Analysis, Simulation and Prediction of Multivariate Random Fields with Package RandomFields

Martin Schlather
Universität Mannheim

Alexander Malinowski
Universität Mannheim

Peter J. Menck
Potsdam Institute for Climate Impact Research

Marco Oesting
AgroParisTech/INRA

Kirstin Strokorb
Universität Mannheim

Abstract

Modeling of and inference on multivariate data that have been measured in space, such as temperature and pressure, are challenging tasks in environmental sciences, physics and materials science. We give an overview over and some background on modeling with cross-covariance models. The R package **RandomFields** supports the simulation, the parameter estimation and the prediction in particular for the linear model of coregionalization, the multivariate Matérn models, the delay model, and a spectrum of physically motivated vector valued models. An example on weather data is considered, illustrating the use of **RandomFields** for parameter estimation and prediction.

Keywords: multivariate geostatistics, bivariate Matérn model, linear model of coregionalization, matrix-valued covariance function, multivariate random field, R, vector-valued field.

This vignette is based on Schlather, Malinowski, Menck, Oesting, and Strokorb (2014a).

1. Introduction

Spatial data very frequently have more than one component. In meteorology, for instance,

temperature, rainfall and pressure are measured at the same locations at predefined instances in time. Similarly, in order to estimate the spatial distribution of rainfall, data from rain gauges are combined with radar data. In economy, to name a further example, price developments of goods are spatio-temporal data consisting of various components at each instant of time and at each market.

Although these kinds of data are ubiquitous, multivariate spatial models are largely underdeveloped and multivariate space-time models are unknown, except for simple constructions. This paper deals with multivariate spatial data, only. We denote a random process on a subset of $\mathbb{R}^d$, $d \geq 1$, with scalar values a *univariate* random field, whilst other authors denote such a process already multivariate if $d \geq 2$. Here, a *multivariate* random field is an $\mathbb{R}^m$-valued random process on a subset of $\mathbb{R}^d$ with $m \geq 2$. Such fields are called *multivariable* in the package **gstat** (Pebesma 2004).

The analysis and the modeling of data stemming from multivariate random fields have been reported in the geostatistical literature since long, cf. Wackernagel (2003) and Chilès and Delfiner (1999), for example. Recently, the subject of multivariate geostatistics has been taken up and various new models have been suggested (Apanasovich and Genton 2010; Gneiting, Kleiber, and Schlather 2010; Bevilacqua, Daley, Porcu, and Schlather 2013).

Even though several R packages (R Core Team 2013) like **fields** (Nychka, Furrer, and Sain 2013), **geospt** (Melo, Santacruz, Melo, and others 2012), **SpatialTools** (French 2014) or **spTimer** (Bakar and Sahu 2013) address the simulation and inference for spatial data, it seems that only the packages **gstat** (Pebesma 2004), **multivator** (Hankin 2012) and **spBayes** (Finley and Banerjee 2013) can handle *multivariate* random fields in the above sense.

While **spBayes** mainly focuses on spatial regression models, **multivator** allows for inference and prediction of multivariate data based on emulator techniques (Oakley 1999). The package **gstat** provides several tools for prediction and model fitting and also allows for multi-Gaussian sequential simulation. However, to the best of our knowledge, the package **RandomFields** (Schlather *et al.* 2014b) is by now unique in its variety of tools and models for multivariate spatial processes.

The paper is organized as follows. In Section 2, a summary of the theoretical background is given. Section 3 discusses principles for analyzing and simulating multivariate random fields. Section 4 offers an overview over multivariate and vector-valued covariance functions that are implemented in **RandomFields**. Section 5 reconsiders the data analysis by Gneiting *et al.* (2010). Section 6 concludes with a brief summary of other features of **RandomFields** that are already available or are intended to be implemented.

2. Background: cross-covariance functions and cross-variograms

Throughout the paper, a second order m-variate random field on $T \subset \mathbb{R}^d$ denotes a collection of real-valued random vectors $Z(x) = (Z_1(x), \ldots, Z_m(x))$ indexed by $x \in T$ with existing second moments. For theoretical considerations it suffices to assume $\mathsf{E}Z_j(x) = 0$ for all $x \in \mathbb{R}^d$ and $j = 1, \ldots, m$. The cross-covariance function $C : T \times T \to \mathbb{R}^{m \times m}$ is defined by

$$C_{jk}(x, y) = \mathsf{COV}(Z_j(x), Z_k(y)), \qquad x, y \in T, j, k = 1, \ldots, m.$$

