Finding High-dimensional Outliers with FastHCS

By Kaveh Vakili and Eric Schmitt

The High-dimensional Congruent Subset (HCS) is a new method for finding outliers in high-dimensional datasets. HCS is supported by FastHCS, a fast, rotation equivariant algorithm which we also detail. Both an extensive simulation study and three real data applications show that FastHCS performs better than its competitors.

1. Introduction. Outliers are observations that do not follow from the pattern of the majority of the data (Rousseeuw and van Zomeren, 1990). Identifying outliers is a major concern in data analysis for at least two reasons. First, because a few outliers, if left unchecked, will exert a disproportionate pull on the fitted parameters of any statistical model, preventing the analyst from uncovering the main structure in the data. Additionally, one may also want to find outliers to study them as objects of interest in their own right. In any case, detecting outliers when there are more than two variables is difficult because we can not inspect the data visually and must rely on algorithms instead.

Formally, this paper concerns itself with the problem of finding outliers in the context of high-dimensional data sets. The general setting is that of the usual variance decomposition model:

$$\boldsymbol{x}_i = \boldsymbol{\mu} + (\boldsymbol{x}_i - \boldsymbol{\mu})\boldsymbol{\Pi}_q\boldsymbol{\Pi}_q' + \boldsymbol{\epsilon}_i, \tag{1.1}$$

where the observations ($\boldsymbol{x}_i$'s) are draws from a p-variate elliptical distribution $\mathscr{E}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with finite second moments, location vector $\boldsymbol{\mu}$ and scatter matrix $\boldsymbol{\Sigma}$. We have that $\boldsymbol{\Sigma} = \boldsymbol{\Pi}\boldsymbol{\Lambda}\boldsymbol{\Pi}'$ for some diagonal matrix $\boldsymbol{\Lambda}$ with decreasing diagonal entries and $\boldsymbol{\Pi} : \boldsymbol{\Pi}'\boldsymbol{\Pi} = \boldsymbol{I}_p$ where $\boldsymbol{I}_p$ is the rank p diagonal matrix and $\boldsymbol{\Pi}$ is often called the loading matrix. Throughout, we posit

Keywords and phrases: High-dimensional Data, Outlier Detection, Principal Component Analysis, Computational Statistics

that there exist $q < p : \sum_{j=1}^{q} \boldsymbol{\Lambda}_{jj} >> \sum_{j=q+1}^{p} \boldsymbol{\Lambda}_{jj}$. Furthermore, we will denote the sample estimate of $\boldsymbol{\Pi}$ (resp. $\boldsymbol{\mu}$) as $\boldsymbol{P}$ (resp. $\boldsymbol{t}$) and the sub-matrix formed of the first q columns of $\boldsymbol{\Pi}$ ($\boldsymbol{P}$) as $\boldsymbol{\Pi}_q$ ($\boldsymbol{P}_q$). Then, $||\boldsymbol{\epsilon}_i||$ is the orthogonal distance of $\boldsymbol{x}_i$ to the subspace spanned by the columns of $\boldsymbol{\Pi}_q$. Finally, $\boldsymbol{X}$; the data matrix, is a collection of n p-vectors $\boldsymbol{x}_i$, at least $h = \lceil (n+q+1)/2 \rceil$ of which are well fitted by Model (1.1) and the remainder which are the outliers. A more comprehensive treatment of this topic can be found in specialized textbooks (Maronna et al. (2006), for example)

In this article we introduce FastHCS, a new procedure for fitting the parameters of Model (1.1) when p is large (potentially even larger than n) and the data may contain outliers. We also detail FastHCS, a fast algorithm for computing it. FastHCS returns $(\boldsymbol{P}_q, \boldsymbol{t})$, each estimating the corresponding parameters of Model (1.1), as well as $\{\text{OD}(\boldsymbol{x}_i, \boldsymbol{t}, \boldsymbol{P}_q)\}_{i=1}^{n}$, an outlyingness index derived from them and measuring how much each observation departs from the multivariate pattern characterizing the bulk of the data. The FastHCS algorithm is rotation equivariant (meaning that the outlyingness ranking of the observations is not affected by rotations of the data) and can be computed efficiently for large values of p and n.

For easier configurations of outliers, we find that FastHCS yield similar results as state of the art outlier detection algorithms. When considering more difficult cases however, we find that the procedure we propose yields much better outcomes. In the next section we motivate and define the HCS outlyingness index and the FastHCS algorithm. Then, in Section 3 we compare FasHCS to several competitors on synthetic data. Finally, in Section 4 we demonstrate the use of FastHCS on three real data examples.

2. The FastHCS outlyingness index.

2.1. *Motivation.* Given a sample of high-dimensional, potentially contaminated data, the goal of FastHCS is to reveal the outliers. It is well known that this problem is also equivalent to that of finding a fit of Model (1.1) close to the one we would have found without the outliers. Indeed, in order to ensure that their orthogonal distances to the fitted model reveals them, it is necessary to prevent the outliers from pulling the fit in their direction. Other rotation equivariant algorithms with the same objective are ROBPCA (Hubert et al., 2005), Projection Pursuit PCA (PcaPP) (Croux and Ruiz-Gazen, 2005) and Spherical PCA (PcaL) (Locantore et al., 1999).

However, in tests and real data examples, we often find situations where the outliers completely sway the fit found by ROBPCA, PcaL and PcaPP, yielding models that do not correctly describe the multivariate pattern of the bulk of the data. Consider the following example. We generated 100 data points $\boldsymbol{x}_i \in \mathbf{R}^{50}$, 80 of which come from Model (1.1) with diagonal covariance matrix $\boldsymbol{\Sigma}$ and diagonal entries $(34, 21, 13, \ldots)$ up to $\boldsymbol{\Sigma}_{8,8} = 1$ and $(0.1, \ldots, 0.01)$ for the last 42 entries, so that in this case $q = 8$. The 20 remaining data points come from a concentrated cluster of observations, shifted from the main group along the subspace spanned by the qth column of $\boldsymbol{\Sigma}$. The four panels in Figure 1 depict the first and qth coordinates of the data matrix (the darker blue dots show the members of main group of 80 observations). The orange, solid ellipses in the first three panels depict the location vectors and scatter matrices fitted by ROBPCA (upper left), PcaPP (lower left) and PcaL (upper right). All three were computed using the `R` package `rrcov` (Todorov and Filzmoser, 2009) with default parameters, except the number of estimated components, q (called `k` in the `rrcov` implementation), which we set to 8 and the robustness parameter `alpha` for ROBPCA which we set to 0.5, the value yielding maximum robustness. In this example, the fits found by ROBPCA, PcaL and PcaPP do not adequately describe, in the sense of Model (1.1), any subset of the data. In effect, the fit found by these algorithms tries to accommodate the multivariate pattern of the outliers instead of focusing on the cluster containing the majority of the data. In particular, the fitted centers of symmetry (orange stars) are not located in areas of high data density, and the fitted ellipses appear visually distinct from the shape of the density contours (drawn as a dashed, dark blue ellipse) governing the distribution of the majority of the data. Remarkably, the outliers have pulled these fits so much in their direction that they actually seem well fitted by the model and, consequently, diagnostic tools based on these fits will not reliably expose them.

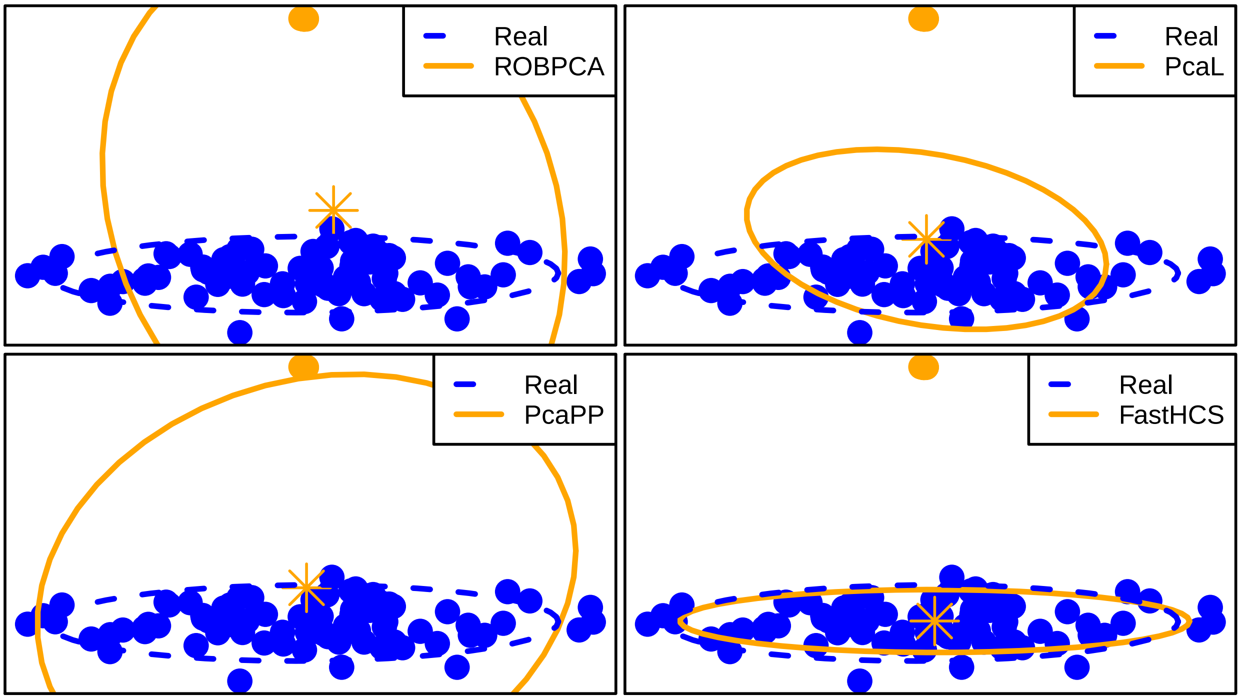

Fig 1. *The four panels depict the same data set. In each panel, the darker blue, dashed line shows the model governing the distribution of the majority –here 80 out of 100– of the observations. The solid orange line shows, respectively, the PCA model fitted by each algorithm.*

FastHCS fits the parameters of Model (1.1) using only a subset of the original data. This subset is selected among a large collection of random subsets according to a criterion. As criterion, FastHCS uses a new measure characterizing the degree of spatial cohesion of a cloud of points which we call the I-index. As we argue below, the main advantage of the I-index lies in its insensitivity to the spatial configuration of the outliers and this makes the FastHCS fit as well as the outlyingness index derived from it more reliable.

2.2. *Outlines of FastHCS.* FastHCS adapts the approach used by FastPCS, a multivariate index of outlyingness of a cloud of points introduced in Vakili and Schmitt (2014), to the high-dimensional setting by computing it on many random projections of the original data onto subspaces. An important feature of HCS and its algorithm, FastHCS, is that they are rotation and shift equivariant. That is, when a orthogonal transformation is applied to the data, the estimates of $\boldsymbol{\Pi}$ and $\boldsymbol{\mu}$ are transformed accordingly. More precisely, this means that for any p by p matrix $\boldsymbol{A}$ satisfying $\boldsymbol{A}' = \boldsymbol{A}^{-1}$ and any p vector $\boldsymbol{v}$, the FastHCS estimates

$(\boldsymbol{P}_q, \boldsymbol{t})$ computed on the transformed data $\boldsymbol{x}_i\boldsymbol{A}' + \boldsymbol{v}, i = 1, \ldots, n$ is $(\boldsymbol{AP}_q, \boldsymbol{At} + \boldsymbol{v})$. Although rotation and shift equivariance seem like natural properties in the context of Model (1.1), many outlier detection procedures designed for high dimensional settings do not in fact enjoy these properties. In the rest of this note we focus on rotation equivariant outlier detection procedures for two practical reasons. First, because they are computationally tractable in higher dimensional settings and second because one can succinctly quantify their robustness to outliers in the data. We illustrate this in section 3 where rotation equivariance allows us to dramatically reduce the number of scenario we need to consider to assess the robustness of the different algorithms without overly curtailing the generality of our conclusions. The FastHCS algorithm involves many steps. In this sub-section we offer a short outline of the principal ones, deferring specific discussion of implementation details to sub-section 2.4.

