

UC Santa Barbara

UC Santa Barbara Electronic Theses and Dissertations

Title

Contributions to Directional Statistics Based Clustering Methods

Permalink

<https://escholarship.org/uc/item/46j0814p>

Author

Wainwright, Brian

Publication Date

2019

Peer reviewed|Thesis/dissertation

University of California
Santa Barbara

**Contributions to Directional Statistics Based Clustering
Methods**

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Statistics & Applied Probability

by

Brian Dru Wainwright

Committee in charge:

Professor S. Rao Jammalamadaka, Chair
Professor Alexander Petersen
Professor John Hsu

September 2019

The Dissertation of Brian Dru Wainwright is approved.

Professor Alexander Petersen

Professor John Hsu

Professor S. Rao Jammalamadaka, Committee Chair

June 2019

Contributions to Directional Statistics Based Clustering Methods

Copyright © 2019

by

Brian Dru Wainwright

Dedicated to my loving wife, Carmen Rhodes, without whom none of this would have been possible, and to my sons, Max, Gus, and Judah, who have yet to know a time when their dad didn't need to work from early till late. And finally, to my mother, Judith Moyer, without your tireless love and support from the beginning, I quite literally wouldn't be here today.

Acknowledgements

I would like to offer my humble and grateful acknowledgement to all of the wonderful collaborators I have had the fortune to work with during my graduate education. Much of the impetus for the ideas presented in this dissertation were derived from our work together. In particular, I would like to thank Professor György Terdik, University of Debrecen, Faculty of Informatics, Department of Information Technology. I would also like to thank Professor Saumyadipta Pyne, School of Public Health, University of Pittsburgh, and Mr. Hasnat Ali, L.V. Prasad Eye Institute, Hyderabad, India. I would like to extend a special thank you to Dr Alexander Petersen, who has held the dual role of serving on my Doctoral supervisory committee as well as wearing the hat of collaborator. I would like to also acknowledge the work of Ms. Qianyu Jin, my collaborator who co-developed many of the components presented in Chapter 2, as part of a joint endeavor.

I would also like to acknowledge Mr. Michael Kawai, for giving me the fundamental tools of problem solving, regardless of the question and to Professor Joshua French, for inspiring me to pursue my passion for statistics.

Finally, I want to acknowledge the tremendous support of my advisor and mentor, Distinguished Professor S. Rao Jammalamadaka. Your counsel, wisdom, and guidance over these years has been a constant and unwavering source of encouragement.

Curriculum Vitæ

Brian Dru Wainwright

Education

- 2019 Ph.D. in Statistics and Applied Probability, University of California, Santa Barbara.
- 2016 M.A. in Statistics, University of California, Santa Barbara.
- 2012 B.S. Mathematics: Applied Mathematics Focus, University of Colorado, Denver

Research

- 2016-2019 Contributions Towards Directional Statistics Based Clustering Models
- 2017-2019 A Functional Clustering Approach to Directional Data
- 2016-2018 3D-Directional/Spherical Statistics, Simulation, and Visualization, Technical Report

Academic Experience

- 2014-2019 Teaching Assistant, Department of Statistics & Applied Probability, University of California, Santa Barbara
- 2015-2018 Head/Lead Teaching Assistant, Department of Statistics & Applied Probability, University of California, Santa Barbara
- 2017-2018 Bridge to Doctorate Mentor, Center for Science and Engineering Partnerships, California NanoSystems Institute, University of California, Santa Barbara
- 2015 Teaching Associate, Department of Statistics & Applied Probability, University of California, Santa Barbara
- 2010-2011 Lab Assistant, Math Educational Resource Center, University of Colorado, Denver

Professional Experience

- 2017 Data Scientist Intern, Dia & Co., Santa Monica, CA
- 2011 Collision Probability Analyst, SpaceNav, Denver, CO

Abstract

Contributions to Directional Statistics Based Clustering Methods

by

Brian Dru Wainwright

Statistical tools like the finite mixture models and model-based clustering have been used extensively in many fields such as natural language processing and genomic research to investigate everything from copyright infringement to unraveling the mysteries of the evolutionary process. In model-based clustering, the samples are assumed to be realizations of a mixture distribution consisting of one or more mixture components, and the model attempts to discern what this original model is, given the observed data. In our investigation we explore directional distributions on the circle, the sphere, and the hypersphere, where the component distributions are themselves respectively the von Mises distributions in 2-dimensions, the von Mises-Fisher distributions in 3-dimensions, and p -dimensional von Mises-Fisher distributions for large p . In each case, the observations lie on the circle, the unit-sphere, or the hypersphere \mathbb{S}^{p-1} embedded in \mathbb{R}^p , due to the inherent structure of the data, or by normalizing the curves. We look specifically at clustering curves around the unit circle \mathbb{S}^1 , treating them first as mixture distributions, and in an alternate approach, as functional data that can be explored via their Fourier coefficients. We also investigate clustering high-dimensional, extremely sparse textual data, by looking at twitter data from the day of the 2016 United States presidential election as document vectors on the unit hypersphere. Finally, we introduce and discuss a broad family of spherical distributions that we call the “Generalized Fisher-Bingham family,” and present details of a software package that we developed to simulate and visualize members of this family.

Contents

| | |
|--------------------------------------------------------------------------------------------|------------|
| Curriculum Vitae | vi |
| Abstract | vii |
| 1 Introduction | 1 |
| 1.1 Overview of Directional Statistics | 1 |
| 1.2 What to expect | 2 |
| 2 Clustering Curves around the Unit Circle | 4 |
| 2.1 Introduction | 4 |
| 2.2 Measures of distance and divergence between two vM models | 5 |
| 2.3 L_2 distance and the SKL divergence for Mixtures | 14 |
| 2.4 Clustering curves around the circle — a Simulation study | 15 |
| 2.5 A real data Application to Neuro Retinal Rim (NRR) area | 23 |
| 3 Functional Clustering of Circular Densities via Fourier coefficients | 28 |
| 3.1 Functional Representation | 28 |
| 3.2 Tuning Parameter Selection | 30 |
| 3.3 Functional and non-functional Mahalanobis Clustering | 31 |
| 3.4 Real Data Example with NRR curves | 32 |
| 4 Clustering Textual Data with Directional Statistics | 37 |
| 4.1 Introduction | 37 |
| 4.2 Textual Data: Definitions and Notation | 37 |
| 4.3 Clustering on the hypersphere with Mixture of von Mises-Fisher Distributions | 40 |
| 4.4 Measures of distance and divergence between two vMF models | 46 |
| 4.5 Data: Into the Expanse of the Twitter world | 51 |
| 5 Simulation and Visualization of Spherical Distributions | 58 |
| 5.1 Introduction | 58 |
| 5.2 GFB family of distributions | 60 |
| 5.3 A Spherical Histogram | 74 |

| | | |
|----------|-------------------------------------------------------|------------|
| 5.4 | Simulation of Random Variates on the Sphere | 75 |
| 6 | Conclusions and Future Directions | 93 |
| 6.1 | Conclusions | 93 |
| 6.2 | Future Directions | 94 |
| 7 | Appendix and Supplementary Material | 96 |
| 7.1 | Chapter 4: Supplementary Material | 97 |
| 7.2 | Chapter 5: Supplementary Material | 98 |
| | Bibliography | 117 |

Chapter 1

Introduction

1.1 Overview of Directional Statistics

In many scientific disciplines, when researchers are collecting data, that data comes in the form of directions, or unit vectors, in either the plane (for 2-dimensional data), the sphere (for 3-dimensional data), or the hypersphere for dimension greater than 3. Directional statistics is the study and development of statistical theory and methodology used to analyze and draw inference from such data. It may be that an ornithologist is interested in flight directions of some species of bird, as it leaves a particular area. Or perhaps, it is the geologist who is researching the movement of the earth's magnetic poles, or the astronomer tracking heavenly bodies as they make their way across the sky. All of these investigations generate data that can be considered directional, and we need appropriate tools to extract any real meaning, as well as to quantify the uncertainty of both the observations and the conclusions.

Unlike much of the linear analogues, directional data requires special treatment due to the unique constructs and features. As an example, the 2-dimensional observation on the unit circle can be represented as a unit vector, or simply as an angle, but neither representation is necessarily unique, as both depend on the selection of some appropriate *zero* from which to measure,

as well as the orientation of rotation [1]. Similarly points on the unit sphere (or hypersphere) can be described in terms of two (or more) angles or as appropriately dimensioned unit vectors, and are equally dependent on the choice of the zero-direction and sense of rotation. All of this is to say, that much care must be taken when measuring distance between directional observations, and distance or similarity between directional distributions. In this dissertation, we will explore these ideas of distance and similarity between directional data and directional distributions, with the goal of finding effective measures to group or cluster such observations or curves defined on these spaces.

1.2 What to expect

In Chapter 2, we look at clustering curves around the unit circle. These curves are modeled as mixtures of von Mises (vM) distributions, and the measures we look at are the L_2 distance and at the standard as well as symmetric variants of the Kullback-Liebler (KL) divergence. We provide a visual comparison of the L_2 and symmetric KL-divergence in Section 2.2.3, and compare their performance by looking at simulated data in Section 2.4. Section 2.5 concludes the chapter by taking a real data example where no *ground truth* is known, and compare the clusters so obtained by looking at clustering agreement between the two methods, via various ultrametric measures of the induced hierarchies.

In Chapter 3 we look at clustering curves about the unit circle through the functional data lens. In this chapter, curves around the circle are given a functional representation, and a functional clustering algorithm named FUNfem [2] is used, after appropriate tuning parameter selection. We compare the resultant clustering corresponding to different tuning parameters, against those generated by our L_2 and symmetric KL-divergence Chapter 2, via the adjusted Rand index. We also compare the clustering hierarchies induced by implementing a functional Mahalanobis distance as well as the more traditional multivariate Mahalanobis distance, with those

obtained in the earlier Chapter 2 using distance and divergence measures.

In Chapter 4, we look at clustering high-dimensional observations on the unit hypersphere, where the data is vectorized textual data normalized to unit length. We illustrate the ideas by using Twitter data (*tweets*) collected on November 8th, 2016 regarding the US presidential election, where relevance is established by restricting to documents that satisfy a range of keywords associated with the election. We compare the L_2 and symmetric KL-divergence measures, after appropriate generalizations of the lower dimensional case.

Chapter 5 is in a slightly different spirit compared to the rest of this dissertation. It describes a broad family of spherical distributions, called the “Generalized Fisher-Bingham” family, and develops a MATLAB package *3D-Simulation-Visualization* [3], that helps in the simulation and visualization of these spherical distributions. This package is in support of the theoretical work [4]. In this chapter, we give the details, algorithms, and examples for simulating and visualizing spherical distributions from this Generalized Fisher-Bingham family of distributions. Also highlighted are new and improved methods for simulation and visualization of spherical data and models, which includes a 3-dimensional analogue of the classical rose diagram for circular data.

Chapter 2

Clustering Curves around the Unit Circle

2.1 Introduction

The von Mises (vM) distribution, also known as the Circular Normal (CN) distribution, is one of the most commonly used models for circular data, and has the density function,

$$f(\alpha; \mu, \kappa) = \frac{e^{\kappa \cos(\alpha - \mu)}}{2\pi I_0(\kappa)}, \quad 0 \leq \alpha < 2\pi$$

where $I_p(\kappa)$ is the modified Bessel function of the first kind, and order p , and $0 \leq \mu < 2\pi$ and $\kappa \geq 0$ are the mean direction and concentration parameter respectively. Although this distribution is unimodal, an appropriate mixture of such distributions can be used to model curves with multiple peaks. Indeed, similar to a corresponding result on the real line which says that any probability distribution on the real line can be approximated by a countable mixture of Normal distributions (see references [5],[6]), one can use a countable mixture of vM distributions to approximate any probability distribution on the circle.

Given that each curve can be approximated by such a mixture, the task of clustering these curves becomes one of defining an appropriate “distance” or a “divergence measure” between

any two curves. To that end, as a first step, we consider such distance or divergence measures between any two vM models in the next section.

2.2 Measures of distance and divergence between two vM models

In this section, we describe two different measures to find the dissimilarity between two such vM models, one an actual L_2 distance, and the other, a measure based on symmetrized Kullback-Liebler divergence. We show later on, that either of these measures provides a good tool to cluster curves around a circle, and that they yield very similar results. We start with the following useful basic result.

Lemma 2.2.1. *For any two vM distributions $f \sim vM(\mu_1, \kappa_1)$ and $g \sim vM(\mu_2, \kappa_2)$,*

$$\int_0^{2\pi} f(\alpha)g(\alpha)d\alpha = \frac{I_0(\kappa)}{2\pi I_0(\kappa_1)I_0(\kappa_2)} \quad (2.1)$$

where

$$\kappa = \sqrt{\kappa_1^2 + \kappa_2^2 + 2\kappa_1\kappa_2 \cos(\mu_1 - \mu_2)} \quad (2.2)$$

Proof:

$$\begin{aligned} \int_0^{2\pi} f(\alpha)g(\alpha)d\alpha &= \frac{1}{2\pi I_0(\kappa_1)I_0(\kappa_2)} \int_0^{2\pi} e^{\kappa_1 \cos(\alpha-\mu_1) + \kappa_2 \cos(\alpha-\mu_2)} d\alpha \\ &= \frac{1}{2\pi I_0(\kappa_1)I_0(\kappa_2)} \int_0^{2\pi} e^{(\kappa_1 \cos \mu_1 + \kappa_2 \cos \mu_2) \cos \alpha + (\kappa_1 \sin \mu_1 + \kappa_2 \sin \mu_2) \sin \alpha} d\alpha \\ &= \frac{1}{2\pi I_0(\kappa_1)I_0(\kappa_2)} \int_0^{2\pi} e^{A \cos \alpha + B \sin \alpha} d\alpha \end{aligned}$$

where

$$A = \kappa_1 \cos \mu_1 + \kappa_2 \cos \mu_2$$

$$B = \kappa_1 \sin \mu_1 + \kappa_2 \sin \mu_2.$$

Rewriting $A = \kappa \cos \mu$, $B = \kappa \sin \mu$, where

$$\mu = \arctan \frac{B}{A}$$

and

$$\begin{aligned} \kappa^2 &= A^2 + B^2 \\ &= \kappa_1^2 \cos^2 \mu_1 + \kappa_2^2 \cos^2 \mu_2 + 2\kappa_1 \kappa_2 \cos \mu_1 \cos \mu_2 \\ &\quad + \kappa_1^2 \sin^2 \mu_1 + \kappa_2^2 \sin^2 \mu_2 + 2\kappa_1 \kappa_2 \sin \mu_1 \sin \mu_2 \\ &= \kappa_1^2 + \kappa_2^2 + 2\kappa_1 \kappa_2 \cos(\mu_1 - \mu_2), \end{aligned}$$

we have

$$\begin{aligned} \int_0^{2\pi} f(\alpha)g(\alpha)d\alpha &= \frac{1}{2\pi I_0(\kappa_1)I_0(\kappa_2)} \int_0^{2\pi} e^{\kappa \cos \mu \cos \alpha + \kappa \sin \mu \sin \alpha} d\alpha \\ &= \frac{1}{2\pi I_0(\kappa_1)I_0(\kappa_2)} \int_0^{2\pi} e^{\kappa \cos(\alpha - \mu)} d\alpha \\ &= \frac{I_0(\kappa)}{2\pi I_0(\kappa_1)I_0(\kappa_2)} \end{aligned}$$

proving the result.

2.2.1 L_2 distance between two vM models

Using Lemma 2.2.1, the L_2 distance between any two vM distributions is obtained in the following,

Proposition 2.2.1. *For any two vM distributions $f \sim vM(\mu_1, \kappa_1)$ and $g \sim vM(\mu_2, \kappa_2)$, the L_2 distance is given by*

$$L_2(f, g) = \frac{1}{2\pi} \left(\frac{I_0(2\kappa_1)}{I_0(\kappa_1)^2} + \frac{I_0(2\kappa_2)}{I_0(\kappa_2)^2} - \frac{2I_0(\kappa)}{I_0(\kappa_1)I_0(\kappa_2)} \right) \quad (2.3)$$

where κ is defined as in Equation 2.2.

Proof: By Lemma 2.2.1,

$$\begin{aligned} \int_0^{2\pi} f^2(\alpha) d\alpha &= \frac{I_0(\sqrt{\kappa_1^2 + \kappa_1^2 + 2\kappa_1\kappa_1 \cos(\mu_1 - \mu_1)})}{2\pi I_0(\kappa_1)I_0(\kappa_1)} \\ &= \frac{I_0(2\kappa_1)}{2\pi(I_0(\kappa_1))^2}, \end{aligned}$$

and similarly

$$\int_0^{2\pi} g^2(\alpha) d\alpha = \frac{I_0(2\kappa_2)}{2\pi(I_0(\kappa_2))^2}.$$

Thus,

$$\begin{aligned} L_2(f, g) &= \int_0^{2\pi} (f(\alpha) - g(\alpha))^2 d\alpha \\ &= \int_0^{2\pi} f^2(\alpha) d\alpha + \int_0^{2\pi} g^2(\alpha) d\alpha - 2 \int_0^{2\pi} f(\alpha)g(\alpha) d\alpha \\ &= \frac{1}{2\pi} \left(\frac{I_0(2\kappa_1)}{(I_0(\kappa_1))^2} + \frac{I_0(2\kappa_2)}{(I_0(\kappa_2))^2} - \frac{2I_0(\kappa)}{I_0(\kappa_1)I_0(\kappa_2)} \right) \end{aligned}$$

giving us the desired result.

2.2.2 Kullback-Liebler (KL) divergence between two vM models and its symmetric version

Kullback-Liebler (KL) divergence between two vM models

Before considering the symmetrized version of KL divergence, we first look at the closed form solution to the standard KL divergence between any two vM models.

Proposition 2.2.2. *For any two vM distributions $f \sim vM(\mu_1, \kappa_1)$ and $g \sim vM(\mu_2, \kappa_2)$, the Kullback-Liebler divergence measure is given by,*

$$KL(f, g) = \log I_0(\kappa_2) - \log I_0(\kappa_1) + \kappa_1 A(\kappa_1) - \kappa_2 \cos(\mu_1 - \mu_2) A(\kappa_1)$$

where $A(\kappa)$ is the ratio of modified Bessel functions of the first kind, given by $A(\kappa) = I_1(\kappa)/I_0(\kappa)$.

Proof: Let $f(\alpha; \mu_1, \kappa_1)$ and $g(\alpha; \mu_2, \kappa_2)$ be two independent vM distributions. Then the KL divergence between f and g is given by,

$$\begin{aligned} KL(f, g) &= \int_0^{2\pi} \log \left(\frac{f(\alpha)}{g(\alpha)} \right) f(\alpha) d\alpha \\ &= \int_0^{2\pi} \left(\log (I_0(\kappa_2)) - \log (I_0(\kappa_1)) + \kappa_1 \cos(\alpha - \mu_1) - \kappa_2 \cos(\alpha - \mu_2) \right) \frac{e^{\kappa_1 \cos(\alpha - \mu_1)}}{2\pi I_0(\kappa_1)} d\alpha \\ &= \log (I_0(\kappa_2)) - \log (I_0(\kappa_1)) + \kappa_1 \frac{I_1(\kappa_1)}{I_0(\kappa_1)} - \kappa_2 \int_0^{2\pi} \cos(\alpha - \mu_2) \frac{e^{\kappa_1 \cos(\alpha - \mu_1)}}{2\pi I_0(\kappa_1)} d\alpha. \end{aligned}$$

Writing the integrand in the last term as,

$$\begin{aligned} \cos(\alpha - \mu_2) &= \cos(\alpha - \mu_1 + \mu_1 - \mu_2) \\ &= \cos(\alpha - \mu_1) \cos(\mu_1 - \mu_2) - \sin(\alpha - \mu_1) \sin(\mu_1 - \mu_2) \end{aligned}$$

and using the relations (see e.g. Jammalamadaka and SenGupta (2003), p.36),

$$\frac{1}{2\pi} \int_0^{2\pi} \cos(p\theta) \exp(\kappa \cos \theta) d\theta = I_p(\kappa) \quad \text{and} \quad \frac{1}{2\pi} \int_0^{2\pi} \sin(n\theta) \exp(\kappa \cos \theta) d\theta = 0,$$

the last term gives us

$$\begin{aligned} & -\kappa_2 \int_0^{2\pi} \cos(\alpha - \mu_2) \frac{e^{\kappa_1 \cos(\alpha - \mu_1)}}{2\pi I_0(\kappa_1)} d\alpha \\ &= -\kappa_2 \int_0^{2\pi} [\cos(\mu_1 - \mu_2) \cos(\alpha - \mu_1) - \sin(\mu_1 - \mu_2) \sin(\alpha - \mu_1)] \frac{e^{\kappa_1 \cos(\alpha - \mu_1)}}{2\pi I_0(\kappa_1)} d\alpha \\ &= -\kappa_2 \cos(\mu_1 - \mu_2) \int_0^{2\pi} \cos(\alpha - \mu_1) \frac{e^{\kappa_1 \cos(\alpha - \mu_1)}}{2\pi I_0(\kappa_1)} d\alpha \\ &\quad + \kappa_2 \sin(\mu_1 - \mu_2) \int_0^{2\pi} \sin(\alpha - \mu_1) \frac{e^{\kappa_1 \cos(\alpha - \mu_1)}}{2\pi I_0(\kappa_1)} d\alpha \\ &= -\kappa_2 \cos(\mu_1 - \mu_2) \frac{I_1(\kappa_1)}{I_0(\kappa_1)} \end{aligned}$$

giving us the desired result.

Remark 1. Note that when $\kappa_1 = \kappa_2$, say a common κ , the D_{KL} divergence between two vM distributions becomes symmetric, and reduces to

$$\kappa A(\kappa) (1 - \cos(\mu_1 - \mu_2))$$

where $A(\kappa) = I_1(\kappa)/I_0(\kappa)$.

This behaviour can be seen graphically below in Figure 2.1.

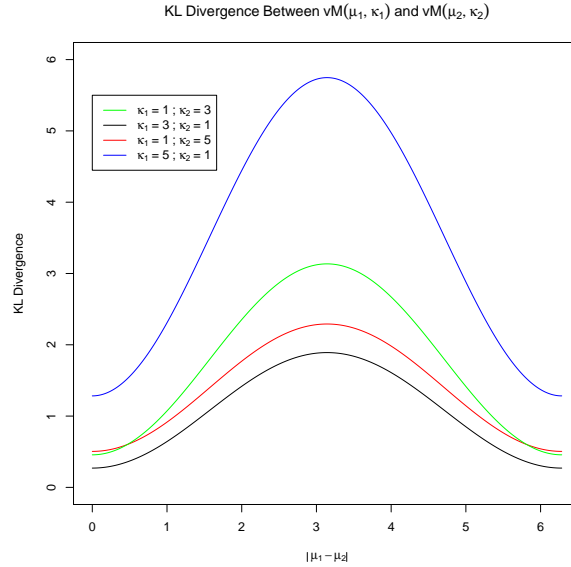


Figure 2.1: KL-Div. vs $|\mu_1 - \mu_2|$, for fixed κ_1, κ_2 .

A Symmetrized Kullback-Liebler distance between two vM models

Although this KL divergence becomes symmetric i.e. $\text{KL}(f, g) = \text{KL}(g, f)$ when the concentration parameters κ_1 and κ_2 are the same, in general it is well known that the KL measure lacks symmetry of this kind.

We will consider a simple symmetric version of the KL divergence namely

$$\text{SKL}(f, g) = D_{\text{KL}}(f, g) + D_{\text{KL}}(g, f).$$

We will refer to this as the symmetric KL-divergence or SKL. From the preceding Proposition, it is easy to check

Proposition 2.2.3. *The symmetric KL divergence SKL, between two independent vM distributions $f \sim vM(\mu_1, \kappa_1)$ and $g \sim vM(\mu_2, \kappa_2)$ is given by*

$$\text{SKL}(f, g) = \kappa_1 A(\kappa_1) + \kappa_2 A(\kappa_2) - \cos(\mu_1 - \mu_2) (\kappa_2 A(\kappa_1) + \kappa_1 A(\kappa_2))$$

where $A(\kappa)$ is the ratio of modified Bessel functions of the first kind, given by $A(\kappa) = I_1(\kappa)/I_0(\kappa)$.

Remark 2. *Jensen-Shannon divergence:*

One may also consider a slightly more general symmetrized version, called the Jensen-Shannon divergence given by

$$SKL_{JS}(f, g) = \frac{D_{KL}(f, \frac{f+g}{2}) + D_{KL}(g, \frac{f+g}{2})}{2}$$

noting that mixtures of vM distributions are known to be identifiable (see eg. Frazer et al (1981)). But we will stick with the simpler SKL mentioned above.

2.2.3 A visual comparison of the L_2 and SKL measures

Figures 2.2 and 2.3 compare the two measures side-by-side in terms of how they change for varying $|\mu_1 - \mu_2|$, or $\kappa_2 - \kappa_1$, or both. Although the parameters in vM models are the same, the magnitudes of the two measurements are different, with SKL nearly 10 times the numerical value of the L_2 distance. Apart from this differences in scale, they provide comparable results as judged by the similarities in the shapes of the line plots and 3D plots between the two.

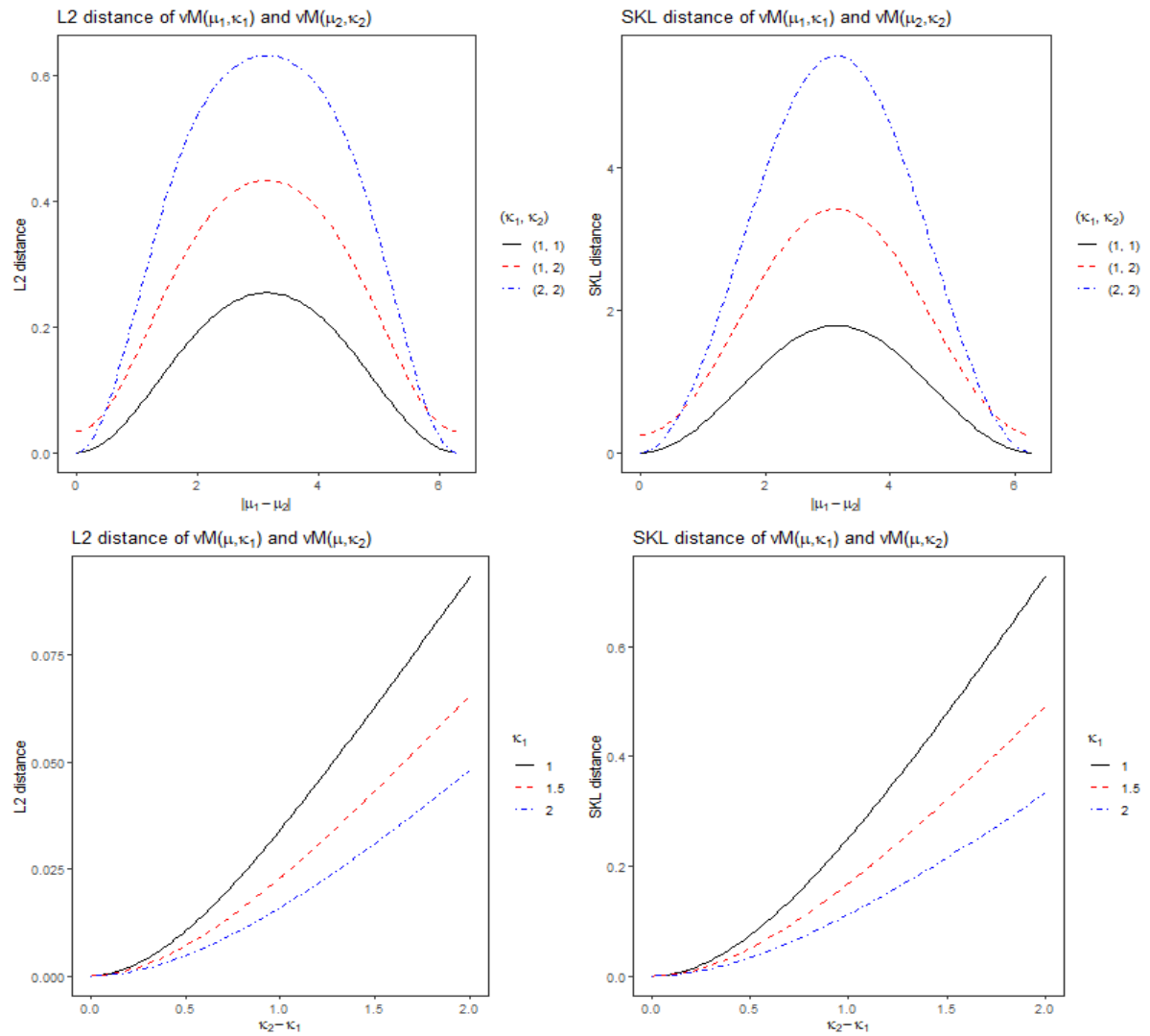


Figure 2.2: Top: L_2 distance (left) and SKL distance (right) vs $|\mu_1 - \mu_2|$, for fixed κ_1, κ_2 . Bottom: L_2 distance (left) and SKL distance (right) vs $\kappa_2 - \kappa_1$, for fixed $\mu_1 = \mu_2 = \mu$ and κ_1 . Note that the y-axis scales are different for L_2 and SKL distance.

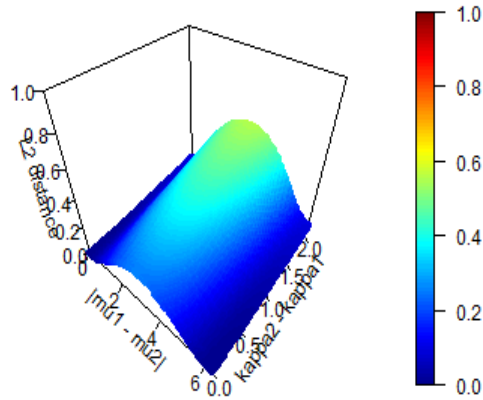
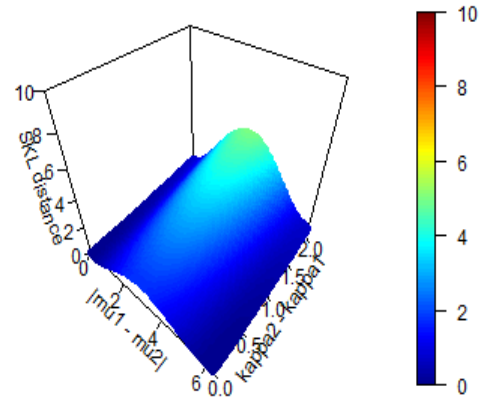
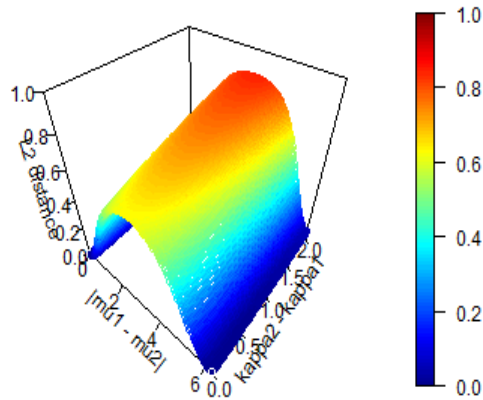
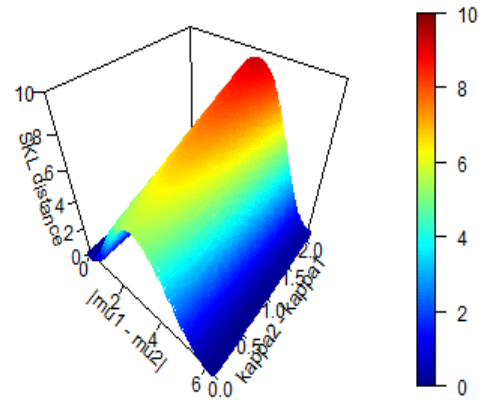
L2 distance of $vM(\mu_1, \kappa_1)$ and $vM(\mu_2, \kappa_2)$, $\kappa_1 = 1$ SKL distance of $vM(\mu_1, \kappa_1)$ and $vM(\mu_2, \kappa_2)$, $\kappa_1 = 1$ L2 distance of $vM(\mu_1, \kappa_1)$ and $vM(\mu_2, \kappa_2)$, $\kappa_1 = 2$ SKL distance of $vM(\mu_1, \kappa_1)$ and $vM(\mu_2, \kappa_2)$, $\kappa_1 = 2$ 

Figure 2.3: 3D plots for L_2 distance (left) and SKL distance (right) vs $|\mu_1 - \mu_2|$ and $\kappa_2 - \kappa_1$ for fixed κ_1 . Note that the color scales are different for L_2 and SKL distance.

