Statistics and Computing (2023) 33:140
https://doi.org/10.1007/s11222-023-10305-4

ORIGINAL PAPER

Density estimation for toroidal data using semiparametric mixtures

Danli Xu[1] · Yong Wang[1]

Received: 15 August 2022 / Accepted: 26 September 2023 / Published online: 16 October 2023

Abstract

Toroidal data is an extension of circular data on a torus and plays a critical part in various scientific fields. This article studies the density estimation of multivariate toroidal data based on semiparametric mixtures. One of the major challenges of semiparametric mixture modelling in a multi-dimensional space is that one can not directly maximize the likelihood over the unrestricted component density as it will result in a degenerate estimate with an unbounded likelihood. To overcome this problem, we propose to fix the maximum of the component density, which subsequently bounds the maximum of the mixture and its likelihood function, hence providing a satisfactory density estimate. The product of univariate circular distributions are utilized to form multivariate toroidal densities as candidates for mixture components. Numerical studies show that the mixture-based density estimator is superior in general to the kernel density estimator.

Keywords Toroidal data · Density estimation · Semiparametric mixture · Bandwidth selection

1 Introduction

A circular observation can be viewed as a point lying on the circumference of a unit circle and is usually represented by an angle $x \in [0, 2\pi)$. It differs from a linear observation in its periodicity, i.e., $x + 2r\pi$ for $r \in \mathbb{Z}$ represents the same point x. It requires special techniques to analyze circular data, because they have a bounded range and lie on a Riemannian manifold. Circular observations can be extended to multi-dimensional to be on the surface of a unit (hyper-)sphere or a unit torus. The d-dimensional spherical observations lie on the unit d-sphere, e.g., the astronomical objects can be treated as points on the unit 2-sphere. Each d-dimensional spherical observation can be represented by d angles in the space $[0, \pi)^{d-1} \times [0, 2\pi)$. By contrast, a d-dimensional toroidal observation, which will be the focus of the study in this paper, corresponds to a point on a d-torus which is the product space of d unit circles, i.e., $\mathbb{T}^d = [0, 2\pi)^d$. An example of 2-dimensional toroidal data is the protein backbone chains, where the two dihedral angles connecting atoms essentially determine the shape of protein backbone structure and can be analyzed as toroidal data (Sittel et al. 2017). There are other types of toroidal data in higher dimensions including the nuclear magnetic resonance (NMR) and ribonucleic acid (RNA) data studied in bioinformatics. It is also a common practice to combine several circular variables into multivariate circular data and treat them jointly as a set of toroidal data, e.g., to study pairs of wind directions measured at different time points (Johnson and Wehrly 1977) or the relationship between the orientation of bird nests and the directions of creek flows (Fisher 1995).

In this paper, we study nonparametric density estimation for toroidal data ($d \geq 2$) owing to their many important applications. In particular, we propose to use semiparametric mixtures with component distributions suitable for toroidal data. As will be shown later, using semiparametric mixtures offers more flexibility than the more popular kernel density estimators (KDE), especially in a multi-dimensional space, and tends to produce simpler models yet with typically better numerical performance. For mixture components, we employ the product of circular distributions, which may or may not belong to the same family, hence at a higher level of methodological generality. To properly use semiparametric mixtures, however, there are a couple of difficult challenges. One is that a direct maximization of the likelihood function will result in an unusable degenerate estimate of the mixing distribution. To overcome this, we consider using a scalar variable and defining it as the “bandwidth”, as similarly used

✉ Yong Wang
yongwang@auckland.ac.nz

Danli Xu
dxu452@aucklanduni.ac.nz

[1] Department of Statistics, The University of Auckland, Private Bag 92019, Auckland 1142, New Zealand

in KDE, that controls the smoothness of a density estimate. This breaks down the maximization problem into two subproblems: maximizing the likelihood function for each fixed value of the bandwidth and finding an appropriate bandwidth value via a model selection method, either deterministically or based on simulation. The other is how to define this bandwidth variable, which is particularly relevant to toroidal data here which have a bounded range. We will solve this problem by controlling the maximum of mixture component densities and using its reciprocal as the bandwidth variable. This very well helps bound the likelihood function and produce appropriate density estimates.

Throughout the paper, a lowercase boldface letter is used to denote a vector, e.g., $\boldsymbol{x}$, $\boldsymbol{x}_i$ and $\boldsymbol{\mu}$, and an uppercase boldface letter a matrix, e.g., $\boldsymbol{M}$ and $\boldsymbol{\Sigma}$. Symbol $\boldsymbol{\beta}$ will be used to designate generically the structural parameter of a semiparametric mixture, which can be a vector or matrix, depending on the situation.

The remainder of the paper is organized as follows. Section 2 describes how to use semiparametric mixtures for density estimation with toroidal data and proposes to control the maximum of component densities. In Sect. 3, the algorithm is described in detail. Some theoretical results regarding the product von Mises component distribution is provided in Sect. 4. Numerical studies including simulation and three real-world data analysis are presented in Sects. 5 and 6, respectively. Some concluding remarks are given in the final section.

2 Using semiparametric mixtures

In this section, we describe how to use semiparametric mixtures for density estimation for data on a torus. We first describe semiparametric mixtures in Sect. 2.1 and then the mixture component densities constructed by products of univariate densities in Sect. 2.2. The central problem of defining the bandwidth is studied in Sect. 2.3. Finally, the choice of the smoothing parameter is discussed in details in Sect. 2.4.

2.1 Semiparametric mixtures

The density of a semiparametric mixture that we use for toroidal data is of form

$$f(\boldsymbol{x}; G, \boldsymbol{\beta}) = \int f(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\beta})\, \mathrm{d}G(\boldsymbol{\mu}), \tag{1}$$

where $f(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\beta})$, $\boldsymbol{x}, \boldsymbol{\mu} \in \mathbb{T}^d$, is a component density, $\boldsymbol{\beta}$ a finite-dimensional parameter that is common to all mixture components, and G a mixing distribution that takes a completely unspecified form. For any fixed $\boldsymbol{\beta}$, the semiparametric mixture reduces to a nonparametric mixture which has only the infinite-dimensional parameter G. It is known that the nonparametric maximum likelihood estimation (NPMLE) $\widehat{G}$ of G must have a discrete solution, which has no more support points than the number of distinct observations in the sample (Laird 1978; Lindsay 1983a). This discrete NPMLE is typically the unique one, e.g., for the exponential family (Lindsay 1983b). It is therefore that one can only consider discrete distributions for G for maximum likelihood estimation. Suppose such a discrete G has m support points $\boldsymbol{M} = (\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_m)$ with probability masses $\boldsymbol{\omega} = (\omega_1, \dots, \omega_m)^\top$, for $\omega_1, \dots, \omega_m > 0$ and $\sum_{j=1}^m \omega_j = 1$. Then mixture (1) can be rewritten as a finite mixture with m components:

$$f(\boldsymbol{x}; G, \boldsymbol{\beta}) = \sum_{j=1}^{m} \omega_j f(\boldsymbol{x}; \boldsymbol{\mu}_j, \boldsymbol{\beta}), \tag{2}$$

which, given a random sample $\boldsymbol{x}_1$, ..., $\boldsymbol{x}_n \in \mathbb{T}^d$, has the log-likelihood function

$$l(G, \boldsymbol{\beta}) = \sum_{i=1}^{n} \log[f(\boldsymbol{x}_i; G, \boldsymbol{\beta})]. \tag{3}$$

Note that G and $(\boldsymbol{\omega}, \boldsymbol{M})$ are interchangeable and that the number of components m is also to be estimated from the data.

For a fixed value of $\boldsymbol{\beta}$, the gradient function is given by

$$d(\boldsymbol{\mu}; G, \boldsymbol{\beta}) = \left.\frac{\partial l((1-\epsilon)G + \epsilon\delta_{\boldsymbol{\mu}}, \boldsymbol{\beta})}{\partial \epsilon}\right|_{\epsilon=0+}$$
$$= \sum_{i=1}^{n} \frac{f(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\beta})}{f(\boldsymbol{x}_i; G, \boldsymbol{\beta})} - n, \tag{4}$$

where $\delta_{\boldsymbol{\mu}}$ denotes the point-mass distribution function at $\boldsymbol{\mu}$. This function characterizes the NPMLE $\widehat{G}$, in the sense that an estimate G is the NPMLE if and only if $\sup_{\boldsymbol{\mu}} d(\boldsymbol{\mu}; G, \boldsymbol{\beta}) = 0$ (Lindsay 1995). It is also highly instrumental for computing $\widehat{G}$ (Wang 2007), which will be detailed below.

In general, however, leaving $\boldsymbol{\beta}$ with full degrees of freedom and maximizing the likelihood function directly will result in useless degenerate mixture components, with the likelihood approaching infinity (Grenander 1981; Geman and Hwang 1982). Wang and Wang (2015) addressed this issue in the Euclidean space with Gaussian components, where $\boldsymbol{\beta}$ is the component covariance matrix $\boldsymbol{\Sigma}$. They suggested to decompose $\boldsymbol{\Sigma}$ into a product of h^2 and a positive-definite matrix $\boldsymbol{B}$ subject to $|\boldsymbol{B}| = 1$, which thus makes the likelihood bounded for any $h > 0$. This interesting technique, however, is not directly applicable to the toroidal data situation here, as will be explained and improved upon in Sect. 2.3.

2.2 Product component densities

To use the semiparametric mixture (1), we need suitable mixture components for toroidal data. The component density here plays a similar role to the kernel function in kernel density estimation. In this paper, we choose to construct component densities by the product of univariate circular densities, i.e., a component density is given by

$$f(\boldsymbol{x};\boldsymbol{\mu},\boldsymbol{\beta}) = \prod_{p=1}^{d} f(x_p;\mu_p,\beta_p),$$

where $f(x_p;\mu_p,\beta_p)$, with a location parameter μ_p and a scale parameter β_p, is a univariate density function for circular data. The spread of the joint distribution is thus determined by the vector of scale parameters for all dimensions. Virtually all univariate circular distributions can be used, and one may even consider using different families for different variables according to their types and ranges.

For our mixture-based density estimation, we implemented two such distributions: the von Mises (VM) and the wrapped normal (WN) distribution. The VM density is of form

$$f_{\mathrm{VM}}(x;\mu,\kappa) = \frac{e^{\kappa\cos(x-\mu)}}{2\pi I_0(\kappa)},$$

where $x,\mu \in \mathbb{T}, \kappa > 0$ and $I_0(\kappa)$ denotes the modified Bessel function of the first kind of order zero, for which

$$I_t(\kappa) = \frac{1}{2\pi}\int_0^{2\pi} \cos(tx)e^{\kappa\cos(x)}\,\mathrm{d}x,$$

for any integer t. Here μ is known as the location parameter and κ the concentration parameter, an opposite to a scale parameter.

The WN density is given by

$$f_{\mathrm{WN}}(x;\mu,v) = \frac{1}{\sqrt{2\pi v}} \sum_{r=-\infty}^{\infty} \exp\left[-\frac{(x-\mu+2r\pi)^2}{2v}\right],$$

where $x,\mu \in \mathbb{T}, v > 0$. For evaluation, the infinite sum above can be easily replaced with a truncated series. With a sufficient number of terms, this is an excellent approximation, as the normal density decreases exponentially as $|r|$ increases. Clearly, substituting for $\boldsymbol{\beta}$, it is $\boldsymbol{\kappa} = (\kappa_1,\ldots,\kappa_m)^\top$ for the product von Mises (PVM) distribution and $\boldsymbol{v} = (v_1,\ldots,v_m)^\top$ for the product wrapped normal (PWN) distribution.

It is also possible to incorporate correlation between univariate variables, but the nonparametric form of G is able to deal with correlation and also account for a considerable amount of other types of complexity in the data. Using such a product distribution ensures that any marginal distribution can be easily produced, which can be very useful for understanding the results in a low-dimensional space and for gaining insights.