Given a dataset $\{\boldsymbol{x}_i\}_{i=1}^n$ and a version of Model (1.1) indexed by a value of q, FastHCS starts by projecting the data on the affine subspace spanned by these n observations. This step is mainly useful when the observations do not span their original p dimensional space and does not compromise the robustness of the overall procedure. Next, FastHCS draws M_q random subsets ('$(q+1)$-subsets') each of size $(q+1)$ of $\{1, \ldots, n\}$ denoted $\{H_0^m\}_{m=1}^{M_q}$ (this is step 'a' of Algorithm 2.4). Then, the algorithm grows each H_0^m into H^m, a subset of size h of $\{1, \ldots, n\}$ (the letter H without a subscript will always denote a subset of size h of $\{1, \ldots, n\}$). This is step b of Algorithm 2.4. The main innovation of our approach lies in the use of the I-index, a new measure we detail in Section 2.3, to characterize each of these H^m's. This is step (c) of Algorithm 2.4. Next, FastHCS selects H^*, the H^m having smallest I-index. Then, the h observations with indexes in H^* determine the so-called raw FastHCS fit. The parameter h determines the size of the subset used to obtain the preliminary (or so called raw) FastHCS estimates and represents an initial guess on the number of good observations in the data. Throughout this paper, h is set to $h = \lceil (n+q+1)/2 \rceil$ because this is the most conservative (in the sense of yielding maximum robustness) possible value of h for rotation equivariant algorithms based on rank-q basis pursuit (Lopuhaä and Rousseeuw , 1991). Finally, in the last stage of the algorithm, we apply a one step re-weighting to this raw FastHCS fit to get the final FastHCS fit. This re-weighting step aims to increase the statistical efficiency of the final fit without compromising its robustness. In one step re-weighting, this objective is achieved by re-fitting Model (1.1) to an enlarged subset of the data that includes the h data points used to obtain the raw fit as well as those observations that are close to it (in a sense that will be made precise in 2.4). Our algorithm depends on two additional parameters (K and M_q) but for clarity, these are not discussed in detail until Section 2.4. Below we introduce some additional notations we will use in the rest of the paper.

FastHCS always begins by computing the Singular Value Decomposition (SVD) of the centered original data, which we will denote as:

$$\operatorname*{svd}_{i=1}^{n}\left((\boldsymbol{x}_i-\boldsymbol{t}_n)/\sqrt{n-1}\right)=\boldsymbol{U}_r\boldsymbol{D}_r\boldsymbol{P}'_r,\quad \boldsymbol{t}_n=\operatorname*{ave}_{i=1}^{n}\boldsymbol{x_i}$$

with $r\leq\min(n,p)$. This creates a new rank r data matrix $\boldsymbol{Z}$ with n rows $\boldsymbol{z}_i,\ i=1,\ldots,n$

$$\boldsymbol{Z}=\boldsymbol{U}_r\boldsymbol{D}_r$$

This initial step has two properties. Firstly, it ensures that the overall algorithm is rotation equivariant. Secondly, it entails no loss of information or robustness, since it merely re-expresses the data in its own dimensionality. Given H_0^m a subset of indexes of size $q+1$ of $\{1,\ldots,n\}$, we will write the SVD decomposition of the observations corresponding to its members as:

$$\operatorname*{svd}_{i\in H_0^m}\left((\boldsymbol{z}_i-\boldsymbol{t}_0^m)/\sqrt{q}\right)=\boldsymbol{U}_q^m\boldsymbol{D}_q^m(\boldsymbol{P}_q^m)',\quad \boldsymbol{t}_0^m=\operatorname*{ave}_{i\in H_0^m}\boldsymbol{z}_i$$

From this decomposition, we compute the matrix $\boldsymbol{S}^m$ with n rows $\boldsymbol{s}_i^m,\ 1\leqslant i\leqslant n$:

$$\boldsymbol{s}_i^m=(\boldsymbol{z}_i-\boldsymbol{t}_0^m)\boldsymbol{U}_q^m,\quad \boldsymbol{t}_0^m=\operatorname*{ave}_{i\in H_0^m}\boldsymbol{z}_i$$

which is the projection of the (re-centered) rows of $\boldsymbol{Z}$ on $\boldsymbol{U}_q^m$, the subspace spanned by the first q (or less than q if the members of H_0^m happen to lie on a subspace) singular vectors of H_0^m.

2.3. *Computing the FastHCS Congruence Index.* We begin by detailing the computation of the I-index for a given matrix $\boldsymbol{S}^m$ and an h-subset H^m (in the next section we will cover how the algorithm constructs such an h-subset from an initial $q+1$-subset H_0^m). Denote $\boldsymbol{A}_k^m$ a $q \times q$ matrix formed of q rows of $\boldsymbol{S}^m$ (we detail below how we pick these q rows) and $\boldsymbol{a}_k^m$ a vector such that $\{\boldsymbol{a}_k^m : (\boldsymbol{A}_k^m)'\boldsymbol{a}_k^m = \boldsymbol{1}_q\}$ (the subscript k indexing these $\boldsymbol{a}_k^m$'s will be used later on), then, the squared orthogonal distance of $\boldsymbol{s}_i^m$ to $\boldsymbol{a}_k^m$ is:

$$d_i^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m) = ((\boldsymbol{s}_i^m)'\boldsymbol{a}_k^m - 1)^2 \Big/ ||\boldsymbol{a}_k^m||^2 \quad , \tag{2.1}$$

and we denote as H^{mk} the set of the indexes of the h smallest entries of $d_i^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m)$. More precisely, denoting $d_{(h)}$ the h-order statistics of a vector $\boldsymbol{d}$ we have:

$$H^{mk} = \{i : d_i^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m) \leqslant d_{(h)}^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m)\}$$

then, we define the *incongruence index* of an H^m along $\boldsymbol{a}_k^m$ as

$$I(H^m, \boldsymbol{S}^m, \boldsymbol{a}_k^m) = \log\left(\operatorname*{ave}_{i \in H^m} d_i^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m) \Big/ \operatorname*{ave}_{i \in H^{mk}} d_i^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m)\right) , \tag{2.2}$$

with the convention that $\log(0/0) := 0$. This index is always positive and will take a small value if the projection of the members of H^m along $\boldsymbol{a}_k^m$ greatly overlaps with that of the members of H^{mk}. To remove the dependence of Equation (2.2) on $\boldsymbol{a}_k^m$ we measure the incongruence of H^m by considering the average over many directions:

$$I(H^m, \boldsymbol{S}^m) = \operatorname*{ave}_{\boldsymbol{a}_k^m \in B(H^m, \boldsymbol{S}^m)} I(H^m, \boldsymbol{S}^m, \boldsymbol{a}_k^m) , \tag{2.3}$$

where $B(H^m, \boldsymbol{S}^m)$ is the collection of all directions orthogonal (in the space of the $\boldsymbol{S}^m$) to a hyperplane spanned by a $(q+1)$-subset of H^m. We call the H^m with smallest $I(H^m, \boldsymbol{S}^m)$ the *most congruent reduced rank subset*. In practice, it would be too laborious to evaluate Equation (2.3) over all members of $B(H^m, \boldsymbol{S}^m)$. A solution is to take the average over a random sample of K hyperplanes $\tilde{B}_K(H^m, \boldsymbol{S}^m)$ instead.

For a given direction $\boldsymbol{a}_k^m$, the value of $I(H^m, \boldsymbol{S}^m, \boldsymbol{a}_k^m)$ measures the size of the overlap between the projection along $\boldsymbol{a}_k^m$ of the members of H^m and those of H_k^m. We say that the observations with indexes in H^m form a spatially cohesive cloud of points if $\#\{H^m \cup H_k^m\}$ is large over many projections. Then, the I-index of an h-subset whose members form such a cloud of points will tend to be lower. In essence, the I-index (of a subset H^m) measures the size of the overlap between H^m and the H^{mk}'s (induced by H^m) over many random projections. This is because, for a projection onto $\boldsymbol{a}_k^m$, the members of H_k^m not in H^m will decrease the denominator in Equation (2.2) without affecting the numerator, increasing the overall ratio. Repeated over many projections, this causes the I-index of an h-subset containing the indexes of a collection of spatially cohesive observations to be smaller. This is illustrated in (Vakili and Schmitt, 2014), where the description of the behavior of the I-index used by FastPCS describes that of the rank reduced I-index used in FastHCS as well, provided that the p-vectors of observations $\boldsymbol{x}_i$ are replaced by the q-vectors of transformed data $\boldsymbol{s}_i^m$. Crucially, the I-index characterizes a cohesive h-subset of observations independently of the spatial configuration of the outliers. For example, this is illustrated in the fourth quadrant of Figure 1, where the fit found by FastHCS is not unduly attracted by members of the smaller cluster of concentrated data points. The finite sample breakdown point of PCS itself has been derived in (Schmitt et al., 2014).

In Sections 3 and 4, we show that using the I-index allows FastHCS to reliably find an uncontaminated fit and reveal the outliers, including in many situations where competing algorithms fail to do so. First though, the following section details the remaining steps of the FastHCS algorithm.

