

CircSpaceTime: an R package for spatial and spatio-temporal modelling of circular data

Original

CircSpaceTime: an R package for spatial and spatio-temporal modelling of circular data / Jona Lasinio, Giovanna; Santoro, Mario; Mastrantonio, Gianluca. - In: JOURNAL OF STATISTICAL COMPUTATION AND SIMULATION. - ISSN 0094-9655. - 90:7(2020), pp. 1315-1345. [10.1080/00949655.2020.1725008]

Availability:

This version is available at: 11583/2837511 since: 2020-06-27T15:38:16Z

Publisher:

Taylor & Francis Group

Published

DOI:10.1080/00949655.2020.1725008

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

CircSpaceTime: an R package for spatial and spatio-temporal modeling of Circular data

Giovanna Jona Lasinio^a, Mario Santoro^b, Gianluca Mastrantonio^c

^aSapienza University of Rome, P.le Aldo Moro 5, 00185 Rome, ITALY; ^bIAC CNR, Via dei Taurini, 19, 00185 Rome, ITALY; ^cPolytechnic of Turin, ITALY, Corso Duca degli Abruzzi, 24, 10129 Turin, ITALY

Abstract

CircSpaceTime is the only R package currently available that implements Bayesian models for spatial and spatio-temporal interpolation of circular data. Such data are often found in applications where, among the many, wind directions, animal movement directions, and wave directions are involved. To analyze such data we need models for observations at locations \mathbf{s} and times t , as the so-called geostatistical models, providing structured dependence assumed to decay in distance and time. The approach we take begins with Gaussian processes defined for linear variables over space and time. Then, we use either wrapping or projection to obtain processes for circular data. The models are cast as hierarchical, with fitting and inference within a Bayesian framework. Altogether, this package implements work developed by a series of papers; the most relevant being Jona Lasinio, Gelfand, and Jona Lasinio (2012); Wang and Gelfand (2014); Mastrantonio, Jona Lasinio, and Gelfand (2016). All procedures are written using Rcpp. Estimates are obtained by MCMC allowing parallelized multiple chains run. The implementation of the proposed models is considerably improved on the simple routines adopted in the research papers. As original running examples, for the spatial and spatio-temporal settings, we use wind directions datasets over central Italy.

1 Introduction

In the last ten years the interest in circular data has received renewed attention, with new theoretical results and models (for a recent and extended review of both theory and applications see Ley and Verdebout, 2017, 2019). Most of the existing literature, before 2012, dealt with the development of univariate distributions on the circle, regression type-models including both circular and linear variable (fundamental textbooks are Mardia, 1972; Mardia and Jupp, 1999; Jammalamadaka and SenGupta, 2001). Coles (1998) addressed circular time series and briefly sketched possible extension to more complex dependence structures, such as spatial and spatio-temporal dependence. However, an under-investigated area in directional data was the development of space and space time processes, where directional observations are taken at spatial locations, usually over time as well, and proximity in space and time affects dependence between directions. Such data are typical of the world of geostatistical modeling (Banerjee, Gelfand, and Carlin, 2014), where structured dependence in space and time is introduced. Here, we review work developed by the authors together with other statisticians in a series of papers published over the last few years focusing on spatial and spatio-temporal modeling and we introduce a new formalisation of these models. We illustrate two case studies involving wind directions that have not been previously analyzed as spatial and/or spatio-temporal processes. In a purely spatial framework, we model wind direction in order to interpolate them in space on a finer grid than the one observed. While the aim of the space-time modeling exercise is the (forward) prediction of an

unobserved time point. All modeling efforts are cast as hierarchical models, with fitting and inference within a Bayesian inference framework. The hierarchical specification is vital; latent variables are introduced to facilitate passage from linear variables to circular. The Bayesian framework is arguably most attractive for this setting. We obtain full posterior inference including uncertainties, we avoid potentially inappropriate asymptotics, and we enable routine prediction (kriging). Furthermore, model fitting is straightforwardly implemented using Markov chain Monte Carlo (Banerjee, Gelfand, and Carlin, 2014). Specifically, Gibbs and Metropolis sampling provide loops, updating parameters given latent variables and data and, then, updating latent variables given parameters and data (Jona Lasinio, Gelfand, and Jona Lasinio, 2012). The main novelty of this work is the implementation of the models we are going to discuss, which has been highly improved on the original settings associated to the research papers Jona Lasinio, Gelfand, and Jona Lasinio (2012); Wang and Gelfand (2013, 2014); Mastrantonio, Jona Lasinio, and Gelfand (2016). In the package the Matern, Gaussian and exponential covariance functions are available for spatial models (only the exponential was explored in the research papers). All estimations are obtained through MCMC and here we implemented the algorithm shown in Andrieu and Thoms (2008), to speed up estimation of structured covariance functions. Further we are adding the possibility of running multiple chains in parallel using the `doParallel` package.

None of the existing packages for circular and directional data deal with spatial or spatio-temporal interpolation, that is the main objective of `CircSpaceTime` see Section 1.1 for a listing of such packages and a short discussion of their features). The package is currently available from CRAN (<https://CRAN.R-project.org/package=CircSpaceTime>).

There are different approaches to specify valid circular distributions, see for example Jammalamadaka and SenGupta (2001), here we focus on the two methods that allow to build a circular distribution starting from a linear one, namely the wrapping, and the projection. Both revealed to be useful in the definition of spatial and spatio-temporal data modeling, see for example Mastrantonio, Gelfand, and Jona Lasinio (2016) and Wang and Gelfand (2014). Under both methods, the resulting distribution has a complex functional form but introducing suitable latent variables, the joint distribution of observed and latent variables, in a fully Bayesian framework, is really easy to handle.

1.1 Existing software for circular and directional data

There exist several software libraries dealing with circular data, most of them are developed under the R project; in this framework the best known are `circular` (Agostinelli and Lund, 2017) and `CircStats` (Lund and Agostinelli, 2018), both implementing inference for univariate data as described in Jammalamadaka and SenGupta (2001). Another recent set of functions specifically devoted to wrapped distributions is `Wrapped` (Nadarajah and Zhang, 2017). The package computes the probability density function, cumulative distribution function, quantile function and many more features for several (about fifty) univariate wrapped distributions. A very interesting set of functions is implemented in `CircSizer` (Oliveira, Crujeiras, and Rodríguez-Casal, 2014) where a non-parametric approach is adopted. Based on scale-space ideas, `CircSiZer` presents a graphical device to assess which observed features are statistically significant, both for density and regression analysis.

In the likelihood and univariate framework, we find a good book on circular data in R (Pewsey, Neuhauser, and Ruxton, 2013) where many nice examples can be found and the narrative of the topic makes easy to learn how to run inferences on univariate data. In 2013 the first version of the package `isocirc` was presented (Barragán et al., 2013), making available functions to perform constrained inference using isotonic regression (Rueda, Fernández, and Peddada, 2009; Fernández, Rueda, and Peddada, 2011). The `CircOutlier` (Ghazanfarihesari and Mashhad, 2016) collects functions to detect outliers in circular-circular regression as proposed in Abuzaid, Hussin, and Mohamed (2013). More recent is the `Directional` package (Tsagris et al., 2018), mostly linked to the textbook by Mardia and Jupp (1999). A series of wrapper functions to implement the 10 maximum likelihood models of animal orientation, described by Schnute and Groot (1992), are included in the `CircMLE`. The proposals in Zimmermann and Wright (2017) are presented in `circumplex`.

Bayesian estimation for univariate regression models is implemented in `bpnreg` (Cremers, 2018)

that presents models developed in Nuñez-Antonio and Gutiérrez-Peña (2014) and Cremers, Mulder, and Klugkist (2017). Again in the Bayesian framework the work in Mulder and Klugkist (2017) is implemented in `circglmbayes`.

Dependent and multivariate circular data are often found in applications (see Ley and Verdebout, 2019, for recent developments). To handle them in a likelihood framework we can refer to the package `CircNNTSR` (Fernández-Duran and Gregorio-Domínguez, 2016), that implements functions to plot, fit by maximum likelihood, and simulate models based on non-negative trigonometric sums for circular, multivariate circular, and spherical data.

