TECHNISCHE UNIVERSITÄT WIEN

DIPLOMARBEIT

Dimension reduction for compositional data with weights based on graph theory

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Statistik und Wirtschaftsmathematik

unter der Anleitung von

Univ.Prof. Dipl.-Ing. Dr.techn. Peter Filzmoser

eingereicht von

Jeremy Oguamalam

Matrikelnummer: 01616719

ausgeführt am Institut für Stochastik und Wirtschaftsmathematik

der Fakultät für Mathematik und Geoinformation

an der Technischen Universität Wien

Wien, am May 25, 2022

(Unterschrift Verfasser) (Unterschrift Betreuer)

Abstract

A popular tool of dimension reduction in many statistical fields is principal component analysis (PCA). For the field of compositional data analysis (CoDA) weighting can be seen as a similar approach of dimension reduction as PCA. It is a desire to find those variables which explain a big part or even the majority of the variance of the whole data. These variables are transformed into a coordinate system where they are expressed by ratios of their logarithms. This concept is referred to as logratios and has many practical advantages. In the considered framework of the Aitchison geometry, weighting can be incorporated into the Aitchison inner product. Combined with graph theory, the weights can be related to the covariance of the distribution of the underlying data. These thoughts lead to so-called inverse variance problems. Next to a short introduction into compositional data, such a problem is considered in this thesis. An iterative algorithm is introduced to estimate a Laplacian matrix that is connected to the distribution of the compositional data. This eventually leads to a sparse solution while keeping the explained variance high.

Kurzfassung

Beobachtet man multivariate, strikt positive Merkmale, welche relative Information zwischen jenen Variablen enthalten, spricht man heutzutage von Kompositionsdaten (englisch: compositional data). Nicht nur in der multivariaten Statistik, sondern auch im Bereich von Kompositionsdaten ist die Hauptkomponentenanalyse (HKA, englisch: principal component analysis) eine begehrte Methode der Dimensionsreduktion von Datensätzen. Verschiedene Merkmale in den beobachteten Daten unterschiedlich zu gewichten, ist eine weitere Möglichkeit, welche die Dimension der Daten unter möglichst geringem Informationsverlust reduziert. Die dabei relevanten Größen werden von den Verhältnisse der Logarithmen der einzelnen Variablen gebildet. Diese Verhältnisse bezeichnet man im Englischen als logratios. Gemeinsam mit der zugrundeliegenden Aitchison-Geometrie, kann die Gewichtung der Daten in das sogenannte Aitchison Skalarprodukt integriert werden. Graphentheorie spielt ebenfalls eine wichtige Rolle, da sie die Struktur der Daten und deren Verteilung mit den Gewichten in Verbindung setzen kann. Dieses Konzept führt zu sogenannten Inversen Kovarianz Problemen. Solch ein Optimierungsalgorithmus wird in dieser Arbeit vorgestellt. Eine iterative Herangehensweise schätzt die sogenannte Laplace Matrix, welche in einem direkten Zusammenhang mit der Verteilung der Daten und den gesuchten Gewichten steht. Diese Matrix ermöglicht eine hinreichend gute Erklärung der Daten in niedriger Dimension.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Bachelorarbeit selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Wien, am 25. Mai 2022

Name des Autors

Acknowledgment

I want to thank everyone who supported me on a technical as well as emotional level during the last few years, especially during the writing process of this thesis.

I am extremely grateful to my supervisor Univ.Prof. Dipl.-Ing. Dr.techn. Peter Filzmoser, TU Wien, for giving me the opportunity to finish my masters under his guidance. I would like to thank you for your valuable advice and the insightful suggestions I received.

I would like to extend my deepest gratitude to Projektass. Dipl.-Ing. Dr.techn. Christopher Rieser MSc., TU Wien, your extensive knowledge on the subject area paired with the helpful contributions.

Furthermore, many thanks to my learning group composed out of Roman Becker, Marcus Mayrhofer and Stefan Spieß. You all played a decisive role through my whole study through mutual assistance on a daily basis. It was a great pleasure studying with you and always being able to count on your helping hands.

A special thanks to Asita Aue. I very much appreciate your love and relentless moral support during any situation. You always motivated to keep it rolling.

Finally, I would also like to acknowledge my family and close friends for their friendly ears, their time and their profound beliefs in my abilities.

Contents

1 Introduction

Compositional data became popular in the early 80s by the work of Aitchison (1982) and have stayed relevant since. Originally they were viewed as multivariate observations on the positive unit simplex, i.e. as vectors which entries would sum up to one. Due to several practical reasons like missing values or different sums between observations, later characterizations as in Egozcue (2009) or Pawlowsky-Glahn et al. (2015) defined them as strictly positive vectors that carry relative information between the components.

According to the former, the field of compositional data should follow the principles of scale invariance, permutation invariance and subcompositional coherence. Three important properties that distinguish them from other statistical areas.

The Aitchison geometry (Pawlowsky-Glahn and Egozcue, 2001) replaces the standard Euclidean geometry as common statistical methods cannot be applied directly onto compositional data. In this framework the simplex of strictly positive vectors forms the sample space. Analogously to the vector addition and scalar multiplication of the Euclidean space operators like perturbation and powering are defined respectively. Furthermore, an inner product, called the Aitchison inner product, was proposed to obtain a Hilbert space. Analysis can theoretically be conducted directly in this setting, but the output would be unreliable.

For that reason a family of transformations to the Euclidean space was introduced. This can also be seen as a representation of compositional data in a coordinate system. That is why they are also referred to as coordinates. Additive logratios coordinates and centered logratio coefficients were already introduced in Aitchison (1982) and Aitchison (1983). Today the primary focus lays on isometric logratios coordinates, as in contrast to the former, they form a basis a hyperplane. The orthonormal or pivot coordinates, as the are also called, additionally lead to a more intuitive interpretation of the resulting vectors.

All of these transformations revolve around the ratios of the logarithms of the data and are called logratios. These ratios are regarded as the source of relative information in compositional data. They not only make interpretations possible, but also direct application of standard statistical methodology like principal component analysis.

Speaking of which, principal component analysis is a well known tool for dimension reduction that can easily be extended to the compositional setting. There the principal components can be derived directly through the data matrix of the isometric logratio coordinates by singular value decomposition. Alternatively an eigenvalue decomposition of an robust estimate of the covariance can lead to more sophisticated solutions.

Another way of dimension reduction can be achieved by integrating weights into the

Aitchison inner product. The idea behind this approach is, that there might be variables which are more relevant for the analysis while others do not have a high contribution to following investigations. Accordingly, features with a higher impact should get higher weights, while less relevant variables should receive smaller weights.

As it turns out, these weights can be linked to the underlying distribution of the data. Graph theory provides the key information to do so. Rieser and Filzmoser (2022) describes that relationship with the use of the so-called Laplacian matrix which is determined by the underlying graph structure of the data. With some properties of the Laplacian following from Mohar (1991), the estimation of weights can be translated into various different optimization problems as seen in Yuan and Lin (2007), Friedman et al. (2008) or Holbrook (2018).

In this thesis an iterative approach to these inverse covariance problems is presented. Derived from principal component analysis it iteratively solves a minimization problem similar to the ones above. An additional sparsity term tries to function as a trade-off between the sparsity of the resulting solution and the explained variance of the projected data. As will be shown, it is possible to reduce the sparsity of the Laplacian significantly while keeping the explained information sufficiently high.

The structure of this thesis is as follows. The second chapter introduces the framework of compositional data combined with the Aitchison geometry and its operators. Furthermore, some thoughts on descriptive statistics are mentioned. Chapter 3 extends principal component analysis to the compositional case and presents different methods to estimate the solution. In the fourth chapter the focus shifts to the concepts of weighting, where graph theory and compositional data analysis are connected. Distributional assumptions next to the class of inverse covariance problems are explained. Finally, an iterative approach to weight estimation is proposed. In the following chapter the algorithm is analyzed by 5-fold cross validation and evaluated on the basis of a real world data set in Chapter 6. The final chapter concludes.

The implementation of the algorithm discussed in this work will be done with the statistical software `R` (R Core Team, 2019) in the development environment `RStudio` (RStudio Team, 2020). For the main part of the code the package for convex optimization problems `CVXR` (Fu et al., 2020) is of most importance. It will be described in detail in Section 5.3. For quick and efficient data manipulation the packages `dplyr` (Wickham et al., 2021) and `data.table` (Dowle and Srinivasan, 2019) are being utilized. To speed up the evaluations over the different parameter configurations are being parallelized with the help of `doParallel` (Corporation and Weston, 2022) and `foreach` (Microsoft and Weston, 2022). The `igraph` package (Csardi and Nepusz, 2006) is used to create graphs according to the various levels of the optimization parameters. The underlying graph structure is then used as an input to randomly simulate the data sets with the `mvtnorm` package (Genz et al., 2021). The following visualizations are done with `ggplot2` (Wickham, 2016).

The code itself will only be partially displayed at some places during the later chapters.

2 An introduction into compositional data analysis

2.1 What is compositional data?

If you talk about compositional data one would assume observations having a special structure. The early definition of Aitchison (1982) postulated compositions as multivariate observations with positive entries which sum up to a constant. That might be the case for proportions or percentages for which the components of the corresponding observations would sum up to 1 or 100. Considering the possibility of variables not being measured or being missing and that rounding errors or different sums between observations could violate the constant sum requirement, later interpretations of Egozcue (2009) or Pawlowsky-Glahn et al. (2015) referred to compositions as strictly positive multivariate vectors which carry relative information between the components. As Filzmoser et al. (2018) argued, these definitions make sense, since if relative information is analyzed, like in the case of compositional data, it is irrelevant whether absolute values, proportions or percentages are present. The ratios between the components are always the same. Accordingly, the value of the sum, which could be different between various observations, is also irrelevant. They further concluded that the actual question, if one deals with compositional data, depends on the interest of the analyst. Thus, a data set might be compositional or not at the same time.

2.2 Applications of compositional data analysis

In order to underline these first thoughts let us start with an example from Filzmoser et al. (2018), where they considered a data set with three observations on household expenditures for different sectors, namely *housing, foodstuff, transport* and *communications*. Other groups are not reported since they are not of importance for this case. In Table 1 the absolute values of expenditures in euros are presented. As we can see, these values differ greatly, also their corresponding sums are not equal. Now, the relative information in form of percentages of expenditures for one sector compared to the total might be of interest. These values are also reported in the table. We notice, that both representations explain the contributions of the variables to the whole, while only for the percentages the values between the observations do not differ. Another possibility would be to consider relative information between the variables themselves. For example

one could look at the ratios where *transport* is in the denominator and the remaining variables form the nominators. The ratio of *housing/transport* for the raw data would result in $1710/570 = 540/180 = 900/300 = 3$ for the three observations. Looking at the percentage data, we get the exact same result in $45/15 = 3$, i.e. all three households spend three times as much for *housing* than for *transport*. Since expressing the original data in a different currency, or rescaling in general, would not alter these ratios, the latter will build the relative information when speaking of compositional data.

Type	Observation	*Housing*	*Foodstuff*	*Transport*	*Communication*	Sum
absolute	1	1710	950	570	570	3800
information	2	540	300	180	180	1200
in EUR	3	900	500	300	300	2000
information	1	45	25	15	15	100
expressed	2	45	25	15	15	100
in %	3	45	25	15	15	100

Table 1: Household expenditures; absolute and relative values.

There are many more common problems from the field of compositional data. If the reader is interested, we refer to Aitchison (2005) at this point for detailed information and corresponding data tables.

For example, the geochemical composition of rocks is studied. Such observations are usually composed out of percentages of weight of oxides or minerals. It might be of interest to explain the variability between different samples or to find atypical compositions. In order to compare specimens of different weights it would be impractical to use absolute values and some kind of standardization to a unit weight is necessary. Equivalently we could say that the results of such an analysis should not depend on the weights of the samples. In Chapter 6 two geochemical data sets are analyzed.

Next to economics and geology other fields like agriculture, manufacturing, medicine or paleontology are mentioned.

2.3 Special treatment for data analysis

After gathering data and performing preprocessing, the next step a statistician would take would be a data analysis. It might seem natural to use already existing techniques of multivariate statistics for analyzing compositional data, but one should be aware of the problems this generates since the most common methods rely on the Euclidean geometry on the real space. Pawlowsky-Glahn et al. (2015) explained that this framework is inappropriate for positive data vectors which components include fractions of a whole. As it will be derived later, the sample space of compositions has its own unique structure.

One of the problems that arises is the so-called *negative bias problem*. If we start with our initial condition, the constant sum, i.e. $x_1 + x_2 + \ldots + x_D = 1$ for some $D > 0$, we

would get

$$\operatorname{cov}(x_1, x_1 + \ldots + x_D) = 0$$

which in turn would result in

$$\begin{aligned} \operatorname{var}(x_1) + \operatorname{cov}(x_1, x_2) + \ldots + \operatorname{cov}(x_1, x_D) = 0 \quad &\Leftrightarrow \\ \operatorname{cov}(x_1, x_2) + \ldots + \operatorname{cov}(x_1, x_D) = -\operatorname{var}(x_1). \end{aligned}$$

Except for the trivial case, in which the first variable has to be a constant, the right hand side of the above equation is negative. Therefore, at least one term on the left hand side must be negative as well. Replacing x_1 by the other components in the first part of the covariance would result in a similar outcome. This means that in all rows of the covariance matrix, there is at least one negative entry, restricting the correlations and not to being free to range between -1 and 1.

2.4 Aitchison geometry

Before a solution for problems like the one above is introduced, definitions and the mentioned unique design of compositional data should be discussed. The following chapter builds on the work of Aitchison (2005) and Filzmoser et al. (2018).

A compositional vector, or just a composition, is a D-variate column vector $x = (x_1, \ldots, x_D)'$ with strictly positive entries. The components of such a vector, which will also be called *parts*, contain relative information of some whole.

One of the most important principles is the one of *scale invariance*. This means that the information of a composition shall not depend on a specific unit or on its size. If we look back at the expenditure example and focus on the first and second observation, we already saw that even though their values differ, the relative information they convey in the form of ratios of their entries is the same. In other words, we could argue that the multiplication of a compositional vector with some $p > 0$ should not change the ratios. For the two considered observations, let us refer to them as x_1 and x_2, this relationship would hold for $p = 19/6$, i.e. $x_1 = \frac{19}{6} x_2$.

This leads to the next definition, which is called *compositionally equivalent*, and applies to two observations if they only differ by a scaling factor. In this sense, all three observations from the initial problem are compositionally equivalent.

Permutation invariance describes that a permutation of the parts of a composition does not change the information content, like in multivariate statistics.

If we consider only some parts of a compositional vector, i.e. a subvector with $d < D$ components, then we speak of a *subcomposition*. With that we can define the next principle, which is referred to as *subcompositional coherence*. For this to be fulfilled, information coming from a subcomposition should not be contradictory to that from the original composition. This can be split into *subcompositional dominance* and *ratio*

preserving. The first one means that any distances between two compositions must at least be as big as the distance between the corresponding subcompositions. The latter one states that ratios between two variables do not depend on other ones. This implies that also for a subcomposition the scale invariance holds.

As stated before, the Euclidean space is not compatible with raw compositional data. In order to make interpretations and analysis of compositions possible, a similar structure has to be introduced. One of the early definitions of compositional data implied the $(D-1)$-standard-simplex

$$\mathcal{S}^D := \{x \in \mathbb{R}^D | x_1 + x_2 + \ldots + x_D = 1; x_i \geq 0 \; \forall \; i \in \{1, \ldots, D\}\},$$

a subspace of $\mathbb{R}^D$, as the sample space of compositions. In three dimensions this describes a triangle spanned by the vertices $e_1 = (1, 0, 0)', e_2 = (0, 0, 1)$ and $e_3 = (0, 1, 0)'$, see Figure 1 below.

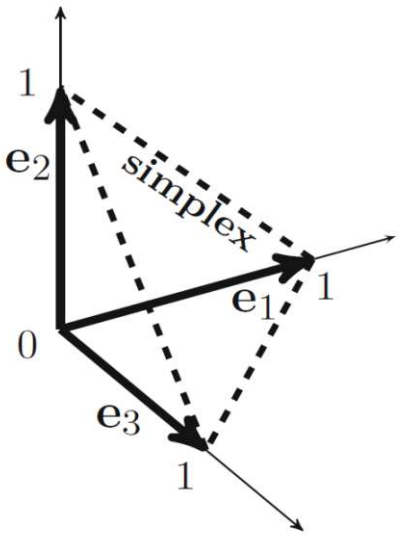

Figure 1: The 2-standard-simplex in $\mathbb{R}^3$. Based on Filzmoser et al. (2018)

In higher dimensions the simplex is a tetrahedron induced by $e_i = (0, \ldots, 0, 1, 0, \ldots, 0)'$, $i \in \{1, \ldots, D\}$ with the 1 at the i-th component. To keep it more general, since the parts of compositions do not always have to sum up to one, the D-part-simplex

$$\mathcal{S}^D := \{x \in \mathbb{R}^D_+ | x_1 + x_2 + \ldots + x_D = c\}$$

is defined. This space includes all observations which sum up to a constant c and have strictly positive entries. Still, this is a little bit too restrictive because not all compositional observations do necessarily have to sum up to the same constant.

For this the so-called *closure-operator* $\mathcal{C}_\kappa$ is introduced. Given a composition x and a parameter κ, the closure-operator is defined as

$$\mathcal{C}_\kappa(x) := \left(\frac{\kappa x_1}{\sum_{i=1}^D x_i}, \ldots, \frac{\kappa x_D}{\sum_{i=1}^D x_i} \right)'$$

which is a standardization of the input x to have a component sum of κ. This leads us back to the definition of compositional equivalence, which now can be translated into two compositions x and y being compositionally equivalent if there exist two parameters κ and ν such that

$$\mathcal{C}_\kappa(x) = \mathcal{C}_\nu(y)$$

holds. In other words, all observations which are compositionally equivalent lay on the same ray through the origin. The corresponding projections onto the simplex have a sum of one and can be interpreted as the proper representation of this equivalence class.