2.4. *A Fast Algorithm for the HCS.* In this section, we discuss the main steps of the FastHCS algorithm. Given a $n \times p$ matrix $\boldsymbol{X}$, Algorithm (2.4) returns an optimal subset H^* corresponding to the H^m with the smallest I-index. Then, given H^*, we get the so-called raw FastHCS estimates $(\boldsymbol{P}_q^*, \boldsymbol{t}^*)$ by fitting the parameters of Model (1.1) to the h observations with indexes in H^*:

$$\underset{i\in H^*}{\operatorname{svd}}\left((\boldsymbol{x}_i-\boldsymbol{t}^*)\Big/\sqrt{h-1}\right)=\boldsymbol{U}^*\boldsymbol{D}^*(\boldsymbol{P}^*)', \quad \boldsymbol{t}^* = \underset{i\in H^*}{\operatorname{ave}}\, \boldsymbol{x}_i$$

(2.4) Algorithm FastHCS

```
$(\boldsymbol{U}_r^m, \boldsymbol{D}_r^m) \leftarrow \underset{i=1}{\overset{n}{\text{svd}}}\left((\boldsymbol{x}_i - \boldsymbol{t}_n)/\sqrt{n-1}\right), \quad \boldsymbol{t}_n = \underset{i=1}{\overset{n}{\text{ave}}}\,\boldsymbol{x}_i$
$\boldsymbol{Z} \leftarrow \boldsymbol{U}_r \boldsymbol{D}_r$
for $m = 1$ to $M_q$ do:
$a$:  $H_0^m \leftarrow \{$random (q+1)$-$subset from $1:n\}$
      $(\boldsymbol{U}_q^m, \boldsymbol{D}_q^m) \leftarrow \underset{i \in H_0^m}{\text{svd}}\left((\boldsymbol{z}_i - \boldsymbol{t}_0^m)/\sqrt{q}\right), \quad \boldsymbol{t}_0^m = \underset{i \in H_0^m}{\text{ave}}\,\boldsymbol{z}_i$
      $\boldsymbol{s}_i^m \leftarrow (\boldsymbol{z}_i - \boldsymbol{t}_0^m)\boldsymbol{U}_q^m, \;\; 1 \leqslant i \leqslant n, \quad \boldsymbol{t}_0^m = \underset{i \in H_0^m}{\text{ave}}\,\boldsymbol{z}_i$
$b$:  for $w = 1$ to $W$ do:
          $D_i(H_{w-1}^m) \leftarrow \underset{k=1}{\overset{K}{\text{ave}}} \dfrac{d_i^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m)}{\underset{i \in H_{w-1}^m}{\text{ave}}\; d_i^2(\boldsymbol{a}_k^m, \boldsymbol{S}^m)}, \;\; 1 \leqslant i \leqslant n$
          set $\omega \leftarrow \lceil (n-q-1)w/(2W) \rceil + q + 1$
          set $H_w^m \leftarrow \left\{i : D_i(H_{w-1}^m) \leqslant D_{(\omega)}(H_{w-1}^m)\right\}$    (‘growing step’)
      end for
      $H^m \leftarrow H_W^m$
$c$:  compute $I(H^m, S^m) \leftarrow \underset{k=1}{\overset{K}{\text{ave}}}\, I(H^m, \boldsymbol{S}^m, a_k^m)$
end for
Keep $H^*$, the subset $H^m$ with lowest $I(H^m, \boldsymbol{S}^m)$.
```

FastHCS uses many random $(q+1)$-subsets as starting points. The number of these initial $(q+1)$-subsets, M_q, must be large enough to ensure that at least one of them is uncontaminated. Then, for each starting $(q+1)$-subset, the computational complexity scales as $\mathcal{O}(q^3 + nq)$. The value of M_q (and therefore the computational complexity of FastHCS) grows exponentially with q. In practice this means that FastHCS becomes impractical for values of q much larger than 25. Nevertheless, the overall time complexity of FastHCS grows with q, instead of p, making it a suitable candidate for high-dimensional applications. Furthermore, FastHCS belongs to the class of so called ‘embarrassingly parallel’ algorithms, i.e. its time complexity scales as the inverse of the number of processors meaning that it is particularly well suited to benefit from modern computing environments. To enhance user experience, we implemented FastHCS in C++ code wrapped in an R package (R Core Team, 2012) distributed through CRAN (package FastHCS).

For each of the M_q starting subsets H_0^m, step b the algorithm increases the size of H_w^m from $(q+1)$ (when $w=0$) to $h=\lceil(n+q+1)/2\rceil$ in W steps. This improves the robustness of the algorithm when outliers are close to the good data. We find that increasing W does not improve performance much if W is greater than 5, so we use $W=5$ as default.

Empirically also, we find that small values for K, the number of elements of $\tilde{B}_K(H^m)$, is sufficient to achieve good results and that we do not gain much by increasing K above 25, so we set $K=25$ as the default (this is the value we use throughout this paper). That such a small number of random projections suffice to reliably identify the outliers is remarkable. This is because FastHCS uses projections along directions generated by $(q+1)$-subsets of H^m rather than, say, indiscriminately from among the entire set of observations. Our choice always ensures a wider spread of directions when H^m is uncontaminated and this yields better results.

Then, in order to improve the accuracy of the algorithm in small, less severely contaminated samples, we add a so-called re-weighting step to FastHCS. In essence, given an optimal h-subset H^*, we get the final value of the parameters by fitting Model (1.1) to the members of a larger subset of observations H_*^+ itself derived from H^*. The objective is to allow those data points that are close enough to the raw fit to also contribute to the final fit. The motivation is that, typically, H_*^+ will encompass a greater share of the uncontaminated data than H^*.

To select the members of H_*^+, we use the orthogonal distance of each observation to the hyperplane spanning the columns of $\boldsymbol{P}_q^*$:

$$\mathrm{OD}(\boldsymbol{x}_i,\boldsymbol{t}^*,\boldsymbol{P}_q^*)=||\boldsymbol{x}_i-\boldsymbol{t}^*-(\boldsymbol{x}_i-\boldsymbol{t}^*)\boldsymbol{P}_q^*(\boldsymbol{P}_q^*)'|| \tag{2.5}$$

and the members of H_*^+ are the indexes of those data points satisfying:

$$H_*^+=\{i:\mathrm{OD}(\boldsymbol{x}_i,\boldsymbol{t}^*,\boldsymbol{P}_q^*)\leqslant c_h^{3/2}\}. \tag{2.6}$$

Where the cut-off in Equation (2.6) comes from the Wilson-Hilferty transformation of the orthogonal distances into approximately normally distributed random variables:

$$c_h = \underset{i\in H^*}{\text{ave}}\,\text{OD}(\boldsymbol{x}_i,\boldsymbol{t}^*,\boldsymbol{P}_q^*)^{2/3} + \Phi^{-1}(0.975)\sqrt{\underset{i\in H^*}{\text{var}}\,\text{OD}(\boldsymbol{x}_i,\boldsymbol{t}^*,\boldsymbol{P}_q^*)^{2/3}\Big/\chi^2_{(h-1)/n}} \tag{2.7}$$

where $\chi^2_{(h-1)/n}$ is the $(h-1)/n$ quantile of the χ^2 distribution with one degree of freedom. Finally, the FastHCS fit of the parameters are the entries $(\boldsymbol{P}_q,\boldsymbol{t})$ obtained as:

$$\underset{i\in H_*^+}{\text{svd}}\left((\boldsymbol{x}_i-\boldsymbol{t})\Big/\sqrt{\#\{H_*^+\}-1}\right)=\boldsymbol{UDP}',\quad \boldsymbol{t}=\underset{i\in H_*^+}{\text{ave}}\,\boldsymbol{x}_i.$$

and the FastHCS outlyingness index is the vector of values of $\text{OD}(\boldsymbol{x}_i,\boldsymbol{t},\boldsymbol{P}_q)$. As an additional output, FastHCS also produces an n-vector of so-called score distances:

$$\text{SD}(\boldsymbol{x}_i,\boldsymbol{t},\boldsymbol{P}_q)=\sqrt{(\boldsymbol{x}_i-\boldsymbol{t})'\boldsymbol{L}_q^{-1}(\boldsymbol{x}_i-\boldsymbol{t})}, \tag{2.8}$$

where $\boldsymbol{L}_q$ is the rank q diagonal matrix with $i^{th}, 1\leqslant i\leqslant q$, entry:

$$(\boldsymbol{L}_q)_{ii}=\underset{i\in H_*^+}{\text{ave}}\left((\boldsymbol{x}_i-\boldsymbol{t})'(\boldsymbol{P}_q)_{\cdot j}\right)^2,$$

and $(\boldsymbol{P}_q)_{\cdot j}$ denotes the jth column of $\boldsymbol{P}_q$.

The vector of orthogonal distance $\{\text{OD}(\boldsymbol{x}_i,\boldsymbol{t},\boldsymbol{P}_q)\}_{i=1}^n$ measures, for each observation, how far it lies from the q-dimensional hyperplane describing the multivariate pattern of the majority to the data. Then, observations with an O.D. greater than $c_h^{3/2}$ are flagged as outliers because, in the PCA context, they correspond to data points that are not well fitted by the model. In contrast, the S.D. distance in Equation (2.8) measures, for each observation, the amount of leverage it exerts on the fitted $\boldsymbol{P}_q$. Observations with high S.D. values (higher than $\sqrt{\chi^2_{0.975,q}}$ say) are called *good leverage* points because, in the PCA context, they increase one's faith in the fitted model. The scatter plot of the S.D. versus O.D. values (together with their respective cut-offs) form the so called PCA outlier map (Hubert et al., 2005), a standard diagnostic tool in the context of variance decomposition models.

3. Empirical Comparison: Simulation Study.

In this section we evaluate the behaviour of FastHCS quantitatively and contrast its performance to that of the ROBPCA, PcaPP and PcaL algorithms. The last three were computed using the `R` package `rrcov` with default settings except for the robustness parameter `alpha` for ROBPCA which we set to 0.5, the value yielding maximum robustness and the value of `k` which we set to q for all the algorithms. Our evaluation criteria is the (finite sample) bias, a quantitative measure of robustness of a fit.

3.1. *Finite sample bias.* Given a central, elliptical model $\mathscr{E}_p$ and an arbitrary distribution $\mathscr{F}_c$ (the index c stands for contamination), consider the ε-contaminated model

$$\mathscr{F}_\varepsilon = (1-\varepsilon)\mathscr{E}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}) + \varepsilon\mathscr{F}_c \ .$$

The (finite sample) bias measures the difference between $\boldsymbol{P}_q$ and $\boldsymbol{\Pi}_q$. For a rotation equivariant estimator of $\boldsymbol{\Pi}_q$, all the information about the bias is contained in the matrix $\boldsymbol{G}_q = \boldsymbol{P}_q\boldsymbol{\Pi}'_q$, or equivalently its condition number (Yohai and Maronna, 1990):

$$\text{bias}(\boldsymbol{P}_q) = \log \lambda_1 \left(\boldsymbol{G}_q\right) - \log \lambda_q \left(\boldsymbol{G}_q\right) \ ,$$

where λ_1 (λ_q) are the largest (qth) eigenvalues of $\boldsymbol{G}_q$ (to ensure that $\boldsymbol{G}_q \succ 0$ we use the convention, also used in `rrcov` package, of flipping the sign of $\boldsymbol{P}_q$ such that its largest-magnitude entry is always positive). Evaluating the maximum bias of $\boldsymbol{P}_q$ is an empirical matter: for a given sample, it depends on the dimensionality of the data, the rate of contamination by outliers as well as the distance separating them from the genuine observations. Finally, the bias also depends on the spatial configuration of the outliers (choice of $\mathscr{F}_c$). Fortunately, all the algorithms we compare are rotation equivariant, meaning that their behaviour is not affected by the off-diagonal entries of $\boldsymbol{\Sigma}_u$, so that w.l.o.g. we can focus on configurations where $\boldsymbol{\Sigma}_u$ is diagonal. Furthermore, because the effect of contamination is most harmful when the contaminating observation belongs to the subspace spanned by $\boldsymbol{\Pi}_q^{\perp}$ (the orthogonal complement of $\boldsymbol{\Pi}_q$) we can, w.l.o.g., concentrate on the much smaller class of configurations of outliers that satisfy these conditions.

