Chapter 4
Hippocampus Shape Analysis via Skeletal Models and Kernel Smoothing

Eduardo García-Portugués and Andrea Meilán-Vila

Abstract Skeletal representations (s-reps) have been successfully adopted to parsimoniously parametrize the shape of three-dimensional objects and have been particularly employed in analyzing hippocampus shape variation. Within this context, we provide a fully nonparametric dimension-reduction tool based on kernel smoothing for determining the main source of variability of hippocampus shapes parametrized by s-reps. The methodology introduces the so-called density ridges for data on the polysphere and involves addressing high-dimensional computational challenges. For the analyzed dataset, our model-free indexing of shape variability reveals that the spokes defining the sharpness of the elongated extremes of hippocampi concentrate the most variation among subjects.

Keywords Density ridges · Dimension reduction · Directional data · Nonparametric statistics · Skeletal representations

4.1 Introduction

Mental illnesses are prevalent and highly debilitating disorders that affect a substantial proportion of society. Various studies have shown that there is a direct relationship between the etiology of mental diseases and the deformation of more vulnerable parts of the brain, such as the hippocampus (see, e.g., [1]). Hence, the analysis of hippocampus shapes is a relevant target of medical research and a useful instrument for informing it. The present work contributes towards this analysis by introducing a new method to determine the main variation of three-dimensional shapes, like hippocampi, that are instantiated in the form of skeletal models.

Statistical shape analysis of three-dimensional objects [2] can be enhanced by using skeletal models, as these capture the interior of objects, and therefore, they

E. García-Portugués (✉) · A. Meilán-Vila
Universidad Carlos III de Madrid, Leganés, Madrid, Spain
e-mail: edgarcia@est-econ.uc3m.es; ameilan@est-econ.uc3m.es

Y. Larriba (ed.), *Statistical Methods at the Forefront of Biomedical Advances*,
https://doi.org/10.1007/978-3-031-32729-2_4

more stably and richly collect object shape than models that capture only its boundary. These models explicitly capture the width of the objects, the normal directions, and the boundary curvatures [3]. Skeletal models contain a skeleton, which is centrally located along the object, and the spokes (vectors originating from the skeleton terminating at the object boundary), such that the spokes do not cross within the object [4]. A general construction of these models is the *skeletal representation*, referred to as *s-rep* [5]. The rigorous statistical analysis of skeletal models requires the development of tailored novel methods, this constituting an instance of the so-called *object oriented data analysis* [6]. There is a substantial literature involving s-reps, see, for example, [7, 8], and [9], among others. See also [10] and the recent survey by [11] for a complete review of skeletal models.

The dataset considered in this work consists of $n = 177$ hippocampus shapes that are instantiated in the form of s-reps (see [3]). Figure 4.4 shows the s-reps of two characteristic hippocampi. The spokes are the segments (in varying colors) joining the inner skeletal points (in black) with the boundary points (also in varying colors, some of them numbered). Each hippocampus has $r = 168$ spokes with associated radii and directions. The directions of these spokes lie on the polysphere $(\mathbb{S}^2)^{168}$, where $(\mathbb{S}^d)^r := \mathbb{S}^d \times \overset{r}{\cdots} \times \mathbb{S}^d$, with $\mathbb{S}^d := \{\mathbf{x} \in \mathbb{R}^{d+1} : \|\mathbf{x}\| = 1\}$ and $r, d \geq 1$. The shapes of the hippocampi constituting the analyzed dataset are different, but their inner skeletal points share roughly matching configurations. Therefore, it is reasonable to consider the average inner skeletal configuration as a common reference, and then investigate the vectors that lead from it to the boundary. Fixing also the radii of these spokes to their averages across subjects allows a reduced representation, as an observation on $(\mathbb{S}^d)^r$ (size is ignored), of the hippocampus shape captured by an s-rep.

Traditionally, Principal Component Analysis (PCA) has been used to describe the main features of the data by estimating the principal directions of its maximum projected variance. In the framework of skeletal models, modes of variation on the sphere based on a non-geodesic approach can provide more appropriate dimensionality reduction [4, 12]. Following this strategy, [13] introduced Principal Arc Analysis (PAA), which uses small circles on the sphere $\mathbb{S}^2$ to parametrize the main source of variation. Principal Nested Spheres (PNS) is the extension of this method to the hypersphere $\mathbb{S}^d$ [14]. An alternative to the previous parametric approaches for summarizing the primary characteristics of the data consists of generating flexible principal curves informed by the underlying density of the data. Density ridges extend the concept of modes and rely on the gradient and Hessian of the density function [15]. Although density ridge estimation is a challenging task, which can be addressed with an appropriate smoothing-based estimator, it entails a much larger flexibility over fixed parametric modes of variation (e.g., small circles on $\mathbb{S}^d$).

We introduce in this work a novel fully nonparametric dimension-reduction technique for polyspherical data. The proposed methodology involves estimating density ridges for $(\mathbb{S}^d)^r$-valued data, which entail a specific kernel density estimator and the computation of its $(\mathbb{S}^d)^r$-adapted gradient and Hessian. The estimation of density ridges applies an Euler-like algorithm that presents several high-dimensional computational challenges, and thus we describe keys and guidelines for its implementation and practical use. We propose an effective data-driven indexing and parametrization of the set of $(\mathbb{S}^d)^r$-valued points that is outputted by the Euler algorithm to attain a ridge analog of a (first) principal component curve. Marching along this ridge principal curve is especially useful to visualize the main mode of variation on the original s-rep space. We also highlight this work gives a proof of concept of the applicability of density ridges in high-dimensional settings.

The rest of this chapter is organized as follows. Section 4.2 introduces a new dimension-reduction method to determine the main shape variation of three-dimensional objects parametrized through s-reps. The approach requires two tailored smoothing techniques (Sect. 4.2.1). On the one hand, a kernel density estimator for polyspherical data and its derivatives (Sects. 4.2.1.1–4.2.1.2), and, on the other hand, a kernel regression estimator for $(\mathbb{S}^d)^r$-valued response (Sect. 4.2.1.3). Density ridges are presented in Sect. 4.2.2 for the population Euclidean (Sect. 4.2.2.1) and sample polyspherical (Sect. 4.2.2.2) cases. The details of the advocated density ridge estimation procedure are elaborated in Sects. 4.2.2.3–4.2.2.5. Section 4.3 shows the results of applying our methodology. Specifically, an illustrative numerical example on $(\mathbb{S}^2)^2$ (Sect. 4.3.1) and the visualization of the main mode of variation of the aforementioned hippocampi data (Sect. 4.3.2). A critical discussion of the methodology and the identified open areas for improvement is provided in Sect. 4.4. Proofs are relegated to the appendix.

4.2 Methodology

4.2.1 Kernel Smoothing on the Polysphere

4.2.1.1 Density Estimation

Let f be a probability density function (pdf) on $(\mathbb{S}^d)^r \subset \mathbb{R}^{r(d+1)}$ with respect to the product measure $\sigma_{d,r} := \sigma_d \times \overset{r}{\cdots} \times \sigma_d$, where σ_d is the surface area measure on $\mathbb{S}^d$. Let $\mathbf{X}_1, \ldots, \mathbf{X}_n$ be an independent and identically distributed (iid) sample from f. Let $\mathbf{x} = (\mathbf{x}_1', \ldots, \mathbf{x}_r')' \in (\mathbb{S}^d)^r$, with $\mathbf{x}_j = (x_{j1}, \ldots, x_{j(d+1)})' \in \mathbb{S}^d$ for $j = 1, \ldots, r$, and set $\mathbf{h} := (h_1, \ldots, h_r)' \in \mathbb{R}_+^r$. We consider the kernel density estimator (kde) of f at $\mathbf{x}$ defined as

$$\hat{f}(\mathbf{x}; \mathbf{h}) := \frac{1}{n} \sum_{i=1}^{n} L_{\mathbf{h}}(\mathbf{x}, \mathbf{X}_i),\ L_{\mathbf{h}}(\mathbf{x}, \mathbf{y}) := c_{\mathbf{d},L}(\mathbf{h}) L\left((1 - \mathbf{x}' \diamond \mathbf{y}) \odot \mathbf{h}^{\odot(-2)}\right), \tag{4.1}$$

$$L(\mathbf{s}) := \prod_{j=1}^{r} L_j(s_j), \quad c_{\mathbf{d},L}(\mathbf{h}) := \prod_{j=1}^{r} c_{d,L_j}(h_j), \tag{4.2}$$

where $\odot$ denotes the Hadamard product, $\mathbf{y} = (\mathbf{y}_1', \ldots, \mathbf{y}_r')' \in (\mathbb{S}^d)^r$, and

$$\mathbf{A} \diamond \mathbf{B} := \begin{pmatrix} \mathbf{A}_{11}\mathbf{B}_{11} & \cdots & \mathbf{A}_{1r}\mathbf{B}_{1r} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{r1}\mathbf{B}_{r1} & \cdots & \mathbf{A}_{rr}\mathbf{B}_{rr} \end{pmatrix}$$

stands for a "block-in-block matrix product" between the r-partitioned (either in their rows, columns, or both) matrices $\mathbf{A} = (\mathbf{A}_{ij})$ and $\mathbf{B} = (\mathbf{B}_{ij})$, $1 \le i, j \le r$. The type of r-partition of the two matrices involved in the product will be clear from the context given the product space structure; e.g., $\mathbf{x}' \diamond \mathbf{y} := (\mathbf{x}_1'\mathbf{y}_1, \ldots, \mathbf{x}_r'\mathbf{y}_r)' \in \mathbb{R}^r$. The normalizing constant of the jth kernel $L_j : \mathbb{R}_0^+ \to \mathbb{R}_0^+$ in (4.2) is defined as

$$c_{d,L_j}(h_j)^{-1} := \int_{\mathbb{S}^d} L_j\left(\frac{1 - \mathbf{x}_j'\mathbf{y}_j}{h_j^2}\right) \sigma_d(\mathrm{d}\mathbf{x}_j).$$

The most common kernel is the von Mises–Fisher (vMF) kernel, $L_{\mathrm{vMF}}(t) := e^{-t}$, for $t \ge 0$, although the "Epanechnikov" kernel, $L_{\mathrm{Epa}}(t) := (1-t)1_{\{0 \le t \le 1\}}$, is more efficient on $\mathbb{S}^d$. The normalizing constant for the vMF kernel is

$$c_{d,L_{\mathrm{vMF}}}(h) = \left[(2\pi)^{(d+1)/2}\mathcal{I}_{(d-1)/2}(h^{-2})e^{-1/h^2}h^{d-1}\right]^{-1}, \qquad (4.3)$$

where $\mathcal{I}_\nu$ is the modified Bessel function of the first kind and νth order. For $d = 2$, a numerically stable form for (4.3) when $h \approx 0$ is

$$\log(c_{2,L_{\mathrm{vMF}}}(h)) = -\left[2\log(h) + \log(2\pi) + \mathrm{log1p}\left(-e^{-2/h^2}\right)\right],$$

where $\mathrm{log1p}(x)$ is the numerically stable computation of $\log(1+x)$ for $x \approx 0$.

