Journal of Data Science, Statistics, and Visualisation

December 2025, Volume V, Issue X. *doi: 10.52933/jdssv.v5i10.165*

Graphical Tools for Visualizing Cellwise and Casewise Outliers

Mehdi Hirari
KU Leuven

Mia Hubert
KU Leuven

Peter J. Rousseeuw
KU Leuven

Abstract

Principal component analysis (PCA) and other dimension reduction methods can be affected by cellwise and casewise outliers. Several approaches have been proposed that downweight outlying cells or cases to ensure a more reliable fitting process. The outputs of these robust methods can be used to detect anomalies by means of graphical displays. Our focus is on new visualizations of deviations from a PCA fit that is robust to both cellwise and casewise outliers, and that provides imputed values. The novelties include a residual cellmap in which outlying cases are shaded, a visualization of outlying cells in functional data, cellmaps of PCA scores, and two displays of the effect of imputation. The graphics are illustrated on several real datasets, including video data. The visualizations are implemented in a Shiny app.

Keywords: Anomaly detection, dimension reduction, graphics, principal component analysis, Silhouette plot.

1. Introduction

Nowadays high-dimensional datasets occur increasingly often. Principal component analysis (PCA) is a popular tool in their analysis. But the presence of outliers poses a substantial challenge, as classical methods are highly sensitive to anomalies and can produce biased results. To remedy this, so-called robust methods have been developed that yield fits that are less affected by anomalies. In the past, research in robust statistics has mainly focused on casewise outliers. These are cases (instances) that deviate from the pattern formed by the majority of the cases. In a typical data matrix of cases by variables (features), cases are rows of that matrix. However, robust statistics has recently seen a paradigm shift towards cellwise outliers (Alqallaf et al. 2009), which are individual entries in the data matrix. High-dimensional datasets may contain a few outlying cells in most of the rows, and then we do not want to remove those rows entirely, since their valid cells still contain much useful information. The situation is somewhat similar to data with missing values, except that we know from the start which cells are missing, whereas we don't know beforehand which cells contain outlying values.

Several dimension reduction methods have been developed that are robust against casewise outliers (Locantore et al. 1999; Croux and Ruiz-Gazen 2005; Hubert et al. 2005). De La Torre and Black (2003) and Maronna and Yohai (2008) introduced robust methods for data containing cellwise outliers. Recently the cellPCA method was constructed (Centofanti et al. 2024), building on earlier work in (Hubert et al. 2019). CellPCA can handle missing values as well as cellwise and casewise outliers, all of which may occur simultaneously. It obtains a robust fit by assigning weights to individual cells and cases, and provides imputations for missing values and outlying cells.

Here we introduce new graphical displays to visualize and help interpret outliers from the output of cellPCA. The proposed tools are based on the cellwise and casewise weights and residuals that will be described in Section 2 below, together with the cellPCA objective function and the imputed values. The new tools include a residual cellmap in which outlying rows are more easily recognizable by shading, and a visualization of outlying cells when the data consist of curves. There is a display of the effect of imputation on geometric distances, and another shows its effect on PCA scores. Finally, a kind of silhouette plot describes how well each datapoint is accommodated by the PCA fit.

Section 3 constructs the new graphical tools for detecting cellwise and casewise anomalies and illustrates them on real data. Section 4 contains an application to video data. Finally, Section 5 presents a Shiny app that integrates all of these tools in an interactive platform for outlier analysis. Section 6 concludes.

2. Cellwise and Casewise Robust PCA

2.1. CellPCA

The p coordinates of the n cases are stored in an $n \times p$ data matrix $\boldsymbol{X}$. In the absence of outliers and missing values, the goal is to represent the data in a lower dimensional space, that is

$$\boldsymbol{X} = \boldsymbol{X}^0 + \mathbf{1}_n \boldsymbol{\mu}^T + \boldsymbol{E}$$

where $\boldsymbol{X}^0$ is an $n \times p$ matrix of rank $k < p$, $\mathbf{1}_n$ is a column vector with all n components equal to 1, the center $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_p)^T$ is a column vector of size p, and $\boldsymbol{E}$ is the error term.

The cellPCA method proposed by Centofanti et al. (2024) copes with cellwise and casewise outliers and missing values. It approximates $\boldsymbol{X}$ by $\widehat{\boldsymbol{X}} := \widehat{\boldsymbol{X}^0} + \mathbf{1}_n \widehat{\boldsymbol{\mu}}^T$ obtained by minimizing the objective function

$$L_{\rho_1,\rho_2}(\boldsymbol{X}, \boldsymbol{X}^0, \boldsymbol{\mu}) := \frac{\widehat{\sigma}_2^2}{m} \sum_{i=1}^{n} m_i \, \rho_2 \left(\frac{1}{\widehat{\sigma}_2} \sqrt{\frac{1}{m_i} \sum_{j=1}^{p} m_{ij} \, \widehat{\sigma}_{1,j}^2 \, \rho_1 \left(\frac{x_{ij} - \widehat{x}_{ij}}{\widehat{\sigma}_{1,j}} \right)} \right) \tag{1}$$

with respect to $(\boldsymbol{X}^0, \boldsymbol{\mu})$, under the constraint that $\boldsymbol{X}^0$ has rank k. Here m_{ij} is 0 if x_{ij} is missing and 1 otherwise, $m_i = \sum_{j=1}^{p} m_{ij}$, and $m = \sum_{i=1}^{n} m_i$. The estimated scales $\widehat{\sigma}_{1,j}$ standardize the *cellwise residuals*

$$r_{ij} := x_{ij} - \widehat{x}_{ij}$$

of variable j. The scale $\widehat{\sigma}_2$ standardizes the *rowwise total deviation* defined as

$$t_i := \sqrt{\frac{1}{m_i} \sum_{j=1}^{p} m_{ij} \, \widehat{\sigma}_{1,j}^2 \, \rho_1 \left(\frac{r_{ij}}{\widehat{\sigma}_{1,j}} \right)} \, . \tag{2}$$

In order to compute the $\widehat{\sigma}_{1,j}$ we start from the initial MacroPCA fit (Hubert et al. 2019) which has its own cellwise residuals $r_{1,j}^*$, to which we apply a univariate M-estimator of scale. That is, we solve the equation

$$\frac{1}{n} \sum_{i=1}^{n} \rho \left(\frac{r_{1,j}^*}{a \, \sigma} \right) = \delta \tag{3}$$

where $\delta = 1.8811$, $a = 0.3431$, and the function ρ is given by

$$\rho_{b,c}(z) = \begin{cases} z^2/2 & \text{if } 0 \leqslant |z| \leqslant b \\ d - (q_1/q_2) \ln(\cosh(q_2(c - |z|))) & \text{if } b \leqslant |z| \leqslant c \\ d & \text{if } c \leqslant |z| \end{cases} \tag{4}$$

where $d = (b^2/2) + (q1/q2) \ln(\cosh(q_2(c - b)))$. The function $\rho_{b,c}$ is smooth, which implies certain constraints on q_1 and q_2. By default we use $b = 1.5$ and $c = 4$ with $q_1 = 1.54$ and $q_2 = 0.86$. From the initial cellwise residuals and the $\widehat{\sigma}_{1,j}$ we can compute the initial rowwise total deviations, and applying the same univariate M-estimator to them yields $\widehat{\sigma}_2$. The loss functions ρ_1 and ρ_2 in the objective (1) are also of the form (4). The iterative algorithm for minimizing (1) uses the $n \times p$ weight matrix

$$\boldsymbol{W} = \{w_{ij}\} = \boldsymbol{W}^{\text{cell}} \odot \boldsymbol{W}^{\text{case}} \odot \boldsymbol{M}$$

where the Hadamard product $\odot$ multiplies matrices entry by entry. The $n \times p$ matrix $\boldsymbol{W}^{\text{cell}} = \{w_{ij}^{\text{cell}}\}$ contains the *cellwise weights*

$$w_{ij}^{\text{cell}} = \psi_1 \left(\frac{r_{ij}}{\widehat{\sigma}_{1,j}} \right) \Big/ \frac{r_{ij}}{\widehat{\sigma}_{1,j}} \qquad \text{for} \qquad i = 1, \ldots, n, \quad j = 1, \ldots, p$$

where $\psi_1 = \rho_1'$ and with the convention $0/0 = 1$. The $n \times p$ matrix $\boldsymbol{W}^{\text{case}}$ has constant rows, where each entry of row i is the *casewise weight* w_i^{case} given by

$$w_i^{\text{case}} = \psi_2\left(\frac{t_i}{\widehat{\sigma}_2}\right) \Big/ \frac{t_i}{\widehat{\sigma}_2} \quad \text{for} \quad i = 1, \ldots, n$$

with $\psi_2 = \rho_2'$. The $n \times p$ matrix $\boldsymbol{M}$ contains the missingness indicators m_{ij}.

When one or more cells of $\boldsymbol{x}_i$ have weights below 1, it is useful to obtain an imputed version $\boldsymbol{x}_i^{\text{imp}}$ whose cells are $x_{ij}^{\text{imp}} = x_{ij}$ for all j with $w_{ij} = 1$, and with different cells x_{ij}^{imp} where $w_{ij} < 1$. The modified cells are defined such that $\boldsymbol{x}_i^{\text{imp}}$ is shrunk toward the PCA subspace, and that the orthogonal projection of $\boldsymbol{x}_i^{\text{imp}}$ coincides with the fitted $\widehat{\boldsymbol{x}}_i$. The imputed point $\boldsymbol{x}_i^{\text{imp}}$ has the coordinates

$$x_{ij}^{\text{imp}} := \widehat{x}_{ij} + w_{ij}^{\text{cell}} m_{ij} \left(x_{ij} - \widehat{x}_{ij}\right) \tag{5}$$

and its orthogonal projection on the PCA subspace indeed equals $\widehat{\boldsymbol{x}}_i$. Note that every imputed cell x_{ij}^{imp} lies between the original cell x_{ij} and the cell $\widehat{x}_{ij}$ of the fitted point.