3.2. *Outlier configurations.* To quantify the robustness of the four algorithms, we generate many contaminated data sets $\boldsymbol{X}_\varepsilon$ of size n with $\boldsymbol{X}_\varepsilon = \boldsymbol{X}_u \cup \boldsymbol{X}_c$ where $\boldsymbol{X}_u$ and $\boldsymbol{X}_c$ are, respectively, the contaminated and uncontaminated part of the sample. Our measure of robustness, the bias, depends on the distance between the outliers and the genuine observations which we will index by

$$\nu = \min_{i \in I_c} \sqrt{(\boldsymbol{x}_i' \boldsymbol{\Sigma}_u^{-1} \boldsymbol{x}_i)/\chi^2_{0.99,p}}\ . \tag{3.1}$$

(where $\chi^2_{0.99,p}$ denotes the 99th percentile of the χ^2 distribution with p degrees of freedom). The bias will also be affected by the diagonal elements of $\boldsymbol{\Sigma}_u$. To facilitate comparison, we consider parameterizations that are popular in the literature on rotation equivariant outlier detection methods:

- $\boldsymbol{D}_M$: the values of the diagonal elements slowly decrease from the first to the qth, and then drop sharply. More precisely, the values of the diagonal elements of $\boldsymbol{\Sigma}_u$ are $20(1+(1-j+q)/2),\ j=1,\ldots,q$ and $p^{-1}(p-j+1)+1,\ j=q+1,\ldots,p$. This is a generalization to arbitrary values of q of the parametrization used in (Maronna , 2005).
- $\boldsymbol{D}_H$: the values of the first q elements of the diagonal of $\boldsymbol{\Sigma}_u$ decrease exponentially and do not drop abruptly before the remaining, smaller, entries. More precisely, the first q entries of the diagonal of $\boldsymbol{\Sigma}_u$ are the first q Fibonacci numbers and the entries $q+1,\ldots,p$ are linearly decreasing as $(0.1,\ldots,0.001)$. This is a generalization to arbitrary values of q of to the parametrization used in (Hubert et al., 2005).

Then, given $\boldsymbol{\Sigma}_u$, the worst-case configurations (those causing the largest bias) of outliers are known. In increasing order of difficulty these are:

- Shift configuration: If we constrain the adversary to (a) $|\boldsymbol{\Sigma}_c| \geqslant |\boldsymbol{\Sigma}_u|$ and (b) place $\boldsymbol{X}_c$ at a distance ν of $\boldsymbol{X}_u$, then, to maximize the bias, the adversary will set $\boldsymbol{\Sigma}_c = \boldsymbol{\Sigma}_u$ (Theorem 1 in (Rocke and Woodruff, 1996)) and $\boldsymbol{\mu}_c$ in order to satisfy (b). Intuitively, this makes the components of the mixture the least distinguishable from one another.
- Point-mass configuration: If we omit the constraint (a) above but keep (b), the adversary will concentrate all the outliers around a single point at a distance ν from $\boldsymbol{X}_u$ (Theorem 2 in (Rocke and Woodruff, 1996)).

3.3. *Simulation parameters.* For the shift (point) configuration we generated the outliers from $\mathcal{N}_p(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$ and set $\boldsymbol{\Sigma}_c$ as $\boldsymbol{\Sigma}_u$ $(10^{-4}\boldsymbol{\Sigma}_u)$ and set $\boldsymbol{\mu}_c$ so that Equation (3.1) is satisfied. The complete list of simulation parameters follows:

- $p \in \{100, 400\}$, $q \in \{5, 10, 15\}$ and $n = 200$,
- The rate of contamination, ε, is one of $\{0.1, 0.2, 0.3, 0.4\}$,
- The configuration of the outliers is either shift or point,
- The distance ν separating the outliers from the genuine observations is one of $\nu = \{1, \dots, 10\}$,
- The diagonal of $\boldsymbol{\Sigma}_u$ is either $\boldsymbol{D}_M$ or $\boldsymbol{D}_H$,
- The number of initial $(q+1)$-subsets M_q is given by:

$$M_q = \frac{\log(0.01)}{\log(1 - (1 - \varepsilon_0)^{q+1})} ,$$

  with $\varepsilon_0 = 0.4$ so that the probability of getting at least one uncontaminated starting subset is always at least 99%.

In Figures 2 to 5, we display the bias curves as lattice plots (Deepayan, 2008) for discrete combinations of p, q and ε. In all cases, we expect the outlier detection problem to become monotonically harder as we increase p and ε, so little information will be lost by considering a discrete grid of a few values for these parameters. The configurations also depend on the distance separating the data from the outliers. Here, the effects of ν on the bias are harder to foresee: clearly nearby outliers will be harder to detect but misclassifying distant outliers will increase the bias more. Therefore, we will test the algorithms for many values (and chart the results as a function) of ν. For each algorithm, a solid colored line will depict the median, and a dotted line (of the same color) the 75th percentile of $\text{Bias}(\boldsymbol{P}_q)$. Each figure is based on 12000 simulations.

Figure 2 displays the bias curves corresponding to the fits found by the various algorithms for $p = 100$ when the diagonal of $\boldsymbol{\Sigma}_u$ is $\boldsymbol{D}_M$ and for the shift (right) and point mass configuration (left). Regardless of the spatial configuration of the outliers or the value of ε, the fits found by PcaPP and PcaL generally have high values of $\text{Bias}(\boldsymbol{P}_q)$, although PcaL obtains

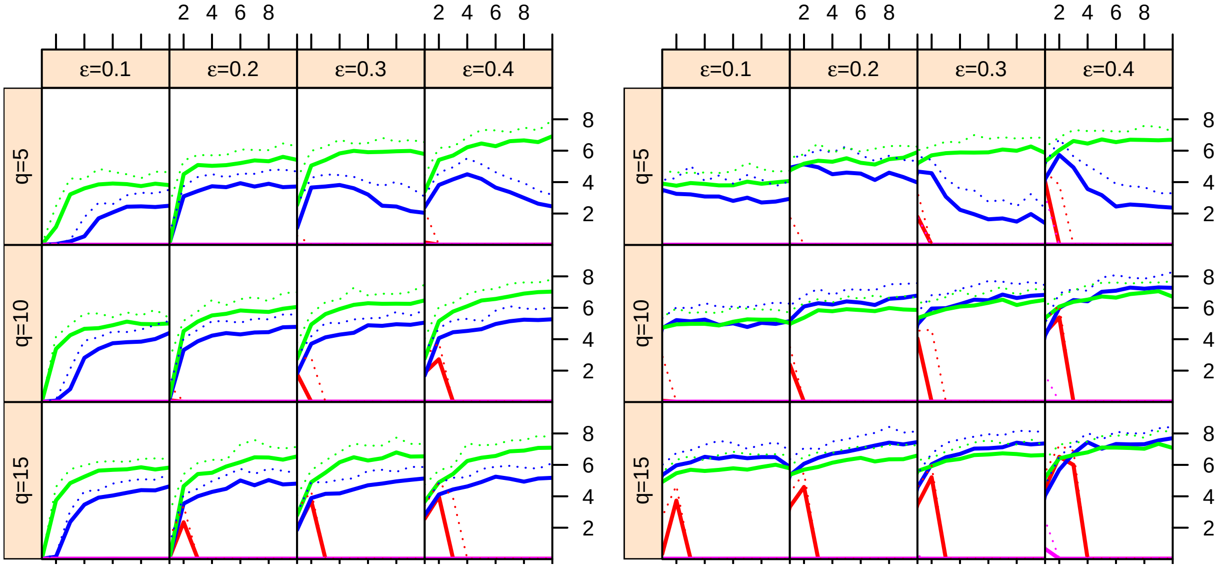

Fig 2. *Bias($\boldsymbol{P}_q$) for $p = 100$, $\boldsymbol{D}_M$ diagonal, shift contamination (left) and point mass contamination (right) as a function of ν. ROBPCA, PcaPP, PcaL, FastHCS.*

better results under point mass, and PcaPP slightly worse ones. The fit found by PcaPP consistently yields lower bias than that found by PcaL when the configuration is of the shift variety. In contrast, under point mass contamination the bias curves of PcaPP and PcaL are roughly of the same magnitude when $q \geqslant 10$, but with PcaL slightly outperforming PcaPP. The performance of ROBPCA is substantially better than the previous two algorithms. In this setting we find that the bias curves associated with the ROBPCA fit are significantly higher for low values of ν, where the outliers are harder to distinguish from the majority of the data. However, the bias curves associated with ROBPCA quickly re-descend as we consider higher values of ν. Although the performance of ROBPCA is generally good in this setting, it tends to be weaker in the presence of concentrated outliers, and also in situations where the rate of contamination of the sample (ε) is larger. In contrast, throughout all the panels of Figure 2, we see that the fit found by FastHCS consistently has low and flat bias curve irrespective of the rate of contamination and the dimensionality of the data set, as well as the spatial configuration or the degree of separation of the outliers.

In Figure 3 we repeat the previous experiment, but this time using the $\boldsymbol{D}_H$ diagonal instead. As we saw with the $\boldsymbol{D}_M$ diagonal, the performance of PcaPP remains consistently

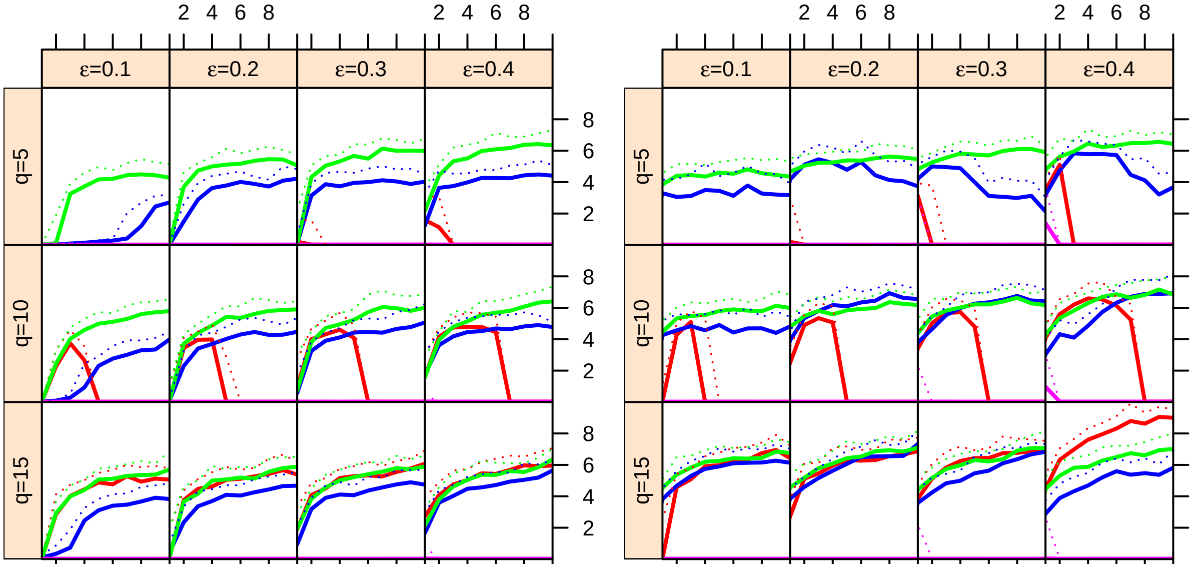

FIG 3. *Bias*($\boldsymbol{P}_q$) *for* $p = 100$, $\boldsymbol{D}_H$ *diagonal, shift contamination (left) and point mass contamination (right) as a function of* ν. *ROBPCA*, *PcaPP*, *PcaL*, *FastHCS*.

superior to that of PcaL for shift contamination, while for point mass contamination, PcaPP and PcaL alternately outperform one another. Compared to the $\boldsymbol{D}_M$ diagonal, the bias curves of both algorithms tend to be somewhat lower. Rather the opposite is true of ROBPCA, where the pattern of higher values of ν being associated with higher bias curves remains. In fact, when $q = 15$, regardless of the outlier configuration we even observe that the bias curves corresponding to the fit found by ROBPCA do not re-descend, even for for the larger values of ν we considered. ROBPCA also yields the most biased fit of the algorithms at various values of ν for many of the plots in Figure 3. Remarkably, we find that in this setting too, FastHCS is not affected by the degree of the separation or the spatial configuration of the outliers as it maintains low and flat bias curves throughout all settings we considered in Figure 3.