With the closure-operator a new definition of the D-part-simplex arises,

$$\tilde{\mathcal{S}}^D := \{x \in \mathbb{R}_+^D | \forall\ \kappa > 0\ \exists^*\ \lambda > 0 : x_1 + \ldots + x_D = \lambda \mathcal{C}_\kappa(x)\}. \tag{1}$$

This can be understood as the collection of all complete rays from the origin which observations only have strictly positive entries. Since the exact value of κ does not matter, this last definition of $\tilde{\mathcal{S}}^D$ in (1) can be seen as a decomposition of $\mathbb{R}^D$ into equivalence classes of observations. Two of these possible equivalence classes are depicted down below in Figure 2 for $D = 3$. The observations x_1, x and x_2 as well as y_1, y and y_2 are compositionally equivalent, respectively, since they share the same ray. The corresponding projections onto the simplex, x and y, are marked by the dashed lines.

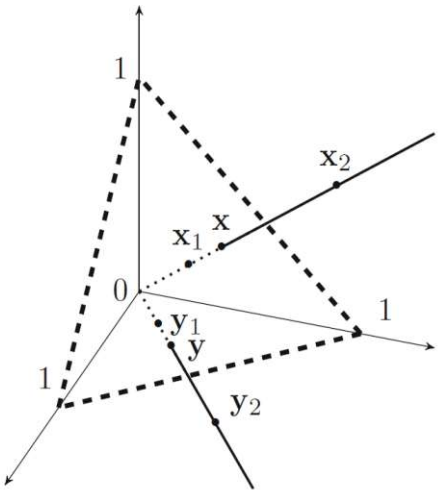

Figure 2: 3-part compositions in $\mathbb{R}^3$ projected onto the standard simplex. Based on Filzmoser et al. (2018)

Given $\tilde{\mathcal{S}}^D$ as sample space, Pawlowsky-Glahn and Egozcue (2001) defined a structure similar to the Euclidean vector space, the so-called *Aitchison geometry*. For this, they used the already in Aitchison (1986) established shifting and multiplication operation.

When speaking of shifting or differences in the context of compositional data, Filzmoser et al. (2018) mentioned an example considering the absolute values of votes of a political party. Lets consider two different villages where the number of votes from one year to the next changes from 300 to 200 and 3000 to 2900 respectively. In this situation one would rather talk about a loss of votes of 10% and $3.\bar{3}$% rather than of a loss of 100 votes. This concept is called the principle of *relative scale*, which states that when talking of the dissimilarity of compositions, the ratio of the components should be used rather than their differences.

This short example should motivate the definition of *perturbation*. Considering two compositions x and y from $\tilde{\mathcal{S}}^D$, the perturbation operator $\oplus$ is defined as

$$x \oplus y := (x_1 y_1, \ldots, x_D y_D)'.$$

Next to that, the *powering operator* $\odot$ was introduced. For a scalar $\alpha \in \mathbb{R}$ and a composition $x \in \tilde{\mathcal{S}}^D$ it is defined as

$$\alpha \odot x := (x_1^\alpha, \ldots, x_D^\alpha)'.$$

Applying both of these concepts can be used to define the *perturbation difference* between two compositions, i.e.

$$x \ominus y := x \oplus (-1 \odot y) = (x_1/y_1, \ldots, x_D/y_D)'.$$

Therefore, the difference between a composition and itself results in the neutral element, i.e.

$$x \ominus x = (x_1/x_1, \ldots, x_D/x_D)' = (1, \ldots, 1)'.$$

As Pawlowsky-Glahn and Egozcue (2001) showed, these definitions of perturbation and powering are sufficient to induce a vector space on the simplex, which from this point on was called the Aitchison geometry.

To obtain a $(D-1)$-dimensional Hilbert space, they furthermore proposed the use of an inner product, i.e.

$$\langle x, y \rangle_a := \frac{1}{2D} \sum_{i,j=1}^{D} \log\left(\frac{x_i}{x_j}\right) \log\left(\frac{y_i}{y_j}\right), \tag{2}$$

a vector norm, i.e.

$$||x||_a := \sqrt{\langle x, x \rangle_a} = \sqrt{\frac{1}{2D} \sum_{i,j=1}^{D} \log\left(\frac{x_i}{x_j}\right)^2}, \tag{3}$$

and a distance, i.e.

$$d(x,y)_a := \sqrt{\frac{1}{2D}\sum_{i,j=1}^{D}\left(\log\left(\frac{x_i}{x_j}\right)-\log\left(\frac{y_i}{y_j}\right)\right)}. \tag{4}$$

These operators are in general referred to as the *Aitchison inner product, norm* and *difference*.

2.5 Logratios in compositional data analysis

Logratios, short for the logarithm of the ratio between two variables, as the ones above in (2), (3) and (4), play an important role for compositional data analysis. As mentioned before, these ratios are the source of relative information. The direct usage of the raw ratios might not be the best option because of their asymmetry. The interval $(0,1)$ would correspond to the variable in the denominator being dominant, a ratio of 1 stands for perfect balance and $(1,\infty)$ would imply that the numerator is larger. It seems reasonable to extend the ratios by their logarithms in order to symmetrize these intervals, as that would result in a range from $-\infty$ to ∞ with a value of 0 representing a balance between two parts. These logratios have several advantages from a mathematical point of view. They can be expressed as difference of the respective logarithms, i.e. $\log(x_i/x_j) = \log(x_i) - \log(x_j)$. This means that the Aitchison distance can be rewritten into

$$\begin{aligned} d(x,y)_a &= \sqrt{\frac{1}{2D}\sum_{i,j=1}^{D}\left(\log\left(\frac{x_i}{x_j}\right)-\log\left(\frac{y_i}{y_j}\right)\right)} \\ &= \sqrt{\frac{1}{2D}\sum_{i,j=1}^{D}\left(\log(x_i)-\log(y_i)-(\log(x_j)-\log(y_j))\right)} \\ &= \sqrt{\frac{1}{2D}\sum_{i,j=1}^{D}\left(\log\left(\frac{x_i}{y_i}\right)-\log\left(\frac{x_j}{y_j}\right)\right)} \end{aligned}$$

which backs up the concept of relative scale.

Using logratios also directly supports the property that the sum is irrelevant for compositional data. If we compare the values of the above defined operators for a composition

x and its scaled counterpart λx for some $\lambda > 0$ we see

$$\begin{aligned}\langle x, y\rangle_a = \frac{1}{2D}\sum_{i,j=1}^{D} \log\left(\frac{x_i}{x_j}\right)\log\left(\frac{y_i}{y_j}\right) = \frac{1}{2D}\sum_{i,j=1}^{D} \log\left(\frac{\lambda x_i}{\lambda x_j}\right)\log\left(\frac{y_i}{y_j}\right)\\ = \langle \lambda x, y\rangle_a,\\ ||x||_a = ||\lambda x||_a,\\ d(x,y)_a = d(\lambda x, y)_a.\end{aligned}$$

Similar to these metric concepts for perturbation and powering we get

$$\begin{aligned}(\lambda x) \oplus y = \lambda(x \oplus y) \text{ and}\\ \alpha \odot (\lambda x) = \lambda^{\alpha}(\alpha \odot x).\end{aligned}$$

Moreover, logratios and their inverse just differ in sign, i.e. $\log(x_i/x_j) = -\log(x_j/x_i)$. This also leads to a relationship of the variance between them, i.e. $\text{var}\left(\log(x_i/x_j)\right) = \text{var}\left(\log(x_j/x_i)\right)$, which could not have been established for just the raw ratios.

Another feature are the one-to-one transformations between compositions and a full set of logratios, for example

$$(y_1, \ldots, y_{D-1})' = \left(\log\left(\frac{x_1}{x_D}\right), \ldots, \log\left(\frac{x_{D-1}}{x_D}\right)\right)' \tag{5}$$

with its scaled inverse

$$(x_1, \ldots, x_D)' = \frac{1}{\exp(y_1) + \ldots + \exp(y_{D-1}) + 1}\left(\exp(y_1), \ldots, \exp(y_{D-1}), 1\right)'.$$

Such transformation will be used to transfer the vectors of logratios from their restricted sample space to the whole unrestricted real space where standard methods of multivariate statistics can be applied.

One has to keep in mind that zeros in one of the components could lead to problems. That is one of the reasons why $\tilde{\mathcal{S}}^D$ only contains strictly positive observations. This might seem to be a big restriction, but the arising disadvantages when dealing with real world data are compensated by many convenient properties of logratios. Filzmoser et al. (2018) devote a whole chapter onto the topic of how to deal with zero values.

Typically the natural logarithm ln is used in the context of compositional data analysis. Since the use of another basis just refers to rescaling, it is a matter of taste which specific logarithm is applied.

2.6 The family of logratio transformations

There are many ways in order to transform the compositional observations from the simplex onto the whole real space. One possibility was already stated in (5). That

representation reduced the dimensionality from the D parts to $D-1$ logratios. In general, for a D-dimensional composition there are $D(D-1)$ possible logratios. Since a logratio and its reciprocal just differ by their sign, the number of distinct ratios reduces to $D(D-1)/2$. It will always be possible to find $D-1$ logratios and express the remaining ones through the identity

$$\log\left(\frac{x_i}{x_j}\right) = \log\left(\frac{x_i}{x_k}\right) + \log\left(\frac{x_k}{x_j}\right),$$

see Filzmoser et al. (2018), i.e. the D-part compositions can be expressed by coordinates in a $D-1$ dimensional subspace of $\mathbb{R}^D$. The results of these transformations can be interpreted as coordinates with respect to the Aitchison geometry. This view also helps in understanding and explaining those.

2.6.1 Additive Logratio Coordinates

The mapping mentioned above is named *additive logratio coordinates*, or alr coordinates for short. It transfers an observation $x \in \tilde{\mathcal{S}}^D$ to $\mathbb{R}^{D-1}$ and results in coordinates $x^{(j)}$, i.e.

$$\begin{aligned} x^{(j)} := alr(x) &= \left(x_1^{(j)}, \ldots, x_{D-1}^{(j)}\right)' \\ &= \left(\log\left(\frac{x_1}{x_j}\right), \ldots, \log\left(\frac{x_{j-1}}{x_j}\right), \log\left(\frac{x_{j+1}}{x_j}\right), \ldots, \log\left(\frac{x_D}{x_j}\right)\right)'. \end{aligned}$$

To get the original data from existing alr coordinates, the back-transformation

$$\begin{aligned} x_i &= \exp\left(x_i^{(j)}\right)' \quad \text{for} \quad i \in \{1, \ldots, D\},\ i \neq j \\ x_j &= 1 \quad \text{for } j \in \{1, \ldots, D\} \end{aligned}$$

has to be applied. It has to be noted that the resulting parts do not necessarily have to sum up to the same value as the original parts or to 1. Since the value of the sum does not matter, scaling can be omitted.

Depending on the situation one might choose different values for j during the analysis. In general, the choice of the variable in the denominator does not matter too much (Aitchison, 1986), and in geochemistry there are some guidances for selecting an appropriate "reference" variable.

Another characteristic is that

$$\begin{aligned} \mathrm{alr}_j(x \oplus y) &= \mathrm{alr}_j(x) + \mathrm{alr}_j(y) \quad \text{and} \\ \mathrm{alr}_j(c \odot x) &= c \odot \mathrm{alr}_j(x) \end{aligned} \tag{6}$$

holds. Principally, this has not to be fulfilled for the Aitchison inner product and norm.

2.6.2 Centered Logratio Coefficients

A transformation which achieves also these features expresses a composition $x \in \tilde{\mathcal{S}}^D$ by a vector $y \in \mathbb{R}^D$, i.e.

$$y := \text{clr}(x) = \left(\log\left(\frac{x_1}{g(x)}\right), \ldots, \log\left(\frac{x_D}{g(x)}\right) \right)'$$

with the geometric mean $g(x) = \left(\prod_{i=1}^D x_i\right)^{\frac{1}{D}}$. These coordinates can be viewed as centered logarithms of the original data. That is why they are called *centered logratio coordinates* or *coefficients*, commonly just referred to as clr coefficients. The corresponding inverse mapping, up to a scaling factor, is

$$x_j = \exp(y_j) \quad \text{for } j \in \{1, \ldots, D\}.$$

The connection between the clr coefficients and the log transformation can be shown as follows, Aitchison (1986). Start by defining

$$W := \mathbf{I}_D - \frac{1}{D}\mathbf{1}_D\mathbf{1}_D'.$$

Then it holds that

$$y = \text{clr}(x) = W\log(x).$$

While the linearity was already fulfilled by the alr coordinates, clr coefficients also satisfy

$$\begin{aligned} \langle x, y\rangle_a &= \langle \text{clr}(x), \text{clr}(y)\rangle_2 \quad \Rightarrow ||x||_A = ||\text{clr}(x)||_2 \quad \text{and} \\ d(x, y)_a &= d\left(\text{clr}(x), \text{clr}(y)\right)_2. \end{aligned} \tag{7}$$

Operations on the right hand side of the above equations refer to Euclidean inner product, norm and distance, respectively. These two properties show that the clr coefficients represent an isometry between $\tilde{\mathcal{S}}^D$ and $\mathbb{R}^D$.

The coordinates build a generating system on a $D-1$ dimensional subspace of $\mathbb{R}^D$ which can be demonstrated by

$$\begin{aligned} \sum_{j=1}^D y_j &= \sum_{j=1}^D \log\left(\frac{x_j}{\left(\prod_{i=1}^D x_i\right)^{\frac{1}{D}}}\right) = \sum_{j=1}^D \left(\log(x_j) - \frac{1}{D}\sum_{i=1}^D \log(x_i)\right) \\ &= \sum_{j=1}^D \log(x_j) - \sum_{i=1}^D \log(x_i) = 0. \end{aligned}$$

This implies that for data analysis, one coordinate cannot be considered on its own without the others being taken into account.

2.6.3 Isometric Logratio Coordinates

The next coordinates aim to establish an orthonormal basis in this hyperplane and try to overcome this restriction. The representation of the clr coefficients is not resulting in full rank, since one always get one more coordinate than necessary for the $D-1$ dimensional Aitchison geometry. Due to that, there is also not an unique way to define an orthonormal basis, which makes the *isometric logratio coordinates* a class of coordinates. Based on their structure it is also common to refer to them as *orthonormal* coordinates. Fišerová and Hron (2011) chose the basis to have the following structure

$$z := \operatorname{ilr}(x) \quad \text{whereas}$$
$$z_j = \sqrt{\frac{D-j}{D-j+1}} \log \left(\frac{x_j}{\left(\prod_{k=j+1}^{D} x_k\right)^{\frac{1}{D-j}}} \right) \quad \text{for } j \in \{1, \dots, D-1\}. \tag{8}$$

To guarantee orthonormality of the resulting coordinates, the above scaling factors in front of the logarithms are selected.

Just as the clr coefficients, the ilr coordinates also supply a one-to-one mapping between the simplex and $\mathbb{R}^{D-1}$. The original parts can be found, up to a scaling factor, by

$$x_1 = \exp\left(\frac{\sqrt{D-1}}{\sqrt{D}} z_1 \right),$$
$$x_j = \exp\left(-\sum_{k=1}^{j-1} \frac{1}{\sqrt{(D-k+1)(D-k)}} z_k + \frac{\sqrt{D-j}}{\sqrt{D-j+1}} z_j \right), j \in \{2, \dots, D-1\},$$
$$x_D = \exp\left(-\sum_{k=1}^{D-1} \frac{1}{\sqrt{(D-k+1)(D-k)}} z_k \right).$$

A direct link between the clr and pivot coordinates can be achieved by the basis vectors of the spanned hyperplane. These orthonormal basis vectors are

$$v_{\cdot j} = \sqrt{\frac{D-j}{D-j+1}} \left(0, \dots, 0, 1, -\frac{1}{D-j}, \dots, -\frac{1}{D-j} \right)' \tag{9}$$

for $j \in \{1, \dots, D-1\}$, whereas each vector has $j-1$ zero entries. Collected into a $D \times (D-1)$ dimensional matrix V, the relationship between the two transformations is

$$y = Vz \quad \text{and} \quad z = V'y, \tag{10}$$

see Egozcue et al. (2003).

As well as the clr coefficients the ilr transformation also describes a isometry. Therefore, analogous results as in (6) and (7) hold.

Interpretation of compositional data is achieved by looking at the resulting logratio transformations. For ilr coordinates we can observe that the first component x_1 only appears in the first coordinate z_1. In contrary to this situation, for the alr and clr transformation one could not interpret the coordinates in terms of the parts of the composition simultaneously. Here all the relative information about one part is devoted into one coefficient, i.e.

$$\begin{aligned} z_1 &= \sqrt{\frac{D-1}{D}} \log \left(\frac{x_1}{\left(\prod_{k=2}^{D} x_k \right)^{\frac{1}{D-1}}} \right) \\ &= \sqrt{\frac{1}{D(D-1)}} \left(\log \left(\frac{x_1}{x_2} \right) + \ldots + \log \left(\frac{x_1}{x_D} \right) \right). \end{aligned}$$

This can be interpreted as the relative *dominance* of x_1 with respect to the other parts on average. Since no other component can be viewed like that, a permutation of the compositional parts makes interpretation of other parts possible. Therefore, ilr coordinates are also called *pivot* coordinates.

So, if a specific interpretation of part l of the composition is desired, one can consider the permuted composition

$$x^{(l)} = (x_l, x_1, \ldots, x_{l-1}, x_{l+1}, \ldots, x_D)' =: \left(x_1^{(l)}, x_2^{(l)}, \ldots, x_l^{(l)}, x_{l+1}^{(l)}, \ldots, x_D^{(l)} \right)'$$

and the corresponding *generalized* pivot coordinates

$$\begin{aligned} z_j^{(l)} &= \sqrt{\frac{D-j}{D-j+1}} \log \left(\frac{x_j^{(l)}}{\left(\prod_{k=j+1}^{D} x_k^{(l)} \right)^{\frac{1}{D-j}}} \right) \quad \text{and} \\ z^{(l)} &= \left(z_1^{(l)}, \ldots, z_{D-1}^{(l)} \right)'. \end{aligned}$$

The first coordinate of $z^{(l)}$ and the l-th clr coefficient y_l are proportional up to a scaling factor

$$z_1^{(l)} = \sqrt{\frac{D}{D-1}} y_l. \tag{11}$$

For multivariate data analysis, the ilr coordinates should be used as they have the necessary properties so that interpretations of their transformed parts are meaningful (Filzmoser et al., 2009b).