3. Graphical Tools

3.1. Shaded residual cellmap

After cellPCA has run we have the fit $\widehat{\boldsymbol{X}} = \widehat{\boldsymbol{X}^0} + \boldsymbol{1}_n \widehat{\boldsymbol{\mu}}^T$. This yields the residual matrix $\boldsymbol{R} := \boldsymbol{X} - \widehat{\boldsymbol{X}}$ which is also of size $n \times p$, so each cell x_{ij} of $\boldsymbol{X}$ has its own residual $r_{ij} = x_{ij} - \widehat{x}_{ij}$. We estimate the scale of each column of $\boldsymbol{R}$ by a robust M-estimator of scale. Dividing each column of $\boldsymbol{R}$ by its scale $\widetilde{\sigma}_{1,j}$ yields the standardized residuals, that are combined in the matrix $\widetilde{\boldsymbol{R}} = \{\widetilde{r}_{ij}\}$. We can then visualize $\widetilde{\boldsymbol{R}}$, or some of its rows and columns, by a cellmap (Rousseeuw and Van den Bossche 2018). It can be drawn by the function `cellMap` in the R package **cellWise** (Raymaekers and Rousseeuw 2023). Note that for clean Gaussian data the squared residuals $\widetilde{r}_{ij}^2$ are roughly χ_1^2 distributed. Therefore, cells with $|\widetilde{r}_{ij}| < \sqrt{\chi^2_{1,0.99}} \approx 2.57$ are considered regular and colored yellow, whereas any missing values are white. Positive residuals bigger than 2.57 receive a color that ranges from light orange to dark red, and negative residuals below -2.57 are shown in light purple to dark blue. The function `cellMap` has an argument `darkestColor`, that determines how big the absolute value of the residual needs to be to receive the darkest tint of red or blue. By default it is set to $\sqrt{\chi^2_{1,0.999}} \approx 3.29$.

The octane data of (Esbensen et al. 1996) contains near infrared (NIR) absorbance spectra of $n = 39$ gasoline samples over $p = 226$ wavelengths ranging from 1,102 nm to 1,552 nm, with measurements every two nanometers. It is available as **octane** in the R package **rrcov** (Todorov 2025) on CRAN, and shown in Figure 1.

We apply cellPCA with $k = 2$, yielding the *residual cellmap* in Figure 2. It contains many cell residuals in the darkest red and blue. In order to highlight the most extreme outliers, we can increase the argument `darkestColor`. Setting it equal to 20 yields Figure 3, in which only cells with $|\widetilde{r}_{ij}| \geqslant 20$ get the darkest red or blue.

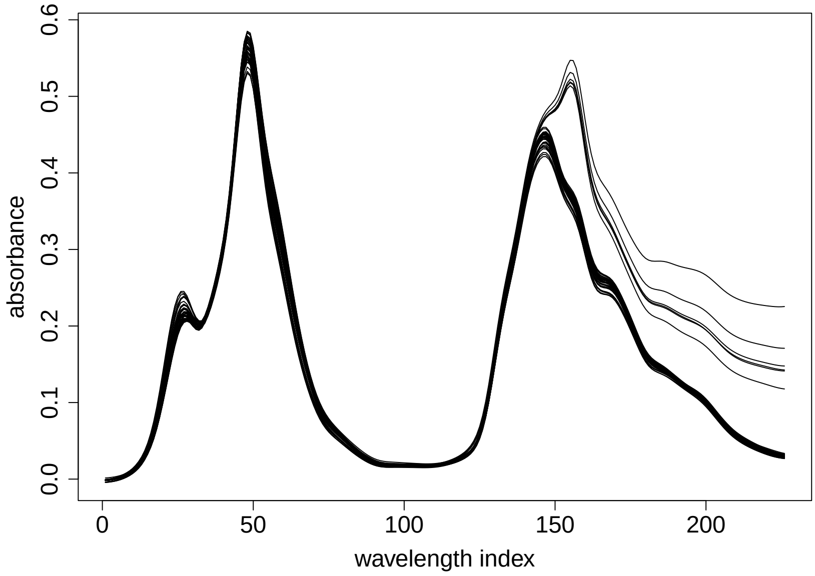


Figure 1: Raw absorbance spectra of the octane data.

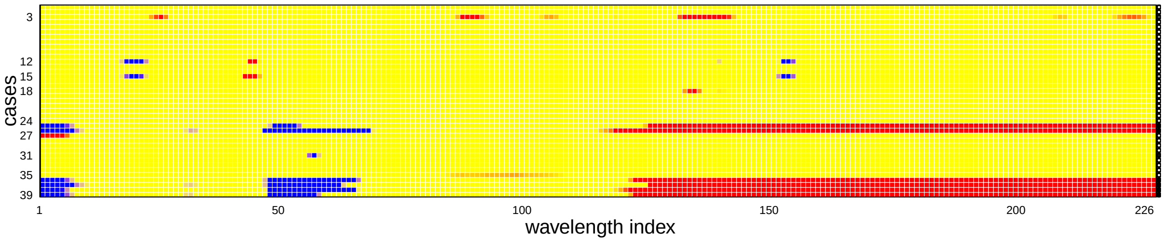


Figure 2: Residual cellmap of the octane data with default settings.

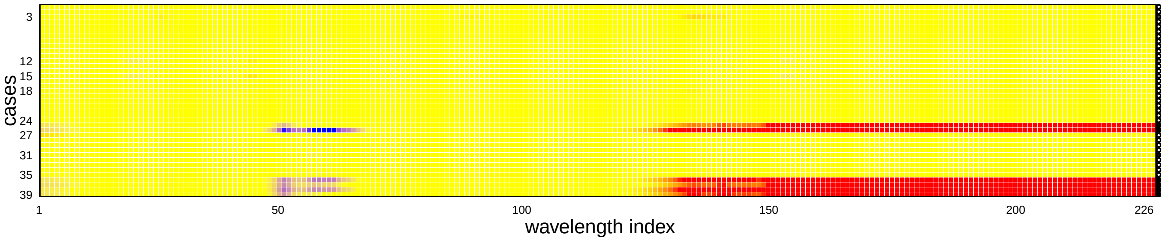


Figure 3: Residual cellmap of the octane data, with higher argument `darkestColor`.

The cases with these extremely outlying cells are the six gasoline samples 25, 26, and 36–39. These were already detected by the casewise robust PCA method in Hubert et al. (2005). Their casewise outlyingness is due to the fact that they contain added ethanol. Since these casewise outliers are essentially members of a different population, we can ask ourselves whether their cellwise residuals are relevant. In fact, cellPCA has assigned a casewise weight $w_i^{\text{case}} = 0$ to them. Figures 2 and 3 drawn by `cellWise::cellMap` do visualize casewise weights in the little circles to the right of each case, whose color ranges from white for small t_i to black for large t_i. However, these are barely noticeable due to their small size, caused by the number of rows in the cellmap.

To address this issue we now propose a *shaded residual cellmap* that makes casewise outliers more visible. It represents a measure of casewise outlyingness by a shade of grey overlaid on the rows of the residual cellmap. For this we recompute the casewise deviation t_i as in (2) but now based on $\widetilde{r}_{ij}$ and $\widetilde{\sigma}_{1,j}$. Then, we compute their M-scale $\widetilde{\sigma}_2$ and finally standardize them, yielding the standardized casewise deviation

$$\tilde{t}_i := \frac{t_i}{\widetilde{\sigma}_2}.$$

The shaded residual cellmap of the octane data is in Figure 4.

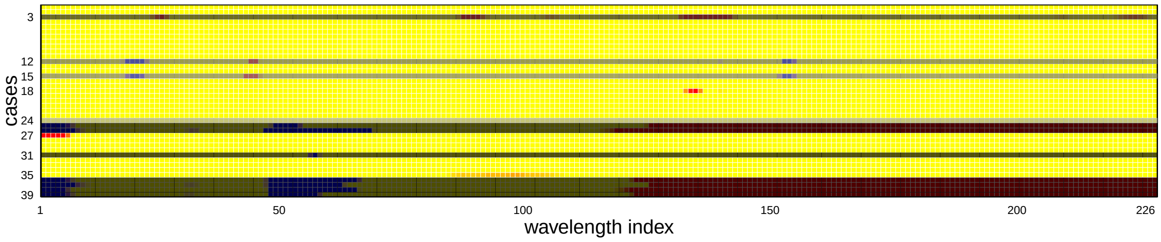


Figure 4: Shaded residual cellmap of the octane data.

The shaded residual cellmap in Figure 4 leaves the rows with low $\tilde{t}_i$ as before. But outlying cases appear in darker shades, deliberately hiding or at least putting less emphasis on cellwise residuals of such cases. To gauge the outlyingness, the code simulates $\tilde{t}_i$ on outlier-free data to estimate the 99th percentile cutoff q_t. Cases with $\tilde{t}_i < q_t$ are considered regular so they remain unchanged, while those with $\tilde{t}_i \geqslant q_t$ become darker. The darkest grey is assigned to cases with $\tilde{t}_i \geqslant 1.5q_t$. The cellwise residuals of the most outlying rows 25–26 and 36–39 are now hard to see.

