# Introduction

## Abstract

Machine learning in microbiome studies is widely used and the interest is growing. However, there is no universal understanding of the algorithmic approaches that can best utilize the information present in the microbiome data. Thus, this is an interesting and widely discussed topic that can have a great impact on the potential applications leveraging microbiome data. A key topic in microbiome research is the sample space of the input data. The sequencing data appears as count data, but, only relative abundance of the microbial features can be observed, commonly called “compositional data”. Thus, transforming the read counts to relative abundances is usually the first step and machine learning methods are usually applied on relative abundances. However, relative abundances raise several limitations, which can have an impact on the performance of the prediction models. Therefore, log-ratio transformations are a proposition made by several studies now, however their impact on machine learning performances has never been tested in large-scale studies. The goal of this benchmarking project is to rectify that and conduct several machine learning models under several log-ratio transformations in comparison to *CoDaCoRe*, an algorithm specifically made with microbiome analysis in mind. This way it will become clearer if a scientist should make the effort in learning about machine learning methods, when automated algorithms perform well enough, and no heavy prior machine learning knowledge is necessary.

## Background

Working with mathematical concepts is always a bit out of the comfort zone for most biologists. Unfortunately, with technical improvements and big data encroaching in our field, and statistical methodology being an essential part in data analysis, ignoring mathematics is just not an option. One concept that became increasingly more important in sequencing data analysis is the concept of “compositional data”. Several papers (Greenacre et al. 2021b; Gloor et al. 2017; Quinn et al. 2018) made it abundantly clear that sequencing data is of compositional nature, which means it has different mathematical characteristics than other data types. Furthermore, as machine learning concepts become more widespread and useful, their performance in combination with compositional data and its necessary transformations have not been fully analyzed.

The goal of this master thesis is an attempt in making the information around compositional data more approachable, summarizing the achieved solutions, and in a practical part, trying to assess if these achievements are also applicable when combining compositional data and machine learning concepts.

As this master thesis uses microbiome sequencing data and was created in a microbiome research group, this text will mostly focus on this field and its papers. However, all results are applicable to other high-throughput sequencing data, as well as any data that is in some way confined by an arbitrary sum. Such data is found for example in geochemistry, ecology, sociology, political sciences, etc., and therefore ultimately spans the problematic into various different fields (Greenacre et al. 2021a).

### Characteristics of Compositional Data

In order to define and illustrate the concept (and problems) of compositional data, let’s assume a classical biological example. The following Figure (1A) shows two different ecological fields: A and B. In field A, four rabbits, seven birds, eight bees and one wolf have been counted, whereas field B contains two rabbits, four birds, four bees and one wolf. It becomes clear that, as similar as the diversity may be, the fact that field B seems to have only half of the population of field A, is already valuable information in itself. The total counts per field can be preserved in our data collection and therefore, the absolute count of each organism in this field matters.

Graphical user interface, application

Description automatically generated with medium confidence

Figure 1: Information Loss of Normalized Data

(A) Illustration of the number of animals found in two different samples. Field A contains four rabbits, eight bees, seven birds and one wolf, whereas field B contains two rabbits, four bees, four birds and one wolf. In (B) the absolute counts have been plotted as a stacked bar plot, with each animal in a different color. (C) shows the stacked bar plot as normalized counts, e.g., percentages.

When using absolute counts, the difference between both fields is easily visible (1B). However, when we really want to compare both fields, we need to transform the samples to a common scale. This is called normalization and we can see the effect in (1C). As soon as the data is normalized, the particular information of absolute counts gets lost. When collecting ecological data ourselves, we can preserve the fact that field B only contained 11 individuals and field A contained 20, by saving that number somewhere on our Excel sheet. However, the problem with sequencing data is: we get the data in the form of 1C.

To demonstrate how the method of using a sequencing machine cannot preserve absolute counts, imagine the following situation: We want to sample field A multiple times a day, but in order to be more efficient, we buy a machine to do the counting for us. Three times a day this machine transmits the number of all the different animals coming to this field. However, this machine has one flaw: it can only count to 20. As soon as the 21st animal on this day comes to the field, it is just simply not counted.

This ultimately means, that the overall number of 20 carries no meaning. Every sample has this exact total number, so it carries no valuable information. Of course, a limit of 20 is weird for us to understand, but sequencing machines do the exact same thing. They are limited in their capacity on the flow cells and even the biggest sequencing machines could never fully sequence the entirety of the organisms RNA contents. And not only the sequencing machines, but the whole RNA-Seq procedure limits the total number of sequences measured. The total number of sequences measured by sequencing machines ultimately depends on the *chemistry of the assay*, not the input material (Quinn et al. 2018).