The function C is positive definite, i.e., $C(x, y) = C^\top(y, x)$ and

$$\sum_{p=1}^{n} \sum_{q=1}^{n} a_p^\top C(x_p, x_q) a_q \geq 0 \qquad (1)$$

for all $n \in \mathbb{N}$, $x, y, x_1, \ldots, x_n \in T$ and $a_1, \ldots, a_n \in \mathbb{R}^m$. The cross-variogram $\gamma : T \times T \to \mathbb{R}^{m \times m}$, is defined by

$$\gamma_{jk}(x, y) = \frac{1}{2} \mathsf{E}(Z_j(x) - Z_j(y))(Z_k(x) - Z_k(y)), \qquad x, y \in T, \ j, k = 1, \ldots, m.$$

If the cross-covariance function C (the cross-variogram γ) depends on x and y only through the distance vector $x - y$, then Z is called weakly stationary or just stationary (intrinsically stationary). We then write $C(h)$ and $\gamma(h)$ as functions that depend only on one argument $h = x - y$. Clearly, stationarity implies intrinsic stationarity. Further relations between covariance functions and variograms exist, see Chilès and Delfiner (1999) and Schlather (2012), for instance, and the references therein.

In the univariate case, (weak) isotropy of Z is uniquely defined and means that the function C or γ is rotation invariant. In the multivariate case the notion of isotropy has different interpretations. Assume that the bivariate field Z models temperature and pressure. Then Z should be called isotropic if

$$C(x, y) = C(Ax, Ay) \text{ or } \gamma(x, y) = \gamma(Ax, Ay) \qquad \text{for all rotation matrices } A \in \mathbb{R}^{d \times d}.$$

Assume now that Z models a vector field of directed quantities in $\mathbb{R}^d$, e.g., wind speed in $\mathbb{R}^2$. Then Z has necessarily $m = d$ components and it should be called isotropic if $Z(\cdot) \overset{\mathcal{D}}{=} A^\top Z(A \cdot)$, i.e.,

$$C(x, y) = A^\top C(Ax, Ay) A \qquad \text{for all rotation matrices } A \in \mathbb{R}^{d \times d}.$$

A similar definition holds for the variogram γ. We call the latter models isotropic *vector-valued models* whilst the former ones are referred to as isotropic *multivariate models*.

3. Simulation and inference

Several methods for the simulation of univariate random fields in $\mathbb{R}^d$ exist, see Lantuéjoul (2002) for an overview. The simulation of univariate random fields based on Cholesky decomposition, see also **SpatialTools**, can be directly transferred to the multivariate case. As the Cholesky decomposition currently reaches its limits with a matrix of about size $10^4 \times 10^4$, only about 2500 locations can be modeled in a bivariate setup and about 1000 locations in a trivariate setup. The circulant embedding method by Dietrich and Newsam (1993) and Wood and Chan (1994) is extremely powerful for simulating on a regular grid in $\mathbb{R}^d$ when d is small and the random field is stationary. Their algorithms are implemented in various packages, e.g., **dvfBm** (Coeurjolly 2009), **fields**, and **stpp** (Gabriel, Rowlingson, and Diggle 2013). The method has been extended by Dietrich and Newsam (1996) and Chan and Wood (1999) to the multivariate case. In **RandomFields**, the Cholesky decomposition and the circulant embedding method suggested by Dietrich and Newsam (1996) are implemented also in the multivariate case. Furthermore, an extended version of Matheron's turning bands method (Matheron 1973) for the multivariate case is implemented.

Various methods exist to estimate the parameters of an intrinsically stationary random field. Least squares methods are based on the variogram cloud or a binned version of it, and do not require further assumptions on the fields (Cressie 1993). For example, the package **gstat** relies mainly on these methods. The second often used method is the maximum likelihood (ML) method which relies on the additional important assumption of the Gaussianity of the field. Showing that the ML approach is not too far from an established approach in approximation theory, Scheuerer (2011) gives some indication why ML often works well even when the assumption of Gaussianity is violated. The ML method and its variants (**geoR**, Ribeiro Jr and Diggle 2001, **CompRandFld**, Padoan and Bevilacqua 2013) is frequently implemented for parametric inference on spatial data. For Gaussian random fields, this amounts to estimating a parametrized trend and the covariance structure. Therefore, ML methods can be readily extended to the multivariate case.