2.3 L_2 distance and the SKL divergence for Mixtures

As suggested earlier, we plan to approximate any curve around the circle via vM mixtures with an appropriate number of components.

For any 2 mixtures, say h_1 and h_2 given by

$$h_1(\alpha) = \sum_{i=1}^k p_i f_i(\alpha) \quad \text{and} \quad h_2(\alpha) = \sum_{j=1}^l q_j g_j(\alpha)$$

with the usual restrictions on the mixture proportions, $\{p_i \geq 0, \sum_i p_i = 1\}$, and $\{q_j \geq 0, \sum_j q_j = 1\}$. For the L_2 distance between two such mixtures of vM distributions, we have

$$\begin{aligned} L_2(h_1, h_2) &= \int_0^{2\pi} \left(\sum_{i=1}^k p_i f_i(\alpha) - \sum_{j=1}^l q_j g_j(\alpha) \right)^2 d\alpha \\ &= \sum_{i=1}^k \sum_{i'=1}^k p_i p_{i'} \int_0^{2\pi} f_i(\alpha) f_{i'}(\alpha) d\alpha + \\ &\quad \sum_{j=1}^l \sum_{j'=1}^l q_j q_{j'} \int_0^{2\pi} g_j(\alpha) g_{j'}(\alpha) d\alpha - \\ &\quad 2 \sum_{i=1}^k \sum_{j=1}^l p_i q_j \int_0^{2\pi} f_i(\alpha) g_j(\alpha) d\alpha \end{aligned}$$

We would like to also extend our single component SKL divergence measure to mixtures of vM distributions, but must proceed carefully as no closed form expression exists, as we had for the single components, although it is calculable. To that end, will make use of numerical integration techniques to approximate the divergence,

$$\text{KL}(h_1, h_2) = \int_0^{2\pi} \log \left(\frac{h_1(\alpha)}{h_2(\alpha)} \right) h_1(\alpha) d\alpha.$$

2.4 Clustering curves around the circle — a Simulation study

The simulation procedure starts with a given vM mixture with a fixed numbers of components, and with specified parameter values for each component. Our goal is to assess how the distance/divergence measures perform in identifying the known clusters, under varying distributional conditions. The chosen parameter values are given in Table 2.1 below, and we briefly discuss the rationale behind the parameter selection case-by-case.

Each of the four designated cases consist of 3 vM mixture distributions — two 2-component mixtures and one 3-component mixture — with different parameter values. Case 1 contains vM mixtures that have equal mixture weights p , moderately high κ 's, and with differences between μ 's that are large enough such that the number of components k equals the number of modes. Cases 2, 3 and 4 are variants of Case 1. Case 2 has smaller κ 's, Case 3 has smaller μ differences such that its 2-component vM mixtures is indeed unimodal, and Case 4 has unequal p 's and thus unequal heights for each mode. These cases aim at discovering how sample-based parameter estimation and component selection results are affected by the underlying mixture distributions.

| Case | Mixture | Component | μ (rad.) | κ | p |
|----------------------------|----------|----------------------------|-------------------------|----------|-------|
| Case 1 | vM-mix 1 | $f_1(\alpha; \mu, \kappa)$ | 0 | 4.0 | 0.5 |
| | | $f_2(\alpha; \mu, \kappa)$ | $2\pi/3 (\approx 2.09)$ | 4.0 | 0.5 |
| | vM-mix 2 | $f_1(\alpha; \mu, \kappa)$ | $\pi (\approx 3.14)$ | 3.0 | 0.5 |
| | | $f_2(\alpha; \mu, \kappa)$ | $5\pi/3 (\approx 5.24)$ | 3.0 | 0.5 |
| | vM-mix 3 | $f_1(\alpha; \mu, \kappa)$ | 0 | 5.0 | 0.333 |
| | | $f_2(\alpha; \mu, \kappa)$ | $2\pi/3 (\approx 2.09)$ | 5.0 | 0.333 |
| $f_3(\alpha; \mu, \kappa)$ | | $4\pi/3 (\approx 4.19)$ | 5.0 | 0.333 | |
| Case 2 | vM-mix 1 | $f_1(\alpha; \mu, \kappa)$ | 0 | 2.0 | 0.5 |
| | | $f_2(\alpha; \mu, \kappa)$ | $2\pi/3 (\approx 2.09)$ | 2.0 | 0.5 |
| | vM-mix 2 | $f_1(\alpha; \mu, \kappa)$ | $\pi (\approx 3.14)$ | 2.0 | 0.5 |
| | | $f_2(\alpha; \mu, \kappa)$ | $5\pi/3 (\approx 5.24)$ | 2.0 | 0.5 |
| | vM-mix 3 | $f_1(\alpha; \mu, \kappa)$ | 0 | 4.0 | 0.333 |
| | | $f_2(\alpha; \mu, \kappa)$ | $2\pi/3 (\approx 2.09)$ | 4.0 | 0.333 |
| $f_3(\alpha; \mu, \kappa)$ | | $4\pi/3 (\approx 4.19)$ | 4.0 | 0.333 | |
| Case 3 | vM-mix 1 | $f_1(\alpha; \mu, \kappa)$ | 0 | 4.0 | 0.5 |
| | | $f_2(\alpha; \mu, \kappa)$ | $\pi/3 (\approx 1.05)$ | 4.0 | 0.5 |
| | vM-mix 2 | $f_1(\alpha; \mu, \kappa)$ | $\pi (\approx 3.14)$ | 3.0 | 0.5 |
| | | $f_2(\alpha; \mu, \kappa)$ | $4\pi/3 (\approx 4.19)$ | 3.0 | 0.5 |
| | vM-mix 3 | $f_1(\alpha; \mu, \kappa)$ | 0 | 5.0 | 0.333 |
| | | $f_2(\alpha; \mu, \kappa)$ | $2\pi/3 (\approx 2.09)$ | 5.0 | 0.333 |
| $f_3(\alpha; \mu, \kappa)$ | | $4\pi/3 (\approx 4.19)$ | 5.0 | 0.333 | |
| Case 4 | vM-mix 1 | $f_1(\alpha; \mu, \kappa)$ | 0 | 4.0 | 0.75 |
| | | $f_2(\alpha; \mu, \kappa)$ | $2\pi/3 (\approx 2.09)$ | 4.0 | 0.25 |
| | vM-mix 2 | $f_1(\alpha; \mu, \kappa)$ | $\pi (\approx 3.14)$ | 3.0 | 0.25 |
| | | $f_2(\alpha; \mu, \kappa)$ | $5\pi/3 (\approx 5.24)$ | 3.0 | 0.75 |
| | vM-mix 3 | $f_1(\alpha; \mu, \kappa)$ | 0 | 5.0 | 0.20 |
| | | $f_2(\alpha; \mu, \kappa)$ | $2\pi/3 (\approx 2.09)$ | 5.0 | 0.60 |
| $f_3(\alpha; \mu, \kappa)$ | | $4\pi/3 (\approx 4.19)$ | 5.0 | 0.20 | |

Table 2.1: Parameter Values for the Simulated Data

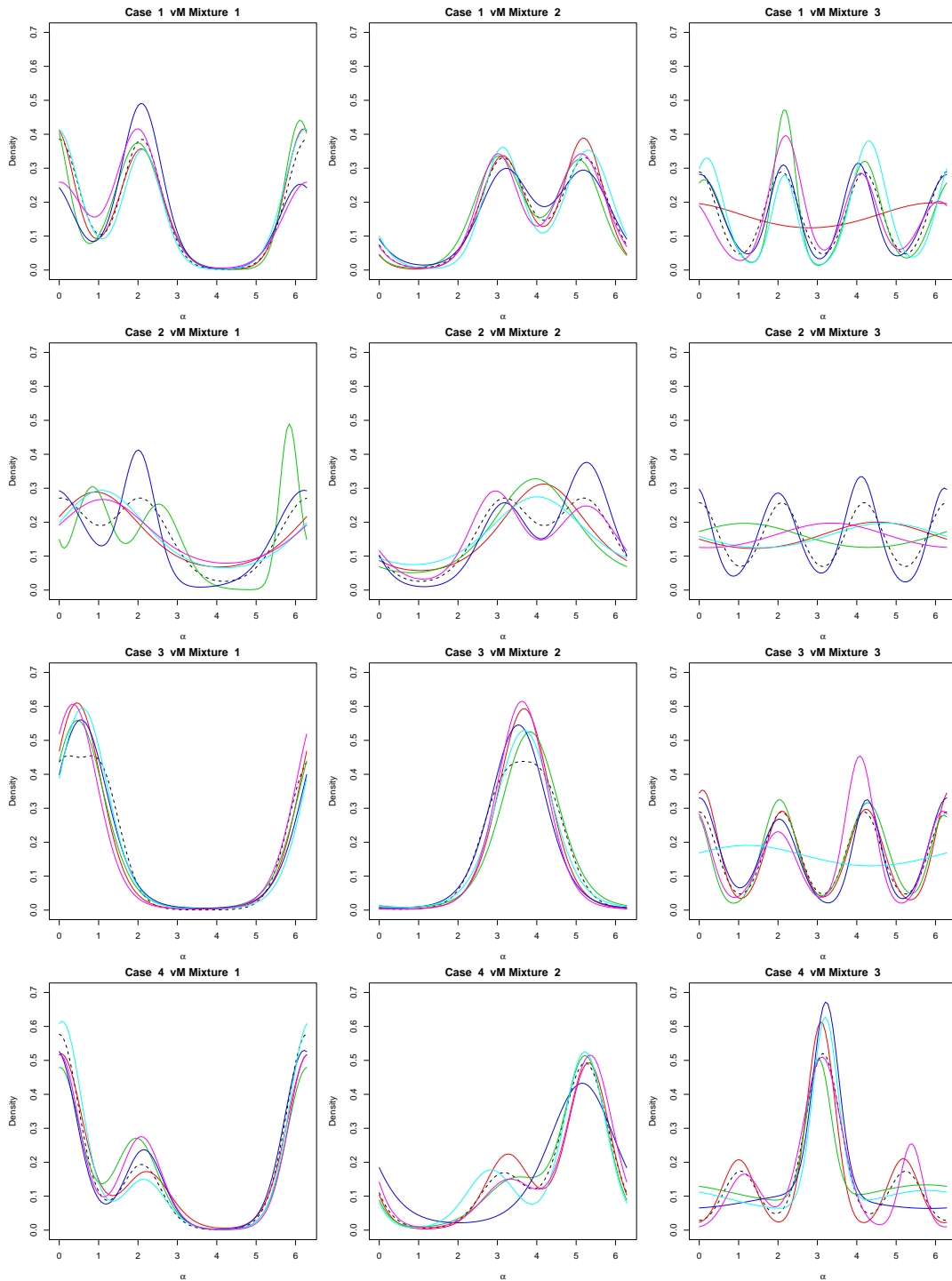


Figure 2.4: Estimated Density Curves vs. True Density Curves. The three plots in each row corresponds to the three vM mixtures in each case. Black dashed lines are the true density curves and colored solid lines are estimated density curves.

With the given set of parameter values for the mixture distribution, we take $n = 5$ samples, each of size $m = 100$ from each mixture distribution. We use the 100 observations from each sample to fit a mixture distribution, estimating the mixture model parameters via the EM algorithm proposed by Dhillon and Sra [7] and Banerjee et al. [8], and implemented by Hornik and Grün in the R statistical software package `movMF` [9].¹ Since the parameter estimation method does not automatically select the number of mixture components, we use the BIC criterion to select the number of components, say between 2, 3, 4, or 5.

So, from each of the 3 *true* mixture distributions for each case study, we generate $n = 5$ estimated mixture distributions using the samples drawn.

Figure 2.4 graphically demonstrates the efficacy of the estimated mixtures in recovering the true mixture. In the grid of plots, the rows represent the case study and the columns represent the three different vM mixture distributions within each case. In any particular plot, the weighted black dashed line is the *true* density, while the colored solid lines are the five associated density estimates. Case 1 shows good consistency between true and estimated density curves, except for the red curve in vM mixture 3 which is flat and gives only $k = 1$ component. Case 2 shows that smaller κ 's can impair this consistency, as wrong values of k are often chosen. Case 3 shows that the selected k 's match with the number of modes, if not at their true values. Nevertheless, the true and estimated density curves are consistent when it is not near the modes. Case 4 shows that the estimation procedure sometimes fails to identify those smaller modes, which correspond to the components with smaller p 's.

¹The algorithm and implementation describe fitting the parameters of von Mises-Fisher (vMF) mixture models, which is a direct extension of the vM distribution to higher dimensions, where the observations occur on the unit sphere/hypersphere. The vMF distribution, in two dimensions, reduces to the vM distribution. We will return to the vMF distribution again in a later chapter.

2.4.1 Clustering of simulated data with L_2 and SKL distances

After estimating vM mixture parameters from simulated samples for each case study, we compute the L_2 or SKL distances between each pair of samples and obtain distance matrices. Then a hierarchical clustering method is used to discover the clusters. We selected the *complete linkage* method and demonstrate the clustering results in heatmaps and dendrograms — heatmaps are to show the magnitudes of distance measures between any given pair of estimated vM mixtures, and dendrograms are to show the structures of clustering hierarchy. Ideally, what we would like to see are the clustering procedure, when stopped at 3 clusters, completely separates the estimated density curves, based on which mixture distribution their samples are generated from, i.e., each of the 3 clusters should contain the 5 densities estimated from the common true density.

Tables 2.2 and 2.3 show that both L_2 -based clustering and SKL-based clustering are able to recover the correct cluster membership for all the samples. Figure 2.5 and 2.6 display row end dendrograms and heatmaps for these two clustering metrics. For Case 1, 3, and 4, the between-cluster L_2 distances are much larger than within-cluster L_2 distances. The heatmap patterns suggest that, for arbitrary samples denoted by A, B, C and D , the distance $L_2(A, B) \approx L_2(C, D)$ as long as A, C are in the same cluster and B, D are in the same cluster. For Case 2, the heatmap pattern is more fuzzy, mainly due to the less desirable consistency between estimated and true density curves as discussed earlier. Similar behaviors are observed for the SKL distances.

| | vM-mix 1 | | | | | vM-mix 2 | | | | | vM-mix 3 | | | | |
|--------|----------|----|----|----|----|----------|----|----|----|-----|----------|-----|-----|-----|-----|
| | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | V9 | V10 | V11 | V12 | V13 | V14 | V15 |
| Case 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| Case 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| Case 3 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| Case 4 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |

Table 2.2: L_2 -based Cluster Membership. $V1, V2, \dots, V15$ designate the respective fitted mixture distribution for each particular case and 100-curve sampling block.

| | vM-mix 1 | | | | | vM-mix 2 | | | | | vM-mix 3 | | | | |
|--------|----------|----|----|----|----|----------|----|----|----|-----|----------|-----|-----|-----|-----|
| | V1 | V2 | V3 | V4 | V5 | V6 | V7 | V8 | V9 | V10 | V11 | V12 | V13 | V14 | V15 |
| Case 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| Case 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| Case 3 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |
| Case 4 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 | 3 |

Table 2.3: SKL-based Cluster Membership. $V1, V2, \dots, V15$ designate the respective fitted mixture distribution for each particular case and 100-curve sampling block.

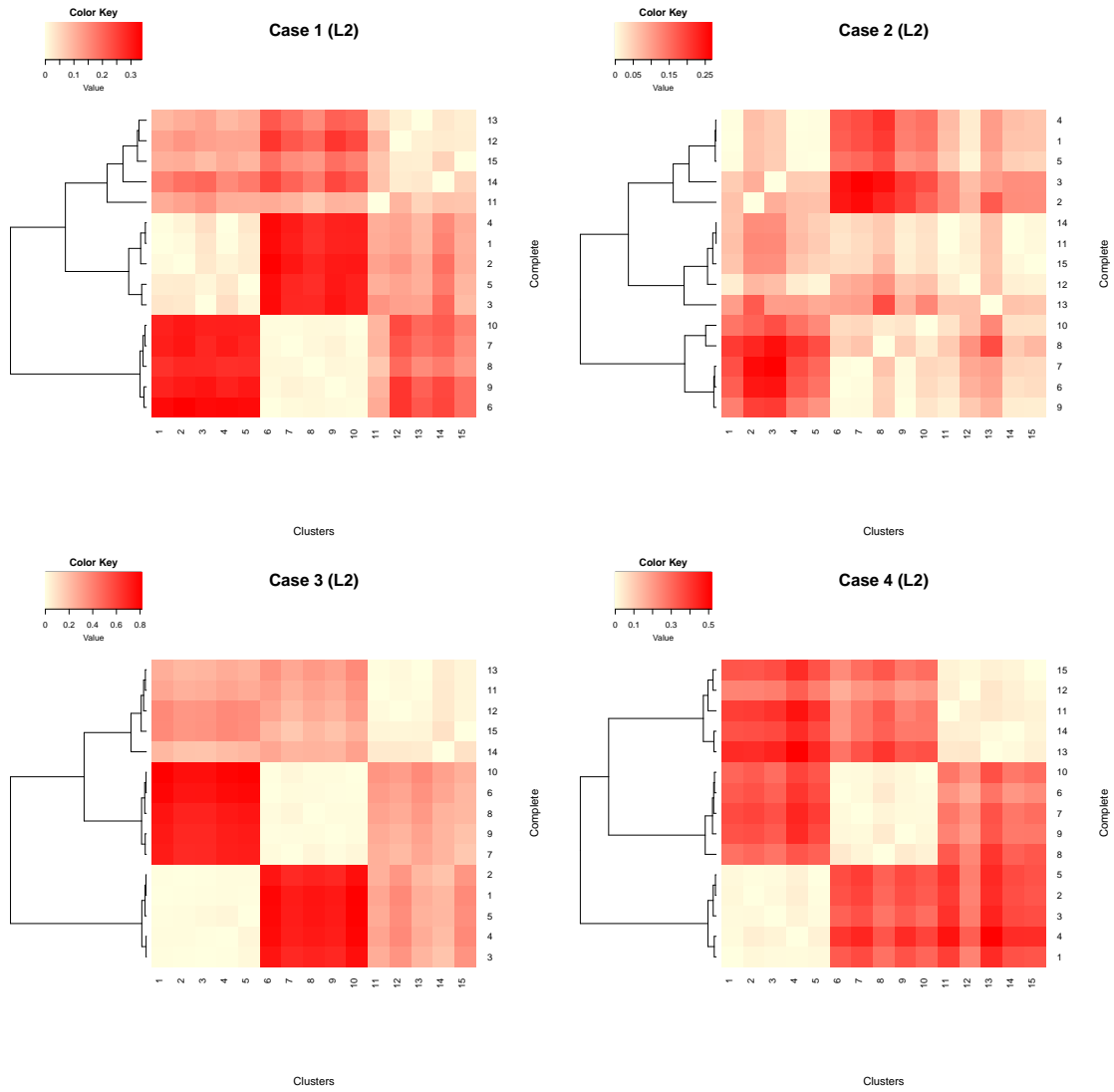


Figure 2.5: L_2 Clustering Heatmap (Simulated Data)

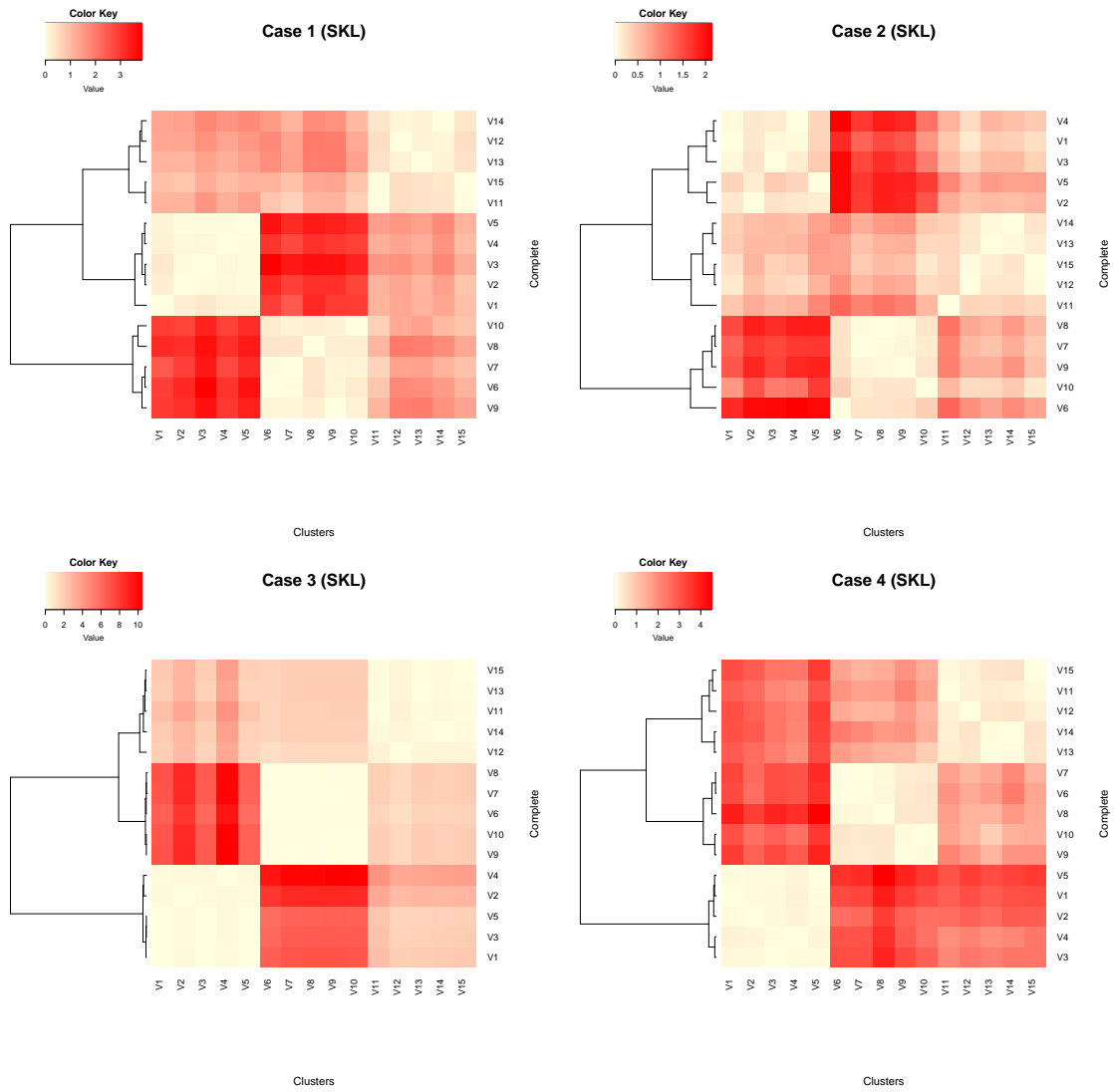


Figure 2.6: SKL Clustering Heatmap (Simulated Data)

2.5 A real data Application to Neuro Retinal Rim (NRR) area

2.5.1 Optical Coherence Tomography

Optical Coherence Tomography (OCT) is a non-invasive imaging technique that uses a broadband light source partitioned into a reference beam and sample beam to generate a reflectivity versus depth profile that details an approximate in-vivo retinal biopsy. OCT yields high-resolution cross-sectional images of the retina, retinal nerve fiber layer, as well as the optic nerve head [10], however in this dissertation, we are concerned with only the neuroretinal rim (NRR) measurements. Figure 2.7 shows a diagram of the retina, with the optic disc, optic cup, neuroretinal rim labeled, as well as labels of the four quadrants of the eye: superior, nasal, inferior, and temporal.²

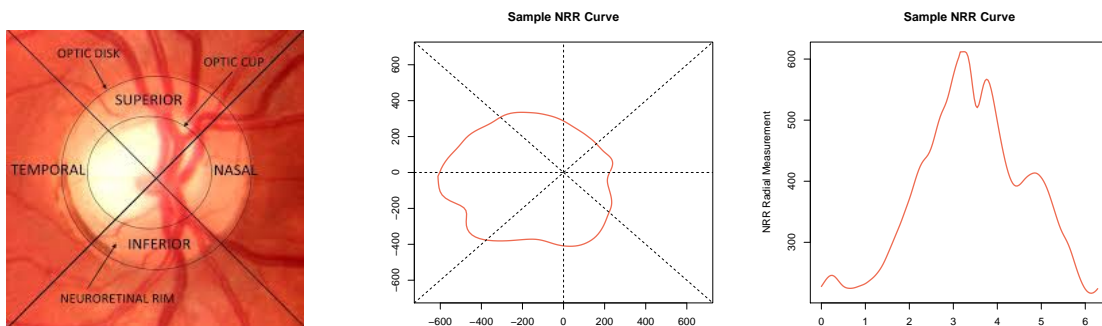


Figure 2.7: From left to right, NRR Image/Diagram, Example NRR Curve Plotted on the Circle, Example NRR Curve Plotted on the Flat

²This problem and related data set was introduced to us by Professor Saumyadipta Pyne, School of Public Health, University of Pittsburgh, and Mr. Hasnat Ali, L.V. Prasad Eye Institute, Hyderabad, India. What we are using here is a small part of the much larger original data, with the limited aim of illustrating our methodologies. A more comprehensive analysis and biological implications will be presented in a future work. See also Section 3.4

2.5.2 General Data Characteristics

Although an NRR measurement for each eye can be taken continuously around the eye, in practice, it is measured on a discrete grid of regularly spaced angles around the eye. Since the measurements are taken radially around the circumference of the retina, these curves are circular in nature, and are amenable to statistical methodologies described earlier for directional statistics. The middle pane of Figure 2.7 is an example of a randomly selected NRR curve plotted on the circle, and the third pane is the same curve plotted on a linear scale. Although both the flat and circular representations are not necessary, they are presented here merely to illustrate the direct mapping between the two representations.

Domain knowledge experts who specialize in ophthalmology, motivated the idea that the most relevant features of the OCT curves, as they relate to the health of the eye, are to be found in the curve shape and not necessarily the curve magnitude. For example, while overall nerve fiber layer thickness may vary naturally from eye to eye within the normal population, the spatial distribution of that thickness may be the key determining factor as to the future health prognosis of the eye. Thus we represent and analyze the NRR curve, as the normalized curve,

$$X_i(t) = \frac{f_i(t)}{\int_0^{2\pi} f_i(t) dt}.$$

It has been suggested that the shapes of clusters formed by these normalized OCT curves found in the population have biological significance. The statistical task is then, to cluster these circular curves into K homogeneous groups based on these curve features. This is done by approximating each curve by a vM mixture, and measuring distances between such curves by one of the measures proposed earlier in this chapter.

The dendrograms in Figures 2.9 and 2.10 are a convenient way to graphically represent the hierarchical clustering schemes induced by the L_2 distance and SKL divergence measures. That said, natural questions when comparing across trees like this, is of cluster quality or performance

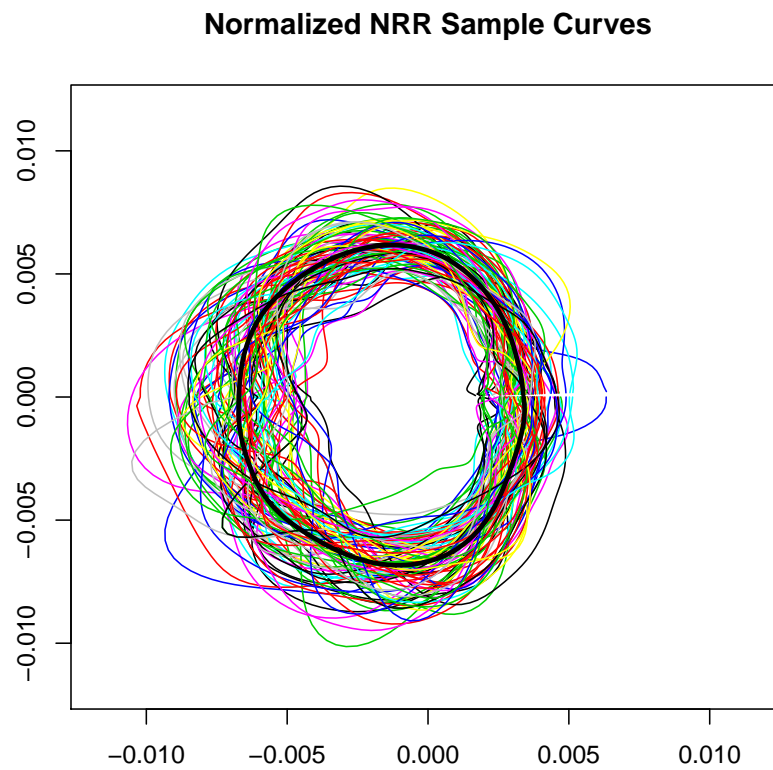


Figure 2.8: Normalized NRR curves, sample size of 100.

of the measures. Because this task is unsupervised, with no ground truth with respect to number of clusters or cluster membership, we will look at cluster agreement between different schemes via their respective ultrametrics. Section 2.5.3 gives a brief description of this approach, but for a full treatment of ultrametrics in hierarchy comparison, see [11] (pg 69-71).

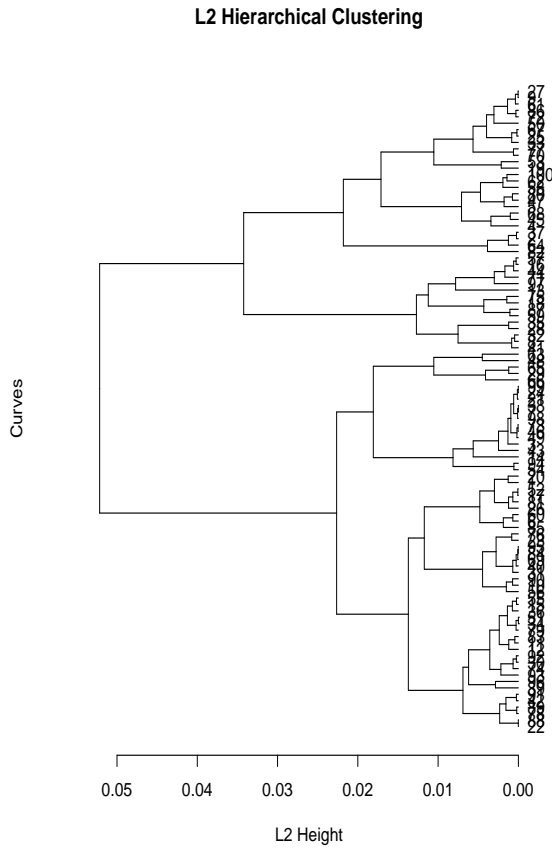


Figure 2.9: L2 induced hierarchy

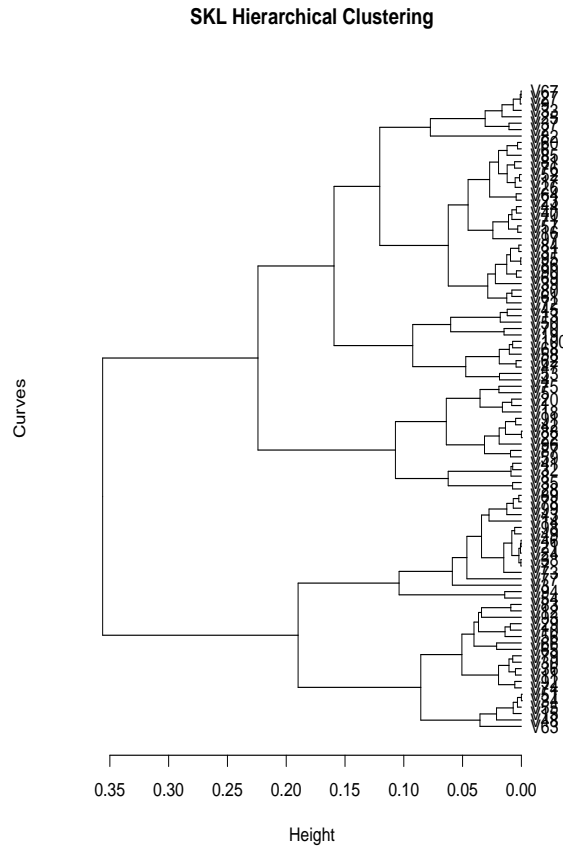


Figure 2.10: SKL induced hierarchy

2.5.3 Cluster Analysis via Ultrametrics

Let the collection of subsets induced by a hierarchical clustering scheme be denoted by, Ω . Below, we compare the total indexed hierarchies by comparing the respective ultrametrics, or distances h_{ij} , between elements i and j , defined as the height of the node that generates the smallest set containing both elements. The collection of heights, $\{h_{ij}; \text{ for } i, j \in \Omega\}$ satisfies the ultrametric inequality,

$$h_{ij} \leq \max(h_{ik}, h_{jk}) \quad \forall i, j, k \in \Omega$$

There are a variety of metrics that can be employed to compare the respective hierarchy ultrametrics, but we will restrict our analysis to looking at four of the metrics that come standard in the R package `clue` [12]:

Euclidean: where we take d as the square root of the sum of the squared differences of ultrametrics, the Euclidean agreement is given as, $1/(1 + d)$

cophenetic: product-moment correlation of the respective ultrametrics, also known as the cophenetic correlation coefficient

cosine: the cosine of the angle between the respective ultrametrics

gamma: a linear transformation of Kruskal's gamma, $1 - d$, where we take d as the rate of inversions between respective ultrametrics, $u_{ij} < u_{kl}$ and $v_{ij} > v_{kl}$, for pairs (i, j) and (k, l)

For a complete description of these agreement metrics, see [13].