The consequence of this sampling problem is, that we have to accept the fact that the sum of counts in sequencing data are irrelevant. This leads us to the concept of “Compositional Data”, first introduced by John Aitchison (Aitchison 1982), roughly 40 years ago. Because thankfully, we can still use sequencing data. We just have to adjust for the fact that the absolute counts are non-informative (Quinn et al. 2018; Greenacre et al. 2021b). We can instead use relative abundances, or the proportions between features in a sample.

### The Simplex Space

As mentioned, absolute counts in compositional data are irrelevant and only relative abundances are of interest. This puts the data in the so-called “simplex space”, instead of the for us more common Euclidean space. The following Figure 2 shows how the data from field A would look like A picture containing diagram

Description automatically generatedin the simplex space:

Figure 2: Biological Example in the Simplex Space

Assuming the collected ecological data from Figure 1 is compositional, puts it in a S3-Simplex space. Geometrically, a tetrahedron is created with all different components (here animals) placed on the four corners of the polytope. A composition is one possible combination of components confined in the simplex space.

We stick with our ecological example and place all our animals as one corner in a geometrical space. With four features, we are able to create a 3-Simplex and a geometric figure called tetrahedron (otherwise called a pyramid).

We use our flawed machine, and one day, we sample 20 rabbits in field A. This would lead to a point in the simplex space that sits directly in the left corner, with the coordinates (0,20,0,0), because we only have rabbits, no other animal. Another day, we sample only 20 bees, and no other animals, then we would find our data point where the bee is, very at the top. Marked in red is a sample where the machine counted 12 bees, 12 wolves and no bees and no rabbits. Every sample round produces one “composition” and the examples show, that the distance between any two variables is sensitive to the presence or absence of other components (Quinn et al. 2018). If a composition is moved from one corner of the animal-simplex, it directly influences the other values in the composition. Consequently, that makes all variables *mutually dependent* on one another and leads, amongst other things, to problems in our assumptions about statistical testing. In statistical literature this data is also called “spurious” because it appears as if the data points have a causal relationship. When a composition is moved from bees in the direction of wolves it seems like there is a causal relationship because the increase in the number of wolves, directly decreases the number of bees.

To describe this a bit more mathematically, the problem described above is formally known as “the negative bias problem” (other names are also the constant-sum problem, the closure problem, or the null correlation difficulty) (Aitchison 2003), which is the main reason why we have a dependency problem. When the sum of a component is constant, then it can be mathematically proven that the covariance between any two compositions equals 0. This has the consequence that some variances would be negative, which is problematic, as variances are always positive. Therefore, negative covariances are presupposed by the limitation of the sum, instead of produced by stochastic factors (Pawlowsky-Glahn and Egozcue 2016; Aitchison 2003).

Additionally, it is commonly assumed - and all experiments are created to accommodate these assumptions – that data is collected IID: independent and identically distributed. The IID assumption is important for e.g., the central limit theorem, Markov sequence, hypothesis testing in general and of course machine learning. Having such an obvious violation in compositional data can have serious consequences on the reproducibility of results. In life sciences, count data are usually modelled using the Poisson distribution or negative binomial, because using anything else would imply that negative and non-integer counts would exist, which is biologically not feasible (Quinn et al. 2018).

Using any form of statistical test or machine learning tool seems redundant, as a type two error is almost preconditioned, and we easily would make false assumptions about the correlation of the data. Thus, a correct handling of compositional data and the simplex space is not optional (Gloor et al. 2017).

### Mapping the Simplex Space into Euclidean Space

The difficulty of confined data points has already been commented on by Pearson (1897) in the context of spurious correlations and has been taken up by Aitchison 1982 in an attempt to overcome the “bounded sum problem”.

In our ecology sampling, we could easily overcome the simplex by e.g., normalizing to a field size from the start, to preserve indirectly an information about the total number of animals.

Similarly, it has been tried for sequencing data to calculate an “effective library size” and to recover this way the original scale of data. For that, normalization methods like trimmed mean of M-values (TMM) have been introduced, as well as RPKM and TPM (Quinn et al. 2018). However, all of those methods involve rescaling counts by the library size and these normalizations come with the drawback that some of these methods are sensitive to the removal of low abundant counts, as well as to data symmetry (Quinn et al. 2018).