At least three different kinds of well known difficulties exist with the classical ML approach: first, the number of data is rather limited as the multivariate Gaussian density function involves the inverse of the covariance matrix, so that the calculation of the likelihood needs of order N^3 operations. There are various work-arounds, for instance, Markov random field theory allows to work directly on the inverse matrix, i.e., the precision matrix, (Rue 2001; Simpson, Lindgren, and Rue 2011). Composite likelihood approaches simplify the likelihood function by additional assumptions, cf. Lindsay (1988). A second difficulty is that the number of parameters increases rapidly with the number m of components resulting frequently in likelihood functions with lots of local maxima. As numerical optimizers such as `optim` are not always reliable in this case, **RandomFields** estimates a sequence of simpler models, starting with the model that has independent components, ending up with the full set of parameters, see Section 5 for an example. The price to be payed are rather long running times.

The third difficulty is the choice of the initial values of the numerical optimization algorithm for maximizing the likelihood. Various strategies exist. In **geoR**, initial starting values can be created by various methods and are then passed as an argument to the fitting procedure. Others seem to use fixed values only (**fields**). Data based initial values are considered by **CompRandFld** using a weighted moment estimator and by **gstat** extracting values from the variogram cloud. Usually, the packages allow for an additional starting value given by the user. **RandomFields** estimates variances and scales very roughly from the data and uses fixed values for other parameters to perform several variants of the least squares fit. Afterwards it uses the best parameter set of the least squares fits as initial values for the ML estimation. Similar to the packages **CompRandFld** and **fields**, the package **RandomFields** recognizes sparse matrices and uses the package **spam** (Furrer and Sain 2010) for calculating the likelihood.

Being also mathematically based on the inverse of the covariance matrix, spatial prediction methods face similar problems as the ML approach (Chilès and Delfiner 1999). Work-arounds that share the same idea with the composite likelihood approach, such as the neighbourhood kriging (De Marsily 1984), have been suggested. Many packages exist that deal with kriging, for instance, **constrainedKriging** (Hofer and Papritz 2011), **DiceKriging** (Roustant, Ginsbourger, and Deville 2012) and **geoR**. The package **gstat** offers an enormous amount of variants for kriging, including multivariate kriging for linear models of coregionalization (Goulard and Voltz 1992). **RandomFields** allows for basic kriging tasks, such as simple kriging, ordinary kriging and universal kriging, on a large variety of models.

A last, also well-known difficulty appears with earth data as only few packages like **sspline** (Xie 2013) handle spherical data. Usually, the data are transformed in different ways to more

convenient distances by the various packages. The packages **gstat** and **fields** use great circle distances. Additionally, **fields** also considers the data as having fully 3-dimensional coordinates, from which the usual Euclidean distances are calculated. The latter is also the case in **RandomFields**. While the package **fields** approximates the earth by a ball, **RandomFields** assumes an ellipsoid.

4. Models

Among the covariance models considered as the most important ones in geostatistics are the Whittle-Matérn model (Stein 1999; Guttorp and Gneiting 2006), its special cases, the exponential model and the Gaussian model, and the spherical model which has compact support (Chilès and Delfiner 1999). Only a few packages allow for a wider spectrum of models, for instance **geoR**. On the multivariate side, the packages **gstat** and **spBayes** allow for some choice of multivariate models.

In the following, we describe the cross-covariance models and the cross-variogram models that can be used within **RandomFields**. A comprehensive list of models is given by Table 1. Multivariate models like the latent dimension model and the linear model of coregionalization can be derived from a wide spectrum of univariate models, some of them being presented along this section.

The pieces of code given suffice to produce all the pictures as presented if the following three options

```
R> RFoptions(seed = 0, height = 4)
```

are set once, at the very beginning. The first option sets the seed for every subsequent simulation to 0. The second one sets the height of a single figure in the graphical display to 4 and indicates also that the aspect ratio of the figure will be calculated internally. The third one has the effect that a new window is opened for each plot.

4.1. Multivariate models

The most commonly used model for a multivariate process $Z = (Z_1, \ldots, Z_m)^\top$ is probably the linear model of coregionalization (Goulard and Voltz 1992), which is also offered by **gstat**. Here, each component Z_i is a linear combination $\sum_{j=1}^{k} m_{ij} Y_j$ of independent, latent univariate processes Y_j and $M = (m_{ij}) \in \mathbb{R}^{m \times k}$ is an arbitrary matrix. Writing $M_j = (m_{1j}, \ldots, m_{mj})^\top$ we have

$$Z = \sum_{j=1}^{k} M_j Y_j.$$

The cross-covariance function of Z equals $MCM^\top$ where C is a diagonal matrix containing the covariance functions of the univariate processes Y_j. Obviously, the vector of independent processes $Y = (Y_1, \ldots, Y_k)$ might be replaced by any arbitrary second order process Y, so that C can be replaced by any arbitrary $k \times k$ cross-covariance function. The function RMmatrix in **RandomFields** allows for this generality. The parameter k must be at most 10. To consider an example, let $k = 2$,

$$M_1 = \begin{pmatrix} 0.9 \\ 0.6 \end{pmatrix}, \qquad M_2 = \begin{pmatrix} \sqrt{0.19} \\ 0.8 \end{pmatrix},$$