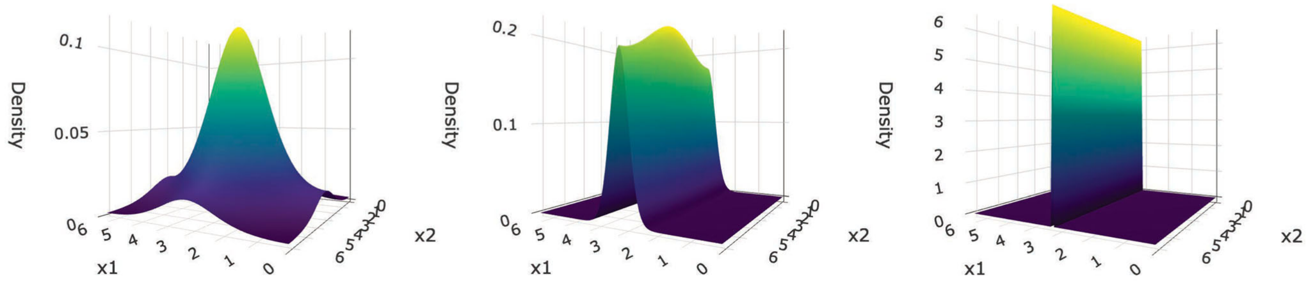
2.3 Controlling the component maximum

As explained in Sect. 2.1, it is not appropriate to maximize the log-likelihood function (3) without restricting $\boldsymbol{\beta}$. It is also desirable, as in Wang and Wang (2015), to restrict a scalar variable that is determined by $\boldsymbol{\beta}$ and can be considered as the smoothing parameter. For Gaussian mixtures, Wang and Wang (2015) use the decomposition $\boldsymbol{\beta} = \boldsymbol{\Sigma} = h^2\boldsymbol{B}$ with $|\boldsymbol{B}| = 1$ and treat h as the smoothing parameter. For a diagonal $\boldsymbol{\Sigma} = \mathrm{diag}\{\sigma_1^2,\ldots,\sigma_d^2\}$, as with a product Gaussian density, $h = \prod_{i=1}^d \sigma_i$, i.e., the product of the scale parameters of all dimensions.

However, this idea can not be directly applied to toroidal data. Take the bivariate PVM density as an example which has mean $(\pi,\pi)^\top$ and concentration $\boldsymbol{\beta} = \boldsymbol{\kappa} = (\kappa_1,\kappa_2)^\top$. By simply holding $\kappa_1\kappa_2$ constant, the density can still become degenerate, as illustrated in Fig. 1, which shows three bivariate densities, all satisfying $\kappa_1\kappa_2 = 1$. Shown in Figs. 1a and 1b, as $\boldsymbol{\kappa}$ varies from $(1,1)^\top$ to $(0.1,10)^\top$, the marginal density for x_1 becomes more concentrated around π, while that for x_2 approaches a uniform one. To further illustrate the effect, Fig. 1c shows the density with $\boldsymbol{\kappa} = (0.0001, 10000)^\top$. The marginal density for x_1 is almost degenerate, and that for x_2 is almost uniform on $[0,2\pi)$. This means that the joint density value around the mean can become arbitrarily large and that maximizing the likelihood function will lead to a mixture with an infinite likelihood value and degenerate component distributions. The resultant mixtures are useless as density estimates.

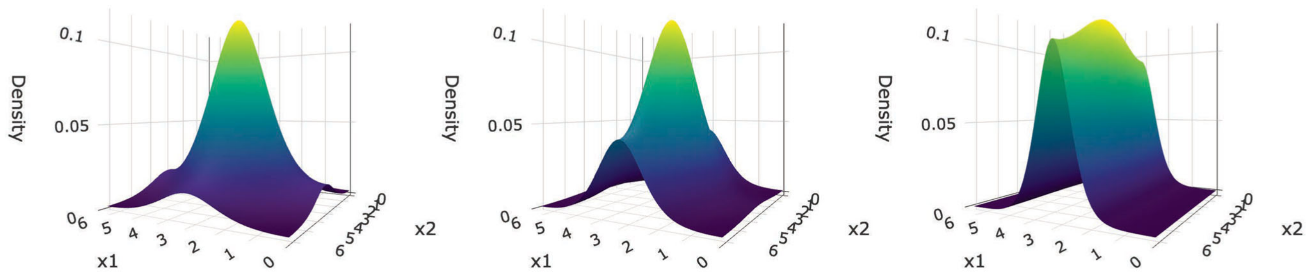
To overcome this challenge, we realize that bounding the likelihood can be achieved by bounding the component density function. For a unimodal mixture component, we propose fixing its maximum. Therefore, we define h, the bandwidth parameter that controls the smoothness of the density estimate, as the reciprocal of the maximum of a mixture component, up to a multiplicative constant. Then the log-likelihood (3) can be easily shown to be bounded by $-n\log(h)$, up to an additive constant. Now let us consider using the PVM and PWN distributions as mixture components. For the PVM components, we define

$$h = h(\boldsymbol{\kappa}) = \prod_{p=1}^{d} \frac{I_0(\kappa_p)}{e^{\kappa_p}}. \tag{5}$$



(a) $\boldsymbol{\kappa} = (1, 1)^\top$ (b) $\boldsymbol{\kappa} = (0.1, 10)^\top$ (c) $\boldsymbol{\kappa} = (0.0001, 10000)^\top$

Fig. 1 Three bivariate PVM densities, holding $\kappa_1\kappa_2 = 1$



(a) $\boldsymbol{\kappa} = (1, 1)^\top$ (b) $\boldsymbol{\kappa} = (0.50, 1.74)^\top$ (c) $\boldsymbol{\kappa} = (0.10, 3.09)^\top$

Fig. 2 Three bivariate PVM densities, holding $h = 8.56$, as defined in (5)

Figure 2 shows three bivariate product von Mises densities, all with $h = 8.56$. It is clear that for a fixed h-value, the maximum of the PVM density remains constant, while the shape is allowed to vary with different $\boldsymbol{\kappa}$-values.

To better determine the maximum of a PWN density, we consider an alternative representation of the univariate WN density that is given by

$$
\begin{aligned}
& f_{\mathrm{WN}}(x; \mu, \rho) \\
& = \frac{1}{2\pi}\left[1 + 2\sum_{k=1}^{\infty} \rho^{k^2} \cos[k(x-\mu)]\right], \\
& = \frac{1}{2\pi}\vartheta_3(x-\mu, \rho), \quad 0 < \rho < 1,
\end{aligned}
$$

where $\rho = e^{-\frac{v}{2}}$ and ϑ_3 is a Jacobi theta function (Abramowitz and Stegun 1964, page 576). In this parametrization, the smoothness of the density is controlled by ρ. The density approaches the circular uniform one as $\rho \to 0$, and a degenerate point mass as $\rho \to 1$. The density maximum is attained at μ. Hence for the PWN components with $\boldsymbol{\rho} = (\rho_1, \ldots, \rho_d)^\top$, we define

$$h = h(\boldsymbol{\rho}) = \prod_{p=1}^{d} \frac{1}{\vartheta_3(0, \rho_p)}. \tag{6}$$

For computation, an infinite sum can be readily replaced with a finite sum as the terms in the series decay exponentially fast to zero as k^2 increases.

2.4 Smoothing parameter selection

Our density estimation problem is now reformulated into two subproblems. The first is to maximize $l(G, \boldsymbol{\beta})$ subject to a constant h, which gives estimates $\widehat{G}_h$ and $\widehat{\boldsymbol{\beta}}_h$. The second is to select an appropriate h-value to determine the smoothness of the final density estimate. We describe the computational algorithm for solving the first subproblem in Sect. 3 and discuss the second subproblem in this section.

We need to choose a suitable h-value and thus determine its corresponding $(\widehat{G}_h, \widehat{\boldsymbol{\beta}}_h)$ or the density estimate. Specifically, a smaller h-value tends to produce a more wiggly density estimate, whereas a larger one will result in a relatively over-smoothed mixture density. There is no doubt that the smoothing parameter selection is a major challenge for nonparametric density estimators, and it may significantly affect the performance of an estimator. To address this important yet difficult problem, we treat it as a model selection procedure. We can either resort to either a deterministic criterion or a simulation-based technique. Since $h > 0$ is real-valued, we can consider only a discrete subset of h-values.

Among the various information-theoretic model selection criteria, we are inclined to utilize the ones that are likelihood-based as our method itself relies on the maximum likelihood estimation. The Akaike information criterion (AIC) appears to be more reliable than the Bayesian information criterion (BIC) for mixture-based density estimators (Wang and Chee 2012). The formula of AIC is given by

$$\mathrm{AIC}(h) = -2\tilde{l}(h) + 2p,$$

where $\tilde{l}(h) = \max_{G,\boldsymbol{\beta}} l(G, \boldsymbol{\beta})$ subject to the fixed h-value, is the profile log-likelihood function of the smoothing parameter h, and p the number of free parameters in the model. For data on a d-dimensional torus, a $\widehat{G}$ with m components and a d-dimensional scale parameter vector, we have $p = (m+1)(d+1) - 3$.

Obviously, the number of free parameters increases with both m and d. Since in nonparametric modelling the number of components is unrestricted, the number of parameters p may be close to or even exceed the number of observations n and more likely so in the multivariate scenario. To avoid such an over-fitted model with an unlimited profile likelihood, an improved version of AIC is adopted here. It is called $\mathrm{AIC_c}$ with formula (Cavanaugh 1997)

$$\mathrm{AIC_c}(h) = \mathrm{AIC} + \frac{2p(p+1)}{(n-p-1)_+},$$

where $(n-p-1)_+ = \max\{n-p-1, 0\}$. With the extra penalty term, $\mathrm{AIC_c}$ is less likely to pick an over-fitted mixture model, especially when n is not sufficiently larger than p. With a simpler model, it is also more interpretable and computationally efficient. We note that, as pointed out by Lindsay (1995), the asymptotic normality theory of maximum likelihood fails for mixture models. In addition, we should also be careful for generating the bandwidth sequence as the profile likelihood will increase monotonically to infinity as h approaches zero. In practice, it is highly unlikely to fit such a small h-value for the final mixture model unless the underlying data structure is extremely concentrated around a few observations. Otherwise, $\mathrm{AIC_c}$ has been proved to perform reasonably well in mixture modelling in both the univariate and the multivariate Euclidean space, as reported in Wang and Chee (2012) and Wang and Wang (2015), respectively.

One may consider simulation-based methods such as cross-validation and bootstrapping. They tend to generate more reliable estimates but at the same time are more computationally demanding. In the following numerical studies, we will only use the cost-effective $\mathrm{AIC_c}$ criterion in simulation studies, whereas for real-world data, the cross-validation approach will also be included to obtain potentially better estimates. More details about its use will be given in Sect. 6.1.

3 Computation

3.1 The algorithm

To find $\widehat{G}_h$ and $\widehat{\boldsymbol{\beta}}_h$ for h fixed, our algorithm can be described as follows:

1. Choose initial estimates: a discrete G_0 and a $\boldsymbol{\beta}_0$ with $h(\boldsymbol{\beta}_0)$ fixed and $l(G_0, \boldsymbol{\beta}_0) < \infty$. Set $s = 0$.
2. Update G_s to $G_{s+\frac{1}{2}}$: Use the constrained Newton method (CNM).
3. Update $(G_{s+\frac{1}{2}}, \boldsymbol{\beta}_s)$ to $(G_{s+1}, \boldsymbol{\beta}_{s+1})$: Use the EM algorithm modified for h fixed.
4. If $l(G_{s+1}, \boldsymbol{\beta}_{s+1}) - l(G_s, \boldsymbol{\beta}_s) \leq$ tolerance, stop. If otherwise, set $s = s + 1$ and repeat Steps 2-4.

We note that since the CNM and EM algorithms are only used here to solve subproblems, one does not have to run each algorithm in full iterations. We find it more efficient to run only 1 CNM iteration in Step 2 and 5 EM iterations in Step 3.

We describe Steps 2 and 3 in details in Sects. 3.2 and 3.3, respectively.

3.2 Update G_s

The algorithm used to update G_s in Step 2 is the CNM (Wang 2007; Wang and Wang 2015; Hu and Wang 2021). Each iteration of the algorithm consists mainly of two steps: finding new candidate support points and updating the mixing proportions of all support points.