4.2.1.2 Gradient and Hessian Density Estimation

To derive the gradient and Hessian of the kernel density estimator introduced in (4.1), let us first consider the radial extension of $f : (\mathbb{S}^d)^r \to \mathbb{R}_0^+$ given by $\bar{f} : \mathbb{R}^{r(d+1)} \backslash \{\mathbf{0}\} \to \mathbb{R}_0^+$ such that $\bar{f}(\mathbf{x}) := f(\bar{\mathbf{x}})$, where $\bar{\mathbf{x}} := \mathrm{proj}_{(\mathbb{S}^d)^r}(\mathbf{x}) := (\mathbf{x}_1'/\|\mathbf{x}_1\|, \ldots, \mathbf{x}_r'/\|\mathbf{x}_r\|)' \in (\mathbb{S}^d)^r$, for $\mathbf{x} \in \mathbb{R}^{r(d+1)} \backslash \{\mathbf{0}\}$. The reason for this extension is the necessity of taking derivatives on f, defined on the closed support $(\mathbb{S}^d)^r$.

The following result provides the expressions of the gradient and Hessian of $\bar{f}(\mathbf{x})$, for $\mathbf{x} \in (\mathbb{S}^d)^r$. These statements are required for deriving the gradient and Hessian of the kernel density estimator, that is, $\bar{\hat{f}}(\mathbf{x}; \mathbf{h}) = \hat{f}(\mathbf{x}; \mathbf{h})$, for $\mathbf{x} \in (\mathbb{S}^d)^r$.

Proposition 1 *Assume that $\bar{f}$ is twice continuously differentiable on $(\mathbb{S}^d)^r$. Then:*

1. *The (row) gradient vector of* $\bar{f}$ *at* $\mathbf{x} \in (\mathbb{S}^d)^r$ *is* $\nabla \bar{f}(\mathbf{x}) = (\nabla_1 \bar{f}(\mathbf{x}), \ldots, \nabla_r \bar{f}(\mathbf{x}))$,

$$\nabla_j \bar{f}(\mathbf{x}) = \nabla_j f(\mathbf{x})(\mathbf{I}_{d+1} - \mathbf{x}_j \mathbf{x}_j'), \quad j = 1, \ldots, r,$$

   where $\mathbf{I}_p$ *stands for the identity matrix of size* p.
2. *The Hessian matrix of* $\bar{f}$ *at* $\mathbf{x} \in (\mathbb{S}^d)^r$ *is*

$$\mathcal{H}\bar{f}(\mathbf{x}) = \begin{pmatrix} \mathcal{H}_{11}\bar{f}(\mathbf{x}) & \cdots & \mathcal{H}_{1r}\bar{f}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \mathcal{H}_{1r}\bar{f}(\mathbf{x})' & \cdots & \mathcal{H}_{rr}\bar{f}(\mathbf{x}) \end{pmatrix},$$

   where

$$\begin{aligned} \mathcal{H}_{jj}\bar{f}(\mathbf{x}) &= (\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j')\mathcal{H}_{jj} f(\mathbf{x})(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j') - (\nabla_j f(\mathbf{x})\mathbf{x}_j)(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j') \\ &\quad - \left[\mathbf{x}_j \nabla_j f(\mathbf{x}) + (\mathbf{x}_j \nabla_j f(\mathbf{x}))' - 2(\nabla_j f(\mathbf{x})\mathbf{x}_j)\mathbf{x}_j\mathbf{x}_j'\right], \qquad (4.4) \\ \mathcal{H}_{kj}\bar{f}(\mathbf{x}) &= (\mathbf{I}_{d+1} - \mathbf{x}_k\mathbf{x}_k')\mathcal{H}_{kj} f(\mathbf{x})(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j'), \end{aligned}$$

   with $j, k = 1, \ldots, r$, $k \neq j$.

Remark 1 The addend (4.4) is the only jth-term in the gradient and Hessian expressions that is not orthogonal to the subspace spanned by $\mathbf{x}_j$.

Remark 2 For a kernel $L : \mathbb{R}_0^+ \to \mathbb{R}_0^+$, we have that $\nabla L((1 - \mathbf{x}'\mathbf{y})/h^2) = -h^{-2}L'((1 - \mathbf{x}'\mathbf{y})/h^2)\mathbf{y}'$ and $\mathcal{H}L((1 - \mathbf{x}'\mathbf{y})/h^2) = h^{-4}L''((1 - \mathbf{x}'\mathbf{y})/h^2)\mathbf{y}\mathbf{y}'$. This holds for the aforementioned kernels L_{vMF} and L_{Epa}. These kernel derivatives are: $L'_{\mathrm{vMF}}(t) = -e^{-t}$, $L''_{\mathrm{vMF}}(t) = e^{-t}$; $L'_{\mathrm{Epa}}(t) = -1_{\{0 \le t \le 1\}}$, $L''_{\mathrm{Epa}}(t) = 0$.

From Proposition 1 and Remark 2, the block gradients and Hessians of $\bar{\hat{f}}(\cdot; \mathbf{h})$, $\nabla_j \bar{\hat{f}}(\cdot; \mathbf{h})$ and $\mathcal{H}_{kj}\bar{\hat{f}}(\cdot; \mathbf{h})$, follow immediately from those of $\hat{f}(\cdot; \mathbf{h})$:

$$\begin{aligned} \nabla_j \hat{f}(\mathbf{x}; \mathbf{h}) &= -\frac{c_{d,L}(h_j)}{nh_j^2} \sum_{i=1}^n \left[L'\left(\frac{1 - \mathbf{x}_j'\mathbf{X}_{ij}}{h_j^2}\right) L_{-j,\mathbf{h}}(\mathbf{x}, \mathbf{X}_i) \right] \mathbf{X}_{ij}', \\ \mathcal{H}_{jj}\hat{f}(\mathbf{x}; \mathbf{h}) &= \frac{c_{d,L}(h_j)}{nh_j^4} \sum_{i=1}^n \left[L''\left(\frac{1 - \mathbf{x}_j'\mathbf{X}_{ij}}{h_j^2}\right) L_{-j,\mathbf{h}}(\mathbf{x}, \mathbf{X}_i) \right] \mathbf{X}_{ij}\mathbf{X}_{ij}', \\ \mathcal{H}_{kj}\hat{f}(\mathbf{x}; \mathbf{h}) &= \frac{c_{d,L}(h_k)c_{d,L}(h_j)}{nh_k^2h_j^2} \\ &\quad \times \sum_{i=1}^n \left[L'\left(\frac{1 - \mathbf{x}_k'\mathbf{X}_{ik}}{h_k^2}\right) L'\left(\frac{1 - \mathbf{x}_j'\mathbf{X}_{ij}}{h_j^2}\right) L_{-k,-j,\mathbf{h}}(\mathbf{x}, \mathbf{X}_i) \right] \\ &\quad \times \ \mathbf{X}_{ik}\mathbf{X}_{ij}', \end{aligned}$$

where $\mathbf{x} \in (\mathbb{S}^d)^r$ and

$$L_{-j,\mathbf{h}}(\mathbf{x},\mathbf{y}) := c_{\mathbf{d}_{-j},L}(\mathbf{h}_{-j}) L\left((1-\mathbf{x}'_{-j} \diamond \mathbf{y}_{-j}) \odot \mathbf{h}_{-j}^{\odot(-2)}\right),$$

$$L_{-k,-j,\mathbf{h}}(\mathbf{x},\mathbf{y}) := c_{\mathbf{d}_{-k,-j},L}(\mathbf{h}_{-k,-j}) L\left((1-\mathbf{x}'_{-k,-j} \diamond \mathbf{y}_{-k,-j}) \odot \mathbf{h}_{-k,-j}^{\odot(-2)}\right).$$

For the vMF kernel, the above gradient and Hessian expressions simplify to

$$\nabla_j \hat{f}(\mathbf{x};\mathbf{h}) = \frac{1}{nh_j^2} \sum_{i=1}^n L_{\mathbf{h}}(\mathbf{x},\mathbf{X}_i)\mathbf{X}'_{ij}, \quad \mathcal{H}_{kj}\hat{f}(\mathbf{x}) = \frac{1}{nh_k^2 h_j^2} \sum_{i=1}^n L_{\mathbf{h}}(\mathbf{x},\mathbf{X}_i)\mathbf{X}_{ik}\mathbf{X}'_{ij},$$

which can be further compressed into

$$\nabla \hat{f}(\mathbf{x};\mathbf{h}) = \mathbf{h}^{\odot(-2)} \diamond \left[\frac{1}{n}\sum_{i=1}^n L_{\mathbf{h}}(\mathbf{x},\mathbf{X}_i)\mathbf{X}'_i\right], \tag{4.5}$$

$$\mathcal{H}\hat{f}(\mathbf{x};\mathbf{h}) = (\mathbf{h}\mathbf{h}')^{\odot(-2)} \diamond \left[\frac{1}{n}\sum_{i=1}^n L_{\mathbf{h}}(\mathbf{x},\mathbf{X}_i)(\mathbf{X}_i \diamond \mathbf{X}'_i)\right]. \tag{4.6}$$

The simplicity of (4.5)–(4.6) is a major practical benefit of the vMF kernel. Therefore, this kernel is adopted henceforth, although the subsequent theory also holds for other kernels.