Name	Description	Particularities
Stationary and isotropic multivariate models		
RMbiwm	bivariate Whittle-Matérn model	modeling differentiability
RMbigneiting	bivariate Gneiting model	finite support
RMparswm	multivariate Whittle-Matérn model	modeling differentiability
Other multivariate models		
RMdelay	bivariate delay effect model	anisotropic
RMexponential	functional exp	operator
RMmatrix	matrix multiplication	operator
RMschur	element-wise product	operator
RMmqam	multivariate quasi-arithmetic mean	operator
Vector-valued models		
RMcurlfree	curl free vector-valued field	operator
RMdivfree	divergence free vector-valued field	operator
RMkolmogorov	Kolmogorov's model of turbulence	isotropic variogram
RMtbm	turning bands operator	operator
RMvector	combining RMcurlfree and RMdivfree	operator

Table 1: Multivariate and vector-valued models implemented in **RandomFields**.

and Y_1, Y_2 be independent univariate Gaussian random fields with Whittle-Matérn covariance function

$$W_\nu(h) = 2^{1-\nu}\Gamma(\nu)^{-1}\|h\|^\nu K_\nu(\|h\|), \qquad h \in \mathbb{R}^d.$$

Here, K is a modified Bessel function, Γ the Gamma function, and $\nu > 0$ a smoothness parameter. The values of M_1 and M_2 imply that all the variables have variance 1. This model can be defined in **RandomFields** as follows, see Figure 1 for a realization:

```
R> M1 <- c(0.9, 0.6)
R> M2 <- c(sqrt(0.19), 0.8)
R> model <- RMmatrix(M = M1, RMwhittle(nu = 0.3)) +
+              RMmatrix(M = M2, RMwhittle(nu = 2))
R> x <- y <- seq(-10, 10, 0.3)
R> simu <- RFsimulate(model, x, y)
R> plot(simu)
```

In particular, ν takes here the value 0.3 and 2, respectively. Note that, in **RandomFields**, all commands that define models start with 'RM' whilst functions start with 'RF'. The models in **RandomFields** mostly include more parameters than given here. These parameters are set to standard values if they are not given. For instance, the standard value for variance and scale is 1.

Wackernagel (2003) suggests a bivariate delay effect model of the form

$$C_{11}(h) = C_{22}(h) = C_0(h), \quad C_{12}(h) = C_0(h+s), \quad C_{21}(h) = C_0(h-s),$$

where C_0 is an arbitrary univariate stationary covariance function and $s \in \mathbb{R}^d$. Here, the
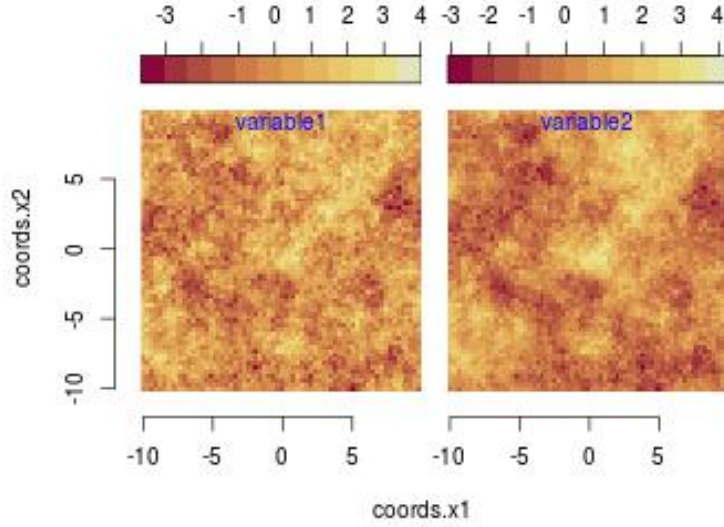
Figure 1: Simulation of a bivariate linear model of coregionalization.

components of the corresponding random fields experience a pure shift, see Figure 2. The corresponding code is

```
R> model <- RMdelay(RMstable(alpha = 1.9, scale = 2), s = c(4, 4))
R> simu <- RFsimulate(model, x, y)
R> plot(simu, zlim = 'joint')
```

Here, the shift equals $s = (4, 4)$. The underlying Gaussian random field possesses the so-called stable covariance model, also called power exponential model,

$$C_0(h) = \exp(-(\|h\|/S)^\alpha)$$

with scale parameter $S = 2$ and shape parameter $\alpha = 1.9$. The parameter `zlim` that is well-known from the function `image` allows here for further options. Both, the value `'joint'` and a simple two-dimensional vector fix the same range for all images. A matrix-valued `zlim` is able to determine a separate range for each column of the plot.