Due to the properties of the gradient function (4), its local maxima are considered to be good candidate support points (Wang 2007). The maximization to find each of these local maxima in a multi-dimensional space can itself be computationally costly, and this is required for each iteration of the CNM. To resolve this issue, Wang and Wang (2015) and Hu and Wang (2021) proposed a strategy that uses a “random grid”, by turning the gradient function into a finite mixture density and drawing a random sample from it. To do this, one first removes the additive constant $-n$ and then turns the remaining sum into a finite mixture density of $\boldsymbol{\mu}$ (not $\boldsymbol{x}$) through normalizing the coefficients. Note that $f(\boldsymbol{x}_i; \boldsymbol{\mu}, \boldsymbol{\beta})$ may not be a density function for $\boldsymbol{\mu}$ and thus may need to be normalized as well. For both of the families that we use as mixture components, $\boldsymbol{\mu}$ and $\boldsymbol{x}$ are symmetric in their density functions and thus the density functions remain the same if they swap positions. Generating a sample from such a finite mixture is straightforward, and we find it often sufficient to use a sample size 20. The rationale behind this strategy is that more random points tend to be generated in the area with large gradient values, thus increasing the possibility of not missing out the areas with a local maximum, in particular with the global maximum. To locate more precisely the local maxima in the areas, we run 100 iterations of the Modal EM algorithm (Li et al. 2007), starting with both the randomly generated points and the current support points of G_s. To save computational cost, one does not have to use all of the resulting points but only the best one near each current support point. Hence, one may simply choose an arbitrary dimension and partition its range into disjoint intervals, each containing a current support point, e.g., using the midpoints between consecutive support points as break points, and then select the one (if there is at least one) with the largest gradient value in each interval. The selected points are the candidate support points that are to be added to the support set of G_s. Note that the dimension can be chosen arbitrarily, as the algorithm will converge to the NPMLE.

Adding new candidate support points with zero probability masses to G_s does not change G_s as a probability measure but only increases the dimension of $\boldsymbol{\mu}_s$ and $\boldsymbol{\omega}_s$. One then proceeds to update all of the mixing proportions to produce $G_{s+\frac{1}{2}}$. The updating of the mixing proportions makes use of the second-order Taylor approximation to the log-likelihood function with respect to $\boldsymbol{\omega}$ only, which is then optimized as solving a quadratic programming problem, in particular using the non-negativity least squares (NNLS) algorithm (Lawson and Hanson 1995; Wang 2007, 2010; Wang and Wang 2015). It should be followed with a line search to ensure a proper increase of the log-likelihood function and the eventual convergence of the algorithm. Any support point with a zero mass after the updating be redundant and can be immediately removed. This strategy allows for an exponentially fast expansion and reduction of the support set of G_s, which will eventually settle down to virtually the one of the NPMLE. We refer the reader to the above references for more details of the algorithm.

3.3 Update $\left(G_{s+\frac{1}{2}}, \boldsymbol{\beta}_s\right)$

To update $(G_{s+\frac{1}{2}}, \boldsymbol{\beta}_s)$ in Step 3, the EM algorithm (Dempster et al. 1977; McLachlan and Krishnan 1997; McLachlan and Peel 2000) is used, which is modified from its standard version for a finite mixture. In the following, we give the EM iterative formulae for updating the parameters of a finite mixture with either PVM or PWN components and how we modify the algorithm when h is fixed.

First, we note that the complete-data log-likelihood Q of a finite mixture with product components can be written as

$$\begin{aligned}
&Q(\boldsymbol{\omega}, \boldsymbol{M}, \boldsymbol{\beta}) \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij} \log[\omega_j f(\boldsymbol{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\beta})] \\
&= \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij} \log(\omega_j) \\
&\quad + \sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij} \sum_{p=1}^{d} \log[f(x_{ip}; \mu_{jp}, \beta_p)],
\end{aligned}$$

where

$$p_{ij} = \frac{\omega_j f(\boldsymbol{x}_i; \boldsymbol{\mu}_j, \boldsymbol{\beta})}{\sum_{l=1}^{m} \omega_l f(\boldsymbol{x}_i; \boldsymbol{\mu}_l, \boldsymbol{\beta})}.$$

The product von Mises mixtures

For a von Mises finite mixture model, we can easily derive the EM formulae as follows:

$$\omega'_j = \frac{1}{n}\sum_{i=1}^{n} p_{ij}, \tag{7}$$

$$\mu'_{jp} = \operatorname{atan2}\left(\sum_{i=1}^{n} p_{ij}\sin(x_{ip}), \sum_{i=1}^{n} p_{ij}\cos(x_{ip})\right), \tag{8}$$

$$\kappa'_p = A^{-1}(\bar{R}_p), \tag{9}$$

where

$$A(\kappa_p) = \frac{I_1(\kappa_p)}{I_0(\kappa_p)}. \tag{10}$$

Note that $\bar{R}_p = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}\cos(x_{ip} - \mu_{jp})$, which is shown in the Appendix.

When h is fixed, to maximize the complete-data log-likelihood the EM formulae (7) and (8) remain unchanged, but (9) is not appropriate any more. Since it is impossible to invert $I_0(\kappa)$ analytically to obtain a closed-form solution, we make use of the numerical optimization tool available in the R package "nloptr" (Johnson 2007). Among various algorithms, we chose to use the derivative-based local optimization algorithm called "conservative convex separable approximation with a quadratic penalty term", abbreviated as CCSAQ (Svanberg 2002). Despite the usage of a local searching algorithm, the solution must also be the unique, global maximum, as shown in Sect. 4.

The product wrapped normal mixtures

The PWN distribution essentially wraps a multivariate normal density with a diagonal covariance matrix onto a torus and is hence also unimodal around its mean $\boldsymbol{\mu}$. Define $\boldsymbol{x}_i^{(\boldsymbol{r})} = \boldsymbol{x}_i + 2\pi\boldsymbol{r}$ for $\boldsymbol{r} \in \mathbb{Z}^d$ as the ith toroidal observation in the $\boldsymbol{r}$th wrapping. Denoting by $\phi(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ the multivariate normal density with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, the EM formulae for a finite PWN mixture can be derived to be

$$\omega'_j = \frac{1}{n}\sum_{i=1}^{n} p_{ij},$$
$$\boldsymbol{\mu}'_j = \frac{\sum_{i=1}^{n}\sum_{\boldsymbol{r}\in\mathbb{Z}^d} p_{ij}^{(\boldsymbol{r})}\boldsymbol{x}_i^{(\boldsymbol{r})}}{\sum_{i=1}^{n}\sum_{\boldsymbol{r}\in\mathbb{Z}^d} p_{ij}^{(\boldsymbol{r})}},$$
$$v'_p = \sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{\boldsymbol{r}\in\mathbb{Z}^d} p_{ij}^{(\boldsymbol{r})}(x_{ip}^{(\boldsymbol{r})} - \mu_{jp})^2, \tag{11}$$

where

$$p_{ij}^{(\boldsymbol{r})} = \frac{\omega_j\phi_j(\boldsymbol{x}_i^{(\boldsymbol{r})}; \boldsymbol{\mu}_j, \boldsymbol{\Sigma})}{\sum_{l=1}^{m}\omega_l f(\boldsymbol{x}_i; \boldsymbol{\mu}_l, \boldsymbol{\Sigma})},$$
$$p_{ij} = \sum_{\boldsymbol{r}\in\mathbb{Z}^d} p_{ij}^{(\boldsymbol{r})},$$

and $f(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{\boldsymbol{r}\in\mathbb{Z}^d}\phi(\boldsymbol{x}^{(\boldsymbol{r})}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$, with $\boldsymbol{\Sigma} = \mathrm{diag}(\boldsymbol{v})$, the diagonal matrix comprising the elements of $\boldsymbol{v}$ along the diagonal.

Although truncation is definitely needed to avoid evaluating the above infinite sums, the computational cost for the PWN mixtures increases exponentially with dimension d. This makes the PVM mixtures a much better choice for computational purposes when d is large.

When h is fixed, the EM formula (11) has to be replaced. We again utilize the R package "nloptr" but with a derivative-free optimization algorithm "Constrained Optimization BY Linear Approximations" (COBYLA) to avoid the costly evaluation of its multi-dimensional derivatives (Powell 1994).

4 Uniqueness

As described in Sect. 3.3, the EM algorithm is applied to update the parameters of a finite mixture model. With a product von Mises component density, we implement a local searching algorithm to solve the constrained optimization problem. Here we show that there is only one local maximum, which is the unique global maximum.

It is known that the linear combination with positive coefficients of strictly concave functions is strictly concave. Hence, to show that $Q(\boldsymbol{\kappa})$ is concave, it suffices to show that

$$\log[f(\kappa_p)] = -\log(I_0(\kappa_p)) + \kappa_p\cos(x_{ip} - \mu_{jp})$$

is concave for any $\kappa_p > 0$. First, it is known that for all $\kappa \in \mathbb{R}$,

$$I_0(\kappa)' = I_1(\kappa), \quad I_1(\kappa)' = I_0(\kappa) - \frac{1}{\kappa}I_1(\kappa)$$

(Abramowitz and Stegun 1964, Section 9.6.28). Notice that

$$\frac{\partial\log(f)}{\partial\kappa_p} = \cos(x_{ip} - \mu_{jp}) - \frac{I_1(\kappa_p)}{I_0(\kappa_p)},$$
$$\frac{\partial^2\log(f)}{\partial\kappa_p^2} = \frac{I_1(\kappa_p)}{I_0(\kappa_p)}\left[\frac{I_1(\kappa_p)}{I_0(\kappa_p)} + \frac{1}{\kappa_p}\right] - 1.$$

By Segura (2022, Theorem 2) for all $\kappa > 0$,

$$\frac{I_0(\kappa)}{I_1(\kappa)} > \frac{\sqrt{\frac{3}{4} + \kappa^2} + \frac{1}{2}}{\kappa}.$$

Therefore,

$$\frac{I_0(\kappa)}{I_1(\kappa)} - \frac{1}{\kappa} > \frac{\sqrt{\frac{3}{4} + \kappa^2} - \frac{1}{2}}{\kappa} = \frac{\kappa + \frac{1}{2\kappa}}{\sqrt{\frac{3}{4} + \kappa^2} + \frac{1}{2}} > \frac{I_1(\kappa)}{I_0(\kappa)},$$

i.e.,

$$\frac{I_1(\kappa)}{I_0(\kappa)}\left[\frac{1}{\kappa} + \frac{I_1(\kappa)}{I_0(\kappa)}\right] < 1.$$

Hence, $\log[f(\kappa_p)]$ is strictly concave, and so is $Q(\boldsymbol{\kappa})$.

Since the maximum of a mixture component is restricted to be a constant, the constraint can be written as

$$g(\boldsymbol{\kappa}) = \sum_{p=1}^{d}\log\left[\frac{I_0(\kappa_p)}{e^{\kappa_p}}\right],$$

with

$$\frac{\partial g}{\partial\kappa_p} = \frac{I_1(\kappa_p)}{I_0(\kappa_p)} - 1,$$
$$\frac{\partial^2 g}{\partial\kappa_p^2} = \frac{[I_0(\kappa_p) + \frac{1}{\kappa_p}I_1(\kappa_p)] - I_1(\kappa_p)^2}{I_0(\kappa_p)^2} > 0.$$

That is, the constraint is a strictly convex function of $\boldsymbol{\kappa}$.

Therefore, it is a strictly concave function $Q(\boldsymbol{\kappa})$ that is to be maximized, subject to a convex constraint. As a result, there exists only one point in the optimal set, and hence a local maximum must also be the unique global maximum (Boyd and Vandenberghe 2004, page 152).

5 Simulation studies

In this section, we report the results of simulation studies that examine the performance of the proposed nonparametric mixture density estimators on a torus. After introducing the three loss functions used as performance metrics in Sect. 5.1, two candidates for the mixture components are compared using four simulation models in Sect. 5.2. In Sect. 5.3, the mixture-based estimator is compared with the kernel density estimator. All computations were carried out in R (R Core Team 2021).