Outside the R project we find commercial softwares too, such as Oriana¹ for Windows performing basic statistics, testing and graphics or NCSS, LLC Statistical Software that has a section devoted to circular data, always performing simple basic statistics. In the Matlab environment we find the Circular Statistics Toolbox (Directional Statistics) (Berens, 2009) again performing descriptive statistics, basic testing (one sample and multiple samples) and graphics.

There are no packages or software performing spatial interpolation nor space-time prediction for directional or circular data.

2 From Linear to Circular models

We introduce, in both univariate and multivariate settings, the two approaches that we exploit to built distributions/processes for circular variables, from probability distributions with support the real line; we start with *linear random variables* to obtain *circular random variables*. The two methods are totally general and can be applied to linear variables with any distribution. We focus on the Gaussian case, since the resulting models are highly flexible and they are those implemented in `CircSpaceTime`.

2.1 The wrapping approach

With the wrapping approach, the idea is to “wrap” the density of a linear variable around the unit circle, obtaining than a circular density. In more details, let $Y \in \mathbb{R}$ be a linear random variable with pdf $f_Y(\cdot|\boldsymbol{\psi})$, where $\boldsymbol{\psi}$ is a vector of parameters. We obtain a circular random variable using the following transformation:

$$\Theta = Y \bmod 2\pi \in [0, 2\pi).$$

The pdf of Θ is then

$$f_{\Theta}(\theta|\boldsymbol{\psi}) = \sum_{k=-\infty}^{\infty} f_Y(\theta + 2\pi k|\boldsymbol{\psi}). \quad (1)$$

It is easy to find the relation between Y and Θ that is the following: $Y = \Theta + 2\pi K$, where K is the so-called *winding number*. Equation (1) wraps the density $f_Y(\cdot|\boldsymbol{\psi})$ around the unit circle and then Θ is called the *wrapped* version of \mathbf{Y} of period 2π . If Y is normally distributed, then Θ follows a *wrapped normal* (WN) distribution.

The evaluation of (1) is not easy since it involves an infinite sum, possibly, making inference based on the wrapped distribution a really complex task. Following Coles and Casson (1998), this problem can be easily overcome if we consider K as a (latent) random variable. We have then $f_{\Theta,K}(\theta, k|\boldsymbol{\psi}) = f_Y(\theta + 2\pi k|\boldsymbol{\psi})$, i.e. the joint density of (Θ, K) is $f_Y(\theta + 2\pi k|\boldsymbol{\psi})$, that is the density Y , seen as function of θ and k . Notice that (1) can be seen as a marginalization over K of the joint density of (Θ, K) . The conditional distribution of K , that is needed for the implementation of the MCMC, is easy to handle since it is proportional to $f_Y(\theta + 2\pi k|\boldsymbol{\psi})$. This result shows that it is much easier to work with the joint density of (Θ, K) , with respect to the marginal of Θ .

The wrapping approach can be easily extended to a multivariate setting, see for example Jona Lasinio, Gelfand, and Jona Lasinio (2012). In details, let $\mathbf{Y} = (Y_1, \dots, Y_n)'$ be a n -variate vector with pdf

¹<https://www.kovcomp.co.uk/index.html>

$f_{\mathbf{Y}}(\cdot|\boldsymbol{\psi})$. We have then that $\boldsymbol{\Theta} = (\Theta_1, \dots, \Theta_n)'$, with

$$\Theta_i = Y_i \bmod 2\pi, i = 1, \dots, n, \quad (2)$$

is a vector of circular variables with density

$$f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}|\boldsymbol{\psi}) = \sum_{k_1=-\infty}^{\infty} \cdots \sum_{k_n=-\infty}^{\infty} f_{\mathbf{Y}}(\boldsymbol{\theta} + 2\pi\mathbf{k}|\boldsymbol{\psi}).$$

Extending the reasoning applied to the univariate setting, we can easily find that the full conditional of \mathbf{K} is proportional to $f_{\mathbf{Y}}(\boldsymbol{\theta} + 2\pi\mathbf{k}|\boldsymbol{\psi})$ and the joint density of $(\boldsymbol{\Theta}, \mathbf{K})$ is $f_{\mathbf{Y}}(\boldsymbol{\theta} + 2\pi\mathbf{k}|\boldsymbol{\psi})$. If we assume that \mathbf{Y} is normally distributed, this means that the joint density of $(\boldsymbol{\Theta}, \mathbf{K})$ is

$$\phi_n(\boldsymbol{\theta} + 2\pi\mathbf{k}|\boldsymbol{\mu}, \boldsymbol{\Lambda}), \quad (3)$$

where $\phi_n(\cdot|\boldsymbol{\mu}, \boldsymbol{\Lambda})$ is the n -variate normal pdf with mean vector and covariance matrix given by $(\boldsymbol{\mu}, \boldsymbol{\Lambda})$. Here again it is easy to see that any type of model implementation is facilitated if \mathbf{K} is introduced as a latent variable.

2.2 The projection approach

With the wrapping approach, from one linear variable we obtain a circular one. The projection approach is quite different in this regard since it requires two linear variables to obtain one circular. The basic idea is to take the two variables, that can be seen as coordinates in a Euclidean space, and express them as polar coordinates. The associated angle is itself a random variable and it is circular.

Let $\mathbf{Y} = (Y_1, Y_2)$ be a bivariate vector of linear variables with pdf $f_{\mathbf{Y}}(\cdot|\boldsymbol{\psi})$. The relation between the angle Θ and \mathbf{Y} is expressed by the following:

$$\tan(\Theta) = \frac{Y_2}{Y_1}. \quad (4)$$

The circular variable can be then obtained using the inverse tangent function but some care must be taken. The inverse tangent has domain of length π while we want our random variable to have domain $[0, 2\pi)$. This is generally achieved by using the `atan*` inverse (see Jammalamadaka and SenGupta, 2001, pag.13 for a detailed definition), that takes into account the sign of the components of \mathbf{Y} , to identify to which of the 4 quadrants the circular variable belongs. Between Θ and \mathbf{Y} the following relation exists $\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} = R \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} = R\mathbf{U}$, with $R = \|\mathbf{Y}\|$, where the joint vector $(\Theta, R)'$ is the representation in polar coordinates of \mathbf{Y} . From equation (4) is easy to see that distributions based on the projection are not identifiable, since $c\mathbf{Y}$ and \mathbf{Y} , with $c > 0$, gives rise to the same circular variables; an identification constraint is then needed (Wang and Gelfand, 2013).

The pdf of Θ is

$$f_{\Theta}(\theta|\boldsymbol{\psi}) = \int_{\mathbb{R}^+} r f_{\mathbf{Y}}((r \cos(\theta), r \sin(\theta))'|\boldsymbol{\psi}) dr, \quad (5)$$

that is obtained by finding first the joint density of (Θ, R) , which is

$$r f_{\mathbf{Y}}((r \cos(\theta), r \sin(\theta))'|\boldsymbol{\psi}),$$

and then, a marginalization over R , gives the density of Θ . The integral in equation (5) is not easy to solve and, even when a closed form exists, the resulting pdf has a complicated functional structure. For example, if we assume $\mathbf{Y} \sim N_2(\boldsymbol{\alpha}, \boldsymbol{\Xi})$, with $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)'$ and

$$\boldsymbol{\Xi} = \begin{pmatrix} \sigma^2 & \tau\sigma \\ \tau\sigma & 1 \end{pmatrix}, \quad (6)$$

where $(\Xi)_{2,2} = 1$ is needed for identifiability purpose, the density of Θ , computed using (5), is

$$f_{\Theta}(\theta|\psi) = \frac{\phi_2(\boldsymbol{\alpha}|\mathbf{0}_2, \Xi) + aD(\theta)\Phi(D(\theta)|\mathbf{0}, \mathbf{I}_2)\phi(aC(\theta)^{-\frac{1}{2}}(\alpha_1 \sin(\theta)) - \alpha_2 \cos(\theta))}{C(\theta)}, \quad (7)$$

where $\Phi_n(\cdot|\cdot, \cdot)$ is the n -variate cumulative Gaussian density function, with

$$\begin{aligned} a &= \left(\sigma\sqrt{1-\tau^2}\right)^{-1}, \\ C(\theta) &= a^2 (\cos^2(\theta) + \sigma^2 \sin^2(\theta) - \tau\sigma \sin(2\theta)), \\ D(\theta) &= a^2 C(\theta)^{-\frac{1}{2}} (\alpha_1(\cos(\theta) - \tau\sigma \sin(\theta)) + \alpha_2\sigma(\sigma \sin(\theta) - \rho \cos(\theta))). \end{aligned}$$

Under the normal assumption, we have that the joint density of (Θ, R) is

$$r\phi_2((r \cos(\theta), r \sin(\theta))|\boldsymbol{\alpha}, \Xi). \quad (8)$$

Equation (8) is less complex than (7) but, as for the wrapping approach, to work with it we need to introduce a latent variable, that is R .