Depending on the type of data and the purpose of the analysis, one may wish to keep seeing the colors of the cells even in highly outlying cases. For this purpose the code has the input argument `opacity`. When set to 0 the original cellmap of Figure 2 is drawn, and for `opacity = 1` the cases with $\tilde{t}_i \geqslant 1.5q_t$ would become completely black. The argument `opacity` can be set to any value between 0 and 1 to regulate the shading of cases with $\tilde{t}_i \geqslant q_t$. Figure 4 has `opacity = 0.7`. The user can thus choose how much information to present about the cell colors in outlying rows. For instance, Figure 5 shows the same cellmap with `opacity = 0.3` so all cell colors remain visible.

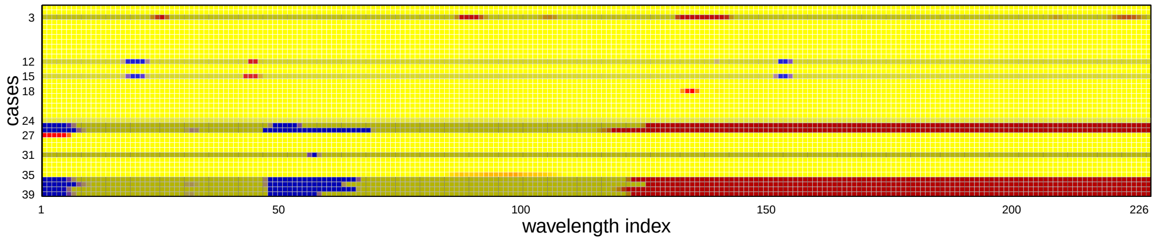


Figure 5: Shaded residual cellmap with `opacity = 0.3` for lighter shading.

The shaded cellmap also highlights cases without extreme cellwise residuals that may not initially appear suspicious but are still worth investigating. For example, cases 3 and 31 show an unusually high casewise deviation. On the other hand cases 18, 27 and 35 have several cellwise outliers but are not flagged. We will come back to this later.

The shaded residual cellmap is also illustrated on a larger dataset, shown in Figure 6. The glass data of Lemberge et al. (2000) is available as `data_glass` in the R package **cellWise**. It consists of EPXMA spectra of $n = 180$ archaeological glass pieces, with $p = 750$ variables that correspond to energy values. The first 13 variables are removed as they are almost constant. From an earlier analysis (Hubert et al. 2005) it is known that cases 143–180 are outlying due to the cleaning of the detector window before the last 38 spectra were measured. Cases 57–63 and 74–76 were also flagged as outliers and had a high concentration of calcic. Cases 19–33 contained higher concentrations of phosphorus.

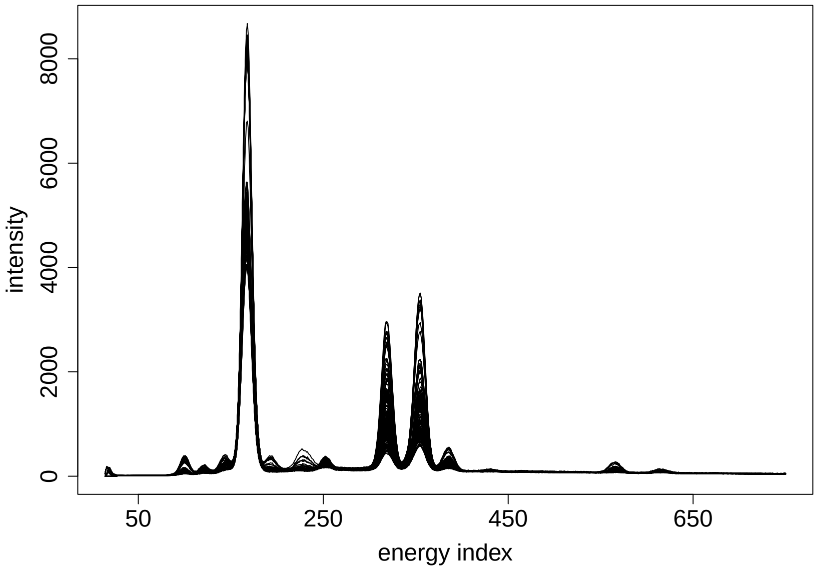


Figure 6: Raw spectra of the glass data.

We now apply cellPCA with $k = 3$. Figure 7 displays the original residual cellmap and the shaded version with `opacity = 0.5`. The known groups of outlying cases are clearly visible as dark bars, although the greyscale varies across the groups. Within the group 19–33 there are however three cases (22, 23, and 33) with regular standardized total deviation. Cases 6, 8, 10 and 15 are noteworthy because they are flagged as casewise outlying without a lot of cellwise outliers. We also note that the cell pattern of case 180 looks quite different from those of cases 143–179.

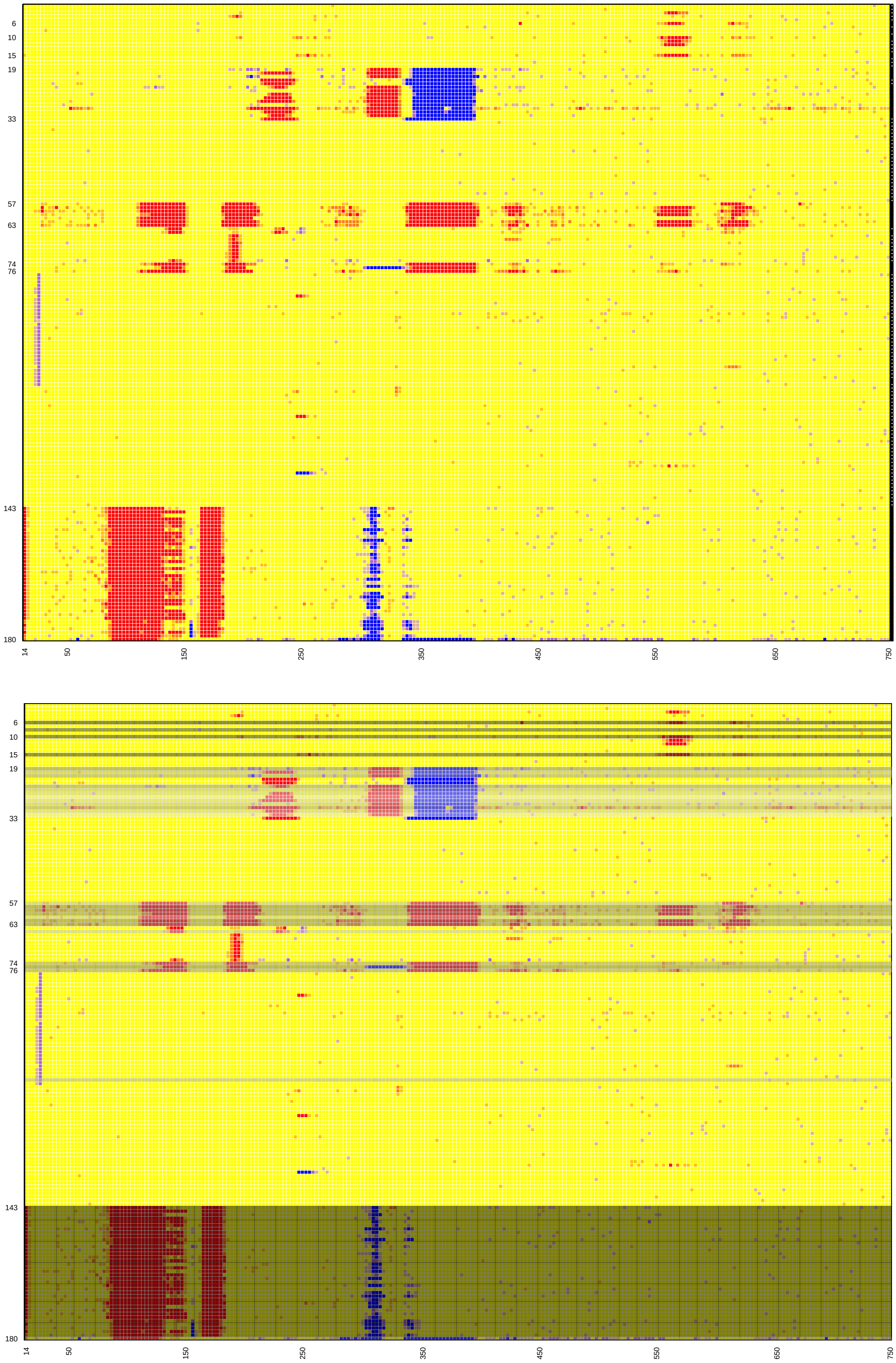


Figure 7: Glass data: (top) residual cellmap, and (bottom) its shaded version that also displays casewise outlyingness.

Figure 8 shows an index plot of the $\tilde{t}_i$ values of the glass data, together with a horizontal line at the cutoff q_t. The group 143–179 is most outlying, while case 180 exhibits a much lower deviation. The groups 19–33, 57–63 and 74–76 show a more moderate degree of outlyingness, with some cases lying just below the cutoff line.