5.1 Performance measures

For performance measures, we consider three loss functions, as used by Wang and Chee (2012). They are the integrated squared error (ISE), the Kullback-Leibler divergence (KL) and the Hellinger distance (HD):

$$\mathrm{ISE}(f, \hat{f}) = \int_{\mathbb{T}} \left[\hat{f}(\boldsymbol{x}) - f(\boldsymbol{x})\right]^2 \, \mathrm{d}\boldsymbol{x},$$

$$\mathrm{KL}(f, \hat{f}) = \int_{\mathbb{T}} f(\boldsymbol{x}) \log \left[\frac{f(\boldsymbol{x})}{\hat{f}(\boldsymbol{x})}\right] \mathrm{d}\boldsymbol{x},$$

$$\mathrm{HD}(f, \hat{f}) = \int_{\mathbb{T}} \left[\hat{f}(\boldsymbol{x})^{\frac{1}{2}} - f(\boldsymbol{x})^{\frac{1}{2}}\right]^2 \mathrm{d}\boldsymbol{x},$$

where $\hat{f}$ is an estimate of a true density function f. The ISE is the most popular loss measure, as was also used by Oliveira et al. (2012), García-Portugués (2013) and Di Marzio et al. (2011) for, respectively, circular, spherical and toroidal kernel density estimation. Aiming to lower the squared difference between two densities, it tends to perform better in high-density areas than in low-density areas. The KL is likelihood-based and computed as the expected log-ratio between the true and estimated densities. It thus may sacrifice estimation in areas with high density to improve the fit in areas with few observations. As for the HD, it can be viewed as a compromise between the other two.

To efficiently evaluate these integrals in a non-Euclidean, high-dimensional space, we use Monte Carlo integration, particularly the importance sampling technique. To use it, one may simply divide an integrand by the true density f, randomly generate sufficient data points according to f, and take the average of the transformed integrand evaluated at these data points. However, if $\hat{f}$ has heavier tails than f, which is not unusual in practice, then the loss will be significantly inflated around tails, resulting in high variation. To solve this problem, one should use a sampling distribution that has heavier tails than f and $\hat{f}$, while having a similar shape to both. We therefore used the true density function but with a higher degree of smoothness. Since the smoother the sampling density is, the larger sample size is required to ensure accurate loss calculation, an appropriate level of smoothness is needed to achieve a trade-off between efficiency and accuracy.

5.2 Comparison between mixture component families

To compare the two candidate families used for mixture components, four bivariate toroidal mixtures are considered which cover the situations with skewness, multi-modality and correlation. These distributions are listed in Table 1, with their contour plots shown in Fig. 3. The parametric wrapped normal and the mixture consisting of six von Mises densities with a skewed shape do not have obvious correlation between the two dimensions. The bimodal wrapped normal mixture and the trimodal von Mises mixture model should be more challenging for density estimation.

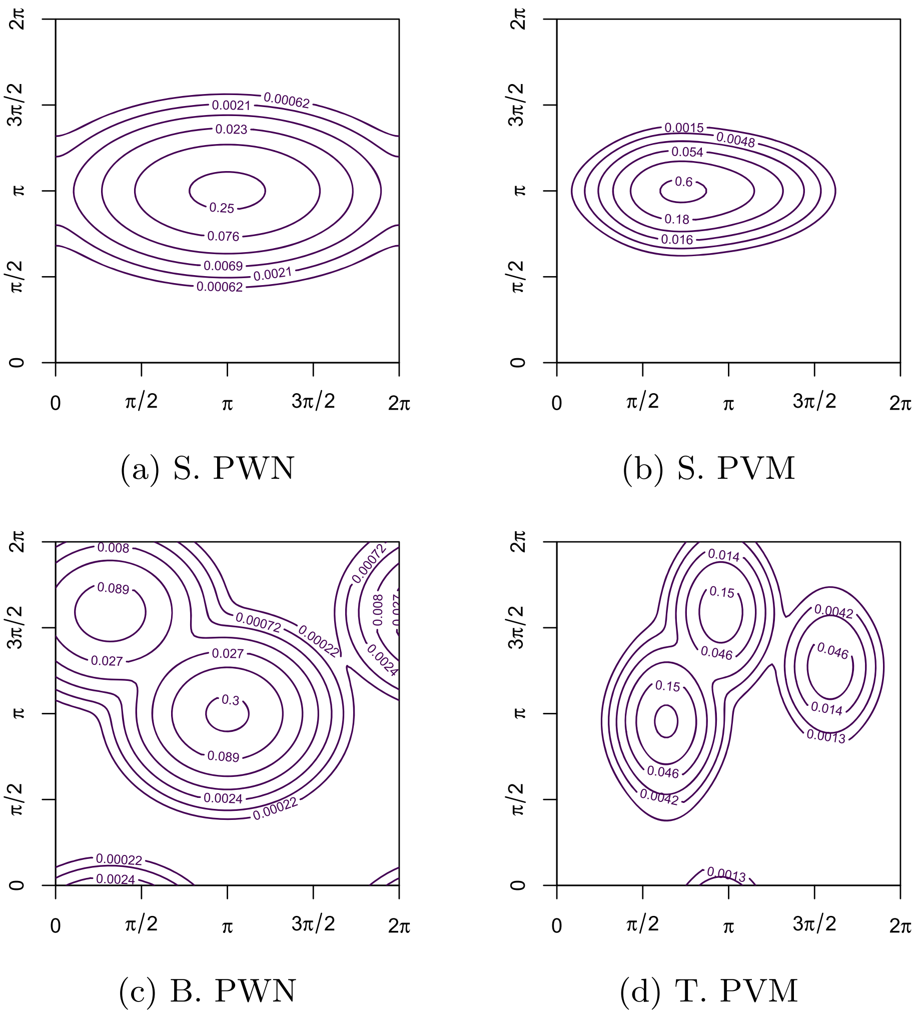
As discussed in Sect. 2.4, the smoothing parameter for our mixture is selected on the basis of the model selection criterion AIC$_\mathrm{c}$. Generally, to generate a sequence of potential h-values, one would like to start tuning the model from an over-smoothed estimate with a large bandwidth, and gradually decrease the h-value until it results in an obviously under-smoothed estimate. For each value of h, $(\widehat{G}_h, \widehat{\boldsymbol{\beta}}_h)$ can be computed by the algorithm described in Sect. 3. The optimal smoothing parameter and its corresponding density estimate are chosen for the lowest AIC$_\mathrm{c}$-value.

In each scenario, 20 data sets are randomly generated with sample sizes 100 and 500, respectively. The average loss including the mean ISE (MISE), the mean KL (MKL) and the mean HD (MHD) are summarized in Table 2, with standard errors given in parentheses. In this table, and later in Tables 4–9, each entry in boldface indicates the smallest value among the methods in comparison in each case. The results show that the two candidate families give similar performance in most cases, and no one dominates the other. Their performance depends on the simulation family, the sample size, and the measurement of loss. In the first two simulated models without obvious correlation, the wrapped normal is superior to the von Mises density, whereas the von Mises is more advantageous in the presence of multi-modality and correlation in the last two models. Comparing among three loss metrics, the von Mises family seems to perform slightly better in terms of the ISE and the wrapped normal performs better in terms of the other two. A plausible reason is that the wrapped normal has lighter tails than the von Mises, and it

Table 1 Distributions for simulation studies

Name	Distribution
(a) Single PWN	$\mathrm{PWN}((\pi, \pi)^\top, (1, 0.5)^\top)$
(b) Skewed PVM	$\sum_{i=0}^{5} \frac{(i+1)^2}{91} \mathrm{PVM}((2$ $+ 0.4i, \pi)^\top, (5, 10)^\top)$
(c) Bimodal PWN	$\frac{3}{10}\mathrm{PWN}((1, 5)^\top, (0.6, 0.5)^\top)$ $+ \frac{7}{10}\mathrm{PWN}((\pi, \pi)^\top, (0.6, 0.5)^\top)$
(d) Trimodal PVM	$\frac{3}{5}\mathrm{PVM}((2, 3)^\top, (10, 5)^\top)$ $+ \frac{3}{10}\mathrm{PVM}((3, 5)^\top, (10, 5)^\top)$ $+\frac{1}{10}\mathrm{PVM}((5, 4)^\top, (10, 5)^\top)$

Fig. 3 Contour plots for the four simulated densities listed in Table 1



(a) S. PWN

(b) S. PVM

(c) B. PWN

(d) T. PVM

Table 2 Simulation results for four distributions in Table 1 for the two component densities

Loss	Comp.	(a) S. PWN	(b) S. PVM	(c) B. PWN	(d) T. PVM
$n = 100$					
MISE	PVM	**0.55** (0.01)	**1.37** (0.03)	**0.72** (0.02)	**2.11** (0.04)
	PWN	0.57 (0.02)	1.57 (0.04)	0.95 (0.02)	2.22 (0.05)
MKL	PVM	5.52 (0.09)	5.50 (0.11)	**5.61** (0.14)	**10.44** (0.23)
	PWN	**4.13**(0.12)	**4.84** (0.11)	6.93 (0.15)	10.58 (0.32)
MHD	PVM	3.23 (0.05)	3.13 (0.06)	4.44 (0.10)	5.98 (0.11)
	PWN	**2.27** (0.07)	**2.53** (0.06)	**3.82** (0.09)	**5.58** (0.13)
$n = 500$					
MISE	PVM	**0.16** (0.00)	**0.37** (0.01)	**0.19** (0.00)	**0.32** (0.01)
	PWN	**0.16** (0.01)	0.45 (0.01)	0.29 (0.01)	0.49 (0.01)
MKL	PVM	2.12 (0.04)	1.60 (0.04)	**2.07** (0.05)	**0.85** (0.04)
	PWN	**1.18** (0.05)	**1.48** (0.04)	2.08 (0.07)	2.07 (0.03)
MHD	PVM	1.19 (0.02)	0.91 (0.03)	1.52 (0.03)	**1.00** (0.02)
	PWN	**0.64** (0.03)	**0.80** (0.02)	**1.18** (0.04)	1.07 (0.01)

may sacrifice the goodness of fit in areas with high density but perform better around the tails.

Between the two, we advocate the use of the von Mises family for mixture components. It is advantageous with increasing sample sizes and model complexity and is also competitive in small data sets. In addition, the density function of the wrapped normal distribution is an infinite sum of terms and is computationally expensive to evaluate (even after ignoring minor terms), especially in higher dimensions. In the following studies, we will only use the von Mises as mixture components due to its competitive performance, cheaper computational cost and wider usage in both parametric and nonparametric modelling.

5.3 Comparison with kernel density estimators

Next, we would like to compare the accuracy between KDEs and MDEs. Regarding our mixture-based density estimators, the mixture component used here is the product von Mises distribution. Both the model selection criterion AIC_c and the cross-validation method minimizing, respectively, the integrated squared error and the Kullback-Leibler divergence are utilized.

Di Marzio et al. (2011) proposed and studied some kernel density estimators and provided their implementations in the MATLAB language. Their simulation studies were undertaken with the toolbox CIRCSTAT written by Berens (2009), and the optimization step made use of the OPTIMIZATION toolbox. To enable a convenient comparison in R, we implemented their LCV and UCV bandwidth selectors. As for the BCV selector, the objective function to be minimized is rather complicated, and it tends to be dominated by either LCV or UCV as shown in the simulation studies of Di Marzio et al. (2011). Therefore, we chose not to implement it. There is also a plug-in estimator using a bivariate von Mises kernel in Taylor et al. (2012). However, it only considers the problem in two-dimensions. In addition, from our experience using plug-in estimators for other types of directional data, they only have a reasonable performance when the underlying data largely have the shape of the kernel. As a consequence, we did not include it in our numerical studies.

To compare between the two kernel density estimators and three mixture-based estimators, four simulation models are considered using mixtures of product von Mises distributions. We consider the sample sizes $n = 100$ and 500 in $d = 2$ or 4 dimensions. The bivariate mixture models and their two-dimensional contour plots are shown in Table 3 and Fig. 4, respectively. The four models have 1, 2, 8 and 32 mixture components, respectively, and the modelling complexity increases correspondingly. The four-dimensional models basically follow from the two-dimensional ones, by adding 2 zeros to the vectors of location parameters and 2 ones to the vectors of concentration parameters.

The averaged losses over 50 repetitions are given in Tables 4, 5 and 6 with standard errors in parentheses. Both the means and standard errors are rounded to two decimal places after scaling.

As evident from the three tables, the three MDEs dominate the two KDEs in almost all cases. The differences between them can sometimes be several times of the standard error, indicating a clear superiority of mixture density estimators. Although in the latter two scenarios in two dimensions, LCV has the lowest MISE when the sample size is 100, there is at least one MDE that is still competitive in this situation. Moreover, MDE is clearly superior for a large sample size or in higher dimensions, which is consistent with our expectation based on the characteristics of the corresponding estimators. Among the three mixture-based estimators, CVKL tends to perform the best in various scenarios, and the two cross-validation approaches both outperform the AIC_c-based one. It is not surprising that MDEs manifest such strong advantages over KDEs. To model toroidal data, KDE bandwidth selectors can only have the same concentration parameters among all dimensions, which drastically decreases its flexibility. On the other hand, MDEs are able to model different concentrations in different dimensions by incorporating the bandwidth factor h, which is the maximum of the estimated joint density. By fixing the h-value, the best choice of $\boldsymbol{\kappa}$ is automatically selected by likelihood maximization. Therefore, one may conclude that the mixture-based estimators on the torus are indeed superior in general.