2.7 Descriptive statistics of compositional data

Standard descriptive statistics like the arithmetic mean or the variance can be applied onto the transformed coordinates like the clr coefficients. Nevertheless, these observatory tools are limited to the logratio coordinates, since only those "live" in the Euclidean space. Directly applied to the compositions in the Aitchison geometry rather leads to problems.

Let us consider two observations $x_1 = (1, 2)'$ and $x_2 = (3, 2)'$. The arithmetic mean of these observations is $\bar{x} = \frac{x_1+x_2}{2} = (2, 2)'$. Due to the scale invariance one would expect the scaled inputs to preserve the proportional information, i.e. $\frac{\bar{x}_1}{\bar{x}_2} = 1$. But for $\mathcal{C}_1(x_1) = \left(\frac{1}{3}, \frac{2}{3}\right)'$ and $\mathcal{C}_1(x_2) = \left(\frac{3}{5}, \frac{2}{5}\right)'$ the mean result in $\bar{x}^* = \left(\frac{7}{15}, \frac{8}{15}\right)'$ with a component ratio of $\frac{7}{8} = 0.875$.

As we saw, the complication comes from the property of scale invariance, which the arithmetic mean fails to maintain. Therefore, when it comes to descriptive analysis of compositional parts, alternative approaches have to be used.

In the context of the mean with respect to the Aitchison geometry, Pawlowsky-Glahn and Egozcue (2002) suggest the application of the component-wise geometric mean, also called *center*. For an $n \times D$ matrix of compositions X it is defined as

$$g_x = (g_1, \ldots, g_D)'$$

with $g_j = \left(\prod_{i=1}^{n} x_{ij}\right)^{\frac{1}{n}}$ for $j = 1, \ldots, D$. The center follows the principle of compositional data, including the mentioned scale invariance. For the above considered observations this can be shown by

$$g_x = \left(\sqrt{1 \cdot 3}, \sqrt{2 \cdot 2}\right)' = \left(\sqrt{3}, \sqrt{4}\right)' \quad \text{and}$$

$$\mathcal{C}_1\left(g_x\right) = \left(\sqrt{\frac{1}{3} \cdot \frac{3}{5}}, \sqrt{\frac{2}{3} \cdot \frac{2}{5}}\right)' = \left(\sqrt{\frac{3}{15}}, \sqrt{\frac{4}{15}}\right)'$$

whereas both proportions result in $\sqrt{3}/\sqrt{4} = \frac{\sqrt{3}}{\sqrt{15}} / \frac{\sqrt{4}}{\sqrt{15}}$.

Centering is carried out directly onto the original compositions. For one observation of X, i.e. $x_{i.} = (x_{i1}, \ldots, x_{iD})'$, the centered composition is

$$x_{i.}^c = x_{i.} \ominus g_{x_{i.}} = \left(\frac{x_{i1}}{g_{x_{i.}}}, \ldots, \frac{x_{iD}}{g_{x_{i.}}}\right)'.$$

Due to the properties of the coordinate systems discussed in the last subsection, this corresponds to mean-centering of the logratio coordinates.

The counterpart for the variance cannot be constructed directly for the compositions. The focus is rather laid on source of information, i.e. the pairwise logratios. Introduced in Aitchison (1986), the so-called *variation matrix* is formed by the variances of all pairwise logratios.

Given a data matrix $X \in \mathbb{R}^{n \times D}$, the variation matrix is

$$T = \begin{pmatrix} t_{11} & \dots & t_{1D} \\ \vdots & \ddots & \vdots \\ t_{D1} & \dots & t_{DD} \end{pmatrix}$$

whereas each element t_{jl} for $j, l = 1, \dots, D$ is the sample variance of the pairwise logratios between part j and l. It is defined by

$$t_{jl} = \frac{1}{n-1} \sum_{i=1}^{D} \left(z_{jl}^i - \bar{z}_{jl} \right)^2$$

with

$$z_{jl}^i = \log \left(\frac{x_{ij}}{x_{il}} \right) \quad \text{for} \quad i \in \{1, \dots, D\}$$

and

$$\bar{z}_{jl} = \frac{1}{n} \sum_{i=1}^{n} z_{jl}^i.$$

The matrix T is symmetric by construction, with zeros on the diagonal. Its entries t_{jl} can be interpreted as the variation between two parts. For values of t_{jl} close to zero, the ratio between the corresponding parts x_j and x_l is nearly constant, implying an almost perfect proportionality.

One downside of this approach is that interpretability in the sense of positivity and negativity, derived from the correlation coefficient is lost. Also, because of the non-linearity of the logarithm, it is not clear which values of t_{jl} correspond to a high or a low correlation.

Finally, the *total variance* of a compositional data set is defined as the scaled sum of all elements of the variation matrix, i.e.

$$\text{totvar}(X) = \frac{1}{2D} \sum_{j,l=1}^{D} t_{jl}. \tag{12}$$

3 Principal component analysis

Principal component analysis, or PCA for short, is an important statistical tool for multivariate data analysis. Often the data set which has to be investigated consists of many variables, each of them having a different level of contribution to the analysis. The goal of PCA is to reduce the dimensionality of the data matrix by constructing new latent variables, called principal components, which are orthonormal. These components are linear combinations of the initial data, oriented in a way to contain the most relevant variation. In general, the dimensionality of the principal components and the original data is the same. Typically only the first few contain already enough information.

The goal of this section is not to introduce PCA as a new statistical method, but rather to extend this tool to the context of compositional data. The compositions will be represented by ilr coordinates, although clr coefficients are commonly applied during PCA. This has methodological reasons.

3.1 Estimation by singular value decomposition

One of the many possible ways to estimate the principal components is by using singular value decomposition or *SVD*.

The singular value decomposition of a real $n \times D$ matrix X is defined as a factorization of the form

$$X = UDW', \tag{13}$$

whereas

- U is an $n \times p$ orthogonal matrix containing the left singular vectors,
- D is a non-negative matrix containing the singular values $d_1, \ldots, d_p$ and
- W is a $D \times p$ orthogonal matrix containing the right singular vectors of X,

where $p = \min\{n, D\}$ refers to the rank of X. This factorization is not unique. Therefore it is always possible to rearrange the columns of U and W with respect to decreasing singular values $d_1 \geq d_2 \geq \ldots \geq d_p$.

Let X be the matrix containing the compositional observations. Since PCA is not scale invariant and we want to obtain directions (components) containing the most variance,

mean-centering is essential. Denote the centered ilr coordinates of X by Z. Choosing a factorization with descending singular values, the SVD of Z can be rearranged into

$$Z = (UD)W' = Z^*W' \tag{14}$$

which marks the PCA transformation. The matrix Z^* represents the original data mapped into an orthogonally rotated coordinate system. These coordinates z^*_{ij} are termed *scores*. Due to the orthonormal equivariance of PCA, the initial choice of orthogonal coordinates does not influence the resulting components. The variances of the columns of Z^* correspond to those from the principal components. These variances λ_i are proportional to the singular values d_i, for $i \in \{1, \ldots, p\}$,

$$\lambda_i = \frac{d_i^2}{n-1}.$$

The sum of the variances of all PCs is equal to the total variability of the data matrix X (12), i.e.

$$\sum_{i=1}^{p} \lambda_i = \text{totvar}(X).$$

Thus, the proportion of variance explained by the i-th principal component is

$$\frac{\lambda_i}{\sum_{i=1}^{p} \lambda_i}.$$

Since the goal of PCA is dimension reduction, one question to ask is how many of the resulting components to use for further analysis. The answer might depend on the purpose of the components. If it just would be for a visual inspection, a "rule of thumb" can be sufficient. For this rule, the proportion of explained variance using the first k principal components on the total variance is of interest, i.e.

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{p} \lambda_i} \geq \alpha.$$

The cut-off value α can be set to 80% or to 90%, for example.

If the user thinks that the last variables just contain some irrelevant noise, the last approach might not be appropriate. For such a situation, a *scree plot* would mark another possibility. It plots the explained variance of each component against its number. All variables where the proportion approximately follows a linear trend can be excluded and the remaining are considered relevant.

In (14) the matrix W is called the *loadings* matrix with the *loadings* in its columns. There are many ways how to interpret them. From a geometrical perspective, they are the basis vectors of the principal components. From a practical point of view, the

entries w_{ij} are weights which reflect the influence of the original ilr coordinates on the new orthonormal compositions. Using definition (8), the first element of each loading vector can be seen as the effect of the relative dominance of the first part x_1 to the respective principal component. By using pivot coordinates before carrying out the PC transformation, the relative dominance of other parts can be interpreted in the same way.

The above transformation using SVD can be reformulated based on a different objective function. If one has chosen k, the relevant number of PCs, the data matrix Z can be approximated by

$$\tilde{Z} = Z_k^* W_k'$$

which is $n \times (D-1)$ dimensional. The matrices Z_k^* and W_k' only contain the first k columns of Z^* and W, respectively. This approximation minimizes the Frobenius norm between itself and the original data matrix Z, i.e.

$$||Z - \tilde{Z}||_F = \left(\sum_{i=1}^{n} \sum_{j=1}^{D-1} (z_{ij} - \tilde{z}_{ij})^2 \right)^{1/2}.$$

For finding the directions of the principal components one thus minimzes residual sum-of-squares, with residuals formed by the matrix $Z - \tilde{Z}$.

As already described above, single compositions can be interpreted by using pivot coordinates. However, due to computational effort in the case of high dimensional compositions, it is not advised to calculate D different PC transformations. That is where some properties of the clr coefficients come in handy. Because of the connection between the coefficient y_l and the first pivot coordinate $z_1^{(l)}$, for $l \in \{1, \ldots, D\}$, see (11). With that one can show that the l-th row of the loading matrix W_y resulting from the clr data matrix Y only differs by a constant from the loading matrix of the pivot coordinates $W^{(l)}$, Kynčlová et al. (2016). Also the scores in Y^* and Z^* corresponding to the non-zero singular values are the same, Filzmoser et al. (2018). These properties imply that it suffices to perform PCA in clr coefficients as the loadings and scores can be derived afterwards. Also the relative importance of each original part can be acquired directly from the loading matrix.

3.2 Estimation by the covariance matrix

Since we want to minimize the information loss in PCA, or in other words, we want to maximize the variance of the transformed data, another method for finding the components would be to look at the covariance matrix of the data. Again, let us assume Z being the mean-centered data matrix of compositions with sample covariance matrix

$$S_z = \frac{1}{n-1} Z'Z.$$

With the SVD of Z (13) this results in

$$S_z = \frac{1}{n-1} WDU'UDW' = \frac{1}{n-1} WD^2W'$$

using the property $U'U = I_D$. Without loss of generality, we assume that the singular values in the diagonal of D are of descending order as before. If we denote $w_{.l}$ by the l-th column of W we get

$$\begin{aligned} S_z w_{.l} &= \frac{1}{n-1} WD^2W'w_{.l} \\ &= \frac{1}{n-1} WD^2 e_l \\ &= \frac{d_{ll}^2}{n-1} We_l = \frac{d_{ll}^2}{n-1} w_{.l}, \end{aligned}$$

i.e. the l-th column of W is just an eigenvector of S_z with respect to the eigenvalue $\frac{d_{ii}^2}{n-1}$. The vector e_l is the l-th unit vector.

It turns out that if we want to find a linear combination of the original data Z under the conditions of variance maximization and orthonormality of the resulting components, the transformation matrix must consist of the eigenvectors of the covariance matrix S_z, i.e. must be equal to W. The corresponding eigenvalues coincide with the variances of the PCs derived in Section 3.1. We conclude that SVD applied on centered ilr coordinates leads to the same result as an eigenvalue decomposition on the covariance matrix of those coordinates. The loading matrix W of both approaches is the same and also the scores are equal, i.e.

$$ZW = UDW'W = UD = Z^*.$$

One advantage of the last attempt is the possibility to incorporate a robust estimate of the covariance matrix S_z. A choice is the so-called MCD, short for *Minimum Covariance Determinant*, estimator. It robustly estimates location and scale of a given data set by calculating the mean and covariance only for the part of the data which minimizes the determinant of the empirical covariance matrix of that proportion. It has a high breakdown point and is affine equivariant. The latter means that the initial choice of ilr coordinates does not alter the resulting estimates of location t_{MCD} and scale C_{MCD}.

Following Filzmoser et al. (2009a), the procedure for robust PCA in the case of $n \geq D-1$ starts with the SVD of C_{MCD}, i.e.

$$C_{MCD} = W_{MCD} D_{MCD} W'_{MCD},$$

with W_{MCD} the robust loading matrix and D_{MCD} containing the singular values with are the robust variances of the resulting principal components. The data matrix then is centered by t_{MCD} resulting in Z_{MCD} and scores are calculated by

$$Z^*_{MCD} = Z_{MCD} W_{MCD}.$$

These scores are robust against outliers, Croux and Haesbroeck (2000). Because of their large variance, these deviations might "attract" the direction of the principal components within a non-robust approach. Since outlier detection is not the objective of PCA, using this robust method should be preferred if one assumes the presence of anomalies in the data set.

One has to keep in mind that the MCD estimator can not be calculated for clr coefficients because the covariance matrix does not have full rank. This time, it suffices to conduct the robust decomposition on the ilr coordinates and determine scores and loadings for the clr coefficients afterwards. Again, scores corresponding to non-zero singular values are the same, i.e $Y^*_{MCD} = Z^*_{MCD}$. To get the loadings, the ilr loading matrix has to be transformed according to (10), i.e.

$$W_{y,MCD} = W_{MCD}V',$$

with V the orthonormal basis of the pivot coordinates described in (9).

3.3 Compositional biplots

Biplots, introduced in the 70s by Gabriel (1971), are a statistical tool to display observations and variables in a two-dimensional plot together. One area of application of biplots is PCA, since the general assumption is that the data approximately has rank two, i.e. the first few components already include the majority of information. If the rank of the data is higher than two, the first two principal components should explain the data variability sufficiently well.

The idea behind a biplot is to distribute the information of the data between two matrices representing the variables and observations, respectively. Let Z be the mean-centered data matrix of ilr coordinates. The best rank two approximation of Z is obtained by taking the first two singular values d_1 and d_2 based on the SVD in (13). Furthermore, the first two columns of U and W are being used, resulting in

$$Z_2 = (u_1 u_2) \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix} \begin{pmatrix} w_1' \\ w_2' \end{pmatrix}.$$

This can be partitioned into

$$Z_2 = GH'$$

with

$$G = (u_1 u_2) \begin{pmatrix} d_1 & 0 \\ 0 & d_2 \end{pmatrix}^{1-c}$$

and

$$H = (w_1 w_2) \begin{pmatrix} d_1 & 0 \\ 0 & d_1 \end{pmatrix}^{c}.$$

The most common choice for c is $c = 1$, which leads to the so-called *covariance biplot*. The plot consists out of the rows of G and H. The rows of G represent the coordinates of the observations and are depicted by points in the plot. Analogously, the rows of H stand for the variables, plotted as the head of arrows which start at the origin.

Rescaling of G and H leads to more sophisticated interpretations. For a direct application in compositional data, coordinates resulting from sequential binary partitioning have to be used (Pawlowsky-Glahn et al., 2015).

If these are not available, biplots have to be adjusted because it is not possible to create them for the original observations. This can be achieved by a set of pivot coordinates. Due to the rotational invariance of SVD, only the matrix $W^{(l)}$ and consequently $H^{(l)}$ for $l \in \{1, \ldots, D\}$ leads to differences. Since it would not be practical to consider D biplots for interpretations, the arrows, i.e. the variables, are joined into one single plot. It is the default approach in compositional data and can be constructed directly in clr coordinates. It is referred to as *compositional biplot*.

Interpretations also differ from the biplots from multivariate statistics. Usually, the angle between the arrows of two variables approximates their correlation. For compositional biplots, the links between the vertices of variables are considered. These links are approximately the variance of the corresponding pairwise logratios of the original parts. The vertices of proportional parts coincide, or nearly so. If we compare two links, namely the cosine of their enclosed angle, we get an approximation of the correlation between the respective logratios.

4 Weighting in compositional data analysis

Weighting is a statistical tool to adjust different methods to specific situations. Rather than having all variables of a collected data set having the same contribution, some are assumed to be more relevant while others should have less to no importance to the statistician.

In the field of compositional data, weighting had been applied in many different situations. When dealing with a measurement device analyzing the components of soil samples, the accuracy might not be the same for all parts. Especially when variables fall below a certain detection limit further analyses could be compromised by these inaccuracies. A possible outcome would be to down-weight variables with small concentrations, as it was considered in Hron et al. (2022). Another situation occurs in Martín-Fernández et al. (2018), where they tried to raise the interpretability of the basis vectors through the help of so-called principal balances. In Greenacre (2019) the author draws a connection between variable selection of pairwise logratios and graph theory. The authors of Rieser and Filzmoser (2022) go one step further and extend the Aitchison geometry to a framework where only selected logratios are considered.

The latter work shall be the basis of the analysis in this theses. Based on distributional assumptions of the ilr coordinates, an optimization problem to estimate a weighting scheme is developed. For this, the Aitchison inner product (2) was chosen. To take the relevance of every term into account, each pair of logratios must receive its own weight. The Aitchison inner product therefore can be generalized to

$$\frac{1}{2}\sum_{i,j=1}^{D}\log\left(\frac{x_i}{x_j}\right)\log\left(\frac{y_i}{y_j}\right)w_{ij}.$$

If one would assume equal weights, i.e. $w_{ij} = 1/D$ for $i \neq j$, the above would coincide with the Aitchison inner product. Hence we also call these weights the *Aitchison weights* and W the *Aitchison weight matrix*.

Before deriving the optimization problem some results from graph theory have to be mentioned. Afterwards a connection between the latter one and compositional data analysis can be drawn.

4.1 Graph theory and compositional data

A *graph* G is an ordered pair $G = (\mathcal{V}, E)$. $\mathcal{V}$ is a set which elements which are called *vertices*, and it denotes a set of indices corresponding to nodes. E stands for the *edge* set of the graph and includes all edges between two vertices of $\mathcal{V}$ which are connected. The graph G contains no self-loops, i.e. an edge from one vertex to itself, it is also called a *simple undirected graph*.

For weighting purposes, the graph G has to be extended. A *weighted* graph G' is defined as an ordered tuple $G' = (\mathcal{V}, E, W)$. Here, W is a symmetric square matrix with a zero diagonal. Its entries correspond to weights between two vertices. In the context of compositional data, $\mathcal{V}$ is chosen to be $\mathcal{V} = \{1, \dots, D\}$ and $W \in \mathbb{R}^{D \times D}$.