The extension of the projection approach to multivariate variable is straightforward. We take \mathbf{Y} as a $2n$ -variate linear variable, and we build the n -variate vector of (projected) circular variables through the following transformation:

$$\Theta_i = \text{atan}^*\left(\frac{Y_{2i}}{Y_{2i-1}}\right), i = 1, \dots, n, \quad (9)$$

i.e., the $2n$ elements of \mathbf{Y} are grouped in n sets, each of them containing only two values, and where one element of \mathbf{Y} can only belong to one of the n sets. Each set gives rise to a circular variable. The density of the vector $\boldsymbol{\Theta} = (\Theta_1, \dots, \Theta_p)'$ is then

$$f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}|\psi) = \int_{\mathbb{R}^+} \dots \int_{\mathbb{R}^+} \prod_{i=1}^n r_i f_{\mathbf{Y}}(\mathbf{y}(\boldsymbol{\theta}, \mathbf{r})|\psi) dr_1 \dots dr_n, \quad (10)$$

where $r_i = \|(y_{2i-1}, y_{2i})'\|$ and $\mathbf{y}(\boldsymbol{\theta}, \mathbf{r})$ is used to indicated that \mathbf{y} must be seen as function of $\boldsymbol{\theta}$ and $\mathbf{r} = (r_1, \dots, r_n)$. Even assuming a normal density for \mathbf{Y} , it is not possible to write in closed form (10). As for the univariate case, it is much easier to work with the joint density of $(\boldsymbol{\Theta}, \mathbf{R})$ since, if for example a normal distribution of \mathbf{Y} is assumed, it is then

$$\prod_{i=1}^n r_i \phi_{2n}(\mathbf{y}(\boldsymbol{\theta}, \mathbf{r})|\boldsymbol{\mu}, \boldsymbol{\Lambda}), \quad (11)$$

where, as in equation (3), $(\boldsymbol{\mu}, \boldsymbol{\Lambda})$ are the mean and covariance matrix.

2.2.1 Differences between the two approaches

There are considerable differences between the two approaches discussed above. The wrapping creates circular distributions that are in general similar to their counter-parts on the real line. For example, the wrapped normal is still unimodal and symmetric. Moreover, the circular mean and variance (see Jammalamadaka and SenGupta, 2001, for the definitions and details), are simple functions of the mean and variance of the linear variable.

On the other hand, even under the normal assumption, the projection approach creates a distribution that often shows very different properties from its linear partners. Again in the Gaussian setting, the projected normal can be bimodal, asymmetric and antipodal. Moreover, only in few special cases, exists a closed form expression of the mean and variance of the projected distribution.

The main reason to propose these two approaches is that, in both, it is easy to introduce dependence (spatial, temporal or both). The wrapping gives results that are really easy to interpret in terms of phenomena behaviour, while the projection is very useful when interpretation is not central, and a highly flexible model is required (Mastrantonio, Jona Lasinio, and Gelfand, 2016).

Name	Function	Parameters
Matérn	$\frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{h_{\text{sp}}}{\rho} \right)^\nu K_\nu \left(\sqrt{2\nu} \frac{h_{\text{sp}}}{\rho} \right)$	ν, ρ
Exponential	$\exp(-\rho h_{\text{sp}})$	ρ
Gaussian	$\exp(-\rho^2 h_{\text{sp}})$	ρ
Gneiting (spatio-temporal)	$\frac{1}{\rho_t h_t^2 + 1} \exp \left(-\frac{\rho_{sp} h_{\text{sp}}}{(\rho_t h_t^2 + 1)^{\frac{\eta}{2}}} \right)$	ρ_t, ρ_{sp}, η

Table 1: Correlation functions implemented. h_{sp} and h_t are, respectively, spatial and temporal distance.

2.3 Spatio-temporal processes for circular variables

A stochastic process can be defined through its finite-dimensional distribution, i.e. the distribution of an n -dimensional realization, that has a multivariate pdf (Gelfand et al., 2010). Since in the previous sections we already described multivariate distributions for circular variables, it is then easy to define circular processes. More precisely, starting from a distribution for linear variables, we can use the wrapping or the projection approach to obtain a multivariate circular distribution and then, as a consequence, from an n -dimensional realization of a linear process we can obtain the associated n -dimensional realization of a circular one.

In particular, in the Gaussian cas, let $\mathbf{Y}(\mathbf{s}) = \{Y_i(\mathbf{s})\}_{i=1}^p \in \mathbb{R}^p$ be a p -variate Gaussian process (GP) defined over a s -dimensional domain, i.e., $\mathbf{s} \in \mathcal{S} \subset \mathbb{R}^d$, and let $\mathbf{y} \equiv (\mathbf{y}_1, \dots, \mathbf{y}_n)' \in \mathbb{R}^p \times \mathbb{R}^n$ be the set of n realizations. The vector \mathbf{y} is then normally distributed. Given $\mathbf{Y}(\mathbf{s})$ we can easily build the wrapped and the projected GP using (2) and (9), respectively. More precisely, the former is obtained as

$$\Theta(\mathbf{b}) = \mathbf{Y}(\mathbf{s}) \bmod 2\pi.$$

while the latter is given by

$$\Theta(\mathbf{s}) = \text{atan}^* \left(\frac{Y_2(\mathbf{s})}{Y_1(\mathbf{s})} \right).$$

The same transformation, applied to \mathbf{y} , give the realization of the wrapped and projected GP.

For both, the spatial or space-time representations are obtained by considering a spatially or spatio-temporally structured covariance matrix.

3 The implemented models

The basic linear model we adopt is the following:

$$\begin{aligned} \mathbf{Y}(\mathbf{s}) &= \boldsymbol{\alpha} + \boldsymbol{\omega}(\mathbf{s}), \\ \boldsymbol{\omega}(\mathbf{s}) &\sim GP_p(\mathbf{0}_p, C(\mathbf{h}; \boldsymbol{\varphi}) \otimes \boldsymbol{\Xi}), \end{aligned} \tag{12}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^p$ is a mean vector, $\boldsymbol{\omega}(\mathbf{s})$ is a zero mean p -variate GP and \otimes is the usual Kronecker product. The correlation function of the GP is $C(\mathbf{h}; \boldsymbol{\varphi})$ and it depends on parameter $\boldsymbol{\varphi}$ and the vector of distances \mathbf{h} , that contains the spatial and temporal distances h_{sp} and h_t if $d = 3$, and only h_{sp} if $d = 2$. Under $p = 1$ the model (12) is used to build a wrapped GP while $p = 2$ is used for the projected. Table 1 shows the correlation functions and Table 2 the choice of prior distributions available in the package CircSpaceTime. The parameter $\boldsymbol{\Xi}$ is defined as in equation (6) if $p = 2$, and coincide with σ^2 if $p = 1$.

Parameter	Wrapped Normal	Projected Normal
Spatial and temporal decay (ρ, ρ_{sp}, ρ_t)	uniform	uniform
σ^2	inverse gamma	inverse gamma
Separability parameter ($\eta = \text{sep_par}$)	Beta	Beta
mean parameters (α)	Wrapped Gaussian	Gaussian
correlation between components (τ)		uniform

Table 2: Available prior distributions

3.1 Model implementation

As we stated in Section 2, the multivariate wrapped and projected normal have densities too complex to be used for model inference but, if we introduce as latent observation the vector \mathbf{k} , for the wrapped normal, and \mathbf{r} for the projected, inference through MCMC algorithm is straightforward.