6 Real-world data analysis

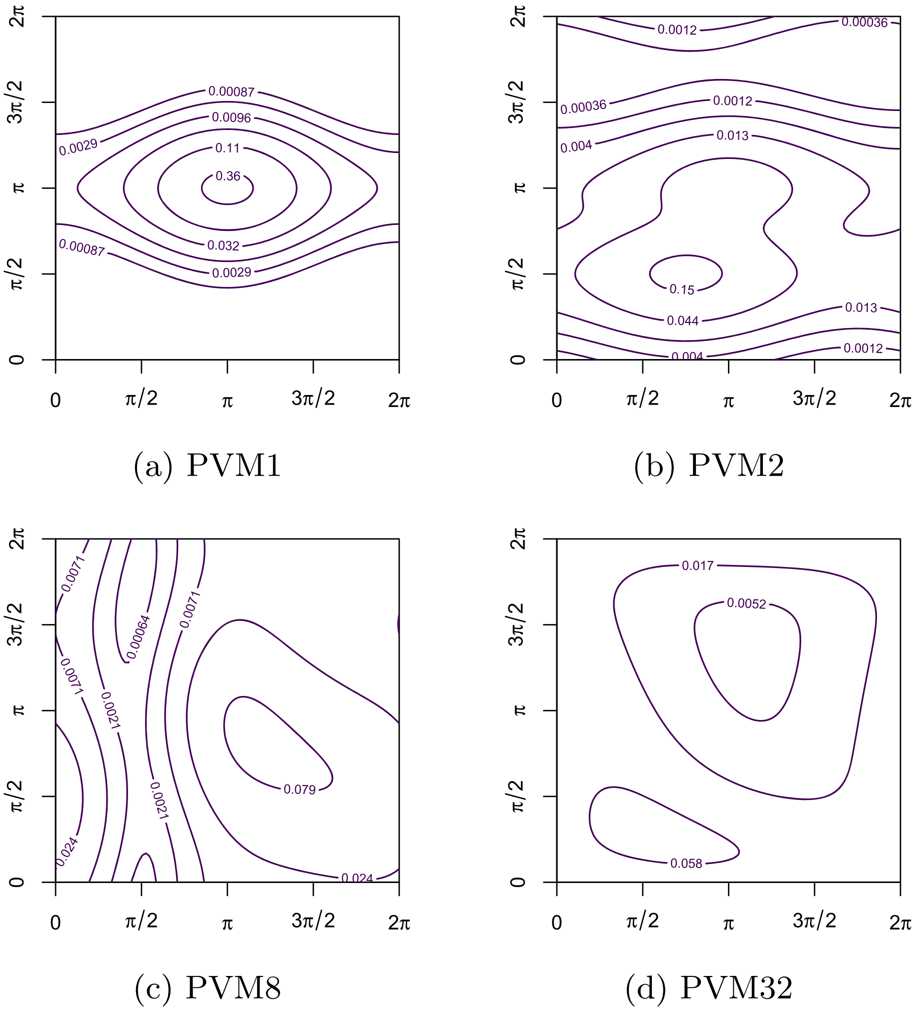
In this section, three real-world density estimation problems are studied. The first one is the classical wind direction data, which usually appears in one dimension and can be analyzed using tools for circular observations, but here the measurements in three locations are combined into trivariate toroidal observations. The other two data sets are related to bioinformatics concerning protein backbones and RNA structures, and such problems form the major applications of toroidal data.

6.1 Setup

Same as simulation studies, we are going to utilize the same five bandwidth selectors to facilitate the comparison between two sets of estimators for real-world data sets. In terms of the mixture density estimators, we use the von Mises component density to take advantage of its computational simplicity, which is more obvious in higher dimensions.

Table 3 Simulated distributions in two-dimensions to compare KDEs and MDEs

Name	Distribution
(a) PVM1	$\mathrm{PVM}((\pi,\pi)^\top,(2,5)^\top)$
(b) PVM2	$\frac{3}{10}\mathrm{PVM}((\pi,\pi)^\top,(1,4)^\top)+\frac{7}{10}\mathrm{PVM}((\frac{3\pi}{4},\frac{\pi}{2})^\top,(1,4)^\top)$
(c) PVM8	$\frac{1}{8}\sum_{i=0}^{7}\mathrm{PVM}((2\pi(\frac{9}{10})^{7-i},\pi(\frac{9}{10})^{i})^\top,(4,1)^\top)$
(c) PVM32	$\frac{1}{32}\sum_{i=0}^{31}\mathrm{PVM}((2\pi(\frac{9}{10})^{7-i},2\pi(\frac{9}{10})^{i})^\top,(1,3)^\top)$



(a) PVM1

(b) PVM2

(c) PVM8

(d) PVM32

Fig. 4 Contour plots for the four simulated densities listed in Table 3

To assess the performance of estimators, the following two sensible losses are considered (omitting additive constants):

$$\mathrm{ISE}(\hat{f}_n,\hat{f})=\int_{\mathbb{T}}\hat{f}(\boldsymbol{x})^2\,\mathrm{d}\boldsymbol{x}-\frac{2}{n}\sum_{i=1}^{n}\hat{f}(\boldsymbol{x}_i),$$

$$\mathrm{KL}(\hat{f}_n,\hat{f})=-\frac{1}{n}\sum_{i=1}^{n}\log[\hat{f}(\boldsymbol{x}_i)],$$

where $\hat{f}$ denotes a density estimate from the training set, and $\hat{f}_n$ the empirical probability mass function of the test data of size n. It is worth noting that with a von Mises component density, there is no need to compute the integral in ISE numerically nor applying the importance sampling as in the simulation. Instead, the product of two von Mises densities is still proportional to a von Mises density. Therefore, to avoid integrating $\hat{f}(\boldsymbol{x})^2$, one may directly compute the sum of the product of the weights multiplied by the corresponding normalizing constants.

Regarding the bandwidth selectors for mixture models, both the model selection criterion $\mathrm{AIC_c}$ and the cross-validation method are adopted here. First we would like to obtain the model with the lowest $\mathrm{AIC_c}$. For the data sets involving wind direction and the protein backbone structure, as they both have small to moderate sample sizes, 10 repetitions of 10-fold cross-validation are conducted for each

Table 4 Simulation results for MISE ($\times 10^3$)

Estimator	(a) PVM1	(b) PVM2	(c) PVM8	(d) PVM32
$d = 2, n = 100$				
LCV	15.90 (0.74)	6.94 (0.35)	**3.87** (0.24)	**3.24** (0.17)
UCV	17.05 (0.89)	7.76 (0.40)	4.01 (0.23)	3.37 (0.18)
AIC$_c$	9.30 (0.81)	5.57 (0.43)	4.29 (0.31)	4.22 (0.13)
CVISE	9.97 (1.59)	5.38 (0.60)	4.97 (0.51)	4.18 (0.31)
CVKL	**7.63** (1.00)	**4.52** (0.43)	4.71 (0.42)	3.89 (0.22)
$d = 2, n = 500$				
LCV	6.17 (0.32)	2.98 (0.10)	1.42 (0.05)	1.24 (0.05)
UCV	6.14 (0.33)	3.21 (0.12)	1.55 (0.06)	1.34 (0.06)
AIC$_c$	2.98 (0.39)	1.52 (0.12)	1.05 (0.08)	1.36 (0.09)
CVISE	2.34 (0.55)	1.29 (0.11)	1.18 (0.09)	1.14 (0.06)
CVKL	**1.52** (0.18)	**1.13** (0.08)	**1.04** (0.07)	**1.13** (0.07)
$d = 4, n = 100$				
LCV	2.84 (0.06)	0.94 (0.02)	0.53 (0.01)	0.42 (0.01)
UCV	2.89 (0.07)	0.98 (0.03)	0.55 (0.02)	0.43 (0.01)
AIC$_c$	2.58 (0.10)	1.11 (0.04)	0.76 (0.02)	0.51 (0.02)
CVISE	1.11 (0.09)	0.58 (0.04)	0.48 (0.03)	0.37 (0.02)
CVKL	**0.98** (0.05)	**0.54** (0.03)	**0.38** (0.02)	**0.35** (0.02)
$d = 4, n = 500$				
LCV	1.62 (0.03)	0.58 (0.01)	0.29 (0.01)	0.24 (0.00)
UCV	1.60 (0.04)	0.59 (0.01)	0.29 (0.01)	0.25 (0.01)
AIC$_c$	1.01 (0.04)	0.29 (0.01)	0.32 (0.01)	0.20 (0.00)
CVISE	0.27 (0.02)	0.14 (0.01)	**0.10** (0.00)	**0.12** (0.00)
CVKL	**0.21** (0.02)	**0.13** (0.01)	**0.10** (0.00)	**0.12** (0.00)

data set, and the average loss will be computed. For the RNA data which has a large number of observations ($n = 8301$), cross-validation is not necessary. Thus, we randomly partition the data set into two subsamples ($1000 + 7301$), using the first one for density estimation and the second one for its performance evaluation.

As for the K-fold cross-validation approach to select the best model, the procedure also slightly differs between data sets with small and large sample sizes. For the wind direction and protein data, the data sets are randomly partitioned into K subsets $P_1, \ldots, P_K$, repeatedly with $K - 1$ parts for training the model and the remaining one for testing the estimate. To measure the performance, we also resort to either the squared error loss or the Kullback-Leibler loss. The formulae without the additive constants are given by

$$\mathrm{CVISE}(h) = \frac{1}{K}\sum_{j=1}^{K}\int f_h^{(j)}(\boldsymbol{x})^2\,\mathrm{d}\boldsymbol{x}$$

$$-\frac{2}{K}\sum_{j=1}^{K}\frac{1}{n_j}\sum_{\boldsymbol{x}_i\in P_j} f_h^{(j)}(\boldsymbol{x}_i),$$

$$\mathrm{CVKL}(h) = -\frac{1}{K}\sum_{j=1}^{K}\frac{1}{n_j}\sum_{\boldsymbol{x}_i\in P_k}\log[f_h^{(k)}(\boldsymbol{x}_i)],$$

where $f_h^{(j)}$ is the mixture model fitted to all observations but those in P_j and n_j the number of observations in P_j. For the first two real data sets, K is chosen to be 10. For all three data sets, the models with the lowest CVISE and CVKL are then selected.

6.2 Wind directions

The trivariate wind directional data contains 1682 sets of wind directions measured at three monitoring stations at 14 p.m. on days between January 1, 1993 and February 29, 2000. The respective locations are San Agustin in the north, Pedregal in the southwest, and Hangares in the southeast of the Mexico Valley. The data set is available from the R package “`CircNNTSR`” (Fernandez-Duran et al. 2016).