In principle, looking at the ultrametrics for the respective hierarchies has clear advantages, in that it does not require a priori knowledge of, nor make a posteriori assertions as to, the *correct* number of clusters. Table 2.4 gives these values for the L_2 and SKL hierarchies represented in Figures 2.9 and 2.10.

| Euclidean | Cosine | Cophenetic | Gamma |
|-----------|--------|------------|--------|
| 0.0337 | 0.8556 | 0.2989 | 0.6126 |

Table 2.4: L_2 and SKL Hierarchy Agreement

In practice, we are looking to find homogeneous groups from the population, so determining at what level to *cut* the tree is of interest. In the next chapter, we take a different clustering approach for this real data example, from a functional perspective.

Chapter 3

Functional Clustering of Circular Densities via Fourier coefficients

Circular densities have a natural periodicity that lends itself to a Fourier basis functional representation. In this chapter, we explore various schemes to cluster circular densities based on this functional representation, and compare the resulting clusterings. We will again look at the NRR curves as our real data example.

3.1 Functional Representation

Suppose we have a collection of curves, f_1, \dots, f_n , to be clustered into K groups. Here, let the f_i be considered as representing the *original* OCT measurements, and X_i represent the *normalized* curves, i.e.

$$X_i(t) = f_i(t) / \int_0^{2\pi} f_i(t) dt.$$

While these curves are actually continuous, in practice, we measure it at a discrete grid of angles, $t \in [0, 2\pi)$ which constitutes our data. Also, since these curves are periodic with period 2π , they can be expressed in terms of the Fourier coefficients $\{\psi_j(t) = e^{i2\pi jt}, j \in \mathbb{N}\}$. Practically

speaking, a finite set of basis functions $\{\psi_j\}_{j=1}^p$ will serve to approximate any curve. Suppose that the curve X_i can be approximated using this set with the curve-specific basis function coefficients γ_{ij} . Thus the curves can be represented in the form,

$$X_i(t) = \sum_{j=1}^p \gamma_{ij} \psi_j(t), i = 1, \dots, n$$

where the random vector $\gamma_i = (\gamma_{i1}, \dots, \gamma_{ip}) \in \mathbb{R}^p$, for some $p \in \mathbb{N}$.

3.1.1 Discriminative Functional Clustering Models

Charles Bouveyron *et al.*[14] propose a functional clustering method called “FunFEM”, that employs a discriminative functional mixture model to facilitate the data being clustered in $F[0, T]$, a discriminative functional subspace of $L_2[0, T]$, spanned by a set of d basis functions $\{\varphi_j\}_{j=1, \dots, d}$ in $L_2[0, T]$. The d -dimensional basis, $\{\varphi_j\}_{j=1, \dots, d}$ is related to the Fourier basis $\{\psi_j\}_{j=1, \dots, p}$ through the linear transformation, $\varphi_j = \sum_{l=1}^p u_{jl} \psi_l$ where the matrix $\{U\}_{ij} = u_{ij}$ is orthogonal and of dimension $p \times d$. This relationship implies that the random vectors $\Lambda \in \mathbb{R}^d$ and $\Gamma \in \mathbb{R}^p$ are related by the linear transformation,

$$\Gamma = U\Lambda + \epsilon,$$

where $\epsilon \in \mathbb{R}^p$ is the independent random error term, and Λ represents the latent coefficients on the d -dimensional *discriminative* subspace. Here $d < \min(p, K)$, since $d = K - 1$ is sufficient to discriminate K groups [15]. Clustering proceeds by fitting a Gaussian mixture model to the coefficients in Λ . This algorithm is implemented in the R package funFEM [14].

3.2 Tuning Parameter Selection

As is the case with most unsupervised clustering methods, several tuning parameters need to be specified by the user. In this instance, for the functional clustering model, we must specify the dimension of the curve data p as well as the set of basis functions, $\{\varphi_j\}_{j=1}^p$, and the number of clusters, K . The FunFEM algorithm of [14] will internally specify the latent dimension d and the orthogonal matrix U . Due to the circular nature of the data, the functions X_i are periodic. Hence, a natural choice is to use the first p Fourier or trigonometric basis functions. As a floor for the number of basis functions p to use, we require that the finite-dimensional representation retain at least 95% of the functional variability in the data. This is akin to a cumulative scree-plot analysis in standard principal component analysis when performing dimension reduction for multivariate data.

To avoid model overfitting, it is necessary to determine the smallest number of basis functions that recover the shapes of the actual curves sufficiently well, recovering the functional variability to meet our threshold. To that end, we will look at the ‘‘Fraction of Variation Explained’’ (FVE) as a function of the number of basis functions [16]. In particular, if we have n curves, $X_1(t), X_2(t), \dots, X_n(t)$, with a sample mean given by,

$$\bar{X}(t) = \frac{1}{n} \sum_{i=1}^n X_i(t), \quad t = 1, 2, \dots, 180$$

The total variation (TV) is given by,

$$\text{TV} = \frac{1}{n-1} \sum_{i=1}^n \int (X_i(t) - \bar{X}(t))^2 dt$$

If $\psi_1, \psi_2, \dots, \psi_p$ are basis functions and γ_{ij} the associated basis coefficients, then the functional

approximation for the i^{th} curve at point t , $X_i(t)$, is given by,

$$X_i(t) \approx \sum_{j=1}^p \gamma_{ij} \psi_j(t) =: X_i^p(t),$$

The fraction of variation explained (FVE) for a given p is then,

$$\text{FVE} = \frac{TV - \frac{1}{n-1} \sum_{i=1}^n \int (X_i^p(t) - X_i(t))^2 dt}{TV}$$

For choosing the number of clusters K , we assess three separate model selection criteria, namely the Akaike Information Criterion (AIC), the Bayesian Information Criterion (BIC), and the Integrated Completed Likelihood criterion (ICL) [17] to balance the competing goals of interpretability, which one can argue improves as the number of clusters K decreases, against the within-cluster variability, which improves as the number of clusters increases.

3.3 Functional and non-functional Mahalanobis Clustering

Finally, we consider two additional metrics that make use of the functional representation described above, to generate a total of four cluster hierarchies, for the purpose of seeing what, if any cluster agreement can be observed.

The first of these, is the standard multivariate Mahalanobis distance, since it can be defined as the distance between a distribution and a point, or, as we have here, the distance between two random vectors, where each curve is represented as an p -dimensional vector comprised of that curve's specific Fourier coefficients.

$$d_M(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T S^{-1} (\mathbf{x} - \mathbf{y})},$$

where \mathbf{x} and \mathbf{y} are the p -dimensional vectors for the x^{th} and y^{th} curves respectively, and S is the

$p \times p$ sample covariance matrix. Furthermore, the Mahalanobis distance is unitless and scale-invariant, measuring the distance between the vectors in standard deviations along the principle component axes, which effectively accounts for the correlation within the dataset.

The second metric is the functional analogue to the Mahalanobis distance, the Functional Mahalanobis semi-distance, as described by Gelano *et al.*[18]. The semi-distance between two i.i.d. functional random variables \mathcal{X}_1 and \mathcal{X}_2 , with mean function $\mu_{\mathcal{X}}$, and compact covariance operator $\Gamma_{\mathcal{X}}$ is defined as,

$$d_{FM}(\mathcal{X}_1, \mathcal{X}_2) = \langle \Gamma_K^{-1/2}(\mathcal{X}_1 - \mathcal{X}_2), \Gamma_K^{-1/2}(\mathcal{X}_1 - \mathcal{X}_2) \rangle,$$

where $\Gamma_K^{-1/2}$ is the regularized square root inverse operator of $\Gamma_{\mathcal{X}}$ defined as,

$$\Gamma_K^{-1/2}(\xi) = \sum_{k=1}^K \frac{1}{\lambda_k^{1/2}} \langle \psi_k, \xi \rangle \psi_k$$

where λ_k and ψ_k are the eigenvalues and (orthonormal) eigen-functions of $\Gamma_{\mathcal{X}}$. In the functional setting, $\Gamma_{\mathcal{X}}$ serves the analogous role of the covariance matrix in the multivariate setting.

3.4 Real Data Example with NRR curves

The FVE values for the NRR sample are given in Table 3.1 and the adjacent Plot 3.1. We see it takes 11 basis functions to break 0.95% fraction of variance explained, and proceed using this as our basis set.

Looking at cluster selection criteria in Figure 3.2, BIC, AIC, and ICL across values of K ranging from two to eight. We see that $K = 6$ is the best fit, for BIC and ICL. AIC values continue to climb after, but there is a pronounced *knee* that does support the findings of the other two selection criteria, and as such, we settle on this value for our example, and implement

| Nmb of Basis Func. | FVE |
|--------------------|-----------|
| $p = 3$ | 0.2963387 |
| $p = 5$ | 0.7766510 |
| $p = 7$ | 0.8841013 |
| $p = 9$ | 0.9335552 |
| $p = 11$ | 0.9623640 |
| $p = 13$ | 0.9788576 |
| $p = 15$ | 0.9877183 |
| $p = 17$ | 0.9924922 |
| $p = 19$ | 0.9956720 |
| $p = 21$ | 0.9974243 |

Table 3.1: FVE Values

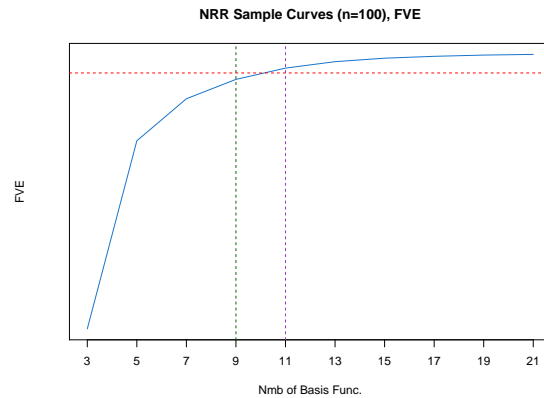


Figure 3.1: Fraction of variance explained

the discriminative functional clustering algorithm. We note that this method does not produce a clustering hierarchy, but rather partitions the curves into K groups, which means we will look at cluster agreement between competing partition schemes, e.g. the adjusted Rand index.

| Adjusted Rand Index | Number of Clusters | | | | | |
|---------------------|--------------------|--------|--------|--------|--------|--------|
| | 2 | 3 | 4 | 5 | 6 | 7 |
| ARI(SKL,L2) | 0.2621 | 0.2541 | 0.2520 | 0.2278 | 0.3000 | 0.2994 |
| ARI(funFEM,SKL) | 0.1355 | 0.3512 | 0.4027 | 0.2716 | 0.1804 | 0.3988 |
| ARI(funFEM,L2) | 0.3296 | 0.3222 | 0.3408 | 0.1487 | 0.1364 | 0.2288 |

Table 3.2: Adjusted Rand Index

Knowing that $K = 6$ is the optimal clustering under the funFEM scheme, but as the algorithm generates partitions we ran it across a range of clusters for the purpose of comparing against the cluster memberships of the L_2 and SKL schemes presented in Chapter 2. To that end, we cut the respective L_2 and SKL trees at the same levels to match the number of clusters from funFEM and compare the respective cluster memberships. Table 3.2 gives the adjusted Rand index (ARI) between the funFEM clustering scheme, and those described above.

In comparing the four clustering schemes that yield hierarchies, we can again look to comparing the respective ultrametrics. As in Chapter 2, we will compare our clustering hierarchies by first looking at the respective dendrograms. One thing that we notice immediately is the dis-

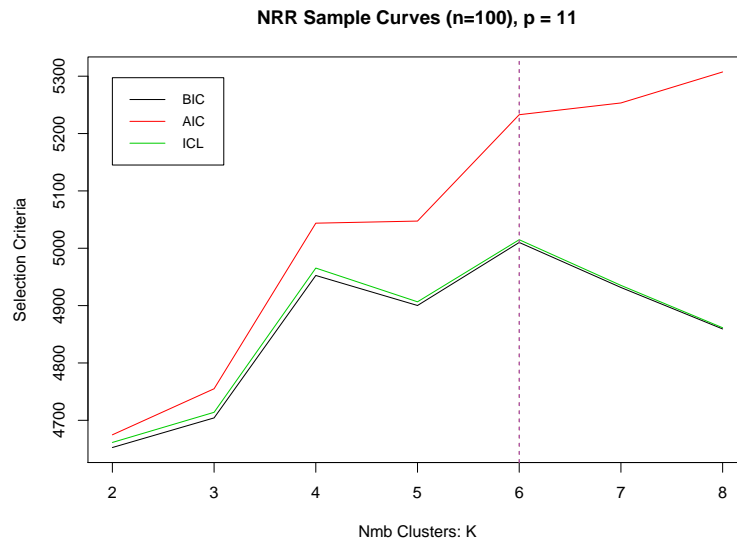


Figure 3.2: Cluster selection criteria

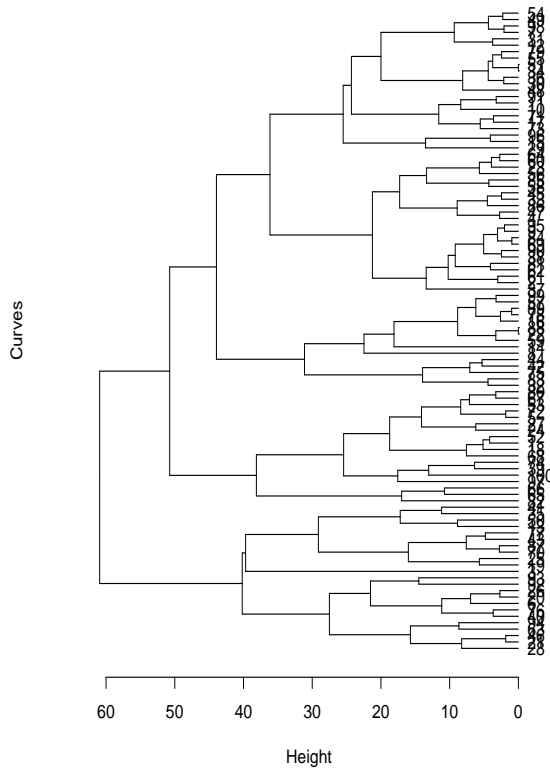
parity in heights of the Mahalanobis and functional Mahalanobis hierarchical clusterings have heights that are orders of magnitude greater than those induced by the L_2 and SKL measures, so before we can compare the respective ultrametrics, we normalize the heights for all of them by,

$$(h_i - \min(h_u)) / (\max(h_u) - \min(h_u))$$

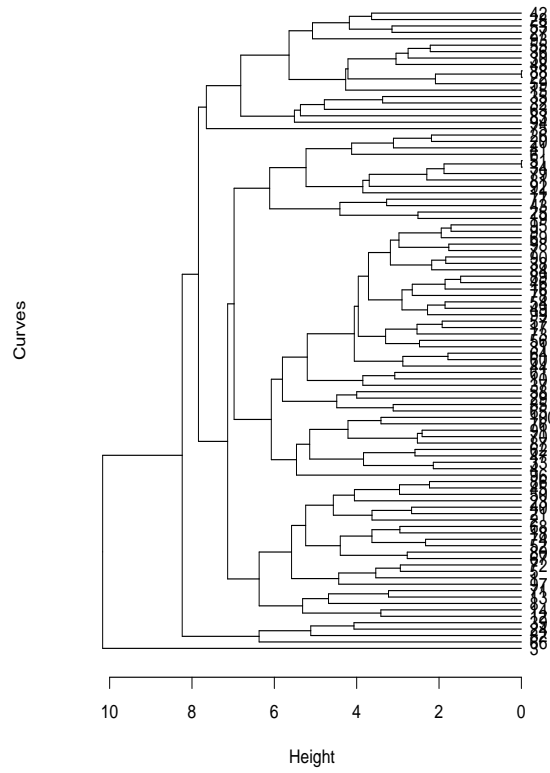
where h_u are the heights of the u^{th} clustering.

Table 3.3 gives the resulting agreement measures for the Euclidean, cosine, cophenetic, and gamma measures of agreement, as well the average over the four measures.

Functional Mahal Hierarchical Clustering



Mahalanobis Hierarchical Clustering



| | | L2 | SKL | Mahalanobis | Fun.Mahalanobis |
|------------|-----------------|--------|--------|-------------|-----------------|
| Euclidean | L2 | 1.0000 | | | |
| | SKL | 0.0337 | 1.0000 | | |
| | Mahalanobis | 0.0371 | 0.0398 | 1.0000 | |
| | Fun.Mahalanobis | 0.0328 | 0.0342 | 0.0528 | 1.0000 |
| Cosine | L2 | 1.0000 | | | |
| | SKL | 0.8556 | 1.0000 | | |
| | Mahalanobis | 0.8733 | 0.8931 | 1.0000 | |
| | Fun.Mahalanobis | 0.8571 | 0.8695 | 0.9517 | 1.0000 |
| Cophenetic | L2 | 1.0000 | | | |
| | SKL | 0.2989 | 1.0000 | | |
| | Mahalanobis | 0.0584 | 0.1114 | 1.0000 | |
| | Fun.Mahalanobis | 0.1107 | 0.0965 | 0.2748 | 1.0000 |
| Gamma | L2 | 1.0000 | | | |
| | SKL | 0.6126 | 1.0000 | | |
| | Mahalanobis | 0.4117 | 0.4376 | 1.0000 | |
| | Fun.Mahalanobis | 0.4651 | 0.4347 | 0.4307 | 1.0000 |
| Average | L2 | 1.0000 | | | |
| | SKL | 0.4502 | 1.0000 | | |
| | Mahalanobis | 0.3451 | 0.3705 | 1.0000 | |
| | Fun.Mahalanobis | 0.3664 | 0.3587 | 0.4275 | 1.0000 |

Table 3.3: Agreements using ultrametric distances

Chapter 4

Clustering Textual Data with Directional Statistics

4.1 Introduction

In this chapter, we first review various modelling techniques that employ directional statistics as the statistical framework for clustering textual documents into homogeneous groups, and afterwards, add our contribution to this body of methods. Sections 4.2 and 4.3 begin with an overview of the vectorization of textual data, clustering on the unit hypersphere. We then define distance and divergence measures between mixtures of the von Mises-Fisher distribution (vMF) in Section 4.4, followed by an application in Section 4.5.

4.2 Textual Data: Definitions and Notation

To get started, it is useful to first define a few basic terms and some of the notation we will be using below.

- Let the collection of unique discrete elements, w_1, w_2, \dots, w_p , hereafter referred to as

words, be our *vocabulary*, which we denote by,

$$W = \{w_1, \dots, w_p\}$$

where $p \in \mathbb{N}$ is the size (cardinality) of the vocabulary, so that when characterizing a document as a vector of (possibly weighted) word counts, we will conform to the convention of working with a vector in p -dimensional space, \mathbb{R}^p .

- \mathbb{S}^{p-1} is defined as the $(p - 1)$ -dimensional (hyper)sphere embedded in \mathbb{R}^p .
- Individual sets of words are called *documents*. We denote the d^{th} document as,

$$\mathbf{d}_d = \{w_{d1}, w_{d2}, \dots, w_{N_d}\}, w_{dk} \in W \forall k$$

where $N_d \in \mathbb{N}$ is the total number of words (including replicates) in document \mathbf{d}_d .

- The collection of distinct texts or documents, is called the *corpus*, and is defined by,

$$C = \{\mathbf{d}_1, \mathbf{d}_2, \dots, \mathbf{d}_D\}$$

with $D \in \mathbb{N}$ denoting the number of texts in our corpus.

- Let n_{jd} be the number of occurrences of word w_j in document \mathbf{d}_d . Then the total number of words is

$$N_d = \sum_{j=1}^p n_{jd}$$

and this allows us to represent each document as a p -dimensional vector,

$$\mathbf{v}_d = [n_{1d} \ n_{2d} \ \dots \ n_{pd}]^T$$

where the entry, n_{jd} corresponds to the number of occurrences of the j^{th} vocabulary word, w_j , in document \mathbf{d} . We make mention here, that for even a moderately sized corpus, textual data can have an extremely large vocabulary and result in sparse document word-count vectors predominately populated by zeros.

- For each *document* \mathbf{d} , define our document vector \mathbf{x} , as the normalized representation of the count vector, \mathbf{v} , so that,

$$\mathbf{x}_d \equiv \mathbf{v}_d / \|\mathbf{v}_d\|, \text{ and } \mathbf{x}_d \in \mathbb{R}^p, \|\mathbf{x}_d\| = 1, \text{ and thus } \mathbf{x}_d \in \mathbb{S}^{p-1} \text{ for } d = 1, 2, \dots, D$$

- The collection of document word-count vectors for a corpus is called a *term-document matrix* (TDM) (or if we were to take the transpose, a *document-term matrix* (DTM)). Define A as the $(p \times D)$ -dimensional TDM, where each row is one of the terms from the vocabulary \mathcal{W} , and each column is a word-count vector for a specific document with entries corresponding to the number of occurrences of the term in that document.

$$A_{p \times D} = \begin{pmatrix} n_{11} & n_{12} & \cdots & n_{1D} \\ n_{21} & n_{22} & \cdots & n_{2D} \\ \vdots & \vdots & \ddots & \vdots \\ n_{p1} & n_{p2} & \cdots & n_{pD} \end{pmatrix}$$

Again, for a moderate to large corpus, we expect to have a (very) sparse matrix. Also, we note that the entries may be a transformation of the word-counts by an appropriate weighing scheme, or composition of weighing schemes. For example, in fields such as textual data analysis or information retrieval, it is common to weight the terms by the *term frequency-inverse document frequency* (tf-idf), where the frequency of a particular word in a particular document is offset by the frequency of that word across all documents in the corpus, in order to attach some measure

of importance to that word for that document. ¹

4.2.1 Textual Data as Directional Data

With the documents being modeled as unit vectors, this allows for a variety of different weighting schemes that can help focus on different components of the feature space. Normalization makes long or short documents comparable. Projecting our documents onto the unit hypersphere is especially useful for sparse and very sparse data, as is typically the case with textual data whenever dealing with a corpus of any significant size (cardinality). This normalization also allows us to use a weighted cosine distance as a measure of similarity, which has been demonstrated to be more robust to feature noise, and is commonly used in directional data clustering and information retrieval.

Taking advantage of the fact that documents are mapped onto the unit hypersphere, one may fit mixture models based on spherical distributions on the hypersphere. In what follows, we work with mixtures of von Mises-Fisher (vMF) distribution, which is well suited to this task. Clustering of such sets of documents can then be done using an L_2 norm, details on which follow.

4.3 Clustering on the hypersphere with Mixture of von Mises-Fisher Distributions

4.3.1 Topic Models and sentiment analysis

Topic models are probabilistic models for uncovering the underlying semantic structure of a document collection based on a hierarchical Bayesian analysis of the original documents, by

¹Although some weighting schemes may be surrogates for others, where the particular situation or user preference is the deciding factor of choosing one over the other, all of them serve a specific purpose.

discovering patterns of word use and connecting documents that exhibit similar patterns. [19] Specifically, we will be working with *Latent Variable Mixture Models*, where observed data interact with unseen, or latent, random variables through a proposed unobserved structure in the observed data. The nature of that structure is unearthed through inference on the posterior distribution.

In Chapter 2 and Chapter 3 we reference the `movMF` package by Hornik and Grün [20]. This package employs the EM algorithms described by A. Banerjee *et al.* [8] to estimate the mixture model parameters. In Section 4.3.3, we give a brief description of these methods, along with any justifications for choices made when options were available, e.g., why we might prefer the `hard-max` clustering scheme over the `soft-max` clustering scheme.

4.3.2 The von Mises-Fisher (vMF) Distribution in high dimensions

Let $x \in \mathbb{R}^p$ be a unit vector such that $\|x\| = 1$. i.e. $x \in \mathbb{S}^{p-1}$, the $(p - 1)$ -dimensional hypersphere with radius 1, embedded in \mathbb{R}^p . We say that \mathbf{x} has a p -variate von Mises-Fisher distribution (vMF), with mean direction vector $\boldsymbol{\mu}$ and concentration parameter $\kappa \geq 0$ if it has the probability density function

$$f(\mathbf{x}; \boldsymbol{\mu}, \kappa) = c_p(\kappa) \exp(\kappa \mathbf{x} \cdot \boldsymbol{\mu}),$$

where $c_p(\kappa)$ is the normalizing constant given by

$$c_p(\kappa) = \frac{\kappa^{p/2-1}}{(2\pi)^{p/2} I_{p/2-1}(\kappa)}$$

and I_k is the modified Bessel function of the first kind and order k , and $\mathbf{x} \cdot \boldsymbol{\mu}$ is the usual inner product.

4.3.3 EM algorithm for a Mixture of vMF distributions

Consider a k -component mixture of vMF distributions denoted by

$$f(\mathbf{x}|\Theta) = \sum_{h=1}^k p_h f_h(\mathbf{x}|\boldsymbol{\mu}_h, \kappa_h),$$

where $f_h(\mathbf{x}|\boldsymbol{\mu}_h, \kappa_h)$ is a vMF distribution with parameters $\boldsymbol{\mu}_h$ and κ_h which represent the mean direction and concentration of the h^{th} mixture component, for $h = 1, 2, \dots, k$. The parameter space consists of

$$\Theta = \{p_1, p_2, \dots, p_k, \boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_k, \kappa_1, \kappa_2, \dots, \kappa_k\}$$

with the usual restriction on the mixture proportions or the probabilities $p_h \geq 0$, $\sum_{h=1}^k p_h = 1$.

In the context of a generative mixture model, we suppose that the process is randomly generated, and to that end, the process of generating a random document vector begins with first randomly select a particular distribution, or topic, from this mixture distribution. We randomly select the h^{th} vMF distribution with probability p_h , and then, from $f_h(\mathbf{x}|\boldsymbol{\mu}_h, \kappa_h)$, sample a point on \mathbb{S}^{p-1} . The distribution from which the document vector is sampled is a latent random variable, but if we suppose that instead, we know which distributions are selected, and $\mathcal{Z} = \{z_1, z_2, \dots, z_n\}$ is the set of topics, where $z_i = h$ if \mathbf{x}_i was sampled from $f_h(\mathbf{x}|\boldsymbol{\mu}_h, \kappa_h)$. The log-likelihood of the observed data is given by,

$$\log \mathbb{P}(\mathcal{X}, \mathcal{Z}|\Theta) = \sum_{i=1}^n \log (p_{z_i} f_{z_i}(\mathbf{x}_i|\boldsymbol{\mu}_{z_i}, \kappa_{z_i}))$$

The true state of nature is one where we do not actually observe the particular topics selected, and as such, instead of an easily calculable conditional density, we have a random variable that depends on the distribution of our topics, \mathcal{Z} . This random variable is the complete data log-likelihood. Now, given observed data and a particular mixture, (\mathcal{X}, Θ) , we can formulate the E-step in the EM algorithm, as the estimate of $\mathcal{Z}|\mathcal{X}, \Theta$.

Given the constraints, $\boldsymbol{\mu}_h \cdot \boldsymbol{\mu}_h = 1$ and $\kappa_h \geq 0$, we get the parameter estimate update equations,

$$p_h = \frac{1}{n} \sum_{i=1}^n p(h|\mathbf{x}_i, \Theta),$$

$$\mathbf{r}_h = \sum_{i=1}^n \mathbf{x}_i p(h|\mathbf{x}_i, \Theta),$$

(M-Step)

$$\hat{\boldsymbol{\mu}}_h = \frac{\mathbf{r}_h}{\|\mathbf{r}_h\|},$$

$$\frac{I_{d/2}(\hat{\kappa}_h)}{I_{d/2-1}(\hat{\kappa}_h)} = \frac{\|\sum_{i=1}^n \mathbf{x}_i p(h|\mathbf{x}_i, \Theta)\|}{\sum_{i=1}^n p(h|\mathbf{x}_i, \Theta)} = \frac{\|\mathbf{r}_h\|}{\sum_{i=1}^n p(h|\mathbf{x}_i, \Theta)}.$$

With these update equations, we update the distributions of $\mathcal{Z}|\mathcal{X}, \Theta$ via two different schemes for our E-Step, that are the basis for the soft assignment algorithm, `soft-moVMF`, and the hard assignment algorithm, `hard-moVMF`.

As described in Neal and Hinton (1998) and Bilmes (1997), the distribution of the hidden variables is given by the standard EM machinery,

$$p(h|\mathbf{x}_i, \Theta) = \frac{p_h f_h(\mathbf{x}_i|\Theta)}{\sum_{l=1}^k p_l f_l(\mathbf{x}_i|\Theta)}.$$

Collins (1997) shows that the *incomplete data log-likelihood*, $\log p(\mathcal{X}|\Theta)$, is non-decreasing at each iteration of the parameter and distribution updates. The `soft-moVMF` algorithm is based on iterating over these two updates. The second update scheme, the `hard-moVMF` algorithm, uses the (unsupervised) hard-assignment heuristic, that yields a distribution of hidden variables,

$$q(h|\mathbf{x}_i, \Theta) = \begin{cases} 1, & \text{if } h = \operatorname{argmax}_{h'} p(h'|\mathbf{x}_i, \Theta) \\ 0, & \text{otherwise.} \end{cases}$$

The hard assignment maximizes a lower bound on the incomplete data log-likelihood, and iterating over the M-step and the hard assignment rule gives us the hard-moVMF algorithm.

4.3.4 Mixture vMF Models

Hard Assignments

In the most simple mixture vMF model, a hard assignment (winner-take-all) strategy is employed, where each document belongs to exactly one topic cluster. The cluster assignment variable, Z_i , is drawn from a multinomial distribution with mixing distribution $\boldsymbol{\pi}$, and each document, \mathbf{v} , is drawn from exactly one of K topics, with vMF distribution parameterized by mean direction $\boldsymbol{\mu}_{z_i}$ and concentration parameter κ .

The hard assignment algorithm also maximizes a tight lower bound on the incomplete log-likelihood of the data, and as such, should give an acceptable clustering with respect to the log-likelihood. Banerjee et al. assert that the advantages of this result are applicable to any mixture model that uses the EM algorithm, but particularly for mixtures of vMF distributions. This is again due to the problem of computing the normalization constant $c_p(\kappa_h)$ which involves Bessel functions. Specifically, in a hard assignment, we do not need to calculate the partition/mixture function, $\sum_{l=1}^k p_l f_l(\mathbf{x}_i | \theta_l)$ for every document, \mathbf{x}_i . This is analogous to reducing the LDA to a more simple Dirichlet-multinomial model where each document belongs to one, and only one, topic. It is also worth noting that for the vMF distribution, these computations must be of sufficiently high-precision to avoid the numerical problems of underflow and overflow, again due to the Bessel functions. With the hard assignment algorithm, we reduce the number of computations to something on the order of $\mathcal{O}(k)$ per iteration to compute $c_p(\kappa_h)$, for $h = 1, 2, \dots, k$, from the number required per iteration in the soft assignment, which is on the order of $\mathcal{O}(nk)$, for all $f_l(\mathbf{x}_i | \theta_l)$, to be able to calculate both the partition function and the associated probabilities, $p(h | \mathbf{x}_i, \Theta)$.

