Robust Imputation of Missing Values in Compositional Data Using the R-Package robCompositions

Matthias Templ\textsuperscript{1,2}, Peter Filzmoser\textsuperscript{1}, Karel Hron\textsuperscript{3}

\textsuperscript{1} Department of Statistics and Probability Theory, TU WIEN, Austria
\textsuperscript{2} Department of Methodology, Statistics Austria
\textsuperscript{3} Department of Mathematical Analysis and Applications of Mathematics, Palacký University, Olomouc, Czech Republic

Brussels, Feb. 19, 2009
1. Compositional Data
2. Transformation
3. Imputation Methods
4. Simulation Results
5. R-package robCompositions
6. Conclusion
Compositional (Closed) Data

- **Multivariate data** that sum up to a constant (e.g., 100%):

\[
x = (x_1, \ldots, x_D)^t, \quad x_i > 0, \quad \sum_{i=1}^{D} x_i = \kappa
\]

(the constant \(\kappa\) could be different for each observation as well)

- The set of all closed observations with positive values forms a **simplex sample space**.

- **the ratios** between the parts are of interest.
Compositional (Closed) Data

- **Multivariate data** that sum up to a constant (e.g. 100%):

\[ x = (x_1, \ldots, x_D)^t, \quad x_i > 0, \quad \sum_{i=1}^{D} x_i = \kappa \]

(the constant \( \kappa \) could be different for each observation as well)

- The set of all closed observations with positive values forms a **simplex sample space**.

- the **ratios** between the parts are of interest.

**Key reference:**
Compositional Data: Example

- Two scatter plots are shown:
  1. Left plot: high-quality production vs. low-quality production.
  2. Right plot: log(high-quality production/machine repair) vs. log(low-quality production/machine repair).

- The data points are represented by plus signs (+).

- The x-axis of the left plot ranges from 0.50 to 0.70.
- The y-axis of the left plot ranges from 0.10 to 0.30.
- The x-axis of the right plot ranges from 1.0 to 2.5.
- The y-axis of the right plot ranges from -0.5 to 1.0.
### Compositional Data: Example

<table>
<thead>
<tr>
<th></th>
<th>qu1</th>
<th>qu2</th>
<th>qu3</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>52</td>
<td>42</td>
<td>6</td>
<td>100%</td>
</tr>
<tr>
<td>2</td>
<td>52</td>
<td>44</td>
<td>4</td>
<td>100%</td>
</tr>
<tr>
<td>3</td>
<td>47</td>
<td>48</td>
<td>5</td>
<td>100%</td>
</tr>
</tbody>
</table>

...
### Compositional Data: Example

<table>
<thead>
<tr>
<th></th>
<th>qu1</th>
<th>qu2</th>
<th>qu3</th>
<th>sum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>52</td>
<td>42</td>
<td>6</td>
<td>100%</td>
</tr>
<tr>
<td>2</td>
<td>52</td>
<td>44</td>
<td>4</td>
<td>100%</td>
</tr>
<tr>
<td>3</td>
<td>47</td>
<td>48</td>
<td>5</td>
<td>100%</td>
</tr>
<tr>
<td>22</td>
<td>14</td>
<td>47</td>
<td>39</td>
<td>100%</td>
</tr>
<tr>
<td>23</td>
<td>24</td>
<td>56</td>
<td>20</td>
<td>100%</td>
</tr>
</tbody>
</table>

[Diagram showing a ternary plot with points marked at various coordinates.]
## Compositional Data: Expenditures

<table>
<thead>
<tr>
<th></th>
<th>housing</th>
<th>foodstuff</th>
<th>alcohol</th>
<th>tobacco</th>
<th>other goods</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>640</td>
<td>328</td>
<td>147</td>
<td>169</td>
<td>196</td>
</tr>
<tr>
<td>2</td>
<td>1800</td>
<td>484</td>
<td>515</td>
<td>2291</td>
<td>912</td>
</tr>
<tr>
<td>3</td>
<td>2085</td>
<td>445</td>
<td>725</td>
<td>8373</td>
<td>1732</td>
</tr>
<tr>
<td>4</td>
<td>616</td>
<td>331</td>
<td>126</td>
<td>117</td>
<td>149</td>
</tr>
<tr>
<td>5</td>
<td>875</td>
<td>368</td>
<td>191</td>
<td>290</td>
<td>275</td>
</tr>
<tr>
<td>6</td>
<td>770</td>
<td>364</td>
<td>196</td>
<td>242</td>
<td>236</td>
</tr>
</tbody>
</table>

...
Compositional Data: Example

**Left plot**: Two-part compositional data **without** the constraint of constant sum. The points could be varied along the lines from the origin **without** changing the ratio of the compositional parts.