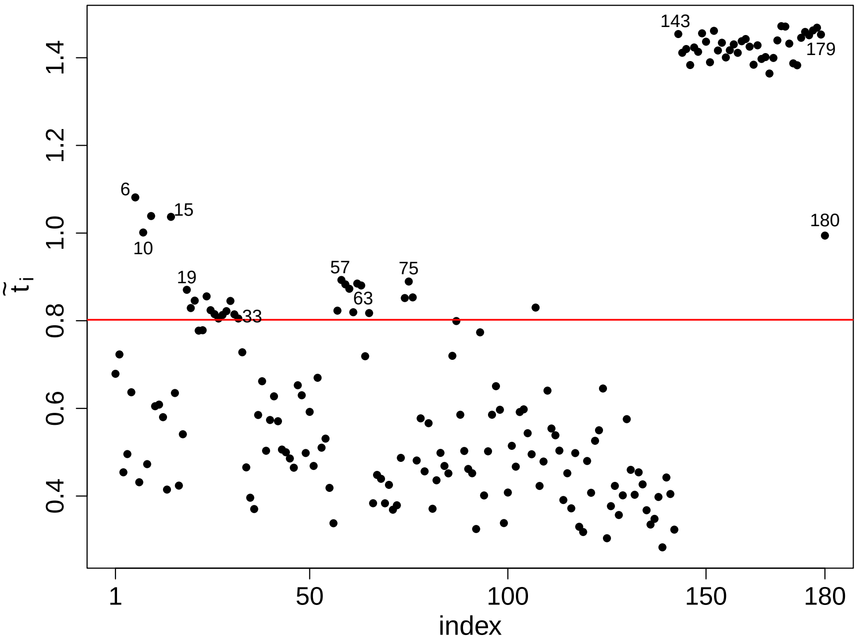
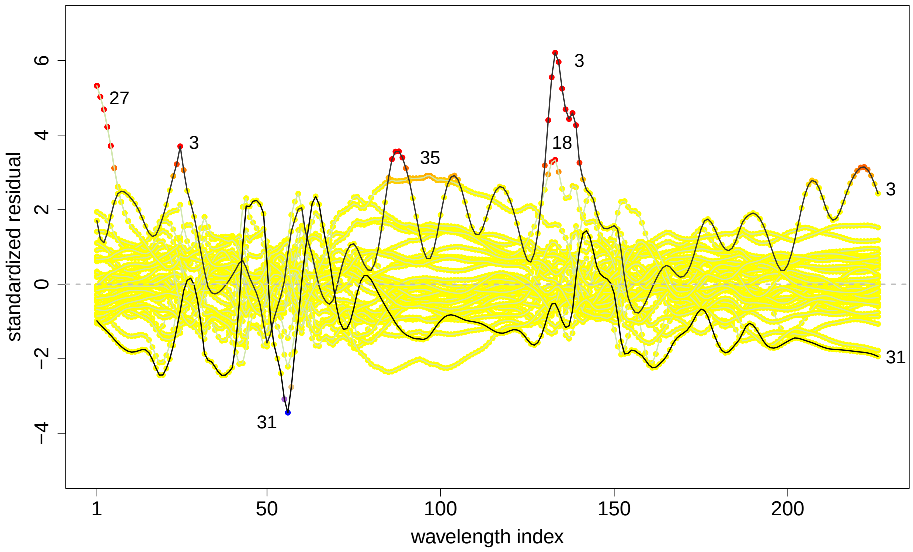


Figure 8: Standardized total deviation of the glass data. Cases above the cutoff line are colored light grey to dark grey in the shaded cell map in the lower panel of Figure 7.

3.2. Functional cellmaps

Datasets that consist of curves can easily be visualized. This allows a better understanding of why certain cases (curves) or cells are outlying, as their outlyingness can be included in the display. Inspired by the residual cellmap, we can plot each curve of the standardized residuals $\widetilde{\boldsymbol{R}}$ with its cells represented as bullets. These bullets share the exact same color as in the residual cellmap. Additionally, each curve is colored based on its $\tilde{t}_i$ value: curves with a regular $\tilde{t}_i$ value are shown in green while curves with high $\tilde{t}_i$ are shaded from light grey to dark grey to visualize their global outlyingness. The color coding is similar to that in the shaded cellmap.

Figure 9 shows *functional residual cellmaps* of the octane and glass data. For illustration purposes they contain all the regular curves, but not all the outlying ones. Note that the regular green curves have many yellow points on them, making them seem yellowish green. In the plot of the octane residuals we have labeled curves 3 and 31, which have an outlying total deviation. It is now clear why they were flagged as casewise outliers. Curve 3 is characterized by mainly positive residuals, in contrast to case 31, which shows mostly negative residuals. We also spot the outlying parts of the unflagged cases 18, 27 and 35. The functional residual map of the glass data (restricted to the energy indices 200–400) clarifies the difference between curves 30 and 33. They share two regions with cellwise outliers, but only curve 30 is outlying at energy indices 304–333.



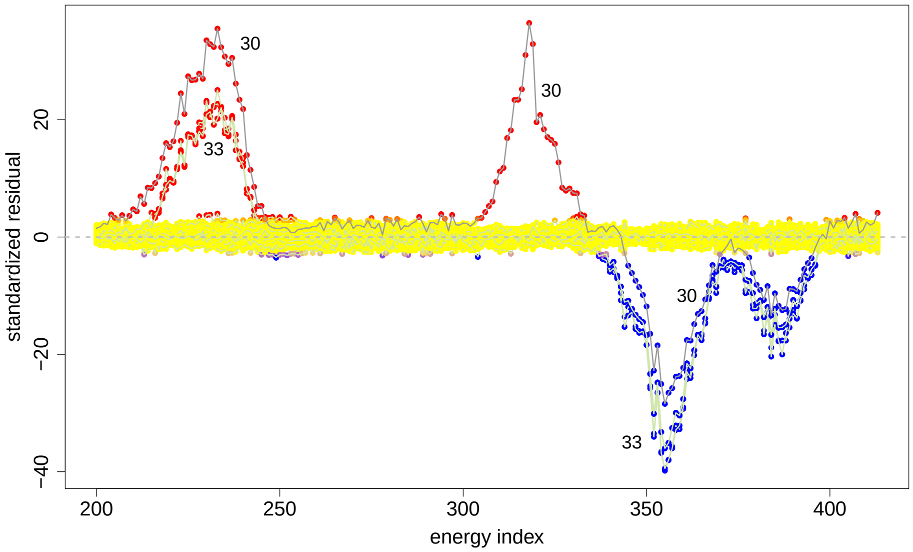
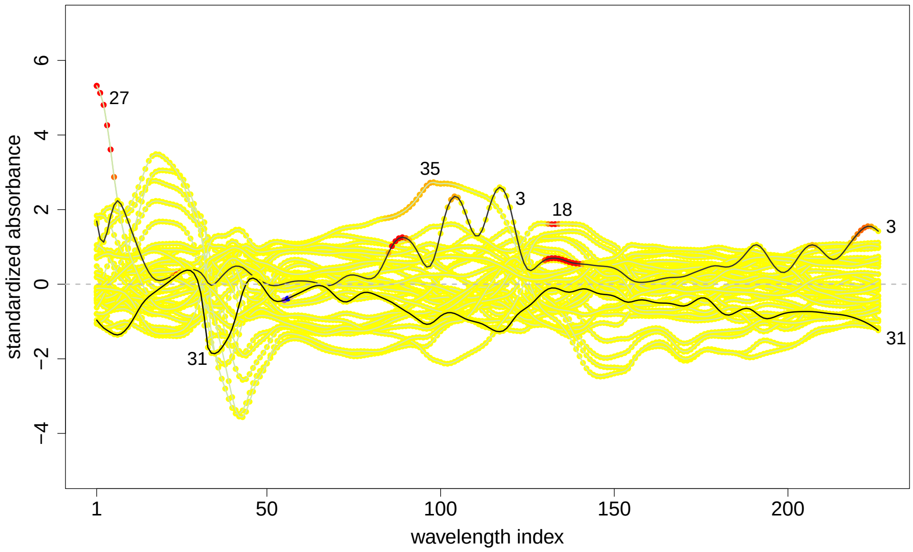


Figure 9: Functional residual cellmaps of the octane data (top) and glass data (bottom).

In addition to plotting the curves of the standardized residuals $\widetilde{\boldsymbol{R}}$ it may also be of interest to plot the data themselves. For this purpose we first standardize the data by subtracting the center $\widehat{\boldsymbol{\mu}}$ and then dividing at each wavelength by an M-scale like (3). This removes the heteroskedasticity. The resulting *functional cellmaps* are displayed in Figure 10. The colors of the points and the curves are the same as in Figure 9. These plots illustrate that cellwise outliers are not always marginally outlying. For instance, the cellwise outliers in curve 3 of the octane data at wavelength indices 130–141 do not stand out here, whereas they did in the residuals in Figure 9.



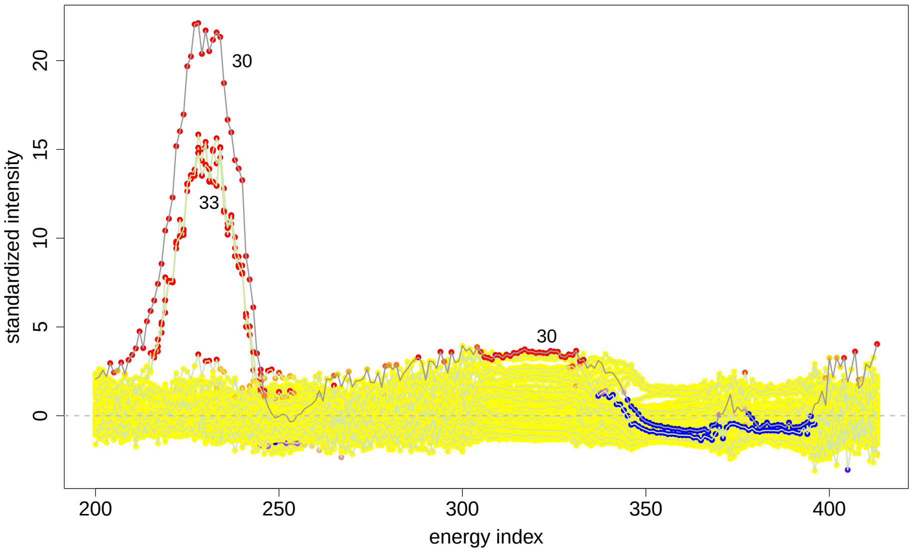


Figure 10: Functional cellmaps of the octane data (top) and glass data (bottom).