Furthermore, Aitchison already criticized very early that there is no “magic to open up closed data” (Aitchison 2003), which is what normalization tries to do. Moreover, since information provided in compositional data is essentially about ratios of the components, it seems logical to also think in terms of ratios. Thus, the only way forward is to transform the data in a way that allows us to use it with Euclidean space rules, first by Aitchison 1982 with several logistic transformations proposed to produce “transformed-normal” models, and later with the definition of the Aitchison geometry (Pawlowsky-Glahn and Egozcue 2001). The general idea is, that the simplex space is endowed with a Euclidean space structure, which has several mathematical advantages: if one can map the simplex space into Euclidean space, then all advantages of the Euclidean space can be accessed, i.e., orthogonal projections are possible, the concepts of linear combination, linear dependence, Euclidean distances, as well as all the typical geometrical elements are available (Pawlowsky-Glahn and Egozcue 2016).

Building on top of the Aitchison geometry, methods of analyzing compositional data were proposed by Mateu-Figueras et al., (2011) with the “staying-in-the-simplex” approach or Greenacres (2017) “pragmatic approach”. In this master thesis, mentioning these methods is as far as I will go here, because they require a technical understanding of the algebraic-geometric structure of the simplex. Here, I will focus more on log-ratio transformations, as they have been more heavily favored in the last decades due to their practicability (Greenacre et al. 2022).

### Log-Ratio Transformations

There are several types of log-ratios, which were proposed of the last 40 years, and I want to take the time and introduce them. Some more in detail than others, as not every log-ratio transforms the data perfectly and it is important to point out here, that there are still ongoing discussions about which log-ratio transformation is preferrable over the other in terms of accuracy, complexity, and interpretability (Greenacre et al. 2022; Quinn and Erb 2020; Rivera-Pinto et al. 2018).

In general, all log-ratio transformations capture the relationship between the features in the data set and taking the logarithm of these ratios makes the data symmetric and linearly related. It moves the simplex into real space and imparts key properties on the data set: scale invariance (compositions do not change with e.g., sequencing depth), perturbation invariance (i.e., converting a composition between equivalent units will not change the results), and permutation invariance (i.e., changing the order of the components within a composition will not change the results).

Two more important properties exist that are transformation-specific: sub-compositional coherence (i.e., scientists A and B get identical results for components when these components are included in compositions) and sub-compositional dominance (i.e., using a subset of a complete composition carries less information than using the whole) (Quinn et al. 2018; Greenacre et al. 2021a; Greenacre et al. 2022). From a scientific standpoint, it seems to be a no-brainer to try to adhere to both of these properties, as they are the gold standard of reproducibility. Not following sub-compositional coherence would mean that two sequencing runs from the same patient (and the same bioinformatics pipeline) couldn’t be compared and ignoring sub-compositional dominance would mean we couldn’t filter data before using.

The log-ratio transformation that imparts all those properties is called isometric log-ratio (ILR). ILRs are considered the “gold standard” of log-ratio transformations, as they engender exactly the same multivariate geometric structure of the sample points as that of the formerly mentioned Aitchison geometry (Greenacre et al. 2021b). The ILR maps a composition in the D-part Aitchison-simplex isometrically to a D-1 dimensional Euclidian vector, which is not just confusing to understand but makes it also difficult to interpret (Greenacre et al. 2021a; Greenacre et al. 2022). Additionally, they are also particularly problematic when the numbers of components are high (Greenacre et al. 2021b), which is a quality worth considering as sequencing data is usually very high-dimensional.

Text, whiteboard

Description automatically generatedThankfully, we have more types of log-ratio transformation, that are easier to use and interpret. They do not fully impart sub-compositional coherence, but interestingly, it has been shown recently that quasi-coherence is sufficient in practice, as well as quasi-isometry (Greenacre et al. 2022), especially in high-dimensional data sets. As a result of this, it was decided to use two types of log-ratio transformation in this master thesis: ALR (additive log-ratio) and CLR (centered log-ratio).

Figure 3: Equation for CLR

The equation describes the calculation of CLR, with xj as vector of sample features, Dj the total number of features, and g(x) the geometric mean of sample vector x. Log-ratio transformations are applied within a sample (i.e., row-wise).