The key element here is that, once the latent variables are obtained, the full conditionals of the model parameters are exactly the same as if we were working with observed linear variables. This is due to the functional form of the likelihood of circular and latent variables, see (3) and (11), which gives the same contribution to the full conditionals of the parameters, as if the data has a multivariate normal density.

Due to the priors available in the package, see Table 2, the update of α , i.e. the mean parameter, is obtained with a Gibbs sample since its full conditional is multivariate normal under the projected model and wrapped normal under the wrapped model. The parameters of the covariance structure are updated all together, within a Metropolis step. To speed up convergence and to have the possibility to choose the final acceptance rate, for the multivariate proposal distribution we implemented an algorithm shown in (Andrieu and Thoms, 2008, algorithm 4). In details, let $\mathbf{x}_{ad}^b \in \mathbb{R}^{n_{ad}}$ be a n_{ad} -dimensional vector, drawn at the b -th iteration of the MCMC algorithm, from a multivariate normal with mean μ_{ad}^b and covariance matrix $\lambda_{ad}^b \Sigma_{ad}^b + 0.0001 \mathbf{I}_{n_{ad}}$, where n_{ad} is the number of parameters of the model covariance matrix. Each element of \mathbf{x}_{ad}^b is used to propose one of the covariance parameters using suitable transformation: $\exp(x_{ad,j})$ for the parameters belonging to \mathbb{R}^+ and $(a + b \exp(x_{ad,j})) / (1 + \exp(x_{ad,j}))$ for the ones defined over $[a, b]$. Parameters μ_{ad}^b , λ_{ad}^b and Σ_{ad}^b are updated to guarantee selected acceptance-ratio d_{ad} , via the following:

$$\begin{aligned}\lambda_{ad}^{b+1} &= \exp \left(\log \lambda_{ad}^b + \frac{1}{b \xi_{ad}} (\alpha_{mh} - d_{ad}) \right) \\ \mu_{ad}^{b+1} &= \mu_{ad}^b + \frac{1}{b \xi_{ad}} (\mathbf{x}_{ad}^* - \mu_{ad}^b) \\ \Sigma_{ad}^{b+1} &= \Sigma_{ad}^b + \frac{1}{b \xi_{ad}} \left((\mathbf{x}_{ad}^* - \mu_{ad}^b)(\mathbf{x}_{ad}^* - \mu_{ad}^b)' - \Sigma_{ad}^b \right)\end{aligned}$$

where α_{mh} is the Metropolis-Hastings ratio, $\xi_{ad} \in (0, 1)$ is a parameter that allows to choose how much the last iteration influences the next one; \mathbf{x}_{ad}^* is equal to \mathbf{x}_{ad}^b if the proposed vector is accepted otherwise the previous iteration value (\mathbf{x}_{ad}^{b-1}) is kept. Notice that the adaptation decreases with the iterations while the larger the parameters ξ_{ad} , the weaker is the action of adaptation. The diagonal element $0.0001 \mathbf{I}_{n_{ad}}$ is needed to always be able to invert the covariance matrix avoiding ill defined matrices. More details on how and which parameters can be selected are given in the next section.

The latent discrete variable \mathbf{k} is updated component-wise with Metropolis steps. In each one of them, given the current value k_i , the proposal is selected from the set $(k_i - 1, k_i, k_i + 1)$, where each element has probability $1/3$ to be selected. The latent variable \mathbf{r} is also updated component-wise with Metropolis steps, using the algorithm proposed by Rosenthal (2007). In this case, at the b -th iteration we propose a new value of r_i from a log-normal with mean given by $\log r_i^{b-1}$ and standard deviation

$sd_{ad,i}^b$. Every n_{batch} iterations, $sd_{ad,i}^b$ is updated using the following:

$$sd_{ad,i}^{b+1} = \exp \left(\log sd_{ad,i}^b + \frac{1}{b\xi_{ad}} (\bar{\alpha}_{mh} - d_{ad}) \right),$$

otherwise $sd_{ad,i}^{b+1} = sd_{ad,i}^b$, where $\bar{\alpha}_{mh}$ is the mean Metropolis-Hastings ratio over the past n_{batch} iterations.

In fitting the wrapped Gaussian models we center data (and all related quantities) around π . Outputs for the mean and the interpolated values are given on the original scale, while the winding numbers are those obtained centering around π .

To improve performances we implemented everything in C++ (Wickham, 2015) and using Rcpp package we simplify the integration between C++ and R codes (Eddelbuettel and Francois, 2011). In particular we used the RcppArmadillo package (Eddelbuettel and Sanderson, 2014), that implements the Armadillo matrix library, for its simplicity and elegance (Eddelbuettel, 2017), although the RcppEigen is a bit faster (Bates and Eddelbuettel, 2013). For a fast multiple chain estimations we use doParallel package (Microsoft Corporation and Weston, 2017).

4 Two case study based on wind directions

In this section we illustrate all the proposed models through real data examples. The code associated to the examples is available as supplementary material.

4.1 Wind data in Central Italy

We consider examples based on wind data downloaded from the Global Forecast System (GFS) of the USA's National Weather Service (NWS) (<https://www.ncdc.noaa.gov/data-access/model-data/model-datasets/global-forecast-system-gfs>). Wind data are taken from NOAA/NCEP Global Forecast System (GFS) Atmospheric Model collection. Geospatial resolution is 0.5 degrees (approximately 50 km), and the wind is calculated for Earth surface, at 10 m. The function `wind.dl` from `rWind` library (Fernández-López and Schliep, 2018) facilitate the download of the data.

Downloaded data are arranged into dataframes with the following information:

- the time point expressed as date-time (`time`)
- the coordinates as decimal longitude and latitude (`lat`, `lon`)
- the eastward and northward wind velocity components at 10m a.s.l. (m/s) (`ugrd10m`, `vgrd10m`)
- the direction of the wind in degrees (`dir`).
- the wind velocity in m/s (`speed`).

Directions are transformed to radians for modeling purposes and rotated of 180 degrees to get values in accordance with the usual wind rose.

We select two days: the 29th of June 2012 and the 29th of October 2018. The first date was a lightly breezy sunny day over the entire central Italy. Light breeze in the area implies a large variability in terms of wind directions. The second date saw one of the worst storms in the last century hitting Europe. Central Italy was strongly affected by strong winds and loss of lives. In this meteorological condition winds are usually coming from one direction and the variability is very low.

We are interested in illustrating different modeling strategies dealing with very different data variability. In the spatial setting, the aim of the study is to provide directions on a grid with finer resolution (about 17×18 km) than the about $50 \text{ km} \times 50 \text{ km}$ grid cells (see Figure 5) of the observations (56 observed points, 400 prediction points). From June 29, 2012 we download data 3 hours apart, starting

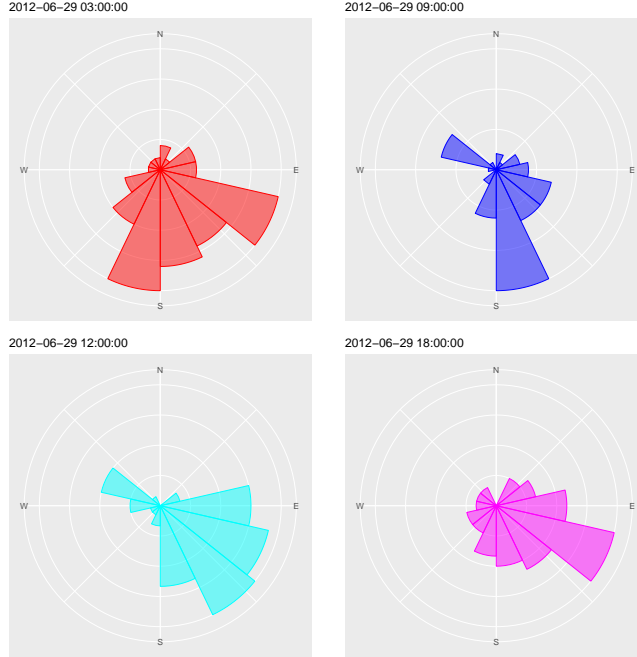


Figure 1: Rose diagram of wind directions of June 29, 2012 at 4 time points

at midnight and we choose the most variable time point for spatial interpolation. On October 29, 2018 we again download data 3 hours apart (from 3pm until 11pm), we choose the 23:00 (11pm) time point when the peak of the storm hit the area, for spatial interpolation and we save the remaining for the spatio-temporal analysis.