A common tool for edge-weighted graphs is the so-called Laplacian matrix L. It is an easy way to define a graph and is calculated by

$$L_W := \operatorname{diag}\left(\sum_{j=1}^{D} w_{1j}, \sum_{j=1}^{D} w_{2j}, \dots, \sum_{j=1}^{D} w_{Dj}\right) - W. \tag{15}$$

The first part is a diagonal matrix of the row-sums of W. In other words, each entry of this diagonal matrix is the sum of weights between one vertex and its adjacent vertices.

In general, there are different types of graph Laplacians. The above definition is also referred to as *combinatorial graph Laplacian* and corresponds to a weighted graph with no self-loops, Egilmez et al. (2017).

Laplacians have a broad field of application. In spectral theory, properties of graphs are derived by investigating the characteristic polynomial, eigenvalues or eigenvectors of the corresponding graph Laplacian. In machine learning, they are used as kernels. In connection to graph signal processing, operations like filtering, sampling or transformations are developed for graphs associated with Laplacians.

Some results which will be used later are mentioned in Lemma (1) below, see Mohar (1991) for a proof.

Lemma 1. *Let W be a symmetric matrix with non-negative entries and a zero diagonal, then*

- *L_W is a symmetric positive semi-definite matrix*
- *the vector $\mathbf{1} := (1, \dots, 1)' \in \mathbb{R}^D$ is an eigenvector of L_W, i.e. $L_W \mathbf{1} = \mathbf{0}$.*

The definition of L_W is motivated by

$$\frac{1}{2} \sum_{i,j=1}^{D} (f_i - f_j)(d_i - d_j) w_{ij} = f' L_W d \tag{16}$$

for some vectors $f = (f_1, \ldots, f_D), d = (d_1, \ldots, d_D)' \in R^D$, see Merris (1994). This equation marks the link to compositional data. If we would use the Aitchison weights, the inner product can be rewritten into

$$\begin{aligned}\langle x, y\rangle_a &= \frac{1}{2}\sum_{i,j=1}^{D} \log\left(\frac{x_i}{x_j}\right)\log\left(\frac{y_i}{y_j}\right)\frac{1}{D} \\ &= \frac{1}{2}\sum_{i,j=1}^{D} \left(\log(x_i) - \log(x_j)\right)\left(\log(y_i) - \log(y_j)\right)\frac{1}{D} \\ &= \log(x)' L_A \log(y).\end{aligned}$$

Here, L_A denotes the Laplacian matrix resulting from the Aitchison weights, i.e. $L_A = (1 - \frac{1}{D})\mathbb{I}_D - (\mathbf{1}\mathbf{1}' - \mathbb{I}_D)\frac{1}{D} = \mathbb{I}_D - \mathbf{1}\mathbf{1}'\frac{1}{D}$.

From this point on, Rieser and Filzmoser (2022) extended operators like perturbation $\oplus$, powering $\odot$ and the inner product (2) to construct a new Hilbert space in order to incorporate weighting.

4.2 On the distribution of compositional data

One class of distributions which resides on the $D-$part simplex just as compositional observations, is the class of *Dirichlet distributions*. For $x \in \mathcal{S}^D$ and $\alpha = (\alpha_1, \ldots, \alpha_D)' \in \mathbb{R}_+^D$ the density function is defined as

$$f(x, \alpha) = \frac{1}{B(\alpha)}\prod_{i=1}^{D} x_i^{\alpha_i - 1}.$$

The normalizing constant is the multivariate beta function composed as a product of gamma functions, i.e.

$$B(\alpha) = \frac{\prod_{i=1}^{D}\Gamma(\alpha_i)}{\Gamma\left(\sum_{i=1}^{D}\alpha_i\right)}.$$

Before the introduction of logratios, this distribution was used for the statistical analysis of compositional data, see Aitchison (1986). It is widely investigated, among others in Bayesian statistics where is is used as a prior and posterior distribution, as well as the conjugate prior of the multivariate normal distribution. Next to that, marginal distributions follow a Dirichlet distribution again. The same holds for subcompositions and amalgamations of the data. Besides that, the fixed proportional representation allows zeros in parts of the composition.

That all seems very beneficial, but already in Aitchison (1982) a few drawbacks of this density were outlined. The isoprobability contour-lines for every Dirichlet distribution

are convex, which means that this class of distribution must fail to characterize concave data patterns as in compositional data. Another impediment is the high independence structure built into its definition, which makes it inconvenient to model compositional data. Also, the Dirichlet distribution only works for observations which exactly sum up to one.

In the more recent setting of coordinate transformations, another severe downside emerges. The Dirichlet distribution also fails to preserve scale invariance. An easy way to show this is by its mode, i.e.

$$mode(x, \alpha) = \left(\frac{\alpha_1 - 1}{\sum_{i=1}^{D} \alpha_i - D}, \ldots, \frac{\alpha_D - 1}{\sum_{i=1}^{D} \alpha_i - D} \right)'.$$

Assume that for $D = 3$ and $\alpha = (1, 2, 3)$, the mode is $(0, 1/6, 2/6)$. Scale invariance implies that rescaled observations, which would lead to a rescaling of α, have the same mode. But if α is multiplied, i.e. $(2, 4, 6)$ by two the resulting mode would be $(1/12, 3/12, 5/12)$

These inconveniences cannot be hurdled even when adjusting the Dirichlet distribution to the Aitchison geometry, see Monti et al. (2011) and Pawlowsky-Glahn et al. (2015).

Another noticeable case discussed in the following is the *multivariate normal distribution*. As groundwork, the relationship between $log(x)$ and the clr coefficients as well as between the latter one and ilr coordinates are utilized, i.e.

$$\begin{aligned} \text{clr}(x) &= L_A \log(x), \\ \text{ilr}(x) &= V' \text{clr}(x), \end{aligned}$$

see Filzmoser et al. (2018) and (10). V denotes the matrix of orthonormal basis vector corresponding to the pivot coordinates. This can be combined into

$$\text{ilr}(x) = V' L_A \log(x).$$

In classical compositional data analysis it is assumed that x follows a normal distribution for some fixed V, if $\text{ilr}_V(x)$ follows a multivariate normal distribution, i.e. $\text{ilr}_V(x) \sim N(0, \Sigma)$, whereas $\Sigma \in \mathbb{R}^{(D-1)\times(D-1)}$ is a positive definite matrix. With that and the connection between the ilr coordinates and the logarithm of the data derived above, $\log(x)$ must follow a multivariate normal distribution with mean 0. Since L_A is singular and V non-quadratic, the resulting distribution is degenerate and requires the use of the Moore-Penrose inverse, i.e. $\log(x) \sim N(0, L)$ with $L = (V' L_A)^\dagger \Sigma (L_A V)^\dagger = (L_A V \Sigma^{-1} V' L_A)^\dagger$. The superscript $\dagger$ indicates the pseudo inverse.

The reciprocal of the covariance, $L_A V \Sigma^{-1} V' L_A$, has a few important properties connecting it to the theory discussed in the last section. It is symmetric, positive definite and the vector of $(1, \ldots, 1) \in \mathbb{R}^D$ is an eigenvector to $(0, \ldots, 0) \in R^D$ or, in other words, lies in the nullspace of L. Every symmetric matrix L with $L\mathbf{1} = \mathbf{0}$ can be partitioned as follows $L = \text{diag}(W\mathbf{1}) - W$, whereas W is also symmetric with zero diagonal.

If we restrain ourselves onto non-negative entries in W, by the above thoughts, the inverse of the covariance of $\log(x)$ follows the structure of a Laplacian matrix.

This coincides with the results in the context of graphical models derived by Lauritzen (1996). There they showed that the precision matrix Σ^{-1} for multivariate normal distributed data u, i.e. $u \sim N(0, \Sigma)$, with positive definite Σ, represents the graph structure by the conditional independence between the variables. This connection can be stated as follows, Egilmez et al. (2017)

$$\begin{aligned}
\mathbb{E}\left[x_i|(x_1,\ldots,x_{i-1},x_{x+1},\ldots,x_D)\right] &= -\frac{1}{\Sigma^{-1}_{ii}}\sum_{j=1}^{D}\Sigma^{-1}_{ij}x_j,\\
\text{Prec}\left[x_i|(x_1,\ldots,x_{i-1},x_{x+1},\ldots,x_D)\right] &= \Sigma^{-1}_{ii},\\
\text{Corr}\left[x_ix_j|(x_1,\ldots,x_{i-1},x_{x+1},\ldots,x_{j-1},x_{j+1},\ldots,x_D)\right] &= -\frac{\Sigma^{-1}_{ij}}{\sqrt{\Sigma^{-1}_{ii}\Sigma^{-1}_{jj}}} \quad \forall \;\; i \neq j.
\end{aligned}$$

These conditional terms are the *minimum mean square error* (MMSE) and the precision of x_i as well as the *partial correlation* between x_i and x_j using all the remaining random variables. For example, if the two variables x_i and x_j are conditionally independent, i.e. $\Sigma^{-1}_{ij} = 0$, there is no edge between the vertices of the respective vertices representing these variables.

It seems natural to estimate L by the inverse of the sample covariance matrix of the data $\hat{\Sigma}$. The problem with that lies in the high number of parameters to be estimated which would result in a non-stable estimator. Also the desired sparsity of the graph respectively in the weight matrix will not be present, because in general there are no zero entries in $\hat{\Sigma}$, Yuan and Lin (2007). That is where specific optimization problems as discussed in the next section come into play.

4.3 Estimating weights

As mentioned before, only non-negative weights are considered in the following. Depending on the situation, it makes sense to choose appropriate weights in advance to put emphasize on certain logratios. If there is no expert knowledge available beforehand, according to Rieser and Filzmoser (2022) it makes sense to assume that the data follows a distribution with improper density

$$\frac{1}{(2\pi)^{D/2}\,|\alpha I + L^{\dagger}_{W}|_{+}} exp\left(-\frac{1}{2}||x||^2_{W,\alpha}\right). \tag{17}$$

Here $||x||_{W,\alpha}$ denotes their extension of the Aitchison inner product to a graph structure on $\log(x)$, i.e.

$$\langle x, y\rangle_{W,\alpha} = \alpha\langle \log(x), \log(y)\rangle_2 + \langle \log(x), L_W \log(y)\rangle_2$$
$$||x||_{W,\alpha} = \sqrt{\langle x, x\rangle_{W,\alpha}}.$$

The term $|.|_+$ stands for the pseudo determinant described in Minka (2000).

According to the last section, in order to detect underlying relationships between the variables one must find the inverse of the covariance of the given (log-transformed) data. An appropriate search space including all possible Laplacian matrices is

$$\mathcal{L} := \{L \in \mathbb{R}^{D\times D} | \forall i \neq j : L_{ij} \leq 0, -L_{ii} = \sum_{i=1}^{D} L_{ij}\}.$$

While the last condition is equivalent to $L\mathbf{1} = \mathbf{0}$, both constraints assure that L can be decomposed as in (15). So-called *inverse covariance* estimation problems consider $\mathcal{L}$ as their target space.

Based on the distributional assumptions of the data $X \in R^{n\times D}$ and $X \sim N(0, \Sigma)$, a penalized log-likelihood problem under different frameworks was considered in Yuan and Lin (2007) and Friedman et al. (2008). The problems were of the form

$$\hat{\Sigma}^{-1} := \underset{\substack{L\in\mathbb{R}^{D\times D},\\ L'=L, L \text{ p.d.}}}{\arg\min} \log(|L|) - \text{trace}\left(L\left(\frac{1}{n}X'X\right)\right) + \lambda||L||_1, \tag{18}$$

whereas p.d. stands for positive definite. This is the Gaussian log-likelihood with an additional lasso constraint in $\lambda||L||_1$ which encourages sparsity of the resulting estimator. In order to extend this to the compositional setting, i.e. $X \in \mathbb{R}_+^{n\times D}$ sampled after (17), the following problem can be considered (Rieser and Filzmoser, 2022),

$$\min_{\substack{\alpha>0,\\ L=L'}} \log\left(|\alpha I + L|\right) - \text{trace}\left((\alpha I + L)\left(\frac{1}{n}\log(X)\log(X)\right)\right) + \lambda||L - \text{ddiag}(L)||_1$$
$$\text{s.t. } L \in \mathcal{L}, \text{trace}(L) = D - 1,$$

with the logarithm applied coordinate-wise to X. The operator ddiag(L) forms a diagonal matrix consisting out of the diagonal elements of L. The constraint trace$(L) = D-1$ ensures compatibility with the compositional case, L_A.

For $\alpha = 0$, the pseudo determinant would be a discontinuous function, Holbrook (2018), leading to a discontinuous problem which is hard to solve.

Another approach for weight estimation of compositional data was established by Kurtz et al. (2015) using centered logratio coefficients. The transformed observations, i.e. $t_i := \text{clr}(x_i)$ for $i \in \{1, \ldots, n\}$, were collected row-wise in a matrix T. After that,

either (18) with $X \equiv T$, or a series of D optimization problems, by Meinshausen and Bühlmann (2006) extended from a non-compositional framework,

$$\min_{\beta \in R^{D-1}} \frac{1}{n} ||T_j - T_{-j}\beta_j||_2^2 + \lambda ||\beta||_1$$

for $\lambda \geq 0$ are solved. In the above notation, T_j refers to the columns of T for $j \in \{1, \ldots, D\}$ while T_{-j} to the latter with the j-th column deleted. To obtain suitable weights, in a further step $\tilde{w}_{ij} := |\frac{1}{2}(\beta_{ij} + \beta_{ji})|$ for $i \neq j$ and $\tilde{w}_{ii} := 0$ for $i = j$ are defined.

In connection to graph theory, the problem of finding weights can be also viewed as in finding a graph which enables smooth data transfer between its nodes, Friedman et al. (2008). A possible way to quantify how smooth given data on a simple weighted graph is by looking at the function

$$\sum_{i,j=1}^{D} w_{ij} ||x_{.i} - x_{.j}||_2^2 = \text{trace}(X'LX)$$

with w_{ij} denoting the weights between the vertex i and j. That can be translated to variables residing on two well connected nodes, i.e. w_{ij} being large, are expected of having a small distance $||x_{.i} - x_{.j}||$ in order for trace$(X'LX)$ being small.

A general graph learning framework could look like

$$\min_{\substack{W=W' \in \mathbb{R}_+^{D \times D}, \\ diag(W)=\mathbf{0}}} \sum_{i,j=1}^{D} w_{ij} ||x_{.i} - x_{.j}||_2^2 - \alpha \mathbf{1}' \log(W\mathbf{1}) + \frac{\beta}{2} ||W||_F^2,$$

see Kalofolias (2016) for a similar representation. Above, the Frobenius norm is denoted by $||.||_F$. The parameters α and β control the sparsity of the resulting graph.

This can also be extended to the compositional case by

$$\min_{\substack{W=W' \in \mathbb{R}_{\geq 0}^{D \times D}, \\ \text{diag}(W)=0}} \sum_{i,j=1}^{D} w_{ij} ||\log(X)_{.i} - \log(X)_{.j}||_2^2 - \alpha \mathbf{1}' \log(W\mathbf{1}) + \beta ||W||_F^2,$$

see Rieser and Filzmoser (2022).

4.3.1 Iteratively estimating weights

In this thesis another algorithm, similar to those above, is proposed. The key difference lies in the distributional assumptions of the data X. The algorithm is based on the PCA approach discussed in Section 3. For the derivation we take a closer look at the optimization steps of classical multivariate PCA. The resulting optimization problem to

find the first principal component subject to variance maximization and orthonormality can be written as

$$\min_{a} \; -\operatorname{trace}\left(a'\hat{\Sigma}a\right) \quad s.t. \; ||a||_2 \leq 1$$

whereas $\hat{\Sigma} = \frac{1}{n}X'X$ is the sample covariance matrix of X.

In a compositional context this can be reformulated into

$$\min_{L=L'} \; -\operatorname{trace}\left(L\frac{1}{n}\log(X)'\log(X)L\right) \quad s.t. \; L \in \mathcal{L}, ||L||_2 \leq 1.$$

Adding a lasso regularization term, the above objective function results in

$$\min_{L=L'} \; -\operatorname{trace}\left(L\frac{1}{n}\log(X)'\log(X)L\right) + \lambda||L - \operatorname{ddiag}(L)||_1 \quad s.t. \; L \in \mathcal{L}, \; ||L||_2 \leq 1. \tag{19}$$

The constraints ensure that the resulting matrix L will be a Laplacian matrix. The distributional difference of the data comes from the fact that the trace in (19) can be rewritten into $\operatorname{trace}(\frac{1}{n}\log(X)'\log(X)LL)$. If we define $\tilde{L} := LL$ in the latter term results in $\operatorname{trace}(\frac{1}{n}\log(X)'\log(X)\tilde{L})$, similar as in the objective functions of the inverse covariance estimation problems considered above. The resulting distribution of $\log(X)$ is again a degenerate multivariate normal distribution with 0 mean and covariance $(LL)^{\dagger}$.

The problem of this algorithm is, that L appears quadratic in the trace. An idea for solving this issue is to start with a fixed $\tilde{L}$ and solve the following optimization problem

$$\min_{L=L'} -\operatorname{trace}\left(\tilde{L}\frac{1}{n}\log(X)'\log(X)L\right) + \lambda||L - \operatorname{ddiag}(L)||_1 \quad s.t. \; L \in \mathcal{L}, \; ||L||_2 \leq \; 1. \tag{20}$$

The result of this minimization problem will be inserted as $\tilde{L}$ for the next iteration. The idea is to start with for example $\tilde{L} = L_A$ and continue this procedure until convergence. A detailed schema is presented in the following example algorithm.

Algorithm 1: Iteratively weight estimating

Result: given a compositional data set $X \in R^{n \times D}$ this algorithm iteratively estimates the inverse covariance of the data under a L^1 and L^2 constraint

Input : dataset X, optimization parameters (max_iters, λ, error_tol,...)