The delay model can be extended to an $(m + 1)$-variate model as well as to a superposition of different shifts:

$$C(h) = \left(\sum_{\kappa=1}^{k} C_{0,\kappa}(h - s_{\kappa,i} + s_{\kappa,j}) \right)_{i,j=0,\dots,m} \tag{2}$$

where $C_{0,\kappa}$ are univariate covariance functions, $s_{\kappa,i} \in \mathbb{R}^d$ and $s_{\kappa,0} = 0$. Even for $k = 2$ and $C_{0,1} = C_{0,2}$ the delays cannot be recovered by visual inspection, see Figure 3. Here, the above `model` has been replaced by

```
R> model <- RMdelay(RMstable(alpha = 1.9, scale = 2), s = c(0, 4)) +
+           RMdelay(RMstable(alpha = 1.9, scale = 2), s = c(4, 0))
```

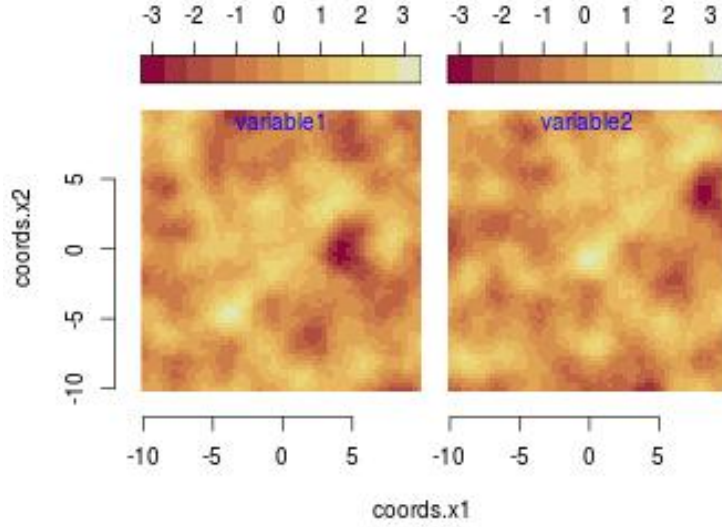The contour plot for the covariance function has been obtained by

Figure 2: Simulation of Wackernagel's (2003) delay effect model. Here, the shift s equals $s = (4, 4)$.

```
R> plot(model, dim = 2, xlim = c(-5, 5), main = "Covariance function",
+          cex = 1.5, col = "brown")
```

The generalized model given by Equation 2 is implemented in **RandomFields** for $k \leq 10$.

A third basic construction principle is the latent dimension model (Apanasovich and Genton 2010). Here, additional spatial dimensions serve as a source for the multivariate components:

$$Z_i(x) = Z_0(x, z_i), \quad x \in \mathbb{R}^d, \ i = 1, \ldots, m,$$

for a random field Z_0 in $\mathbb{R}^{d+k}$ and vectors $z_i \in \mathbb{R}^k$. An example is given in Figure 4, where $m = d = 2$, $k = 1$, $z_1 = 0$ and $z_2 = 1$,

```
R> model <- RMgencauchy(alpha = 1.5, beta = 3)
R> simu <- RFsimulate(model, x, y, z = c(0, 1))
R> plot(simu, MARGIN.slices = 3, n.slices = 2)
```

Here, the covariance function of Z_0 is a generalized Cauchy model (Gneiting and Schlather 2004),

$$C(h) = (1 + \|h\|^{\alpha})^{-\beta/\alpha}$$

with parameter $\alpha = 1.5$ and $\beta = 3$. The argument `MARGIN.slices` gives the latent dimension $d + k$. The argument `n.slices` equals m if we are interested in a m-variate field. The slices are equally distributed on a grid given for the latent dimension. As this plotting technique of slices is independent of any theory on latent dimensions the names of the arguments refer to the technical aspect only.