In Figure 1 we explore some of the time points on June 29, 2012. We choose the time slice at 03 : 00 as highly variable and almost bimodal. We compare June 29, 2012 and October 29, 2018 variability in Figure 2, the circular variance for the first one is 0.48 and for the second dataset is 0.048, ten times smaller. In Figure 3 we illustrate the observed wind directions together with the wind speed surface.

In what follows our strategy is to estimate and compare prediction from both wrapped and projected normal models. Comparison is carried out by computing the Average Prediction Error (APE) (see Jona Lasinio, Gelfand, and Jona Lasinio, 2012) and the circular continuous ranked probability score (CRPS) (Grimmett et al., 2006).

4.2 The 29th of June data: spatial interpolation of highly variable data

We start estimating the wrapped normal posterior distribution for the June 29 data. Notice that distances are computed using the Euclidean distance and hence it is suggested to convert coordinates into UTM system. Computations requires the definition of starting values and prior distributions for the parameters. The latter are:

- $\alpha \sim WN(\pi, 10)$ (weakly informative).
- $\rho \sim U(0.0068, 0.0736)$ the interval is chosen by computing practical ranges using the minimum and maximum distances in the data and use them as interval's extremes.
- $\sigma^2 \sim IG(3, 0.5)$ (informative prior).

The covariance is exponential. Notice that we performed some sensitivity analysis to prior definition. The wrapped model is not very sensitive, a weakly informative prior on σ^2 will simply require a larger iterations number to achieve convergence. The MCMC computation in these examples runs with 2

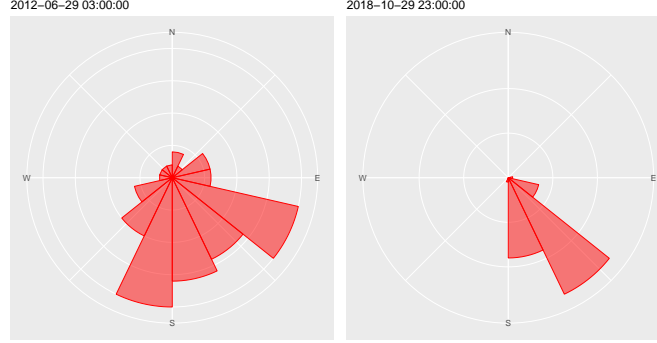


Figure 2: Rose diagram of wind directions of June 29, 2012 at 3am and October 29, 2018 at 23:00 (11pm)

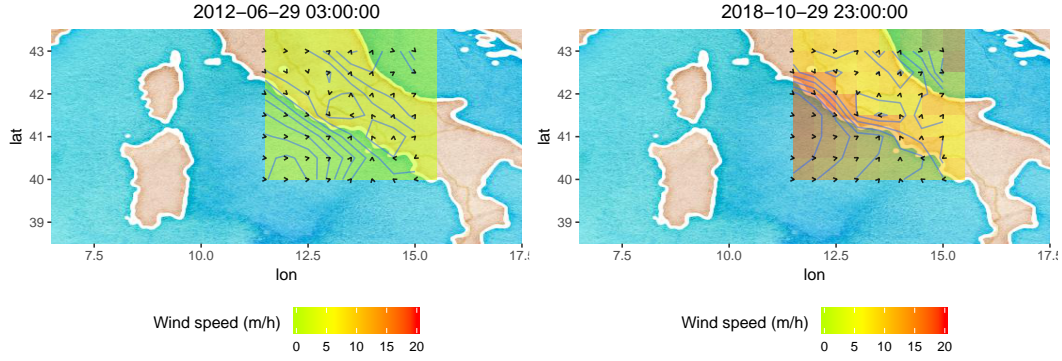


Figure 3: Central Italy: wind directions on June 29, 2012 at 3am and October 29, 2018 at 11pm, the color scale follows the speed of the wind from green to red

chains in parallel allowing the use of several convergence checks already available in R (package coda). In particular we compute the Potential scale reduction factors (\hat{R}) of the model parameters (Gelman and Rubin, 1992; Brooks and Gelman, 1998).

An important practical point is when to start and end the adaptive procedure of the Metropolis steps, we can use the command `start` and `end` in the list `adapt_param`; `end` should be smaller than `burnin` to ensure that saved samples are drawn from the desired posterior distribution. The parameter ξ_{ad} is set using the command `exp`, the desired acceptance-ratio d_{ad} is `accept_ratio` while we assume Σ_{ad}^0 to be a diagonal matrix with diagonal elements given by the `sd_prop`. We set 60000 iterations, 30000 of burnin and a thinning of 10. The adaptation starts at the 100th iterations and it ends at iteration 10000.

Graphical checks are obtained using a specific function `ConvCheck` and must be read recalling that `alpha` is a circular variable. Convergence is confirmed by \hat{R} s values, all equals to 1. We can proceed to interpolate the values on the 20×20 cells grid reported in Figure 5. Once the predictions are obtained, we compute the APE and the circular continuous ranked probability score (CRPS). These measures are evaluated between data points and their nearest grid points. Both functions return the average index value, and a vector of scores computed on each site. Results are reported in Table 3.

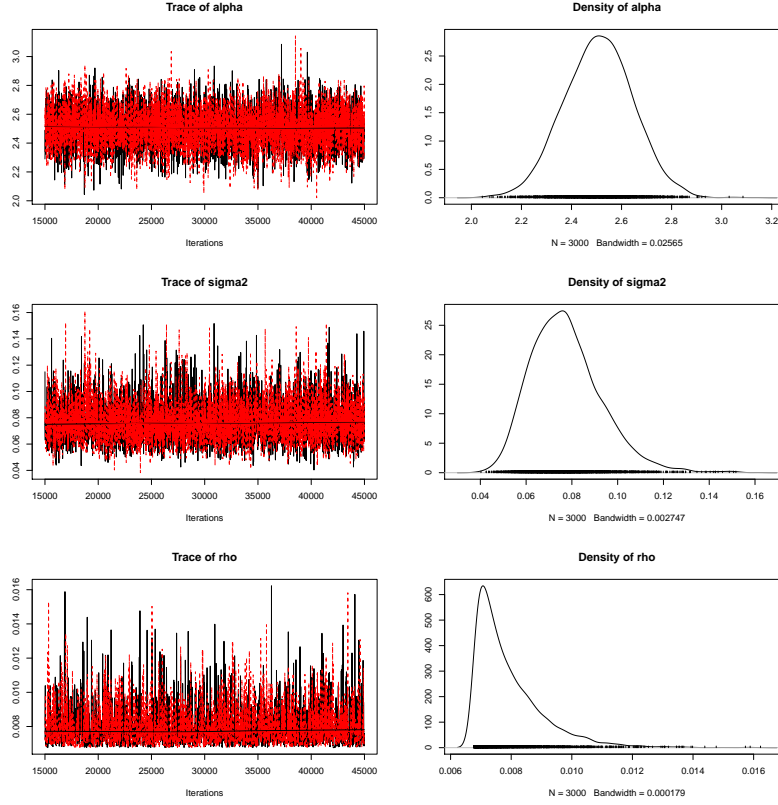


Figure 4: Traces and densities from the mcmc run for the wrapped Gaussian spatial model - June 29, 2012 data

Now we fit the projected normal model to the same data. Some care must be taken in setting prior distributions for the projected normal model. This model is more sensitive to the choice of priors for the decay parameter. We propose a uniform distribution on a fixed interval, as already mentioned, when the decay parameter suggested by the data is close to the extreme of the chosen interval we may obtain non-positive definite covariance matrices and the computation will stop.

- $\rho \sim U(0.0068, 0.0746)$
- $\alpha \sim N_2\left(\begin{pmatrix} \frac{\pi}{3} \\ \frac{2\pi}{3} \end{pmatrix}, \begin{pmatrix} 20 & 0 \\ 0 & 20 \end{pmatrix}\right)$
- $\tau \sim U(-1, 1)$
- $\sigma^2 \sim IG(3, 2)$

Again the spatial covariance is exponential. With the projected model we run our chains for a longer time (100000 iterations half of them are kept) as the convergence is slower. The adaptation algorithm differ slightly, as the standard deviation of the log-normal proposals for \mathbf{r} must be set.

