Regression Models for Compositional Data

Ragib Zaman

An essay submitted in partial fulfillment of
the requirements for the course
COMP8755 - Individual Computing Project

College of Engineering and Computer Science
Australian National University

May 2019

Abstract

In this report we explore regression models involving Compositional Data (CoDA). We begin by defining CoDA, looking at examples and understanding why the analysis of CoDA requires special treatment. Then we give an overview of the current standard methods of dealing specifically with CoDA developed and popularised primary by J. Aitchison as well as several others. We then review some regression models for CoDA, giving particular focus to a model presented in (H. Wang et al, 2013) where the dependent variable and all independent variables are compositions. Motivated by ideas from Information Geometry, we consider alternate loss functions for this regression model and evaluate its success compared to traditional models across several metrics.

Acknowledgments

First and foremost I would like to thank Dr. Cheng Soon Ong. I greatly value the wisdom he has shared with me, helping me identify the core ideas in machine learning and the links between. His willingness to give his time so generously and with such patience has been very much appreciated.

I would also like to thank Dr. Weifa Liang and Dr. Stephen Gould for taking the responsibilities of course convener and examiner respectively. Without their contribution I would not have been able to have the pleasure of completing this course.

CONTENTS

CHAPTER 1

Compositional Data

1.1. Introduction

A composition with d parts is a point in $\mathbb{R}^d_{\geq 0}$ modulo scaling by positive factors. Data consisting of compositions is called Compositional Data (CoDA). As the name suggestions, compositions are data which contains the information of the relative abdundance of d parts which form a whole. They commonly arise in fields such as chemistry, biology, geology and survey design, where samples are taken from an object due to it being impractical or unnecessary to take measurement of the entire object. In compositions the relative proportions of the d parts contain the meaningful information rather than the actual values of the parts themselves. Below we formally define compositions and give some examples.
-Include an appendix for equivalence relations, equivalence classes and quotient sets?

Definition 1.1. Let d be a positive integer. The positive orthant of $\mathbb{R}^d$ is the set $\mathbb{R}^d_{\geq 0} := \{x \in \mathbb{R}^d \mid x_i \geq 0,\ i = 1, \ldots, d\}$. We can define as equivalence relation $\sim$ on $\mathbb{R}^d_{\geq 0}$ where for $x, x' \in \mathbb{R}^d_{\geq 0}$ we have $x \sim x'$ if and only if $x = \lambda x'$ for some $\lambda > 0$. A composition is an equivalence class which is an element of the quotient set $\mathbb{R}^d_{\geq 0} / \sim$.

1.2. Examples

Example 1.2. Include an example with unnormalised compositions.

1.3. Obstacles in treating CoDA with standard methods

In machine learning and statistics we wish to analyse data to perform tasks such as regression, classification or representation learning, however most standard methods such as linear regression, logistic regression and principal component analysis assumes that the data resides in some Euclidean space $\mathbb{R}^d$. While a composition may appear to be as such, they are actually equivalence classes with infinitely many representatives, each representative being a point in $\mathbb{R}^d$. The most naive approach to modelling CoDA would be to simply forget the equivalence class structure and replace it with the representative which was originally provided in the data. This quickly becomes problematic, since the standard methods are not robust against positive scaling of the input data as one requires when modelling CoDA. One way to address this is to define a unique representative from each class, with a natural choice being the representative whose components sum to 1.

Definition 1.3. The set $\Delta^d = \{x \in \mathbb{R}^d_{\geq 0} \mid \|x\|_1 = 1\}$ is called the probability simplex. By associating a composition (an equivalence class) with its unique representative in the probability simplex, for the rest of this report we shall refer to points in the probability simplex as compositions.