3.3. Drop plot

Imputing outlying cells by (5) is an important part of cellPCA, yielding imputed points $\boldsymbol{x}_i^{\text{imp}}$. Visualizing the imputed points can reveal additional information about the data. For this purpose we compute some distances. The Euclidean distance $d_{1,i} = \|\boldsymbol{x}_i - \boldsymbol{x}_i^{\text{imp}}\|$ measures how far the data point $\boldsymbol{x}_i$ has to travel to remedy its outlying cells. When $w_{ij}^{\text{cell}} = 1$ for all j, no imputation is needed so $d_{1,i} = 0$. Next, we compute the distance $d_{2,i} = \|\boldsymbol{x}_i^{\text{imp}} - \widehat{\boldsymbol{x}}_i\|$ that says how far the imputed point is from the k-dimensional principal subspace. In the bad situation where all cells have zero weight, that is $w_{ij}^{\text{cell}} = 0$ for all j, the imputed point will equal the fitted point so $d_{2,i} = 0$. Finally, the distance

$d_{3,i} = \|\boldsymbol{x}_i - \pi(\boldsymbol{x}_i)\|$ is between the observed point $\boldsymbol{x}_i$ and its projection $\pi(\boldsymbol{x}_i)$, where π is the orthogonal projection on the principal subspace.

For distances inside the k-dimensional subspace we compute the so-called *score distances* (Hubert et al. 2005). For any point in the principal subspace we consider its PCA scores, that we combine in a k-variate scores vector denoted as $\boldsymbol{u}$. The center $\widehat{\boldsymbol{\mu}}$ gets $\boldsymbol{u} = \boldsymbol{0}$. The score distance SD of such a point is defined as

$$\mathrm{SD}(\boldsymbol{u}) = \sqrt{\sum_{j=1}^{k} \frac{u_{ij}^2}{\widehat{\lambda}_j}}$$

in which $\widehat{\lambda}_j$ is the j-th eigenvalue of the robust $k \times k$ covariance matrix provided by cellPCA. Its contours are ellipsoids with axes parallel to the principal directions. Figure 11 is a sketch of all these points and distances for a setting with data dimension $p = 3$ and subspace dimension $k = 2$. It shows an observed point $\boldsymbol{x}_i$ and its orthogonal projection $\pi(\boldsymbol{x}_i)$, as well as the imputed point $\boldsymbol{x}_i^{\text{imp}}$ and its projection $\widehat{\boldsymbol{x}}_i$. The score distance of the fitted point $\widehat{\boldsymbol{x}}_i$ is $\mathrm{SD}(\widehat{\boldsymbol{x}}_i)$, and the score distance of the projected point $\pi(\boldsymbol{x}_i)$ is $\mathrm{SD}(\pi(\boldsymbol{x}_i))$.

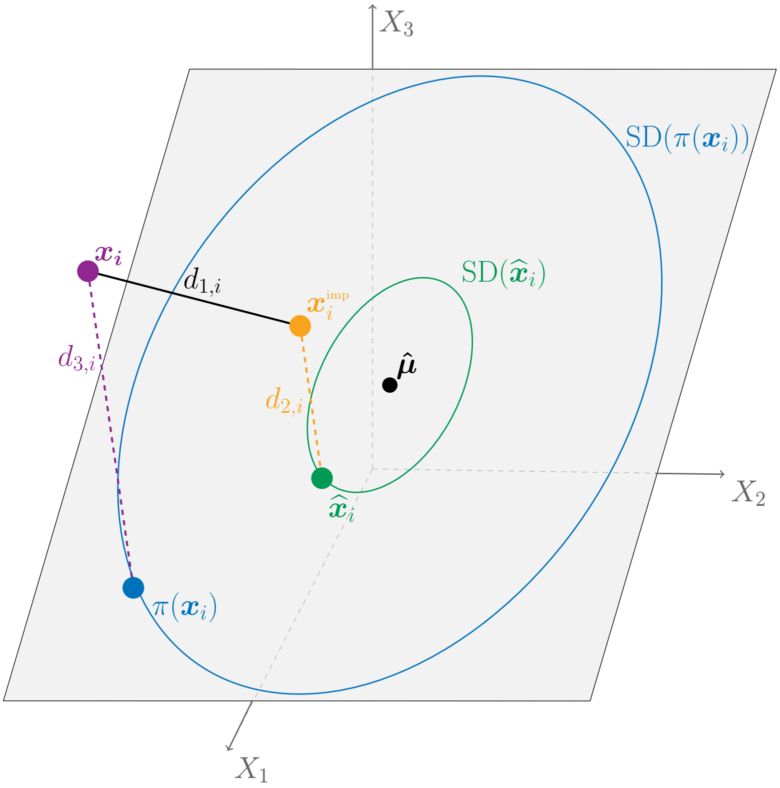

X_3
$\mathrm{SD}(\pi(\boldsymbol{x}_i))$
$\boldsymbol{x}_i$
$d_{1,i}$
$\mathrm{SD}(\widehat{\boldsymbol{x}}_i)$
$\boldsymbol{x}_i^{\text{imp}}$
$d_{3,i}$
$\hat{\boldsymbol{\mu}}$
$d_{2,i}$
X_2
$\widehat{\boldsymbol{x}}_i$
$\pi(\boldsymbol{x}_i)$
X_1

Figure 11: Illustration with an observed point $\boldsymbol{x}_i$ and its projection $\pi(\boldsymbol{x}_i)$, as well as the imputed point $\boldsymbol{x}_i^{\text{imp}}$ and its projection $\widehat{\boldsymbol{x}}_i$.

The *drop plot* of the glass data is in the bottom panel of Figure 12. It summarizes the imputation scheme for all cases. Its horizontal axis holds score distances, and the vertical axis shows distances orthogonal to the subspace. Observed points $\boldsymbol{x}_i$ (purple) and their projections $\pi(\boldsymbol{x}_i)$ (blue) are connected by dashed purple line segments of length $d_{3,i}$, their orthogonal distance to the subspace. The imputed points $\boldsymbol{x}_i^{\text{imp}}$ (orange) are connected to their projections $\widehat{\boldsymbol{x}}_i$ (dark green) by dashed orange line segments of length $d_{2,i}$. The dark grey lines connect the purple observed points $\boldsymbol{x}_i$ to the orange imputed points $\boldsymbol{x}_i^{\text{imp}}$. The red vertical line is at the $\sqrt{\chi^2_{k,0.99}}$ cutoff for the score distances. The drop plot shows the effect of the imputation, which brings us from the original data points (in purple) to the imputed points (in orange).

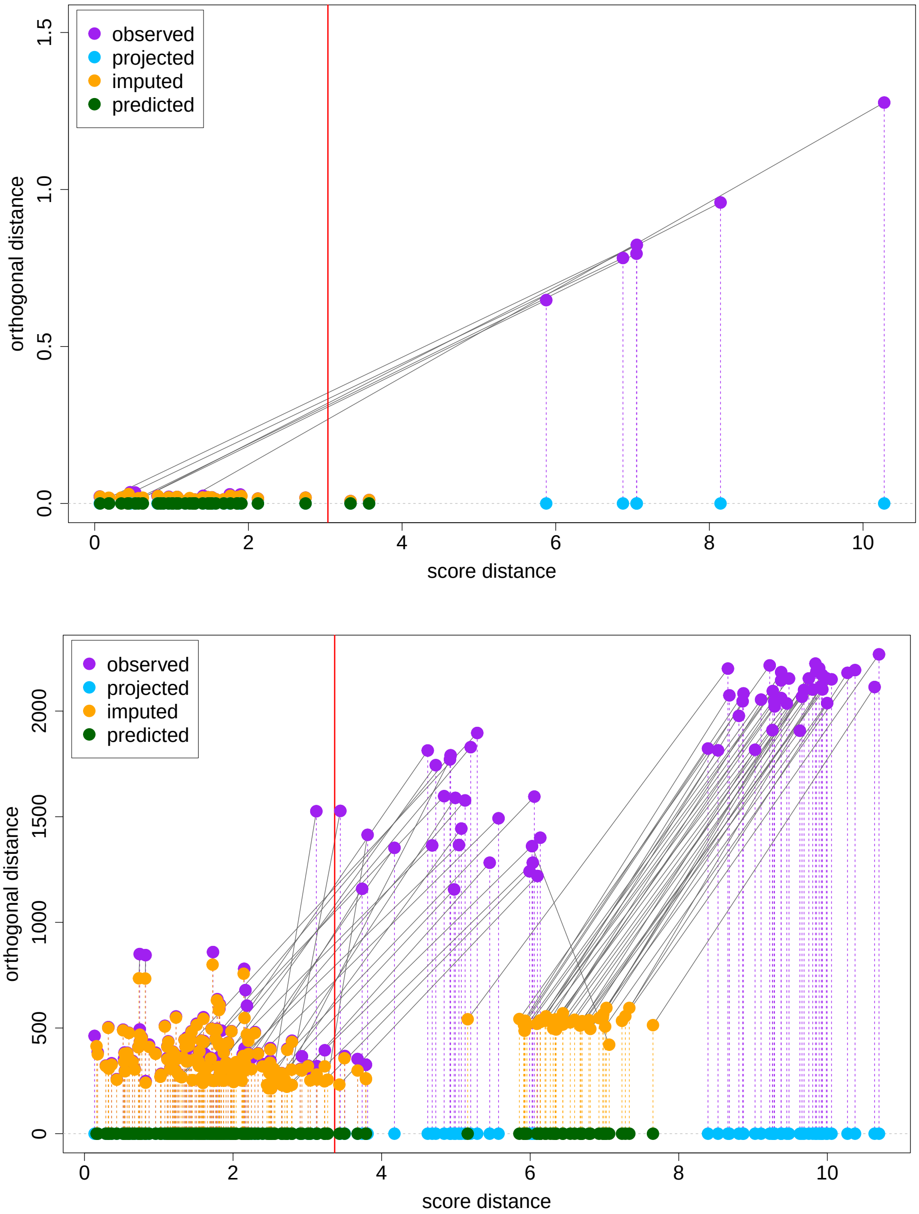


Figure 12: Drop plot of the octane data (top) and the glass data (bottom).