From the convergence check (including the traceplots that are not reported) we see that the chains reached convergence. In case an update is required it can be easily done by setting as starting values the last values of all chains (see the Supplementary material). We can then move to the prediction step.

With the data from June 29, 2012 the projected normal flexibility allows a better fit of the model (see Table 3 as it is also shown in Figure 6). In Figure 7 the field representation of interpolated wind

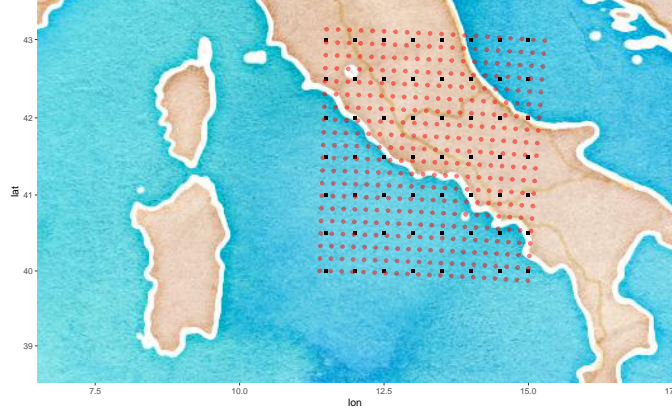


Figure 5: Interpolation grid (red) and data points (black)

directions are shown together with the observed values under both models, wrapped and projected, and it can be seen as the differences between the two models, however relevant in terms of APE and CRPS, are hardly visible.

4.3 Data from the 29th of October 2018: spatial interpolation of concentrated values

On October 29 and 30 2018, at least 11 people were killed and many injured after severe storms produced heavy rain and strong winds across Italy, causing widespread floods and damage. Some of the worst affected areas were on the west coast, particularly in Liguria. Many people were evacuated due to increased river levels across the country. Winds up to 180 km/h (112 mph) were reported on October 29, 2018, and blamed for the deaths of at least 6 of the 11 killed people. Trees were downed from the country's North down to Rome and further South to Naples. Winds were mostly from the South-East quadrant (Sirocco).

Storms usually implies very little variability in the wind direction, then a priori no model would be preferable. The flexible projected normal would most likely adapt well, but it will take some time and patience to get estimated values. The WN would most likely perform well as the distribution of the angle is unimodal and concentrated (see Figure 2).

We briefly summarize the setting and results. Prior distributions and MCMC settings were:

- **Wrapped Normal**

- $\alpha \sim N(\frac{2\pi}{3}, 10)$
- $\rho \sim U(0.0068, 0.0736)$
- $\sigma^2 \sim IG(2, 0.3)$ we can set an improper prior
- 60000 iterations with 30000 burn-in and thin of 10

- **Projected Normal**

- $\rho \sim U(0.0068, 0.0746)$
- $\tau \sim U(-0.5, 1)$ we are more informative to speed convergence
- $\sigma^2 \sim IG(6, 5)$



Figure 6: Interpolated values (red) and observed values (blue)

Date	Ape PN	Ape WN	CRPS PN	CRPS WN
29 June 2012	0.0499	0.0736	0.0076	0.012
29 October 2018	0.0043	0.0043	0.0008	0.0008

Table 3: Spatial data examples: model choice based on APE and CRPS

- $\alpha \sim N_2 \left(\begin{pmatrix} \frac{\pi}{6} \\ \frac{2\pi}{3} \end{pmatrix}, \begin{pmatrix} 20 & 0 \\ 0 & 20 \end{pmatrix} \right)$
- 100000 iterations with 50000 burn-in and thin of 10

- exponential correlation function
- 2 chains with parallel computation

The adaptive part is set as in the MCMC described for the June 29 data. Convergence was achieved and the interpolation computed on the grid, shown in Figure 5. The APE and CRPS returned the same values up to the fourth decimal Figure: 0.0043 APE and 0.0008 CRPS. The best model for this dataset can be the wrapped or the projected equivalently if the best fit is the main focus. Wrapped Normal model was computationally less demanding and allowed the choice of less informative priors, parameters can be easily interpreted. Hence it can be a preferred choice. In Figure 8 we compare graphically the two prediction it is clear that they are indistinguishable.

4.4 Spatio-temporal circular wrapped normal model

To illustrate the space-time version of the models implemented in CircSpaceTime we use data from the 29th of October 2018 from 15:00 to 21 every three hours and we predict the time point at 23:00 that we used in the spatial example. Again we compare the two modeling approaches. the model complexity increases as well as the computational time.

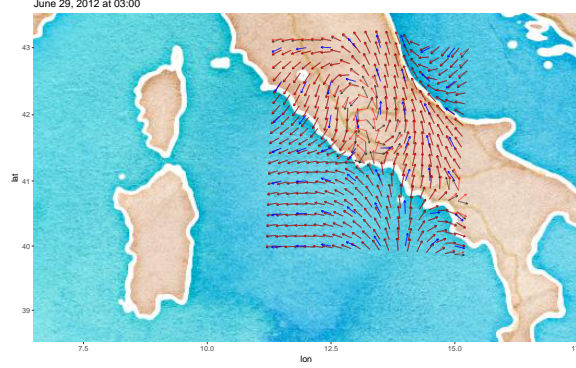


Figure 7: June 29,2012. Interpolated values under the wrapped normal model (red), projected normal model (black) together with the observed values (blue)

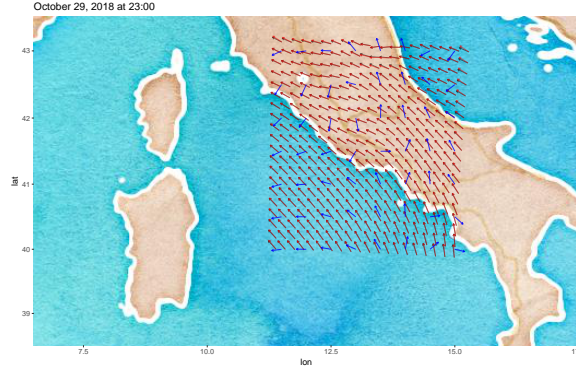


Figure 8: October 29, 2018. Interpolated values under the wrapped normal model (red), projected normal model (black) together with the observed values (blue)

As in the spatial case we set values for the spatial and temporal range priors

- Wrapped Normal spatio-temporal model's settings:
 - $\rho_{sp} \sim U(0.001, 0.100)$, $\rho_t \sim U(1, 3)$ the complexity of the model suggests less informative priors for single parameters and hence for the decay parameters
 - $\eta \sim Beta(1, 1)$
 - $\sigma^2 \sim IG(5, 5)$
 - $\alpha \sim WN(\frac{2\pi}{3}, 20)$
 - iterations 150000, of which 50000 burn-in and we keep one sample out of 10.
- Projected Normal spatio-temporal model's settings:
 - $\rho_{sp} \sim U(0.001, 0.574)$, $\rho_t \sim U(0.01, 3.5)$
 - $\eta \sim Beta(1, 1)$
 - $\sigma^2 \sim IG(5, 5)$
 - $\tau \sim U(-1, 1)$

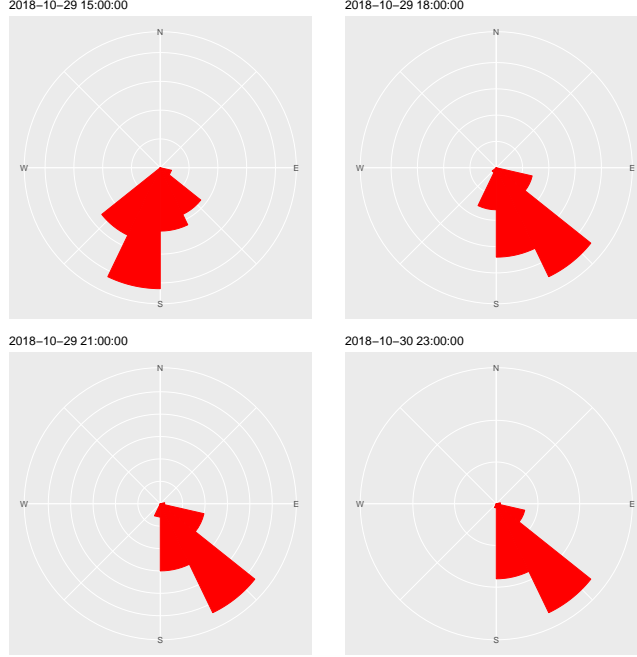


Figure 9: Time points from October 29, 2018. The first three are used to predict the 23:00 hour