We next consider the high dimensional case of $p > n$. Figure 4 again depicts the simulation results for the situation where $\boldsymbol{\Sigma}$ has diagonal $\boldsymbol{D}_M$, but this time for $p = 400$. Although the bias curves corresponding to the fits found by PcaPP and PcaL are roughly of the same magnitude, we note that PcaPP is now unambiguously outperformed by PcaL when $q \geqslant 10$ and point mass contamination is present. When $p > n$ and the diagonal of $\boldsymbol{\Sigma}_u$ is $\boldsymbol{D}_M$, The

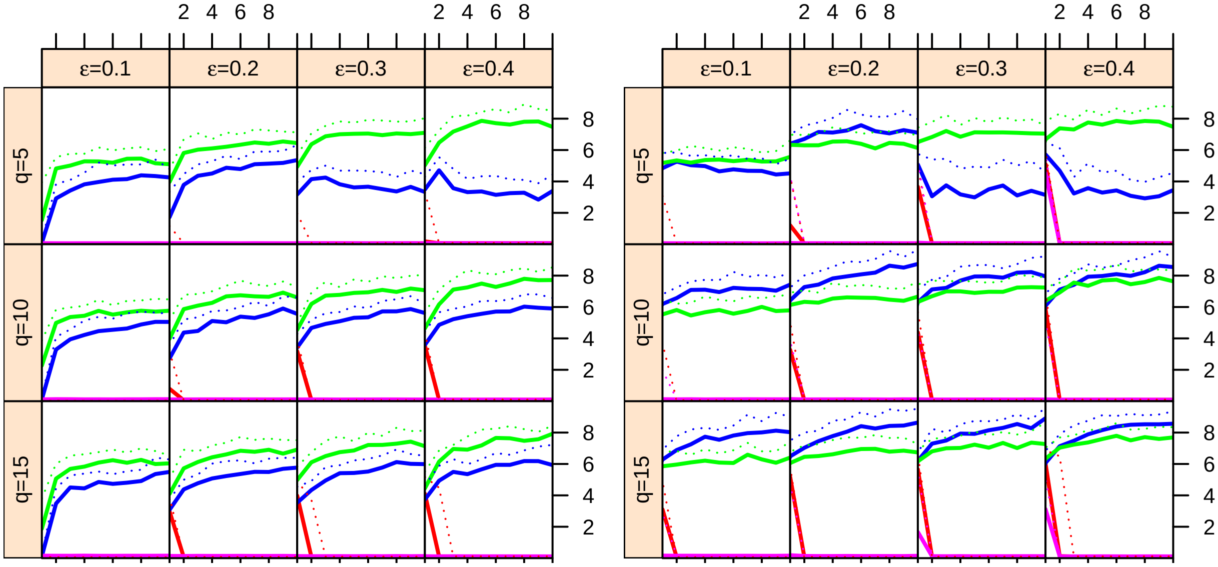

Fig 4. *Bias($\boldsymbol{P}_q$) for $p = 400$, $\boldsymbol{D}_M$ diagonal, shift contamination (left) and point mass contamination (right) as a function of ν. ROBPCA, PcaPP, PcaL, FastHCS.*

bias curves corresponding to the fit found by ROBPCA are again generally low (for the smaller values of ε and q) or descending rapidly as we set the outliers far enough from the genuine observations (e.g. for the larger values of ν). In the high-dimensional case too, we see that the bias curves for FastHCS remain again low and flat across values of ε, ν, p and q and regardless of whether the outliers are of the point-mass or shift variety.

Finally, in Figure 5 we examine the results when the $p > n$ and when we set the diagonal of $\boldsymbol{\Sigma}_u$ to $\boldsymbol{D}_H$. Between $q = 5$ and when $\varepsilon \leqslant 0.2$ and $q = 10$, PcaPP and PcaL trade places as the algorithm producing the fits with the largest biases. However, in subsequent settings, we find that the bias curve associated with the ROBPCA fit increases markedly, even becoming the highest for some or all values of ν. We see that as in all of the previous cases, the bias curves corresponding to the PcaPP and PcaL fits do not re-descend within the range of values of ν we considered, while those corresponding to the ROBPCA fits do so for $q \leqslant 10$, and for $q = 15$ when $\varepsilon \leqslant 0.2$ for point mass (and $\varepsilon \leqslant 0.3$ for shift) outliers. On the other hand, the bias curves corresponding to the ROBPCA fit also do not re-descend at all over the values of ν we considered when $q = 15$ and ε is large. In this situation too, we see that the fits found by FastHCS have low, flat bias curves when shift contamination is present.

When there is point mass contamination and $\varepsilon \geqslant 0.2$, some bias is visible for low values of ν, but the bias curves quickly re-descend as we increase the separation between the genuine data and the outliers.

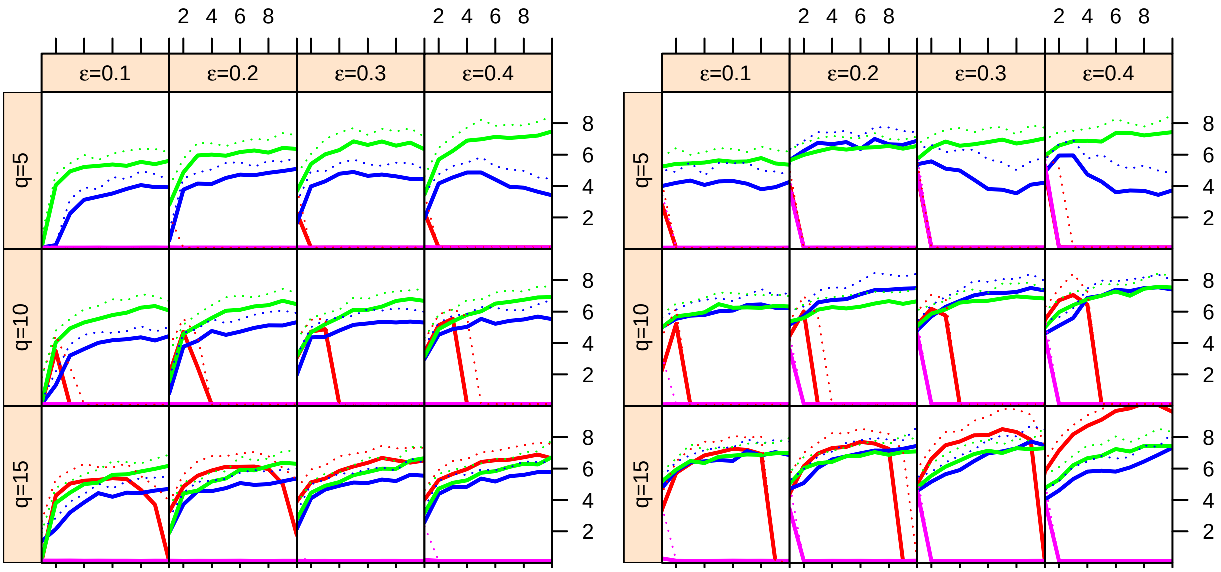

FIG 5. *Bias*($\boldsymbol{P}_q$) *for* $p = 400$, $\boldsymbol{D}_H$, *shift contamination (left) and point mass contamination (right) as a function of* ν. *ROBPCA*, *PcaPP*, *PcaL*, *FastHCS*.

In this note we focused on configurations of outliers that are worst cases for all the algorithms we compared, so as to to give a sense of the maximum bias that each of these methods can be expected to incur. Considering the results overall, we see that PcaPP and PcaL tend to produce the most biased fits amongst the algorithms we considered. The bias curves for these two algorithms do not exhibit any re-descending behavior in the settings we considered and generally remain high, even when we consider cases where the outliers are well separated from the genuine observations. Furthermore, the bias curves corresponding to the fit found by these two algorithms does not vary much across the configurations we considered in our tests. While PcaPP tends to perform slightly better overall, we find that in the presence of point-mass, PcaL sometimes achieves better results. ROBPCA produces markedly better results than the previous two algorithms in cases where the diagonal of $\boldsymbol{\Sigma}_u$ falls abruptly between it's qth and $(q+1)$th entry (corresponding to the $\boldsymbol{D}_M$ diagonal). However, in the seemingly more difficult situation where diagonal entry of $\boldsymbol{\Sigma}_u$ transitions smoothly between its noise and signal components (as in the $\boldsymbol{D}_H$ diagonal), we find that the bias curve corresponding to the fits found by ROBPCA remains high for large stretches of values of ν, especially when the rate of contamination of the sample is high. In contrast, throughout the settings we considered, FastHCS distinguishes itself by with bias curves that are not only lower than those of the other algorithms but also much more stable, being essentially unaffected by the choice of the parametrization for the diagonal entries of $\boldsymbol{\Sigma}_u$, the rate of contamination of the sample, the degree of separation between the outlier and the genuine observations, the extent to which the outliers are spatially concentrated, the value of p and the choice of q. To wit, we also observe that the bias curves corresponding to the fit found by FastHCS are also less variable: throughout the adversary configurations we considered, the 75th percentile of the bias corresponding to the FastHCS fit is typically closer to the median bias than is the case for the other algorithms. This result, repeated over many settings designed to be worst case, indicates that FastHCS is more robust, reliably returning fits that are very close to the ones we would have found without the outliers, including in many situations where the other methods fail to do so.

4. Empirical Comparison: Case Studies. In the previous section, we compared FastHCS to three state of the art outlier detection algorithms in situations that were designed to be as challenging as possible for rotation equivariant procedures. In this section, we will also compare FastHCS to ROBPCA, PcaPP and PcaL, but this time using three real data examples. We selected these three case studies because in each the observations in the data can be separated into two distinct subgroups from which we construct a majority and an outlier group. They are taken from three of the most common fields in which PCA is used: chemometrics, character recognition, and genetics. In all these examples, we set q, the number of estimated components, to a high value (subject to computational limitations). This is because for all methods, using higher values of q always helps reavealing the outliers but also results in higher computational costs. We use $q = 15$ in all the data examples to strike a balance between these two constraints. The implementations of ROBPCA, PcaPP and PcaL we use do not have an option to set the seed, but to ensure reproducibility of the results for FastHCS, we set `seed=1`. Furthermore, we include all three data sets used in this section in the FastHCS package. As in the simulations, we run each algorithm with default settings, except for the alpha parameter in ROBPCA which we set to 0.5 (the value yielding maximal robustness to outliers).