The random fields for the different variables are visually similar for all models given here. In particular, the paths of all components show the same degree of smoothness. The multivariate quasi-arithmetic mean model by Porcu, Mateu, and Christakos (2009),

$$C_{ij}(h) = \varphi\left(\sqrt{\theta(\varphi^{-1}(C_{0,i}(h)))^2 + (1 - \theta)(\varphi^{-1}(C_{0,j}(h)))^2}\right), \qquad i, j = 1, \ldots, m,$$
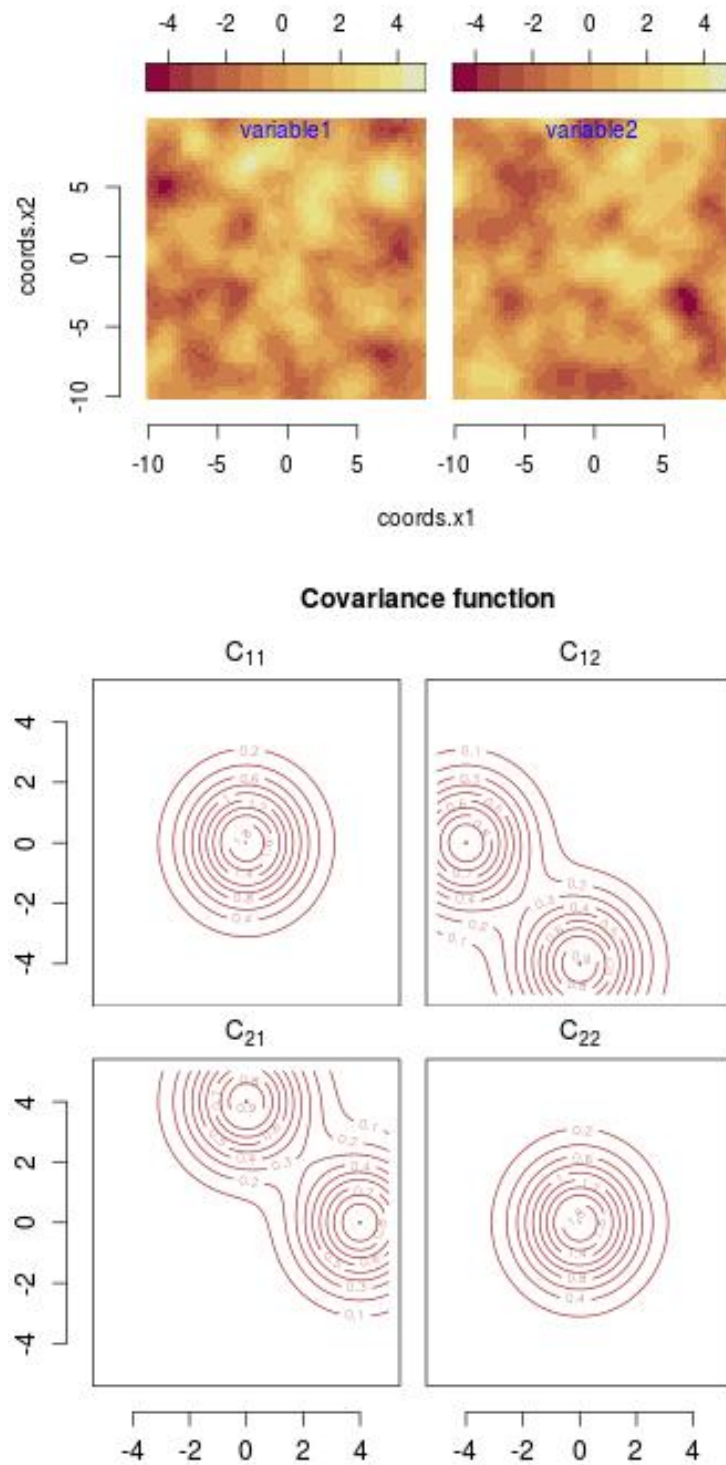
Figure 3: Bivariate generalized delay effect model according to Equation 2: simulation (top) and contour plots for the components of the corresponding matrix-valued covariance function (bottom).
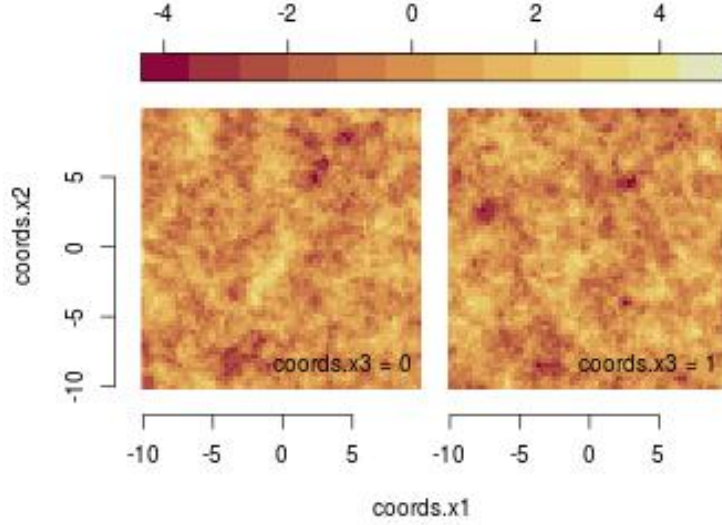
Figure 4: Simulation of the latent dimension model.

exhibits similar limitations. Here the $C_{0,i}$ are arbitrary univariate covariance functions, φ is a completely monotone function (Berg, Christensen, and Ressel 1984; Schilling, Song, and Vondracek 2010) and φ^{-1} is its inverse.