- $\alpha \sim N_2 \left(\begin{pmatrix} \frac{2\pi}{3} \\ \pi \end{pmatrix}, \begin{pmatrix} 10 & 0 \\ 0 & 10 \end{pmatrix} \right)$
- a total of 300000 (two updates) with 150000 burnin and thinning of 10

The wrapped normal model convergence check returns a Multivariate Potential Scale reduction Factor (\hat{R}_M) of 1.01 after one run and all parameters chains seem to have reached convergence (not shown). We move to the prediction of the 23:00 hour. The Average prediction error on the predicted locations is 0.16 radians, ranging from 0.1435 to 0.2258, while the average CRPS is 0.0243, with point values ranging from 0.0103 to 0.0917.

While the projected normal model convergence is achieved with multivariate \hat{R}_M equal to 1.05. We move again to the prediction of the time point at 23:00 and compare the two models through APE and CRPS. The APE ranges between 0.028 and 0.133 with an average value of 0.068, CRPS ranges between 0.0005 and 0.034, with an average value of 0.008.

It is very clear that the projected normal captures way better the data features when prediction forward in time is required. In Figure 10 we can see that the wrapped normal model does not capture the second peak in the wind direction data.

In Figure 11 we can see point by point the behaviour of the models with respect to the observed values. In the northern section of the area, where a slightly larger variability is found, the wrapped normal model produces values further a part from the observed ones as the projected normal model.

5 Concluding Remarks

In this paper we presented the CircSpaceTime package, that collects functions implementing spatial and spatio-temporal Bayesian models allowing interpolation of dependent circular data. The implemented models have been introduced in a series of papers that are briefly summarized here using a new formalization (see Jona Lasinio, Gelfand, and Jona Lasinio, 2012; Wang and Gelfand, 2013, 2014; Mastrantonio, Jona Lasinio, and Gelfand, 2016, for details). The current version of the package does not include models with spatial nugget and allows for constant mean only. Future versions will include

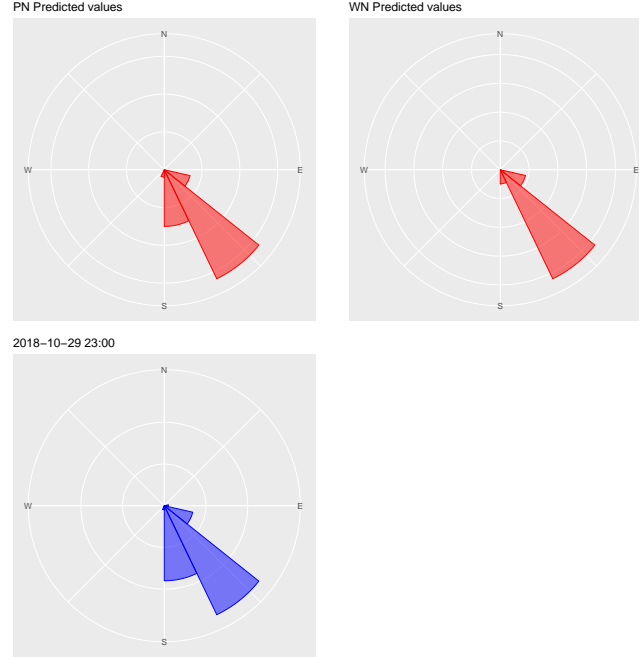


Figure 10: Predicted and observed values, 29th of October 2018 at 23:00

nugget and regression-type mean. We also plan to include more models for dependent circular data spreading from the works Bulla et al. (2012), Lagona and Picone (2012), Bulla et al. (2015), Lagona et al. (2015) and Maruotti et al. (2016), that develop likelihood-based estimation for complex modeling of circular and cylindrical data.

Acknowledgments

This work has been partially developed under the PRIN2015 supported-project *Environmental processes and human activities: capturing their interactions via statistical methods* (EPHASat) funded by MIUR (Italian Ministry of Education, University and Scientific Research) (20154X8K23-SH3). Gianluca Mastrantonio research has been partially supported by MIUR grant Dipartimenti di Eccellenza 2018 - 2022 (E11G18000350001).

References

- Abuzaid, A. H., A. G. Hussin, and I. B. Mohamed. 2013. “Detection of outliers in simple circular regression models using the mean circular error statistic.” *Journal of Statistical Computation and Simulation* 83 (2): 269–277. <https://doi.org/10.1080/00949655.2011.602679>.
- Agostinelli, Claudio, and Ulric Lund. 2017. *R package circular: Circular Statistics (version 0.4-93)*. CA: Department of Environmental Sciences, Informatics and Statistics, Ca’ Foscari University, Venice, Italy. UL: Department of Statistics, California Polytechnic State University, San Luis Obispo, California, USA. <https://r-forge.r-project.org/projects/circular/>.
- Andrieu, Christophe, and Johannes Thoms. 2008. “A tutorial on adaptive MCMC.” *Statistical computing* 18: 343–373.
- Banerjee, S., A. E. Gelfand, and B. P. Carlin. 2014. *Hierarchical Modeling and Analysis for Spatial Data*. 2nd ed. New York: Chapman and Hall/CRC.

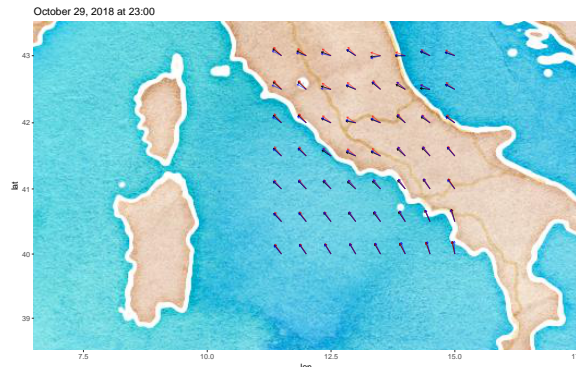


Figure 11: Predicted and observed values, 29th of October 2018 at 23:00. Wrapped normal predictions (in red), projected normal predictions (black) and observed values (blue)

- Barragán, Sandra, Miguel A. Fernández, Cristina Rueda, and Shyamal Das Peddada. 2013. “isocir: An R Package for Constrained Inference Using Isotonic Regression for Circular Data, with an Application to Cell Biology.” *Journal of Statistical Software* 54 (4): 1–17.
- Bates, Douglas, and Dirk Eddebuettel. 2013. “Fast and Elegant Numerical Linear Algebra Using the RcppEigen Package.” *Journal of Statistical Software, Articles* 52 (5): 1–24.
- Berens, Philipp. 2009. “CircStat: A Matlab Toolbox for Circular Statistics.” *Journal of Statistical Software* 31 (10). <http://www.jstatsoft.org/v31/i10>.
- Brooks, Stephen P., and Andrew Gelman. 1998. “General Methods for Monitoring Convergence of Iterative Simulations.” *Journal of Computational and Graphical Statistics* 7 (4): 434–455. <https://www.tandfonline.com/doi/abs/10.1080/10618600.1998.10474787>.
- Bulla, J., F. Lagona, A. Maruotti, and M. Picone. 2012. “A multivariate hidden Markov model for the identification of sea regimes from incomplete skewed and circular time series.” *Journal of Agricultural, Biological, and Environmental Statistics* 17 (4): 544–567.
- Bulla, Jan, Francesco Lagona, Antonello Maruotti, and Marco Picone. 2015. “Environmental conditions in semi-enclosed basins: A dynamic latent class approach for mixed-type multivariate variables.” *Journal de la Société Française de Statistique* 156 (1): 114–137.
- Coles, S. 1998. “Inference for Circular Distributions and Processes.” *Statistics and Computing* 8 (2): 105–113.
- Coles, Stuart, and Edward Casson. 1998. “Extreme value modelling of hurricane wind speeds.” *Structural Safety* 20 (3): 283 – 296.
- Cremers, Jolien. 2018. *bpnreg: Bayesian Projected Normal Regression Models for Circular Data*. R package version 1.0.0, <https://CRAN.R-project.org/package=bpnreg>.
- Cremers, Jolien, Kees Tim Mulder, and Irene Klugkist. 2017. “Circular interpretation of regression coefficients.” *British Journal of Mathematical and Statistical Psychology* 71: 5–95.
- Eddebuettel, Dirk. 2017. “Higher-Performance R Programming with C++ Extensions.” Zurich R Courses 2017, <https://goo.gl/Ng4oGQ>.
- Eddebuettel, Dirk, and Romain Francois. 2011. “Rcpp: Seamless R and C++ Integration.” *Journal of Statistical Software, Articles* 40 (8): 1–18.