The CLR uses the geometric mean of the whole composition as the reference feature (Gloor et al. 2017). It has the advantage that it is computationally easy to do, which becomes more important with high-dimensional data sets. Furthermore, it reproduces the log-ratio geometry perfectly, but is not sub-compositionally coherent, because the whole composition (i.e., sample) is used to calculate the geometric mean and every sample will therefore use a different geometric mean. Unfortunately, it is not very easy to interpret and it is not very useful in sparse data containing a lot of 0s (Gloor et al. 2017).

A picture containing text, watch

Description automatically generatedThe second log-ratio transformation is ALR. Here, the log-ratio is taken of each measurement within a composition and divided by a chosen reference feature.

Figure 5: Equation ALR

The equation describes the calculation of ALR, with xj as vector of sample features, D the total number of features, and xref the reference feature. Log-ratio transformations are applied within a feature (i.e., column-wise).

Thus, the interpretation of ALR log-ratios is very straight-forward and it is also sub-compositional coherent, which is traded for a small loss of isometry. The biggest problem with ALR has always been the choice of reference. When choosing a reference, Greenacre et al. 2021 proposed to use three criteria to find a good reference: (i) the reference component should maximize the Procrustes correlation between the additive log-ratio geometry and the exact log-ratio geometry, (ii) the reference should minimize the variance the relative abundances of log-transformed components, and (iii) it should be a well populated component. Using these guidelines produces additive log-ratios close to being isometric, which would make them a favorable log-transformation. The obvious drawback is the computational complexity (if a Procrustes analysis is used beforehand), which increases especially in higher-dimensional data.

In general, log-ratio transformation do not normalize the data (does not “open it”), but makes the interpretation of the transformed data dependent on the reference used and aim for a straight-forward interpretation of the data (Quinn et al. 2018). For machine learning purposes, it is still unclear if any log-ratio transformation improves the performance in a prediction task. This will be one of the core goals of this benchmarking project and previous studies and results will be described in the next section.

### Compositional Data in Machine learning

Predictive methods such as random forests (RF), artificial neural networks (ANN), deep learning (DL) or support-vector machines (SVM) and other methods have become in the last years increasingly popular (Tolosana-Delgado et al. 2019). Vital for good machine learning conclusions is the balancing of predictive power with the explainability, similarly to log-ratio transformations.

In terms of statistical analysis, machine learning models are of great interest for microbiome analysis, as they allow predictions of biomarkers, phenotypes or microbial taxa, as well as other interesting tasks, that are not possible with the standard microbiome tool kit (Marcos-Zambrano et al. 2021). Therefore, a correct application of machine learning models is key to reproducible and interpretable research results. Several studies (Zhang and Shi 2019; Coenders and Greenacre 2021) showed log-ratio transformations in machine learning models with mixed performances. In 2019, Zhang and Shi compared several machine learning algorithms on geological compositional data and showed that overall, RF was the best performing model and that ILR and CLR were superior to ALR (Zhang and Shi 2019). Tolosana-Delgado et al. (2019) concluded that ridge regression and SVM both need ILR. More observations were also made by Quinn et al. 2020. They performed linear discriminant analysis (LDR) on ILR-transformed data and partial least squares (PLS) to CLR-transformed data and showed good predictive results (Quinn and Erb 2020). Neural Networks require further research, but does not seem to be equivariant (Tolosana-Delgado et al. 2019), i.e. not any log-ratio transformation works similarly well.

These observations demonstrate the current predicament between compositional data and machine learning. Log-ratio transformation in linear and generalized linear models are not easily chosen and depend heavily on the observations at hand. In general, log-ratio transformations seem to outperform raw proportions for classification tasks, but it is not clear how log-ratio transformations relate to the changes in predictive performance. Furthermore, employing log-ratio transformations leads to an increase in complexity in the correct application of machine learning models. Thus, it is of increasing importance to create a practical guide for all scientists who want to employ such analysis.

The question arises if machine learning models are “worth the hassle” considering microbiome-specific algorithms like *CoDaCoRe* exist, that are faster and do not require a lot of background knowledge to use. The next section describes this algorithm more in detail and their potential effectiveness.