4.2.1.3 Polysphere-on-Scalar Regression Estimation

The indexing of density ridges benefits from an auxiliary smoothing of $(\mathbb{S}^d)^r$-valued data with respect to a scalar variable. This smoothing can be cast within a regression framework where one is interested in estimating the *extrinsic* regression function $t \in \mathbb{R} \mapsto m(t) := \mathrm{proj}_{(\mathbb{S}^d)^r}(\mathbb{E}[\mathbf{X}|T=t])$ given the iid sample $(T_1,\mathbf{X}_1),\ldots,(T_n,\mathbf{X}_n)$ on $\mathbb{R} \times (\mathbb{S}^d)^r$. An alternative *intrinsic* approach based on the conditional Fréchet mean is also possible, yet it would involve several issues (non-explicitness, non-uniqueness, and potential smeariness; see [16] on the latter).

Within this extrinsic regression setup, given $t \in \mathbb{R}$, we consider the Nadaraya–Watson estimator

$$\hat{m}(t;h) := \mathrm{proj}_{(\mathbb{S}^d)^r}\left(\sum_{i=1}^n W_i(t;h)\mathbf{X}_i\right), \quad W_i(t;h) := \frac{K_h(t-T_i)}{\sum_{j=1}^n K_h(t-T_j)}, \tag{4.7}$$

which acts as a weighted local mean informed by the scaled kernel $K_h(\cdot) = K(\cdot/h)/h$ (typically, a Gaussian pdf) and the bandwidth $h > 0$. Bandwidth selection for (4.7) can be approached by cross-validation:

$$\hat{h}_{\mathrm{CV}} := \arg\min_{h>0} \mathrm{CV}(h), \quad \mathrm{CV}(h) := \frac{1}{n}\sum_{i=1}^{n} d_{(\mathbb{S}^d)^r}\left(\mathbf{X}_i, \hat{m}_{-i}(T_i; h)\right)^2, \tag{4.8}$$

where $\hat{m}_{-i}(\cdot; h)$ denotes (4.7) computed without the ith observation and prevents a spurious overfitting. In (4.8), $d_{(\mathbb{S}^d)^r}$ stands for the geodesic distance on the product manifold $(\mathbb{S}^d)^r$, which arises from the Euclidean combination of geodesic distances on $\mathbb{S}^d$ (see, e.g., [13, p. 600]):

$$d_{(\mathbb{S}^d)^r}(\mathbf{x}, \mathbf{y}) = \left(\sum_{j=1}^{r}\left[\cos^{-1}(\mathbf{x}_j'\mathbf{y}_j)\right]^2\right)^{1/2}. \tag{4.9}$$

The cross-validated bandwidth (4.8) can be smoothed according to the *one-standard error rule* principle from the `glmnet` package [17]. The rule favors regression simplicity within a one-standard error neighborhood of $\mathrm{CV}(\hat{h}_{\mathrm{CV}})$, that is $\hat{h}_{1\mathrm{SE}} := \max\left\{h > 0 : \mathrm{CV}(h) = \mathrm{CV}(\hat{h}_{\mathrm{CV}}) + \hat{\mathrm{SE}}(\mathrm{CV}(\hat{h}_{\mathrm{CV}}))\right\}$, where $\hat{\mathrm{SE}}^2(\mathrm{CV}(h)) := \frac{1}{n-1}\sum_{i=1}^{n}(\mathrm{CV}_i(h) - \mathrm{CV}(h))^2$ and $\mathrm{CV}_i(h) := d_{(\mathbb{S}^d)^r}\left(\mathbf{X}_i, \hat{m}_{-i}(T_i; h)\right)^2$.

A faster and equivalent expression for $\mathrm{CV}(h)$ in (4.8) is given in the following result. For a sample size $n = 200$, the median computation time of evaluating $\mathrm{CV}(h)$ as described in Proposition 2 is approximately just 8.6% of that of the naive form (4.8).

Proposition 2 *Let $\tilde{\mathbf{K}}$ and $\tilde{\mathbf{W}}$ be $n \times n$ matrices with ij-entries $k_{ij} := (1 - \delta_{ij})K_h(T_i - T_j)$ and $w_{ij} := k_{ij}/(\sum_{j=1}^{n} k_{ij})$, respectively, where δ_{ij} denotes Kronecker's delta and $i, j = 1, \ldots, n$. Let $\tilde{\mathbb{X}} := \tilde{\mathbf{W}}\mathbb{X}$, where $\mathbb{X}$ is the $n \times (r(d+1))$ response matrix whose rows are $\mathbf{X}_1', \ldots, \mathbf{X}_n'$. Then $\mathrm{CV}(h) = \sum_{i=1}^{n} d_{(\mathbb{S}^d)^r}\left(\mathbf{X}_i, \mathrm{proj}_{(\mathbb{S}^d)^r}(\tilde{\mathbf{X}}_i)\right)^2$, where $\tilde{\mathbf{X}}_i'$ is the ith row of $\tilde{\mathbb{X}}$.*

A more sophisticated local polynomial estimator could be considered instead of (4.7), yet with higher computational cost and higher variability at low-density regions.

4.2.2 Density Ridges

4.2.2.1 Population Euclidean Case

Density ridges are higher-dimensional extensions of the concept of mode that inform on the main features of a density f on $\mathbb{R}^p$. Density ridges are defined through the gradient and Hessian of f. In particular, they require the eigendecomposition $\mathcal{H}f(\mathbf{x}) = \mathbf{U}(\mathbf{x})\boldsymbol{\Lambda}(\mathbf{x})\mathbf{U}(\mathbf{x})'$, for $\mathbf{x} \in \mathbb{R}^p$, where $\mathbf{U}(\mathbf{x}) = (\mathbf{u}_1(\mathbf{x}), \ldots, \mathbf{u}_p(\mathbf{x}))$ is a matrix whose columns are the eigenvectors and $\boldsymbol{\Lambda}(\mathbf{x}) = \mathrm{diag}(\lambda_1(\mathbf{x}), \ldots,$

$\lambda_p(\mathbf{x}))$, $\lambda_1(\mathbf{x}) \geq \ldots \geq \lambda_p(\mathbf{x})$, contains the corresponding eigenvalues. Denoting $\mathbf{U}_{(p-1)}(\mathbf{x}) := (\mathbf{u}_2(\mathbf{x}), \ldots, \mathbf{u}_p(\mathbf{x}))$, the projected gradient onto $\{\mathbf{u}_2(\mathbf{x}), \ldots, \mathbf{u}_p(\mathbf{x})\}$ is

$$\nabla_{(p-1)} f(\mathbf{x}) := \nabla f(\mathbf{x}) \mathbf{U}_{(p-1)}(\mathbf{x}) \mathbf{U}_{(p-1)}(\mathbf{x})'. \tag{4.10}$$

The density ridge of f is defined by [15] as the set

$$\mathcal{R}(f) := \left\{ \mathbf{x} \in \mathbb{R}^p : \|\nabla_{(p-1)} f(\mathbf{x})'\| = 0,\ \lambda_2(\mathbf{x}), \ldots, \lambda_p(\mathbf{x}) < 0 \right\}. \tag{4.11}$$

Note that $\mathbf{x} \in \mathcal{R}(f)$ if either $\mathbf{x}$ is a maximum or a saddle point, or $\nabla f(\mathbf{x})'$ is parallel to $\mathbf{u}_1(\mathbf{x})$, i.e., the directions of maximum ascent and largest (signed) curvature coincide.

To determine $\mathcal{R}(f)$ in practice, assuming that f is a known density on $\mathbb{R}^p$, an iterative Euler algorithm that starts at an arbitrary point $\mathbf{x}_0 \in \mathbb{R}^p$ and converges to a certain point in $\mathcal{R}(f)$ is often used. The algorithm is based on the updating

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{H}\, \boldsymbol{\eta}_{(p-1)}(\mathbf{x}_t)' \tag{4.12}$$

until convergence, using a step matrix $\mathbf{H}$ and the normalized projected gradient

$$\boldsymbol{\eta}_{(p-1)}(\mathbf{x}) := \nabla_{(p-1)} f(\mathbf{x}) / f(\mathbf{x}). \tag{4.13}$$

The gradient (4.13) boosts the passing through low-density regions and modulates its magnitude at high-density regions.

We refer to [15] and [18, Section 6.3] for further details on the population case and for the exposition of the sample version. For the sake of brevity, we directly address next the sample polyspherical case.

4.2.2.2 Sample Polyspherical Case

We turn on back to the setting in the present work: a density f supported over $(\mathbb{S}^d)^r \subset \mathbb{R}^p$, with $p = r(d+1)$ henceforth, that is unknown. The recipe for estimating density ridges from a sample $\mathbf{X}_1, \ldots, \mathbf{X}_n$ on $(\mathbb{S}^d)^r$ rests on two main adaptations: (*i*) plug-in the kde $\hat{f}(\cdot; \mathbf{h})$ instead of f in (4.10), (4.11), and (4.13); (*ii*) conform to the polysphere $(\mathbb{S}^d)^r$ the Euler step given in (4.12).