Output: estimate of weights W and Laplacian matrix L

initialize the starting Laplacian $\tilde{L}$ as $\tilde{L}_{old}$, e.g. with L_A, iter = 0, rel_error = Inf

while rel_error > error_tol and iter < max_iters **do**

solve (20) with $\tilde{L} = \tilde{L}_{old}$ for L

set $\tilde{L}_{new}$ as the solution of the above minimization problem

rel_error = $||\tilde{L}_{new} - \tilde{L}_{old}||_2 / ||\tilde{L}_{old}||_2$

$\tilde{L}_{old} = \tilde{L}_{new}$

iter = iter + 1

end while

recalculate estimate of weights W out of $\tilde{L}_{old}$.

An extension to this algorithm would be to rewrite it in dependence of the weights w_{ij}. Since the weight matrix is symmetric with zeros on the diagonal it seems natural to only consider for example the entries in W above the diagonal. For convenience, we define $\tilde{X} := \frac{1}{n}\log(X)\tilde{L} \in \mathbb{R}^{n\times D}$. We also use the notation of $x_{.l}$ and $x_{i.}$ for column and row vectors respectively. With that, the trace in (20) can be rewritten into

$$\begin{aligned}\operatorname{trace}\left(\tilde{L}\frac{1}{n}\log(X)'\log(X)L\right) &= \operatorname{trace}(\tilde{X}'\log(X)L)\\ &= \operatorname{trace}(\log(X)L\tilde{X}')\\ &= \sum_{i=1}^{n}\log(x_{i.})L\tilde{x}_{i.}'.\end{aligned}$$

In the next step we make use of equation (16)

$$\begin{aligned}\sum_{i=1}^{n}\log(x_{i.})L\tilde{x}_{i.}' &= \sum_{i=1}^{n}\frac{1}{2}\sum_{l,k=1}^{D}w_{lk}\left(\tilde{x}_{il}-\tilde{x}_{ik}\right)\left(\log(x_{il})-\log(x_{ik})\right)\\ &= \frac{1}{2}\sum_{l,k=1}^{D}w_{lk}\sum_{i=1}^{n}\left(\tilde{x}_{il}-\tilde{x}_{ik}\right)\left(\log(x_{il})-v(x_{ik})\right)\\ &= \frac{1}{2}\sum_{l,k=1}^{D}w_{lk}\langle\tilde{x}_{.l}-\tilde{x}_{.k},\log(x_{.l})-\log(x_{.k})\rangle_2.\end{aligned}$$

Since $w_{lk} = w_{kl}$ holds for $k,l \in \{1,\dots,D\}$ and we have that $\tilde{x}_{.l}-\tilde{x}_{.k} = -(\tilde{x}_{.l}-\tilde{x}_{.k})$ as well as $\log(x_{.l})-\log(x_{.k}) = -(\log(x_{.l})-\log(x_{.k}))$ the last term can be simplified as follows

$$\frac{1}{2}\sum_{l,k=1}^{D}w_{lk}\langle\tilde{x}_{.l}-\tilde{x}_{.k},\log(x_{.l})-\log(x_{.k})\rangle_2 = \sum_{l=1}^{D}\sum_{k>l}^{D}w_{lk}\langle\tilde{x}_{.l}-\tilde{x}_{.k},\log(x_{.l})-\log(x_{.k})\rangle_2.$$

The trace in the objective function is now strictly a function of the upper triangular part of W. Lets define w as these entries, column-wise stacked into a vector of size $D(D-1)/2$. Also let $Z := (z_{lk}) \in \mathbb{R}^{D\times D}$ with $z_{lk} = \langle\tilde{x}_{.l}-\tilde{x}_{.k},\log(x_{.l})-\log(x_{.k})\rangle_2$.

This matrix is also symmetric and has a zero diagonal. Therefore, we analogously define $z \in R^{D(D-1)/2}$ as a vector consisting of the stacked columns of the strictly upper triangular part of Z. Then the conversion can be finalized as follows

$$\begin{aligned}trace\left(\tilde{L}\frac{1}{n}X'XL\right) &= \sum_{l<k=2}^{D}w_{lk}\langle\tilde{x}_{.l}-\tilde{x}_{.k},log(x_{.l})-log(x_{.k})\rangle_2\\ &= w'z.\end{aligned}$$

With that the first three conditions in (20) are automatically fulfilled. The L^2 norm appearing in the constraints can be calculated by recreating W and L out of w. The L^1 norm in the objective function (20) is just the sum of absolute values of each non-diagonal entry of L, in other words two times the sum of absolute values of the elements from w.

Thus, one iteration of optimization problem can be rewritten into

$$\begin{aligned} \min_{w \in R_+^{D(D-1)/2}} & -w'z + \kappa ||w||_1 \\ s.t.\ & w \in \mathbb{R}_{\geq 0}^{D(D-1)/2}, ||L_w||_2 \leq 1. \end{aligned} \tag{21}$$

Below, an example algorithm describes the derived optimization problem in a similar fashion as before. Note that in the simulation in Chapter 5 the vectorized version of the algorithm is not considered.

Algorithm 2: Iteratively weight estimating - vectorized

Result: given a compositional data set $X \in R^{n \times D}$ this algorithm iteratively estimates the strictly upper triangular part of the weight matrix W via a vectorized inverse covariance optimization problem

Input : dataset X, optimization parameters (max_iters, κ, error_tol,...)

Output: estimate of weights W

1 initialize the starting weight vector w as w_{old}, e.g. with $\frac{1}{D}\mathbf{1}$, iter = 0, rel_error = Inf

2 calculate $\tilde{X} = \frac{1}{n}\log(X)L_A$ and initialize z as z_{old} by using $\tilde{X}$

3 **while** rel_error > error_tol and iter < max_iters **do**

  solve (21) with $z = z_{old}$ for w

  set w_{new} as the solution of the above minimization problem

  rel_error = $||w_{new} - w_{old}||_2 / ||w_{old}||_2$

  $w_{old} = w_{new}$

  recalculate $\tilde{L}$ from w_{new}

  set $\tilde{X} = \frac{1}{n}\log(X)\tilde{L}$ and calculate z

  iter = iter + 1

4 **end while**

5 calculate the weight estimate W out of w_{new}.

5 Simulation

In the first part of the empirical simulation in this thesis we consider artificial data sets to analyze the behavior of the algorithm for problem in Equation (20) over different sparsity parameters λ. The simulation itself is carried out in `R` (R Core Team, 2019). The data originates from the `igraph` (Csardi and Nepusz, 2006) package which allows a simple handling of graphs. For the following minimization the convex optimization tool `CVXR` (Fu et al., 2020) is utilized. Different dimensions of observations as well as compositional parts are considered while also the number of relevant logratios changes throughout the procedure. After that the performance of each configuration is evaluated over different aspects.

5.1 Data generation

As described in (4.3.1), for the generation of a data we need to assume our data to follow a graph structure. For that the `igraph` package is considered. It is an efficient open source programming tool for an easy analysis of graphs and networks. It includes many functions to randomly generate different types of graphs and visualize them afterwards. In this work we consider the function

```
g <- sample_gnp(D,p)
```

which produces a simple graph `g` for `D` vertices, i.e. compositions. The fixed probability `p` of two nodes being connected can be adjusted manually. These parameters are the only one relevant in this work.

From `g` the adjacency matrix A can be extracted by the function

```
A <- as_adjacency_matrix(g)
```

This matrix describes the connectivity between the nodes of a graph. An entry a_{ij} equals one if there is an edge between vertex i and j and zero otherwise. Therefore, it is a symmetric matrix and since we only consider graphs with no self-loops, its diagonal is zero. For simplicity we only consider weights to be zero or one, so that the weight matrix W equals the adjacency matrix directly. This makes it also easier to recalculate the weights from the Laplacian after the algorithm is finished.

With (15) and the thoughts in Section 4.3.1, the Laplacian L and the covariance $(LL)^\dagger$ are computed. Under the considered distributional assumptions $\log(X)$ is generated followed by row and column centering which replaces an observations $x_{i\cdot}$ by its centered logratio and also ensures that $\frac{1}{n}\sum_{i=1}^{D}\log(x_{i\cdot}) = \mathbf{0}$.

5.2 Evaluation

The performance of the algorithm is evaluated by the *explained variance* EV similar to R^2, the coefficient of determination. For that the resulting Laplacian L is needed. The computation of EV itself consists of several steps:

1. We start with the SVD of the Laplacian L, i.e. $L = U\Sigma U'$. Note that for L in (13) $U = W$ holds. Then we calculate the projections of $\log(X)$ onto eigenvectors which correspond to positive eigenvalues in Σ. Let us define these columns of U with $\tilde{U}$. For the next step we start with an index set $I := \{\}$.

2. Now, for each index $j \in 1, \ldots, n$ and $j \notin I$, the projection of the combined indices $I \cup j$, i.e. $\left(\tilde{U}'\log(x_{i.})\right)_{i \in I \cup j}$, are fitted onto $\log(X)$ by a linear regression model. Let the resulting fitted values be denoted by $\log(X)_{j,fitted}$.

3. In the final step a goodness of fit between $\log(X)_{j,fitted}$ and $\log(X)$ is consulted as the quality measure. The explained information of these projections is calculated by

$$EV = 1 - \frac{||\log(X) - \log(X)_{j,fitted}||^2}{||\log(X)||^2}.$$

Given the index, we set $I := I \cup \{j\}$ for that component j for which the above information is maximized and continue the procedure at step **2**. This iteration leads to projection directions which subsequently increase the explained variance. The idea of this measurement is to see how good (lower dimensional) projections of $\log(X)$ onto the eigenvectors of L given the distributional assumption of the data approximate $\log(X)$.

To get a first visual impression of this, the explained variance can be plotted against the number of components. They will be called *variance component plots*. It is obvious that the higher the number of components get, the higher the explained variance will be since more information of the original data gets available. It would be ideal to get results where already a low number of compositional parts describes a high amount of variance of the data.

Regarding the resulting sparsity of $\tilde{L}$, a plot of the sparsity parameter λ versus the actual sparsity of the Laplacian can be analyzed. Increasing values for λ should result in a higher sparsity of $\tilde{L}$.

What also would be interesting is the trade-off between the explained variance and the sparsity of the Laplacian. Ideally, we would obtain a sparse solution for $\tilde{L}$ while the explained variance is high. For lower levels of the connectivity `p`, the resulting structure should be close to the underlying one and therefore also achieve higher values. When the number of relevant logratios increases, a sparse solution might not be able to result in a high explained variance.

5.3 CVXR

For solving the minimization problem in (20) the package `CVXR`, see Fu et al. (2020), is used. It supplies an object oriented modeling language for convex optimization problems. It allows the user to intuitively implement convex problems.

In the beginning the target variable is defined by

```
Ltilde <- Variable(n,D)
```

where `n` is the row dimension and `D` is the number of columns.

An objective function of a minimization problem, as in this work, can be defined by

```
objective <- Minimize(-sum(diag(Lold %*% Sig %*% Ltilde))
                      + lambda * sum(abs(Ltilde - diag(diag(Ltilde)))))
```

whereas the input of the function follows the classical `R` syntax. The package also provides built-in functions that simplify formulations.

Additional constraints on the variable are collected into a list object by

```
constraints <- list(Ltilde - diag(diag(Ltilde)) <= 0,
                    Ltilde - t(Ltilde) == 0,
                    Ltilde %*% rep(1,D) == 0,
                    sum(Ltilde^2) <= 1)
```

Then the optimization problem is defined by

```
problem <- Problem(objective,constraints)
```

and solved with

```
result <- solve(problem)
```

`CVXR` internally converts the problem into standard form by graph implementations before passing it on to a solver. For this analysis the cone solver *ECOS* is used.

Next to the target variable, values like the number of iterations and the solving time are returned.

5.4 Optimization parameters

The parameter λ in (20) controls the sparsity of the resulting Laplacian. The higher it is, the more zeros L should contain. The behavior of different values for λ will be determined by *cross validation* over different configurations of the number of observations n, the compositional parts D and the connectivity `p` of the underlying graph structure.

During k-fold cross validation the data set is split into k equal parts. In each iteration of this procedure one of the k sets is considered the test set while the algorithm is executed on the remaining $k-1$ folds. The performance is then evaluated by calculating the explained variance for the left out fold. For each configuration an overall efficiency can be calculated as the mean explained variance over all k iterations.

In the following we will consider 5-fold cross validation. The parameter n can take the values $50, 100, 200$ and 500 whereas D ranges between $5, 10$ or 20. These values are

considered to give a good overview how the algorithm behaves on various parameter configurations. Next to that, also different levels of connectivity between the compositional parts are considered. This is achieved by varying the value of `p` in `sample_gnp(D,p)` between 0.1 and 0.5. These values seem reasonable since they simulate situations where not all logratios are relevant. Too low values could lead to an empty graph with no relevant connection implying the Laplacian equaling a matrix of zeros. That is why for $D = 5$ only vales above 0.1 are considered. On the contrary, if an analyst assumes a high number of relevant ratios beforehand, finding the latter might not be the main purpose of such an algorithm. It would rather be of interest to only get a validation for this assumption.

The lower `p` is, the less the probability of nodes being connected and the more sparse W gets. This implies that less logratios seem to be relevant. Therefore, it should be possible to capture the variance of the data by a potentially smaller number of these logratios. If `p` attains higher values, more nodes are connected and a higher number of logratios may be needed to explain the data sufficiently well.

We examine twenty different values for the sparsity parameter. For that, we consider equidistant steps in the interval between $\log(0.01)$ and $\log(20)$, i.e. $\log(0.01) = \phi_1 < \phi_2 < \ldots < \phi_{20} = \log(20)$. From that we take $\lambda_i = \exp(\phi_i)$ as the sparsity parameters. Logarithmic scaling of lasso parameters an established method during cross-validation.

The maximum number of updates for $\tilde{L}$ is set to 30. Next to that, the error threshold for the relative error in each iteration is 10^{-5}. Table 2 gives an overview of the just discussed domains of the optimization parameters.

Optimization parameter	Domain
sparsity parameter λ	$\{\exp((i-1)(\log(20) - \log(0.01)/20)) \ , i \in \{1,\ldots,20\}\}$
number of observations n	{50,100,200,500}
number of compositional parts D	{5,10,20}
connectivity probability `p`	{0.1,0.2,...,0.5}
error tolerance	10^{-5}
maximum iterations	30

Table 2: Optimization parameters for the 5-fold cross validation.

5.5 Results

In the following we will take a look at the resulting variance plots over various configurations in order to get a feeling of the behavior of the algorithm. We will start with lower values of D and increase it subsequently. It might happen that due to numeric errors not all eigenvalues of the resulting Laplacian are non-negative, or that all values are zero or less. In these instances the explained variance can only be calculated for less

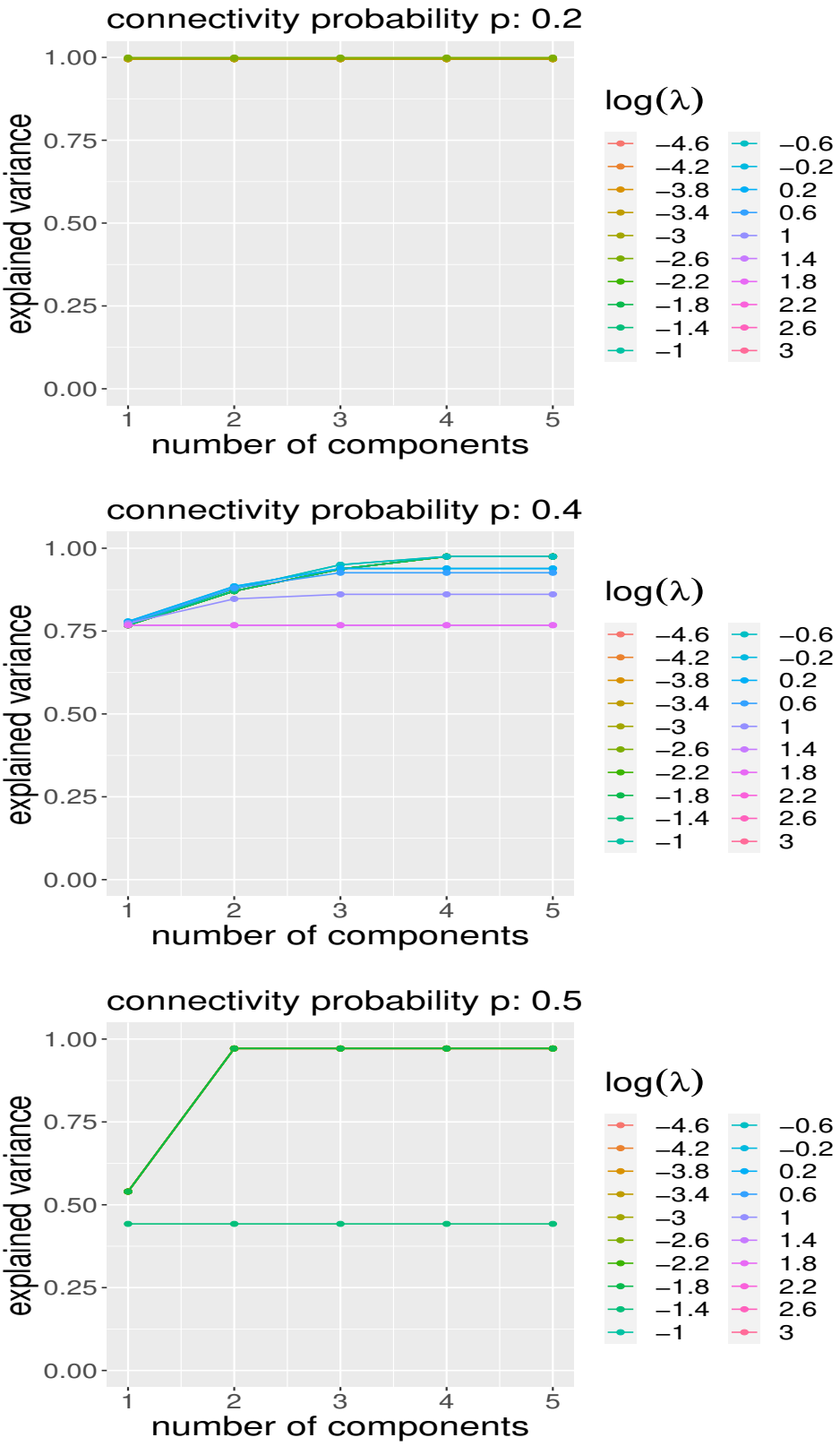

Figure 3: Variance component plots for different sparsity levels; $D = 5, n = 200$.

than D, say $\tilde{D}$, components. If that situation occurs, the variance for the projections based on $\tilde{D}+1,\ldots,D$ ratios is set to the last attained value.