Additionally, the hard assignment algorithm only requires storing n integers for the cluster assignments, while the soft assignment algorithm needs to maintain nk floating point numbers for all of the conditional probabilities, at the user specified level of precision. For extremely large data sets and large numbers of clusters, this can create computational issues around available space. Thus, the hard assignment algorithm lends itself well to scalability and is favored in terms of computational efficiency.

The authors [8] present arguments as to the choice of q , namely

$$q(h|\mathbf{x}_i, \Theta) = \begin{cases} 1, & \text{if } h = \operatorname{argmax}_{h'} p(h'|\mathbf{x}_i, \Theta) \\ 0, & \text{otherwise} \end{cases}$$

$q \in \mathcal{H} \subset \mathcal{P}$, where \mathcal{H} is the class of probability distributions that assume probability value 1 for some mixture component and 0 for all others, and is a subset of all possible distributions \mathcal{P} . The hard assignment algorithm constrains the latent variables to have distributions that are members of \mathcal{H} , and we note that a mixture model following $p(h|\mathbf{x}_i, \Theta)$, as defined above will not be a member of \mathcal{H} . It is possible to reasonably bound the incomplete log-likelihood of the data from below, by using expectations over some *optimal* distribution $q \in \mathcal{H}$ (see Banerjee et al. for details). Therefore, we have that the hard assignment clustering algorithm (basically) maximizes a lower bound on the incomplete log-likelihood.

Banerjee et al. [8] introduce the function $F(\tilde{p}, \Theta)$ namely,

$$F(\tilde{p}, \Theta) = \mathbb{E}_{\tilde{p}}[\log P(X, Z|\Theta)] + H(\tilde{p})$$

where $H(\tilde{p})$ is the Shannon entropy of a discrete distribution \tilde{p} . The EM algorithm alternates between choosing a distribution \tilde{p} for a fixed Θ that will maximize the function $F(\tilde{p}, \Theta)$ in the E-step, and subsequently for that particular choice of \tilde{p} , estimate the parameters Θ that will

again maximize the function F in the M-step. Neal and Hinton (1998) have shown that for a given Θ ,

$$p = \operatorname{argmax}_p F(\tilde{p}, \Theta).$$

It is the distribution that maximizes the function F given by

$$\begin{aligned} F(p, \Theta) &= \mathbb{E}_p [\log P(X, Z|\Theta)] + H(p) \\ &= \mathbb{E}_p [\log P(X, Z|\Theta)] - \mathbb{E}_p [\log P(Z|X, \Theta)] \\ &= \mathbb{E}_p \left[\log \left(\frac{P(X, Z|\Theta)}{P(Z|X, \Theta)} \right) \right] \\ &= \mathbb{E}_p [\log P(X|\Theta)] \\ &= \log P(X|\Theta). \end{aligned}$$

4.4 Measures of distance and divergence between two vMF models

In this section, we extend the results of Chapter 2, regarding the the L_2 distance, and the symmetrized Kullback-Liebler divergence between two vMF models in any arbitrary higher dimension p . In following the same approach as in Chapter 2, we begin with Lemma 4.4.1 to find the inner product of two vMF distributions, similar to the result in Lemma 2.2.1.

4.4.1 L_2 distance between two vMF models

Lemma 4.4.1. *For any two vMF distributions $f \sim \text{vMF}(\mathbf{x}; \boldsymbol{\mu}_1, \kappa_1)$ and $g \sim \text{vMF}(\mathbf{x}; \boldsymbol{\mu}_2, \kappa_2)$*

$$\int_{\mathbb{S}^{p-1}} f(\mathbf{x})g(\mathbf{x})d\mathbf{x} = \frac{\kappa_1^s \kappa_2^s I_s(\kappa_*)}{(2\pi)^{s+1} I_s(\kappa_1) I_s(\kappa_2)}$$

where $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathbb{R}^p$ are the mean direction vectors such that $\|\boldsymbol{\mu}_i\| = 1$, κ_1 and κ_2 the concentration parameters for the respective vMF distributions, $f(\mathbf{x})$ and $g(\mathbf{x})$, $s = p/2 - 1$, $I_\nu(\kappa)$ is the modified Bessel function of the first kind and order ν , and κ_* is defined as,

$$\kappa_* = \|\kappa_1 \boldsymbol{\mu}_1 + \kappa_2 \boldsymbol{\mu}_2\|$$

where $\|\cdot\|$ is the usual L_2 norm.

Proof: For $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$, and $\mathbf{x} \in \mathbb{S}^{p-1}$,

$$\begin{aligned} \int_{\mathbb{S}^{p-1}} f(\mathbf{x})g(\mathbf{x})d\mathbf{x} &= \int_{\mathbb{S}^{p-1}} \frac{\kappa_1^s}{(2\pi)^{s+1}I_s(\kappa_1)} \exp\{\kappa_1 \mathbf{x} \cdot \boldsymbol{\mu}_1\} \frac{\kappa_2^s}{(2\pi)^{s+1}I_s(\kappa_2)} \exp\{\kappa_2 \mathbf{x} \cdot \boldsymbol{\mu}_2\} d\mathbf{x} \\ &= c_p(\kappa_1)c_p(\kappa_2) \int_{\mathbb{S}^{p-1}} \exp\{\mathbf{x} \cdot (\kappa_1 \boldsymbol{\mu}_1 + \kappa_2 \boldsymbol{\mu}_2)\} d\mathbf{x} \end{aligned}$$

If we let $\mathbf{u} = \kappa_1 \boldsymbol{\mu}_1 + \kappa_2 \boldsymbol{\mu}_2$, and define,

$$\kappa_* = \|\mathbf{u}\| \quad \text{and} \quad \boldsymbol{\mu}_* = \frac{\mathbf{u}}{\|\mathbf{u}\|},$$

then we can write,

$$\begin{aligned} \int_{\mathbb{S}^{p-1}} f(\mathbf{x})g(\mathbf{x})d\mathbf{x} &= c_p(\kappa_1)c_p(\kappa_2) \int_{\mathbb{S}^{p-1}} \exp\{\kappa_* \mathbf{x} \cdot \boldsymbol{\mu}_*\} d\mathbf{x} \\ &= c_p(\kappa_1)c_p(\kappa_2) \frac{(2\pi)^{s+1}I_s(\kappa_*)}{\kappa_*^s} \\ &= c_p(\kappa_1)c_p(\kappa_2) [c_p(\kappa_*)]^{-1} \\ &= \frac{\kappa_1^s \kappa_2^s I_s(\kappa_*)}{(2\pi)^{s+1}I_s(\kappa_1)I_s(\kappa_2)} \end{aligned}$$

Using Lemma 4.4.1, we can proceed with finding the L_2 distance between our respective vMF distributions.

Proposition 4.4.1. *The L_2 distance between two vMF distributions, $f(\mathbf{x}; \boldsymbol{\mu}_1, \kappa_1)$ and $g(\mathbf{x}; \boldsymbol{\mu}_2, \kappa_2)$ with $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$, and $\mathbf{x} \in \mathbb{S}^{p-1}$, is given by,*

$$\int_{\mathbb{S}^{p-1}} (f(\mathbf{x}) - g(\mathbf{x}))^2 d\mathbf{x} = \frac{1}{(2\pi)^{s+1}} \left[\frac{\kappa_1^s I_s(2\kappa_1)}{[I_s(\kappa_1)]^2} + \frac{\kappa_2^s I_s(2\kappa_2)}{[I_s(\kappa_2)]^2} - 2 \frac{\kappa_1^s \kappa_2^s I_s(\kappa_*)}{I_s(\kappa_1) I_s(\kappa_2)} \right]$$

where κ_* is defined as in Lemma 4.4.1.

Proof:

$$\begin{aligned} \int_{\mathbb{S}^{p-1}} (f(\mathbf{x}) - g(\mathbf{x}))^2 d\mathbf{x} &= \int_{\mathbb{S}^{p-1}} (f(\mathbf{x}))^2 d\mathbf{x} + \int_{\mathbb{S}^{p-1}} (g(\mathbf{x}))^2 d\mathbf{x} - 2 \int_{\mathbb{S}^{p-1}} f(\mathbf{x})g(\mathbf{x}) d\mathbf{x} \\ &= [c_p(\kappa_1)]^2 \int_{\mathbb{S}^{p-1}} \exp\{2\kappa_1 \mathbf{x} \cdot \boldsymbol{\mu}_1\} d\mathbf{x} + \\ &\quad [c_p(\kappa_2)]^2 \int_{\mathbb{S}^{p-1}} \exp\{2\kappa_2 \mathbf{x} \cdot \boldsymbol{\mu}_2\} d\mathbf{x} - \\ &\quad 2c_p(\kappa_1)c_p(\kappa_2) [c_p(\kappa_*)]^{-1} \\ &= c_p(\kappa_1) \cdot \frac{I_s(2\kappa_1)}{I_s(\kappa_1)} + c_p(\kappa_2) \cdot \frac{I_s(2\kappa_2)}{I_s(\kappa_2)} - 2 \left(c_p(\kappa_1)c_p(\kappa_2) [c_p(\kappa_*)]^{-1} \right) \\ &= \frac{1}{(2\pi)^{s+1}} \left[\frac{\kappa_1^s I_s(2\kappa_1)}{[I_s(\kappa_1)]^2} + \frac{\kappa_2^s I_s(2\kappa_2)}{[I_s(\kappa_2)]^2} - 2 \frac{\kappa_1^s \kappa_2^s I_s(\kappa_*)}{I_s(\kappa_1) I_s(\kappa_2)} \right] \end{aligned}$$

4.4.2 L_2 distance between two vMF Mixtures

We finish the L_2 distance investigation, by giving said distance between two vMF mixture models. To accomplish this, we only need the standard calculus machinery and to apply Lemma 4.4.1.

Proposition 4.4.2. *Define two vMF mixture distributions by,*

$$h_1(\mathbf{x}) = \sum_{i=1}^k p_i f_i(\mathbf{x}; \boldsymbol{\mu}_i, \kappa_i) \quad \text{and} \quad h_2(\mathbf{x}) = \sum_{j=1}^l q_j g_j(\mathbf{x}; \boldsymbol{\mu}_j, \kappa_j), \quad (4.1)$$

with the usual constraints, $p_i \geq 0$ for $i = 1, \dots, k$, $q_j \geq 0$, for $j = 1, \dots, l$, $\sum_{i=1}^k p_i = \sum_{j=1}^l q_j = 1$,

and component distributions,

$$f_i(\mathbf{x}) = \text{vMF}(\mathbf{x}; \boldsymbol{\mu}_i, \kappa_i) \quad \text{and} \quad g_j(\mathbf{x}) = \text{vMF}(\mathbf{x}; \boldsymbol{\mu}_j, \kappa_j),$$

for $\mathbf{x} \in \mathbb{S}^{p-1}$, $\boldsymbol{\mu}_i, \boldsymbol{\mu}_j \in \mathbb{S}^{p-1}$, $\kappa_i \geq 0$, $\kappa_j \geq 0$ for $i = 1, \dots, k$ and $j = 1, \dots, l$ respectively.

Then, the L_2 distance between $h_1(\mathbf{x})$ and $h_2(\mathbf{x})$ is given by,

$$\begin{aligned} & \int_{\mathbb{S}^{p-1}} (h_1(\mathbf{x}) - h_2(\mathbf{x}))^2 d\mathbf{x} \\ &= \sum_{i=1}^k \sum_{i'=1}^k p_i p_{i'} \frac{\kappa_i^s \kappa_{i'}^s I_s(\kappa_{ii'}^*)}{I_s(\kappa_i) I_s(\kappa_{i'})} + \sum_{j=1}^l \sum_{j'=1}^l q_j q_{j'} \frac{\kappa_j^s \kappa_{j'}^s I_s(\kappa_{jj'}^*)}{I_s(\kappa_j) I_s(\kappa_{j'})} - 2 \sum_{i=1}^k \sum_{j=1}^l p_i q_j \frac{\kappa_i^s \kappa_j^s I_s(\kappa_{ij}^*)}{I_s(\kappa_i) I_s(\kappa_j)} \end{aligned}$$

where $s = p/2 - 1$, $I_p(\cdot)$ is the modified Bessel function of the first kind, order p , and $\kappa_{ii'}^*$, $\kappa_{jj'}^*$, and κ_{ij}^* are defined in the same way as κ_* in Lemma 4.4.1, for the respective component pairs.

Proof:

$$\begin{aligned} & \int_{\mathbb{S}^{p-1}} (h_1(\mathbf{x}) - h_2(\mathbf{x}))^2 d\mathbf{x} \\ &= \int_{\mathbb{S}^{p-1}} (h_1(\mathbf{x}))^2 d\mathbf{x} + \int_{\mathbb{S}^{p-1}} (h_2(\mathbf{x}))^2 d\mathbf{x} - 2 \int_{\mathbb{S}^{p-1}} h_1(\mathbf{x}) h_2(\mathbf{x}) d\mathbf{x} \\ &= \sum_{i=1}^k \sum_{i'=1}^k p_i p_{i'} \int_{\mathbb{S}^{p-1}} f_i(\mathbf{x}) f_{i'}(\mathbf{x}) d\mathbf{x} + \sum_{j=1}^l \sum_{j'=1}^l q_j q_{j'} \int_{\mathbb{S}^{p-1}} g_j(\mathbf{x}) g_{j'}(\mathbf{x}) d\mathbf{x} - \\ & \quad 2 \sum_{i=1}^k \sum_{j=1}^l p_i q_j \int_{\mathbb{S}^{p-1}} f_i(\mathbf{x}) g_j(\mathbf{x}) d\mathbf{x} \\ &= \sum_{i=1}^k \sum_{i'=1}^k p_i p_{i'} \frac{\kappa_i^s \kappa_{i'}^s I_s(\kappa_{ii'}^*)}{I_s(\kappa_i) I_s(\kappa_{i'})} + \sum_{j=1}^l \sum_{j'=1}^l q_j q_{j'} \frac{\kappa_j^s \kappa_{j'}^s I_s(\kappa_{jj'}^*)}{I_s(\kappa_j) I_s(\kappa_{j'})} - 2 \sum_{i=1}^k \sum_{j=1}^l p_i q_j \frac{\kappa_i^s \kappa_j^s I_s(\kappa_{ij}^*)}{I_s(\kappa_i) I_s(\kappa_j)} \end{aligned}$$

4.4.3 KL divergence between two vMF models and its symmetric version

KL divergence between two vMF models

Extending our results above, from the unit circle in two-dimensions, to the von Mises-Fisher model on the unit hyper-sphere.

Proposition 4.4.3. *For any two vMF distributions $f \sim \text{vMF}(\mathbf{x}; \boldsymbol{\mu}_1, \kappa_1)$ and $g \sim \text{vMF}(\mathbf{x}; \boldsymbol{\mu}_2, \kappa_2)$, where, $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2$, and $\mathbf{x} \in \mathbb{S}^{p-1}$, the Kullback-Liebler divergence measure is given by,*

$$D_{KL}(f, g) = s \log(\kappa_1) - s \log(\kappa_2) - \log(I_s(\kappa_1)) + \log(I_s(\kappa_2)) + \kappa_1 - \kappa_2(\boldsymbol{\mu}_2 \cdot \boldsymbol{\mu}_1)$$

where $s = p/2 - 1$, $I_\nu(\kappa)$ is the modified Bessel function of the first kind.

Proof: Let $f(\mathbf{x}; \boldsymbol{\mu}_1, \kappa_1)$ and $g(\mathbf{x}; \boldsymbol{\mu}_2, \kappa_2)$ be two independent vMF distributions, for mean vectors $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathbb{S}^{p-1}$.

Then the KL divergence between f and g is given by,

$$\begin{aligned} D_{KL}(f, g) &= \int_{\mathbb{S}^{p-1}} \log\left(\frac{f(\mathbf{x})}{g(\mathbf{x})}\right) f(\mathbf{x}) d\mathbf{x} \\ &= \int_{\mathbb{S}^{p-1}} \log(f(\mathbf{x})) f(\mathbf{x}) d\mathbf{x} - \int_{\mathbb{S}^{p-1}} \log(g(\mathbf{x})) f(\mathbf{x}) d\mathbf{x} \\ &= s \log(\kappa_1) - \log(I_s(\kappa_1)) + \int_{\mathbb{S}^{p-1}} \kappa_1(\mathbf{x} \cdot \boldsymbol{\mu}_1) f(\mathbf{x}) d\mathbf{x} - \\ &\quad s \log(\kappa_2) + \log(I_s(\kappa_2)) - \int_{\mathbb{S}^{p-1}} \kappa_2(\mathbf{x} \cdot \boldsymbol{\mu}_2) f(\mathbf{x}) d\mathbf{x} \\ &= s \log(\kappa_1) - \log(I_s(\kappa_1)) + \kappa_1 \boldsymbol{\mu}_1^T \int_{\mathbb{S}^{p-1}} \mathbf{x} f(\mathbf{x}) d\mathbf{x} - \\ &\quad s \log(\kappa_2) + \log(I_s(\kappa_2)) - \kappa_2 \boldsymbol{\mu}_2^T \int_{\mathbb{S}^{p-1}} \mathbf{x} f(\mathbf{x}) d\mathbf{x} \\ &= s \log(\kappa_1) - \log(I_s(\kappa_1)) + \kappa_1(\boldsymbol{\mu}_1 \cdot \boldsymbol{\mu}_1) - \\ &\quad s \log(\kappa_2) + \log(I_s(\kappa_2)) - \kappa_2(\boldsymbol{\mu}_2 \cdot \boldsymbol{\mu}_1) \\ &= s \log(\kappa_1) - \log(I_s(\kappa_1)) + \kappa_1 - s \log(\kappa_2) + \log(I_s(\kappa_2)) - \kappa_2(\boldsymbol{\mu}_2 \cdot \boldsymbol{\mu}_1), \end{aligned}$$

verifying the proposition above.

A Symmetric KL divergence between two vMF models

Extending the simple SKL divergence to the vMF distribution,

Proposition 4.4.4. *The symmetric KL divergence, SKL, between any two vMF distributions $f \sim \text{vMF}(\boldsymbol{\mu}_1, \kappa_1)$ and $g \sim \text{vMF}(\boldsymbol{\mu}_2, \kappa_2)$, where, $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \in \mathbb{R}^p$ are of unit length, so that they lie on the unit hypersphere \mathbb{S}^{p-1} , the SKL divergence measure is given by,*

$$SKL(f, g) = (\kappa_1 + \kappa_2) (1 - (\boldsymbol{\mu}_1 \cdot \boldsymbol{\mu}_2))$$

Proof:

$$\begin{aligned} SKL(f, g) &= D_{KL}(f, g) + D_{KL}(g, f) \\ &= s \log(\kappa_1) - \log(I_s(\kappa_1)) + \kappa_1 - s \log(\kappa_2) + \log(I_s(\kappa_2)) - \kappa_2(\boldsymbol{\mu}_2 \cdot \boldsymbol{\mu}_1) + \\ &\quad s \log(\kappa_2) - \log(I_s(\kappa_2)) + \kappa_2 - s \log(\kappa_1) + \log(I_s(\kappa_1)) - \kappa_1(\boldsymbol{\mu}_1 \cdot \boldsymbol{\mu}_2) \\ &= (\kappa_1 + \kappa_2) (1 - (\boldsymbol{\mu}_1 \cdot \boldsymbol{\mu}_2)) \end{aligned}$$

giving us the desired result.

4.5 Data: Into the Expanse of the Twitter world

The data set we are working with is a subset of a larger "Twitter" data collected in real time, from October 14, 2016 until November 24, 2016. Our subset contains data collected on just one day, the November 8th, 2016, which was the day of the last United States Presidential election. Twitter allows free collection of public tweets, provided the collector has an account. All of the reported data has been anonymized to some extent, as we are only collecting the actual text

of the tweet. To capture the data, I used the Twitter *firehose* API, with a list of keywords that pertained to the 2016 United States Presidential election.

Over the course of the collection period, we managed to capture more than 100 million tweets, which we have now stored on a MySQL database, for ease of retrieval and scalability. Given the extremely large corpus size (or even the relatively large corpora sizes, if we consider the dataset a collection of corpora over time), the data is *very...very* sparse. This lends itself well to the spherical representation. Also, one would expect the topics to evolve over time. Political pundits use the phrase momentum swing, but can we capture such swings in the data? Is there a statistically significant change in model parameters as the focal point of the national conversation surrounding the election, changes? Can we detect and model any changes, whether that is weekly, daily, or hourly, as the news cycle refreshes itself? Can we detect *shocks* to the twitter-verse. For example, can we identify the effect of James Comey's announcement on the second round of investigations into H. Clinton's email server, and if so, what were any lasting effects in the two weeks leading up to the election?

Is the contentious and extremist/nationalistic rhetoric used by the respective campaigns echoed in the different clusters of tweets. Furthermore, can we identify features of different clusters that provide statistically significant evidence of affiliation with one campaign or ideology over another?

Finally, we ask if we can identify change points in our model that might provide quantitative evidence regarding the impending election. This last point is interesting from both an empirical and sociological perspective, as very nearly every prediction about the election results, went awry.

To assess this, we will look at the L_2 distance between the fitted mixtures for successive hours, over the twenty-four hour period of November 8th. One issue of note is the extremely large vocabulary of our corpus which contains over 1.2 million terms. However, given the abbreviations, colloquialisms, and slang terms inherent in twitter data, we need to find terms

that are meaningful for differentiating documents and identifying groups within the larger set, so after appropriate pre-processing of the data, we can and do remove many of these uninformative terms.

Further, for the purposes of this work, we will pare down the vocabulary further to a *toy* example size, ensuring the problem is tractable with respect to computational requirements, so that we are able to run the job on a local machine, instead of on a large cluster of processors. We were able to remove extremely sparse terms, where we set the sparsity threshold to 0.975, which means that we only keep vocabulary terms that appear in at least $D * (1 - 0.975)$ documents. This does result in some documents becoming empty, once their respective vocabulary terms are excluded, and as such, we remove such documents as well. From our initial 5,781,560 documents, we retain 3,361,574 tweets, with a vocabulary of size $p = 23$. In Table 4.1, we give our final example dictionary and the associated term frequency - inverse document frequency (tf-idf) weights,

In Figure 4.1, we plot the volume of tweets collected each hour, for both the *raw* number of tweets represented by the upper line (gold points and blue line) as well as the volume after reducing the vocabulary and removing empty documents represented by the lower line (blue points with the gold line).

We can see that the hour-by-hour trends are nearly identical volume-wise, with the only difference being the magnitude of the of tweets considered. Finally, in Figure 4.3 we present the L_2 distance between successive hours of tweets, modeled as estimated mixtures of vMF distributions on the hypersphere, \mathbb{S}^{22} .

In Figure 4.2 we see the number of vMF mixture components by hour over the course of the day. To more fully illuminate how that translates to the mixture formulation, we will use the *zero*-hour and first hour, as our h_1 and h_2 from Equation (4.4.2). For the sake of brevity, we will not include the 23-dimensional mean direction vectors in the equations, but do include the mixture proportions and the concentration values, with the mean vectors given in Table 4.2.

| Term | <i>tf-idf</i> Weight |
|-------------|----------------------|
| amp | 3.4896 |
| get | 3.9376 |
| election | 2.5576 |
| like | 3.9635 |
| just | 3.1467 |
| think | 4.3785 |
| will | 3.0584 |
| now | 4.0120 |
| can | 4.2325 |
| people | 3.9808 |
| voted | 4.2897 |
| america | 0.9622 |
| make | 0.9706 |
| today | 0.8906 |
| tomorrow | 1.0104 |
| day | 1.0815 |
| electionday | 1.0065 |
| new | 4.4971 |
| wins | 2.2296 |
| dont | 1.7931 |
| voting | 3.4751 |
| win | 2.8243 |
| president | 1.9347 |

Table 4.1: Selected Vocabulary and *tf-idf* Weights

$$\begin{aligned}
h_1(\mathbf{x}) &= \sum_{i=1}^3 p_i f_i(\mathbf{x}; \hat{\boldsymbol{\mu}}_{f_i}, \hat{\kappa}_i) \\
&= 0.8099 f_1(\mathbf{x}; \hat{\boldsymbol{\mu}}_{f_1}, 6.9644) + \\
&\quad 0.0759 f_2(\mathbf{x}; \hat{\boldsymbol{\mu}}_{f_2}, 66.1739) + \\
&\quad 0.1142 f_3(\mathbf{x}; \hat{\boldsymbol{\mu}}_{f_3}, 32.6826) \\
h_2(\mathbf{x}) &= \sum_{j=1}^2 q_j g_j(\mathbf{x}; \hat{\boldsymbol{\mu}}_{g_j}, \hat{\kappa}_j) \\
&= 0.0825 g_1(\mathbf{x}; \hat{\boldsymbol{\mu}}_{g_1}, 69.2713) + \\
&\quad 0.9175 g_2(\mathbf{x}; \hat{\boldsymbol{\mu}}_{g_2}, 6.9784)
\end{aligned}$$

Notice that the components with extreme concentration values, have small proportion values. These two elements work to offset each other, in some sense, as a single component vMF distribution with a concentration value of 60 or 100 would be a near degenerate distribution,

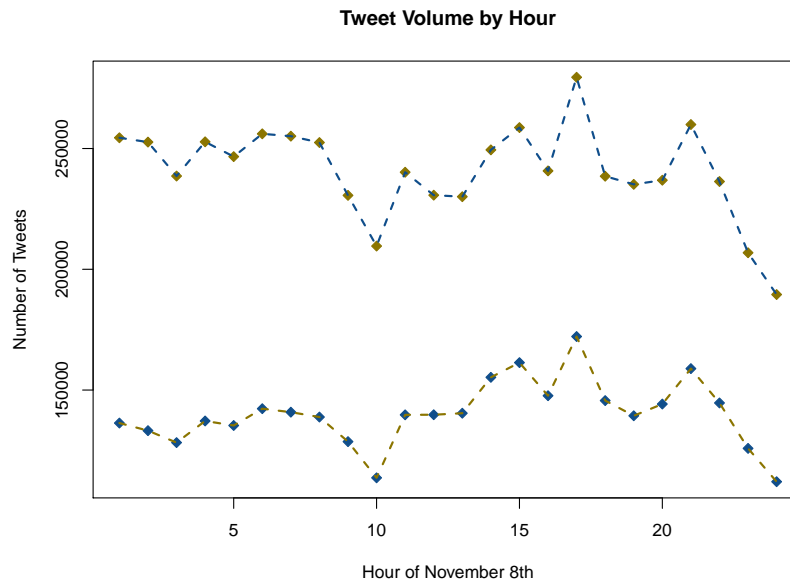


Figure 4.1: Volume of tweets per hour, upper line is the raw volume, lower line is the post vocabulary reduced volume

with all of the mass concentrated at the mean.

Keeping in mind that we are working with the 24-hour clock, the first observation in Figure 4.3 of L_2 model difference is between the vMF model of the 0-hour tweets collected between midnight, 00:00:00 and 00:59:59, and the model of the 1-2 a.m. tweets, collected from 01:00:00 and 01:59:59, which produces a 1-hour offset, we see two smaller *bumps* and one dramatic spike. The first of the bumps seems to correlate with the start of the day, and when the election polls were opening on the east coast, while the second seems to coincide with the end of the day on the east coast, and approximately when the election was called for the eventual winner, Donald Trump, after the contest was called in his favor in Utah and Iowa. The large spike seems to coincide with when the results began to come in, with projections favoring a Trump win in states of Indiana and Kentucky.

One thing to note is that this is still an illustrative subset from the very large data set. Admittedly, with such a small vocabulary, we are looking at election day from a high-level overview,

| Term | $\hat{\mu}_{f_1}$ | $\hat{\mu}_{f_2}$ | $\hat{\mu}_{f_3}$ | $\hat{\mu}_{g_1}$ | $\hat{\mu}_{g_2}$ |
|-------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| amp | 0.4028 | 0.0001 | 0.0042 | 0.0001 | 0.3447 |
| get | 0.2012 | 0.0007 | 0.0021 | 0.0004 | 0.1861 |
| election | 0.1228 | 0.0342 | 0.7556 | 0.0086 | 0.3190 |
| like | 0.2540 | 0.0064 | 0.0008 | 0.0032 | 0.2085 |
| just | 0.1299 | 0.8692 | 0.0030 | 0.8561 | 0.1231 |
| think | 0.1089 | 0.3545 | 0.0019 | 0.3777 | 0.0973 |
| will | 0.1775 | 0.0011 | 0.6517 | 0.0003 | 0.3579 |
| now | 0.1946 | 0.3428 | 0.0010 | 0.3526 | 0.1723 |
| can | 0.2022 | 0.0002 | 0.0012 | 0.0001 | 0.1740 |
| people | 0.2364 | 0.0001 | 0.0024 | 0.0002 | 0.2225 |
| voted | 0.0716 | 0.0002 | 0.0006 | 0.0003 | 0.0809 |
| america | 0.2257 | 0.0000 | 0.0035 | 0.0001 | 0.2001 |
| make | 0.1484 | 0.0000 | 0.0278 | 0.0001 | 0.1335 |
| today | 0.0727 | 0.0001 | 0.0010 | 0.0002 | 0.0570 |
| tomorrow | 0.2949 | 0.0003 | 0.0151 | 0.0000 | 0.2722 |
| day | 0.1247 | 0.0000 | 0.0113 | 0.0000 | 0.0958 |
| electionday | 0.0315 | 0.0000 | 0.0009 | 0.0000 | 0.0295 |
| new | 0.2671 | 0.0001 | 0.0005 | 0.0002 | 0.2020 |
| wins | 0.1163 | 0.0001 | 0.0057 | 0.0001 | 0.1030 |
| dont | 0.2971 | 0.0006 | 0.0010 | 0.0004 | 0.3003 |
| voting | 0.2724 | 0.0005 | 0.0053 | 0.0007 | 0.2801 |
| win | 0.1797 | 0.0003 | 0.0556 | 0.0000 | 0.1561 |
| president | 0.2372 | 0.0002 | 0.0061 | 0.0001 | 0.2071 |

Table 4.2: Vocabulary and Estimated vMF Means for Hours 1 and 2

painted with a broad brush. What is interesting however, is that even at this macro level, changes in our model correlate with events that reflect *on-the-ground* realities.