In the top panel of Figure 12 we see the effect of imputation on the octane data. The purple points that were furthest from the principal subspace, that is, with the highest

$d_{3,i}$, were turned into orange points with much smaller distance $d_{2,i}$ from it. That is the nature of imputation. The purple points also had a high SD, whereas the resulting orange points happen to get a much lower SD, to the left of the vertical cutoff line.

The drop plot of the glass data in the bottom panel of Figure 12 has more intricate patterns. We see that most of the dark grey lines point in the same direction: the fitted points tend to have a lower SD than the original points. The exception is case 180 that gets a larger SD after fitting. This can happen because the cellPCA fitting mechanism deliberately uses only the principal subspace, and not the center $\widehat{\boldsymbol{\mu}}$. Note that the cases with the highest SD did not move to the left of the SD cutoff, whereas the less extreme ones did.

3.4. Score cellmap

In addition to the drop plot, we can draw a cellmap of the scores of the projected points $\pi(\boldsymbol{x}_i)$. The cellmap is computed as before, with a cell u_{ij} considered regular and colored yellow when $|u_{ij}/\sqrt{\widehat{\lambda}_j}| < \sqrt{\chi^2_{1,0.99}} \approx 2.57$. Cells with $|u_{ij}/\sqrt{\widehat{\lambda}_j}| \geqslant 2.57$ receive a color that ranges from light orange to dark red. Note that the sign of the scores u_{ij} is not used, because the loading vectors are only determined up to sign. The first cellmap in the left panel of Figure 13 shows the score cellmap of the octane data (for which $k = 2$) for selected cases. The circles on the right of the cellmap indicate whether $\pi(\boldsymbol{x}_i)$ is considered casewise outlying in the principal subspace, according to $\text{SD}(\pi(\boldsymbol{x}_i)) < \sqrt{\chi^2_{k,0.99}}$ (white) or $\text{SD}(\pi(\boldsymbol{x}_i)) \geqslant \sqrt{\chi^2_{k,0.99}}$ (black). So points to the left of the red vertical line in the drop plot get white circles, and the others obtain black circles.

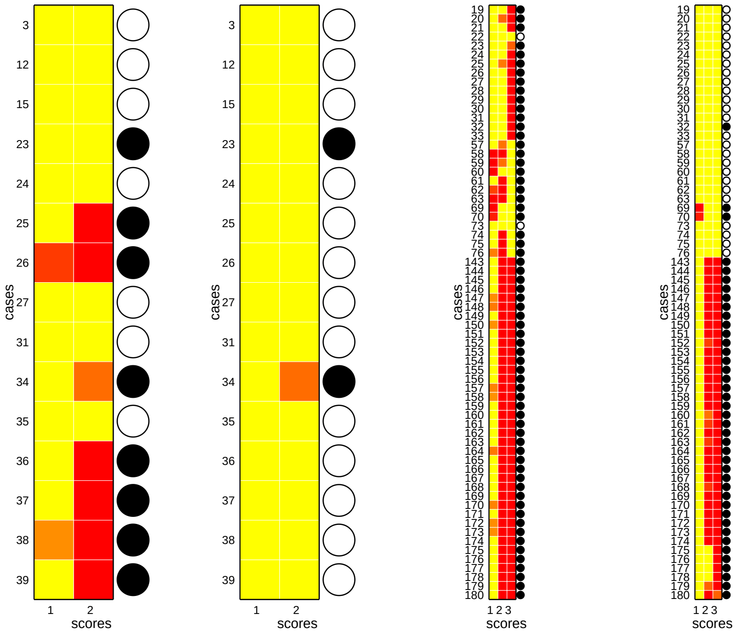


Figure 13: Score cellmaps of the octane data (left panel) and the glass data (right panel). Each panel first shows the scores of the projected points $\pi(\boldsymbol{x}_i)$ followed by those of the fitted points $\widehat{\boldsymbol{x}}_i$.

We can also draw a cellmap of the scores of the $\widehat{\boldsymbol{x}}_i$ fitted by cellPCA. This is the second cellmap in the left panel of Figure 13. Now the circles to its right are based on $\text{SD}(\widehat{\boldsymbol{x}}_i)$. Comparing the score cellmap of the $\pi(\boldsymbol{x}_i)$ to that of the $\widehat{\boldsymbol{x}}_i$ gives some idea about the effect of imputation on the scores. For the octane data we see that the fitted $\widehat{\boldsymbol{x}}_i$ have fewer outlying scores than the projections $\pi(\boldsymbol{x}_i)$, and the rows have fewer black circles.

The right panel of Figure 13 shows score cellmaps of the glass data (for which $k = 3$). As we have seen in the drop plot, the most extreme outlying group remains outlying in the PCA subspace after imputation, whereas most of the scores of the less extreme outlying groups become inlying.

3.5. Silhouette plot

Combining the distance $d_{1,i}$ (of the point $\boldsymbol{x}_i$ to its imputation $\boldsymbol{x}_i^{\text{imp}}$) with the distance $d_{2,i}$ (of $\boldsymbol{x}_i^{\text{imp}}$ to the fitted $\widehat{\boldsymbol{x}}_i$) can give an idea of how well the principal subspace fits the case $\boldsymbol{x}_i$. We compute the ratio between $d_{2,i}$ and $d_{1,i} + d_{2,i}$, the total distance that $\boldsymbol{x}_i$ travels to get to $\widehat{\boldsymbol{x}}_i$. This yields

$$s(i) = \frac{d_{2,i}}{d_{1,i} + d_{2,i}} \;.$$

We call the resulting $s(i)$ the *silhouette width* of case i, by analogy to the silhouette width of Rousseeuw (1987) which describes how well a case is fitted by a clustering. In the best situation all cells of $\boldsymbol{x}_i$ have weight 1 so no imputation is needed, hence $d_{1,i} = 0$ and therefore $s(i) = 1$. In the worst situation all the cells of case $\boldsymbol{x}_i$ have zero cellwise weight, and then $\boldsymbol{x}_i^{\text{imp}}$ coincides with $\widehat{\boldsymbol{x}}_i$ which yields $s(i) = 0$.

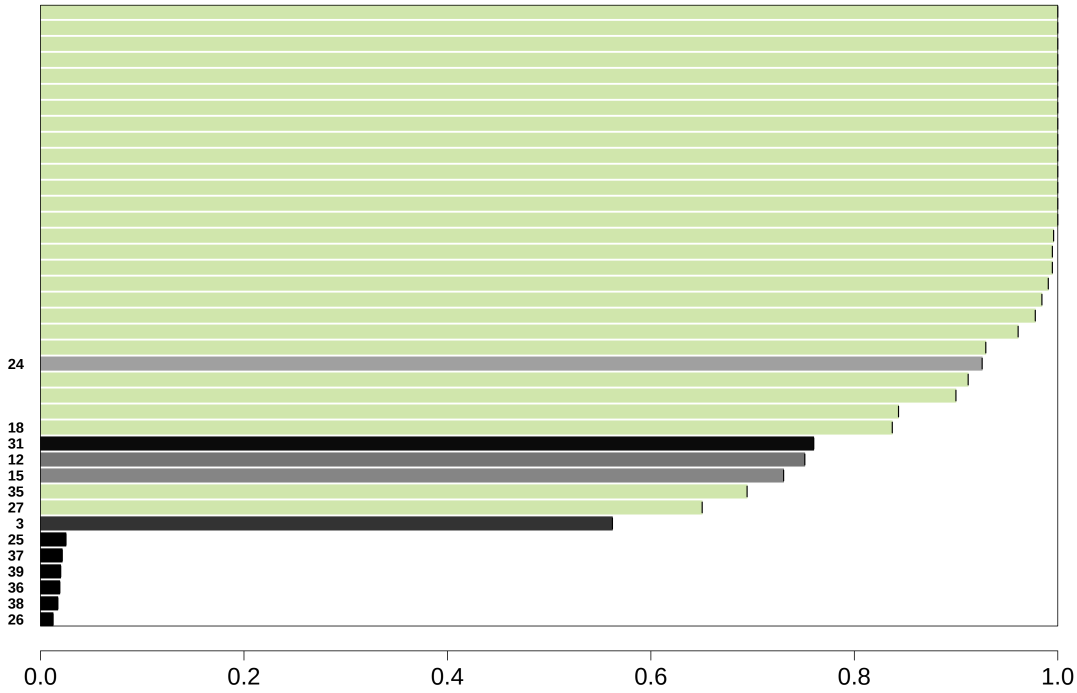


Figure 14: Silhouette plot of the octane data.