The first plots start with $D = 5$ and $n = 200$. In Figure 3 the results for the connectivity probabilities p $0.2, 0.4$ and 0.5 are displayed. For $p = 0.2$ the corresponding graph is plotted in Figure 4. As we can see, there is only one edge between two components. This seem to be a relevant connection, since already for one component the explained variance is nearly one. Hence there is little to no improvement when the number of components is increased. For higher values of p the number of components must be increased in order to achieve a higher explained variance. Even though more logratios are relevant due to the construction of the underlying graphs, less projection direction can already explain a lot of variance of the data. In contrary to the linear increase in variability for p$= 0.4$, for a value of 0.5 there is a high jump for some sparsity levels when increasing the components from one to two.

As we will see later, the sparsity of $\tilde{L}$ gets close to one, which results in a explained variance of zero. What might look like some lines are missing, can also follow from the fact that the lower values of λ are close together, resulting in similar explained variances and therefore overlapping paths.

As explained before, constant paths result due to numeric issues or no positive eigenvalues being present.

It slowly becomes apparent that the higher the sparsity parameter get, the lower the maximally achievable explained information gets. For higher values of λ the paths seem to converge to a lower maximum.

Different counts for the number of observations n lead to similar results.

Figure 4: Simple graphs for 5 and 10 nodes and connection probability p$= 0.2$, respectively.

For $D = 10$ and $n = 200$ we get an even clearer picture of our last thoughts. Here, we look at four different values for p. The more connections there are between the variables, the more components are need to achieve a higher explained variance. For lower values of p, i.e. 0.1, we notice that two components already explain the whole data very well, while additional components only minorly improve the variance. Increasing p subsequently lead to lower explained variances for a lower number of components used

for the projections. Whilst for $\mathtt{p} = 0.2$ the available information starts at around 0.65, for 0.4 and 0.5 these values drop to 0.55 and 0.4 respectively.

We again notice that the level of the explained variance does not converges to one for some sparsity parameters. That is what we would expect for higher values of λ as the level of sparsity in L will not be able to capture the relevant information of the original data.

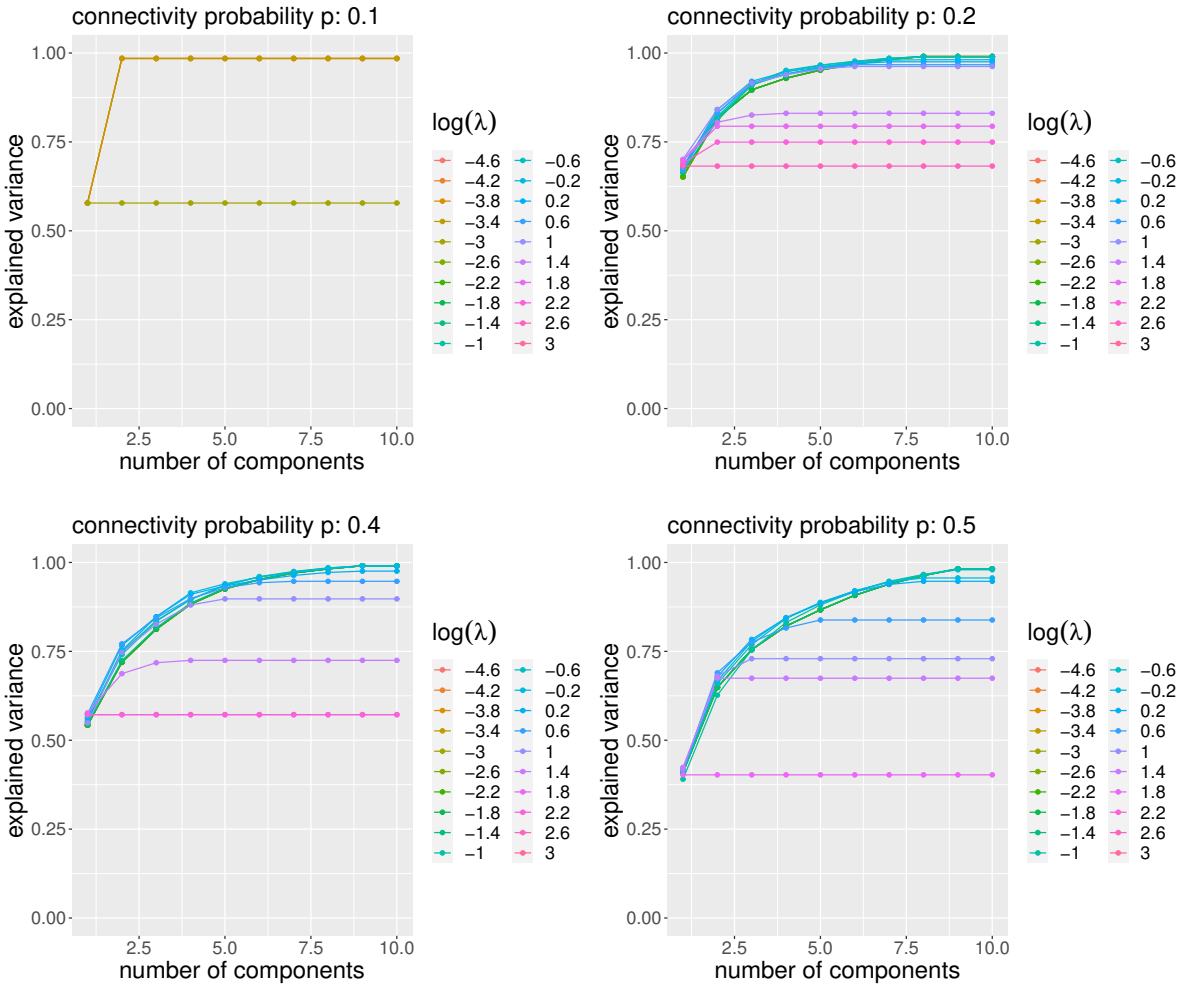

Figure 5: Variance component plots for different sparsity levels; $D = 10, n = 200$.

Increasing D to 20 yields a more smooth results for $\mathtt{p} = 0.1$. Still, a low number of components can explain the variability of the data sufficiently well. For $\mathtt{p} = 0.2$ we again see that the higher λ gets, the lower the achievable variance gets. Values of 0.4 and 0.5 conclude in similar fashion.

What we also notice is, that for all three dimensions the different sparsity parameters λ act very similar.

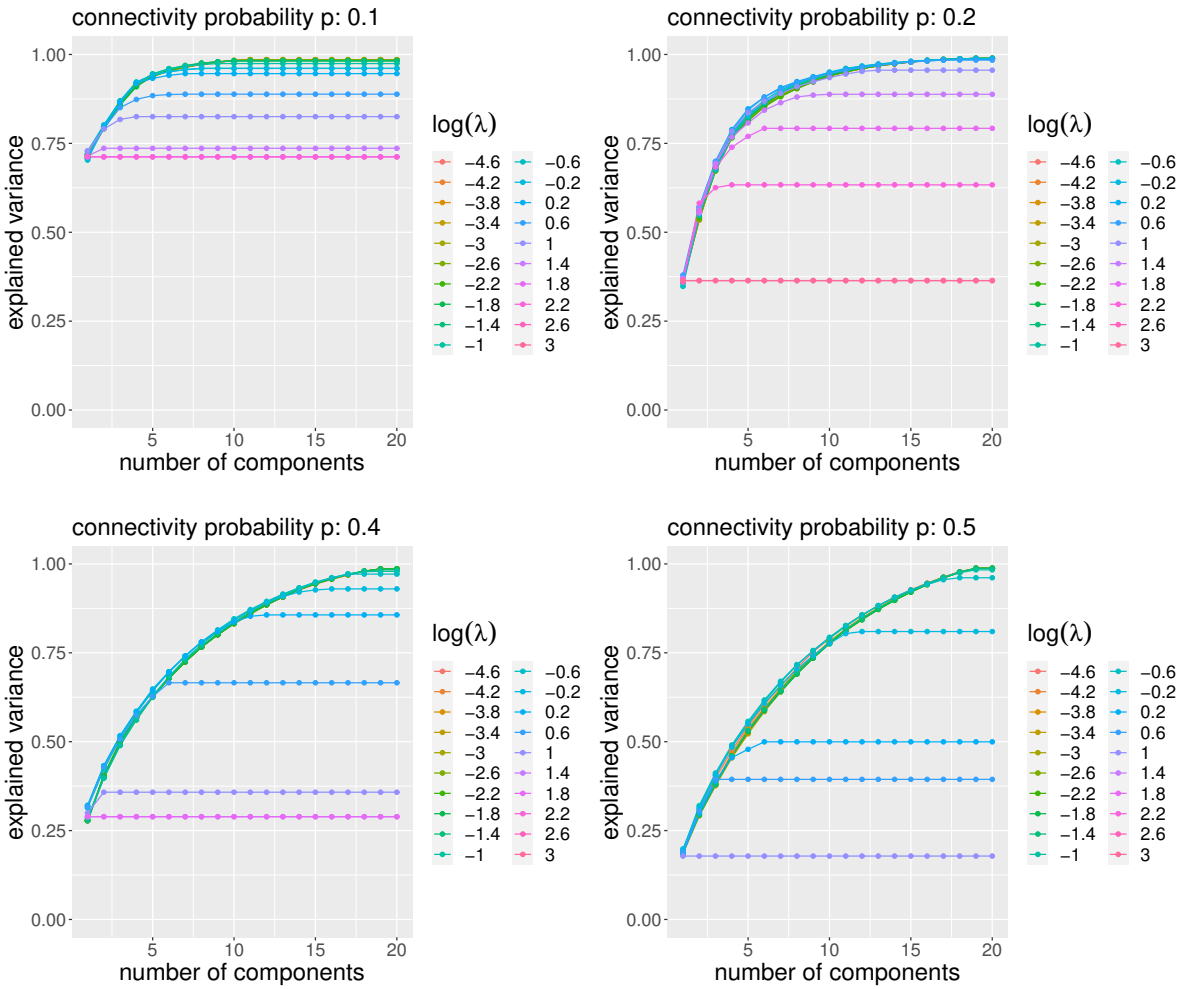

Figure 6: Variance component plots for different sparsity levels; $D = 20, n = 200$.

Another question might be the direct impact of λ onto the sparsity of the weight matrix respectively the Laplacian. For this, the relative sparsity of the estimated Laplacian is calculated by the number of entries being zero divided by the total number of parameters in $\tilde{L}$, i.e.

$$\frac{\#\{(i,j)|\hat{L}_{ij} = 0\}}{D(D-1)/2}. \tag{22}$$

These sparsities are averaged over the folds and collected for each λ. In Figure 7 we see the resulting trends for different values of D and p.

In all configurations we get an increase in the sparsity of $\tilde{L}$ when the sparsity λ increases. Due to numeric issues it might occur that the number of zero entries in $\tilde{L}$ already starts relatively high or begins rather low. Nevertheless, due to increasing λ, the algorithm always reduces the sparsity of the Laplacian.

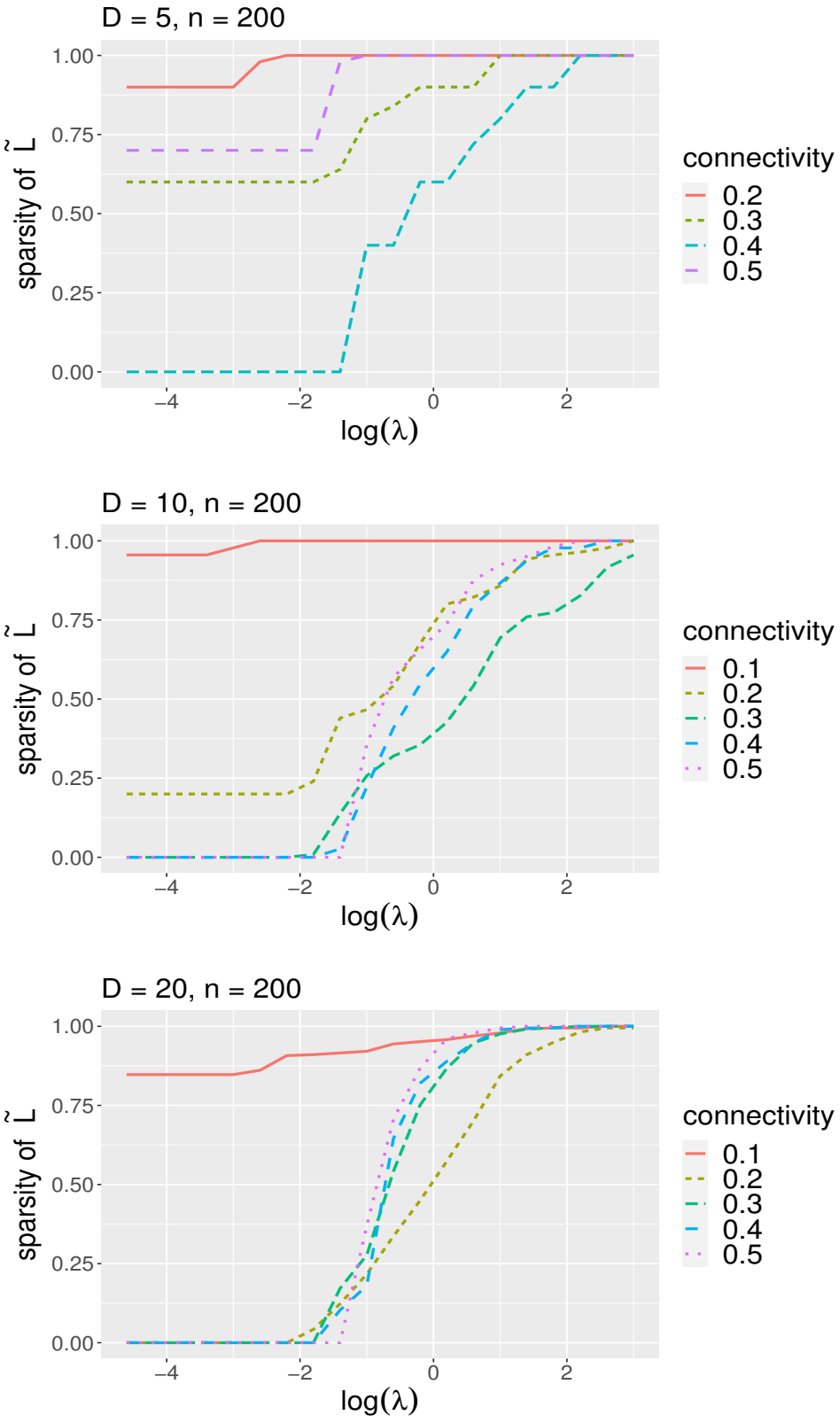

Figure 7: Relative sparsity of the Laplacian $\tilde{L}$.

For the lowest value of compositional parts the behavior of this increase seems to be different for each connectivity level. If $\mathtt{p} = 0.2$ the underlying graph structure is already very sparse. The algorithm seems to catch that, resulting in the Laplacian in being nearly fully spars from the beginning and rising very quickly afterwards. For 0.3 and 0.4 this changes takes place significantly later. If D now increases to 10 or 20 these paths align more and more while configurations with higher connectivity levels get sparse faster.

What is interesting is that the sparsity for $\mathtt{p} = 0.1$ always starts significantly higher than for the remaining levels. It seems that if the number of relevant logratios is lower, the algorithm is able to reproduce the underlying graph structure quite well from the beginning.

In the final part of this section we will look at the trade-off between the sparsity of $\tilde{L}$ and the contained information. In Figure 8 we have plotted the mean values resulting from the cross-validation for the relative sparsity against the explained variance. Higher values of λ are denoted by brighter colors, while different connectivity levels $\mathtt{p}$ are labeled by different symbols.

The first thing to notice again here is, that for higher values of D the explained variance decreases subsequently.

Higher values for D do not necessarily lead to less sparse solutions. For $D = 5$ there is only one configurations where the sparsity of $\tilde{L}$ is significantly low. For remaining ones, not only the corresponding sparsity is high, but also the explained variance. A smaller number of components is already able to explain a high proportion of the information in the data.

Different levels of λ influence the sparsity in the same way for all values of D. For lower values of λ a small increase does lead to an increase in the sparsity as well, while is does not decrease the explained variance. If the lasso parameter reaches higher values, the explained variance suddenly drops down, implying a too sparse solution. This phenomenon occurs in all connectivity levels $\mathtt{p}$.

Another aspect from before can also be observed in these plots again. The higher $\mathtt{p}$ gets, the lower the explained variance is.