The projected gradient of $\hat{f}(\cdot; \mathbf{h})$ involves the extended gradient $\nabla \bar{\hat{f}}(\cdot; \mathbf{h})$ and Hessian $\mathcal{H}\bar{\hat{f}}(\cdot; \mathbf{h})$ obtained in Proposition 1. However, some care is needed, as the direct translation of (4.10) to $(\mathbb{S}^d)^r$ leads to three important issues. First, repeatedly computing the full eigendecomposition $\mathcal{H}\bar{\hat{f}}(\mathbf{x}; \mathbf{h}) = \hat{\mathbf{U}}(\mathbf{x}; \mathbf{h}) \hat{\boldsymbol{\Lambda}}(\mathbf{x}; \mathbf{h}) \hat{\mathbf{U}}(\mathbf{x}; \mathbf{h})'$ for $\mathbf{x} \in (\mathbb{S}^d)^r$ is expensive, especially for large p. However, due to orthogonality, $\hat{\mathbf{U}}_{(p-1)}(\mathbf{x}; \mathbf{h}) \hat{\mathbf{U}}_{(p-1)}(\mathbf{x}; \mathbf{h})' = \mathbf{I}_p - \hat{\mathbf{u}}_1(\mathbf{x}; \mathbf{h}) \hat{\mathbf{u}}_1(\mathbf{x}; \mathbf{h})'$, and this expression has the advantage of involving only the first eigenvector $\hat{\mathbf{u}}_1(\mathbf{x}; \mathbf{h})$ and not the full eigendecomposition. The computation of the first eigenvector, or a set of prescribed eigenvectors, can be done efficiently with the implicitly restarted Arnoldi algorithm in `ARPACK` [19], ported in Armadillo's `eigs_sym` [20]. Second, as advanced in Remark 1, the Hessian $\mathcal{H}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$ has a component that is non-orthogonal to $\mathbf{x}$ and that corresponds to the terms (4.4) in the r diagonal blocks. Due to the specificity of $(\mathbb{S}^d)^r$, this component has to be subtracted before the gradient projection: if included, $\mathbf{I}_p - \hat{\mathbf{u}}_1(\mathbf{x};\mathbf{h})\hat{\mathbf{u}}_1(\mathbf{x};\mathbf{h})'$ would be projecting the gradient $\boldsymbol{\nabla}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$ partly along $\mathbf{x}$, that is, outside the tangent space at $\mathbf{x}$, spanned by $\mathbf{I}_p - \mathrm{diag}(\mathbf{x}_1\mathbf{x}_1', \ldots, \mathbf{x}_r\mathbf{x}_r')$, where $\boldsymbol{\nabla}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$ lies. We denote by $\widetilde{\mathcal{H}}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$ the Hessian matrix projected on the orthogonal space to $\mathbf{x}$ that does not include the terms (4.4) in each of the r diagonal blocks of $\mathcal{H}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$. Third, $\widetilde{\mathcal{H}}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$ has r null eigenvalues, which is apparent given the mismatch between p and dr, the intrinsic dimension of $(\mathbb{S}^d)^r$. If they are not specifically filtered out, null eigenvalues in the form of machine epsilons may arise in the r largest (signed) eigenvalues.

Taking into account the three previous issues, we denote with $\tilde{\mathbf{u}}_1(\mathbf{x};\mathbf{h})$ the eigenvector associated with the largest (signed) non-null eigenvalue of $\widetilde{\mathcal{H}}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$. Then, we define the kde analog of the projected gradient (4.10) as

$$\boldsymbol{\nabla}_{(p-1)}\bar{\hat{f}}(\mathbf{x};\mathbf{h}) := \boldsymbol{\nabla}\bar{\hat{f}}(\mathbf{x};\mathbf{h})\big(\mathbf{I}_p - \tilde{\mathbf{u}}_1(\mathbf{x};\mathbf{h})\tilde{\mathbf{u}}_1(\mathbf{x};\mathbf{h})'\big). \tag{4.14}$$

The kde-normalized projected gradient is then defined as

$$\hat{\boldsymbol{\eta}}_{(p-1)}(\mathbf{x};\mathbf{h}) := \boldsymbol{\nabla}_{(p-1)}\bar{\hat{f}}(\mathbf{x};\mathbf{h})/\hat{f}(\mathbf{x};\mathbf{h}). \tag{4.15}$$

The Euler step (4.12) transforms into

$$\mathbf{x}_{t+1} := \mathrm{proj}_{(\mathbb{S}^d)^r}\big(\mathbf{x}_t + \mathbf{h}^{\odot 2} \diamond \hat{\boldsymbol{\eta}}_{(p-1)}(\mathbf{x}_t;\mathbf{h})'\big). \tag{4.16}$$

In (4.16), $\mathrm{proj}_{(\mathbb{S}^d)^r}$ preserves each new iteration within $(\mathbb{S}^d)^r$ and $\mathbf{h}^{\odot 2} \diamond \hat{\boldsymbol{\eta}}_{(p-1)}(\mathbf{x}_t;\mathbf{h})' = \big(h_1^2\hat{\boldsymbol{\eta}}_{1,(p-1)}(\mathbf{x}_t;\mathbf{h}), \ldots, h_r^2\hat{\boldsymbol{\eta}}_{r,(p-1)}(\mathbf{x}_t;\mathbf{h})\big)'$ multiplies the jth projected gradient according to the corresponding squared bandwidth. Squares appear as an analogy to the Euclidean case (4.12), where $\mathbf{H}$, not $\mathbf{H}^{1/2}$, is considered to modulate the Euler step [18, Section 6.3]. The recurrence (4.16) is iterated until convergence, when $\mathbf{x}_{t+1}$ approximately belongs to the sample version of the ridge:

$$\mathcal{R}\big(\hat{f}(\cdot;\mathbf{h})\big) := \big\{\mathbf{x} \in (\mathbb{S}^d)^r : \|\boldsymbol{\nabla}_{(p-1)}\bar{\hat{f}}(\cdot;\mathbf{h})(\mathbf{x})'\| = 0,\ \tilde{\lambda}_2(\mathbf{x};\mathbf{h}), \ldots, \tilde{\lambda}_{dr}(\mathbf{x};\mathbf{h}) < 0\big\},$$

being $\tilde{\lambda}_2(\mathbf{x};\mathbf{h}) > \ldots > \tilde{\lambda}_{dr}(\mathbf{x};\mathbf{h})$ the non-null eigenvalues of $\widetilde{\mathcal{H}}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$.

Computing the gradient and Hessian behind (4.14) in high-dimensional setups has to be done carefully, as their entries quickly underflow. Thanks to (4.5) and (4.6), this issue can be prevented by (*i*) working in logarithmic scale and (*ii*) computing rather the gradient and Hessian standardized by the kde (4.1). Obviously, dividing by

$\hat{f}(\mathbf{x};\mathbf{h})$ does not affect the eigenvectors of $\widetilde{\mathcal{H}}\bar{\hat{f}}(\mathbf{x};\mathbf{h})$, yet it makes them numerically stable.

To guide the discussion of the specifics in the proposed density ridge estimation procedure on $(\mathbb{S}^d)^r$, we summarize in Algorithm 1 its main steps.

Algorithm 1 Ridge estimation and indexing on $(\mathbb{S}^d)^r$

Given a sample $\mathbf{X}_1,\ldots,\mathbf{X}_n$ on $(\mathbb{S}^d)^r$, its estimated ridge is determined and indexed as follows:

1. Select a "suitable" data-driven bandwidth $\hat{\mathbf{h}}$ (Sect. 4.2.2.3).
2. For each element in an initial grid $\{\mathbf{x}_{0,1},\ldots,\mathbf{x}_{0,m}\}\subset(\mathbb{S}^d)^r$, iterate (4.16) "until convergence" to a given $\mathbf{x}_j$, $j=1,\ldots,m$ (Sect. 4.2.2.4).
3. "Index" the estimated ridge $\hat{\mathcal{R}}\big(\hat{f}(\cdot;\hat{\mathbf{h}})\big):=\{\mathbf{x}_1,\ldots,\mathbf{x}_m\}\subset(\mathbb{S}^d)^r$ and assign "scores" to $\mathbf{X}_1,\ldots,\mathbf{X}_n$ (Sect. 4.2.2.5).

4.2.2.3 Bandwidth Selection

Bandwidth selection in Step 1 can be done with "upscaled versions" of plug-in bandwidths. [21] proposed a simple plug-in bandwidth selector for the kde (4.1) on $\mathbb{S}^d$. This estimator is an analog to Silverman's rule-of-thumb [22], as it assumes that the underlying population is a vMF distribution with concentration κ to estimate the curvature term present in the so-called Asymptotic Mean Integrated Squared Error (AMISE) bandwidth. Within the setting of the present work, the marginal bandwidth selector in the jth $\mathbb{S}^d$ is

$$\hat{h}_{j,\mathrm{ROT}}:=\left[\frac{4\pi^{1/2}\mathcal{I}_{(d-1)/2}(\hat{\kappa}_j)^2}{\hat{\kappa}_j^{(d+1)/2}\big[2d\mathcal{I}_{(d+1)/2}(2\hat{\kappa}_j)+(2+d)\hat{\kappa}_j\mathcal{I}_{(d+3)/2}(2\hat{\kappa}_j)\big]n}\right]^{1/(4+d)}, \tag{4.17}$$

where $\hat{\kappa}_j:=A_d^{-1}(\|\frac{1}{n}\sum_{i=1}^n\mathbf{X}_{ij}\|)$, with $A_d(r):=\mathcal{I}_{(d+1)/2}(r)/\mathcal{I}_{(d-1)/2}(r)$, is the maximum likelihood estimate of κ_j. Independently combining the marginal bandwidth selectors (4.17) gives $\hat{\mathbf{h}}_{\mathrm{IROT}}:=\big(\hat{h}_{1,\mathrm{ROT}},\ldots,\hat{h}_{r,\mathrm{ROT}}\big)'\in\mathbb{R}_+^r$. This admittedly simple selector is explicit and easy to compute, but it undersmooths the underlying density in $(\mathbb{S}^d)^r$. Besides, following the discussion in [18, Section 6.3], the kind of bandwidth selectors recommended for density ridge estimation are the ones designed for Hessian density estimation, since (4.15) critically depends on adequately estimating the Hessian's first eigenvector. To solve both issues in a computationally tractable manner, given the current lack of theory for derivative bandwidth selectors on $(\mathbb{S}^d)^r$, we consider the following upscaled version of $\hat{\mathbf{h}}_{\mathrm{IROT}}$:

$$\hat{\mathbf{h}}_{\mathrm{UIROT}}^{(s)}:=\hat{\mathbf{h}}_{\mathrm{IROT}}\times n^{1/(d+4)-1/(dr+2s+4)},$$

where s denotes the order of the derivatives of f that are being estimated. The entries of $\hat{\mathbf{h}}^{(s)}_{\mathrm{UIROT}}$ have order $O\big(n^{-1/(dr+2s+4)}\big)$, i.e., the standard rate an AMISE bandwidth for the sth derivatives of a pdf on $\mathbb{R}^{dr}$ has [18, Section 5.5]. We will consider $C \times \hat{\mathbf{h}}^{(2)}_{\mathrm{UIROT}}$ in Step 1, where $C > 0$ is determined experimentally.