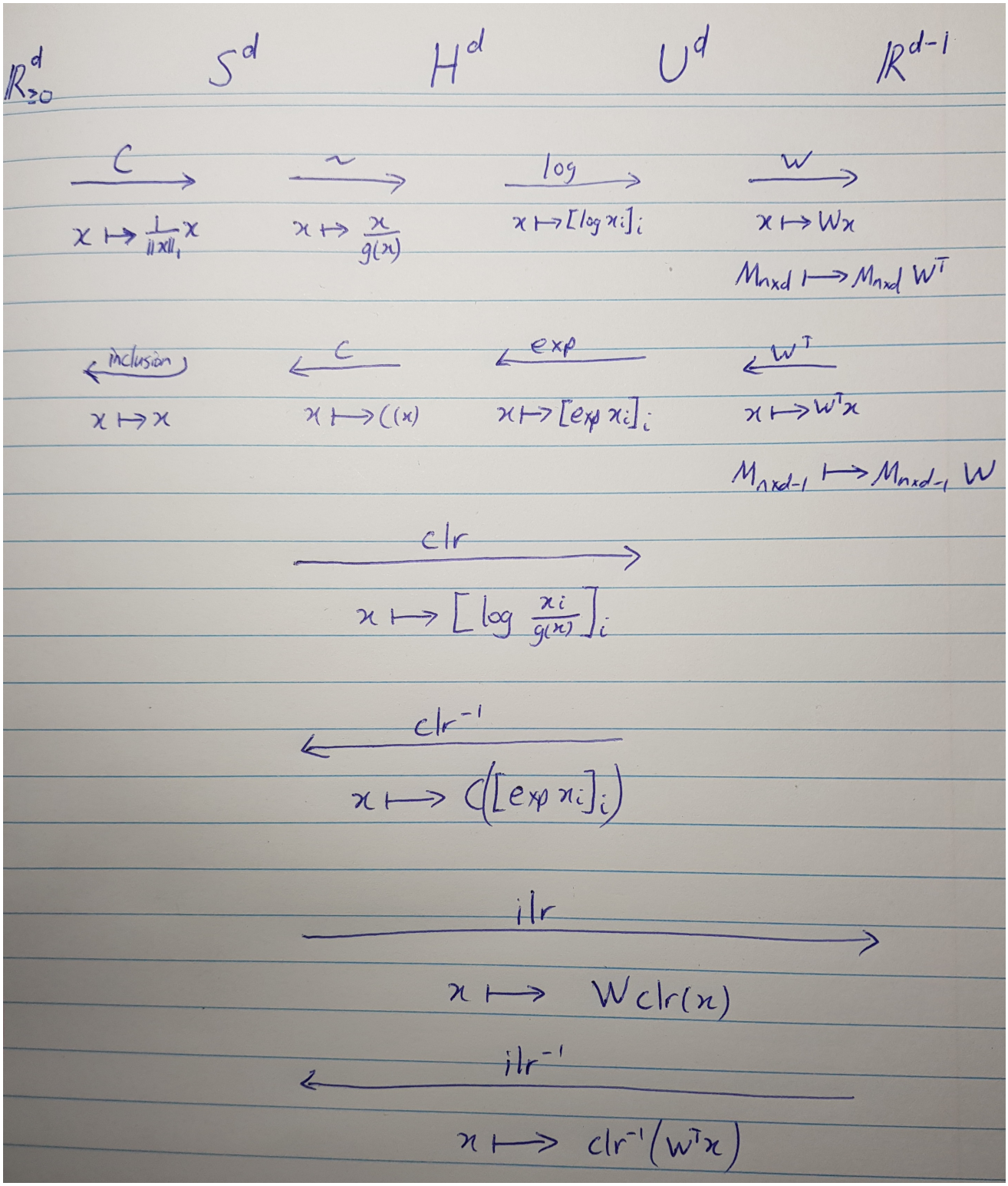
While representing compositions with points in the probability simplex resolves the issue of being robust with respect to scaling by positive factors, several issues still remain.

- The predictor functions of standard methods require us to add points together and multiply by real coefficients. If we use the usual vector space operations of $\mathbb{R}^d$ on compositions, the result will generally not be a composition (it will not lie on the probability simplex). Normalising the result is not an adequate solution, as e.g. multiplication by real coefficients reduces to the identity map. We require a natural vector space structure on the probablity simplex to form similar predictor functions to the standard ones.
- To minimise the loss functions of standard methods, we require a suitable notion of distance for compositions. While Δ^d inherits the Euclidean metric from $\mathbb{R}^d$, it is unclear whether this is most appropriate.
- The probability simplex lies in a $d-1$ dimensional subspace of $\mathbb{R}^d$. This poses issues in models such as linear regression and PCA which requires the data matrix to have full rank.
- Vapnik's Principle - "When solving a problem of interest, do not solve a more general problem as an intermediate step" [Cite]. A corollary of this is to always use as much structure available in a (machine learning/statistics) problem as possible. Thus, we expect that models which specifically address the special structure of compositions will outperform models which were designed for the more general space $\mathbb{R}^d$.