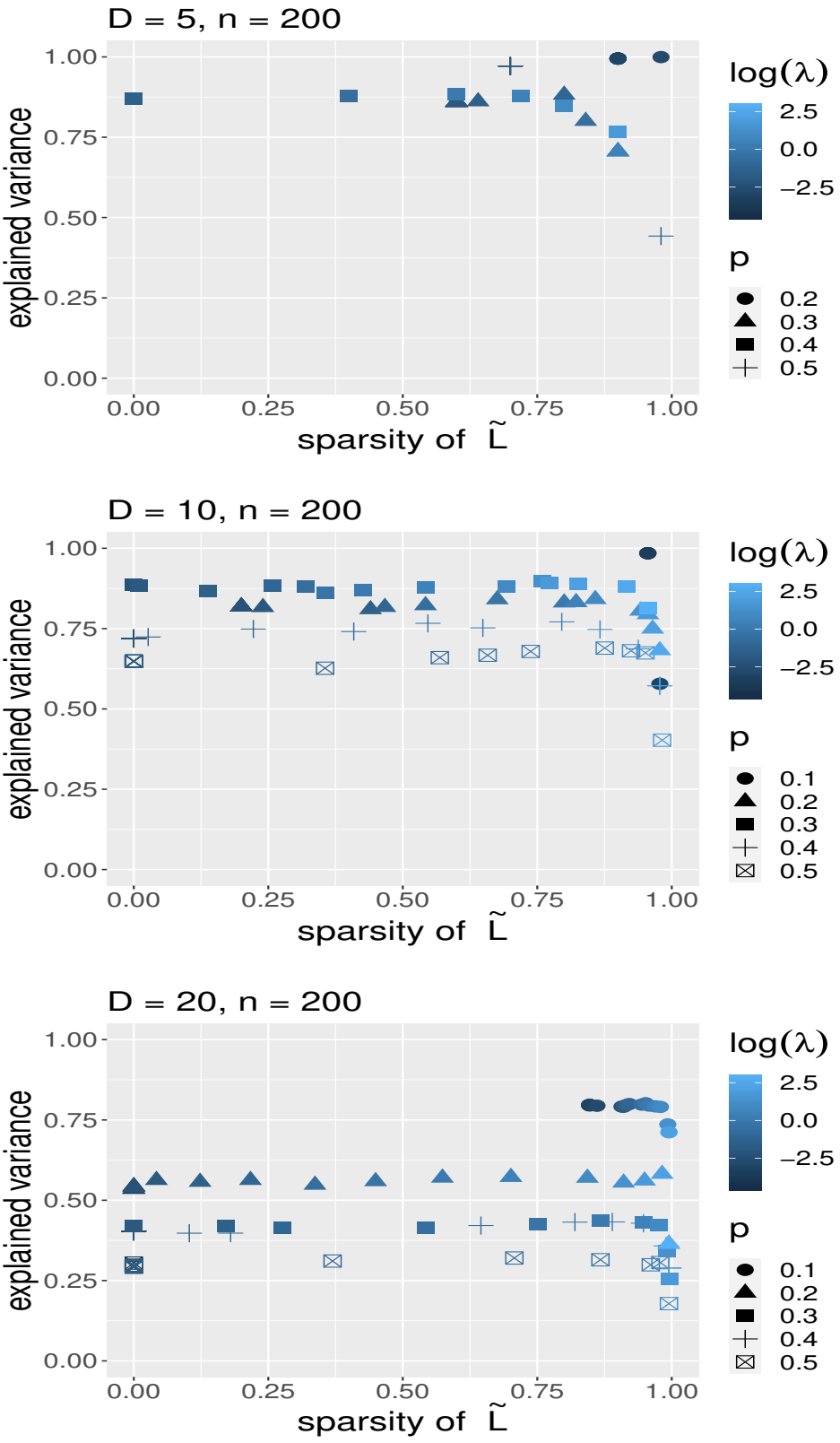

Figure 8: Trade-off between sparsity of $\tilde{L}$ and the explained variance for two components.

6 Real world data application

6.1 Data sets

In this section we will analyze two different compositional data sets with the before introduced iterative algorithm. The first one is the *Kola* data set collected in the Kola Project in the 90s in Finland, Norway and Russia, see Reimann et al. (2010). It contains over 600 observations of soil samples of five different layers. For this analysis we consider a subset of 10 variables corresponding to the concentration of different chemical elements in the C horizon, a specific layer. The data set can be found in the `StatDA` package, see Filzmoser (2020).

The second data we examine originates from the *GEMAS* (Geochemical mapping of agricultural and grazing land soils) project, see Reimann et al. (2012). Among others, the concentration of geochemical elements in agricultural soil have been collected in mg/kg. The data is available in `robCompositions` (Templ et al., 2011). It includes 2108 observations on 30 different variables. In this analysis we deal with 17 of those variables corresponding to the geochemical elements.

Before applying the procedure, the data has to be scaled appropriately. At first the centered logratios from each data set are calculated. Afterwards these ratios are centered by their corresponding column means. The results are the row and column centered logarithms of the original data. These values are assumed to follow a degenerate normal distribution, see Section 4.2. Looking at the histogram of various compositions of the C horizon data set, this assumption can be confirmed.

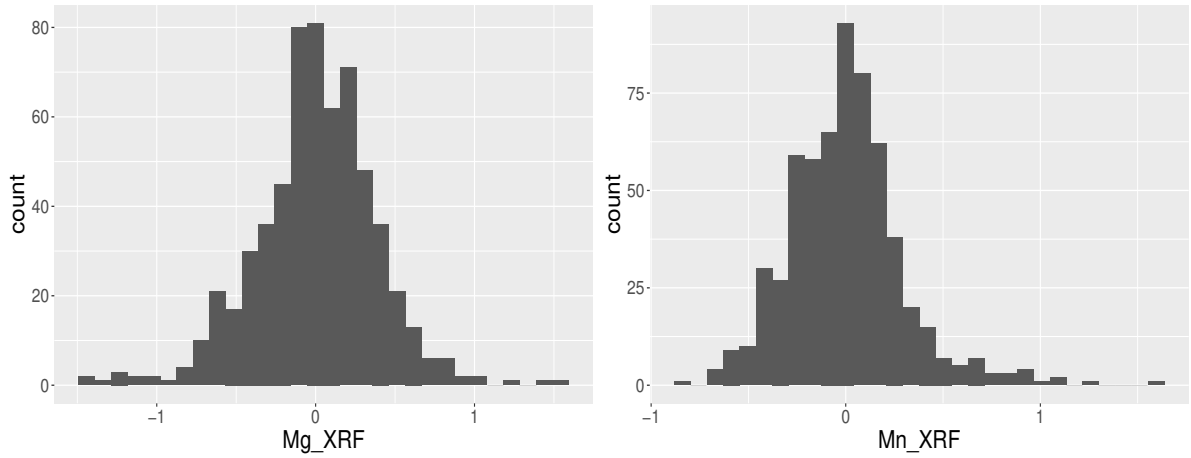

Figure 9: Histogram of different variables of the scaled C horizon data set.

Also for the first two parts of the GEMAS data set, the histograms yield similar results.

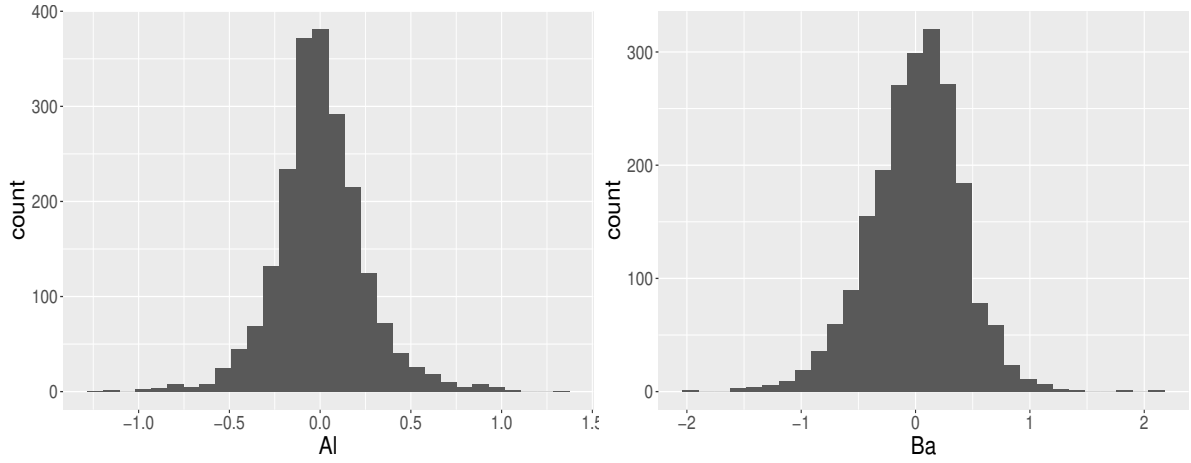

Figure 10: Histogram of different variables of the scaled GEMAS data set.

6.2 Evaluation and results

For the evaluation of these two data sets we shorten the range of sparsity parameters used in Chapter 5 to equidistant points between log(0.01) and log(2). We compare resulting variance plots and look at the sparsity of the resulting Laplacians as well as graphs resulting from different values of λ.

6.2.1 Kola data

We start with the Kola data set. In Figure 11 we see the development of the sparsity of the resulting Laplacian for increasing λ. We notice a steady increase of the sparsity in the beginning for the lower values of the lasso parameter. At a certain level this increase suddenly stagnates.

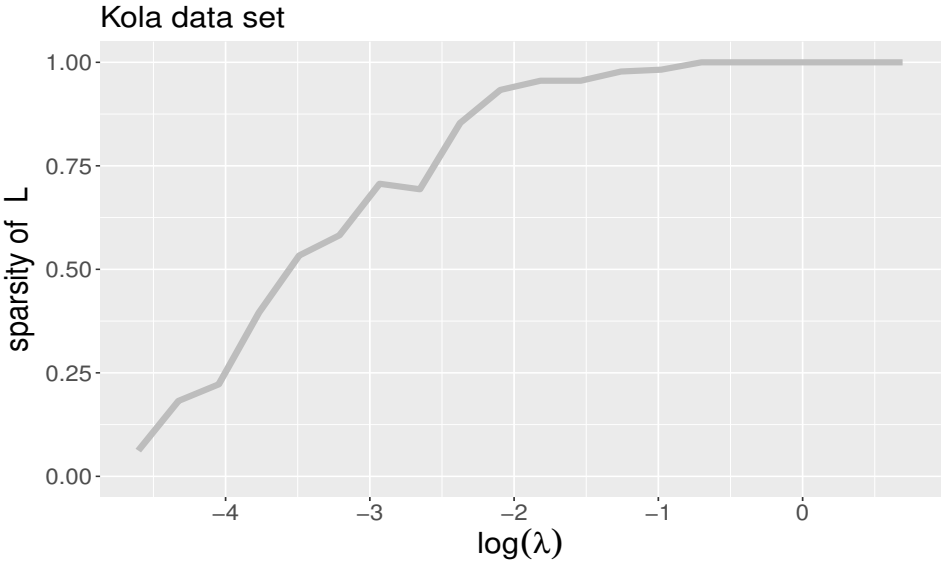

Figure 11: Sparsity of $\tilde{L}$. Kola data set.

Next, in Figure 12 we get another view on the influence of increasing λ. We get an insight at the trade-off between the sparsity of $\tilde{L}$ and the explained variance when projecting the data onto the first two eigenvectors of the Laplacian. With an increase in λ the sparsity increases while the explained variance stagnates. Eventually for the higher lasso parameter there is a sudden drop in the explained variance.

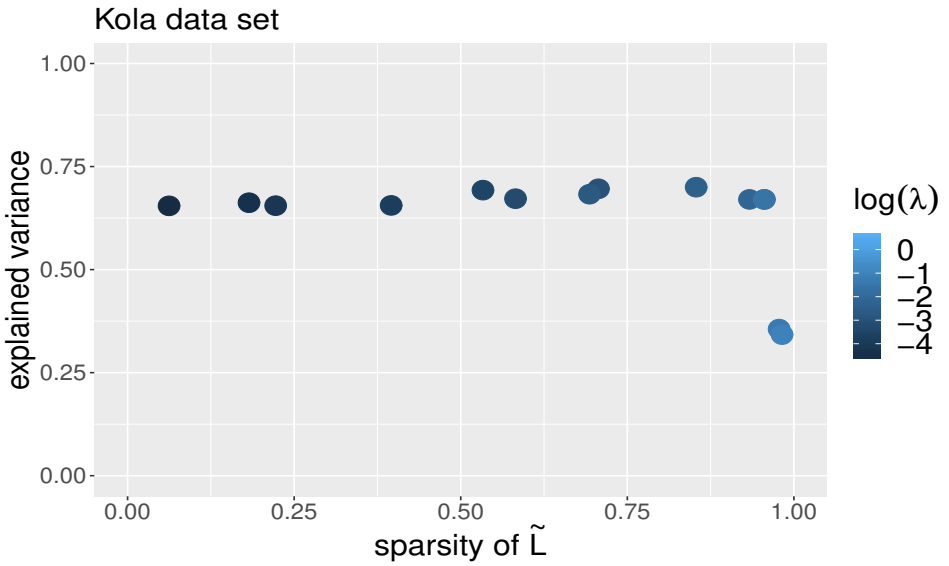

Figure 12: Trade-off between sparsity of $\tilde{L}$ and the explained variance for two components for the Kola data.

In Figure 13 we see the underlying graph structure of the Kola data set for λ before and after this drop in variance. These formations result from the Laplacian. The estimate of the adjacency matrix $\tilde{A}$, which is needed for the construction of the graphs, is extracted from the previous by

$$\tilde{A} = \tilde{L} - \text{ddiag}(\tilde{L}).$$

For simplicity we additionally set all non-zero entries in $\tilde{A}$ to one, i.e. we only consider weights being zero or one.

The upper left graph results from the lasso parameter $\lambda \approx 0.013$. We observe no clear structure or a unique relationship between the compositional parts. The sparsity of the underlying Laplacian is about 18%. The upper right graph comes from $\lambda \approx 0.093$. Now, the relative sparsity is already at 0.85. We detect two clusters. For $\lambda \approx 0.214$ we get the graph on the bottom left. The logratio between the elements phosphorus P and silicon Si does not seem to be important anymore. Furthermore, the second cluster has reduced itself to a single connection between magnesium Mg and sodium Na. All of the previous graphs have an explained variance of about 0.65 to 0.7 for projections onto the first two eigenvectors of $\tilde{L}$. Interestingly, the explained variance drops to 0.34 when the Laplacian gets even more sparse. This is represented by the bottom right graph where the link between Mg and Na has been removed due to change in the sparsity level λ to approximately 0.375. That might imply that the logratio between the parts Mg and Na plays a significant role when analyzing the data.

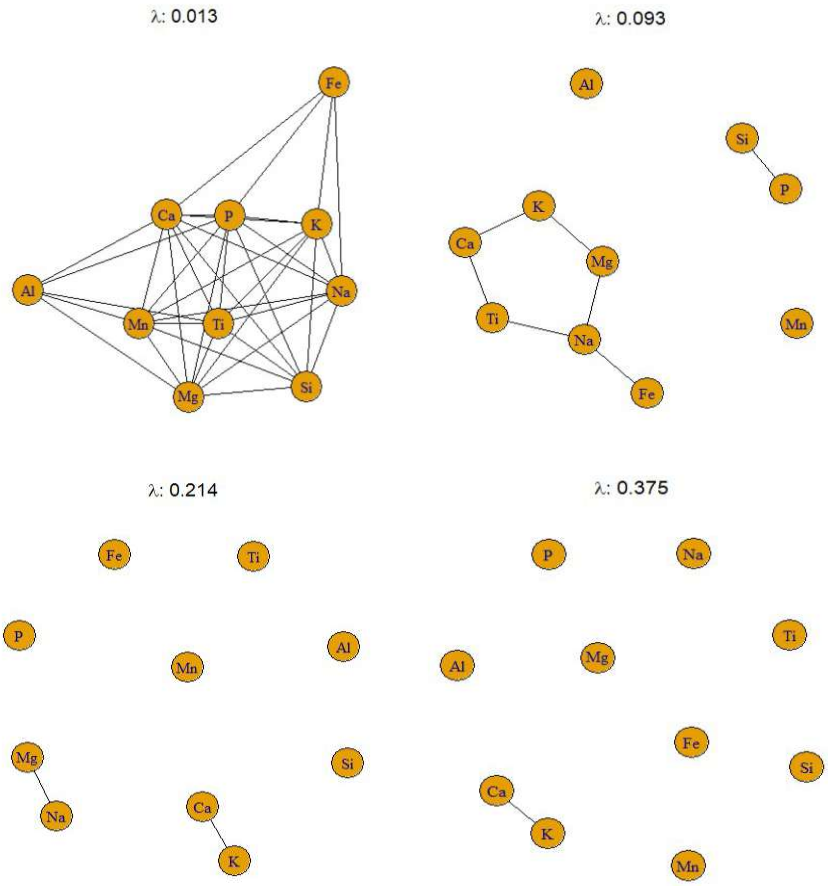

Figure 13: Graphs for different sparsity parameters. Kola data set.

Finally, in Figure 14 the variance plot of the data set is displayed. The results seem reasonable as the higher values of λ lead to a lower maximum of the explained variance.

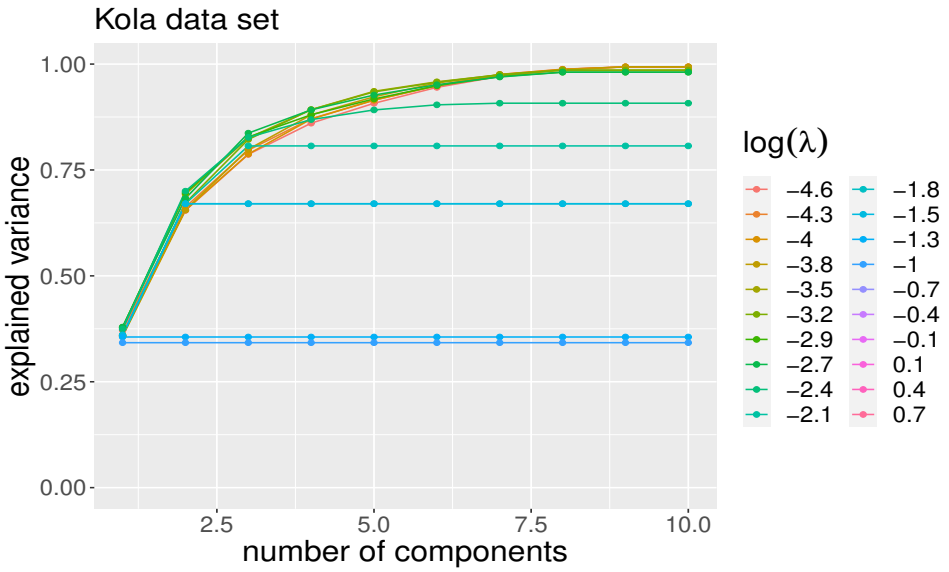

Figure 14: Variance component plot for the Kola data set.

6.2.2 GEMAS data

In contrary to the last example, the graph in the sparsity plot of the GEMAS data set 15 follows a more s-shaped curve. It seems like it takes the algorithm some time before a higher value of λ leads to a higher sparsity. Roughly at about the same value for λ as in 11 the graph flattens and the influence of λ onto the structure of $\tilde{L}$ falls off.