Gneiting *et al.* (2010) present a multivariate extension of the Whittle-Matérn model W_ν. Let $C_{ij}(h) = (c_{ij}W_{\nu_{ij}}(h/s_{ij}))_{i,j=1,\ldots,m}$ with $c_{ij} = c_{ji} \in \mathbb{R}$, $c_{ii} \geq 0$, $\nu_{ij} = \nu_{ji} > 0$ and $s_{ij} = s_{ji} > 0$ for $i,j = 1,\ldots,m$. Let $C = (C_{ij})_{i,j=1,\ldots,m}$. If $m = 2$ then C is a cross-covariance function if and only if

$$c_{12} = \rho_{\text{red}}\left(c_{11}c_{22}G_{12}\frac{s_{11}^d s_{22}^d}{s_{12}^{2d}}\inf_{t\geq 0}\frac{(1 + t^2/s_{12}^2)^{2\nu_{12}+d}}{\prod_{i=1}^2(1 + t^2/s_{ii}^2)^{\nu_{ii}+d/2}}\right)^{1/2}$$

for some $\rho_{\text{red}} \in [-1, 1]$ and

$$G_{ij} = \frac{\Gamma(\nu_{ii} + d/2)\Gamma(\nu_{jj} + d/2)\Gamma(\nu_{ij})^2}{\Gamma(\nu_{ii})\Gamma(\nu_{jj})\Gamma(\nu_{ij} + d/2)^2}.$$

Gneiting *et al.* (2010) show that the right-hand side of the above equality is 0, i.e., c_{12} is necessarily 0 if and only if $\nu_{12} < \frac{1}{2}(\nu_{11} + \nu_{22})$. Hence, we may introduce $\nu_{\text{red}} = 2\nu_{12}/(\nu_{11} + \nu_{22})$ which must be greater than or equal to 1 to have dependent components. Figure 5 gives an illustration obtained by the following code:

```
R> model <- RMbiwm(nudiag = c(1, 2), nured = 1, rhored = 1, cdiag = c(1, 5),
+                  s = c(1, 1, 2))
R> simu <- RFsimulate(model, x, y)
R> plot(simu)
```

i.e., in the above model we have

$$\nu_{11} = 1, \quad \nu_{22} = 2, \quad \nu_{12} = 1.5, \quad \rho_{\text{red}} = 1, \quad c_{11} = 1, \quad c_{22} = 5, \quad s_{11} = 1, \quad s_{12} = 1, \quad s_{22} = 2.$$
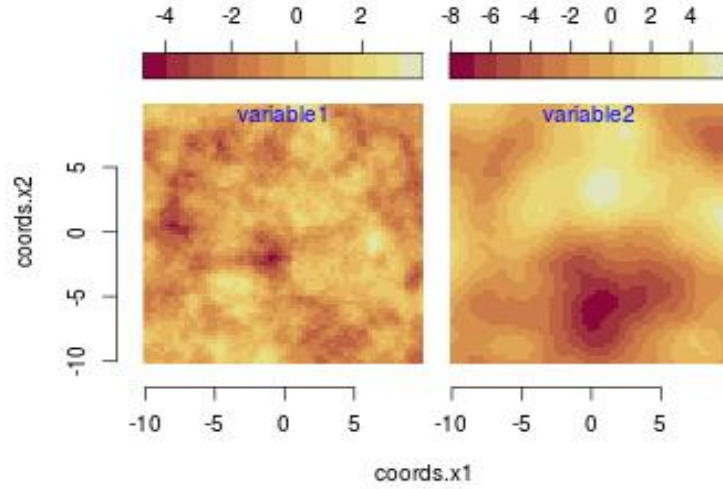
Figure 5: Simulation of the bivariate Whittle-Matérn model.