The resultant estimated density is shown in Fig. 5. As we can see from the plot, observations are spread all around the circle in all three dimensions, indicating relatively small concentration parameter values for mixture components. Comparing the results among the five bandwidth selectors

Table 5 Simulation results for MKL ($\times 10^2$)

Estimator	(a) PVM1	(b) PVM2	(c) PVM8	(d) PVM32
$d = 2, n = 100$				
LCV	9.61 (0.35)	8.65 (0.43)	7.21 (0.56)	5.92 (0.37)
UCV	10.42 (0.41)	9.46 (0.46)	7.24 (0.39)	**5.83** (0.32)
AIC$_c$	**4.71** (0.40)	6.14 (0.53)	6.97 (0.57)	7.34 (0.25)
CVISE	6.41 (1.17)	5.91 (0.60)	7.14 (0.68)	7.04 (0.55)
CVKL	4.97 (0.94)	**4.98** (0.45)	**6.87** (0.65)	6.63 (0.44)
$d = 2, n = 500$				
LCV	3.76 (0.12)	3.54 (0.12)	2.30 (0.08)	2.00 (0.08)
UCV	4.01 (0.13)	3.63 (0.13)	2.55 (0.09)	2.08 (0.09)
AIC$_c$	1.57 (0.19)	1.39 (0.13)	1.55 (0.14)	1.84 (0.13)
CVISE	1.34 (0.28)	1.27 (0.12)	1.67 (0.15)	**1.64** (0.10)
CVKL	**0.96** (0.09)	**1.21** (0.11)	**1.43** (0.12)	1.68 (0.11)
$d = 4, n = 100$				
LCV	30.68 (0.48)	23.23 (0.38)	18.03 (0.46)	15.79 (0.41)
UCV	30.83 (0.50)	23.71 (0.46)	18.45 (0.48)	15.98 (0.45)
AIC$_c$	18.71 (0.78)	23.81 (1.08)	24.37 (0.90)	18.95 (0.67)
CVISE	11.11 (0.96)	13.06 (0.80)	15.40 (1.05)	14.32 (1.19)
CVKL	**9.25** (0.38)	**12.09** (0.57)	**12.77** (0.64)	**13.24** (0.91)
$d = 4, n = 500$				
LCV	16.57 (0.24)	12.89 (0.20)	9.13 (0.16)	8.22 (0.16)
UCV	16.66 (0.23)	13.06 (0.21)	9.18 (0.16)	8.34 (0.18)
AIC$_c$	6.24 (0.27)	4.67 (0.24)	8.95 (0.21)	6.30 (0.14)
CVISE	2.86 (0.21)	3.12 (0.18)	3.17 (0.16)	4.16 (0.15)
CVKL	**2.31** (0.16)	**2.85** (0.15)	**2.93** (0.12)	**4.07** (0.16)

in Table 7, the CVISE fit outperforms the UCV one and has the lowest MISE, while the CVKL model outperforms the LCV one and has the lowest MKL loss. Among the three mixture density estimates, the model selected based on the lowest CVISE has the smallest h-value and contains the largest number of components. As indicated from the plot, it is the least smooth one.

It is worth mentioning that in Fig. 5, the marginal estimated densities are superimposed onto the circular histograms of the corresponding dimension. Note that they are all area-proportional to better reveal the true underlying structure of the data set. In particular, the term "area-proportional" indicates that, in a circular histogram, the frequency of each bar is proportional to its area rather than height. Similarly, one can also interpret a circular density in terms of the enclosed area. The plots are constructed using the R package "`cplots`" (Xu and Wang 2019) and more details can be found in Xu and Wang (2020).

6.3 Protein dihedral angles

Proteins are a class of macromolecules composed by numbers of peptide-bonded amino acids, where a linear chain of amino acid residues is called a polypeptide. Proteins perform a diverse range of functions within the body including transmitting signals to coordinate cells, allowing metabolic reactions, building and repairing tissues, etc. (Liljas et al. 2016). Hence it is a critical topic in bioinformatics to determine the protein structure by polypeptide backbones. The configuration of the backbone can be described by three dihedral angles ϕ, ψ and ω, where a dihedral angle is the angle between amide planes through sets of atoms. Due to the planarity of the peptide bond, ω is often restricted to be 180° (the *trans* case) or 0° (the *cis* case), whereas the other two conformational angles connected to the C_α atoms are free to rotate (Boomsma et al. 2008). In particular, ϕ involves the backbone atoms $\mathrm{C} - \mathrm{N} - \mathrm{C}_\alpha - \mathrm{C}$ and ψ is the torsion angle within $\mathrm{N} - \mathrm{C}_\alpha - \mathrm{C} - \mathrm{N}$ (García-Portugués et al. 2018). The tertiary structure of a peptide bond can then be determined if all dihedral angles related to the corresponding C_α atoms are known. Thus, we are interested in $(\phi, \psi)^\top$ which reveals the structure of protein backbones and lies naturally on a two-dimensional torus.

The data set used here forms a representative sample from the Protein Data Bank (Berman et al. 2006) and is retrieved from the R package "`CircNNTSR`". It contains 233 pairs of conformational angles ϕ and ψ. Unlike the wind directional data which is relatively uniform, the protein data is

Table 6 Simulation results for MHD ($\times 10^2$)

Estimator	(a) PVM1	(b) PVM2	(c) PVM8	(d) PVM32
$d = 2, n = 100$				
LCV	4.97 (0.18)	4.60 (0.21)	3.99 (0.21)	**3.22** (0.17)
UCV	5.32 (0.19)	5.31 (0.29)	4.38 (0.22)	3.35 (0.18)
AIC$_c$	2.61 (0.23)	3.45 (0.31)	4.41 (0.35)	4.45 (0.13)
CVISE	2.83 (0.34)	3.00 (0.27)	3.84 (0.31)	3.81 (0.24)
CVKL	**2.34** (0.27)	**2.61** (0.21)	**3.65** (0.28)	3.58 (0.20)
$d = 2, n = 500$				
LCV	1.89 (0.06)	1.91 (0.05)	1.41 (0.03)	1.16 (0.04)
UCV	1.96 (0.06)	1.99 (0.05)	1.62 (0.05)	1.25 (0.05)
AIC$_c$	0.82 (0.09)	0.83 (0.07)	1.03 (0.09)	1.28 (0.10)
CVISE	0.67 (0.12)	0.71 (0.05)	0.97 (0.06)	0.97 (0.05)
CVKL	**0.49** (0.05)	**0.66** (0.05)	**0.84** (0.05)	**0.96** (0.06)
$d = 4, n = 100$				
LCV	16.80 (0.26)	12.96 (0.24)	10.17 (0.24)	8.42 (0.19)
UCV	17.04 (0.26)	13.39 (0.28)	10.47 (0.27)	8.56 (0.22)
AIC$_c$	9.92 (0.40)	13.07 (0.61)	14.27 (0.51)	10.69 (0.34)
CVISE	5.80 (0.42)	6.77 (0.38)	8.19 (0.49)	7.70 (0.47)
CVKL	**4.92** (0.19)	**6.31** (0.28)	**6.98** (0.32)	**7.28** (0.39)
$d = 4, n = 500$				
LCV	8.96 (0.10)	7.10 (0.09)	5.33 (0.07)	4.53 (0.07)
UCV	8.95 (0.10)	7.19 (0.11)	5.38 (0.07)	4.60 (0.07)
AIC$_c$	3.16 (0.12)	2.54 (0.12)	5.56 (0.10)	4.01 (0.06)
CVISE	1.51 (0.10)	1.68 (0.08)	1.90 (0.07)	2.49 (0.08)
CVKL	**1.21** (0.08)	**1.56** (0.07)	**1.78** (0.05)	**2.42** (0.08)

quite clustered in each dimension. Thus, to better present the bivariate data, observations from $[0, \frac{\pi}{2})$ are transformed to $[2\pi, \frac{5\pi}{2})$. In Fig. 6, the contour plots represent the final model chosen by the five density estimators, respectively. Combining with the corresponding results in Table 8, it is evident that the AIC$_c$ model is in good agreement with the CVKL model in all aspects, which may be attributed to their basis of maximum likelihood. Hence, they both perform significantly well and are superior to the LCV selector in terms of MKL. On the other hand, the estimated density chosen by the lowest CVISE dominates all the others in terms of MISE. It has more than twice as many mixture components as the AIC$_c$-selected model.

Table 7 Results of models based on five bandwidth selectors for the wind direction data set, with MISE and MKL ($\times 10^2$), the bandwidth parameter and the number of components

Estimator	MISE	MKL	h	#Comp
LCV	−3.20 (0.01)	451.49 (0.17)	–	–
UCV	−3.55 (0.01)	460.41 (0.22)	–	–
AIC$_c$	−2.56 (0.00)	459.65 (0.15)	0.50	32
CVISE	−**3.74** (0.00)	452.82 (0.20)	0.19	355
CVKL	−3.58 (0.01)	**448.31** (0.04)	0.29	157

6.4 RNA structure

Ribonucleic acid (RNA) is another polymer essential for all known forms of life. While a protein is made up of chained amino acids, RNA and deoxyribonucleic acid (DNA) are both nucleic acids comprising chains of nucleotides. RNA carries out a broad range of functions including regulating gene expression, facilitating the translation of DNA into proteins and catalyzing biological reactions. RNA resembles DNA with the same basic components, but it has a single strand folded onto itself rather than a paired double helix. Along the RNA backbone, each nucleotide comprises six backbone dihedral angles $\alpha, \beta, \gamma, \delta, \epsilon, \xi$ and one torsion angle χ describing the rotation of the base relative to the sugar, which is more complicated than the backbone structure of the amino acids in a protein (Lee and Gutell 2004). We are going to analyze the RNA backbone conformations using the torus formed by the seven torsion angles.

The data set used here contains 8301 residues. It was first obtained by Duarte and Pyle (1998) using high experimental X-ray precision and later updated by Wadley et al. (2007).

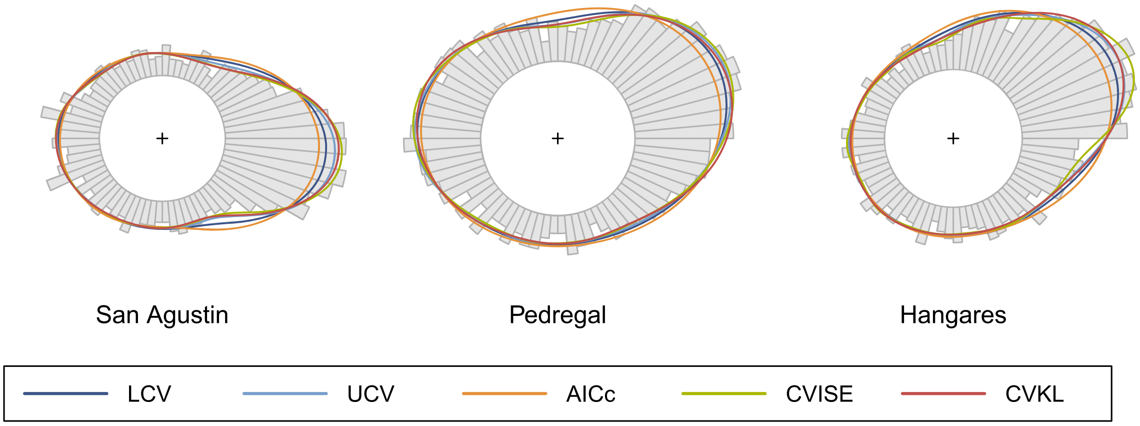


Fig. 5 The area-proportional circular histogram of each dimension of the wind directional data set ($n = 1682$), superimposed with five marginal estimated densities represents the result for LCV, UCV, AIC$_c$, CVISE and CVKL, respectively

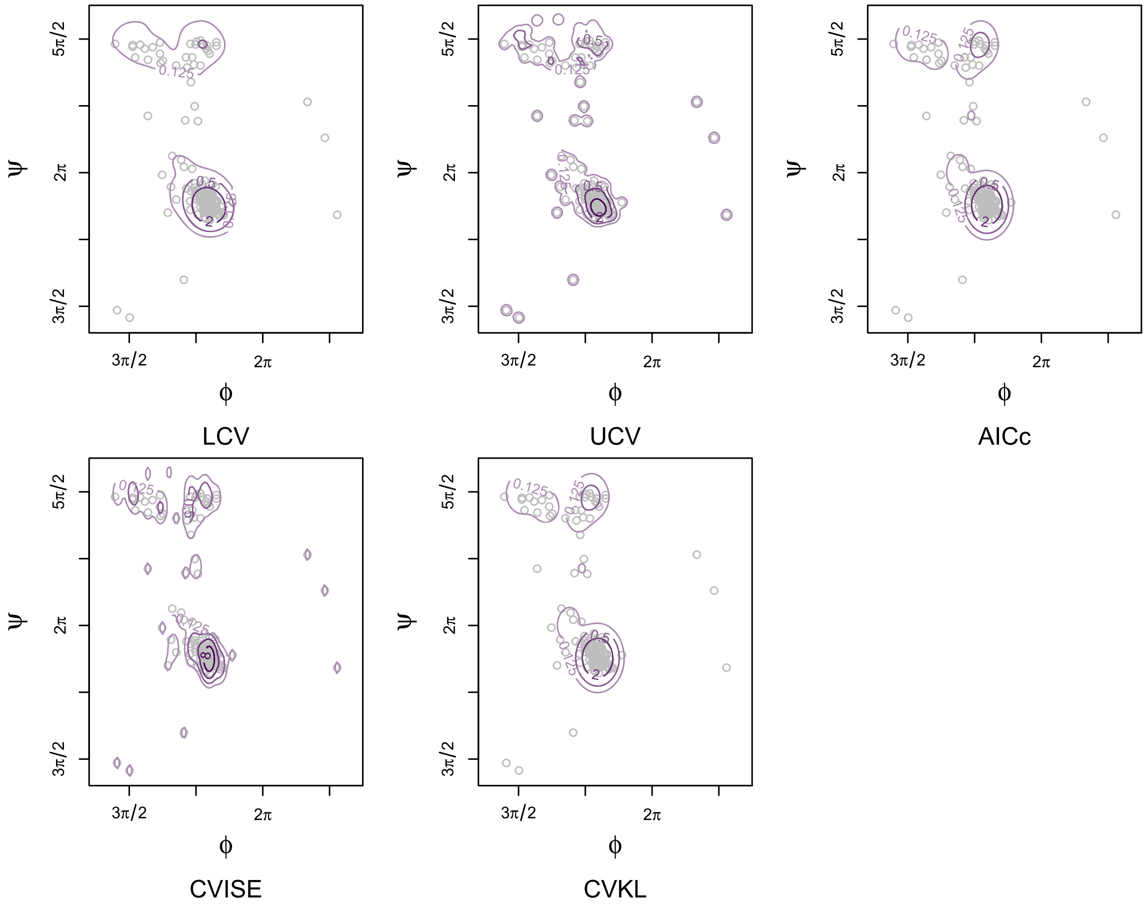


Fig. 6 Contour plots of five estimated densities for the protein data set ($n = 233$), where the contour lines are all drawn at density levels of 8, 2, 0.5 and 0.125