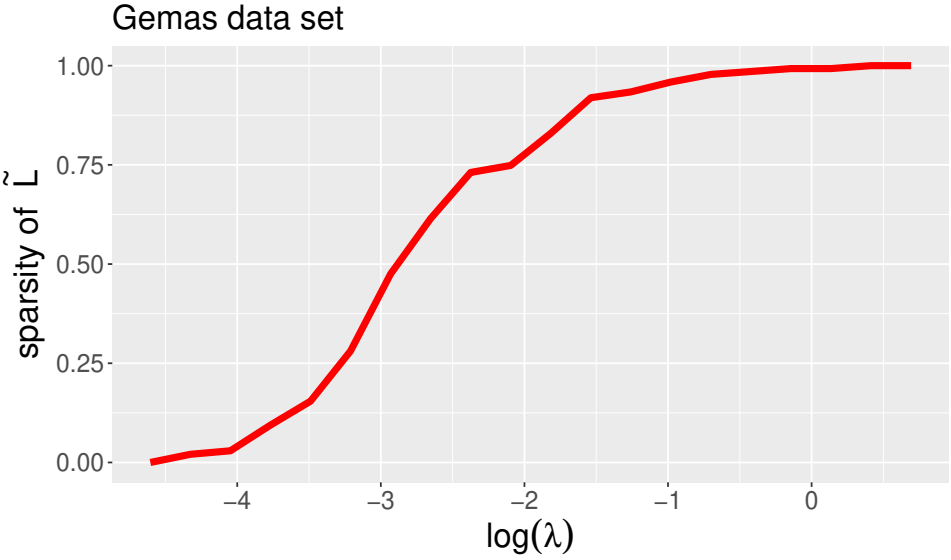

Figure 15: Sparsity of $\tilde{L}$. GEMAS data set.

The sparsity variance trade off for the GEMAS data set follows a similar structure. For a projection on only the first two eigenvectors of $\tilde{L}$, the explained variance is already very high. Although the number of components is significantly higher than for the Kola data set, the variances are almost the same. This would imply a more sparse structure of the GEMAS data.

Furthermore, an increase in λ leads to a higher sparsity while the explained variance remains nearly constant. Ultimately, this also concludes in a sudden shift of the variance to a lower level. It seems like this drop happens earlier than for the Kola data set.

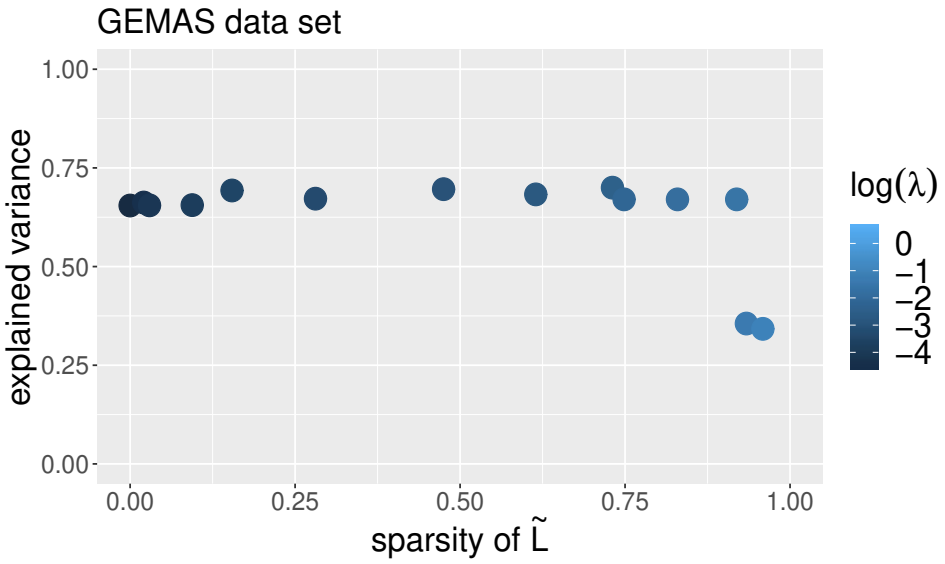

Figure 16: Trade-off between sparsity of $\tilde{L}$ and the explained variance for two components for the GEMAS data.

In Figure 17 we again study the change in the underlying graph structure due to an increase in λ. As we already saw in Figure 15, the increase in sparsity is rather slow when compared to the Kola data set. That becomes noticeable when we compare the graphs for the same levels of λ. The two upper graphs correspond to $\lambda \approx 0.214$ and $\lambda \approx 0.375$. Before the algorithm already had achieved a very sparse structure with only a few logratios being relevant. Here there a still a few more connections between the compositional parts. The explained variance lies between 0.55 and 0.6. If λ gets even higher the number of relevant links continues to reduce and the explained variance drops to around 0.35 to 0.4. The bottom graphs result from $\lambda \approx 0.655$ and $\lambda \approx 0.866$.

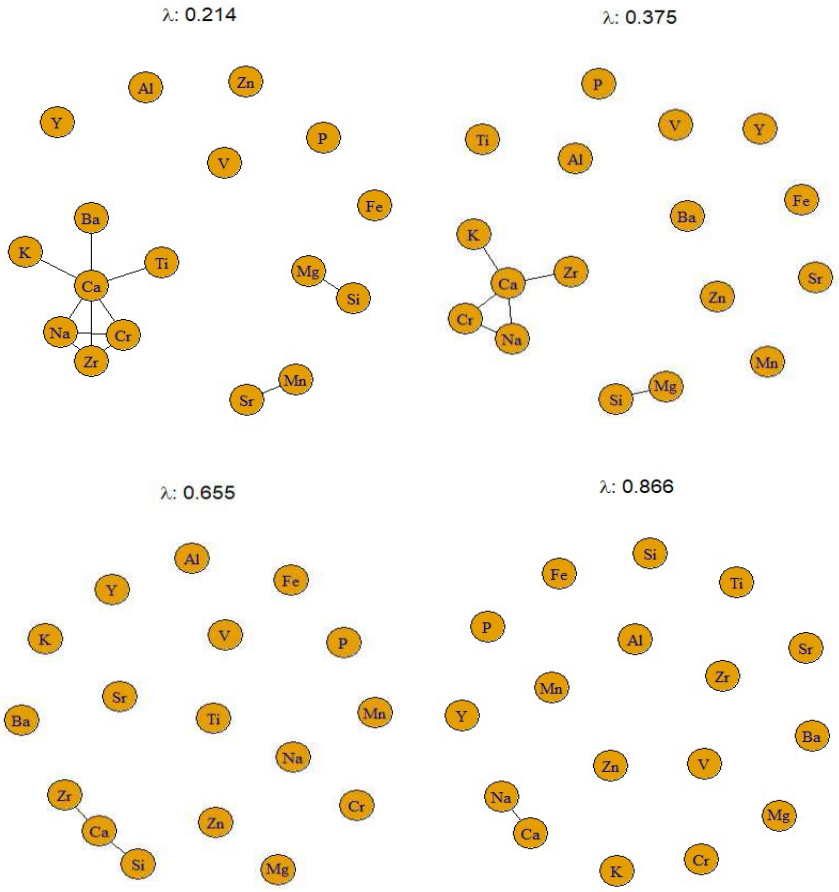

Figure 17: Graphs for different sparsity parameters. GEMAS data set.

In the final variance plot in Figure 18 the implication of different sparsity parameters in the objective function is even more diverse. In contrary to before, we get various paths converging to lower levels of explained variance for higher λ.

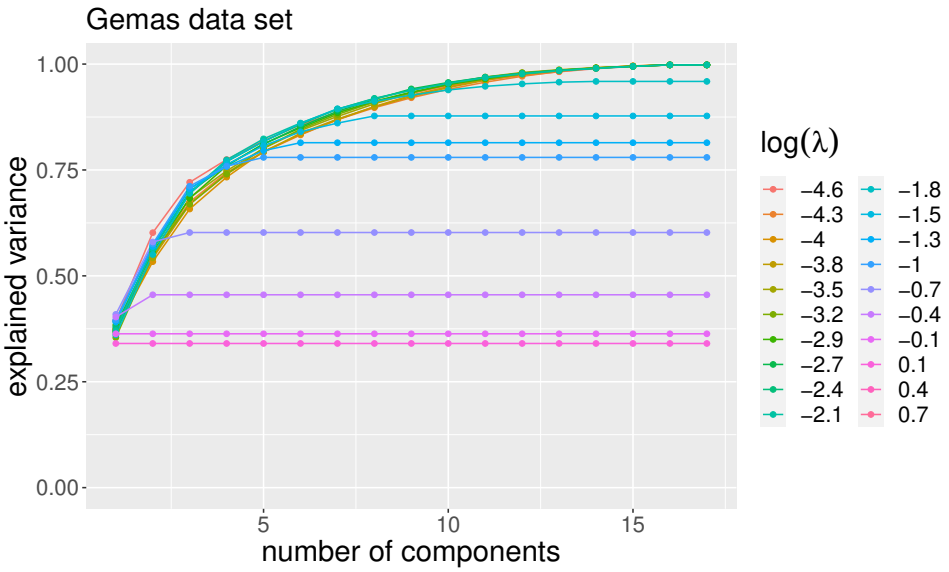

Figure 18: Variance component plot for the GEMAS data set.

7 Conclusions

In the beginning of this work, the concept of positive vectors that contain relative information of a whole was introduced. The "raison d'être" of this framework becomes apparent when looking at the corresponding sample space of such observations, which is significantly different from the well established Euclidean space. These compositions rather originate in the (unit) simplex. Accordingly, it turned out that already sophisticated methods from many areas of multivariate statistics were applied incorrectly onto the constrained data.

The foundation of a solution to this problem was set in the early 80s by the debut of compositional data. That approach was expanded about twenty years later by the Aitchison geometry, which made that idea more tangible.

Over the course of this thesis we came in contact with several fields of application of compositional data like economy or geology. Furthermore, different extensions from standard statistical techniques are considered.

We took a closer look at the concept of dimension reduction. At first we dealt with principal component analysis which could be easily extended to the compositional case.

For the second approach we considered weighting. Relevant variables or ratios of the latter should be higher weighted during the analysis. On the contrary, unimportant logratios should receive little to no weight which would equal a dimension reduction as well. This thought was incorporated into the inner product of the Aitchison geometry and together with results from graph theory a series of optimization problems could be established. Through these the Laplacian matrix of underlying graph can be estimated as the inverse covariance of the corresponding data.

Analogously to principal component analysis, an iterative algorithm for this kind of problem was introduced. An additional lasso term should control the sparsity of the solution. By 5-fold cross-validation over various parameter configuration, including the number of observations or compositional parts, the behavior of the algorithm was analyzed on simulated data. The explained variance was consulted as goodness of fit. The procedure performed as expected. Higher lasso parameters led to more sparse solutions, while the obtainable variance decreased. Still, it was possible to get to a relative high sparsity of the Laplacian before the contained information reduced significantly. The results over different dimensions were similar.

Afterwards, the performance of the algorithm was tested on two geochemical data sets. The Kola and GEMAS, which were both collected in Europe, contain concentrations of chemical elements in soil. Here, results were also promising. The algorithm was able to find sparse solutions, yet containing enough information to describe the data sufficiently

well. Through the structure of the underlying graphs for different sparsity levels it was also possible to find those ratios that might be of importance.

Since these outcomes already look very reasonable, tasks of future work shall extend this framework. Now it is of interest to find a criterion that, among other, selects an optimal sparsity parameter based on a trade-off between the sparsity of the solution and the explained variance.

List of Tables

List of Figures

Bibliography

Aitchison, J. (1982). The statistial analysis of compositional data. *Journal of the Royal Statistical Society*, 44(2):139–177.

Aitchison, J. (1983). Principal component analysis of compositional data. *Biometrika*, 70(1):57–65.

Aitchison, J. (1986). *The Statistical Analysis of Compositional Data*. Chapman Hall, reprinted in 2003 with additional material by The Blackburn Press.

Aitchison, J. (2005). *A Concise Guide to Compositional Data Analysis*. Technical Report. University of Glasgow.

Corporation, M. and Weston, S. (2022). *doParallel: Foreach Parallel Adaptor for the 'parallel' Package*. R package version 1.0.17, https://CRAN.R-project.org/package=doParallel.

Croux, C. and Haesbroeck, G. (2000). Principal component analysis based on robust estimators of the covariance or correlation matrix: influence functions and efficiencies. *Biometrika*, 87(3):603–618.

Csardi, G. and Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal*, Complex Systems:1695. https://igraph.org.

Dowle, M. and Srinivasan, A. (2019). *data.table: Extension of 'data.frame'*. R package version 1.12.8, https://CRAN.R-project.org/package=data.table.

Egilmez, H. E., Pavez, E., and Ortega, A. (2017). Graph learning from data under laplacian and structural constraints. *IEEE Journal of Selected Topics in Signal Processing*, 11(6):825–841.

Egozcue, J. J. (2009). Reply to "On the Harker variation diagrams;..." by J. A. Cortés. *Mathematical Geosciences*, 41(1):829–834.

Egozcue, J. J., Pawlowsky-Glahn, V., Mateu-Figueras, G., and Barcelo-Vidal, C. (2003). Isometric logratio transformations for compositional data analysis. *Mathematical Geology*, 35(3):279–300.

Filzmoser, P. (2020). *StatDA: Statistical Analysis for Environmental Data*. R package version 1.7.4, https://CRAN.R-project.org/package=StatDA.

Filzmoser, P., Hron, K., and Reimann, C. (2009a). Principal component analysis for compositional data with outliers. *Environmetrics*, 20(6):621–632.

Filzmoser, P., Hron, K., and Reimann, C. (2009b). Univariate statistical analysis of environmental (compositional) data: Problems and possibilities. *Science of The Total Environment*, 407(23):6100–6108.

Filzmoser, P., Hron, K., and Templ, M. (2018). *Applied Compositional Data Analysis - With Worked Examples in R*, volume 1. Springer.

Fišerová, E. and Hron, K. (2011). On the interpretation of orthonormal coordinates for compositional data. *Mathematical Geosciences*, 43(4):455–468.

Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

Fu, A., Narasimhan, B., and Boyd, S. (2020). CVXR: An R package for disciplined convex optimization. *Journal of Statistical Software*, 94(14):1–34. https://cvxr.rbind.io/.

Gabriel, K. R. (1971). The biplot graphic display of matrices with application to principal component analysis. *Biometrika*, 58(3):453–467.

Genz, A., Bretz, F., Miwa, T., Mi, X., Leisch, F., Scheipl, F., and Hothorn, T. (2021). *mvtnorm: Multivariate Normal and t Distributions*. R package version 1.1-3, https://CRAN.R-project.org/package=mvtnorm.

Greenacre, M. (2019). Variable selection in compositional data analysis using pairwise logratios. *Mathematical Geosciences*, 51(5):649–682.

Holbrook, A. (2018). Differentiating the pseudo determinant. *Linear Algebra and its Applications*, 548:293–304.

Hron, K., Palarea-Albaladejo, J., Filzmoser, P., and Egozcue, J. J. (2022). Weighting of parts in compositional data analysis: Advances and applications. *Mathematical Geosciences*, 75(1):71–93.

Kalofolias, V. (2016). How to learn a graph from smooth signals. In *Artificial Intelligence and Statistics*, pages 920–929. PMLR.

Kurtz, Z. D., Müller, C. L., Miraldi, E. R., Littman, D. R., Blaser, M. J., and Bonneau, R. A. (2015). Sparse and compositionally robust inference of microbial ecological networks. *PLoS computational biology*, 11(5):e1004226.

Kynčlová, P., Filzmoser, P., and Hron, K. (2016). Compositional biplots including external non-compositional variables. *Statistics*, 50(5):1132–1148.

Lauritzen, S. L. (1996). *Graphical Models*. Oxford Statistical Science Series. Oxford. University Press, 17 edition.

Martín-Fernández, J. A., Pawlowsky-Glahn, V., Egozcue, J. J., and Tolosona-Delgado, R. (2018). Advances in principal balances for compositional data. *Mathematical Geosciences*, 50(3):273–298.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the Lasso. *The Annals of Statistics*, 34(3):1436 – 1462.

Merris, R. (1994). Laplacian matrices of graphs: a survey. *Linear Algebra and its Applications*, 197-198:143–176.

Microsoft and Weston, S. (2022). *foreach: Provides Foreach Looping Construct*. R package version 1.5.2, https://CRAN.R-project.org/package=foreach.

Minka, T. P. (2000). *Inferring a gaussian distribution*. Technical Report. Massachusetts Institute of Technology.

Mohar, B. (1991). The laplacian spectrum on graphs. *Graph Theory, Combinatorics, and Applications"*, 2:871–898.

Monti, G. S., Mateu-Figueras, G., and Pawlowsky-Glahn, V. (2011). *Notes on the scaled Dirichlet distribution*. John Wiley & Sons, Chichester.

Pawlowsky-Glahn, V. and Egozcue, J. J. (2001). Geometric approach to statistical analysis on the simplex. *Stochastic Environmental Research and Risk Assessment*, 15:384–498.

Pawlowsky-Glahn, V. and Egozcue, J. J. (2002). Blu estimators and compositional data. *Mathematical Geology*, 34(3):259–274.

Pawlowsky-Glahn, V., Egozcue, J. J., and Tolosana-Delgado, R. (2015). *Modeling and Analysis of Compositional Data*, volume 1. Wiley.

R Core Team (2019). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/.

Reimann, C., Äyräs, M., Chekushin, V., Bogatyrev, I. V., Boyd, R., Caritat, P. d., Dutter, R., Finne, T. E., Halleraker, J. H., Jæger, , Kashulina, G., Lehto, O., Niskavaar, H., Pavlov, V. A., Räisänen, M. L., Strand, T., and Volden, T. (2010). *Environmental Geochemical Atlas of the Central Barents Region*. Schweizerbart Science Publishers, Stuttgart, Germany.

Reimann, C., Filzmoser, P., Fabian, K., Hron, K., Birke, M., Demetriades, A., Dinelli, E., and Ladenberger, A. (2012). The concept of compositional data analysis in practice — total major element concentrations in agricultural and grazing land soils of europe. *Science of The Total Environment*, 426:196–210.

Rieser, C. and Filzmoser, P. (2022). Extending compositional data analysis from a graph signal processing perspective. *arXiv:2201.10610*.

RStudio Team (2020). *RStudio: Integrated Development Environment for R*. RStudio, PBC., Boston, MA. https://www.R-project.org/.

Templ, M., Hron, K., and Filzmoser, P. (2011). *robCompositions: an R-package for robust statistical analysis of compositional data*. John Wiley and Sons. https://cran.r-project.org/web/packages/robCompositions/index.html.

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org.

Wickham, H., François, R., Henry, L., and Müller, K. (2021). *dplyr: A Grammar of Data Manipulation*. R package version 1.0.6, https://CRAN.R-project.org/package=dplyr.

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the gaussian graphical model. *Biometrika*, 94:19–35.