Each of the four algorithms produces an n-vector of orthogonal distances (O.D.), computed as in Equation (2.5), but using the parameters $(\boldsymbol{P}_q, \boldsymbol{t})$ fitted by each algorithm. Each method also produces a cut-off value for the O.D. values, $c_h^{3/2}$, with c_h computed as in Equation (2.7) (but again, based on the parameters fitted by each of the four algorithms). In all cases, observations with an O.D. greater than $c_h^{3/2}$ are flagged as outliers. In addition, each algorithm also returns an n-vector of score distances (S.D.) computed as in Equation (2.8). These outputs are combined to produce the usual PCA diagnostic plot associated with each of these fits. The former (orthogonal distance) emphasizes those observations that are not well described by the fitted model while the latter (score distances) emphasizes those observations that are have a large influence on the fitted parameters. To facilitate the visual comparison between the diagnostic plots returned by the different methods, we will display, the scaled (or standardized) PCA outlier map, which consists, for each algorithm, of a scatter-plot of the n-vector of orthogonal distances (returned by each algorithm) scaled by their respective cut-off values:

$$\tilde{\text{OD}}(\boldsymbol{x}_i, \boldsymbol{t}, \boldsymbol{P}_q) = \text{OD}(\boldsymbol{x}_i, \boldsymbol{t}, \boldsymbol{P}_q) \Big/ c_h^{3/2}$$

against the n-vector of score distances scaled by their respective cut-off values:

$$\tilde{\text{SD}}(\boldsymbol{x}_i, \boldsymbol{t}, \boldsymbol{P}_q) = \text{SD}(\boldsymbol{x}_i, \boldsymbol{t}, \boldsymbol{P}_q) \Big/ \sqrt{\chi^2_{0.975;q}}$$

4.1. *The Tablet Data.* We begin this section with the study of a data set from chemometrics. The tablet data (Dyrby et al., 2002) contains the results of an analysis on Escitalopram® tablets from the pharmaceutical company H. Lundbeck A/S using near-infrared (NIR) spectroscopy. The study includes tablets of four different dosages from pilot, laboratory and full scale production settings are included. Each tablet (the observations) is measured along 404 wavelengths (the variables). From this data, we extract two subsets of observations which we combine to obtain a new data set formed of two heterogeneous subgroups. Basically, we will use the rows corresponding to the 80 tablets with a nominal weight of 80mg as the majority group and the rows corresponding to the first 50 tablets with a nominal weight of 250mg will serve as the outliers. This yields a high-dimensional data set (i.e. $p > n$) with $n = 130$, $p = 404$ and a contamination rate of $\varepsilon = 38\%$.

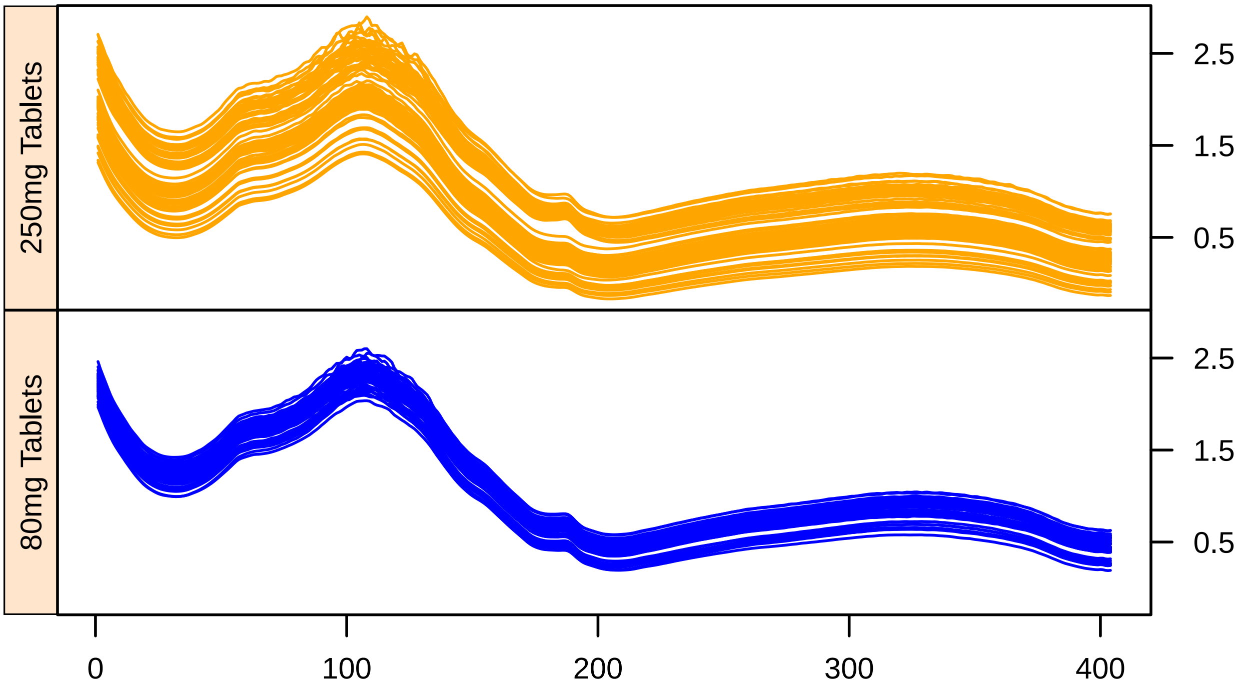

Fig 6. *Spectra of fifty 250mg (light orange) tablets and eighty 80mg (dark blue) tablets.*

Figure 6, depicts the spectra of the 250mg (orange) and 80mg (blue) tablets, respectively the outliers and majority group in this example. The spectra of the 250mg tablets follow a different multivariate pattern than those of the 80mg tablets. For example, the spectra of the former are generally lower and more spread out than the spectra of the latter. In Dyrby et al. (2002), the authors explain that accurate models for NIR analyses of medical tablets are valuable for quality control purposes, since they are fast, nondestructive, noninvasive, and require little preparation. In this exercise, the goal of the algorithms will be to find a fit that describes the 80mg tablets (the majority group) well, despite the presence in the sample of many observations corresponding to 250mg tablets.

Figure 7 depicts the diagnostic plots of the scaled outlyingness measures obtained from each of the algorithms. To enhance the distinction between the two groups in our data, we show the 80mg tablets as (dark) blue circles and the 250mg tablets as (light) orange triangles. We begin with the diagnostic plot corresponding to the fit found by PcaPP (shown in the lower left corner of Figure 7). Only seven observations are identified as outliers (have a scaled O.D. larger than 1), suggesting a modest contamination rate of only 5% of the sample. Visually, the scatter plot of S.D. and O.D. values corresponding to both types of tablets largely overlap. This shows that observations belonging to the two subgroups in the data are treated indiscriminately, as if they are realization of a single homogeneous process. Broadly speaking, the diagnostic plot corresponding to the PcaL fit, shown in the upper right corner of Figure 7, is similar to that of PcaPP. These diagnostic plots are similar because in both cases, the fits found by the algorithms strive to accommodate the two subgroups in the data. Consequently, the members of the cluster consisting of the 250mg tablets are allowed to exert a significant pull on the estimated subspaces returned by these algorithms and they now appear well fitted by both. The diagnostic plot corresponding to the ROBPCA fit (shown in the upper left corner of Figure 7) is somewhat different from the previous two. For example, contrary to the diagnostic plot returned by PcaPP and PcaL, we now see that when the data is projected on the subspace spanned by $\boldsymbol{P}_q^m$ (the subspace fitted by ROBPCA), some observations (all also belonging to the 80mg group) form a cluster, seemingly well separated from the rest of the data (these are the observations for which the standardized S.D. is greater than 1) and consequently exert a large influence on the ROBPCA fit. All in all the diagnostic plot corresponding to the ROBPCA fit flags 17 data points, or about 13% of the original sample, as outliers. This is a considerably higher rate than either PcaPP or PcaL. However, somewhat surprisingly, the observations flagged as outliers all correspond to 80mg tablets (the actual majority group in this data).

In contrast to the other algorithms, the diagnostic plot corresponding to the fit found by FastHCS clearly reveals the 250mg tablets (the actual minority group in this data set) as far flung outliers. The separation is quite clean, as all of the 250mg tablets have high O.D. values and only a few of the 80mg tablets end up being classified as outliers. This clear

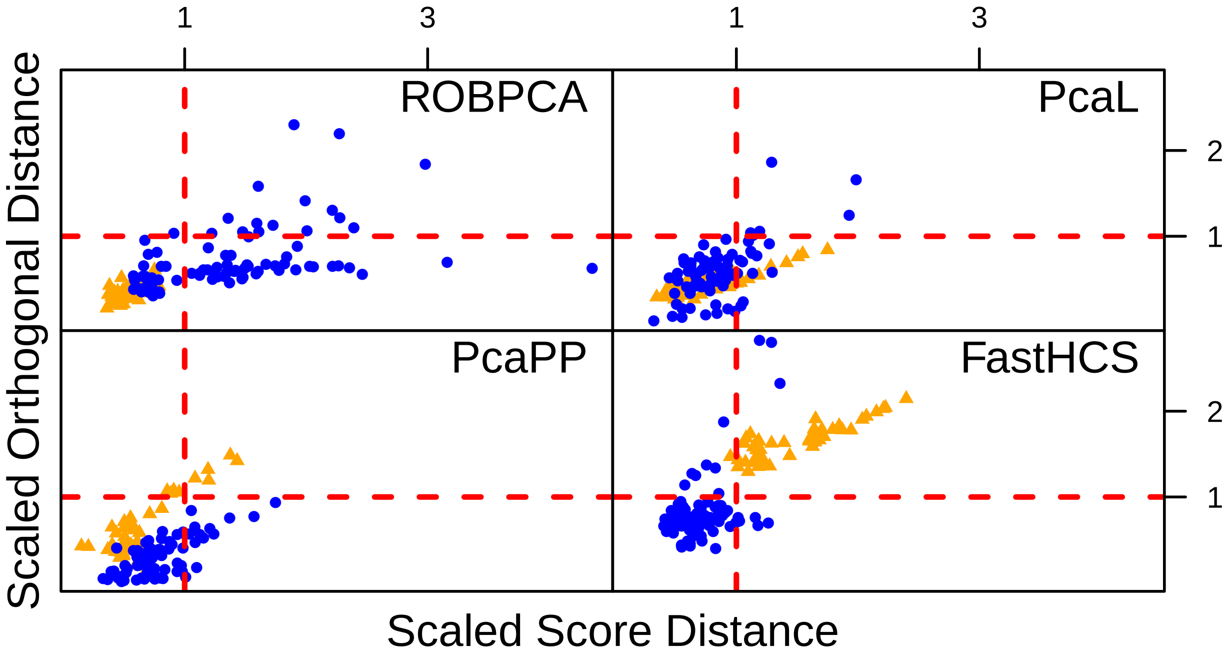

FIG 7. *Diagnostic plots of the scaled score and orthogonal distances of the measured spectra. The observations corresponding to 80mg (250mg) tablets are shown as dark blue circles (light orange triangles).*

separation between the two type of tablets in the diagnostic plot is remarkable and leads to two related but distinct interpretations. First, it shows that the subspace found by FastHCS does not try to accommodate all the observations in the data set but concentrates on the 80mg tablets instead, the true majority in this data set. Second, the diagnostic plot derived from the FastHCS solution corroborates the visual evidence from Figure 7 and establishes that the 250mg tablets do not follow the same multivariate patterns as (the majority of the) 80mg tablets and, in fact, depart very significantly from it. These two elements and the fact that the observations belonging to both types of tablets are allowed to indiscriminately influence the fit in the solution found by ROBPCA, PcaPP and PcaL, together imply that the parameters returned by FastHCS characterizes the multivariate pattern of the (bulk of the) 80mg tablets much more faithfully than those returned by the other algorithms.