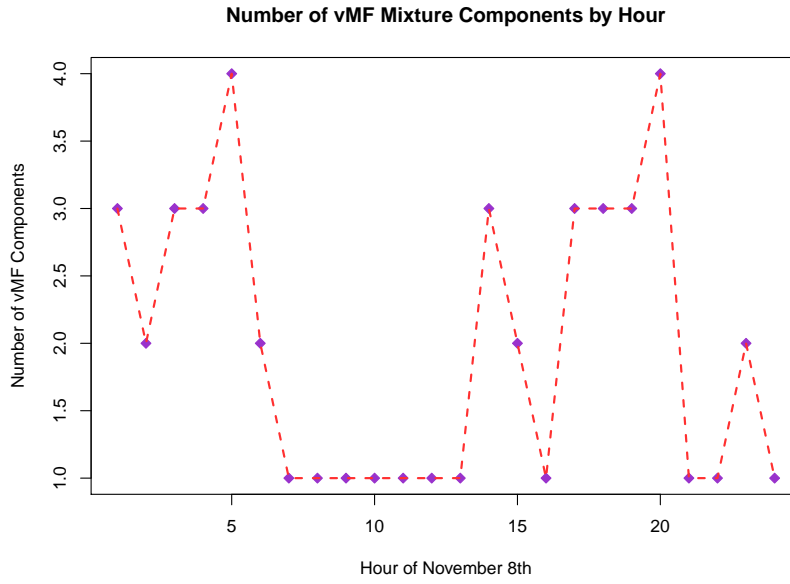


Figure 4.2: Number of vMF Mixture Components by Hour

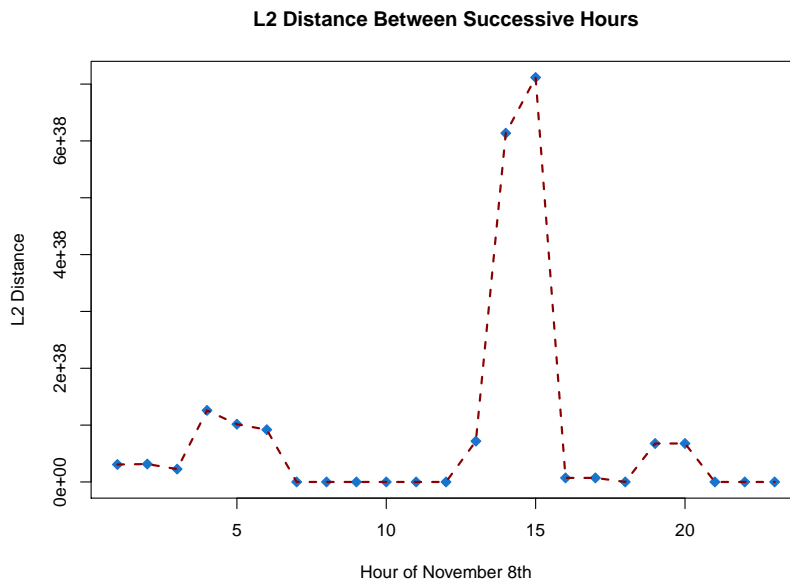


Figure 4.3: L_2 distance between successive hours of vMF models on November 8th

Chapter 5

Simulation and Visualization of Spherical Distributions

5.1 Introduction

In a recent paper titled “*Harmonic analysis and distribution-free inference for spherical distributions*” [4], Jammalamadaka and Terdik consider new spherical models and inference for directional data in 3-dimensions. In this chapter, we consider the simulation, visualization, and fitting of such spherical distributions, in support of the dictum, “A picture is worth a thousand words.” Here, we provide computational algorithms to simulate several spherical models, and in selected cases, also provide alternate and improved algorithms for some existing models, so that a user can compare alternate approaches for generating such random variates. The MATLAB scripts used to generate the figures contained herein are made available as a MATLAB package titled “3D-Directional Statistics, Simulation and Visualization” (3D-Directional-SSV).

Vector space representation provides a very efficient method for displaying and analyzing spherical data, both in the natural world as well as for the theoretical framework. Further practical exploration and analysis in this area depends on related software and associated algorithms,

which is our goal here. We consider spherical distributions on the surface of the unit sphere \mathbb{S}_2 embedded in \mathbb{R}^3 , with the aim of validating and testing the theory presented in [4], as well as in related papers by others on this topic. We develop the necessary computational tools in the form of the MATLAB package “3D-Directional-SSV” to achieve this goal, which we believe will be a valuable resource for scientists interested in statistical analysis of directional data in 3-dimensions.

We start with a broad family of distributions, which we call the “Generalized Fisher-Bingham (**GFB**) family”, and point out the structure and relationship between various subfamilies therein. We review and reconsider earlier work by [21], [22], and a more recent R-package by [23] for the simulation of selected distributions belonging to this **GFB** family. Our MATLAB package makes appropriate modifications needed for our specific context, taking advantage of recent developments in software routines. Just to cite two examples, first our methods help increase the computational efficiency for the equal area projection case of the Kent model in the $\mathbf{GFB}_{5,K}$ family, and second, consider additional parameter scenarios that have not been discussed before.

The main impetus for this computational exploration is in support of the theoretical work given in [4]. Besides covering the entire **GFB** family of distributions, our work provides visualization and simulation tools for new spherical distributions characterized by their spherical harmonics as discussed there. Such visualization of spherical models and plotting of densities and histograms is made possible on the surface of the unit sphere in high relief, as opposed to the more typical *heatmap* representation. The topographical composition of the plots (density, simulated data, and histogram) provide a much clearer graphical representation.

In Section 2, a brief account of the **GFB** family of distributions is given, along with a schematic diagram of the relationships between **GFB** subfamilies. We then describe the inter-relationships for the various subfamilies and the background theory needed for later simulations. Section 3 describes how a histogram for a random sample of observations on the unit sphere \mathbb{S}_2 , may be plotted. Section 4 includes various algorithms for the simulations and visualization,

along with requisite explanations. We provide as supplementary material to this chapter, a comprehensive MATLAB package which contains: (i) MATLAB scripts for the Figures and Algorithms contained in this chapter (ii) A MATLAB package, called “3D-Directional-SSV” containing various MATLAB scripts for generating different spherical models described below, as well as tools for implementing other manipulations on spherical data such as plotting, etc.

5.2 GFB family of distributions

Let

$$\underline{\tilde{x}} = \underline{\tilde{x}}(\vartheta, \varphi) = (\sin \vartheta \cos \varphi, \sin \vartheta \sin \varphi, \cos \vartheta)^\top,$$

represent a point on the surface of the unit sphere \mathbb{S}_2 in \mathbb{R}^3 , with colatitude $\vartheta \in [0, \pi]$ and longitude $\varphi \in [0, 2\pi]$. We consider density functions $f(\underline{\tilde{x}})$ on such an \mathbb{S}_2 .

We shall use two alternate notations for a density viz. $f(\underline{\tilde{x}})$ and $f(\vartheta, \varphi)$, the main difference between them being that $f(\underline{\tilde{x}})$ corresponds to the measure $\Omega(d\underline{\tilde{x}}) = \sin \vartheta d\vartheta d\varphi$, while $f(\vartheta, \varphi)$ includes $\sin \vartheta$ in it, and corresponds to the measure $d\vartheta d\varphi$ on the set $(\vartheta, \varphi) \in [0, \pi] \times [0, 2\pi]$. For instance, if X is a random unit vector which is distributed uniformly on \mathbb{S}_2 (see [24], Section 9.3.1), then the *Uniform* density $f(\underline{\tilde{x}}) = 1/4\pi$, $\underline{\tilde{x}} \in \mathbb{S}_2$ is constant, while at the same time, in terms of the coordinates (Θ, Φ) , the probability density becomes

$$f(\vartheta, \varphi) = \frac{1}{4\pi} \sin \vartheta,$$

so that Θ and Φ are independent, with Θ distributed as $(\sin \vartheta)/2$ on $[0, \pi]$ and Φ is uniform on $[0, 2\pi]$.

A consequence of this to the simulation of random variates on the unit sphere \mathbb{S}_2 is the following. One may simulate a random variate (Θ, Φ) according to the density $f(\vartheta, \varphi)$ and then use $\underline{\tilde{x}}(\Theta, \Phi) = (\sin \Theta \cos \Phi, \sin \Theta \sin \Phi, \cos \Theta)^\top$ for a random point on \mathbb{S}_2 , or simulate

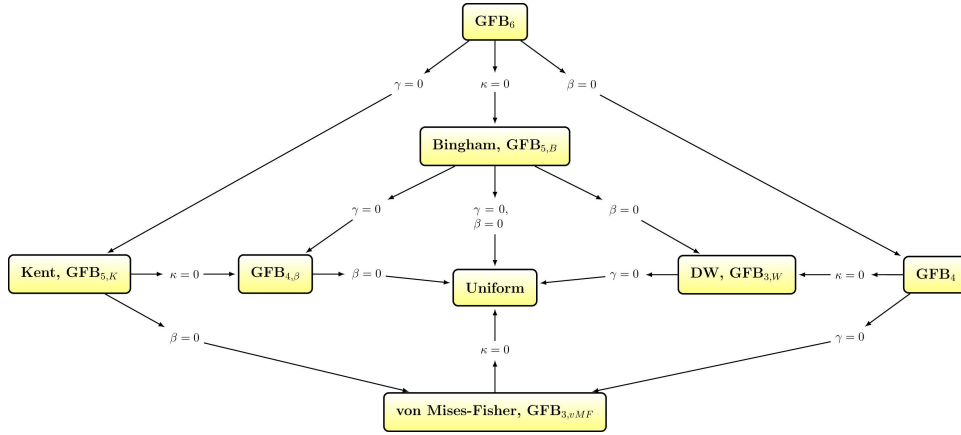


Figure 5.1: Relationships Between GFB Family Distributions

directly a random variate $\tilde{\underline{X}} \in \mathbb{S}_2$ according to the density $f(\tilde{\underline{x}})$.

A distribution is considered *uni-*, *bi-*, and *multi-modal*, if it has one, two, or more modes. In case there are two modes which are diagonally opposite, i.e. there is an axis joining the two modes, we call this distribution axial or *bipolar*. If the mass is concentrated around the main circle, it will be called a *girdle distribution*. Examples that follow, will make these notions clear.

One of the earliest and commonly used models for spherical data is the (unimodal) von Mises-Fisher distribution [25] which can be extended to an antipodally symmetric model, as given by the Dimroth [26] and Watson [27]. Such antipodally symmetric models were further generalized by Bingham [28], and these models, belonging to a particular set of exponential type distributions, shall be called the **GFB** family. It has several subfamilies according to the number of parameters, the Figure 5.1 shows a Diagram of Relationships between **GFB** Families of Distributions.

The most general **GFB** family, which we shall call **FB**₈, contains 8 parameters, and can be represented by the density [28],

$$f_8(\tilde{\underline{x}}; \tilde{\underline{\mu}}_0, \kappa, A) \cong \exp\left(\kappa \tilde{\underline{\mu}}_0 \cdot \tilde{\underline{x}} + \tilde{\underline{x}}^T A \tilde{\underline{x}}\right), \quad (5.1)$$

where \cong denotes equality up to a multiplicative constant and where $\underline{\tilde{\mu}}_0 \in \mathbb{S}_2$, A is a symmetric 3×3 matrix and $\underline{\tilde{\mu}}_0 \cdot \underline{\tilde{x}}$ denotes the usual inner product. Matrix A has the form $A = M Z M^\top$, where $M = \begin{bmatrix} \underline{\tilde{\mu}}_1 & \underline{\tilde{\mu}}_2 & \underline{\tilde{\mu}}_3 \end{bmatrix}$ is an orthogonal matrix and $Z = \text{diag}(\zeta_1, \zeta_2, \zeta_3)$ is a diagonal matrix.

Letting ζ be an arbitrary constant, we can show that changing the matrix A to $A_1 = M Z_1 M^\top$ in (5.1), where

$$Z_1 = \text{diag}(\zeta_1 - \zeta, \zeta_2 - \zeta, \zeta_3 - \zeta), \quad (5.2)$$

the density (5.1) will not change. To avoid identifiability problems it is necessary to impose some constraints. Setting $\zeta = (\zeta_1 + \zeta_2 + \zeta_3) / 3$, one usually assumes that

$$\text{tr}(A) = \zeta_1 + \zeta_2 + \zeta_3 = 0. \quad (5.3)$$

Under this assumption $\zeta_3 = -\zeta_1 - \zeta_2$, i.e. Z is given by the two parameters ζ_1 and ζ_2 .

The density (5.1) has 8 parameters, namely

1. the parameter κ ,
2. two angles $(\vartheta_\mu, \varphi_\mu)$, which define $\underline{\tilde{\mu}}_0 = (\sin \vartheta_\mu \cos \varphi_\mu, \sin \vartheta_\mu \sin \varphi_\mu, \cos \vartheta_\mu)^\top$,
3. the two eigenvalues ζ_1, ζ_2 , of A , and
4. three Euler angles which define the orthogonal matrix M , considered as a rotation matrix.

Using the terminology of Kent ([22]), this is the **GFB**₈ model, and has the density (5.1) of the form,

$$f_8(\underline{\tilde{x}}; \underline{\tilde{\mu}}_0, \kappa, \zeta_1, \zeta_2, M) \cong \exp\left(\kappa \underline{\tilde{\mu}}_0 \cdot \underline{\tilde{x}} + \sum_{k=1}^3 \zeta_k (\underline{\tilde{\mu}}_k \cdot \underline{\tilde{x}})^2\right).$$

If we set $\zeta = (\zeta_1 + \zeta_2) / 2$ in (5.2), an equivalent form of (5.1) can be derived such that the

density is given by,

$$f_8(\underline{\tilde{x}}; \kappa, \beta, \gamma, \underline{\tilde{\mu}}_0, M) \cong \exp\left(\kappa \underline{\tilde{\mu}}_0 \cdot \underline{\tilde{x}} + \gamma \left(\underline{\tilde{\mu}}_3 \cdot \underline{\tilde{x}}\right)^2 + \beta \left(\left(\underline{\tilde{\mu}}_1 \cdot \underline{\tilde{x}}\right)^2 - \left(\underline{\tilde{\mu}}_2 \cdot \underline{\tilde{x}}\right)^2\right)\right). \quad (5.4)$$

Here we shall consider densities of the form (5.4). In this form, the set of parameters κ, β , and γ satisfy: $\kappa \in \mathbb{R}$, $\beta \geq 0$, $\gamma \in \mathbb{R}$.

Remark 3. Notice that we are allowing for $\kappa \in \mathbb{R}$ — however a sign change of κ is equivalent to changing either $\underline{\tilde{\mu}}_0$ to $-\underline{\tilde{\mu}}_0$, or $\underline{\tilde{x}}$ to $-\underline{\tilde{x}}$, since the quadratic form in the density (5.4) is invariant under the transformation $\underline{\tilde{x}}$ to $-\underline{\tilde{x}}$. The sign change of κ will interchange hemispheres only, therefore modulus $|\kappa|$, in some cases, can be considered as the parameter of concentration. If $\kappa \leq 0$, then

$$\begin{aligned} f_8(\underline{\tilde{x}}; \kappa, \beta, \gamma, \underline{\tilde{\mu}}_0, M) &= f_8(-\underline{\tilde{x}}; -\kappa, \beta, \gamma, \underline{\tilde{\mu}}_0, M) \\ &= f_8(\underline{\tilde{x}}; -\kappa, \beta, \gamma, -\underline{\tilde{\mu}}_0, M). \end{aligned}$$

In terms of colatitude and longitude one can use the transformation $\vartheta \rightarrow \pi - \vartheta$, and $\varphi \rightarrow \varphi + \pi$, to change $\underline{\tilde{x}} \rightarrow -\underline{\tilde{x}}$.

Although we mention the \mathbf{GFB}_8 distribution here for completeness, in practice convenient restrictions discussed in the next section, will allow us to reduce the number of parameters without loss of any generality or flexibility.

5.2.1 Model \mathbf{GFB}_6

We apply a restriction on the \mathbf{GFB}_8 by setting either $\underline{\tilde{\mu}}_0 = \underline{\tilde{\mu}}_3$, or by assuming $\underline{\tilde{\mu}}_0$ is collinear to $\underline{\tilde{\mu}}_3$ in (5.4). In this way, we can reduce the number of parameters by two i.e. by the two angles

$(\vartheta_{\mu_0}, \varphi_{\mu_0})$, which define $\underline{\tilde{\mu}}_0$. The resulting **GFB**₆ model has a density given by,

$$f_6(\underline{\tilde{x}}; \kappa, \beta, \gamma, M) \cong \exp\left(\kappa \underline{\tilde{\mu}}_3 \cdot \underline{\tilde{x}} + \gamma \left(\underline{\tilde{\mu}}_3 \cdot \underline{\tilde{x}}\right)^2 + \beta \left(\left(\underline{\tilde{\mu}}_1 \cdot \underline{\tilde{x}}\right)^2 - \left(\underline{\tilde{\mu}}_2 \cdot \underline{\tilde{x}}\right)^2\right)\right). \quad (5.5)$$

We can transform the original xyz system into an orthogonal system defined by $M = [\underline{\tilde{\mu}}_1, \underline{\tilde{\mu}}_2, \underline{\tilde{\mu}}_3]$, via two rotations. The first rotation, $G_{\underline{\tilde{\mu}}_3, \underline{\tilde{N}}}$, will rotate the North pole $\underline{\tilde{N}} = (0, 0, 1)^\top$ to $\underline{\tilde{\mu}}_3$, i.e. $G_{\underline{\tilde{\mu}}_3, \underline{\tilde{N}}} \underline{\tilde{N}} = \underline{\tilde{\mu}}_3$ which can be constructed in the following way. The vectors $\underline{\tilde{N}}$ and $\underline{\tilde{\mu}}_3$ define a plane with normal vector $\underline{\tilde{N}} \times \underline{\tilde{\mu}}_3$ (cross product). Now, let $\underline{\tilde{N}} \times \underline{\tilde{\mu}}_3$ be the axis of rotation and rotate $\underline{\tilde{N}}$ to $\underline{\tilde{\mu}}_3$. This rotation depends solely on $\underline{\tilde{\mu}}_3$, i.e. by the two angles ϑ_μ and φ_μ of $\underline{\tilde{\mu}}_3$. When we rotate the sphere by $G_{\underline{\tilde{\mu}}_3, \underline{\tilde{N}}}$, the plane of rotated x - and y - axes coincides with the plane defined by $\underline{\tilde{\mu}}_1$ and $\underline{\tilde{\mu}}_2$, since $\underline{\tilde{\mu}}_3$ is now perpendicular to both planes. Using $\underline{\tilde{\mu}}_3$ as the axis of rotation, next we rotate the sphere by an angle $\psi \in [0, 2\pi]$ so that $\underline{\tilde{\mu}}_1$ coincides with the rotated x -axis, and $\underline{\tilde{\mu}}_2$ coincides with the rotated y -axis. We will denote this rotation by $G_{\underline{\tilde{\mu}}_1, \underline{\tilde{\mu}}_2, xy}$. These three angles ϑ_μ , φ_μ and ψ , called Euler angles, characterize the rotation from the original orthogonal system to the orthogonal system $M = [\underline{\tilde{\mu}}_1, \underline{\tilde{\mu}}_2, \underline{\tilde{\mu}}_3]$.

Note here that the rotations from one system to another one are not unique and in our simulations we shall apply the above rotation for the densities, their simulations, and histograms.

The six parameters of Model **GFB**₆ are then κ, β, γ and M . If $\underline{\tilde{\mu}}_3$ is known we can apply $G_{\underline{\tilde{\mu}}_3, \underline{\tilde{N}}}$ and rotate it to the North pole $\underline{\tilde{N}}$ and (5.5) will have the form

$$f_6(\underline{\tilde{x}}; \kappa, \beta, \gamma, \underline{\tilde{N}}, \psi) \cong \exp\left(\kappa \tilde{x}_3 + \gamma \tilde{x}_3^2 + \beta \left(\left(\underline{\tilde{\mu}}_1^1 \cdot \underline{\tilde{x}}\right)^2 - \left(\underline{\tilde{\mu}}_2^1 \cdot \underline{\tilde{x}}\right)^2\right)\right),$$

where again, $\underline{\tilde{x}} = (\tilde{x}_1, \tilde{x}_2, \tilde{x}_3)$, and $\underline{\tilde{\mu}}_k^1$ are the transformed $\underline{\tilde{\mu}}_k$, by $G_{\underline{\tilde{\mu}}_3, \underline{\tilde{N}}}$. Applying $G_{\underline{\tilde{\mu}}_1, \underline{\tilde{\mu}}_2, xy}$, the density in the new coordinate system is

$$f_6(\underline{\tilde{x}}; \kappa, \beta, \gamma, \underline{\tilde{N}}, 0) \cong e^{\kappa \tilde{x}_3 + \gamma \tilde{x}_3^2 + \beta(\tilde{x}_1^2 - \tilde{x}_2^2)} = e^{\kappa \cos \vartheta + \gamma \cos^2 \vartheta + \beta \sin^2 \vartheta \cos 2\varphi}.$$

For simplicity, if $\underline{\tilde{\mu}}_3 = \underline{\tilde{N}}$ and $\psi = 0$, we introduce the notation

$$f_6(\underline{\tilde{x}}; \kappa, \beta, \gamma) = f_6(\underline{\tilde{x}}; \kappa, \beta, \gamma, 0, \underline{\tilde{N}}),$$

and call it the *canonical form*.

The mean direction $\underline{\tilde{\mu}}$ of the model \mathbf{GFB}_6 , given by (5.5), is characterized by a constant times $\kappa M^\top \underline{\tilde{N}}$, in particular if $\kappa = 0$, then $\underline{\tilde{\mu}}$ is undefined ($\underline{\tilde{\mu}} = 0$, see [4] for details).

Remark 4. *The Model \mathbf{GFB}_6 is rotationally symmetric about the axis $\underline{\tilde{N}}$ if and only if $\beta = 0$ (see [4]). An example of rotational symmetry of \mathbf{GFB}_6 , when $\beta \neq 0$, is given in Remark 7, where the axis of rotation is the y -axis.*

From now on we consider densities in the canonical form and simulate random variates when $\underline{\tilde{\mu}}_3 = \underline{\tilde{N}}$ and $\psi = 0$. If we are given a matrix of rotation $M = \begin{bmatrix} \underline{\tilde{\mu}}_1 & \underline{\tilde{\mu}}_2 & \underline{\tilde{\mu}}_3 \end{bmatrix}$, then we apply this rotation as the last step of the simulation. As we have seen above, this rotation can be given by $\underline{\tilde{\mu}}_3$ and angle ψ , such that we rotate $\underline{\tilde{N}}$ to $\underline{\tilde{\mu}}_3$, then use $\underline{\tilde{\mu}}_3$ as the axis of rotation and rotate the sphere by angle ψ . This is the reverse rotation from what we described above. Formally, we apply the product of the two rotations $G_{\underline{\tilde{N}}, \underline{\tilde{\mu}}_3} \cdot G_{xy, \underline{\tilde{\mu}}_{1,2}}$. (The product of two rotations $G_{\underline{\tilde{\mu}}_3, \underline{\tilde{N}}} \cdot G_{\underline{\tilde{\mu}}_{1,2}, xy}$ is again, a rotation.)

Considering the density under the transformation $x = \cos \vartheta$ with the derivative $-dx = \sin \vartheta d\vartheta$, as is done in [21],

$$\tilde{x}_1 = \sqrt{1-x^2} \cos \varphi, \quad \tilde{x}_2 = \sqrt{1-x^2} \sin \varphi, \quad \tilde{x}_3 = x, \quad (5.6)$$

we obtain what we will hereafter refer to as, the *basic* form of the density f_6 , given by,

$$\begin{aligned} g_6(x, \varphi; \kappa, \beta, \gamma) &\cong e^{\kappa x + \gamma x^2 + \beta(1-x^2) \cos 2\varphi} \\ &= e^{\beta(1-x^2) \cos 2\varphi} e^{\kappa x + \gamma x^2}. \end{aligned} \quad (5.7)$$

This is the joint density in x and φ , and can be considered as the product of two densities, where $\exp(\kappa x + \gamma x^2)$ is a density in x , and $\exp(\beta(1-x^2)\cos 2\varphi)$ is the conditional density of φ given x . More precisely

$$g_6(x, \varphi; \kappa, \beta, \gamma) = g_{\Phi|X}(\varphi|x; \beta) g_{6,X}(x; \kappa, \gamma),$$

where

$$g_{6,X}(x; \kappa, \beta, \gamma) \cong \exp(\kappa x + \gamma x^2) \int_0^{2\pi} e^{\beta(1-x^2)\cos 2\varphi} d\varphi = I_0(\beta(1-x^2)) e^{\kappa x + \gamma x^2}.$$

It follows from (5.7) that

$$g_{\Phi|X}(\varphi|x; \beta) \cong e^{\beta(1-x^2)\cos 2\varphi}. \quad (5.8)$$

The densities $g_{6,X}(x; \kappa, \beta, \gamma)$ and $g_{\Phi|X}(\varphi|x; \beta)$ will be called the *marginal density*, and *conditional density* respectively. Since this conditional density $g_{\Phi|X}$ remains the same throughout this discussion, we do not refer to the specific subfamily to which it corresponds.

Remark 5. When $X = x$ is given, the conditional density (5.8) follows vMF distribution (see Section 5.2.6) with parameter $\beta(1-x^2)$ and is a function of 2φ . It is worth mentioning that $g_{\Phi|X}(\varphi|x; \beta)$ is not a vMF distribution on the unit sphere \mathbb{S}_2 since the $\sin 2\varphi$ component is missing from (5.8), but it is vMF distributed on the unit circle \mathbb{S}_1 .

We now consider special cases of Model **GFB**₆, when one of the basic parameters κ, β, γ is zero. Although the densities in these cases follow directly from the densities f_6 and g_6 , we list them individually in order of decreasing complexity. The algorithms for the simulation of these models will be given in Section 5.4, in reverse order, incrementally increasing the number of parameters and model complexity.

5.2.2 Kent Model $\mathbf{GFB}_{5,K}$

Let $\gamma = 0$, in (5.5). Then, the number of parameters in $f_6(\underline{\tilde{x}}; \kappa, \beta, \gamma, M)$ is reduced by one (i.e. by parameter γ), and has the form

$$f_5(\underline{\tilde{x}}; \kappa, \beta, M) \cong \exp\left(\kappa \underline{\tilde{\mu}}_3 \cdot \underline{\tilde{x}} + \beta \left(\left(\underline{\tilde{\mu}}_1 \cdot \underline{\tilde{x}} \right)^2 - \left(\underline{\tilde{\mu}}_2 \cdot \underline{\tilde{x}} \right)^2 \right)\right), \quad (5.9)$$

which defines the five parameter model $\mathbf{GFB}_{5,K}$. The five parameters are κ, β , and M , with canonical form

$$f_5(\underline{\tilde{x}}; \kappa, \beta) \cong e^{\kappa \tilde{x}_3 + \beta(\tilde{x}_1^2 - \tilde{x}_2^2)}.$$

In terms of ϑ and φ

$$f_5(\underline{\tilde{x}}; \kappa, \beta) \cong e^{\kappa \cos \vartheta + \beta \sin^2 \vartheta \cos 2\varphi}, \quad (5.10)$$

see [21]. If we apply transformation (5.6) then the corresponding basic form is

$$g_5(x, \varphi; \kappa, \beta) = g_{\Phi|X}(\varphi|x; \beta) g_{5,X}(x; \kappa),$$

where

$$g_{5,X}(x; \kappa, \beta) \cong I_0(\beta(1-x^2)) e^{\kappa x}, \quad (5.11)$$

see (5.8) for conditional density $g_{\Phi|X}$. Furthermore, if $2\beta \leq \kappa$ the distribution is unimodal with mode at $\underline{\tilde{\mu}}_3$, which is the case Kent originally considered in [22]. If $2\beta > \kappa$ the distribution is bimodal, with modes at longitude $\varphi = 0$ and π , with the third coordinate, \tilde{x}_3 , defined by the equation $\cos \vartheta = \kappa/2\beta$ for both modes.

Remark 6. *The MATLAB scripts are all listed in the Supplement, and are part of the package “3D-Directional-SSV”.*

When $2\beta \leq \kappa$, the parameters in this model have the interpretation (see [22])

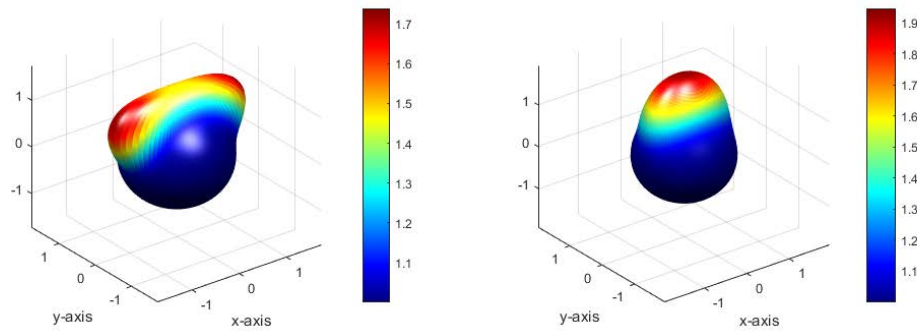


Figure 5.2: Kent density, left: bimodal, $2\beta > \kappa$, modes are at $\vartheta = \pi/4$, and $\varphi = 0, \pi$; right: unimodal, $2\beta \leq \kappa$

1. $|\kappa|$ represents the concentration
2. $\tilde{\underline{\mu}}_3$ is the mean direction (pole or mode)
3. $\tilde{\underline{\mu}}_1$ and $\tilde{\underline{\mu}}_2$ are, respectively, the major and minor axes of constant probability ellipses near the mode, and are determined only up to sign.

Equal area projection and separability of \mathbf{GFB}_6

Equal area projection of spherical densities provide some possibility of simulations on a disk in the plane. As mentioned in [29], in the particular case when $2\beta \leq \kappa$, model $\mathbf{GFB}_{5,K}$ represents Lambert's equal area projection of the sphere

$$y_1 = r \cos \varphi, y_2 = r \sin \varphi, r = \sin(\vartheta/2), \quad (5.12)$$

and separates the density (5.10) by coordinates y_1 and y_2 . The inverse of this transform

$$\cos \vartheta = 1 - 2(y_1^2 + y_2^2), \quad (5.13)$$

$$\sin \varphi = \frac{y_2^2}{\sqrt{y_1^2 + y_2^2}},$$

$$\cos \varphi = \frac{y_1^2}{\sqrt{y_1^2 + y_2^2}},$$

will be useful in simulations.

The transformed Kent model $\mathbf{GFB}_{5,K}$ can be written as

$$f_5(y_1, y_2; \kappa, \zeta_1, \zeta_2) \cong e^{-2(\kappa-2\beta)y_1^2-4\beta y_1^4} e^{-2(\kappa+2\beta)y_2^2+4\beta y_2^4},$$

as shown in the Lemma below. The transformed coordinates thus become independent, and in fact the Lemma also shows that the converse of this statement is true.

Lemma 5.2.1. *Consider the model \mathbf{GFB}_6 with density given in the form*

$$f_6(\underline{\tilde{x}}; \kappa, \zeta_1, \zeta_2) \cong \exp\left(\kappa \tilde{x}_3 + \sum_{k=1}^3 \zeta_k \tilde{x}_k^2\right). \quad (5.14)$$

The equal area projection $f_6(y_1, y_2; \kappa, \zeta_1, \zeta_2)$ of (5.14) will be separable in y_1 and y_2 if and only if $f_6(y_1, y_2; \kappa, \zeta_1, \zeta_2)$ corresponds to the Kent model $\mathbf{GFB}_{5,K}$.

Proof: Let us start from the \mathbf{GFB}_6 density

$$f_6(\underline{\tilde{x}}; \kappa, \zeta_1, \zeta_2) \cong \exp\left(\kappa \tilde{x}_3 + \sum_{k=1}^3 \zeta_k \tilde{x}_k^2\right).$$

Now, applying the equal area projection (5.12), We have

$$\begin{aligned}\tilde{x}_3 &= \cos \vartheta = 1 - 2r^2, \\ \tilde{x}_1^2 &= 4(1 - r^2)y_1^2, \\ \tilde{x}_2^2 &= 4(1 - r^2)y_2^2, \\ \tilde{x}_3^2 &= 1 - 4r^2(1 - r^2)\end{aligned}$$

so that

$$f_6(y_1, y_2; \kappa, \zeta_1, \zeta_2) \cong \exp(-2\kappa r^2 + 4(1 - r^2)(\zeta_1 y_1^2 + \zeta_2 y_2^2 - \zeta_3 r^2)), \quad (5.15)$$

where $r^2 = y_1^2 + y_2^2$. Now, expanding the exponent gives us

$$\begin{aligned}-2\kappa(y_1^2 + y_2^2) + 4\zeta_1 y_1^2 + \zeta_2 y_2^2 - \zeta_3(y_1^2 + y_2^2) \\ - 4\zeta_1 y_1^4 - 4\zeta_2 y_2^4 + 4\zeta_3(y_1^4 + y_2^4) - 4y_1^2 y_2^2(\zeta_1 + \zeta_2 - 2\zeta_3).\end{aligned}$$

It is clear the density (5.15) will be separable in y_1 and y_2 if and only if

$$\zeta_1 + \zeta_2 - 2\zeta_3 = 0. \quad (5.16)$$

The sum $\zeta_1 + \zeta_2 - 2\zeta_3$ is invariant under the transformation (5.2), therefore setting $\zeta = (\zeta_1 + \zeta_2) / 2$, in (5.2), (5.16) implies $\gamma = \zeta_3 - (\zeta_1 + \zeta_2) / 2 = 0$, in other words the model is the Kent model

GFB_{5,K}.