4.2.2.4 Euler Iteration

An important practical issue is to initiate (4.16) in Step 2 from a sensible grid of points $\{\mathbf{x}_{0,1}, \ldots, \mathbf{x}_{0,m}\} \subset (\mathbb{S}^d)^r$. This can be challenging in $(\mathbb{S}^d)^r$ due to two reasons: (*i*) the likely vastness of the domain, which forbids considering a product of uniform-like grids on each $\mathbb{S}^d$ (besides, such uniform grids are unknown for $d > 1$); (*ii*) the ubiquitous low-density regions, with associated long convergence paths to ridge points that are usually spurious. Solutions to both problems include setting the initial grid by sampling from $\hat{f}\big(\cdot; \hat{\mathbf{h}}\big)$ or by directly using the sample $\mathbf{X}_1, \ldots, \mathbf{X}_n$, thus building data-driven grids across $(\mathbb{S}^d)^r$ adapted to the purpose of Algorithm 1.

In practice, the iteration of the recurrence (4.16) can be done for a maximum number of iterations N or until a certain stopping ε-criterion on the standardized version of distance (4.9) is met: $(d_{(\mathbb{S}^d)^r}(\mathbf{x}_{t+1}, \mathbf{x}_t)/(\pi\sqrt{r})) < \varepsilon$. The standardization allows securing the same accuracy within different polyspheres. In our experiments, we found that $N = 1000$ and $\varepsilon = 10^{-5}$ gave a good accuracy–speed trade-off.

When applying Step 2 on a high-dimensional space $(\mathbb{S}^d)^r$, we have found that a convenient way to speed up and monitor the obtention of the ridge on $(\mathbb{S}^d)^r$ is to initialize the (expensive) Euler algorithm with the endpoints of (much faster) marginal Euler algorithms on each of the r/k blocks formed by $(\mathbb{S}^d)^k$. This process can be refined by using ℓ passes forming the sequence $1 \leq k_1 < \ldots < k_\ell = r$.

Finally, in practice Step 2 is followed by a filtering process that removes spurious endpoints $\mathbf{x}_j$ meeting any of the next conditions: (*i*) ε-convergence was not achieved in N iterations; (*ii*) $\tilde{\lambda}_2(\mathbf{x}_j; \hat{\mathbf{h}}) \geq 0$; (*iii*) $\hat{f}(\mathbf{x}_j; \hat{\mathbf{h}}) < \hat{f}_\alpha$, where $\hat{f}_\alpha$ is the α-quantile of $\hat{f}(\mathbf{X}_1; \hat{\mathbf{h}}), \ldots, \hat{f}(\mathbf{X}_n; \hat{\mathbf{h}})$ for, say, $\alpha = 0.01$ (i.e., $\mathbf{x}_j$ is in a low-density region).

4.2.2.5 Indexing Ridges

The estimated ridge $\hat{\mathcal{R}}\big(\hat{f}\big(\cdot; \hat{\mathbf{h}}\big)\big) = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\} \subset (\mathbb{S}^d)^r$ obtained in Step 2 is a set of points without an explicit notion of order. To build a flexible analog of a first principal component, an indexing of $\hat{\mathcal{R}}\big(\hat{f}\big(\cdot; \hat{\mathbf{h}}\big)\big)$ is essential. Inspired by the use of MultiDimensional Scaling (MDS) in [23] for non-Euclidean dimension-reduction purposes, we advocate the use of a metric MDS (see, e.g., Section 9.1 in [24]) on the matrix of geodesic distances $\mathbf{D} := (d_{(\mathbb{S}^d)^r}(\mathbf{x}_i, \mathbf{x}_j))$, with $1 \leq i, j \leq m$. Metric MDS from $(\mathbb{S}^d)^r$ to $\mathbb{R}$ produces

$$(\hat{t}_1, \ldots, \hat{t}_m) = \arg \min_{t_1,\ldots,t_m \in \mathbb{R}} \sum_{i,j=1}^{m} \left(d_{(\mathbb{S}^d)^r}(\mathbf{x}_i, \mathbf{x}_j) - |t_i - t_j| \right)^2. \tag{4.18}$$

The indexes $\hat{t}_1, \ldots, \hat{t}_m$ give an effective handle to traverse $\hat{\mathcal{R}}\big(\hat{f}\left(\cdot; \hat{\mathbf{h}}\right)\big)$. Optimization of (4.18) can be done with the `smacof` package [25].

The smoother (4.7) becomes now relevant to (*i*) smooth out irregularities in the estimated ridge and, more importantly, (*ii*) evaluate the ridge at arbitrary indexes beyond those in (4.18). Consequently, we define the *Smoothed–Indexed–Estimated Ridge* (SIER) as the curve $t \in \mathbb{R} \mapsto \hat{\mathbf{r}}(t; h) \in (\mathbb{S}^d)^r$ generated by (4.7) acting on the sample $(\hat{t}_1, \mathbf{x}_1), \ldots, (\hat{t}_m, \mathbf{x}_m)$. $\hat{\mathcal{R}}\big(\hat{f}\left(\cdot; \hat{\mathbf{h}}\right)\big)$ neither contains nor is contained by $\{\hat{\mathbf{r}}(t; h) : t \in \mathbb{R}\}$, yet the latter can be regarded as the estimated mean of the former.

The score of an arbitrary point $\mathbf{x} \in (\mathbb{S}^d)^r$ on the SIER is defined as the index of its projection on the SIER curve:

$$\mathrm{score}_{\hat{\mathbf{r}}(\cdot;h)}(\mathbf{x}) := \arg \min_{t \in \mathbb{R}} d_{(\mathbb{S}^d)^r}\left(\hat{\mathbf{r}}(t; h), \mathbf{x}\right). \tag{4.19}$$

Note that $\mathbf{x}_j$, the "Euler-projection" of $\mathbf{x}_{j,0}$, and the projection $\mathrm{proj}_{\hat{\mathbf{r}}(;h)}(\mathbf{x}_{j,0}) := \hat{\mathbf{r}}\big(\mathrm{score}_{\hat{\mathbf{r}}(\cdot;h)}(\mathbf{x}_{j,0}); h\big)$ can be very different since the Euler paths follow the projected gradient flow and not the geodesic to the closest point on the ridge. This difference is clearly illustrated in Fig. 4.1, where the Euler-projections introduce distortions in the color gradient of the triangles (e.g., longest blue and green paths), which are not present in the sample scores $\{\mathrm{score}_{\hat{\mathbf{r}}(\cdot;h)}(\mathbf{X}_i)\}_{i=1}^n$ shown in the rug of the right plot.

4.3 Results

4.3.1 An Illustrative Numerical Example

We demonstrate the performance of Algorithm 1 for dimension-reduction with a numerical example on $(\mathbb{S}^2)^2$. The left and central panels of Fig. 4.1 display a sample of size $n = 200$ in solid points. The dependence pattern on each $\mathbb{S}^2$ follows a small circle variation that is coupled between $\mathbb{S}^2$'s and that is indicated by a *common* rainbow palette; i.e., points with the same colors in the two panels represent the $\mathbb{S}^2$-coordinates of a single point on $(\mathbb{S}^2)^2$. The Euler paths arising from running Algorithm 1 taking the sample as the initial grid are shown in transparent color. These paths converge to the triangular points defining $\hat{\mathcal{R}}\big(\hat{f}\left(\cdot; \hat{\mathbf{h}}\right)\big)$. The SIER, shown in the black curves, is then obtained with $C = 2$ and $\hat{h}_{1\mathrm{SE}}$ (Sect. 4.2.1.3) for (4.7).

The right panel of Fig. 4.1 shows the scores of the sample points on the SIER, evidencing that the color gradient encoding the one-dimensional mode of variation of the data is recovered (rainbow rug). Indeed, the Spearman correlation between the order of the colors and the order of the sample scores is 0.9999.

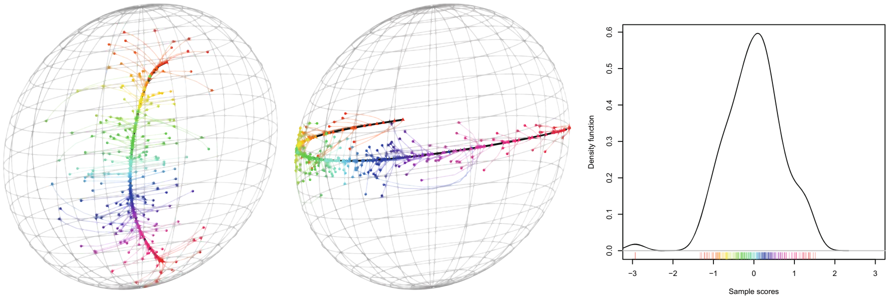

Fig. 4.1 Numerical example on $(\mathbb{S}^2)^2$. The left and central plots display the (joint) main mode of variation of the data, encoded by a common rainbow color palette. The sample is shown in solid points, the Euler paths in transparent curves, and the ridge points in triangles. The black curves represent the two $\mathbb{S}^2$-views of the common SIER. The right plot shows the kde of the sample scores

4.3.2 Main Mode of Variation of Hippocampus Shapes

We analyze now the hippocampi dataset mentioned in Sect. 4.1. The data consists of $n = 177$ hippocampi parametrized using s-reps, where each of the subjects has $r = 168$ spokes. The s-reps were fitted to a set of binary images of the hippocampi that were segmented from magnetic resonance imaging [3]. Fixing the radii of these vectors to their sample means, hence taking into account only the shape of the hippocampus and not its size, each s-rep is reduced to a value on $(\mathbb{S}^2)^r$.

A main form of variation is not well-defined in densities that are rotationally symmetric and unimodal about a certain location, as it distinctively happens with the vMF distribution. To detect such cases delivering spurious ridges, we run in each of the $r = 168$ samples of spokes the hybrid test for rotational symmetry with unspecified location from [26], implemented in `rotasym` [27]. To control for false discoveries, we corrected the r resulting p-values using the false discovery rate by [28]. For a conservative 1% significance level, $r^* = 88$ non-rotationally symmetric spokes were found, for which we then ran Algorithm 1 on $(\mathbb{S}^2)^{r^*}$. Figure 4.2 shows in color the r^* non-rotationally symmetric spokes and in gray the $r - r^* = 89$ rotationally symmetric spokes.