**Right plot**: The points at the boundary are more distant than the central points. The **Aitchison distance** accounts for this fact.
Aitchison Distance and the Simplex

A distance measure that is accounting for this relative scale property is the Aitchison distance (Aitchison, 1992, Aitchison et al., 2000), defined for two compositions \( x = (x_1, \ldots, x_D)^t \) and \( y = (y_1, \ldots, y_D)^t \) as

\[
d^2_A(x, y) = \frac{1}{D} \sum_{i=1}^{D-1} \sum_{j=i+1}^{D} \left( \ln \frac{x_i}{x_j} - \ln \frac{y_i}{y_j} \right)^2.
\]

As an example, the boundary points in the previous Figure (right) have an Aitchison distance of 0.33, whereas the central points have Aitchison distance 0.08.

Replacing the Euclidean distance by the Aitchison distance is necessary because the simplex sample space has a different geometrical structure than the classical Euclidean space.
Logratio Transformations

Family of one-to-one transformations from the simplex to the real space (Aitchison, 1986):

- additive logratio (alr) transformation
- centred logratio (clr) transformation
- isometric logratio (ilr) transformation
Divide all values by the $j$-th part:

$$x^{(j)} = \left( x_1^{(j)}, \ldots, x_{D-1}^{(j)} \right)^t = \left( \log \frac{x_1}{x_j}, \ldots, \log \frac{x_{j-1}}{x_j}, \log \frac{x_{j+1}}{x_j}, \ldots, \log \frac{x_D}{x_j} \right)^t$$

The index $j \in \{1, \ldots, D\}$ refers to the “ratioing” variable.
clr Transformation

Divide all values by the geometric mean:

\[ y = \left( y_1, \ldots, y_D \right)^t = \log \frac{x_1}{\sqrt[\prod_{i=1}^D x_i}^D}, \ldots, \log \frac{x_D}{\sqrt[\prod_{i=1}^D x_i}^D} \]  

**Advantage:** symmetric with respect to variables, easier interpretation

**Disadvantage:** singularity problem, because

\[
\log \frac{x_1}{\sqrt[\prod_{i=1}^D x_i}^D} + \cdots + \log \frac{x_D}{\sqrt[\prod_{i=1}^D x_i}^D} = \\
\sum_{j=1}^D \log(x_j) - \frac{1}{D} \sum_{j=1}^D \sum_{i=1}^D \log(x_i) = 0
\]
ilr Transformation

Take an orthonormal basis \( \mathbf{V} = (\mathbf{v}_1, \ldots, \mathbf{v}_{D-1}) \) (of dimension \( D \times D - 1 \)) with

\[
\mathbf{v}_i = \sqrt{i \over i + 1} \left( {1 \over i}, \ldots, {1 \over i}, -1, 0, \ldots, 0 \right)
\]

for \( i = 1, \ldots, D - 1 \), in the hyperplane \( \mathcal{H} : y_1 + \cdots + y_D = 0 \) in \( \mathbb{R}^D \).

The ilr-transformed data are

\[
\mathbf{z} = (z_1, \ldots, z_{D-1})^t = \mathbf{V}^t \mathbf{y}.
\]

\( z_i \) are coefficients to the chosen basis.

**Advantage:** no singularity problem, good geometric properties

**Disadvantage:** \( z_i \) is not easy to interpret.
Properties of the ILR Transformation

**Left plot:** Two-part compositional data without the constraint of constant sum (symbols ○), and projections on the line indication a constant sum of 1 (symbols +).

**Right plot:** In the upper part the ilr transformed original data (with symbols ○ are shown. The lower plot shows the ilr transformed data with constant sum constraint (symbols +). This demonstrates that the constant sum constraint does not change the ilr transformed data.
If, for example, the missing values are mainly contained in the first compositional part of the data, one can choose the ilr transformation as

\[ ilr(x) = (z_1, \ldots, z_{D-1})^t, \quad z_j = \sqrt{\frac{D - j}{D - j + 1}} \ln \frac{D - j}{\prod_{l=j+1}^{D} x_l} x_j \]

with \( j = 1, \ldots, D - 1 \).

Only this choice of the balances guarantees that missing values in \( x_1 \) does not affect \( z_2, \ldots, z_{D-1} \).
Outliers

Left plot: Two-part compositional data consisting of three groups. While the relative information of the groups with symbols ◦ and + is similar, the data points corresponding to the open triangles contain very different information.