Table 8 Results of models based on five bandwidth selectors for the protein data set, with MISE, MKL, the bandwidth parameter and the number of components

Estimator	MISE	MKL	h	#comp.
LCV	−5.02 (0.01)	−0.20 (0.01)	–	–
UCV	−7.70 (0.02)	0.05 (0.01)	–	–
AIC$_c$	−5.33 (0.01)	−0.34 (0.00)	0.28	17
CVISE	**−9.51** (0.01)	−0.09 (0.02)	0.14	40
CVKL	−6.68 (0.04)	**−0.43** (0.00)	0.27	18

This large and classical data set has been analyzed by many researchers, for example, Eltzner et al. (2018) and Nodehi et al. (2021), where the latter paper carried out the principal component and clustering analysis on toroidal data. Here we are interested in estimating the density of the seven-dimensional toroidal data to reveal the structure of RNA backbones. Owing to the high dimensionality, it is impossible to view the whole data set and the joint distribution on a torus. Therefore, we present their univariate marginal plots and one of the bivariate marginal plots in Figs. 7 and 8, respectively, but one has to bear in mind that the goodness-of-fit marginally is not equivalent to the goodness-of-fit jointly in seven dimensions. Along with the results shown in Table 9, it is evident that the model generated by AIC$_c$ is clearly over-smoothed with the largest MISE and MKL value and only 122 components, which seems unrealistic for this data set. In terms of the cross-validation methods, CVISE has a markedly low MISE value whereas CVKL has a substantially small MKL value as expected. Similarly as the previous two data sets, they both outperform their KDE competitors UCV and LCV, respectively. Based on the shape of marginal densities and the bandwidth parameters, the AIC$_c$ and UCV models have the highest and lowest level of smoothness, respectively, while LCV and CVKL models provide similar and moderate level of smoothness.

Comparing with the previous two examples, RNA data is in seven-dimensional space which results in much more parameters to estimate, and consequently the model selection criterion AIC$_c$ will have much larger penalty for model complexity. Therefore, AIC$_c$ tends to choose over-simplified models with increasing dimensionality whereas the cross-validation approach is relatively robust in regard to dimensions.

Using the product distributions makes it straightforward to produce marginal densities of any dimension. Figure 8 shows the contour plots of the 2-dimensional marginal density for (δ, ξ) obtained by the five estimators, respectively. These plots provide extra information about the performance of these estimates. Though using only 2 out of 5 variables, they show that the UCV estimate is under-smoothed, the AIC$_c$ one is likely over-smoothed, and the other three seem acceptable. Such visual observation is largely consistent with the results given in Table 9.

7 Concluding remarks

In the above, we studied density estimation of multi-dimensional toroidal data using semiparametric mixture models. It takes the form of an integral of the mixture component with respect to a mixing distribution G, where G can take a completely unspecified form. There always exists a discrete solution of the NPMLE of G.

In the multivariate setting, one of the major difficulties in the nonparametric modelling is to decide the smoothness of the mixture estimate. It is inappropriate to leave it entirely determined by the likelihood maximization as the resultant mixture will become degenerate with an infinite likelihood. Moreover, for toroidal observations, one also needs to avoid the situation where the estimated density tends to uniformity in some dimensions while clustering around a single point in the others. Therefore, we directly fix the maximum of the mixture component density and define h to be the reciprocal of the maximum to set the level of smoothness of the mixture density. For such a fixed h-value, one can maximize the bounded likelihood $l(G, \boldsymbol{\beta})$ to obtain a meaningful mixture model estimated by our algorithm. In fact, the concept of controlling the maximum can be applied to any type of mixture components to effectively bound the likelihood. In terms of determining the value of the maximum, we find it rather straightforward to resort to the model selection procedure. In particular, with a sequence of h-values, their corresponding $(\widehat{G}_h, \widehat{\boldsymbol{\beta}}_h)$ can be found, and they form a family of mixture models indexed by h. One may choose the best fitting model with the information criterion AIC$_c$ or apply a simulation-based method such as the cross-validation approach if the computational cost is acceptable.

Regarding the toroidal component distributions, the product of univariate distributions is considered in this paper, including the von Mises and the wrapped normal distribution. They are both widely used in the study of directional data and have similar numerical performance in our simulation studies. As the von Mises density belongs to the exponential family and is mathematically simpler, it is more advantageous in statistical analysis. Adapting a von Mises component in our mixture model, the MDE clearly compares favorably against the KDE in almost all cases. For the KDE being a convolution of the kernel and the empirical probability mass function, it tends to flatten the estimated density and results in a higher bias. In contrast, the MDE is a convolution between the mixing distribution and the component density. Thus, it is a de-convolution process that computes directly the mixing distribution, and a well-fitted mixture density may have a lower bias. In addition, MDE possesses a much simpler

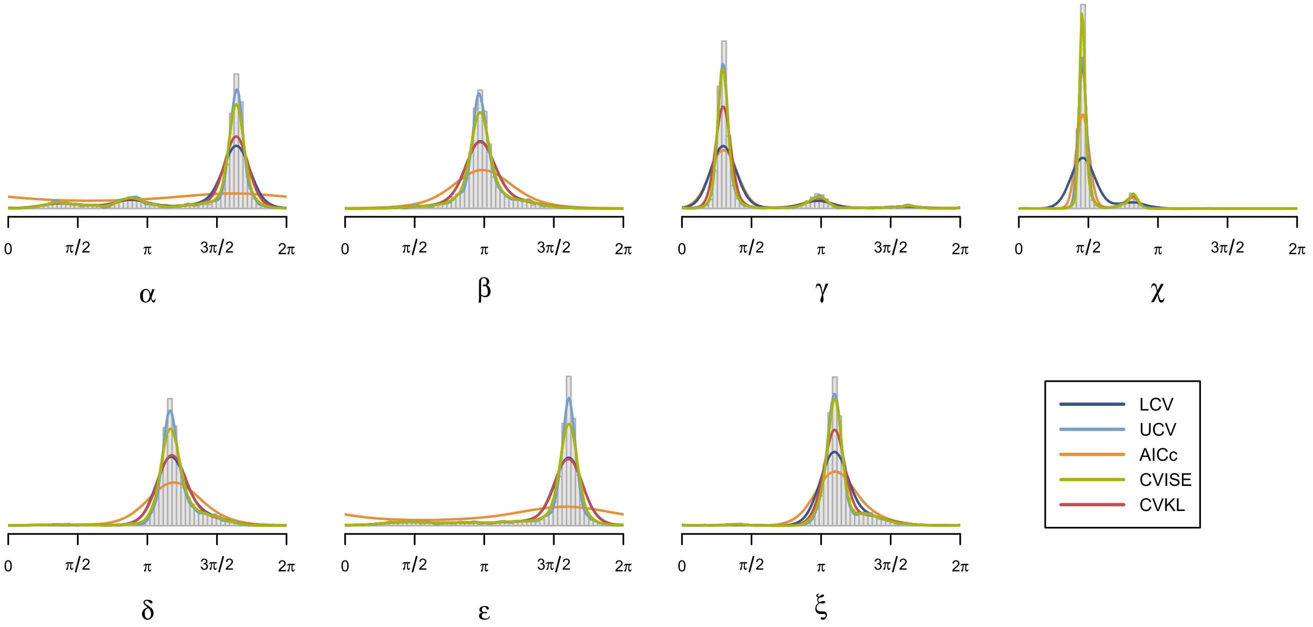


Fig. 7 The histogram of each dimension of the RNA data set ($n = 8301$), superimposed with five marginal estimated densities represents the result for LCV, UCV, AIC$_c$, CVISE and CVKL, respectively

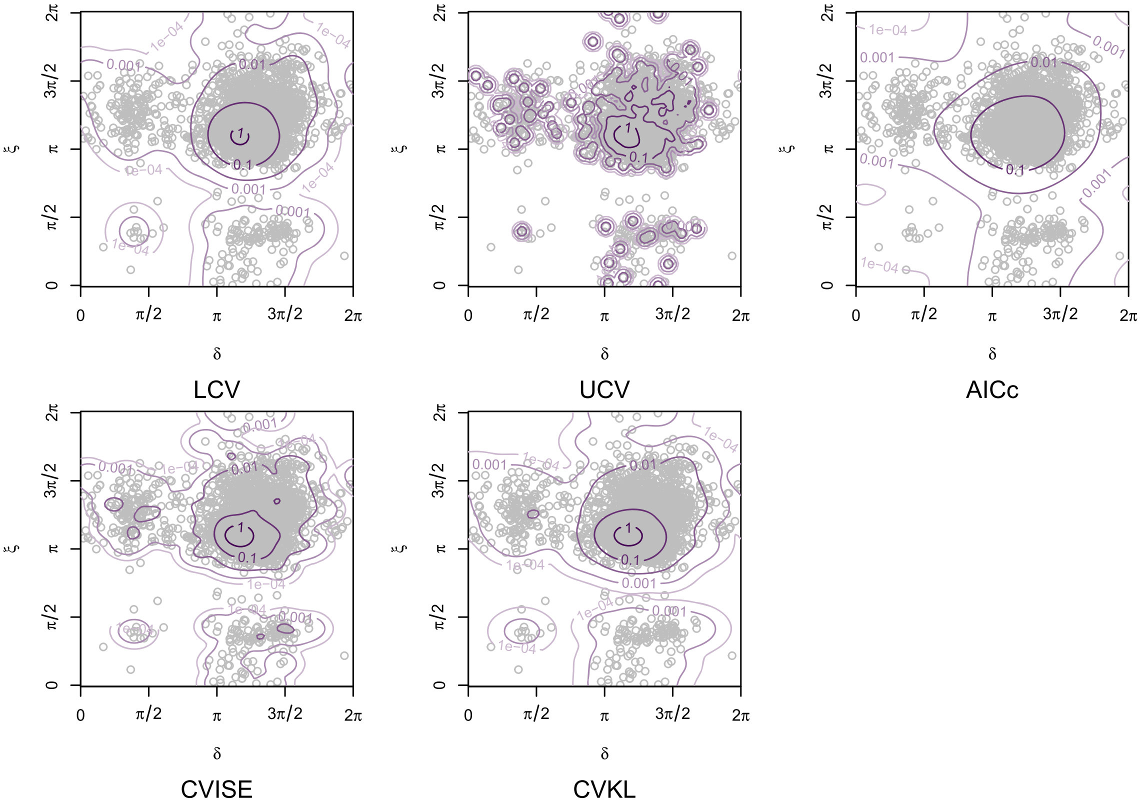


Fig. 8 Contour plots of five estimated densities for the RNA data set ($n = 8301$), where the contour lines are all drawn at density levels of 1, 0.1, 0.01, 0.001 and 0.0001

Table 9 Results of models based on five bandwidth selectors for the RNA data set, with MISE, MKL, the bandwidth parameter and the number of components

Estimator	MISE	MKL	h	#comp
LCV	− 2.62	3.22	–	–
UCV	− 103.96	12.48	–	–
AIC$_c$	− 0.06	4.73	1.00	122
CVISE	**− 139.17**	4.64	0.23	468
CVKL	**− 15.35**	**2.26**	0.38	307

expression than KDE as it can possibly be represented by the weighted sum of a few mixture components, whereas KDE always keeps record of all observations. On account of the prevalence of the curse of dimensionality, KDE can hardly provide an accurate density estimation in high dimensions.