### CODACORE

The following section is a summary of the paper *“Learning Sparse Log-Ratios for High-Throughput Sequencing Data”* published by Gordon-Rodriguez et al in 2021, where they first introduce *CoDaCoRe.* *CoDaCoRe* is a novel learning algorithm for finding balances (Compositional Data via Continuous Relaxations). Balances are defined as the log-ratios between geometric means of two features of the input variables. Translated, CoDaCoRe finds ratios between two features that are explanatory for the given classification task. Such ratios are commonly used as biomarkers of gut health e.g., the Firmicutes-to-Bacteroidetes ratio (Crovesy et al., 2020; Magne et al.; 2020).

Balances are essentially pairwise log-ratios; however, they allow the aggregation of more than one variable in the numerator and denominator of the log-ratio. This leads to a richer set of features and therefore more flexible models. Usually, pairwise log-ratios are computationally very taxing, which is why they are not separately included in the master thesis. However, in *CoDaCoRe* Gordon-Rodrigues et al. use a deep learning technology called “continuous relaxation” and only approximate the optimization problem, which has the advantage of greatly reducing the runtime.

In its basic formulation, *CoDaCoRe* learns a regression function, which uses balances as weights. The goal of *CoDaCoRe* is to find the balance that is maximally associated with the response variable by minimizing the cross-entropy loss. The continuous relaxation appqroximates the geometric averages over subsets of the inputs, by weighted geometric averages of all components. This makes the relaxation and balances differentiable and allows the use of gradient descent. This has the advantage of a linearly scaling computational cost instead of exponential, which reduces the runtime drastically.

At this step, weighted geometric averages are not easily interpretable. Therefore, *CoDaCoRe* implements a discretization procedure, i.e., fitting a linear model to assess if the previously found balance is impactful. This step can be regularized by influencing lambda in the model creation, which becomes a regularization hyperparameter that can be tuned. In practice, lower lambda is more useful when the emphasis is on predictive accuracy rather than interpretability or sparsity.

In summary, in the full *CoDaCoRe* algorithm, multiple regressors are trained in a stage-wise additive fashion and afterwards each successive balance is fitted on the residual from the current model. Thus, *CoDaCoRe* identifies a sequence of balances, in decreasing order of importance, each of which is sparse and interpretable.

*CoDaCoRe* is a promising algorithm that is created to also work efficiently on big data sets with a lot of features. In their paper, the authors compare *CoDaCoRe* against several machine learning models (Lasso, RF and XGBoost) and show that their algorithm does not sacrifice interpretability nor predictive accuracy.

## Implementation

Recent years made it clear that machine learning is a tool that should be available to all biologists, but comes with high complexity and its own pitfalls, even without the addition of mathematical characteristics of compositional data and log-ratio transformations. The above-described examples already mention that machine learning models in combination with log-ratio transformations do not show clear-cut results. The goal of this project focuses on collecting insights on the performances of machine learning models, but also practicality.

Furthermore, four log-ratio transformations will be compared: CLR (centered log-ratio), and three ALR methods (worst, random and optimal). The performances will be directly compared to TSS-transformed data (total sum scaling transformation) and *CoDaCoRe* in the following conceptual Diagram

Description automatically generatedframework:

Figure 5: Used Pipeline

The graph shows the proposed pipeline for the benchmarking project. Data sets will be collected by their characteristics large/small, high/low correlations and continuous/discrete variables. Afterwards, data sets will be pre-processed by zero-imputation methods and filtering. Microbiome-native methods will be employed and compared to the data being log-transformed and used in machine learning models.

The general pipeline will be constructed of the following building blocks: Pre-processing, Imputation, Transformation, and Machine Learning Models/Microbiome Approaches. The core idea is to observe statistical fluctuations in two chosen machine learning models, given the same training data set. After the data split, a repeated cross-validation is used to find the best model and its performance is saved for plotting. It will be explored in further detail if filtering affects model performances, as well as the different transformations itself. Descriptions on the exact methodology can be found in the respective section further down.

It has to be acknowledged that his conceptual pipeline is considered “leaky” as the imputation procedure use the whole data set for zero replacement. Only afterwards is the data set split into training and test set, which makes the test set not totally unbiased. As not all data sets are big enough to allow a separate imputation for the small test set, it was therefore decided to conduct imputation on the whole data set and additionally test the impact of data leakage on transformations and machine learning models in a separate test.

Additionally, as the study design for the CRC data set had the concept of holdout test sets in mind, this opens the possibility of comparing model performances for holdout sets versus a classic 80/20 split, as the latter is planned for the general pipeline to allow better comparison between model performances.

The goal of these tests will be to find practical guidelines for compositional data and machine learning models.