Right plot: The ilr transformed data reveal that the group with open triangles are indeed different. They are potentially influencing non-robust statistical methods.
KNN Imputation

When imputing one missing value we

- use the **Aitchison distance** to find $k$ nearest neighbors.
- adjust the corresponding cells according to the overall size of the parts.
- take the median of these cells to impute the missing.
Iterative Model Based Imputation

- Start: knn solution.
- Order the data so that the first variable includes the highest amount of missing values, ...
- Until convergence:
  - For $i$ in $1 : D$
    - Apply a specific, well-defined ilr-transformation
    - Update former missing values in $z_i$ by regression imputation in the ilr-space; $z_i$ is chosen as the response variable.
    - back-transformation to the original space
  - end inner “loop”
Simulated Data
Simulation Results

Results

![Graph 1](image1.png)

- knn (Euclidean), $k=6$
- knn (Euclidean), $k=8$
- knn (Euclidean), $k=10$
- iterative LS (no transf.)
- iterative LTS (no transf.)

![Graph 2](image2.png)

- knn (Aitchison), $k=6$
- knn (Aitchison), $k=8$
- knn (Aitchison), $k=10$
- iterative LS (ilr)
- iterative LTS (ilr)
Usage

http://cran.r-project.org/web/packages/robCompositions/index.html

> library(robCompositions)
> help(package=robCompositions)

Description:

Package: robCompositions
Type: Package
Title: Robust Estimation for Compositional Data.
Version: 1.1
Date: 2009-01-22
Depends: utils, e1071, robustbase, compositions, car, MASS
Author: Peter Filzmoser, Karel Hron, Matthias Templ
Maintainer: Matthias Templ <templ@tuwien.ac.at>
Description: This first version of the package includes methods for
imputation of compositional data including robust
methods and Anderson-Darling normality tests for
compositional data. The package will be enhanced with
other multivariate methods for compositional data in
near future.
License: GPL-2
LazyLoad: yes
Built: R 2.8.0; ; 2009-01-22 16:53:39; windows

Index:
Data

We use the randomly generated data as used in the previous Figure.

```r
> head(x)

[,1] [,2] [,3]
[1,] 0.29395572 0.16181078 0.09169296
[2,] 0.24290463 0.24092547 0.16041012
[3,] NA 0.05278444 0.51727452
[4,] NA 0.09599913 0.11838661
[5,] 0.31172499 0.22095742 0.35843191
[6,] 0.02038967 0.04858723 0.55728004

> dim(x)

[1] 100  3
Data
Missing Values

```r
> library(VIM)
> plot(aggr(x))
```
Missing Values

```r
> library(VIM)
> plot(aggr(x))
```
Imputation with robCompositions

```r
> xImp <- impKNNa(x, k = 6)
> class(xImp)
```
Imputation with robCompositions

```r
> xImp <- impKNNa(x, k = 6)
> class(xImp)

[1] "imp"
```
Imputation with robCompositions

> xImp <- impKNNa(x, k = 6)
> class(xImp)

[1] "imp"

> methods(class = "imp")
Imputation with robCompositions

```r
> xImp <- impKNNa(x, k = 6)
> class(xImp)

[1] "imp"

> methods(class = "imp")

[1] plot.imp print.imp summary.imp
```
Imputation with robCompositions

```r
> xImp <- impKNNa(x, k = 6)
> class(xImp)

[1] "imp"

> methods(class = "imp")

[1] plot.imp print.imp summary.imp

> xImp
```
Imputation with robCompositions

```r
> xImp <- impKNNa(x, k = 6)
> class(xImp)

[1] "imp"

> methods(class = "imp")

[1] plot.imp print.imp summary.imp

> xImp

---------------------------------------
[1] "31 missing values were imputed"
---------------------------------------
> names(xImp)

> xImp$xImp[1,3]

> xImp1 <- impCoda(x, method = "lm")

> xImp2 <- impCoda(x, method = "ltsReg")
Imputation with robCompositions

```r
> names(xImp)

[1] "xOrig" "xImp" "criteria" "iter" "w" "wind" "metric"
```
Imputation with robCompositions

```r
> names(xImp)
[1] "xOrig" "xImp" "criteria" "iter" "w" "wind" "metric"

> xImp$xImp[1, 3]
```
Imputation with robCompositions

> names(xImp)

[1] "xOrig" "xImp" "criteria" "iter" "w" "wind" "metric"

> xImp$xImp[1, 3]

[1] 0.09169296
> names(xImp)

[1] "xOrig" "xImp" "criteria" "iter" "w" "wind" "metric"

> xImp$xImp[1, 3]

[1] 0.09169296

> xImp1 <- impCoda(x, method = "lm")
> xImp2 <- impCoda(x, method = "ltsReg")
\texttt{plot(xImp2, which = 1)}
Diagnostics

> plot(xImp2, which = 3)
We tested more than 20 imputation procedures which all were outperformed by our method (Hron, Templ, Filzmoser, 2008) for compositional data.

Robustness is an issue. We proposed new robust imputation methods for compositional data.

R-package robCompositions includes these methods, but other methods are implemented as well. Diagnostic tools are available within the package.

A lot of important issues were not mentioned in this presentation, but they have been discussed in our NTTS-paper or in Hron, Templ, Filzmoser (2008).