In our PCA version of the silhouette plot, the $s(i)$ are shown as the lengths of horizontal bars, ordered from longest to shortest. Figure 14 is the silhouette plot of the octane data. The $s(i)$ summarize information on the cells of the datapoints. To this the plot adds information on casewise deviations. The colors of the horizontal bars are exactly the same as in the shaded residual cellmap and the functional residual cellmap. We see that the darkest horizontal bars occur near the bottom of the plot. This is because cases with high casewise deviations typically require much imputation. However, some cases do not follow this pattern. For cases 3 and 31 this was explained by the functional residual cellmap in the top panel of Figure 9.

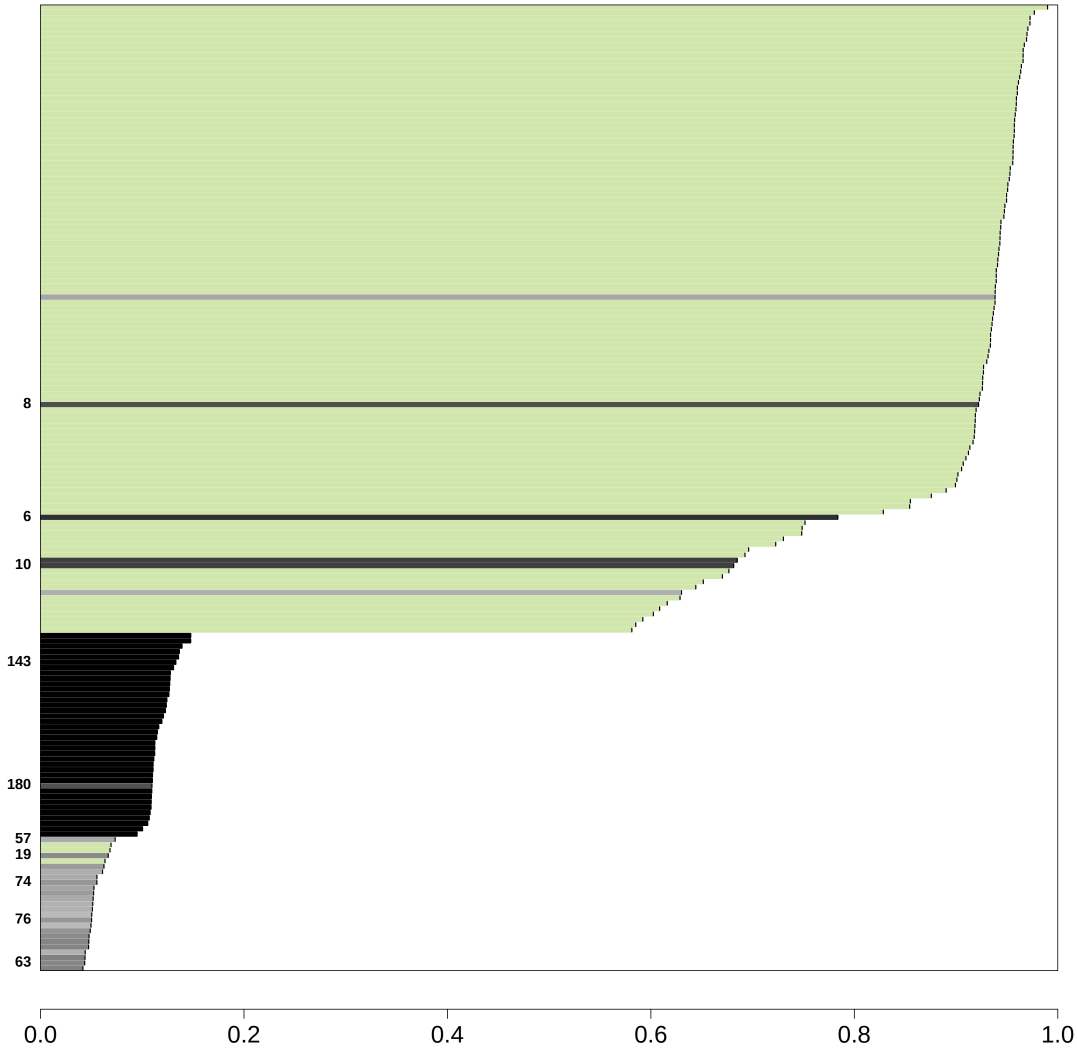


Figure 15: Silhouette plot of the glass data.

Figure 15 shows the silhouette plot of the glass data. Case 180 is in a lighter grey compared to its neighbors, which are part of the same group of outliers. This case indeed behaves somewhat differently, as we saw in the residual cellmap. We also note that the less extreme outlying groups, shown in lighter grey, have lower $s(i)$ values indicating that they were imputed more heavily than cases 143–180. The distinction arises because they are outlying for different reasons, as we saw in the description of the dataset.

4. A Video Data Example

The dog walker data is a surveillance video of a man walking his dog. The video was filmed using a static camera and contains 54 color frames. Each case is a frame with 72×90 pixels stored using the RGB (Red, Green, Blue) color model. Figure 16 shows three frames of the video. We see that the man walks from the left to the right in a street. The data for the blue channel is available in the supplementary material. Each cell of the data corresponds to a pixel, and its value is the intensity on a scale from zero to one.

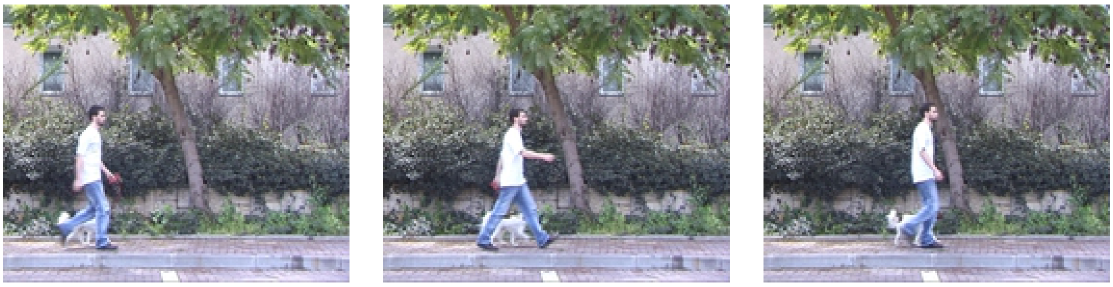


Figure 16: Frames of the dog walker data.

In order to be able to apply cellPCA we first turn every frame into a vector, by concatenating the columns of the frame. The data matrix thus has $n = 54$ rows, one for each frame, and $p = 72 \times 90 = 6{,}480$ columns, one for each pixel. Next, we apply cellPCA for $k = 2$ components. The resulting residual cellmap is in Figure 17. Since we cannot plot 6,480 cells side by side, we asked the cellmap function to combine 25 cells of a row into a single cell.

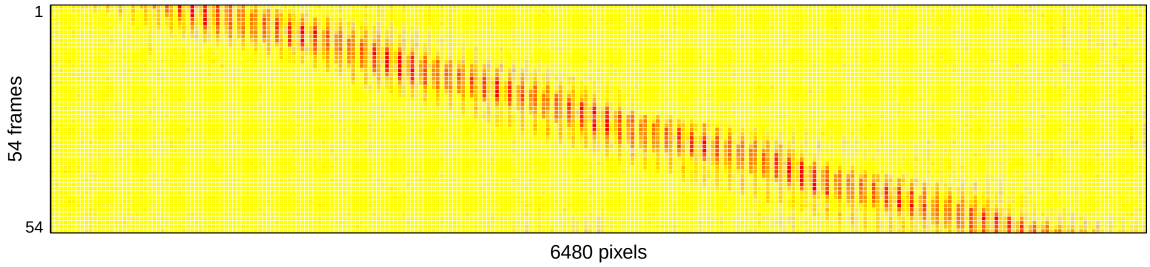


Figure 17: Residual cellmap of the dog walker dataset.

Figure 17 mainly contains yellow cells, plus some outlying cells in red, that move to the right in subsequent frames (rows). This is the man with the dog, walking from left to right. Their movement is outlying relative to the static background.

In order to better visualize what is happening, we can undo the concatenation so the residuals of each frame are again in the form of a matrix that has the same dimensions as the original image. For instance, in Figure 18, we see the residual cellmaps of the three frames in Figure 16.

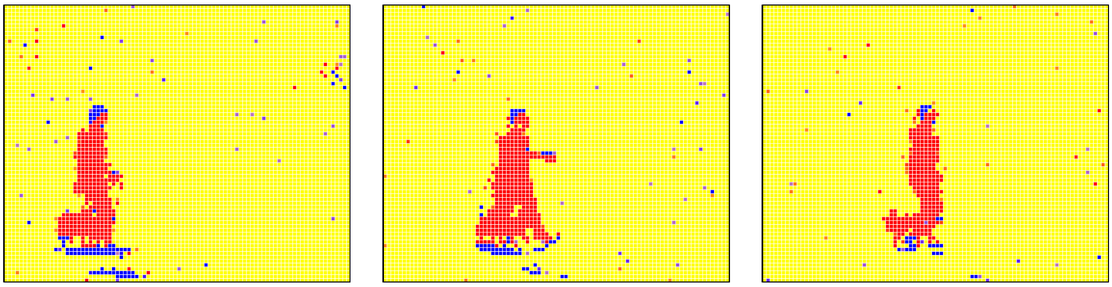


Figure 18: Residual cellmaps of 3 frames of the dog walker data.

We could also show the residual cellmaps of each frame one after the other, so it becomes a video. Alternatively, we can draw the 3D residual cellmap in Figure 19. There the frame number corresponds to time, and the other dimensions are the horizontal and vertical coordinates. It gives a nice overview of outlying cells. We can clearly see the man walking his dog, which are outliers relative to the static background with mainly yellow cells, that are 3D voxels here. The 3D display is interactive: after it is produced, the user can rotate it in all directions to explore it.