1.4. Transformations for CoDA

We now present a set of coordinate systems and maps between them which arise frequently and prove to be useful when attempting to circumvent the issues described above. In the following we pose the additional assumption that all compositions have no zero valued components. [Literature that naturally addresses zeros] [Literature that addresses zeros with ad-hoc approach].

Definition 1.4. Define $S^d := \{x \in \mathbb{R}^d \mid \|x\|_1 = 1 \text{ and } x_i > 0, i = 1, \dots, d\}$, i.e. S^d is the interior (wrt $\mathbb{R}^d$) of the probability simplex. Define the clr-plane as $U^d = \{x \in \mathbb{R}^d \mid \sum x_i = 0\}$. There is a bijection $clr : S^d \to U^d$ given by $[x_i] \mapsto [\log \frac{x_i}{g(x)}]$ where $g(x)$ is the geometric mean of the components of x. Let $W \in \mathbb{R}^{(d-1)\times d}$ be such that $WW^T = I_{d-1}$ and $W^TW = I_d - \frac{1}{d}\mathbf{1}_{d\times d}$. Then premultiplication by W gives a bijection $U^d \to \mathbb{R}^{d-1}$ (with the inverse map being premultiplication by W^T). Composing this with the clr map gives a bijection $ilr : S^d \to \mathbb{R}^{d-1}$. Define $S^d_n \subset \mathbb{R}^{n\times d}$ as the set of matrices with n rows, each row being a composition. We extend clr and ilr to such matrices by applying the functions row-wise.
Through the ilr map, S^d inherits a Hilbert space structure from $\mathbb{R}^{d-1}$. When the set S^d is equipped with this Hilbert space structure it is called the Aitchison simplex. We shall denote addition and multiplication in the Aitchison simplex by $\oplus$ and $\otimes$ respectively. The clr and ilr maps are trivially isometric isomorphisms of S^d. A summary of these coordinate systems and maps between them is given in the diagram below. [Latex the Diagram, Mention the alr transform in a footnote?]



1.5. Hilbert Space Structure of the Aitchison Simplex

Explicit forms of addition, scalar multiplication, inner product, norm, distance.

Points in S^d are subject to an equality condition and d inequality conditions. The clr map reduces this to a single equality condition, and the ilr map further reduces this to no conditions at all. Thus, one approach for machine learning on CoDA is to transport the data to $\mathbb{R}^{d-1}$ via the ilr map, use a standard model in that space, and then transform back to S^d. Empircally, this approach to dealing with CoDA has proved to be quite effective and has been considered as state of the art since the ilr transform was first proposed in (1984?) until quite recently. It addresses the points in the previous section about dealing specifically with CoDA and providing a vector space structure, a metric, and a full rank data matrix so that standard methods can be applied, but we still need to investigate whether other ways to model CoDA exist and on what types of CoDA such models are most effective on.

CHAPTER 2

Information Geometry of the Probability Simplex

[Expand background on statistical manifolds, FIM, KL-divergence, distance on Riemannian manifolds as geodesic distance...] Each point in the probability simplex Δ^d corresponds naturally to a discrete probability distribution over d states, so Δ^d can be viewed as a $d-1$ dimensional statistical manifold. A natural Riemannian metric for statistical manifolds is the Fisher information metric. The Fisher-Rao distance between two points on a statistical manifold as the distance between those points on the Riemannian manifold with metric given by the FIM.

Proposition 2.1. The Fisher-Rao distance between $x, y \in \Delta^d$ is given by

$$d_{FR}(x,y) = \arccos\left(\sum_{i=1}^{d}\sqrt{x_i y_i}\right).$$

Proof. □

Proposition 2.2. As $p \to q$ in Δ^d, we have $D_{KL}(p,q) \sim 2d_{FR}^2(p,q)$. Thus, the KL divergence provides a local approximation to the Fisher-Rao distance.

Proof. □

Remark 2.3. As $p \to e_i$ and $q \to e_j$, $i \neq j$, the KL divergence $D_{KL}(p,q)$ tends to infinity. Under the same conditions, $2d_{FR}^2(p,q)$ approaches a finite value.

Remark 2.4. Suppose we have two models which approximates a composition $y \in \Delta^d$ with the same parameterised function $f_\beta(x_1,\ldots,x_m)$ and seek to find the $\beta \in \mathbb{R}^m$ which minimizes a loss function of the form $\ell = \sum_{j=1}^{m} d(y_j, f_\beta(x_{j1},\ldots,x_{jm}))$, with the only difference between the two models being the distance measures d_1 and d_2.