As advanced in Sect. 4.2.2.4, Algorithm 1 was run in a blockwise fashion to facilitate faster convergences. Precisely, $\ell = 3$ passes were applied to r^*/k_l blocks of sizes $k_1 = 1$, $k_2 = 22$, and $k_3 = r^*$. The initial grid for the first pass was set as the sample, and then subsequent passes were fed with the endpoints of the former pass. After each pass, the spurious endpoints were removed ($\alpha = 0.01$) to prevent their propagation into long convergence paths, hence successively trimming the size of the initial grids. The bandwidths applied on each pass were $C_l \times \hat{\mathbf{h}}^{(2)}_{\mathrm{UIROT}}$ (adapted to $(\mathbb{S}^2)^{k_l}$), with $C_l = 2^l$ experimentally determined, for $l = 1, 2, 3$. Our implementation of Algorithm 1 based on a hybrid of C++ (for the core routines) and R (for interfacing) yielded running times of 64, 170, and 3121 s for the three

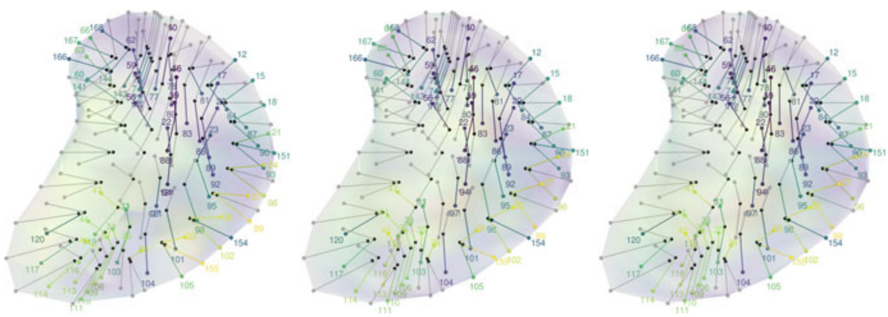

Fig. 4.2 March along the SIER of hippocampi. From left to right, the three plots show the reconstructed hippocampi for the score quantiles 1%, 50%, and 99%. In them, $r^* = 88$ non-rotationally symmetric spokes (colored) vary along the march, while the remaining $r - r^* = 89$ spokes (in gray) remain fixed at their spherical means. The yellow/purple color gradient codifies the large/small degree of change along the march. The black points are the average inner skeletal points of the n hippocampi. Surfaces were constructed using `alphashape3d` [29]

respective blocks. These times were measured on an Apple M1 processor. The fast runs for blockwise fits were convenient for quick exploration of approximate ridges. Finally, the SIER was computed with $\hat{h}_{1\mathrm{SE}}$. Replicating code is available from the authors upon reasonable request.

Figure 4.2 depicts the main outcome of our dimension-reduction tool on the hippocampi dataset: the march along the SIER, instantiated for the sake of conciseness at the quantiles 1%, 50%, and 99% of the sample scores. The coloring indicates that the largest variation appears at the yellow/green spokes (e.g., see spokes 28 and 99), with purple indicating virtually no variation, and gray denoting rotationally symmetric spokes. This march shows that: (*i*) most of the variation is concentrated at the spokes describing the sharpness of the elongated convex edge (right-positioned in the plots), and at the narrowest extreme of the hippocampus form (bottom); (*ii*) the joint variation of the previous spokes is elucidated as a "synchronous opening of pincers" given by pairs of spokes, i.e., there is a variation gradient from sharper to thicker edges in the hippocampus shapes; (*iii*) low variation occurs on the normal spokes to the elongated form of the hippocampus; (*iv*) the concave edge (left) and the widest extreme (top) concentrate most of the rotationally symmetric spokes; (*v*) on overall, the determined main shape variation across subjects is mild. Figure 4.3 shows the $\mathbb{S}^2$-projections of the $(\mathbb{S}^2)^{r^*}$-valued SIER, indicating with a rainbow palette the score-driven march along the SIER for which three quantiles were shown in Fig. 4.2. The density of the scores in Fig. 4.3 points towards an asymmetric distribution of the subjects, with a secondary cluster at the right of the main mode.

The scores given by the SIER serve to identify the "median hippocampus shape" and "most extreme hippocampus shape" in a straightforward way, given its univariate nature. Indeed, we define the first as the hippocampus whose score is the median of the scores (-0.30, see the rightmost plot of Fig. 4.3), while we set the second as that hippocampus with the largest absolute score (10.33). These

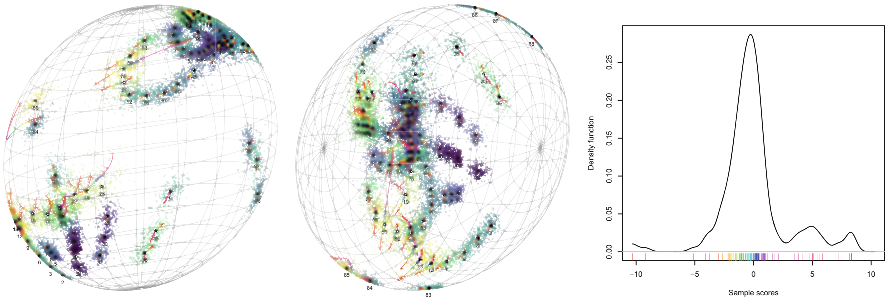

Fig. 4.3 The left and central plots show two views of the same $\mathbb{S}^2$ in which (*i*) each of the $n = 177$ directions of the $r^* = 88$ spokes has been drawn (colored points) and (*ii*) the r^* $\mathbb{S}^2$-projections of the $(\mathbb{S}^2)^{r^*}$-valued SIER are jointly plotted. The yellow/purple color gradient of the directions is assigned according to the spoke to which they belong. The rainbow palette is common to the r^* SIER $\mathbb{S}^2$-projections and is determined from the order of the sample scores (right plot)

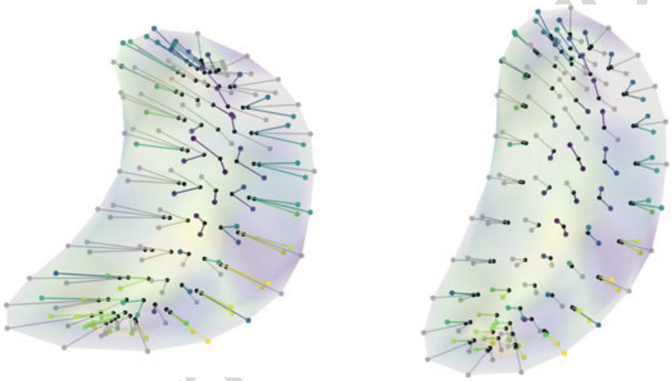

Fig. 4.4 Hippocampus shapes corresponding to the most extreme (left) and median (right) hippocampus. The first corresponds to the hippocampus with the largest absolute score, while the second is the hippocampus whose score is the median of the sample scores (see Fig. 4.3)

particular hippocampi are depicted in Fig. 4.4. The medial hippocampus is highly symmetric and has a small curvature along its elongated direction. The most extreme hippocampus (left plot) is a vertically-squeezed hippocampus that is notably thick, especially in the upper part displayed in Fig. 4.4. Indeed, the height ratio between the upper and lower parts is unusually high (not visible in Fig. 4.4). In opposition to the medial hippocampus, its elongated convex edge is markedly curved and asymmetric.

4.4 Discussion

A new fully nonparametric dimension-reduction procedure for finding the main mode of variability of the shape of s-reps was introduced in this chapter. The technique targets the polyspherical reduction of s-reps and provides a complete pipeline for attaining an analog of the first principal component on $(\mathbb{S}^d)^r$ based on density ridges. As demonstrated with the hippocampi dataset, the tool can be used for flexible dimension-reduction analyses in medical applications involving s-reps, also in high-dimensional settings, that deliver useful visualizations and insights.

The proposed technique presents some limitations and is subject to future improvements. As in any kernel-based method, bandwidth selection is a crucial issue. In that regard, the upscaled marginal bandwidths are simple choices open to large enhancements with the development of a theory for density derivative estimation on $(\mathbb{S}^d)^r$, for example, in the direction of cross-validatory or plug-in methods. From an application standpoint, the presented analysis also has certain limitations that constitute opportunities for further research. Arguably, the most relevant improvement would be a more holistic approach to determining the main mode of variation of the hippocampi involving the radii of the spokes and the position of the inner skeletal, as well as their interactions with the directions of the spokes. Addressing this case in a fully nonparametric way would involve substantial further complexities in the definition and estimation of the involved density ridges, caused by the notable increment of dimension and the different nature of the components involved.

The presented methodology, ultimately instantiated with the SIER march and scores, has medical potential in regard to analyzing the shapes of hippocampi and other three-dimensional objects parametrized by s-reps. On the one hand, it delivers a rich exploratory data analysis of the morphology of these objects, either through the SIER march or through the sample scores. The univariate scores, in particular, allow investigating the existence of possible clusters, determining the most prototypical subjects, and outlier-hunting signaling abnormal shapes. On the other hand, the methodology can be applied to obtain effective comparisons between treatment and control groups, either through the SIER march (visual, qualitative) or through the metrics on the distribution of the sample scores (quantitative).

Acknowledgments Both authors acknowledge support from grant PID2021-124051NB-I00, funded by MCIN/AEI/10.13039/501100011033 and by "ERDF A way of making Europe". The authors greatly acknowledge Prof. Stephen M. Pizer and Dr. Zhiyuan Liu (University of North Carolina at Chapel Hill) for kindly providing the analyzed s-reps hippocampi data. Comments by the editor and a referee are appreciated.