4.2. *The Multiple Features Data Set.* Next, we will consider the Multiple Features Data Set (Van Breukelen et al., 1998). This data set contains many replications of hand written numerals ('0'-'9') extracted from nine original maps of a Dutch public utility. For each

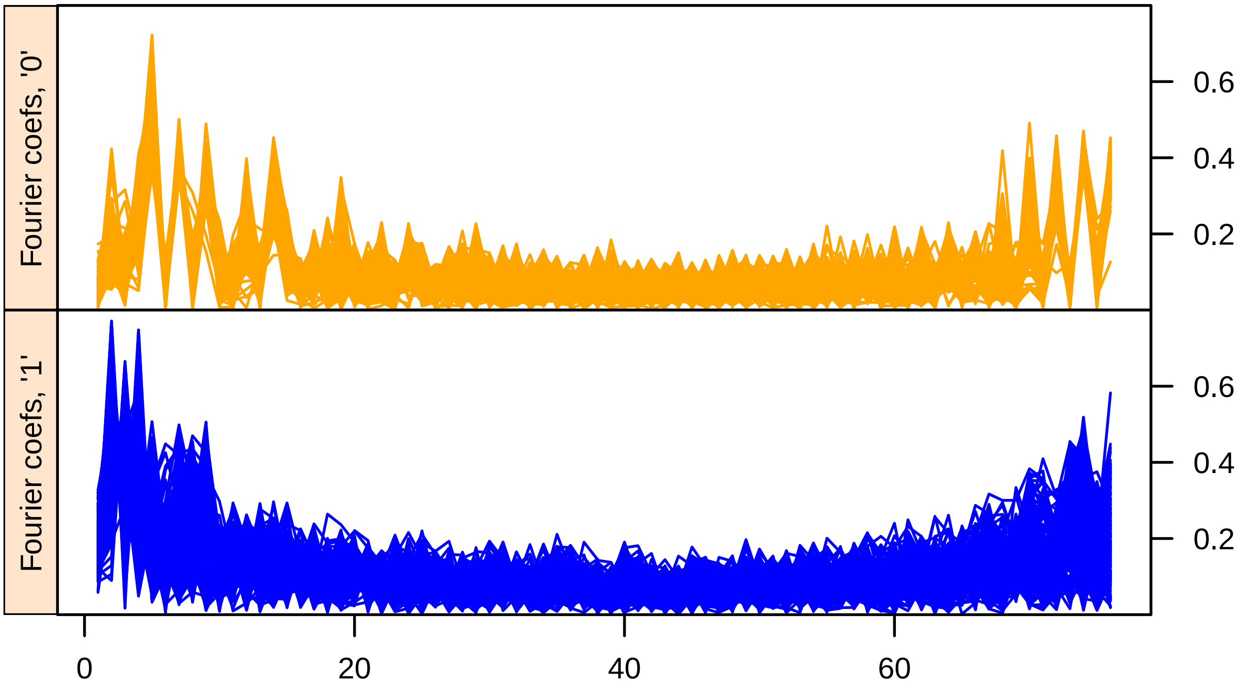

Fig 8. *The 350 vectors of Fourier coefficients of the character shapes. The first 150 curves (corresponding to observations with labels "0") are show in the top panel in orange. The Main group (200 curves) corresponding to observations with labels "1" are shown in the bottom panel.*

numeral, we have 200 replications (the observations) expressed as a vector of 76 of Fourier coefficients (the features) describing its shape. Finally, each numeral has been manually identified, yielding an extra vector of class labels. In this application, we will combine the vectors of Fourier coefficients corresponding to the 200 replications of the digit "1" to the vector of Fourier coefficients corresponding to the first 150 replications of the digit "0" (so that $n = 350$ and $p = 76$). As before, we do not include the class labels in our data matrix so that, in effect, the observations with label "0" are cast as the outliers and the task of the algorithms will be to expose them.

In order to give a qualitative impression of the differences between the two groups, we plot the Fourier coefficients corresponding to the main (outlier) subgroup of our data set in the bottom (top) panels of Figure 8 as dark blue (light orange) curves. In general, the curves corresponding to the members of the two groups are visually similar. In particular, the vertical ranges of both largely overlap, and both set of curves exhibit a similar pattern of variance clustering where the central 40 Fourier coefficients have systematically less dispersion than higher or lower ones.

Figure 9 depicts the resulting four (scaled) PCA outlier maps. Again, we assign to each observation a color (dark blue or light orange) and a plot symbol (round or triangle) depending on whether the corresponding curve describes a member of class "1" or "0", respectively. We begin by the (scaled) outlier map corresponding to the fit produced by PcaPP and PcaL (depicted in the lower left and upper right corners of Figure 9, respectively). They are visually similar and can be described together. The members of the two subgroups (corresponding to observations with label "0" and "1" respectively) are depicted as lumped together, preventing the true outliers from standing out on either of the diagnostic plots. For example, only 30 observations (all belonging to the majority subgroup, that of the curves corresponding to the observations with label "1") are flagged as outliers by PcaPP. For PcaL, only 5 observations are flagged as outliers (a single one of which belongs to the observations with label "0"). The scaled outlier map corresponding to the ROBPCA fit (shown in the upper left corner of Figure 9) classifies only 13 observations (or about 4% of the sample) as outliers. Of these, only 5 actually belong to the group of curves with label "0" (the true outliers in this experiment). Furthermore, the ROBPCA fit flags many of the members of the actual majority group as poorly fitted data points with large corresponding (adjusted) score distances. Consequently, the fit returned by none of these algorithm can be expected to adequately describe the pattern of the bulk of the data and that the diagnostic plots obtained from these fits cannot be used to reliably separate the data set into the true subgroups that constitute it.

Contrast these results with the scaled outlier map derived from the FastHCS fit (lower left corner of Figure 9). Here also, the outlier map corresponding to the fit found by FastHCS evinces a far richer structure in the data. Contrary to the other methods, the fit found by FastHCS cleanly discriminates between two heterogeneous groups, and consequently, the observations corresponding to zeros stand out through their large orthogonal distance from it. For example, all the 190 observations not flagged as outliers by FastHCS now only include members of the true majority group (e.g. those curve with label "1"). Furthermore, the outlier map associated with the FastHCS fit correctly flags all the true outliers as observations clearly

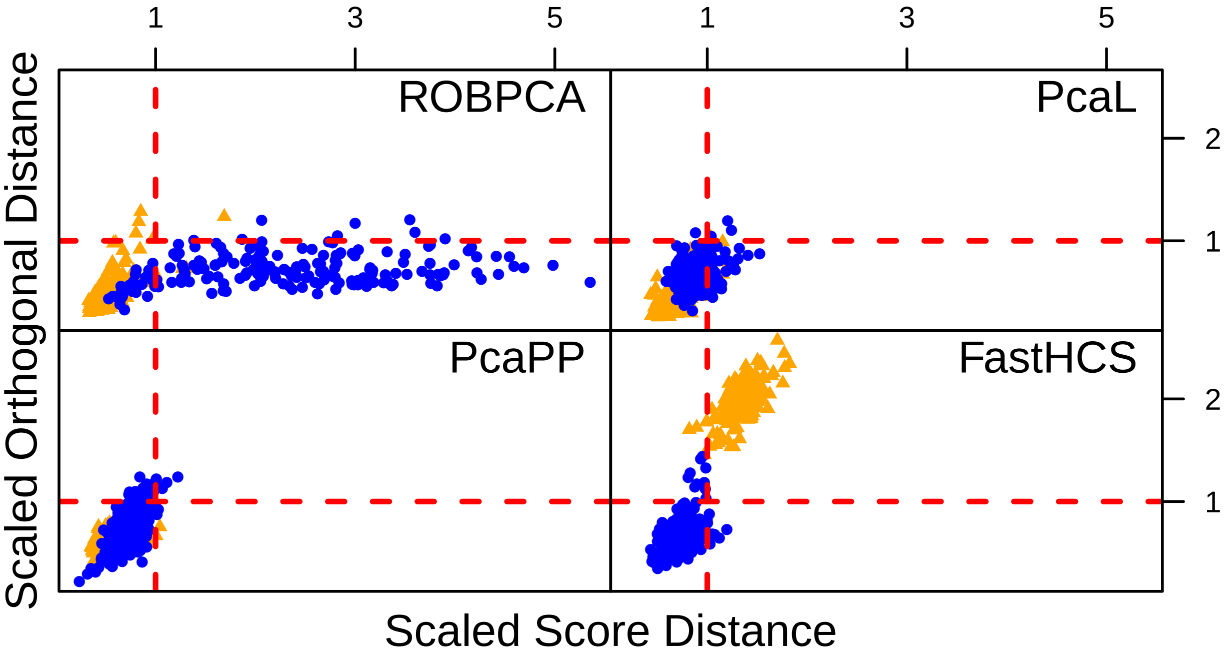

Fig 9. *Diagnostic plots of the scaled score and orthogonal distances of the Fourier coefficients of the numerals. The observations corresponding to numerals with labels "1" ("0") are shown as dark blue circles (light orange triangles).*

deviating from the multivariate pattern of the bulk of the data.

4.3. *DNA Alteration Dataset.* In our final case study, we examine another high-dimensional data set; the DNA Alteration Dataset (Christensen et al., 2009). This data set consists of cytosine methylation β values collected at 1413 autosomal CpG loci (the variables) in a sample of 217 non-pathological human tissue specimens (the observations) taken from 10 different anatosites. In (Christensen et al., 2009), the authors show that the tissue samples in this data set form three well separated subgroups. The first of these constitutes all 113 observations corresponding to cytosine methylation β values measured on "non-blood, non placenta" (henceforth, simply "non-blood") tissues. A second subgroup of data points comprises the 85 cytosine methylation β measurements taken on blood tissues.

In this application, we will combine the 113 vectors of cytosine methylation β values corresponding to the samples "non-blood" tissue with 85 measurements taken blood tissues (so that $n = 198$ and $p = 1413$). As before, we do not include the vector of class label

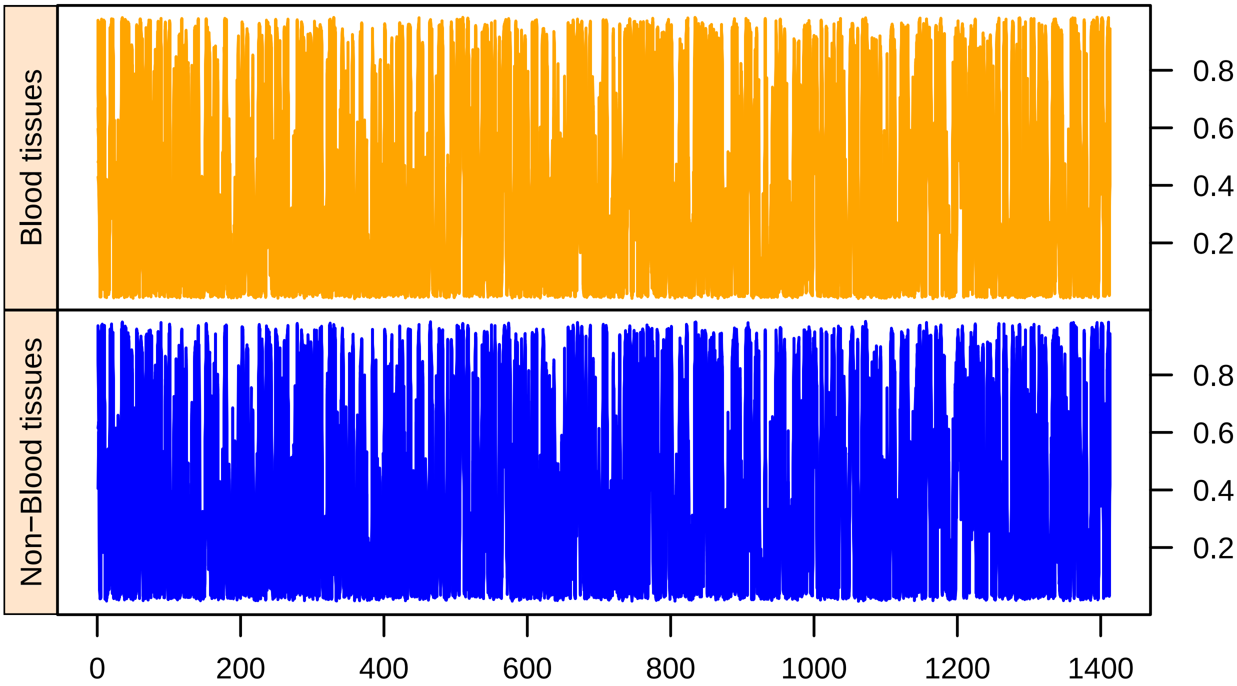

Fig 10. *The 198 vectors of cytosine methylation β values. The first 85 curves (corresponding to observations taken from blood tissues) are show in the top panel in light orange. The main group (113 curves) corresponding to observations taken from non-blood, tissues are shown in the bottom panel in dark blue.*