There are some potential improvements for our method. To better incorporate the correlation between dimensions, it is beneficial to consider the correlation between univariate densities to enable a more flexible mixture. In addition, one may also vary the smoothing parameter for each mixture component depending on its location. It is similar in spirit to the adaptive or variable-bandwidth kernel density estimation. In the real-world data set, it is not uncommon to observe some extreme tails such as the ones in the protein and RNA data. In this case, using a fixed global bandwidth may result in a mixture with extremely small bandwidth to cope with low-density areas, but it will create redundant components in areas with large numbers of observations. Thus, a variable bandwidth matrix may effectively reduce the number of components needed in the model, and it will be particularly useful when the sample space is in high dimensions as the size of the problem increases exponentially with the dimensionality.

Apart from data lying on a torus, the nonparametric mixture modelling can also be applied to other types of multidimensional data such as those in a Euclidean space, on the surface of a hyper-sphere, or even the combination of them. One simply needs to find the appropriate univariate density function to model each dimension and take the product of them to form the joint density. Once the maximum of the joint density is bounded, the corresponding $(\widehat{G}_h, \widehat{\boldsymbol{\beta}}_h)$ can be solved using the maximum likelihood. Thus, our method is compatible for density estimation in various scenarios, and it has potential to be more flexible in the future.

Acknowledgements The authors thank the editor and an anonymous reviewer, whose comments led to many significant improvements in the manuscript.

Author Contributions YW contributed the conception and major ideas and guided the research throughout. DX wrote the initial manuscript text, prepared figures and carried out numerical studies. Both authors revised the manuscript.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions

Declarations

Competing interest Authors declare that there exist no competing financial or non-financial interests that are directly or indirectly related to the work submitted for publication.

Appendix

Consider a univariate von Mises density. To compute the maximum likelihood estimator $\hat{\kappa}$ of its concentration parameter, we need to incorporate a unique measurement of concentration for directional data, the mean resultant length, denoted by

$$R = \sqrt{C^2 + S^2},$$

where

$$C = \mathbb{E}[\cos(X)], \quad S = \mathbb{E}[\sin(X)]. \tag{12}$$

In particular, R is close to zero if observations disperse widely around the circle, whereas $R = 1$ for a point mass. Mardia and Jupp (2000, page 85) showed that $\hat{\kappa}$ is the solution of

$$A(\hat{\kappa}) = \frac{I_1(\hat{\kappa})}{I_0(\hat{\kappa})} = R,$$

or equivalently we have

$$\hat{\kappa} = A^{-1}(R).$$

As outlined in Sect. 3, for the EM algorithm of a product von Mises mixture density, denoting $\bar{R} = \sum_{j=1}^{m} \bar{R}_j$, we have

$$\bar{R} = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{m} p_{ij} \cos(x_{id} - \mu_{jd}).$$

To verify this relationship, firstly define

$$\bar{C}_j = \frac{1}{n}\sum_{i=1}^{n} p_{ij}\cos(x_i),$$
$$\bar{S}_j = \frac{1}{n}\sum_{i=1}^{n} p_{ij}\sin(x_i).$$

By Eq. (12), it is not hard to see that

$$\bar{C} = \frac{1}{n}\sum_{i=1}^{n}\cos(x_i) = \sum_{j=1}^{m}\bar{C}_j,$$

and similarly,

$$\bar{S} = \frac{1}{n}\sum_{i=1}^{n}\sin(x_i) = \sum_{j=1}^{m}\bar{S}_j.$$

Thus, for

$$\bar{R}_j = \sqrt{\bar{C}_j^2 + \bar{S}_j^2},$$

$\bar{x}_j$ is the solution to equations

$$\bar{C}_j = \bar{R}_j\cos(\bar{x}_j),\ \bar{S}_j = \bar{R}_j\sin(\bar{x}_j),$$

and then given Eq. (8), we have

$$\bar{x}_j = \operatorname{atan2}(\bar{S}_j, \bar{C}_j) = \mu_j.$$

Therefore,

$$\begin{aligned}
&\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}\cos(x_i - \mu_j)\\
&= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} p_{ij}\left[\cos(x_i)\cos(\mu_j) + \sin(x_i)\sin(\mu_j)\right]\\
&= \sum_{j=1}^{m}\cos(\mu_j)\bar{R}_j\cos(\bar{x}_j) + \sum_{j=1}^{m}\sin(\mu_j)\bar{R}_j\sin(\bar{x}_j)\\
&= \sum_{j=1}^{m}\bar{R}_j\cos(\mu_j - \bar{x}_j)\\
&= \bar{R}.
\end{aligned}$$

References

Abramowitz, M., Stegun, I.A.: Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables, vol. 55. US Government Printing Office, Washington, D.C (1964)

Berens, P.: CircStat: a MATLAB toolbox for circular statistics. J. Stat. Softw. **31**, 1–21 (2009)

Berman, H., Henrick, K., Nakamura, H., Markley, J.L.: The worldwide Protein Data Bank (wwPDB): Ensuring a single, uniform archive of PDB data. Nucleic Acids Res. **35**, D301–D303 (2006)

Boomsma, W., Mardia, K.V., Taylor, C.C., Ferkinghoff-Borg, J., Krogh, A., Hamelryck, T.: A generative, probabilistic model of local protein structure. Proc. Natl. Acad. Sci. **105**, 8932–8937 (2008)

Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)

Cavanaugh, J.E.: Unifying the derivations for the Akaike and corrected Akaike information criteria. Stat. Probab. Lett. **33**, 201–208 (1997)

Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. Roy. Stat. Soc.: Ser. B (Methodol.) **39**, 1–22 (1977)

Di Marzio, M., Panzera, A., Taylor, C.C.: Kernel density estimation on the torus. J. Stat. Plan. Inference **141**, 2156–2173 (2011)

Duarte, C.M., Pyle, A.M.: Stepping through an RNA structure: a novel approach to conformational analysis. J. Mol. Biol. **284**, 1465–1478 (1998)

Eltzner, B., Huckemann, S., Mardia, K.V.: Torus principal component analysis with applications to RNA structure. Ann. Appl. Stat. **12**, 1332–1359 (2018)

Fernandez-Duran, J.J., Gregorio-Dominguez, M.M.: CircNNTSR: an R package for the statistical analysis of circular, multivariate circular, and spherical data using nonnegative trigonometric sums. J. Stat. Softw. **70**, 1–19 (2016). https://doi.org/10.18637/jss.v070.i06

Fisher, N.I.: Statistical Analysis of Circular Data. Cambridge University Press, Cambridge (1995)

García-Portugués, E.: Exact risk improvement of bandwidth selectors for kernel density estimation with directional data. Electron. J. Stat. **7**, 1655–1685 (2013)

García-Portugués, E., Golden, M., Sørensen, M., Mardia, K.V., Hamelryck, T., Hein, J.: Toroidal diffusions and protein structure evolution. In: Applied Directional Statistics, pp. 17–40. Chapman and Hall/CRC (2018)

Geman, S., Hwang, C.R.: Nonparametric maximum likelihood estimation by the method of sieves. Ann. Stat. **10**, 401–414 (1982)

Grenander, U.: Abstract Inference. Wiley, New York (1981)

Hu, S., Wang, Y.: Modal clustering using semiparametric mixtures and mode flattening. Stat. Comput. **31**, 1–18 (2021)

Johnson, R.A., Wehrly, T.: Measures and models for angular correlation and angular-linear correlation. J. Roy. Stat. Soc.: Ser. B (Methodol.) **39**, 222–229 (1977)

Johnson, S.G.: The NLopt nonlinear-optimization package. https://github.com/stevengj/nlopt (2007)

Laird, N.: Nonparametric maximum likelihood estimation of a mixing distribution. J. Am. Stat. Assoc. **73**, 805–811 (1978)

Lawson, C.L., Hanson, R.J.: Solving Least Squares Problems. SIAM, Philadelphia (1995)

Lee, J.C., Gutell, R.R.: Diversity of base-pair conformations and their occurrence in rRNA structure and RNA structural motifs. J. Mol. Biol. **344**, 1225–1249 (2004)

Li, J., Ray, S., Lindsay, B.G.: A nonparametric statistical approach to clustering via mode identification. J. Mach. Learn. Res. **8**, 1687–1723 (2007)

Liljas, A., Liljas, L., Lindblom, G., Nissen, P., Kjeldgaard, M., Ash, M.R.: Textbook of Structural Biology, vol. 8. World Scientific, Singapore (2016)

Lindsay, B.G.: The geometry of mixture likelihoods: a general theory. Ann. Stat. **11**, 86–94 (1983)

Lindsay, B.G.: The geometry of mixture likelihoods, Part II: The exponential family. Ann. Stat. **11**, 783–792 (1983)

Lindsay, B.G.: Mixture models: theory, geometry and applications. In: NSF-CBMS Regional Conference Series in Probability and Statistics. Institute for Mathematical Statistics, Hayward (1995)

Mardia, K.V., Jupp, P.E.: Directional Statistics. John Wiley & Sons, New York (2000)

McLachlan, G.J., Krishnan, T.: The EM Algorithm and Extensions. Wiley, New York (1997)

McLachlan, G.J., Peel, D.: Finite Mixture Models. Wiley, New York (2000)

Nodehi, A., Golalizadeh, M., Maadooliat, M., Agostinelli, C.: Estimation of parameters in multivariate wrapped models for data on a p-torus. Comput. Stat. **36**, 193–215 (2021)

Oliveira, M., Crujeiras, R.M., Rodríguez-Casal, A.: A plug-in rule for bandwidth selection in circular density estimation. Comput. Stat. Data Anal. **56**, 3898–3908 (2012)

Powell, M.J.: A Direct Search Optimization Method that Models the Objective and Constraint Functions by Linear Interpolation, Advances in Optimization and Numerical Analysis, pp. 51–67. Springer, Berlin (1994)

R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2021)

Segura, J.: June. Best algebraic bounds for ratios of modified Bessel functions. arXiv preprint arXiv:2207.02713 (2022)

Sittel, F., Filk, T., Stock, G.: Principal component analysis on a torus: theory and application to protein dynamics. J. Chem. Phys. **147**, 244101 (2017)

Svanberg, K.: A class of globally convergent optimization methods based on conservative convex separable approximations. SIAM J. Optim. **12**, 555–573 (2002)

Taylor, C.C., Mardia, K.V., Di Marzio, M., Panzera, A.: Validating protein structure using kernel density estimates. J. Appl. Stat. **39**, 2379–2388 (2012)

Wadley, L.M., Keating, K.S., Duarte, C.M., Pyle, A.M.: Evaluating and learning from RNA Pseudotorsional space: quantitative validation of a reduced representation for RNA structure. J. Mol. Biol. **372**, 942–957 (2007)

Wang, X., Wang, Y.: Nonparametric multivariate density estimation using mixtures. Stat. Comput. **25**, 349–364 (2015)

Wang, Y.: On fast computation of the non-parametric maximum likelihood estimate of a mixing distribution. J. R. Stat. Soc. Ser. B (Stat. Methodol.) **69**, 185–198 (2007)

Wang, Y.: Maximum likelihood computation for fitting semiparametric mixture models. Stat. Comput. **20**, 75–86 (2010)

Wang, Y., Chee, C.S.: Density estimation using non-parametric and semi-parametric mixtures. Stat. Model. **12**, 67–92 (2012)

Xu, D., Wang, Y.: Cplots: plots for Circular Data. Department of Statistics, University of Auckland, New Zealand. R package version 0.4-0 (2019)

Xu, D., Wang, Y.: Area-proportional visualization for circular data. J. Comput. Graph. Stat. **29**, 351–357 (2020)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.