5.2.3 Bingham Model $\text{GFB}_{5,B}$

An alternate way of reducing the number of parameters in (5.5) by one, is by setting $\kappa = 0$. This yields the density, with canonical form

$$f_B(\underline{\tilde{x}}; \beta, \gamma) \cong e^{\gamma \tilde{x}_3^2 + \beta(\tilde{x}_1^2 - \tilde{x}_2^2)} = e^{\gamma \cos^2 \vartheta + \beta \sin^2 \vartheta \cos 2\varphi}. \quad (5.17)$$

Application of transformation (5.6) gives the basic form

$$g_B(x, \varphi; \beta, \gamma) = g_{\Phi|X}(\varphi|x; \beta) g_{B,X}(x; \gamma),$$

where

$$g_{B,X}(x, \beta, \gamma) \cong I_0(\beta(1-x^2)) e^{\gamma x^2},$$

see (5.8) for conditional density $g_{\Phi|X}$. Here, the marginal density $g_{B,X}$ contains the expression $\exp(\gamma x^2)$ which corresponds to the Dimroth-Watson (DW) distribution, defined below in Equation (5.19).

Remark 7. *In the special case when $\beta = \gamma$, the density (5.17) has the form*

$$f_B(\underline{\tilde{x}}; \beta, \gamma) \cong e^{\gamma \tilde{x}_3^2 + \beta(\tilde{x}_1^2 - \tilde{x}_2^2)} \cong e^{-2\gamma \tilde{x}_2^2}. \quad (5.18)$$

This model (5.18) is a rotation of the DW model (see Equation (5.19)) with negative parameter (recall that $\beta > 0$) and therefore (5.18) is a girdle distribution around the main circle $\tilde{x}_2 = 0$.

We now fix β and change γ to consider three scenarios: if $\gamma < \beta$, then it is bipolar with modal direction x -axis, if $\beta = \gamma$, then it is girdle, finally, if $\beta < \gamma$, then the modal direction is the North pole $\underline{\tilde{N}}$, see Figure 5.3.

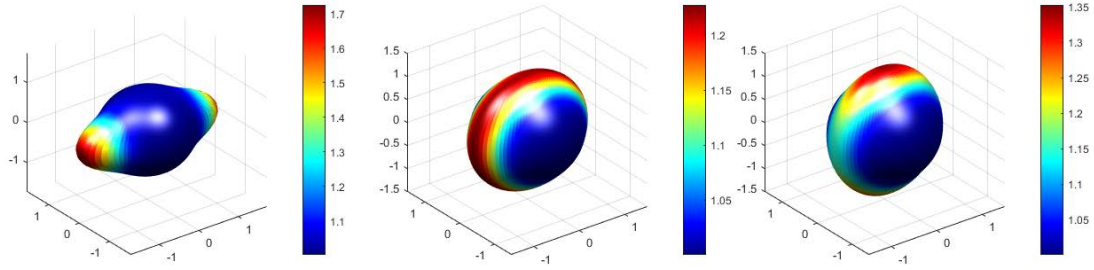


Figure 5.3: Bingham model; $\beta = 3.2$, for all cases, $\gamma = -1.1 < \beta$, $\gamma = 3.2 = \beta$ and $\gamma = 4.1 > \beta$ from left to right respectively, here $\underline{\tilde{\mu}} = (0, 0, 1)^\top$, and $\psi = 0$.

5.2.4 Model $\text{GFB}_{4,\beta}$

We now consider the simplest form when the conditional density $g_{\Phi|X}$ is not constant, namely when the Θ and Φ are not independent random variables. Setting either $\kappa = 0$ and $\gamma = 0$ in (5.5), or $\kappa = 0$ in (5.9), or $\gamma = 0$ in (5.17), we get the density function which is a four-parameter family of distributions, with the canonical form

$$f_{\beta}(\underline{\tilde{x}}; \beta) \cong e^{\beta(\tilde{x}_1^2 - \tilde{x}_2^2)} = e^{\beta \sin^2 \theta \cos 2\varphi}.$$

The basic form

$$g_{\beta}(x, \varphi; \beta) \cong e^{\beta(1-x^2)\cos 2\varphi}$$

coincides formally with the conditional density $g_{\Phi|X}$. Nevertheless, the marginal density is not uniform, since

$$g_{\beta}(x, \varphi; \beta) = g_{\Phi|X}(\varphi|x; \beta) g_{\beta,X}(x, \beta),$$

where

$$g_{\beta,X}(x, \beta) \cong I_0(\beta(1-x^2)),$$

and as mentioned earlier, $g_{\Phi|X}$ follows vMF distribution on the circle see (5.8).

5.2.5 Model \mathbf{GFB}_4

Setting $\beta = 0$, in (5.5) we reduce the number of parameters by two (β , and angle ψ) the result in canonical form is

$$f_4(\underline{\tilde{x}}; \kappa, \gamma) \cong e^{\kappa \tilde{x}_3 + \gamma \tilde{x}_3^2} = e^{\kappa \cos \vartheta + \gamma \cos^2 \vartheta}.$$

This density does not depend on longitude φ . Instead, it depends only on $\cos \vartheta$ and hence, is rotationally symmetric (see Lemma 4 [4]). Transformation (5.6) provides the basic form

$$g_4(x; \kappa, \gamma) \cong e^{\kappa x + \gamma x^2}.$$

A particular case of Model \mathbf{GFB}_4 when $\kappa = 0$, is the following three parameter family:

5.2.6 Dimroth-Watson Model $\mathbf{GFB}_{3,DW}$

The Dimroth-Watson Distribution (DW) [30] distribution is given by the density function,

$$f_W(\underline{\tilde{x}}; \gamma, \underline{\tilde{\mu}}) = \frac{1}{M(1/2, 3/2, \gamma)} e^{\gamma(\underline{\tilde{\mu}} \cdot \underline{\tilde{x}})^2} = \frac{1}{M(1/2, 3/2, \gamma)} e^{\gamma \cos^2 \vartheta}, \quad (5.19)$$

where $M(1/2, 3/2, \gamma)$ is the Kummer function (not to be confused with the orthogonal matrix M described above) and where $\vartheta = \arccos(\underline{\tilde{\mu}} \cdot \underline{\tilde{x}})$, [4]. Either setting $\gamma = 0$ in Model \mathbf{GFB}_4 , or $\beta = 0$ in the Kent Model $\mathbf{GFB}_{5,K}$, we arrive at the most basic model on the sphere viz. the von Mises-Fisher distribution.

Model $\mathbf{GFB}_{3,vMF}$ The density of this widely used von Mises-Fisher Model (**vMF**) is given by

$$f_{vMF}(\underline{\tilde{x}}; \kappa, \underline{\tilde{\mu}}) = \frac{\sqrt{\kappa}}{(2\pi)^{3/2} I_{1/2}(\kappa)} e^{\kappa \underline{\tilde{\mu}} \cdot \underline{\tilde{x}}}, \quad (5.20)$$

with basic form

$$g_{vMF}(x; \kappa) = \frac{\sqrt{\kappa}}{(2\pi)^{3/2} I_{1/2}(\kappa)} e^{\kappa x}.$$

5.3 A Spherical Histogram

A histogram for a random sample on the unit sphere \mathbb{S}_2 assumes an equal-area discretization of the sphere. We consider the **HEALPix** (**H**ierarchical, **E**qual Area and iso **L**atitude **P**ixelization) discretization, a detailed description of which can be found in [31]. In the first step of the pixelization the sphere is partitioned into 12 equal-area spherical quadrilaterals (pixels), and in each subsequent step, all the existing pixels are divided into 4 equal-area quadrilaterals. HEALPix is a discretization with the resolution parameter n_{side} (number of steps of division, which is a power of 2) and total number of pixels equal to $n_{pix} := 12n_{side}^2$, such that $\sum_{k=1}^{n_{pix}} n_k = n$, and each with an area of $4\pi/n_{pix}$.

If we are given a random sample $\tilde{X}_j, j = 1, 2, \dots, n$, on the sphere \mathbb{S}_2 , then for each pixel/quadrilateral $\Pi_k, k = 1, 2, \dots, n_{pix}$ we set an integer n_k which counts the number of the sample elements contained in the pixel Π_k . We use the nested numbering scheme for ordering pixels, where Π_k is the k^{th} pixel according to this scheme. The nested scheme is appropriate for decreasing the resolution, since one can easily accumulate the samples included in 4 ‘neighboring’ pixels.

We define the histogram $H(\tilde{x})$ such that it is constant over a quadrilateral Π_k and the integral over the sphere is 1:

$$H(\tilde{x}) = \frac{n_{pix} n_k}{4\pi n}, \quad \tilde{x} \in \Pi_k.$$

We plot the histogram drawing a column with height $n_{pix} n_k / 4\pi n$ over each quadrilateral. For instance, let $\tilde{X}_j, j = 1, 2, \dots, n = 2^{12}$, be a random sample from the uniform distribution and $n_{side} = 2^3$, so that $n_{pix} = 768$. We plot the sample and the corresponding histogram:

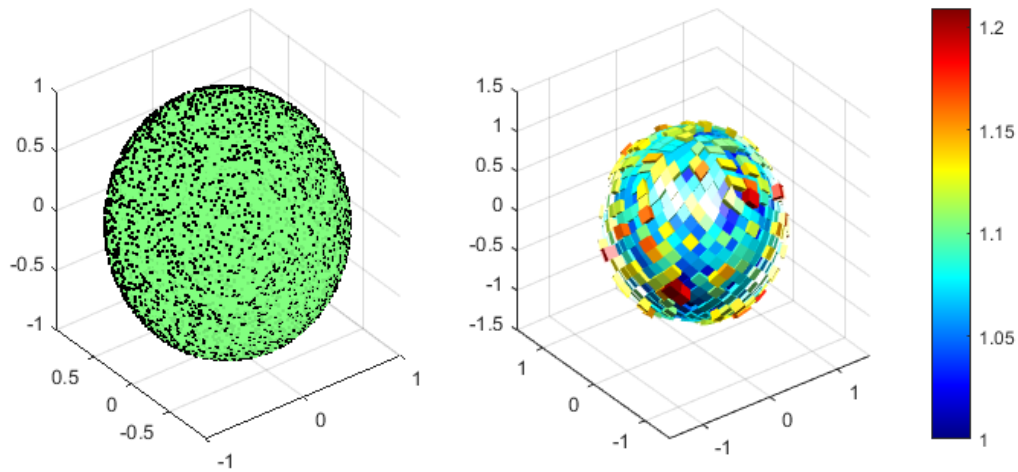


Figure 5.4: Uniform sample and its Histogram

An important feature of the histogram lies in the plot construction. Mathematically, when the columns from each quadrilateral are extruded from the surface of the sphere, the height is measured normal to the center of the pixel, and the volume is given as that height times the area of the quadrilateral, as described above, and as is as expected. However, graphically we also extrude all of the pixel edges normal to their respective location on the sphere. This results in taller columns having larger patches and shorter patches having smaller patches at their respective termini, and stands in contrast to other methods currently in use, that maintain pixel surface area at all elevations, which forces adjacent columns to becoming increasingly separated as column height increases, and results in a *pin cushion* type histogram.

5.4 Simulation of Random Variates on the Sphere

There will be some common features of all simulations below. These are the following:

1. We consider the case $\kappa \geq 0$, in formulae, but also handle the the case $\kappa \leq 0$, during simulation. We simulate a random variate \tilde{X} with $|\kappa|$ and as a final step (but before

- rotation) change the variate $\underline{\tilde{X}}$ to $-\underline{\tilde{X}}$ (see Remark 3).
2. All algorithms below concern the case when the frame of reference is xyz , with the modal/mean direction as the North pole, i.e. $\underline{\tilde{\mu}} = \underline{\tilde{N}}$.
 3. The plots are generated by the following parameters: Sample size is $n = 2^{12} = 8,192$, resolution of the discretized sphere S_2 is $n_{side} = 2^3$, which yields a total pixel count of $n_{pix} = 768$. The resolution for theoretical density plots is 101×101 of the set of angles $(\vartheta, \varphi) \in [0, \pi] \times [0, 2\pi]$.
 4. We characterize a rotation corresponding to $M = \left[\underline{\tilde{\mu}}_1, \underline{\tilde{\mu}}_2, \underline{\tilde{\mu}}_3 \right]$, by a vector $\underline{\tilde{\mu}}$ and an angle ψ , such that $\underline{\tilde{\mu}}_3$ represents $\underline{\tilde{\mu}}$ and $\underline{\tilde{\mu}}_1, \underline{\tilde{\mu}}_2$ defines the angle ψ , see Subsection 5.2.1 for details. See Algorithm 1, in Supplement 7.2 for the case, when a vector $\underline{\tilde{\mu}}$ and an angle ψ are given.

Note, $P(\Psi + \psi \leq x) = P(\Psi \leq x - \psi)$, hence when an angle of a random variate is changed by ψ , then the corresponding density is changing by $-\psi$, see Supplement ?? for more details.

5.4.1 Acceptance-rejection method

For the simulation of random variates from the various distributions presented here, we make frequent use of the standard acceptance-rejection method. For a detailed account of this classic simulation technique, see [32], [33], but a general overview of the method for any particular distribution is as follows.

We wish to sample random variate X from a calculable target distribution with density $f_X(x)$, for which the simulation process is either exceptionally difficult, or perhaps even unknown. Because $f_X(x)$ is calculable, we can write the density $f_X(x) = cg(x)h(x)$, where the constant $c \geq 1$, $0 < g(x) \leq 1$, and $h(x)$ is a probability density function from which we know

how to sample.

The algorithm of this general method based on testing the inequality $U \leq g(Y)$, where U is uniform on the interval $(0, 1)$, and if it is fulfilled, then Y will be accepted as a random variate from $f_X(x)$. The corresponding algorithm is given in appendix 7.2.3, Algorithm 2.

In practice, we seek a density $h(x)$ such that

$$f_X(x) \leq c_1 h(x),$$

where $c_1 \geq 1$, then set $g(x) = f_X(x)/c_1 h(x)$ and denote to the denominator as the *envelope*, and use the above algorithm. Optimizing over algorithmic efficiency, the acceptance ratio $1/c_1$ should be maximized such that the constant c_1 is as close to one as possible, which naturally depends on the choice of density $h(x)$.

If \tilde{f}_X coincides with f_X except some constant c_f , $f_X \cong \tilde{f}_X$, and similarly \tilde{h} coincides with h except some constant c_h , $h_X \cong \tilde{h}_X$, and $\tilde{f}_X \leq \tilde{h}$, then

$$c_f \tilde{f}_X(x) = f_X(x) \leq c_f \tilde{h}(x) = \frac{c_f}{c_h} c_h \tilde{h}(x) = c_1 h(x),$$

where $1/c_1 = c_h/c_f$ is the acceptance ratio. Therefore the test can be performed by checking the inequality

$$U \leq g(Y) = \frac{f_X(Y)}{c_1 h(Y)} = \frac{\tilde{f}_X(Y)}{\tilde{h}(Y)}, \quad (5.21)$$

where again Y and U , are distributed by $h(y)$ and uniform respectively. This method has been used extensively by Wood [34], [21], and we employ this technique in many ways. Ulrich [35] proposes a general method for simulating a rotationally symmetric variable.

5.4.2 Simulation of GFB families

We shall describe the simulation of the canonical variates. The necessary rotation can be done separately in each case and is included in the appropriate scripts of the MATLAB package ‘3D-Directional Statistics, Simulation and Visualization’ (3D-Directional-SSV).

We shall use the well known basic algorithms for simulation of spherical uniform, vMF and DW random variates (Sup.C.Alg.3-6).

The inequality (5.21) shows that for the simulation we do not need the normalizing constants, although they are needed for calculating the acceptance ratios. If these constants are not available analytically, we use numerical integration for calculating the acceptance ratios.

Model GFB₄

GFB₄ generalizes both the vMF and DW models. We use the envelopes proposed by Wood [21] in the application of acceptance-rejection sampling as described above.

The density in basic form

$$g_4(x; \kappa, \gamma) \cong e^{\kappa x + \gamma x^2},$$

implies that the longitude $\Phi = 2\pi U$, where U is uniform distributed independently from the colatitude Θ , and hence, we shall concentrate on simulation of $X = \cos \Theta$. Also, we assume that parameters κ and γ are not identically zero, since otherwise the model is reduced to either vMF or DW.

We consider three cases according to the relation between the parameters κ and γ .

1. If $\gamma < 0$, $0 \leq \kappa \leq -2\gamma$, complete the quadratic form in the exponent and rewrite g_4 as

$$g_4(x; \kappa, \gamma) \cong e^{\gamma(x + \kappa/2\gamma)^2}, \quad x \in [-1, 1],$$

and use the Gaussian envelope. The acceptance ratio is large if we assume that the mean

$-\kappa/2\gamma$ belongs to $[-1, 1]$, which is the case under this assumption.

The algorithm of this case is given in the first part of Algorithm 7, Supplement 7.2.3.

2. If $\gamma < 0$, $\kappa \geq -2\gamma$, then we use vMF $g_{vMF}(x, \kappa + 2\gamma)$ on \mathbb{S}_2 , as an envelope
3. If $\gamma > 0$, an envelope composed of a mixture of $g_{vMF}(x, \kappa + \gamma)$ and $g_{vMF}(x, \kappa - \gamma)$ vMF distributions is used.

The difference between our method, Algorithm 7, Supplement 7.2.3, and the one described in [21], is how we partition the case where $\gamma < 0$.¹

Some particular cases with interesting features arise in the simulation of models in the **GFB**₅ family. In this vein, we consider **GFB**_{4,β}, which demonstrates the difficulty of combining two algorithms, as well as the problem of simulation of Φ according to the conditional density $g_{\Phi|X}$.

Model **GFB**_{4,β},

If $\beta \neq 0$, $\gamma = 0$, $\kappa = 0$, the density in basic form is given by

$$g_{\beta}(x, \varphi; \kappa, \beta) \cong e^{\beta(1-x^2)\cos 2\varphi},$$

and has marginal

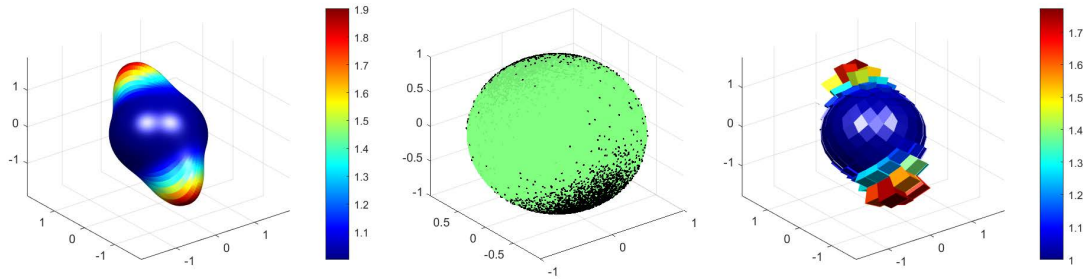
$$g_{\beta,X}(x, \beta) \cong I_0(\beta(1-x^2)),$$

and conditional $g_{\Phi|X}$ (5.8) densities.

The simulation of the marginal X is based on the inequality

$$\frac{1}{c_I} I_0(\beta(1-x^2)) \leq \frac{c}{2c_I} (p_1 f_W(x; -\beta) + p_2 f_W(x; \beta)).$$

¹Note, [21] Procedure **GFB**₄⁻, p. 890 has a misprint. The correct expressions are $\mu_1 = \frac{2\gamma-\kappa}{\sqrt{-2\gamma}}$, $\mu_2 = \frac{-2\gamma-\kappa}{\sqrt{-2\gamma}}$

Figure 5.5: Model $\mathbf{GFB}_{4,\beta}$

With the mixture of two DW densities, one is bipolar with $\beta > 0$, and the other one is girdle with $-\beta < 0$. where

$$c_I = \int_{-1}^1 I_0(\beta(1-x^2)) dx.$$

Now, we use the acceptance-rejection method for simulation of $X = \cos \Theta$, with an acceptance ratio of $2c_I/c$ that is decreasing as β is increasing. The corresponding algorithm is Algorithm 8, Supplement 7.2.3,

see Figure 5.5.

The simulation of longitude Φ according to the conditional density $g_{\Phi|X}$ is given by simulating 2Φ as a vMF distributed random variate on \mathbb{S}_1 (see Remark 5). It yields a value in $[0, \pi]$, and hence, Φ itself will be an element in $[0, \pi/2]$. The density function g_β has the same value at Φ , $-\Phi$, $\Phi + \pi$, and $-\Phi + \pi$, therefore we shall extend the simulated value randomly.

The Algorithm 9, Supplement 7.2.3, will be used in more general cases as well, since the conditional densities do not change in different sub-families.

Model $\mathbf{GFB}_{5,K}$

Here we consider the Kent model $\mathbf{GFB}_{5,K}$, and note that the equal-area projection simplifies the problem of simulating two independent random variates.²

²See subsection 5.2.2 for details.

Equal-area projection, case $2\beta \leq \kappa$ This part of the simulation of the Kent model $\mathbf{GFB}_{5,K}$

$$f_5(\tilde{x}; \kappa, \beta) \cong e^{\kappa\tilde{x}_3 + \beta(\tilde{x}_1^2 - \tilde{x}_2^2)},$$

has been considered by Kent [29] under the assumption,

$$0 \leq 2\beta \leq \kappa. \quad (5.22)$$

This is the case when the exponent of the density is a non-increasing function of ϑ for each φ , and when the model is unimodal (see Figure 5.6). The Kent model, $\mathbf{GFB}_{5,K}$, under transformation (5.12) has the form

$$f_5(y_1, y_2; \kappa, \beta) \cong e^{-2(\kappa-2\beta)y_1^2 - 4\beta y_1^4} e^{-2(\kappa+2\beta)y_2^2 + 4\beta y_2^4},$$

which is a product of two densities. Introduce

$$\alpha_1 = \kappa - 2\beta, \quad \alpha_2 = \kappa + 2\beta,$$

in this parametrization the density has the form

$$f_5(y_1, y_2; \kappa, \beta) \cong e^{-2\alpha_1 y_1^2 - 4\beta y_1^4} e^{-2\alpha_2 y_2^2 + 4\beta y_2^4}. \quad (5.23)$$

If assumption (5.22) holds then

$$\alpha_1 \geq 0, \quad \alpha_2 \geq 0,$$

and we use Kent's algorithm [29], given in Algorithm 10.

We simplify Kent's algorithm, using a Gaussian envelope instead of the exponential envelope. From Kent's method, we stop at the first inequality and apply the acceptance-rejection

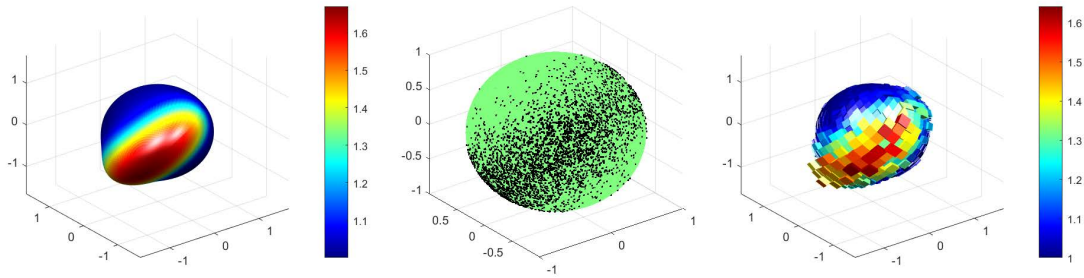


Figure 5.6: Model $\mathbf{FB}_{5,\kappa}$, unimodal: $2\beta \leq \kappa$, from left to right: density, sample, histogram

method right there. The basic inequality is given by,

$$-\frac{1}{2}\delta^2 w^2 \leq \frac{1}{2}\tau^2 - \delta\tau |w|, \quad \delta, \tau > 0.$$

Setting $\delta = \sqrt{8\beta}$, and $\tau = 1$, we have

$$\exp(-2\alpha_1 y_1^2 - 4\beta y_1^4) \leq \exp\left(\frac{1}{2} - \frac{1}{2}(4\alpha_1 + \sqrt{8\beta})y_1^2\right),$$

which gives the Gaussian envelop with variance $\sigma_1^2 = 1/(4\alpha_1 + \sqrt{8\beta})$. Similarly

$$\exp(-2\alpha_2 y_2^2 + 4\beta y_2^4) \leq \exp\left(-\frac{1}{2}(4\alpha_2 - 8\beta)y_2^2\right),$$

yields a Gaussian envelop as well, with variance $\sigma_2^2 = 1/4\kappa$.³ We combine these two and use acceptance-rejection method in the first part of Algorithm 11, Supplement 7.2.3.

Equal-area projection, case $2\beta > \kappa$ We are also interested in the case when the density increasing and decreasing in ϑ around $\varphi = 0$.

$$0 \leq \kappa \leq 2\beta,$$

³We have corrected a misprint in [29] in our algorithm

In this case,

$$\alpha_1 = \kappa - 2\beta \leq 0,$$

$$\alpha_2 = \kappa + 2\beta \geq 0.$$

Let us start with the first component of (5.23), $\exp(-2\alpha_1 y_1^2 - 4\beta y_1^4)$. The polynomial

$$p(y_1) = -\alpha_1 y_1^2 - 2\beta y_1^4,$$

in the exponent is symmetric to zero. We restrict the variable y_1 such that $|y_1| \leq 1$, hence we have two maximums at $y_0 = \pm \sqrt{(1 - \kappa/2\beta)/2}$, where

$$p(y_0) = \frac{\beta}{2} \left(1 - \frac{\kappa}{2\beta}\right)^2.$$

Therefore we separate the interval $[-1, 1]$ and use a Gaussian envelope

$$\exp(-2\alpha_1 y_1^2 - 4\beta y_1^4) \leq \exp\left(-2\left(\frac{\beta}{2}\left(1 - \frac{\kappa}{2\beta}\right)(y_1 - y_0^+)^2 + p(y_0)\right)\right), \quad (5.24)$$

on $[0, 1]$. This envelope will exactly match the target density at the maximum, otherwise it is greater.

Similarly

$$\exp(-2\alpha_2 y_2^2 + 4\beta y_2^4) \leq \exp(-(2\alpha_2 - 4\beta)y_2^2) = \exp(-2\kappa y_2^2), \quad (5.25)$$

provide a Gaussian envelop as well, with variance $\sigma_2^2 = 1/4\kappa$. The second part of Algorithm 11, Supplement 7.2.3, is based on these two inequalities (5.24) and (5.25).

If $2\beta > \kappa$, then the model is bimodal, but not bipolar, and the cosine of the angle, η , between

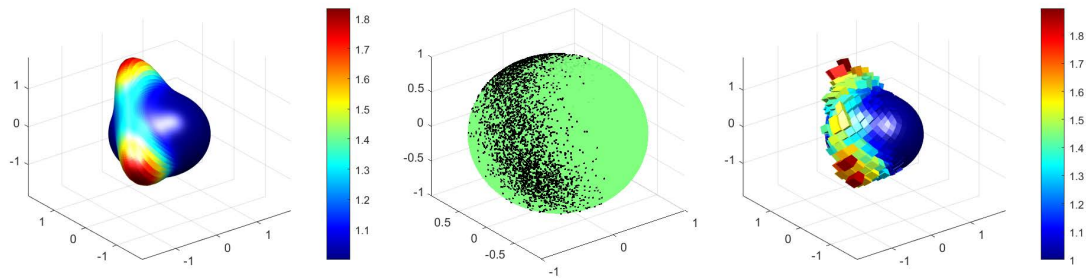


Figure 5.7: If $2\beta > \kappa$, then the model $\mathbf{GFB}_{5,K}$, is bimodal

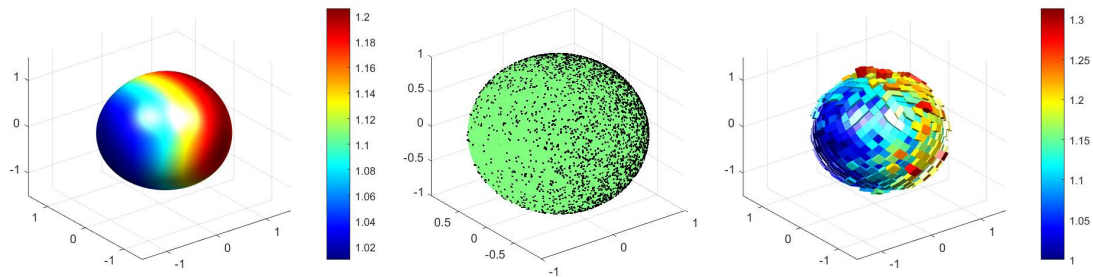


Figure 5.8: Model \mathbf{FB}_6 , $\kappa = 1.5$; $\beta = 0.61$; $\gamma = 0.31$; $\mu = [1, -1, 1]$, $\psi = 0$;

the modal directions is $\cos \eta = \kappa/2\beta$. We put $\kappa/2\beta = 1/2$, and hence $\eta = \pi/3$ (see Figure 5.7).

Model \mathbf{GFB}_6

The Kent model $\mathbf{GFB}_{5,K}$ and the Bingham Model $\mathbf{GFB}_{5,B}$ are included here since, as we shall see, there is only a small difference whether $\kappa = 0$, or $\gamma = 0$. If $\kappa = 0$, we have paid particular attention to the case $\gamma = \beta \neq 0$, see Remark 7.

We repeat the method applied for the \mathbf{GFB}_β , $\beta \neq 0$, with κ and γ , then we use a mixture of two \mathbf{GFB}_4 densities. The corresponding algorithms can be found in Algorithms 9 and 12, in Supplement 7.2.3.

5.4.3 Simulation of some spherical distributions via their spherical harmonics

If a density $f(\tilde{\mathbf{x}})$ is continuous then it has series expansion in terms of spherical harmonics Y_ℓ^m ,

$$f(\tilde{\mathbf{x}}) = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} a_\ell^m Y_\ell^m(\tilde{\mathbf{x}}). \quad (5.26)$$

The coefficients $\{a_\ell^m\}$ can be considered as a characteristic function, they are complex valued, $a_\ell^{m*} = (-1)^m a_\ell^{-m}$, and are given by

$$a_\ell^m = \int_{\mathbb{S}_2} f(\tilde{\mathbf{x}}) Y_\ell^{m*}(\tilde{\mathbf{x}}) \Omega(d\tilde{\mathbf{x}}). \quad (5.27)$$

Several symmetries of distributions are characterized in terms of coefficients a_ℓ^m , (5.27), see [4].

Notice that the spherical harmonic⁴ $Y_0^0 = 1/\sqrt{4\pi}$, and hence $a_0^0 = 1/\sqrt{4\pi}$ is the normalizing constant for $f(\tilde{\mathbf{x}})$ and $\Omega(d\tilde{\mathbf{x}}) = \sin \vartheta d\vartheta d\varphi$ is the Lebesgue element of surface area on \mathbb{S}_2 . The notation $*$ is defined as the transpose and conjugate of a matrix and just the conjugate for a scalar. For a detailed account of Spherical Distributions and Harmonic Analysis see [4]

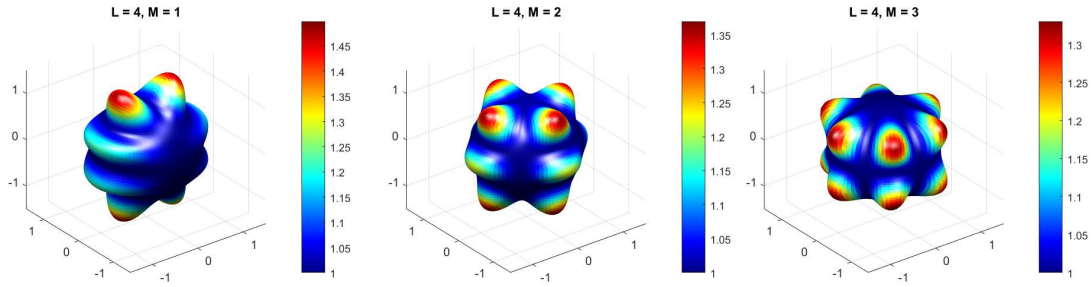
Let $\tilde{\mathbf{x}}(\vartheta, \varphi)$ be a point on the unit sphere \mathbb{S}_2 , then the spherical harmonics are defined by

$$Y_\ell^m(\vartheta, \varphi) = (-1)^m \sqrt{\frac{2\ell+1}{4\pi} \frac{(\ell-m)!}{(\ell+m)!}} P_\ell^m(\cos \vartheta) e^{im\varphi}, \quad \varphi \in [0, 2\pi], \quad \vartheta \in [0, \pi],$$

where P_ℓ^m denotes *associated normalized Legendre function of the first kind*.