(indicating, for each observation, to which of the two types of tissue it belongs) in our data matrix so that, in effect, we cast the observations with label "blood" as the outliers and the task of all four algorithms is, again, to reveal them. Because the original data are measurements taken on the same scale, we can plot the 1413 β values corresponding to each observation as line plots: in the top and bottom panel of Figure 10 we do this separately for the members of the outliers (corresponding to the blood tissues, shown in light orange) and majority group (corresponding to the non-blood tissues and shown in dark blue) in our data. Visually, the curves of these two groups appear difficult to distinguish from one another. In particular, the vertical range of both overlap and neither set of curves exhibit any discernible pattern of heteroskedasticity. Figure 11 depicts the resulting four (scaled) PCA outlier maps, in which we assign each observation a color (dark blue or light orange) and a plot symbol (round or triangle) depending on whether the corresponding curve describes a member of class "non-blood" (the majority group) or "blood" (the outlier group), respectively.

We begin by describing the outlier map associated with the PcaL fit. Here the algorithm flags only 21 observations (or $\approx 11\%$ of the sample) as outliers. Of these, 6 actually belong to the actual outlying group (those observations corresponding to blood tissues). Considering now those observations not flagged as outliers by PcaL, we see that they are nearly evenly split between the two sub-components of our data set. For example, over 45% of these data points turn out to belong to the measurements taken from "blood" tissues. This indicates that the PCA model fitted by PcaL is fitted to data points belonging to the two disjoint clusters and consequently that the diagnostic plot derived from it does not reveal the true outliers. Similarly, the outlier map corresponding to the PcaPP fit finds only 13 outliers (or $\approx 7\%$ of the sample). Members of the two subgroups are also nearly evenly represented among those observations with distances to the fitted PcaPP hyperplane less than the O.D. threshold. Turning to ROBPCA, as with the Multiple Feature data set, there is evidence that the fit it returns has been pulled towards the outliers. Here too, the set of observations classified as good by ROBPCA is nearly evenly composed of representatives of both types of tissues. Furthermore, a large group of 82 data points (over 40% of the original sample) corresponding to the true majority group (in this case the samples of type "non blood") are flagged as good leverage points.

For all these algorithms, the lack of correspondence between the patterns revealed by the outlyingness maps and the actual (known but hidden) grouping of the observations suggests that the procedures try to accommodate observations belonging to the two distinct types of tissues as if these were draws from a single, common, distribution. Consequently, the parameters estimated by either PcaPP, PcaL or ROBPCA do not to faithfully describe (and therefore cannot be used to reliably discriminate between) the multivariate patterns of the subgroups. Compare this to the outlier map corresponding to the solution found by FastHCS. Here again, the outlier map derived from the FastHCS fit clearly exposes the true outliers as observations that are inconsistent with the multivariate pattern of the bulk of the data. To illustrate, the actual outliers (the samples from "blood" tissues) are all flagged as outliers by FastHCS. Similarly, the 108 observations not flagged as outliers by FastHCS all belong to the actual majority group. As before, the clear separation of the members of the actual minority

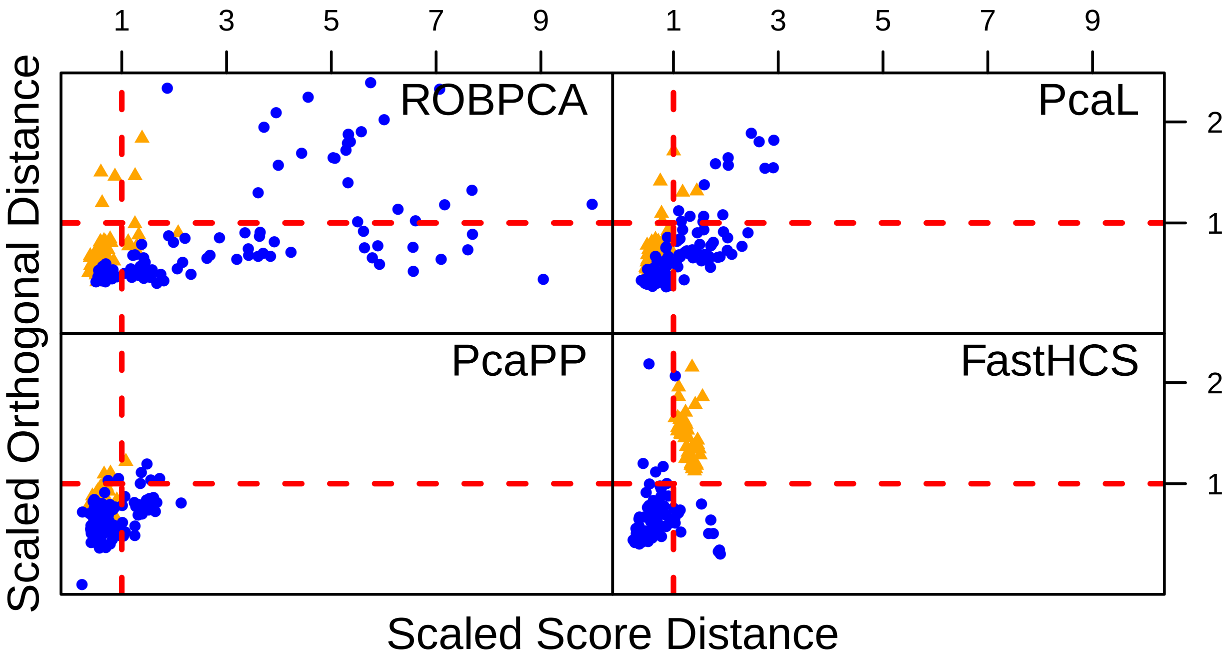

Fig 11. *Diagnostic plots of the scaled score and orthogonal distances of cytosine methylation β values. The observations with labels "non-blood" ("blood") are shown as blue circles (orange triangles).*

subgroup away from it indicates that the fit found by FastHCS is not trying to accommodate the outliers and remains, therefore, close to the one we would have found without them.

5. Outlook. In this article we introduced HCS, a new outlyingness index for high-dimensional data, and FastHCS, a fast and rotation equivariant algorithm for computing it. Like many other outlier detection algorithms, the performance of FastHCS hinges crucially on correctly identifying an h-subset of uncontaminated observations. Our main contribution is to characterize this h-subset using a new measure of homogeneity of a high dimensional cloud of points based on many random projections of the data on lower-dimensional subspaces. This new characterization differs from those used by competing outlier detection procedures in that it was designed to be insensitive to the configuration of the outliers.

Through simulations, we focused on configurations of outliers that are worst-case for rotation equivariant algorithms, and found that FastHCS behaves notably better than the other procedures we considered, often revealing outliers that would not have been identified by these alternative approaches. In most applications, admittedly, contamination patterns will not always be as difficult as those we featured in our simulations and in many cases the different methods will, hopefully, concur. Nevertheless, using three real data examples from fields where PCA is widely used, we were able to establish that it is possible for real world situations to be sufficiently challenging as to push current state of the art outlier detection procedures to their limits and beyond, justifying the development of better solutions. In any case, given that in practice we do not know the configuration of the outliers, as data analysts, we prefer to carry our inferences while planing for the worst contingencies.

References.

Christensen, B.C Houseman, E.A. Marsit, C.J. Zheng, S. Wrench, M.R. Wiemels, J.L. Nelson, H.H. Karagas, M.R. Padbury, J.F. Bueno, R. Sugarbaker, D.J Yeh, R., Wiencke, J.K. Kelsey, K.T. (2009). Aging and Environemental Exposure Alter Tissue-Specific DNA Methylation Dependent upon CpG Island Context. PLoS Genetics 5(8), e1000602.

Croux, C. and Ruiz-Gazen, A. (2005). High breakdown estimators for principal components: the projection-pursuit approach revisited. Journal of Multivariate Analysis, 95, 206–226.

Deepayan, S. (2008). Lattice: Multivariate Data Visualization with R. Springer, New York.

Dyrby, M. Engelsen, S.B. Nørgaard, L. Bruhn, M. and Lundsberg Nielsen, L. (2002). Chemometric Quantitation of the Active Substance in a Pharmaceutical Tablet Using Near Infrared (NIR) Transmittance and NIR FT Raman Spectra Applied Spectroscopy 56(5): 579–585 .

Hubert, M. Rousseeuw, P. J. and Vanden Branden, K. (2005). ROBPCA: a new approach to robust principal components analysis. Technometrics, 47, 64–79.

Locantore, N., Marron, J. S., Simpson, D. G., Tripoli, N., Zhang, J. T., and Cohen, K. L. (1999). Robust principal component analysis for functional data. Test. 8(1), 1–73.

Lopuhaä, H.P. and Rousseeuw, P. J. (1991). Breakdown points of affine equivariant estimators of multivariate location and covariance matrices. Ann. Statist. 19, no. 1, 229–248.

Maronna, R. (2005). Principal Components and Orthogonal Regression Based on Robust Scales. Technometrics, 47, 264–273.

Maronna, R. A.; Martin, R. D. and Yohai, V. J. (2006). Robust Statistics: Theory and Methods. Wiley, New York.

R Core Team (2012). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria.

Rocke, D. M. and Woodruff, D. L. (1996). Identification of Outliers in Multivariate Data. Journal of the American Statistical Association, 91, 1047–1061.

Rousseeuw, P. J. and van Zomeren, B.C. (1990). Unmasking Multivariate Outliers and Leverage Points. Journal of the American Statistical Association, 85, pp. 633–639.

Schmitt, E. Öllerer, V. and Vakili, K. (2014). Finite sample breakdown point of PCS. *Submitted.* arXiv 1402.2986.

Todorov V. and Filzmoser P. (2009). An Object-Oriented Framework for Robust Multivariate Analysis. Journal of Statistical Software, 32, 1–47.

Vakili, K. and Schmitt, E. (2014). Finding multivariate outliers with FastPCS. Computational Statistics & Data Analysis, 69, 54–66.

Van Breukelen, M. Duin, R.P.W. Tax, D.M.J. and Den Hartog, J.E. (1998). Handwritten digit recognition by combined classifiers. Kybernetika, 34, 381–386.

Yohai, V.J. and Maronna, R.A. (1990). The Maximum Bias of Robust Covariances. Communications in Statistics–Theory and Methods, 19, 2925–2933.

E-mail: kaveh.vakili@wis.kuleuven.be
eric.schmitt@wis.kuleuven.be