Intuitively, we could expose a difference between the two models by using a dataset such that the labels are near points p in the simplex Δ^d for which the difference between $d_1(p,q)$ and $d_2(p,q)$ (where q is a point near p) is large compared to other areas of the simplex. Specifically (See Section 3.3 in the next chapter) in the case where d_1 is the KL divergence and d_2 is the Aitchison distance (i.e. the distance on Δ^d induced by the Euclidean distance on clr coordinates) , are such points related to points for which the Fisher Information Metric under clr coordinates is most dissimilar to the identity matrix? (And what is the correct notion of dissimilar?)
As an example, take $x \in \Delta^d$, $x = (1/d,\ldots,1/d)$. Then $clr(x) = (0,\ldots,0)$ and the FIM under clr coordinates is given by $\frac{1}{d}I_d - \frac{1}{d^2}\mathbf{1}_{d\times d}$. What does this indicate exactly?
Another example - Let $x = (0.998, 0.001, 0.001) \in \Delta^3$. Then the FIM under the coordinates $\tilde{x}$ is approximately

$$\begin{bmatrix} 1.2\cdot 10^{-6} & 10^{-1} \\ 10^{-1} & 10^{-4} \end{bmatrix}$$

How should this be interpreted?

CHAPTER 3

Regression models on CoDA

3.1. Simple linear regression with CoDA feature and real prediction

In (Combettes, Muller, Regression models for compositional data, 2019) they consider the case of predicting a real value with the independent variable being a single composition. In cases where the predicted values are real numbers, we can map the predictions onto S^2 by the inverse-ilr map so that we may consider an alternate loss function as implied by the previous chapter, but this is a forced unnatural connection to the Aitchison simplex and we do not expect improved information geometry by doing this.

3.2. Multiple linear regression with real features and CoDA prediction

To predict one composition z from m independent real variables x_k, we can formulate the model

$$z = b_0 \oplus (\oplus_k x_k \otimes b_k)$$

and solve for the compositions b_k. This is referred to as "Model 2" in (H. Wang, 2013). Similarly to the previous section, we can consider a new loss function which may give better information geometry.

3.3. Multiple linear regression for CoDA features and CoDA prediction

In this section we focus on models which predict an independent composition variable with several dependent composition variables.

3.3.1. clr-LR by Wang et al. In (H. Wang et al, Multiple linear regression modeling for compositional data, 2013) the authors wish to model a dependent composition variable as a linear combination (in the Aitchison geometry) of k independent composition variables, and learn the m coefficients $\beta_1, \ldots, \beta_m \in \mathbb{R}$ which make this estimator minimize the sum of the Aitchison distances between the labelled values and the predictions.

More specifically, their model (which we call clr-LR) has the following form. Suppose we have labels V and features $U^{(k)}, k = 1, \ldots, m$ in S_n^d, assumed to be centralized. Their model is given by

$$\hat{V} = \oplus_{k=1}^m \beta_k \otimes U^{(k)}$$

Let $Y = ilr(V), \hat{Y} = ilr(\hat{V}), X^{(k)} = ilr(U^{(k)})$. Since ilr is an isometric isomorphism, we have

$$\hat{Y} = \sum_{k=1}^m \beta_k X^{(k)}.$$

To find $\beta \in \mathbb{R}^m$, the authors minimize the Frobenius norm $\|Y - \hat{Y}\|_F$. [Derive closed form solution]. Following (Section 4.2 of) (M Avalos-Fernandez et al, Representation Learning of Compositional Data, 2018), we expect that the model may have better information geometry if we consider minimizing the following loss instead:

$$l_{CoDA} := D_{exp}\left(\sum_k \beta_k clr(U^{(k)}), clr(V)\right)$$

$$= (\mathbf{1}_{n\times 1})^T \exp\left((\sum_i \beta_i X^{(i)})W\right)\mathbf{1}_{d\times 1} - \text{trace}(\tilde{V}^T(\sum_i \beta_i X^{(i)})W)$$

Let g and h denote the first and second term in the gauged-KL-loss respectively.

$$g(\beta) = \sum_{i=1}^{d}\sum_{j=1}^{n} \exp\left(\beta_1 C_{ij}^{(1)} + \ldots + \beta_k C_{ij}^{(k)}\right)$$

$$h(\beta) = \text{trace}\left(\tilde{V}^T\left(\sum_k \beta_k C^{(k)}\right)\right) = \sum_k \beta_k \,\text{trace}\left(\tilde{V}^T C^{(k)}\right)$$