Real-valued spherical harmonics. Spherical harmonics are, in general, complex valued, due to the dependence of longitude φ given by $e^{im\varphi}$. Now, $e^{im\varphi}$ is a complete orthogonal system on the circle which is equivalent to the sine-cosine system for real-valued functions. Similarly,

⁴Some of the first spherical harmonics are listed in [4].

Figure 5.9: Square of real spherical harmonics, $L = 4$, $m = 1, 2, 3$

real-valued spherical harmonic functions are defined as

$$Y_{\ell,m} = \begin{cases} \frac{1}{\sqrt{2}} (Y_{\ell}^m + (-1)^m Y_{\ell}^{-m}) & m > 0 \\ Y_{\ell}^m & m = 0 \\ \frac{1}{i\sqrt{2}} (Y_{\ell}^{-m} - (-1)^m Y_{\ell}^m) & m < 0 \end{cases} \quad (5.28)$$

The harmonics with order $m > 0$ are said to be of cosine type, and those with $m < 0$ of sine type.

Figure 5.9 contains three densities among $Y_{4,m}^2$, when $m = 1, 2, 3$ ⁵.

Exponential family class of distributions [30],

$$g(\tilde{\mathbf{x}}) = \exp \sum_{\ell=0}^M \sum_{m=-\ell}^{\ell} b_{\ell}^m Y_{\ell}^m(\tilde{\mathbf{x}}),$$

where $b_{\ell}^{m*} = (-1)^m b_{\ell}^{-m}$. The normalizing constant corresponds to b_0^0 and depends on the rest of the b_{ℓ}^m parameters, such that the integral is necessarily 1. The likelihood of observations often

⁵One may plot densities according to module square of complex and real spherical harmonics for any given L and all $m \in [-L, L]$ by the command `dx = Density_SphHarm_All(L, m, Real,resolution)`; if $m \notin [-L, L]$ then all $Y_{L,m}^2$ will be plotted, see our MATLAB package '3D-Directional Statistics, Simulation and Visualization' (3D-Directional-SSV) for details.

has this form. In the case of rotational symmetry around axis \tilde{x}_0

$$f_e(\tilde{x}) = \exp\left(\sum_{\ell=0}^M b_\ell P_\ell(\cos \vartheta)\right)$$

where ϑ is the angle between \tilde{x} and \tilde{x}_0 , and P_ℓ denotes the *Standardized* Legendre polynomial ($P_0(x) = 1$).

Another useful class of density functions is given by,

$$f(\tilde{x}) = \left| \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} d_\ell^m Y_\ell^m(\tilde{x}) \right|^2,$$

where $\sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} |d_\ell^m|^2 = 1$. This class of distributions are of interest in, amongst other topics, the modeling of atoms.

In quantum mechanics, $Y_{\ell,m}(\tilde{x})^2$ is considered as a probability density function, and plays an important role in the modeling of the hydrogen atom. The module square $|Y_\ell^m(\tilde{x})|^2$ is also a density function that serves as an example of a rotational symmetric density function on the sphere, as it only depends on $\cos \vartheta$.

Simulation of modulus-square of complex harmonics

Consider the density $|Y_\ell^m(\tilde{x})|^2$ on \mathbb{S}_2 with respect to Lebesgue measure $\sin \vartheta d\vartheta d\varphi$. We have

$$|Y_\ell^m(\tilde{x})|^2 = \frac{2\ell + 1}{4\pi} \frac{(\ell - m)!}{(\ell + m)!} P_\ell^m(\cos \vartheta)^2,$$

where P_ℓ^m is associated normalized Legendre function of the first kind.

$|Y_\ell^m(\tilde{x})|^2$ is rotational symmetric and $2\pi |Y_\ell^m|^2$ will be a density on $[-1, 1]$. More precisely, the function

$$f_\ell^m(x) = c_\ell^m P_\ell^m(x)^2,$$

where

$$c_{\ell}^m = \frac{2\ell + 1}{2} \frac{(\ell - m)!}{(\ell + m)!},$$

is a density on $[-1, 1]$. We have $\left| Y_{\ell}^m(\tilde{\underline{x}}) \right|^2 = f_{\ell}^m(\cos \vartheta)^2 / 2\pi$. For convenience (MATLAB) we use the fully normalized associated Legendre function p_{ℓ}^m , such that $f_{\ell}^m(x) = p_{\ell}^m(x)^2$.

For general ℓ and m and $p_{\ell}^m(x)^2$, one can use Forsyth's method [36], with intervals defined by the zeros of $p_{\ell}^m(x)^2$ and Beta envelopes, say. In particular, let $\ell = 3$, $m = 2$. then $p_3^2(x)^2$ has a root at 0 and is symmetric around zero. For generating X , we apply the Neumann Theorem for $2f_3^2(x)$, Algorithm 14, since the Beta distribution with shape parameters $\alpha = 3.08$ and $\beta = 2.5249$, proved to be an efficient envelope for the density $2f_3^2(x)$ (see Figure ??).

The beta density is given by the function $b(x, \alpha, \beta) = x^{\alpha-1} (1-x)^{\beta-1} / B(\alpha, \beta)$, and the above shape parameters ($\alpha = 3.08$ and $\beta = 2.5249$) were found with the following steps:

1. let $\alpha, \beta > 1$,
2. use the relation $(\alpha - 1) / (\alpha + \beta - 2) = x_M$, between α and β , where x_M is the maximum point (modus) of $2f_3^2(x)$,
3. find the set of α and β under the (simultaneous) constraints:
 - i) $\max_x (2f_3^2(x) / b(x, \alpha, \beta)) \geq 1$
 - ii) $\max_{\alpha} \left(\min_x (2f_3^2(x) / cb(x, \alpha, \beta)) \right) \leq 1$
4. choose a pair α and β , such that $\min_{\alpha} \left(\max_x (2f_3^2(x) / b(x, \alpha, \beta)) \right)$ is achieved

We then have,

$$\begin{aligned} 2f_3^2(x) &= \frac{2f_3^2(x)}{b(x, \alpha, \beta)} b(x, \alpha, \beta) \\ &= cg_3^2(x) h_3^2(x), \end{aligned}$$

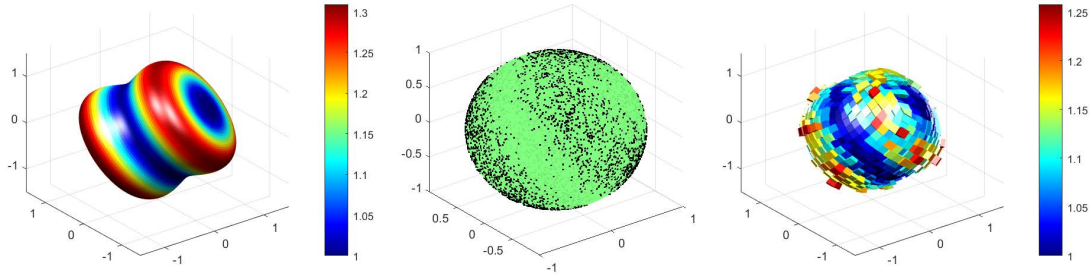


Figure 5.10: Simulation of density $\left|Y_{\ell}^m(\tilde{x})\right|^2$

with $c = \max(2f_3^2(x)/b(x, \alpha, \beta))$, $h_3^2(x) = b(x, \alpha, \beta)$, and

$0 \leq g_3^2(x) = 2f_3^2(x)/(cb(x, \alpha, \beta)) \leq 1$. We have found that $c = 1.074$, shows a high acceptance ratio.

Simulation of real harmonics $Y_{3,2}^2$

Consider the real spherical harmonics $Y_{3,2}$ (5.28). The density function $Y_{3,2}^2$ is given by

$$\begin{aligned} Y_{3,2}^2 &= \frac{1}{2} (Y_3^2 + Y_3^{-2})^2 = \frac{1}{2} \frac{7}{4\pi} \frac{1}{5!} P_3^2(\cos \vartheta)^2 (e^{i2\varphi} + e^{-i2\varphi})^2 \\ &= 2\sqrt{\frac{7}{4\pi} \frac{1}{5!}} P_3^2(\cos \vartheta)^2 \cos^2 2\varphi. \end{aligned}$$

This is a multimodal, non-rotationally symmetric distribution. The density function $Y_{3,2}^2$ is the multiplication of the density of Θ and the density of Φ respectively, which implies that the angles Θ and Φ are changing independently. The simulation of Θ according to the density $\cong P_3^2(\cos \vartheta)^2 \sin \vartheta$, has been solved during the simulation of module square of complex harmonics above.

The simulation of Φ by the density $\cong \cos^2 \varphi$ is based on the following:

Let B be Beta distributed with $\alpha = 3/2$ and $\beta = 1/2$, and define $X = \arccos(\sqrt{B})$. Then,

from the equality

$$F_X(x) = P\left(\arccos\left(\sqrt{B}\right) \leq x\right) = P\left(B > \cos^2 x\right),$$

it follows

$$\begin{aligned} f_X(x) &= 2f_B(\cos^2 x) \cos x \sin x \\ &= \frac{2}{b(\alpha, \beta)} (\cos^2 x)^{1/2} (\sin^2 x)^{-1/2} \cos x \sin x = \frac{4 \cos^2 x}{\pi} \end{aligned} \quad (5.29)$$

Similarly, if B is a Beta distributed variate with $\alpha = 1/2$ and $\beta = 3/2$,

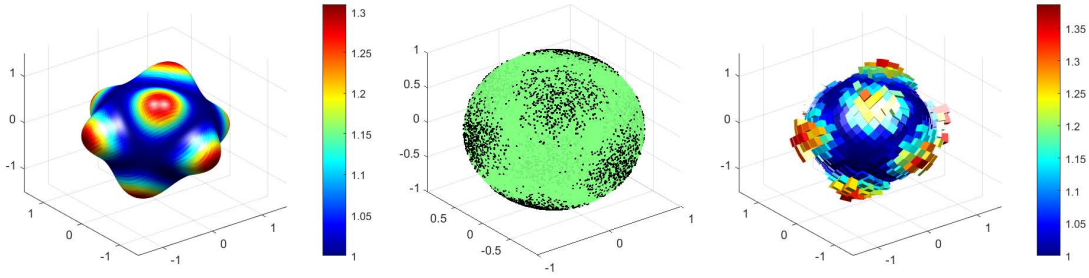
$$\begin{aligned} f_X(x) &= 2f_B(\cos^2 x) \cos x \sin x \\ &= \frac{2}{b(\alpha, \beta)} (\cos^2 x)^{-1/2} (\sin^2 x)^{3/2} \cos x \sin x = \frac{4 \sin^2 x}{\pi}. \end{aligned}$$

The latter of the two formulations for $f_X(x)$ is used for the simulation of $Y_{3,-2}^2$.

We conclude that the density of $\Phi = \arccos\left(\sqrt{B}\right)$ is $\cong \cos^2 \varphi$, on $[-\pi/2, \pi/2]$. Once we have a random variate on $[-\pi/2, \pi/2]$ with density function depending on $\cos \varphi$ we can easily transform its values periodically to the interval $[0, 2m\pi]$, say. Algorithm 14, Supplement 7.2.3, works for the density

$$f(\tilde{x}) \cong P_3^2 (\cos \vartheta)^2 \cos^2 2\varphi,$$

in this case we choose $m = 2$, since the longitude φ should have values on $[0, 2\pi]$, and we have the random variate 2Φ on $[-\pi/2, \pi/2]$. One can generalize this method for density $\cong \cos^2 k\varphi$, where k is an integer.

Figure 5.11: Simulation of density $Y_{3,2}^2$

5.4.4 Simulation of U -distribution

Finally, we examine what we call the “ U -distribution” which is not to be confused with the uniform distribution on the sphere. This is a very simple and interesting example of a spherical distribution that is antipodal, but not isotropic or rotationally symmetric.

Consider U_1, U_2, U_3 uniform on $[0, 1]$, independent variates. Then we define

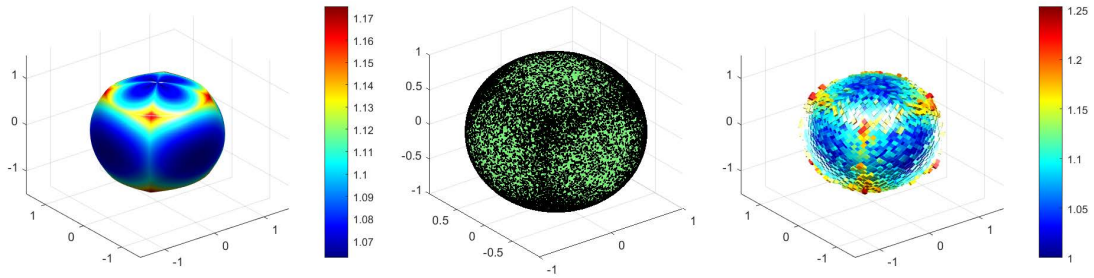
$$\tilde{\underline{Z}} = [Z_1, Z_2, Z_3] = (U_1, U_2, U_3)^\top / \sqrt{U_1^2 + U_2^2 + U_3^2}, \quad \tilde{\underline{Z}} \in \mathbb{S}_2.$$

We may write $\tilde{\underline{Z}} = (\sin \Theta \cos \Phi, \sin \Theta \sin \Phi, \cos \Theta)$, where

$$\begin{aligned} \tan(\Phi) &= \frac{Z_2}{Z_1} = \frac{U_2}{U_1} \\ P(\Phi < \varphi) &= P\left(\frac{U_2}{U_1} < \tan(\varphi)\right) \\ &= \begin{cases} \frac{\tan(\varphi)}{2} & \text{if } 0 \leq \varphi < \pi/4 \\ 1 - \frac{1}{2 \tan(\varphi)} & \text{if } \pi/4 \leq \varphi < \pi/2 \end{cases}, \end{aligned}$$

and see formally that it is not the uniform density function since

$$f_\Phi(\varphi) = \begin{cases} \frac{1}{2 \cos^2(\varphi)} & \text{if } 0 \leq \varphi < \pi/4 \\ \frac{1}{2 \sin^2(\varphi)} & \text{if } \pi/4 \leq \varphi < \pi/2. \end{cases}$$

Figure 5.12: U -distribution.

Now consider the conditional density $f_{\Theta|\Phi}(\vartheta|\varphi)$ for a fixed $\Phi = \varphi$:

$$\tan(\Theta) \cos(\Phi) = \frac{Z_1}{Z_3} = \frac{U_1}{U_3};$$

$$P(\Theta < \vartheta | \Phi = \varphi) = P\left(\frac{U_1}{U_3} < \cos(\varphi) \tan(\vartheta)\right)$$

$$= \begin{cases} \frac{\cos(\varphi) \tan(\vartheta)}{2} & \text{if } \varphi \in [0, \pi/4], \vartheta \in [0, \pi/2], \cos(\varphi) \tan(\vartheta) \leq 1 \\ 1 - \frac{1}{2 \cos(\varphi) \tan(\vartheta)} & \text{if } \varphi \in [0, \pi/4], \vartheta \in [0, \pi/2], \cos(\varphi) \tan(\vartheta) > 1 \end{cases},$$

$$f_{\Theta|\Phi}(\vartheta|\varphi) f_{\Phi}(\varphi) = \begin{cases} \frac{\cos(\varphi)}{16 \cos^2(\vartheta) \cos^2(\varphi)} & \text{if } \varphi \in [0, \pi/4], \vartheta \in [0, \pi/2], \cos(\varphi) \tan(\vartheta) \leq 1 \\ \frac{1}{16 \sin^2(\vartheta) \cos^3(\varphi)} & \text{if } \varphi \in [0, \pi/4], \vartheta \in [0, \pi/2], \cos(\varphi) \tan(\vartheta) > 1 \end{cases}$$

Chapter 6

Conclusions and Future Directions

6.1 Conclusions

In this investigation, we have seen the application of the L_2 distance and SKL divergence between von Mises and von Mises-Fisher models, across a variety of scenarios, as a means to quantify model similarity. Both measures performed well in the lower dimensional case, where they were able to recover the correct groupings of the simulated data in chapter 2.

When looking at the NRR curves as the target data, where no ground truth was known, their efficacy was measured by comparative cluster agreement, where we took various ultrametric measures to comparing the hierarchies induced by the respective measures. Again, there seemed to be substantive agreement, which would lead us to believe there is some underlying structure that both the L_2 and SKL measure are detecting. These same ultrametrics corroborated the findings, even when changing our viewpoint to look at the data from a functional perspective. We also included implementations of the Mahalanobis distance as well as the functional Mahalanobis semi-distance, and found that all of these were in agreement, at various levels.

Increasing dimension, to work with textual data, the L_2 distance was able to detect, at least at some rudimentary level, changes in successive models that would seem to correspond with

changes in the public conversation, via twitter, and the 2016 U.S. Presidential election. A positive contribution is that we can write both measures in a closed form, but the drawback to both is the curse of dimensionality. In the case of the L_2 distance, we need to calculate the values for the modified Bessel functions for extremely high values of κ and even higher orders. This can lead to numerical issues, which may mandate some form of dimensionality reduction.

6.2 Future Directions

6.2.1 Commander-in-Tweet, and the 2020 U.S. Presidential Election

With the 2020 U.S. Presidential election rapidly approaching, there is ample opportunity to turn our work towards the upcoming contest. In addition to the Presidential election itself, there is also the Democratic primary contest to observe and quantify, with respect to improving prediction by adding additional feature vectors to current methods. We aim to extend the current results by incorporating additional distributions within the Generalized Fisher-Bingham family of distributions, to model the spherical data. In addition, we will be looking to add other social media platforms to the collection of data under inspection. We will also be looking to develop computationally efficient methods for dealing with ultra-high dimensional L_2 distance approximation, as well as additional methods for dimension reduction, to better manage the data without losing features or meaning.

6.2.2 Early Glaucoma Detection

The NRR curves are part of a much larger and richer data set that was introduced to us by Professor Saumyadipta Pyne, School of Public Health, University of Pittsburgh, and Mr. Hasnat Ali, L.V. Prasad Eye Institute, Hyderabad, India. We made use of a small part of the much larger original data, with the limited aim of illustrating our methodologies. A more comprehensive

analysis and biological implications is forthcoming, where we intend to investigate the potential for improved early glaucoma detection, in addition to various other eye related phenomenon.

Chapter 7

Appendix and Supplementary Material

7.1 Chapter 4: Supplementary Material

Mixture von Mises-Fisher (Hard Assignment):

- The parameters, $\mathcal{P} = \{\boldsymbol{\mu}, \boldsymbol{\kappa}, \boldsymbol{\pi}\}$ are treated as unknown, but fixed constants, and the only latent variables are the topic assignments $\mathbf{Z} = \{z_1, z_2, \dots, z_D\}$

1. Draw cluster assignment,

$$Z_i \sim \text{Multinomial}(\boldsymbol{\pi}), \text{ for } i = 1, 2, \dots, D$$

where $z_i \in \{1, 2, \dots, K\}$.

2. Draw document vector,

$$\mathbf{v}_i \sim \text{vMF}(\boldsymbol{\mu}_{z_i}, \boldsymbol{\kappa}), \text{ for } i = 1, 2, \dots, D$$

We employ the EM algorithm, where we alternate between the

- E-Step: Find $\mathbb{E}[\mathbf{Z}]$ for fixed \mathcal{P}
- M-Step: Maximizing the likelihood over \mathcal{P} for fixed $\mathbb{E}[\mathbf{Z}]$

EM update equations, for the k^{th} topic are:

- E-Step

$$\mathbb{E}[\mathbf{Z}_{i,k}] = \frac{\boldsymbol{\pi}_k \text{vMF}(\mathbf{v}_i | \boldsymbol{\mu}_k, \boldsymbol{\kappa})}{\sum_{j=1}^K \boldsymbol{\pi}_j \text{vMF}(\mathbf{v}_i | \boldsymbol{\mu}_j, \boldsymbol{\kappa})}$$

- M-Step

$$R_k = \sum_{i=1}^D \mathbb{E}[\mathbf{Z}_{i,k}] \mathbf{v}_i \quad \boldsymbol{\pi}_k = \sum_{i=1}^D \frac{\mathbb{E}[\mathbf{Z}_{i,k}]}{D}, \quad \boldsymbol{\mu}_k = \frac{R_k}{\|R_k\|}, \quad \bar{r} = \sum_{k=1}^K \frac{\|R_k\|}{D}, \quad \boldsymbol{\kappa} = \frac{\bar{r}d - \bar{r}^3}{1 - \bar{r}^2}$$

7.2 Chapter 5: Supplementary Material

The supplementary section is organized as follows:

1. Spherical harmonics description:

Form and basic properties of spherical harmonics, as specified in [4].

2. MATLAB scripts for Figures and Algorithms:

1. Figures: Listing of all the MATLAB scripts for Figures contained in the paper using the package "3D-Directional-SSV"
2. Algorithms: Listing of all the Algorithms in this paper using the package "3D-Directional-SSV").

3. 3D-Directional Statistics, Simulation and Visualization ("3D-Directional-SSV")

MATLAB Package to simulate and visualize spherical distributions in 3-dimensions including can be found at,

<https://github.com/TerdikGyorgy/3D-Simulation-Visualization>

7.2.1 Supplement: Spherical harmonics

Orthonormal spherical harmonics with complex values $Y_\ell^m(\vartheta, \varphi)$, $\ell = 0, 1, 2, \dots$, $m = -\ell, -\ell + 1, \dots, -1, 0, 1, \dots, \ell - 1, \ell$ of **degree** ℓ and **order** m (rank ℓ and projection m)

$$Y_\ell^m(\vartheta, \varphi) = (-1)^m \sqrt{\frac{2\ell + 1}{4\pi} \frac{(\ell - m)!}{(\ell + m)!}} P_\ell^m(\cos \vartheta) e^{im\varphi}, \quad \varphi \in [0, 2\pi], \quad \vartheta \in [0, \pi], \quad (7.1)$$

where P_ℓ^m denotes *associated normalized Legendre function of the first kind*. The spherical harmonics are eigenfunctions of the square of the orbital angular momentum operator.

$$Y_\ell^0(\vartheta, \varphi) = \sqrt{\frac{2\ell + 1}{4\pi}} P_\ell(\cos \vartheta), \quad (7.2)$$

$$Y_0^0(\vartheta, \varphi) = \sqrt{\frac{1}{4\pi}},$$

more over

$$Y_\ell^m(\underline{\tilde{N}}) = \delta_{m,0} \sqrt{\frac{2\ell + 1}{4\pi}}. \quad (7.3)$$

Y_ℓ^m is fully normalized

$$\int_0^{2\pi} \int_0^\pi |Y_\ell^m(\vartheta, \varphi)|^2 \sin \vartheta d\vartheta d\varphi = 1.$$

Some detailed account of spherical harmonics Y_ℓ^m can be found in [37] and [38].

some authors do not apply $1/\sqrt{4\pi}$ in the definition of Y_ℓ^m , also for a sphere with radius R *spherical harmonics are normalized additionally* $Y_\ell^m(\vartheta, \varphi)/R$. It also follows

$$\begin{aligned} Y_\ell^{m*}(\vartheta, \varphi) &= Y_\ell^m(\vartheta, -\varphi) \\ &= (-1)^m Y_\ell^{-m}(\vartheta, \varphi), \\ Y_\ell^{-m}(\vartheta, \varphi) &= (-1)^m e^{-i2m\varphi} Y_\ell^m(\vartheta, \varphi). \end{aligned}$$

Inversion $\underline{\tilde{x}} \rightarrow -\underline{\tilde{x}}, (\vartheta, \varphi) \rightarrow (\pi - \vartheta, \pi + \varphi)$

$$Y_\ell^m(-\underline{\tilde{x}}) = (-1)^\ell Y_\ell^m(\underline{\tilde{x}}). \quad (7.4)$$

Addition formula (see [39], 8.814,[40], 11.4(8)),

$$\sum_{m=-\ell}^{\ell} Y_\ell^{m*}(\underline{\tilde{x}}_1) Y_\ell^m(\underline{\tilde{x}}_2) = \frac{2\ell+1}{4\pi} P_\ell(\cos \vartheta), \quad (7.5)$$

where $\cos \vartheta = \underline{\tilde{x}}_1 \cdot \underline{\tilde{x}}_2$.

$$\sum_{m=-\ell}^{\ell} Y_\ell^{m*}(\underline{\tilde{x}}) Y_\ell^m(\underline{\tilde{x}}) = \frac{2\ell+1}{4\pi}, \quad (7.6)$$

7.2.2 Supplement: MATLAB Scripts for Figures

In this section we list all the MATLAB Scripts of Figures contained in the paper using package 3D-Directional-SSV.

MATLAB Script of Figure 5.2

```
pV1 = [0.1,0.1,0.43,0.82]; pV2 = [0.5,0.1,0.43,0.82];
Psi=0; Mu= [0,0,1]; resolution=100; gamm=6.364; bet1=4.5; bet2=1.5;
figure('Position',[1 1 756 343])
subplot('Position',pV1); gx5 = Density_FB5(gamm,bet1,Mu,Psi,resolution);
subplot('Position',pV2); gx5 = Density_FB5(gamm,bet2,Mu,Psi,resolution);
```

MATLAB Script of Figure 5.3

```
Mu= [0,0,1]; Psi=0; resolution=100;
pV1 = [0.05,0.1,0.28,0.82];pV2 = [0.38,0.1,0.28,0.82];pV3 = [0.69,0.1,0.28,0.82];
figure('Position',[300 300 1300 343])
subplot('Position',pV1); gx6 = Density_FB6(0,3.2,-1.1,Mu,Psi,resolution);
subplot('Position',pV2); gx6 = Density_FB6(0,3.2,3.2,Mu,Psi,resolution);
subplot('Position',pV3); gx6 = Density_FB6(0,3.2,4.1,Mu,Psi,resolution);
```

MATLAB Script of Figure 5.4

```

n=2^12; nSide=2^3; nPix= nSide2nPix(nSide);
Y = Random_Uni_Inv(n);
hY = Hist2DSphere(Y, nPix);
pV1 = [0.1,0.1,0.3,0.82]; pV2 = [0.5,0.1,0.43,0.82];
figure( 'Position', [1 1 756 343])
subplot( 'Position', pV1); Plot_DataRandomS2(Y);
subplot( 'Position', pV2); Plot_Hist2DSphere(hY);

```

MATLAB Script of Figure 5.5

```

n=2^12; nSide=2^3; nPix= nSide2nPix(nSide); resolution=100;
MuB=[1,-1,1]; betB =4.5 ; PsiB= pi/2;
Y = Random_FB4_Beta_GyT( betB ,MuB, PsiB ,n);
hY=Hist2DSphere(Y, nPix);
pV1 = [0.05,0.1,0.3,0.82]; pV2 = [0.4,0.1,0.22,0.82]; pV3 = [0.67,0.1,0.3,0.82];
figure( 'Position', [300 300 1300 343])
subplot( 'Position', pV1); dx = Density_FB4_Beta(betB ,MuB, PsiB );
subplot( 'Position', pV2); Plot_DataRandomS2(Y);
subplot( 'Position', pV3); Plot_Hist2DSphere(hY);

```

MATLAB Script of Figure 5.6

```

n=2^12; nSide=2^3; nPix= nSide2nPix(nSide); resolution=100;
kappaK = 5 ; betK =2 ; MuK=[-0.9 -1 .2]; PsiK= 0;
Y = Random_FB5_GyT(kappaK ,betK ,MuK, PsiK ,n);
hY=Hist2DSphere(Y, nPix);
pV1 = [0.05,0.1,0.3, 0.82]; pV2 = [0.4,0.1,0.22,0.82]; pV3 = [0.67,0.1,0.3,0.82];
figure( 'Position', [300 300 1300 343])
subplot( 'Position', pV1); gx5 = Density_FB5(kappaK ,betK ,MuK, PsiK , resolution);
subplot( 'Position', pV2); Plot_DataRandomS2(Y);
subplot( 'Position', pV3); Plot_Hist2DSphere(hY);

```

MATLAB Script of Figure 5.7

```

n=2^12; nSide=2^3; nPix= nSide2nPix(nSide); resolution=100;
kappaK = 5 ; betK =5 ; MuK=[-.5 0 .5]; PsiK= pi/4;
Y = Random_FB5_GyT(kappaK ,betK ,MuK,PsiK ,n);
hY=Hist2DSphere(Y, nPix);
pV1 = [0.05,0.1,0.3, 0.82]; pV2 = [0.4,0.1,0.22,0.82]; pV3 = [0.67,0.1,0.3,0.82];
figure( 'Position',[300 300 1300 343])
subplot('Position',pV1); gx5 = Density_FB5(kappaK ,betK ,MuK,PsiK ,resolution);
subplot('Position',pV2); Plot_DataRandomS2(Y);
subplot('Position',pV3); Plot_Hist2DSphere(hY);

```

MATLAB Script of Figure 5.8

```

n=2^12; nSide=2^3; nPix= nSide2nPix(nSide); resolution=100;
Psi6= 0; Mu6= [1,-1,1]; kappa6 = 1.5; bet6 = .61 ; gamm6 = -0.31;
pV1 = [0.05,0.1,0.3,0.82]; pV2 = [0.4,0.1,0.22,0.82]; pV3 = [0.67,0.1,0.3,0.82];
Y = Random_FB6( kappa6 ,bet6 ,0,Mu6,Psi6 ,n);
hY=Hist2DSphere(Y, nPix);
figure( 'Position',[300 300 1300 343])
subplot('Position',pV1);
gx6 = Density_FB6(kappa6 ,bet6 ,gamm6,Mu6,Psi6 ,resolution);
subplot('Position',pV2); Plot_DataRandomS2(Y);
subplot('Position',pV3); Plot_Hist2DSphere(hY);

```

MATLAB Script of Figure 5.9

```

pV1 = [0.05,0.1,0.28,0.82]; pV2 = [0.38,0.1,0.28,0.82]; pV3 = [0.69,0.1,0.28,0.82];
figure( 'Position',[300 300 1300 343]); Real=true;
subplot('Position',pV1);
px = Density_SphHarm(4, 1, Real);
subplot('Position',pV2);
px = Density_SphHarm(4, 2, Real);
subplot('Position',pV3);
px = Density_SphHarm(4, 3, Real);

```


MATLAB Script of Figure 5.10

```

n=2^12; nSide=2^3; nPix= nSide2nPix(nSide);
L=3; m=2; Mu=[1 -1 1]; Psi=pi/2; Real=false;
Y = Random_Y3_2Compl_square( Mu,n);
hY=Hist2DSphere(Y, nPix);
pV1 = [0.05,0.1,0.3,0.82]; pV2 = [0.4,0.1,0.22,0.82]; pV3 = [0.67,0.1,0.3,0.82];
figure( 'Position',[300 300 1300 343])
subplot('Position',pV1); px = Density_SphHarm(L, m, Real);
subplot('Position',pV2); Plot_DataRandomS2(Y);
subplot('Position',pV3); Plot_Hist2DSphere(hY);

```

MATLAB Script of Figure 5.11

```

n=2^12; nSide=2^3; nPix= nSide2nPix(nSide);
L=3; m=2; Mu=[1 -1 1]; Psi=pi/2; Real=true;
Y = Random_Y3_2Real_square( Mu, Psi, n);
hY=Hist2DSphere(Y, nPix);
pV1 = [0.05,0.1,0.3,0.82]; pV2 = [0.4,0.1,0.22,0.82]; pV3 = [0.67,0.1,0.3,0.82];
figure( 'Position',[300 300 1300 343])
subplot('Position',pV1); px = Density_SphHarm(L, m, Real);
subplot('Position',pV2); Plot_DataRandomS2(Y);
subplot('Position',pV3); Plot_Hist2DSphere(hY);

```

MATLAB Script of Figure 5.12

```

n=2^13; nSide=2^3; nPix= nSide2nPix(nSide); Mu=[0 0 1]; Psi= pi/2;
pV1 = [0.05,0.1,0.3,0.82]; pV2 = [0.4,0.1,0.22,0.82]; pV3 = [0.67,0.1,0.3,0.82];
Y = Random_U_Distr(Mu, n);
hY=Hist2DSphere(Y, nPix);
figure( 'Position',[300 300 1300 343])
subplot('Position',pV1); gx = Density_U(Mu, Psi);
subplot('Position',pV2); Plot_DataRandomS2(Y);
subplot('Position',pV3); Plot_Hist2DSphere(hY);

```

7.2.3 Supplement: Algorithms

In this section we list all the algorithms of our paper implemented in a MATLAB package '3D-Directional Statistics, Simulation and Visualization' (3D-Directional-SSV). We introduce the indicator function $\mathbf{1}_A = 1$ if A is true otherwise it is 0.