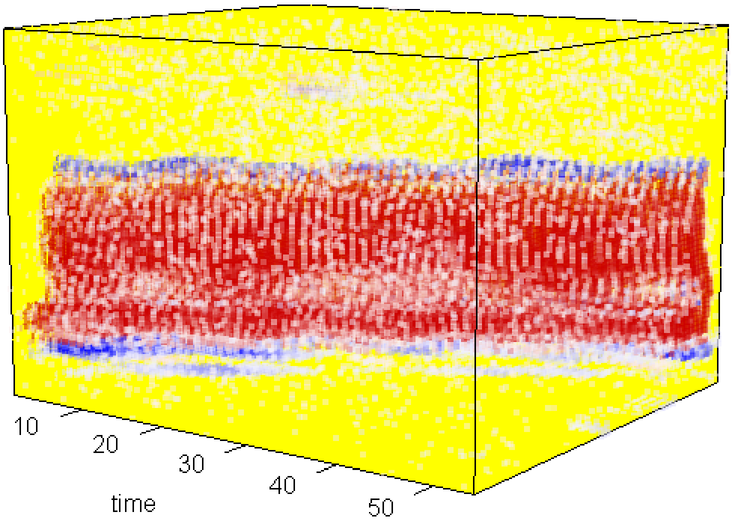
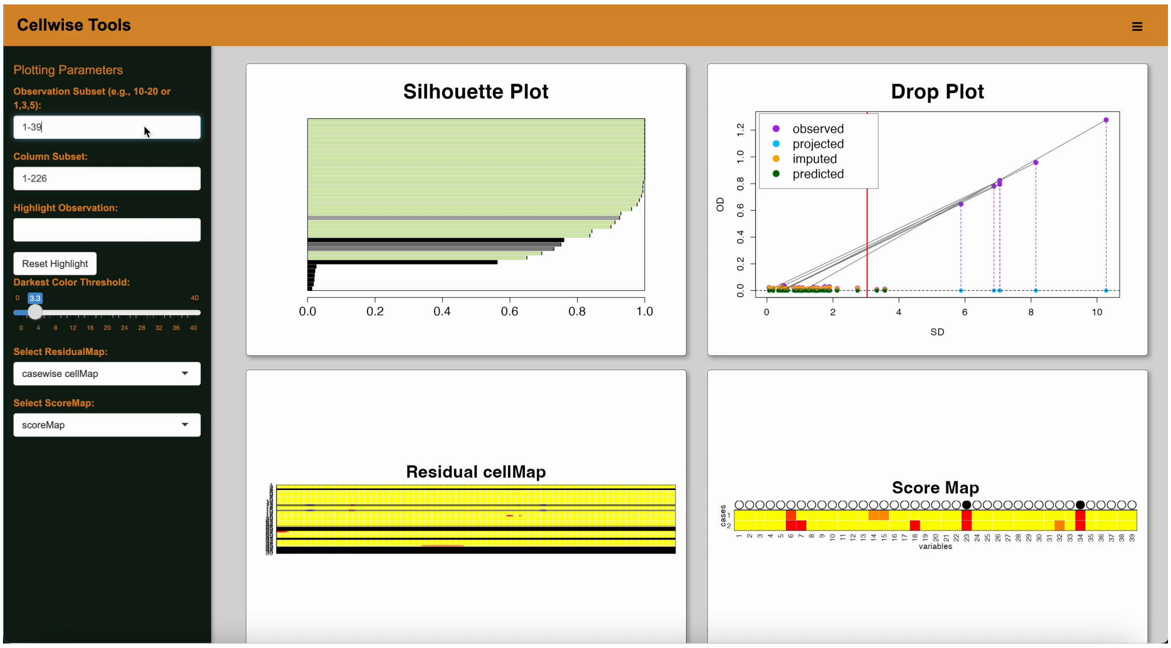


Figure 19: 3D Residual cellmap of the dog walker dataset.

5. Shiny App

The diagnostic tools we have described can be used together, as each of the displays offers a different view of the data and the outliers. A Shiny app has been developed to combine all the plots in one interactive platform. The video in Figure 19 shows a tutorial of the Shiny app.



Cellwise Tools

Plotting Parameters

Observation Subset (e.g., 10-20 or 1,3,5): 1-39

Column Subset: 1-226

Highlight Observation:

Reset Highlight

Darkest Color Threshold: 3.3

Select ResidualMap: casewise cellMap

Select ScoreMap: scoreMap

Figure 20: Video of the Shiny app.

6. Conclusion

We have introduced new graphical tools for detecting and visualizing cellwise and casewise anomalies from the output of cellPCA, a robust principal component analysis method. We made extensive use of the residuals and imputations this method provides. The displays open up new possibilities for interpreting outliers.

The proposed displays include a shaded version of the residual cellmap, where a measure of casewise outlyingness adds an additional layer of information. For functional data the new functional cellmap complements the residual cellmap, and allows for a more direct understanding of the nature of the outliers. The drop plot provides a comprehensive view of the imputations that were carried out. We also construct a cellmap of the principal scores of the data. Finally, the new silhouette plot shows how well each case is fitted by the model, making use of the amount of imputation that was required and including casewise information as well.

All these displays were illustrated on real data, with interpretations of outlier behavior. The paper concluded with a demo of the Shiny app we have developed to integrate all of these tools into an interactive platform.

Supplementary Material

This is a zip file with the R code and an example script that reproduces all the visualizations in the paper, as well as the Shiny app illustrated in Section 5.

References

Alqallaf, F., Van Aelst, S., Yohai, V. J., and Zamar, R. H. (2009). Propagation of outliers in multivariate data. *The Annals of Statistics*, 37:311–331, DOI: 10.1214/07-AOS588.

Centofanti, F., Hubert, M., and Rousseeuw, P. J. (2024). Robust principal components by casewise and cellwise weighting. *arXiv*, https://arxiv.org/abs/2408.13596.

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, DOI: 10.1016/j.jmva.2004.08.002.

De La Torre, F. and Black, M. J. (2003). A framework for robust subspace learning. *International Journal of Computer Vision*, 54:117–142, https://link.springer.com/article/10.1023/A:1023709501986.

Esbensen, K., Midtgaard, T., and Schönkopf, S. (1996). *Multivariate Analysis in Practice: A Training Package*. Camo As, Oslo. DOI: 10.1002/cem.1180090609.

Hubert, M., Rousseeuw, P. J., and Van den Bossche, W. (2019). MacroPCA: An all-in-one PCA method allowing for missing values as well as cellwise and rowwise outliers. *Technometrics*, 61:459–473, https://www.tandfonline.com/doi/full/10.1080/00401706.2018.1562989.

Hubert, M., Rousseeuw, P. J., and Vanden Branden, K. (2005). ROBPCA: A new approach to robust principal component analysis. *Technometrics*, 47:64–79, https://www.tandfonline.com/doi/abs/10.1198/004017004000000563.

Lemberge, P., De Raedt, I., Janssens, K., Wei, F., and Van Espen, P. J. (2000). Quantitative analysis of 16–17th century archaeological glass vessels using PLS regression of EPXMA and μ-XRF data. *Journal of Chemometrics*, 14:751–763, https://analyticalsciencejournals.onlinelibrary.wiley.com/doi/abs/10.1002/1099-128X%28200009/12%2914%3A5/6%3C751%3A%3AAID-CEM622%3E3.0.CO%3B2-D.

Locantore, N., Marron, J., Simpson, D., Tripoli, N., Zhang, J., and Cohen, K. (1999). Robust principal component analysis for functional data. *Test*, 8:1–73, https://link.springer.com/article/10.1007/BF02595862.

Maronna, R. A. and Yohai, V. J. (2008). Robust low-rank approximation of data matrices with elementwise contamination. *Technometrics*, 50(3):295–304, https://www.jstor.org/stable/25471491.

Raymaekers, J. and Rousseeuw, P. J. (2023). ***cellWise**: Analyzing Data with Cellwise Outliers*. R package, CRAN, https://CRAN.R-project.org/package=cellWise.

Rousseeuw, P. J. (1987). Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, https://www.sciencedirect.com/science/article/pii/0377042787901257.

Rousseeuw, P. J. and Van den Bossche, W. (2018). Detecting deviating data cells. *Technometrics*, 60:135–145, DOI: `10.1080/00401706.2017.1340909`.

Todorov, V. (2025). ***rrcov**: Scalable Robust Estimators with High Breakdown Point*. R package, CRAN, `https://CRAN.R-project.org/package=rrcov`.

Affiliation:

Mehdi Hirari
Section of Statistics and Data Science
Department of Mathematics
KU Leuven
Celestijnenlaan 200B
BE-3001 Leuven, Belgium
E-mail: mehdi.hirari@kuleuven.be
URL: https://wis.kuleuven.be/statdatascience/robust

Mia Hubert
Section of Statistics and Data Science
Department of Mathematics
KU Leuven
Celestijnenlaan 200B
BE-3001 Leuven, Belgium
E-mail: mia.hubert@kuleuven.be
URL: https://wis.kuleuven.be/statdatascience/robust

Peter J. Rousseeuw
Section of Statistics and Data Science
Department of Mathematics
KU Leuven
Celestijnenlaan 200B
BE-3001 Leuven, Belgium
E-mail: peter.rousseeuw@kuleuven.be
URL: https://wis.kuleuven.be/statdatascience/robust

Journal of Data Science, Statistics, and Visualisation https://jdssv.org/
published by the International Association for Statistical Computing
http://iasc-isi.org/

ISSN 2773-0689
December 2025, Volume V, Issue X *Submitted:* 2025-10-26
doi:10.52933/jdssv.v5i10.165 *Accepted:* 2025-12-15