** for robust multivariate inference.

Currently all functions in the package are written in plain R code. To speed up the computations, future work will therefore include replacing some of the code by an implementation in a lower-level language such as C, in particular the fast-S algorithm for multivariate regression. Furthermore, the recently introduced robust MANOVA tests with the FRB method (Van Aelst and Willems 2011) are intended to be added to the package in the near future. Other applications and developments of FRB will be added in updates of the package when they become available.

References

Berrendero JR, Mendes BVM, Tyler D (2007). "On the Maximum Bias Functions of MM-Estimates and Constrained M-Estimates of Regression." *The Annals of Statistics*, **35**, 13–40.

Canty A, Ripley BD (2013). ***boot**: Bootstrap R (S-PLUS) Functions.* R package version 1.3-9, URL http://CRAN.R-project.org/package=boot.

Charnes A, Cooper WW, Rhodes E (1981). "Evaluating Program and Managerial Efficiency: An Application of Data Envelopment Analysis to Program Follow Through." *Management Science*, **27**, 668–697.

Croux C, Rousseeuw PJ, Hössjer O (1994). "Generalized S-Estimators." *Journal of the American Statistical Association*, **89**, 1271–1281.

Davies L (1987). "Asymptotic Behavior of S-Estimators of Multivariate Location Parameters and Dispersion Matrices." *The Annals of Statistics*, **15**, 1269–1292.

Davison AC, Hinkley DV (1997). *Bootstrap Methods and Their Applications.* Cambridge University Press, Cambridge. URL http://statwww.epfl.ch/davison/BMA/.

Flury B, Riedwyl H (1988). *Multivariate Statistics: A Practical Approach.* Chapman and Hall, London.

Habemma JDF, Hermans J, Van den Broek K (1974). "Stepwise Discriminant Analysis Program Using Density Estimation." In G Bruckmann, F Ferschl, L Schmetterer (eds.), *Proceedings in Computational Statistics, COMPSTAT 1974*, pp. 101–110. Physica Verlag, Heidelberg.

He X, Fung WK (2000). "High Breakdown Estimation for Multiple Populations with Applications to Discriminant Analysis." *Journal of Multivariate Analysis*, **72**, 151–162.

Hubert M, Rousseeuw PJ, Van Aelst S (2008). "High-Breakdown Robust Multivariate Methods." *Statistical Science*, **23**, 92–119.

Johnson RA, Wichern DW (1988). *Applied Multivariate Statistical Analysis.* Prentice Hall Inc., Englewood Cliffs.

Lopuhaä HP (1989). "On the Relation between S-Estimators and M-Estimators of Multivariate Location and Covariance." *The Annals of Statistics*, **17**, 1662–1683.

Maronna RA, Martin DR, Yohai VJ (2006). *Robust Statistics: Theory and Methods.* John Wiley & Sons, New York.

Maronna RA, Yohai VJ (1998). "Robust Estimation of Multivariate Location and Scatter." In S Kotz, C Read, D Banks (eds.), *Encyclopedia of Statistical Sciences Update Volume 2*, pp. 589–596. John Wiley & Sons, New York.

Pison G, Van Aelst S (2004). "Diagnostic Plots for Robust Multivariate Methods." *Journal of Computational and Graphical Statistics*, **13**, 310–329.

R Core Team (2012). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Roelant E, Van Aelst S, Croux C (2009). "Multivariate Generalized S-Estimators." *Journal of Multivariate Analysis*, **100**, 876–887.

Roelant E, Van Aelst S, Willems G (2008). "Fast Bootstrap for Robust Hotelling Tests." In P Brito (ed.), *COMPSTAT 2008: Proceedings in Computational Statistics*, pp. 709–719. Physica Verlag, Heidelberg.

Rousseeuw P, Croux C, Todorov V, Ruckstuhl A, Salibian-Barrera M, Verbeke T, Koller M, Maechler M (2012). ***robustbase**: Basic Robust Statistics*. R package version 0.9-4, URL http://CRAN.R-project.org/package=robustbase.

Rousseeuw PJ, Van Aelst S, Van Driessen K, Agulló J (2004). "Robust Multivariate Regression." *Technometrics*, **46**, 293–305.

Rousseeuw PJ, Van Zomeren BC (1990). "Unmasking Multivariate Outliers and Leverage Points." *Journal of the American Statistical Association*, **85**, 633–651.

Rousseeuw PJ, Yohai VJ (1984). "Robust Regression by Means of S-Estimators." In J Franke, W Härdle, RD Martin (eds.), *Robust and Nonlinear Time Series Analysis*, number 26 in Lecture Notes in Statistics, pp. 256–272. Springer-Verlag, New York.

Salibian-Barrera M (2005). "Estimating the P-Values of Robust Tests for the Linear Model." *Journal of Statistical Planning and Inference*, **128**, 241–257.

Salibian-Barrera M, Van Aelst S, Willems G (2006). "PCA Based on Multivariate MM-Estimators with Fast and Robust Bootstrap." *Journal of the American Statistical Association*, **101**, 1198–1211.

Salibian-Barrera M, Willems G, Zamar R (2008). "The Fast-Tau Estimator for Regression." *Journal of Computational and Graphical Statistics*, **17**, 659–682.

Salibian-Barrera M, Zamar RH (2002). "Bootstrapping Robust Estimates of Regression." *The Annals of Statistics*, **30**, 556–582.

Schäfer J, Opgen-Rhein R, Zuber V, Ahdesmäki M, Silva APD, Strimmer K (2012). ***corpcor**: Efficient Estimation of Covariance and (Partial) Correlation*. R package version 1.6.4, URL http://CRAN.R-project.org/package=corpcor.

Tatsuoka KS, Tyler DE (2000). "On the Uniqueness of S-Functionals and M-Functionals under Nonelliptical Distributions." *The Annals of Statistics*, **28**, 1219–1243.

Todorov V, Filzmoser P (2009). "An Object-Oriented Framework for Robust Multivariate Analysis." *Journal of Statistical Software*, **32**(3), 1–47. URL http://www.jstatsoft.org/v32/i03/.

Van Aelst S, Willems G (2005). "Multivariate Regression S-Estimators For Robust Estimation and Inference." *Statistica Sinica*, **15**, 981–1001.

Van Aelst S, Willems G (2011). "Robust and Efficient One-way MANOVA Tests." *Journal of the American Statistical Association*, **106**, 706–718.

Affiliation:

Stefan Van Aelst
Department of Applied Mathematics & Computer Science
Ghent University
Krijgslaan 281 - S9
9000 Ghent, Belgium
E-mail: Stefan.VanAelst@ugent.be

Journal of Statistical Software http://www.jstatsoft.org/
published by the American Statistical Association http://www.amstat.org/
Volume 53, Issue 3 *Submitted:* 2009-09-21
April 2013 *Accepted:* 2012-11-13