Proofs

Proof (Proposition 1) For $\bar{\mathbf{x}} = (\mathbf{x}_1'/\|\mathbf{x}_1\|, \ldots, \mathbf{x}_r'/\|\mathbf{x}_r\|)' =: (\bar{\mathbf{x}}_1', \ldots, \bar{\mathbf{x}}_r')' \in (\mathbb{S}^d)^r$,

$$\frac{\partial \bar{\mathbf{x}}_j}{\partial x_{ij}} = \|\mathbf{x}_j\|^{-3}\Big(\|\mathbf{x}_j\|^2\mathbf{e}_i - x_{ij}\mathbf{x}_j\Big) \quad \text{and} \quad \frac{\partial \bar{\mathbf{x}}_j}{\partial x_{ik}} = 0,$$

where $\mathbf{e}_i$ is the ith canonical vector of $\mathbb{R}^{d+1}$, $i = 1, \ldots, d+1$, and $j, k = 1, \ldots, r$, $j \neq k$. It now follows that, for $\mathbf{x} \in (\mathbb{R}^{d+1})^r$,

$$\frac{\partial}{\partial x_{ij}} f(\bar{\mathbf{x}}) = \|\mathbf{x}_j\|^{-3} \boldsymbol{\nabla}_j f(\bar{\mathbf{x}}) \Big(\|\mathbf{x}_j\|^2 \mathbf{e}_i - x_{ij}\mathbf{x}_j \Big), \quad j = 1, \ldots, r, \quad i = 1, \ldots, d+1.$$

Hence, for $\mathbf{x} \in (\mathbb{S}^d)^r$,

$$\boldsymbol{\nabla}_j \bar{f}(\mathbf{x}) = \boldsymbol{\nabla}_j f(\mathbf{x})(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j'), \quad j = 1, \ldots, r.$$

To obtain the Hessian of $\bar{f}$, we first compute the entries of $\mathcal{H}_{jj}\bar{f}(\mathbf{x})$ for $\mathbf{x} \in (\mathbb{R}^{d+1})^r$ and $j = 1, \ldots, r$:

$$\begin{aligned}
&\frac{\partial^2}{\partial x_{pj}\partial x_{qj}} \bar{f}(\mathbf{x}) \\
&\quad = \frac{\partial}{\partial x_{pj}} \left(\|\mathbf{x}_j\|^{-1} \frac{\partial}{\partial x_{qj}} f(\bar{\mathbf{x}}) - \|\mathbf{x}_j\|^{-3} \sum_{l=1}^{d+1} \frac{\partial}{\partial x_{lj}} f(\bar{\mathbf{x}}) x_{lj} x_{qj} \right) \\
&\quad = \left(\frac{\partial}{\partial x_{pj}} \|\mathbf{x}_j\|^{-1} \right) \frac{\partial}{\partial x_{qj}} f(\bar{\mathbf{x}}) + \|\mathbf{x}_j\|^{-1} \frac{\partial}{\partial x_{pj}} \left(\frac{\partial}{\partial x_{qj}} f(\bar{\mathbf{x}}) \right) \\
&\qquad - \left(\frac{\partial}{\partial x_{pj}} \|\mathbf{x}_j\|^{-3} \right) \sum_{l=1}^{d+1} \frac{\partial}{\partial x_{lj}} f(\bar{\mathbf{x}}) x_{lj} x_{qj} \\
&\qquad - \|\mathbf{x}_j\|^{-3} \sum_{l=1}^{d+1} \left[\frac{\partial}{\partial x_{pj}} \left(\frac{\partial}{\partial x_{lj}} f(\bar{\mathbf{x}}) \right) x_{lj} x_{qj} + \frac{\partial}{\partial x_{lj}} f(\bar{\mathbf{x}}) \frac{\partial}{\partial x_{pj}} (x_{lj} x_{qj}) \right] \\
&\quad = \|\mathbf{x}_j\|^{-3} \Bigg\{ - x_{pj} \frac{\partial}{\partial x_{qj}} f(\bar{\mathbf{x}}) - x_{qj} \frac{\partial}{\partial x_{pj}} f(\bar{\mathbf{x}}) \\
&\qquad + \Big(3\|\mathbf{x}_j\|^{-2} x_{pj} x_{qj} - \delta_{pq} \Big) \sum_{l=1}^{d+1} x_{lj} \frac{\partial}{\partial x_{lj}} f(\bar{\mathbf{x}}) + \|\mathbf{x}_j\| \frac{\partial^2}{\partial x_{pj}\partial x_{qj}} f(\bar{\mathbf{x}}) \\
&\qquad - \|\mathbf{x}_j\|^{-1} \left(x_{pj} \sum_{s=1}^{d+1} x_{sj} \frac{\partial^2}{\partial x_{sj}\partial x_{qj}} f(\bar{\mathbf{x}}) + x_{qj} \sum_{l=1}^{d+1} x_{lj} \frac{\partial^2}{\partial x_{pj}\partial x_{lj}} f(\bar{\mathbf{x}}) \right) \\
&\qquad + x_{pj} x_{qj} \sum_{l=1}^{d+1} \sum_{s=1}^{d+1} x_{sj} x_{lj} \frac{\partial^2}{\partial x_{sj}\partial x_{lj}} f(\bar{\mathbf{x}}) \Bigg\} \\
&\quad = \|\mathbf{x}_j\|^{-3} \Big\{ - \mathbf{e}_p'\mathbf{x}_j \boldsymbol{\nabla} f(\bar{\mathbf{x}})'\mathbf{e}_q - \mathbf{e}_p' \boldsymbol{\nabla} f(\bar{\mathbf{x}}) \mathbf{x}_j' \mathbf{e}_q \\
&\qquad + \mathbf{e}_p' \big(3\|\mathbf{x}_j\|^{-2} \mathbf{x}_j \mathbf{x}_j' - \mathbf{I}_{d+1} \big) \mathbf{e}_q \mathbf{x}_j' \nabla f(\bar{\mathbf{x}}) + \|\mathbf{x}_j\| \mathbf{e}_p' \mathcal{H}_{jj} f(\bar{\mathbf{x}}) \mathbf{e}_q
\end{aligned}$$

$$
\begin{aligned}
&- \|\mathbf{x}_j\|^{-1}\big(\mathbf{e}_p'\mathbf{x}_j\mathbf{x}_j'\mathcal{H}_{jj}f(\bar{\mathbf{x}})\mathbf{e}_q + \mathbf{e}_p'\mathcal{H}_{jj}f(\bar{\mathbf{x}})\mathbf{x}_j\mathbf{x}_j'\mathbf{e}_q\big)\\
&+ \mathbf{e}_p'\mathbf{x}_j\mathbf{x}_j'\mathbf{e}_q\mathbf{x}_j'\mathcal{H}_{jj}f(\bar{\mathbf{x}})\mathbf{x}_j\Big\},
\end{aligned} \tag{4.20}
$$

with $p, q = 1, \ldots, d+1$. Collecting the entries in (4.20) into $\mathcal{H}_{jj}\bar{f}(\mathbf{x})$, it follows that, for $\mathbf{x} \in (\mathbb{S}^d)^r$,

$$
\begin{aligned}
\mathcal{H}_{jj}\bar{f}(\mathbf{x}) =& - \mathbf{x}_j\nabla_j f(\mathbf{x}) - \nabla_j f(\mathbf{x})'\mathbf{x}_j' + (3\mathbf{x}_j\mathbf{x}_j' - \mathbf{I}_{d+1})(\nabla_j f(\mathbf{x})\mathbf{x}_j)\\
&+ \mathcal{H}_{jj}f(\mathbf{x}) - (\mathbf{x}_j\mathbf{x}_j'\mathcal{H}_{jj}f(\mathbf{x}) + \mathcal{H}_{jj}f(\mathbf{x})\mathbf{x}_j\mathbf{x}_j') + \mathbf{x}_j\mathbf{x}_j'(\mathbf{x}_j'\mathcal{H}_{jj}f(\mathbf{x})\mathbf{x}_j)\\
=& (\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j')\mathcal{H}_{jj}f(\mathbf{x})(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j')\\
&- (\nabla_j f(\mathbf{x})\mathbf{x}_j)(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j') - \mathbf{A},
\end{aligned} \tag{4.21}
$$

where $\mathbf{A} := \big[\mathbf{x}_j\nabla_j f(\mathbf{x}) + (\mathbf{x}_j\nabla_j f(\mathbf{x}))' - 2(\nabla_j f(\mathbf{x})\mathbf{x}_j)\mathbf{x}_j\mathbf{x}_j'\big]$ is a symmetric matrix that, differently from the other terms in (4.21), is non-orthogonal to $\mathbf{x}_j\mathbf{x}_j'$:

$$
\mathbf{A}(\mathbf{x}_j\mathbf{x}_j') = (\mathbf{x}_j\nabla_j f(\mathbf{x}))' - (\nabla_j f(\mathbf{x})\mathbf{x}_j)\mathbf{x}_j\mathbf{x}_j',
$$

despite being easy to check that $(\mathbf{x}_j\mathbf{x}_j')\mathbf{A}(\mathbf{x}_j\mathbf{x}_j') = (\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j')\mathbf{A}(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j')$ $= \mathbf{0}$.