We have

$$\frac{\partial g}{\partial \beta_r} = \sum_{i=1}^{d}\sum_{j=1}^{n} C_{ij}^{(r)} \exp\left(\beta_1 C_{ij}^{(1)} + \ldots + \beta_k C_{ij}^{(k)}\right)$$

$$\frac{\partial h}{\partial \beta_r} = \text{trace}\left(\tilde{V}^T C^{(r)}\right)$$

This loss function is convex in β and we can compute the gradient, so we can find the optimal β by standard methods. In our experiments we used Scipy's BFGS optimizer and evaluated the results against metrics on Δ^d such as Fisher-Rao distance, symmetric KL distance, L1 distance and L2 distance.

On a variety of datasets these two loss functions produce quite similar results. CoDA-LR usually has 1 to 2 percent higher error than clr-LR, although there are instances where CoDA-LR has lower error than clr-LR. The similarity between the results is perhaps unsurprising given the relatively small number of learnable paramaters (m) in clr-LR.

3.3.2. Learning a bias composition. In (Wang. et al) the authors wished to learn only the real coefficients of a linear estimator. By doing this, they maintained a strong parallel with linear regression on real valued data and were able to produce a simple closed form solution for their optimal coefficients. However, since there is no appropriate identity composition they can not learn a bias parameter $\beta_0 \in \mathbb{R}$. To address this issue they centralized their matrices of compositional data, and during inference they would uncentralize the output of the linear estimator by adding the center of the training set.

Instead of estimating a bias term by the center of the training labels, we could learn a bias term from the data in the following way. Again, suppose we have compositional datasets $V, U^{(k)}$ as defined above and some loss function $l(v, v')$. Consider the problem of finding m real numbers $\beta_1, \ldots, \beta_m \in \mathbb{R}$ and a composition $\boldsymbol{\beta}_0 = [\beta_{0,1}, \ldots, \beta_{0,d}] \in S^d$ such that the estimator

$$\hat{v} = \boldsymbol{\beta}_0 \oplus \beta_1 \otimes u_1 \oplus \ldots \oplus \beta_m \otimes u_m$$

minimizes the total loss $\sum_{i=1}^{n} l(v, \hat{v})$. Through the ilr transformation one could view this model as having $m + d - 1$ learnable real parameters:

$$\hat{y} = ilr(\boldsymbol{\beta}_0) + \beta_1 x_1 + \ldots + \beta_m x_m$$

Again, such a model may not have enough parameters to model compositional datasets with many parts (large d).

3.3.3. Matrix Coefficient Model. In the previous two sections the form of the estimators had simple forms in the ilr domain with ilr-transformed compositional data $Y, X^{(k)}$, and optimization was most easily performed in the ilr domain. Pursuing this to its most general form, we could consider the estimator:

$$\hat{y} = \boldsymbol{\beta}_0 + x_1\boldsymbol{\beta}_1 + \ldots + x_m\boldsymbol{\beta}_m$$

where $\boldsymbol{\beta}_0$ is a row vector of length $d-1$, and $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_m$ are $(d-1) \times (d-1)$ matrices.
This estimator has $m(d-1)^2 + d - 1$ learnable paramaters. While this may be able to express complex relations between compositional variables, most compositional datasets do not have enough samples to learn these parameters with good generalisation. As a compromise, one may consider the special case where $\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_m$ are diagonal matrices. In that case there are $(m+1)(d-1)$ learnable parameters.
A drawback to this model is that it has no simple formulation in the original domain of the Aitchison simplex, and the learnable parameters become very difficult to interpret.

Appendix

.

References

[1] M. AVALOS-FERNANDEX, R. NOCK, C.S. ONG, J. ROUAR, K. SUN, "Representation Learning of Compositional data", NIPS'18 Proceedings of the 30th International Conference on Neural Information Processing Systems, p. 19-27, 2016.

[2] R. NOCK, A. MENON AND C.S. ONG, "A scaled Bregman theorem with applications", Advances in Neural Information Processing Systems 31 (NIPS 2018)

[3] H. WANG, L. SHANGGUAN, J. WU, R. GUAN, "Multiple linear regression modeling for compositional data", J. Neurocomputing, Volume 122, p. 490-500, 2013.