Algorithm 1 Rotation

Require: $\underline{\tilde{\mu}} \in \mathbb{S}_2$ and $\psi \in [0, 2\pi]$ are given

Ensure: Rotated frame of reference

- 1: Calculate $\underline{\tilde{N}} \times \underline{\tilde{\mu}}$ the #1 axis of rotation
 - 2: Rotate $\underline{\tilde{N}}$ to $\underline{\tilde{\mu}}$
 - 3: Use $\underline{\tilde{\mu}} = \underline{\tilde{N}}$ for the #2 axis of rotation and rotate the sphere by the angle ψ .
 { If only $\underline{\tilde{\mu}}$ is given, this algorithm simplifies to steps 1. and 2 or set $\psi = 0$. }
-

Algorithm 2 Rejection

Require: $f_X(x)$ (evaluatable), $h(y)$ (samplable), and proportionality constant c are given

Ensure: Variate X is distributed according to density $f_X(x)$

- 1: Generate two random variates Y and U , from $h(y)$ and $Unif(0, 1)$ respectively
 - 2: **if** $U \leq g(Y) = f_X(Y)/(c \cdot h(Y))$ **then**
 - 3: **return**
 $X = Y$ as a variate generated from f_X
 - 4: **else**
 - 5: reject Y and **go to** 1
 - 6: **end if**
-

Algorithm 3 Random_vMF_Inv

Require: $\kappa \in \mathbb{R}$ is given**Ensure:** Variate $\tilde{\underline{Y}}$ is distributed according to the von Mises-Fisher distribution, $vMF(\kappa)$ 1: Generate independent U_1, U_2 from $Unif(0, 1)$ 2: Calculate $X = \cos \Theta$, by

$$X = \frac{\log(2U_1 \sinh \kappa + e^{-\kappa})}{\kappa}$$

3: Calculate longitude $\Phi = 2\pi U_2$ 4: **return**

$$\tilde{\underline{Y}} = \left(\sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X \right)^\top.$$

Algorithm 4 Random_vMF_Wood**Require:** κ and $d \geq 2$ are given**Ensure:** Variate \tilde{X} is distributed according to the von Mises-Fisher distribution, $vMF(\kappa)$ on \mathbb{S}_{d-1}

1: Set

$$b = \frac{-2\kappa + \sqrt{4\kappa^2 + (d-1)^2}}{d-1},$$

$$x_0 = \frac{1-b}{1+b},$$

$$c = \kappa x_0 + (d-1) \log(1-x_0^2)$$

2: Generate variate B from $Beta(\alpha, \beta)$ with shape parameters given by

$$\alpha = (d-1)/2, \quad \beta = (d-1)/2,$$

3: Calculate

$$X = \frac{1 - (1+b)B}{1 - (1-b)B},$$

and generate U from $Unif(0, 1)$ 4: **if** $\kappa X + (d-1) \log(1-x_0 X) - c < \log U$ **then**5: **go to** step 2.6: **else**7: Generate a $d-1$ -dimensional spherical uniform vector W ,8: **return**

$$\tilde{X} = \left(\sqrt{1-X^2} W^\top, X \right)^\top.$$

9: **end if**

Algorithm 5 Random_Watson_Fish**Require:** $\gamma \in \mathbb{R}$ is given**Ensure:** Variate \tilde{Y} is distributed according to the Dimroth-Watson distribution, $DW(\gamma)$

```

1: if  $\gamma > 0$ , (Bipolar) then
2:   Set  $c = 1 / (e^\gamma - 1)$ 
3:   Generate independent  $U_1, U_2$  and  $U$  from  $Unif(0, 1)$ 
4:   Set  $X = (\log(U_1/c + 1)) / \gamma$ 
5:   if  $U_2 \leq \exp(\gamma X^2 - \gamma X)$  then
6:      $X = (\mathbf{1}_{U < 1/2} - \mathbf{1}_{U \geq 1/2}) X$ ,
       {Since X is positive and the density is symmetric to the equator}
7:      $\Phi = 2\pi U_2$ 
8:     return
        $\tilde{Y} = (\sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X)$ 
9:   else
10:    go to step 3.
11:  end if
12: else  $\{\gamma < 0, \text{(Girdle)}\}$ 
13:   Set  $c_1 = \sqrt{|\gamma|}$ ,  $c_2 = \arctan c_1$ .
14:   Generate independent  $U_1, U_2$  and  $U$  from  $Unif(0, 1)$ 
15:   Set  $X = (1/c_1) \tan(U_1 c_2)$ 
16:   if  $U_2 \leq (1 - \gamma X^2) \exp(\gamma X^2)$  then
17:      $X = (\mathbf{1}_{U < 1/2} - \mathbf{1}_{U \geq 1/2}) X$ ;
       {Since S is positive and the density is symmetric to the equator}
18:      $\Phi = 2\pi U_0$ ,
19:     return
        $\tilde{Y} = (\sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X)^\top$ .
20:   else
21:    go to step 13.
22:  end if
23: end if

```

Algorithm 6 Random_Watson_LW**Require:** $\gamma \in \mathbb{R}$ is given**Ensure:** Variate \tilde{Y} is distributed according to the Dimroth-Watson distribution, $DW(\gamma)$

1: **if** $\gamma > 0$, (Bipolar) **then**
2: Set

$$\rho = \frac{4\gamma}{2\gamma + 3 + \sqrt{(2\gamma + 3)^2 - 16\gamma}}, \quad r = \left(\frac{3\rho}{2\gamma}\right)^3 e^{-3+2\gamma/\rho},$$

3: Generate independent U_1 and U_2 from $Unif(0, 1)$
4: Set

$$S = \frac{U_1^2}{1 - \rho(1 - U_0^2)}, \quad V = \frac{rU_2^2}{(1 - \rho S)^3}, \quad W = \gamma S,$$

5: **if** $V \leq e^{2W}$ **then**
6: Put $\Theta = \arccos \sqrt{S}$
7: Generate U_3 from $Unif(0, 1)$
8: Calculate $X = \cos \mathbf{1}_{U_3 < 1/2} (\pi - \Theta)$, $\Phi = 4\pi \mathbf{1}_{U_3 < 1/2} U_3 + 2\pi(2U_3 - 1)$,
9: **else**
10: **go to** step 3.
11: **end if**
12: **return** $\tilde{Y} = \left(\sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X\right)^\top$.
13: **else** $\{\gamma < 0$, (Girdle) $\}$
14: Set $b = e^{2\gamma} - 1$,
 { Begin, generate variates X_k from truncated normal distribution }
15: Generate independent U_1, U_2 from $Unif(0, 1)$
16: Set $V = \ln(1 + U_1 b) / \gamma$, $\xi = 2\pi U_2$ and $c = \cos \xi$
17: Set $S_1 = Vc^2$, and $S_2 = V - S_1$
18: **if** either $S_1 > 1$ or $S_2 > 1$ **then**
19: **go to** step 15.
20: **else**
21: Set $X_1 = \sqrt{V}c$, $X_2 = \sin \xi$
22: **end if**
 { End of Generation X_k }
23: Generate independent U_4, U_5 from $Unif(0, 1)$
24: Calculate $\Phi_1 = 2\pi U_4$, $\Phi_2 = 2\pi U_5$
25: **return**

$$\tilde{Y}_1 = \left(\sqrt{1 - X_1^2} \cos(\Phi), \sqrt{1 - X_1^2} \sin(\Phi), X_1\right)^\top,$$

$$\tilde{Y}_2 = \left(\sqrt{1 - X_2^2} \cos(\Phi), \sqrt{1 - X_2^2} \sin(\Phi), X_2\right)^\top,$$

two independent variates from DW distribution.

26: **end if**

Algorithm 7 Random_FB4**Require:** κ, γ are given**Ensure:** Variate \tilde{Y} is distributed according to the Fisher-Bingham₄ (**GFB**₄) distribution

```

1: if  $\gamma < 0$  then
2:   if  $0 \leq \kappa \leq -2\gamma$  then
3:     Generate  $Z$  from normal  $\mathcal{N}(-\kappa/2\gamma, -1/2\gamma)$ 
4:     if  $Z \in [-1, 1]$  then
5:       Generate  $U_1$  from  $Unif(0, 1)$ 
6:       Set  $\Phi = 2\pi U_1$ ,
7:       return  $\tilde{Y} = \left( \sqrt{1 - Z^2} \cos(\Phi), \sqrt{1 - Z^2} \sin(\Phi), Z \right)^\top$ 
8:     else
9:       Reject  $Z$  and go to 3
10:    end if
11:  else
12:    Generate  $U$  from  $Unif(0, 1)$ 
13:    Generate  $X$  from  $vMF(\kappa + 2\gamma)$ 
14:    if  $U \leq e^\gamma \exp(\gamma(X^2 - 2X))$  then
15:      Generate  $U_2$  from  $Unif(0, 1)$ 
16:      Set  $\Phi = 2\pi U_2$ ,
17:      return  $\tilde{Y} = \left( \sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X \right)^\top$ .
18:    else
19:      reject  $X$  and go to 12
20:    end if
21:  end if
22: else if  $\gamma > 0$  then
23:   Set  $c_\kappa = \frac{\kappa}{2\pi(e^\gamma - e^{-\gamma\kappa})}$ ,  $p_1 = \frac{c_{\kappa+\gamma}}{c_{\kappa+\gamma} + c_{\kappa-\gamma}}$ ,
24:   Generate independent  $U$  and  $U_1$  from  $Unif(0, 1)$ 
25:   Generate independent  $X_1$  and  $X_2$  from  $vMF$  with parameters  $\kappa - \gamma$ , and  $\kappa + \gamma$  respectively
26:   Calculate the mixture  $X$ ,  $X = \mathbf{1}_{U_1 \leq p_1} X_1 + \mathbf{1}_{U_1 > p_1} X_2$ ,
27:   if  $U \leq (1 + e^{-2\gamma}) \frac{e^\gamma X^2}{e^\gamma X + e^{-\gamma X}}$  then
28:     Generate  $U_3$  with uniform distribution
29:     set  $\Phi = 2\pi U_3$ ,
30:     return  $\tilde{Y} = \left( \sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X \right)^\top$ .
31:   else
32:     reject  $X$  and go to 24
33:   end if
34: else
35:    $\gamma = 0$ , generate  $vMF(\kappa)$ 
36: end if

```

Algorithm 8 Random_FB4_Beta**Require:** $\beta \neq 0$ is given**Ensure:** Variate \tilde{Y} is distributed according to the $\mathbf{GFB}_{4,\beta}$ distribution

1: Calculate

$$c_- = \int_{-1}^1 e^{-\beta x^2} dx; \quad c_+ = \int_{-1}^1 e^{\beta x^2} dx$$

$$c = e^\beta c_- + e^{-\beta} c_+; \quad p_1 = e^\beta c_- / c$$

2: Generate independent U_1 and U_2 from $Unif(0, 1)$ 3: Generate independent V_1 and V_2 from DW with parameters β and $-\beta$ respectively4: Set $V = (\mathbf{1}_{U_1 < p_1}) V_1 + (1 - \mathbf{1}_{U_1 < p_1}) V_2$ 5: **if** $U_2 \leq I_0(\beta(1 - V^2)) / \cosh(\beta(1 - V^2))$ **then**6: Set $X = \cos \Theta$, by $X = V$,7: **else**8: **go to** step 29: **end if**10: Use Algorithm 9 with $\beta(1 - X^2)$ to get Φ 11: **return** $\tilde{Y} = (\sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X)^\top$ **Algorithm 9** vMF_Circ_Phi**Require:** $\beta \neq 0$ and x **Ensure:** Variate Φ distributed vMF on \mathbb{S}_1 extended to $[0, 2\pi]$ 1: Generate random variate Φ_1 from $vMF(\beta(1 - x^2))$, on \mathbb{S}_1 2: Generate random variate U uniformly distributed on integers $(1, 2, 3, 4)$ 3: **return** $\Phi = (\mathbf{1}_{U=1} + \mathbf{1}_{U=3}) \Phi + (\mathbf{1}_{U=2} + \mathbf{1}_{U=3}) \pi - (\mathbf{1}_{U=2} + \mathbf{1}_{U=4}) \Phi + 2\pi \mathbf{1}_{U=4}$

Algorithm 10 Random_FB5_Kent

Require: $\beta \geq 0$, $\kappa \geq 0$ are given such that we have the case where, $2\beta \leq \kappa$ (equal-area projection), for $\kappa < 0$, we use $|\kappa|$ and transform back in a final step

Ensure: Variate \tilde{Y} is distributed according to the Kent distribution (Model **GFB**_{5, κ})

1: Set

$$a = 4\kappa - 8\beta; \quad b = 4\kappa + 8\beta; \quad \gamma = 8\beta$$

$$\lambda_1 = \sqrt{a + 2\sqrt{\gamma}}; \quad \lambda_2 = \sqrt{b}; \quad c_2 = b/8\kappa$$

2: Generate independent U_1 and U_2 from $Unif(0, 1)$

3: Generate independent R_1 and R_2 from exponential distribution with parameters λ_1 and λ_2 respectively

4: **if** $U_1 \leq \exp(-(aR_1^2 + \lambda_1 R_1^4)/2 + \lambda_1 R_1 - 1)$ **then**

5: Accept R_1

6: **else**

7: **go to** step 3

8: **end if**

9: **if** $U_2 \leq \exp(-(bR_2^2 - \gamma R_2^4)/2 + \lambda_2 R_2 - c_2)$ **then**

10: Accept R_2

11: **else**

12: **go to** step 3

13: **end if**

14: **if** $R_1^2 + R_2^2 < 1$ **then**

15: Accept (R_1, R_2)

16: **else**

17: **go to** step 3

18: **end if**

19: Using trigonometric identities

$$\cos \vartheta = 1 - 2(R_1^2 + R_2^2), \quad \sin \varphi = \frac{R_2}{\sqrt{R_1^2 + R_2^2}}, \quad \cos \varphi = \frac{R_1}{\sqrt{R_1^2 + R_2^2}},$$

calculate

$$X = 1 - 2(R_1^2 + R_2^2); \quad S_\varphi = \frac{R_2}{\sqrt{R_1^2 + R_2^2}}; \quad C_\varphi = \frac{R_1}{\sqrt{R_1^2 + R_2^2}},$$

20: **return** $\tilde{Y} = (\sqrt{1 - X^2}C_\varphi, \sqrt{1 - X^2}S_\varphi, X)^\top$

Algorithm 11 Random_FB5_GyT

Require: $\beta \geq 0, \kappa \geq 0$ are given, for $\kappa < 0$, we use $|\kappa|$ and transform back in a final step

Ensure: Variate \tilde{Y} is distributed according to the Kent distribution (Model **GFB_{5,K}**)

```

1: if  $2\beta \leq \kappa$  then
2:   Set:  $\alpha_1 = \kappa - 2\beta$ ;  $\alpha_2 = \kappa + 2\beta$ ;  $\sigma_1^2 = 1/(4\alpha_1 + \sqrt{8\beta})$ ;  $\sigma_2^2 = 1/4\kappa$ 
3:   Generate  $U_1$  from  $Unif(0, 1)$  and  $Z_1$  from normal  $\mathcal{N}(0, \sigma_1^2)$ 
4:   if  $U_1 \leq \exp\left(-\left(2\alpha_1 Z_1^2 + 4\beta Z_1^4\right) - \frac{1}{2} + \left(2\alpha_1 + \sqrt{8\beta}\right) Z_1^2\right)$  then
5:     Accept  $Z_1$ 
6:   else
7:     go to step 2
8:   end if
9:   Generate  $U_2$  from  $Unif(0, 1)$  and  $Z_2$  from normal  $\mathcal{N}(0, \sigma_2^2)$ 
10:  if  $U_2 \leq \exp\left(-\left(2\alpha_2 Z_2^2 + 4\beta Z_2^4\right) + (2\alpha_2 - 4\beta) Z_2^2\right)$  then
11:    Accept  $Z_2$ 
12:  else
13:    go to step 9
14:  end if
15:  if  $Z_1^2 + Z_2^2 < 1$  then
16:    go to step 37
17:  else
18:    go to step 2
19:  end if
20: else
21:  Set  $\alpha_1 = \kappa - 2\beta$ ;  $\alpha_2 = \kappa + 2\beta$ ;  $\eta = 1 - \kappa/(2\beta)$ ;  $y_0 = \sqrt{\eta/2}$ ;  $p_{y_0} = (\beta/2)\eta^2$ ;
    $\sigma_1^2 = 1/2\beta\eta$ ;  $\sigma_2^2 = 1/4\kappa$ .
22:  Generate  $U_1$  from  $Unif(0, 1)$  and  $Z_1$  from normal,  $\mathcal{N}(y_0, \sigma_1^2)$ 
23:  if  $U_1 \leq \exp\left(-\left(2\alpha_1 Z_1^2 + 4\beta Z_1^4\right) + \beta\eta(Z_1 - y_0)^2 - 2p_{y_0}\right)$  then
24:    Accept  $Z_1$ 
25:  else
26:    go to step 22
27:  end if
28:  Generate  $U$  from  $Unif(0, 1)$ 
29:  Set  $Z_1 = Z_1 (\mathbf{1}_{U < 1/2} - \mathbf{1}_{U \geq 1/2})$ 
30:  Generate  $U_2$  from  $Unif(0, 1)$  and  $Z_2$  from normal,  $\mathcal{N}(p_{y_0}, \sigma_2^2)$ 
31:  if  $U_2 \leq \exp\left(-\left(\alpha_2 R_2^2 + 4\beta R_2^4\right) + (2\alpha_2 - 4\beta) Z_2^2\right)$  then
32:    Accept  $Z_2$ 
33:  else
34:    go to step 27
35:  end if
36:  if  $Z_1^2 + Z_2^2 < 1$  then
37:    Using trigonometric identities, calculate
      $X = 1 - 2\left(Z_1^2 + Z_2^2\right)$ ;  $S_\varphi = Z_2/\sqrt{Z_1^2 + Z_2^2}$ ;  $C_\varphi = Z_1/\sqrt{Z_1^2 + Z_2^2}$ ,
38:    return  $\tilde{Y} = \left(\sqrt{1 - X^2}C_\varphi, \sqrt{1 - X^2}S_\varphi, X\right)^\top$ .
39:  else
40:    go to step 22
41:  end if
42: end if

```

Algorithm 12 Random_FB6

Require: $\kappa \geq 0$, $\beta \neq 0$, and $\gamma \in \mathbb{R}$ are given

Ensure: Variate \tilde{Y} is distributed according to the Fisher-Bingham₆ distribution (Model **GFB**₆)

1: Calculate

$$c_- = \int_{-1}^1 e^{\kappa x + (\gamma - \beta)x^2} dx; \quad c_+ = \int_{-1}^1 e^{\kappa x + (\gamma + \beta)x^2} dx$$

$$c_6 = e^\beta c_- + e^{-\beta} c_+, \quad p_1 = \frac{e^\beta c_-}{c_6}$$

2: Generate independent U_1 and U_2 from $Unif(0, 1)$

3: Generate independent V_1 and V_2 from **GFB**₄ distribution on the sphere with parameters $(\kappa, \gamma - \beta)$ and $(\kappa, \gamma + \beta)$ respectively

4: Set $V = (\mathbf{1}_{U_1 < p_1}) V_1 + (1 - \mathbf{1}_{U_1 < p_1}) V_2$

5: **if** $U_2 \leq I_0(\beta(1 - V^2)) / \cosh(\beta(1 - V^2))$ **then**

6: $X = V$

7: **else**

8: **go to** step 2

9: **end if**

10: Use Algorithm 9 with $\beta(1 - V^2)$ to get Φ

11: **return** $\tilde{Y} = (\sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X)^\top$.

Algorithm 13 Random_Y3_2Compl_square**Require:** $n \in \mathbb{N}$ is given (sample size)**Ensure:** Variate \tilde{Y} distributed according to the distribution characterized by $\left|Y_3^2(\tilde{x})\right|^2$ { Use a Beta envelope for generating X with the density $2f_3^2(u)$ where $u \in (0, 1)$ }1: Generate U_1 from $Unif(0, 1)$ 2: Generate variate B from $Beta(\alpha, \beta)$ distribution with shape parameters given by

$$\alpha = 3.08, \quad \beta = 2.5249,$$

3: **if**

$$U_1 > \frac{2g_3^2(B)}{b(B, \alpha, \beta)}.$$

then4: **go to** step 15: **else**6: Accept $X = B$ with acceptance ratio $c^{-1} = 1/1.074 = 0.9311$.7: **end if**{ End of generation of X }8: Generate U from $Unif(0, 1)$ 9: Set $X = X(\mathbf{1}_{U < 1/2} - \mathbf{1}_{U \geq 1/2})$ 10: Generate independent Z_1 and Z_2 from standard normal, $\mathcal{N}(0, 1)$ 11: Calculate variate W , from the uniform distribution on circle, given by

$$W = \left(Z_1 / \sqrt{Z_1^2 + Z_2^2}, Z_2 / \sqrt{Z_1^2 + Z_2^2} \right)^\top,$$

12: **return** $\tilde{Y} = \left(\sqrt{1 - X^2}W, X \right)^\top$.

Algorithm 14 Random_Y3_2Real_squared**Require:** $n \in \mathbb{N}$ is given (sample size)**Ensure:** Variate \tilde{Y} distributed according to the distribution characterized by

$$f(\tilde{x}) \cong P_3^2(\cos \vartheta)^2 \cos^2 2\varphi$$

- 1: Generate X by as in Algorithm 13,
 { Simulate Φ by density $\cos^2 K\varphi$. on $[0, 2\pi]$ }
- 2: Generate variate B from $Beta(\alpha, \beta)$ with shape parameters given by

$$\alpha = 3/2, \quad \beta = 1/2$$

with density $b(x, \alpha, \beta)$

- 3: Calculate $\Phi = \arccos \sqrt{B} \in [0, \pi/2]$,
- 4: Generate variate U from $Unif(0, 1)$
- 5: Generate variate K from $Discrete Uniform$ on the set $\{1, 2, 3, 4\}$
- 6: Set

$$\Phi = \Phi (\mathbf{1}_{K=1} + \mathbf{1}_{K=2} - \mathbf{1}_{K=3} - \mathbf{1}_{K=4}) + \pi (\mathbf{1}_{K=2} - 2\mathbf{1}_{K=3} + \mathbf{1}_{K=4}) + (\mathbf{1}_{U < 1/2}) 2\pi,$$

$$\Phi = \Phi/2,$$

7: **return**

$$\tilde{Y} = \left(\sqrt{1 - X^2} \cos(\Phi), \sqrt{1 - X^2} \sin(\Phi), X \right)^\top.$$

Algorithm: hard-moVMF

Input: Set of \mathcal{X} of data points on \mathbb{S}^{d-1}

Output: A disjoint k -partitioning of \mathcal{X}

Initialize all $\alpha_h, \boldsymbol{\mu}_h, \kappa_h, h = 1, \dots, k$

repeat

{The Hardened E-Step of EM }

for $i = 1$ to n **do**

for $h = 1$ to k **do**

$$f_h(\mathbf{x}_i | \boldsymbol{\theta}_h) \leftarrow c_d(\kappa_h) \exp\{\kappa_h \boldsymbol{\mu}_h^T \mathbf{x}_i\}$$

end

for $h = 1$ to k **do**

$$q(h | \mathbf{x}_i, \Theta) = \begin{cases} 1, & \text{if } h = \operatorname{argmax}_{h'} p(h' | \mathbf{x}_i, \Theta) \\ 0, & \text{otherwise.} \end{cases}$$

end

end

{The M-Step of EM }

for $h = 1$ to k **do**

$$\alpha_h \leftarrow \frac{1}{n} \sum_{i=1}^n p(h | \mathbf{x}_i, \Theta)$$

$$\boldsymbol{\mu}_h \leftarrow \sum_{i=1}^n \mathbf{x}_i p(h | \mathbf{x}_i, \Theta)$$

$$\bar{r} \leftarrow \|\boldsymbol{\mu}_h\| / (n\alpha_h)$$

$$\boldsymbol{\mu}_h \leftarrow \boldsymbol{\mu}_h / \|\boldsymbol{\mu}_h\|$$

$$\kappa_h \leftarrow \frac{\bar{r}d - \bar{r}^3}{1 - \bar{r}^2}$$

end

until convergence

Bibliography

- [1] S. R. Jammalamadaka and A. SenGupta, *Topics In Circular Statistics*, vol. 5 of *Series on Multivariate Analysis*. World Scientific Publishing Co. Pte. Ltd., Singapore, London, New Jersey, Hong Kong, 2001.
- [2] C. Bouveyron, *funFEM: Clustering in the Discriminative Functional Subspace*, 2015. R package version 1.1.
- [3] G.Terdik and B.Wainwright, *3d-simulation-visualization*, 2018, version 1.0.0.0.
- [4] S. R. Jammalamadaka and G. H. Terdik, *Harmonic analysis and distribution-free inference for spherical distributions*, *Journal of Multivariate Analysis* **171** (2019) 436 – 451.
- [5] H. Teicher, *On the mixture of distributions*, *The Annals of Mathematical Statistics* **31** (1960), no. 1 55–73.
- [6] T. Ferguson, *Bayesian density estimation by mixtures of normal distributions*, *Recent Advances in Statistics (M. Rizvi, J. Rustagi, and D. Siegmund, eds.)* **24** (1983) 287–302.
- [7] S. S. Dhillon I.S., *Modeling data using directional distributions*, Tech. Rep. TR-03-06, Department of Computer Sciences, The University of Texas at Austin, Austin, TX 78712, 1, 2003.
- [8] J. A.Banerjee, I.Dhillon and S.Sra, *Clustering on the unit hypersphere using von Mises-Fisher distributions*, *Journal of Machine Learning Research* **6** (2005) 1345–1382.
- [9] K. Hornik and B. Grün, *On maximum likelihood estimation of the concentration parameter of von Mises–Fisher distributions*, *Computational Statistics* **29** (2014), no. 5 945–957.
- [10] M. Adhi and J. S. Duker, *Optical coherence tomography –current and future applications*, *Current Opinion in Ophthalmology* **24** (2013), no. 3 213–221.
- [11] A. Gordon, *Classification, 2nd Edition*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. CRC Press, 1999.
- [12] K. Hornik, *clue: Cluster ensembles*, 2019. R package version 0.3-57.

- [13] K. Hornik, *A CLUE for Cluster Ensembles*, *Journal of Statistical Software* **14** (September, 2005).
- [14] C. Bouveyron, E. Côme, J. Jacques, *et. al.*, *The discriminative functional mixture model for a comparative analysis of bike sharing systems*, *The Annals of Applied Statistics* **9** (2015), no. 4 1726–1760.
- [15] K. Fukunaga, *Introduction to statistical pattern recognition. Second edition*. Academic Press, Inc., Boston, MA, 1990.
- [16] A. Petersen and H. G. Müller, *Functional data analysis for density functions by transformation to a hilbert space*, *Annals of Statistics* **44** (2016), no. 1 183–218.
- [17] G. C. Biernacki and G. Govaert, *Assessing a mixture model for clustering with the integrated completed likelihood*, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22** (2000), no. 7 719–725.
- [18] P. Galeano, E. Joseph, and R. E. Lillo, *The Mahalanobis distance for functional data with applications to classification*, *Technometrics* **57** (2015), no. 2 281–291.
- [19] D. Blei and J. Lafferty, *Topic models*, in *Text Mining: Classification, Clustering, and Applications* (A. Srivastava and M. Sahami, eds.), ch. 4, pp. 71–93. Chapman & Hall/CRC Data Mining and Knowledge Discovery Series, Boca Raton, FL, 2009.
- [20] K. Hornik and B. Grün, *movMF: An R package for fitting mixtures of von Mises-Fisher distributions*, *Journal of Statistical Software* **58** (2014), no. 10 1–31.
- [21] A. Wood, *The simulation of spherical distributions in the Fisher-Bingham family*, *Communications in Statistics - Simulation and Computation* **16** (1987), no. 3 885–898.
- [22] J. Kent, *The Fisher-Bingham distribution on the sphere*, *Journal of the Royal Statistical Society Series B (Methodological)* (1982) 71–80.
- [23] J. Kent, A. Ganeiber, and K. Mardia, *A new unified approach for the simulation of a wide class of directional distributions*, *Journal of Computational and Graphical Statistics* **27** (2018), no. 2 291–301.
- [24] K. Mardia and P. Jupp, *Directional statistics*. Wiley series in probability and statistics. Wiley, Chichester, 2009.
- [25] R. Fisher, *Dispersions on a sphere*, *Proceedings of the Royal Society of London A*. **217(1130)** (1953) 295–305.
- [26] E. Dimroth, *Untersüchungen aim wechanismiis von blastais und syntexis*, *Phylliterr and Hodelsen des siidmßlichen Fichtelgebirges* **8** (1962) 248–274.
- [27] G. Watson, *Equatorial distributions on a sphere*, *Biometrika* **52(1/2)** (1965) 193–201.

- [28] C. Bingham, *An antipodally symmetric distribution on the sphere*, *The Annals of Statistics* **2** (1974), no. 6 1201–1225.
- [29] J. Kent and T. Hamelryck, *Using the Fisher-Bingham distribution in stochastic models for protein structure*, *Quantitative Biology, Shape Analysis, and Wavelets* **24** (2005) 57–60.
- [30] G. Watson, *Statistics on spheres. Number 6 in University of Arkansas Lecture Notes in the Mathematical Sciences*. John Wiley & Sons, Inc., New York, NY, USA, 1983.
- [31] K.M.Gorski, E.Hivon, A.J.Banday, B.D.Wandelt, F.K.Hansen, M.Reinecke, and M.Bartelmann, *HEALPix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere*, *The Astrophysical Journal* **622** (April, 2005) 759–771.
- [32] J. von Neumann, *Various techniques used in connection with random digits*, in *Monte Carlo Method* (A. Householder, G. Forsythe, and H. Germond, eds.), pp. 36–38. National Bureau of Standards Applied Mathematics Series, 12, Washington, D.C.: U.S. Government Printing Office, 1951.
- [33] R. Y. Rubinstein, *Simulation and the Monte Carlo Method*. John Wiley & Sons, Inc., New York, NY, USA, 1st ed., 1981.
- [34] A. Wood, *Some notes on the Fisher–Bingham family on the sphere*, *Communications in Statistics - Theory and Methods* **17** (1988), no. 11 3881–3897.
- [35] G. Ulrich, *Computer generation of distributions on the m-sphere*, *Applied Statistics* (1984) 158–163.
- [36] A. C. Atkinson and M. C. Pearce, *The computer generation of beta, gamma and normal random variables*, *Journal of the Royal Statistical Society. Series A (General)* **139** (1976), no. 4 431–461.
- [37] V. Khersonskii, A. Moskalev, and D. Varshalovich, *Quantum Theory Of Angular Momentum*. World Scientific Publishing Company, 1988.
- [38] E. Stein and G. Weiss, *Introduction to Fourier Analysis on Euclidean Spaces*. Mathematical Series. Princeton University Press, 1971.
- [39] I. Gradshteyn and I. M. Ryzhik, *Translation edited*, in *Table of Integrals, Series, and Products (Sixth Edition)* (T. from the Russian, with a preface by Alan Jeffrey, and D. Zwillinger, eds.). Academic Press, San Diego, CA, 2000.
- [40] A.Erdélyi and Bateman Manuscript Project, *Higher Transcendental Functions: Based, in Pt., on Notes Left by Harry Bateman*. Robert E. Krieger Publishing Co. Inc., Melbourne, Fl., 1981.