In addition, for $k, j = 1, \ldots, r$, $k \neq j$, and $p, q = 1, \ldots, d+1$,

$$
\begin{aligned}
&\frac{\partial^2}{\partial x_{pk}\partial x_{qj}}\bar{f}(\mathbf{x})\\
&\quad= \|\mathbf{x}_j\|^{-1}\frac{\partial}{\partial x_{pk}}\left(\frac{\partial}{\partial x_{qj}}f(\bar{\mathbf{x}})\right) - \|\mathbf{x}_j\|^{-3}\sum_{l=1}^{d+1}\left[\frac{\partial}{\partial x_{pk}}\left(\frac{\partial}{\partial x_{lj}}f(\bar{\mathbf{x}})\right)x_{lj}x_{qj}\right]\\
&\quad= \|\mathbf{x}_j\|^{-1}\|\mathbf{x}_k\|^{-1}\Bigg\{\frac{\partial^2}{\partial x_{pk}\partial x_{qj}}f(\bar{\mathbf{x}})\\
&\qquad- \|\mathbf{x}_k\|^{-2}x_{pk}\sum_{s=1}^{d+1}\frac{\partial^2}{\partial x_{sk}\partial x_{qj}}f(\bar{\mathbf{x}})x_{sk} - \|\mathbf{x}_j\|^{-2}x_{qj}\sum_{l=1}^{d+1}\frac{\partial^2}{\partial x_{pk}\partial x_{lj}}f(\bar{\mathbf{x}})x_{lj}\\
&\qquad- \|\mathbf{x}_j\|^{-2}\|\mathbf{x}_k\|^{-2}x_{pk}x_{qj}\sum_{l=1}^{d+1}\sum_{s=1}^{d+1}x_{sk}x_{lj}\frac{\partial^2}{\partial x_{sk}\partial x_{lj}}f(\bar{\mathbf{x}})\Bigg\}\\
&\quad= \|\mathbf{x}_j\|^{-1}\|\mathbf{x}_k\|^{-1}\Big\{\mathbf{e}_p'\mathcal{H}_{kj}f(\bar{\mathbf{x}})\mathbf{e}_q\\
&\qquad- \|\mathbf{x}_k\|^{-2}\mathbf{e}_p'\mathbf{x}_k\mathbf{x}_k'\mathcal{H}_{kj}f(\bar{\mathbf{x}})\mathbf{e}_q - \|\mathbf{x}_j\|^{-2}\mathbf{e}_p'\mathcal{H}_{kj}f(\bar{\mathbf{x}})\mathbf{x}_j\mathbf{x}_j'\mathbf{e}_q\\
&\qquad+ \|\mathbf{x}_j\|^{-2}\|\mathbf{x}_k\|^{-2}\mathbf{e}_p'\mathbf{x}_j\mathbf{x}_k'\mathbf{e}_q\mathbf{x}_k'\mathcal{H}_{kj}f(\bar{\mathbf{x}})\mathbf{x}_j\Big\}.
\end{aligned}
$$

By an analogous collection of terms to that in (4.21), for $\mathbf{x} \in (\mathbb{S}^d)^r$, $\mathcal{H}_{kj}\bar{f}(\mathbf{x}) = (\mathbf{I}_{d+1} - \mathbf{x}_k\mathbf{x}_k')\mathcal{H}_{kj} f(\mathbf{x})(\mathbf{I}_{d+1} - \mathbf{x}_j\mathbf{x}_j')$. □

Proof (Proposition 2) The proof follows after recalling that the unprojected estimator $\tilde{m}(t; h) := \sum_{j=1}^{n} W_j(t; h)\mathbf{X}_i$ satisfies $\tilde{m}_{-i}(t; h) = \sum_{j=1,\, j\neq i}^{n} W_{-i,j}(t; h)\mathbf{X}_j$ with $W_{-i,j}(t; h) = W_j(t; h)/(1 - W_i(t; h))$ since $\sum_{i=1}^{n} W_i(t; h) = 1$, for all $t \in \mathbb{R}$. □

References

1. Csernansky J.G., Wang L., Swank J., Miller J.P., Gado M., Mckeel D., Miller M.I., Morris J.C.: Preclinical detection of Alzheimer's disease: hippocampal shape and volume predict dementia onset in the elderly. Neuroimage **25**(3), 783 (2005) doi: https://doi.org/j.neuroimage.2004.12.036
2. Dryden I.L., Mardia K.V.: Statistical Shape Analysis, with Applications in R. Wiley Series in Probability and Statistics. Wiley, Chichester (2016) doi: https://doi.org/10.1002/9781119072492
3. Liu Z., Hong J., Vicory J., Damon J.N., Pizer S.M.: Fitting unbranching skeletal structures to objects. Med. Image Anal. **70**, 102020 (2021) doi: https://doi.org/10.1016/j.media.2021.102020
4. Pizer S.M., Marron J.S.: In Zheng, G., Li, S., Székely, G. (eds.) Statistical Shape and Deformation Analysis, pp. 137–164. Academic Press, London (2017) doi: https://doi.org/10.1016/B978-0-12-810493-4.00007-9
5. Pizer S.M., Jung S., Goswami D., Vicory J., Zhao X., Chaudhuri R., Damon J.N., Huckemann S., Marron J.: Nested sphere statistics of skeletal models. In Breuß, M., Bruckstein, A., Maragos, P. (eds) Innovations for Shape Analysis, pp. 93–115. Springer, Heidelberg (2013) doi: https://doi.org/10.1007/978-3-642-34141-0_5
6. Marron J.S., Dryden I.L.: Object Oriented Data Analysis, Monographs on Statistics and Applied Probability, vol. 169. CRC Press, Boca Raton (2021) doi: https://doi.org/10.1201/9781351189675
7. Hong J., Vicory J., Schulz J., Styner M., Marron J.S., Pizer S.M.: Non-Euclidean classification of medically imaged objects via s-reps. Med. Image Anal. **31**, 37 (2016) doi: https://doi.org/10.1016/j.media.2016.01.007
8. Schulz J., Pizer S.M., Marron J., Godtliebsen F.: Non-linear hypothesis testing of geometric object properties of shapes applied to hippocampi. J. Math. Imaging Vis. **54**(1), 15 (2016) doi: https://doi.org/10.1007/s10851-015-0587-7
9. Pizer S.M., Hong J., Vicory J., Liu Z., Marron J.S., Choi H.Y., Damon J., Jung S., Paniagua B., Schulz J., Sharma A., Tu L., Wang J.: Object shape representation via skeletal models (s-reps) and statistical analysis. In Pennec, X., Sommer, S., Fletcher, T. (eds.) Riemannian Geometric Statistics in Medical Image Analysis, pp. 233–271. Elsevier (2020) doi: https://doi.org/10.1016/B978-0-12-814725-2.00014-5
10. Siddiqi K., Pizer S.: Medial Representations: Mathematics, Algorithms and Applications, Computational Imaging and Vision, vol. 37. Springer Science & Business Media (2008) doi: https://doi.org/10.1007/978-1-4020-8658-8
11. Pizer S., Marron J., Damon J.N., Vicory J., Krishna A., Liu Z., Taheri M.: Skeletons, object shape, statistics. Front. Comput. Sci. **4**, 842637 (2022) doi: https://doi.org/10.3389/fcomp.2022.842637
12. Pizer S.M., Hong J., Vicory J., Liu Z., Marron J.S., Choi H.Y., Damon J., Jung S., Paniagua B., Schulz J., Sharma A., Tu L., Wang J.: Object shape representation via skeletal models (s-reps) and statistical analysis. In Pennec, X., Sommer, S., Fletcher, T. (eds.) Riemannian

Geometric Statistics in Medical Image Analysis, pp. 233–271. Academic Press, London (2020) doi: https://doi.org/10.1016/B978-0-12-814725-2.00014-5
13. Jung S., Foskey M., Marron J.S.: Principal arc analysis on direct product manifolds. Ann. Appl. Stat. **5**(1), 578 (2011) doi: https://doi.org/10.1214/10-aoas370
14. Jung S., Dryden I.L., Marron J.S.: Analysis of principal nested spheres. Biometrika **99**(3), 551 (2012) doi: https://doi.org/10.1093/biomet/ass022
15. Genovese C.R., Perone-Pacifico M., Verdinelli I., Wasserman L.: Nonparametric ridge estimation. Ann. Stat. **42**(4), 1511 (2014) doi: https://doi.org/10.1214/14-AOS1218
16. Eltzner B.: Geometrical smeariness – a new phenomenon of Fréchet means. Bernoulli **28**(1), 239 (2022) doi: https://doi.org/10.3150/21-BEJ1340
17. Friedman J., Hastie T., Tibshirani R.: Regularization paths for generalized linear models via coordinate descent. J. Stat. Softw. **33**(1), 1 (2010) doi: https://doi.org/10.18637/jss.v033.i01
18. Chacón J.E., Duong T.: Multivariate Kernel Smoothing and its Applications, Monographs on Statistics and Applied Probability, vol. 160. CRC Press, Boca Raton (2018) doi: https://doi.org/10.1201/9780429485572
19. Lehoucq R.B., Sorensen D.C., Yang C.: ARPACK Users' Guide. Society for Industrial and Applied Mathematics. Philadelphia (1998) doi: https://doi.org/10.1137/1.9780898719628
20. Sanderson C., Curtin R.: A user-friendly hybrid sparse matrix class in C++. In Davenport, J.H., Kauers, M., Labahn, G., Urban, J. (eds.) Mathematical Software – ICMS, pp. 422–430. Springer International Publishing, Cham (2018) doi: https://doi.org/10.1007/978-3-319-96418-8_50
21. García-Portugués E.: Exact risk improvement of bandwidth selectors for kernel density estimation with directional data. Electron. J. Stat. **7**, 1655 (2013) doi: https://doi.org/10.1214/13-ejs821
22. Silverman B.W.: Density Estimation for Statistics and Data Analysis. Monographs on Statistics and Applied Probability. Chapman & Hall, London (1986) doi: https://doi.org/10.1007/978-1-4899-3324-9
23. Zoubouloglou P., García-Portugués E., Marron J.S.: Scaled torus principal component analysis. J. Comput. Graph. Stat. (to appear) (2022) doi: https://doi.org/10.1080/10618600.2022.2119985
24. Borg I., Groenen P.J.: Modern Multidimensional Scaling: Theory and Applications. Springer Series in Statistics. Springer Science & Business Media (2005) doi: https://doi.org/10.1007/0-387-28981-X
25. de Leeuw J., Mair P.: Multidimensional scaling using majorization: SMACOF in R. J. Stat. Softw. **31**(3), 1 (2009) doi: https://doi.org/10.18637/jss.v031.i03
26. García-Portugués E., Paindaveine D., Verdebout T.: On optimal tests for rotational symmetry against new classes of hyperspherical distributions. J. Am. Stat. Assoc. **115**(532), 1873 (2020) doi: https://doi.org/10.1080/01621459.2019.1665527
27. García-Portugués E., Paindaveine D., Verdebout T.: rotasym: Tests for Rotational Symmetry on the Hypersphere (2022). https://CRAN.R-project.org/package=rotasym. R package version 1.1.4
28. Benjamini Y., Yekutieli D.: The control of the false discovery rate in multiple testing under dependency. Ann. Stat. **29**(4), 1165 (2001) doi: https://doi.org/10.1214/aos/1013699998
29. Lafarge T., Pateiro-López B.: Implementation of the 3D Alpha-Shape for the Reconstruction of 3D Sets from a Point Cloud (2020). https://CRAN.R-project.org/package=alphashape3d. R package version 1.3.1