- Eddelbuettel, Dirk, and Conrad Sanderson. 2014. “RcppArmadillo: Accelerating R with high-performance C++ linear algebra.” *Computational Statistics and Data Analysis* 71: 1054 – 1063.
- Fernández, Miguel A., Cristina Rueda, and Shyamal D. Peddada. 2011. “Identification of a core set of signature cell cycle genes whose relative order of time to peak expression is conserved across species.” *Nucleic Acids Research* 40 (7): 2823–2832.
- Fernández-Duran, J.J., and M.M. Gregorio-Domínguez. 2016. “CircNNTSR: An R Package for the Statistical Analysis of Circular, Multivariate Circular, and Spherical Data Using Nonnegative Trigonometric Sums.” *Journal of Statistical Software* 70 (6): 1–19.
- Fernández-López, Javier, and Klaus Schliep. 2018. *rWind: Download, Edit and Include Wind Data in Ecological and Evolutionary Analysis*. R package version 1.0.2, <https://CRAN.R-project.org/package=rWind>.
- Gelfand, A., P. Diggle, M. Fuentes, and P. Guttorp. 2010. *Handbook of Spatial Statistics*. Chapman and Hall.
- Gelman, Andrew, and Donald B. Rubin. 1992. “Inference from Iterative Simulation Using Multiple Sequences.” *Statistical Science* 7 (4): 457–472. <http://www.jstor.org/stable/2246093>.
- Ghazanfarihesari, Azade, and Majid Sarmad-Ferdowsi University Of Mashhad. 2016. *CircOutlier: Detection of Outliers in Circular-Circular Regression*. R package version 3.2.3, <https://CRAN.R-project.org/package=CircOutlier>.
- Grimt, E. P., T. Gneiting, V. J. Berrocal, and N. A. Johnson. 2006. “The continuous ranked probability score for circular variables and its application to mesoscale forecast ensemble verification.” *Quarterly Journal of the Royal Meteorological Society* 132 (621C): 2925–2942.
- Jammalamadaka, S. Rao, and A. SenGupta. 2001. *Topics in Circular Statistics*. Singapore: World Scientific.
- Jona Lasinio, G., A. E. Gelfand, and M. Jona Lasinio. 2012. “Spatial analysis of wave direction data using wrapped Gaussian processes.” *Annals of Applied Statistics* 6 (4): 1478–1498.
- Lagona, F., M. Picone, A. Maruotti, and S. Cosoli. 2015. “A hidden Markov approach to the analysis of space-time environmental data with linear and circular components.” *Stochastic Environmental Research and Risk Assessment* 29 (2): 397–409.
- Lagona, Francesco, and Marco Picone. 2012. “Model-based clustering of multivariate skew data with circular components and missing values.” *Journal of Applied Statistics* 39 (5): 927–945.
- Ley, Christophe, and Thomas Verdebout. 2017. *Modern Directional Statistics*. Interdisciplinary statistics. Chapman and Hall/CRC.
- Ley, Christophe, and Thomas Verdebout, eds. 2019. *Applied Directional Statistics: Modern Methods and Case Studies*. Interdisciplinary statistics. Chapman and Hall/CRC.
- Lund, Ulric, and Claudio Agostinelli. 2018. *CircStats: Circular Statistics, from "Topics in Circular Statistics" (2001)*. R package version 0.2-6, <https://CRAN.R-project.org/package=CircStats>.
- Mardia, K. V. 1972. *Statistics of Directional Data*. London and New York: Academic Press.
- Mardia, K. V., and P. E. Jupp. 1999. *Directional Statistics*. Chichester: John Wiley and Sons.
- Maruotti, A., A. Punzo, G. Mastrantonio, and F. Lagona. 2016. “A time-dependent extension of the projected normal regression model for longitudinal circular data based on a hidden Markov heterogeneity structure.” *Stochastic Environmental Research and Risk Assessment* 30 (6): 1725–1740.

- Mastrantonio, G., A. E. Gelfand, and G. Jona Lasinio. 2016. “The wrapped skew Gaussian process for analyzing spatio-temporal data.” *Stochastic Environmental Research and Risk Assessment* 30 (8): 2231–2242.
- Mastrantonio, G., G. Jona Lasinio, and A. E. Gelfand. 2016. “Spatio-temporal circular models with non-separable covariance structure.” *TEST* 25: 331–350.
- Microsoft Corporation, and Steve Weston. 2017. *doParallel: Foreach Parallel Adaptor for the ‘parallel’ Package*. R package version 1.0.11, <https://CRAN.R-project.org/package=doParallel>.
- Mulder, Kees, and Irene Klugkist. 2017. “Bayesian estimation and hypothesis tests for a circular Generalized Linear Model.” *Journal of Mathematical Psychology* 80: 4 – 14.
- Nadarajah, Saralees, and Yuanyuan Zhang. 2017. *Wrapped: Computes Pdf, Cdf, Quantile, Random Numbers and Provides Estimation for any Univariate Wrapped Distributions*. R package version 2.0, <https://CRAN.R-project.org/package=Wrapped>.
- Nuñez-Antonio, G., and E. Gutiérrez-Peña. 2014. “A Bayesian model for longitudinal circular data based on the projected normal distribution.” *Computational Statistics and Data Analysis* 71 (C): 506–519.
- Oliveira, María, Rosa M. Crujeiras, and Alberto Rodríguez-Casal. 2014. “CircSiZer: an exploratory tool for circular data.” *Environmental and Ecological Statistics* 21 (1): 143–159.
- Pewsey, Arthur, Markus Neuhauser, and Graeme D Ruxton. 2013. *Circular Statistics in R*. Oxford University Press.
- Rosenthal, Jeffrey S. 2007. “AMCMC: An R interface for adaptive MCMC.” *Computational Statistics and Data Analysis* 51 (12): 5467 – 5470.
- Rueda, Cristina, Miguel A. Fernández, and Shyamal Das Peddada. 2009. “Estimation of parameters subject to order restrictions on a circle with application to estimation of phase angles of cell cycle genes.” *Journal of the American Statistical Association* 104 (485): 338–347.
- Schnute, Jon T., and Kees Groot. 1992. “Statistical analysis of animal orientation data.” *Animal Behaviour* 43 (1): 15 – 33.
- Tsagris, Michail T, Giorgos Athineou, Anamul Sajib, Eli Amson, and Micah J. Waldstein. 2018. *Directional: Directional Statistics*. R package version 3.3, <https://CRAN.R-project.org/package=Directional>.
- Wang, F., and A. E. Gelfand. 2013. “Directional data analysis under the general projected normal distribution.” *Statistical Methodology* 10 (1): 113–127.
- Wang, F., and A. E. Gelfand. 2014. “Modeling space and space-time directional data using projected Gaussian processes.” *Journal of the American Statistical Association* 109 (508): 1565–1580.
- Wickham, H. 2015. *Advanced R*. Chapman & Hall/CRC The R Series. CRC Press.
- Zimmermann, Johannes, and Aidan G. C. Wright. 2017. “Beyond description in interpersonal construct validation: methodological advances in the circumplex structural summary approach.” *Assessment* 24 (1): 3–23. PMID: 26685192.