For the general m-variate case, Gneiting *et al.* (2010) give a sufficient condition for C being a cross-covariance model,

$$s_{ij} = s > 0, \quad \nu_{ij} = \frac{1}{2}(\nu_{ii} + \nu_{jj}) \text{ and } c_{ij} = \sqrt{G_{ij}} \quad \text{for all } i, j = 1, \ldots, m. \tag{3}$$

The multivariate model for three or more variables is available in **RandomFields** by the function `RMparswm`. The parsimonious model K introduced in Gneiting *et al.* (2010) allows for more flexibility, $K_{ij} = \rho_{ij}C_{ij}$ where C is given by the equalities in Equation 3 and ρ is an arbitrary covariance matrix. This generalization is implemented by the model `RMparswmX` defined as

```
R> RMparswmX <- function(nudiag, rho, var, scale, Aniso, proj) {
+      RMschur(M = rho, RMparswm(nudiag, var, scale, Aniso, proj))
+  }
```

Here, `RMschur` defines the Schur product between two matrices, here ρ and $C(h)$ for any fixed value of h.

Anisotropic models can be defined through the argument `Aniso`. For instance, the code

```
R> A1 <- RMangle(angle = pi / 4, diag = c(0.1, 0.5))
R> A2 <- RMangle(angle = 0, diag = c(0.1, 0.5))
R> model <- RMmatrix(M = M1, RMgengneiting(kappa = 0, mu = 2, Aniso = A1)) +
+              RMmatrix(M = M2, RMgengneiting(kappa = 3, mu = 2, Aniso = A2))
R> simu <- RFsimulate(model, x, y)
R> plot(simu)
```

defines an anisotropic model of coregionalization. Here, `M1` and `M2` are defined as above, `angle` gives the clockwise rotation and `diag` the inverse of the stretch factors in the new directions, cf. Figure 6. The covariance model `RMgengneiting` refers to the Gneiting class (Gneiting 1999; Wendland 2005), where `kappa` denotes a parameter for the differentiability of the model
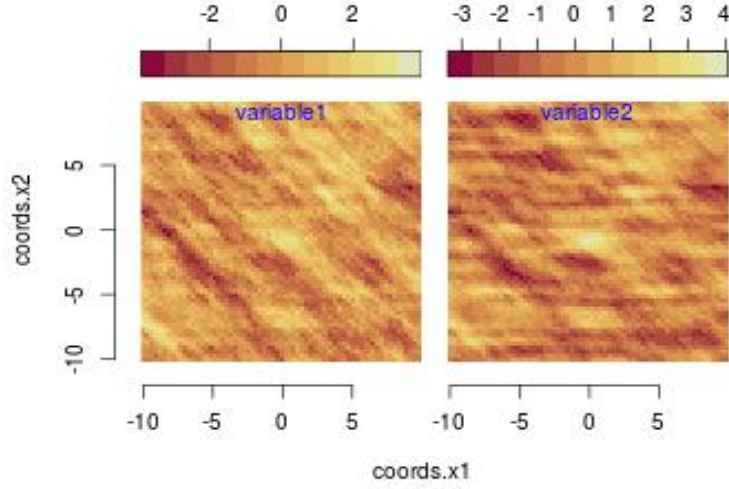
Figure 6: Simulation of an anisotropic model of coregionalization.

and `mu` denotes an additional form parameter. These models have compact support. For $\kappa = 0$ and $\kappa = 3$ they are defined as

$$C(h) = (1 - \|h\|)^{\beta} I(\|h\| \leq 1)$$

and

$$C(h) = (1 + \beta\|h\| + (2\beta^2 - 3)\|h\|^2/5 + (\beta^2 - 4)\beta\|h\|^3/15)(1 - \|h\|)^{\beta} I(\|h\| \leq 1),$$

respectively. Here, $\beta = \mu + 2\kappa + 1/2$ and $I(\|h\| \leq 1)$ is the indicator function.

Finally, Bevilacqua *et al.* (2013) derive multivariate models with compact support based on the Gneiting class. The bivariate model is available as `RMbigneiting`.

4.2. Vector-valued models

Vector fields are used to describe fluxes of liquids or gases in fluid mechanics (Patankar 1980; Chorin and Marsden 1990), as well as fields of forces, e.g., electro-magnetic fields (Stratton 2007) or, more generally, as gradients of potentials (Kellog 1953).

Stochastic vector fields appear naturally within Kolmogorov's theory of turbulent flows (Pope 2000, Section 6.2). Here, micro-fluctuations are described by an intrinsically stationary, isotropic vector field, given by the variogram γ,

$$\gamma_{ij}(h) = \|h\|^{2/3} \left(\frac{4}{3} \delta_{ij} - \frac{1}{3} \frac{h_i h_j}{\|h\|^2} \right), \qquad h \in \mathbb{R}^3. \tag{4}$$

Here $\delta_{ij} = 1$ for $i = j$ and $\delta_{ij} = 0$, otherwise. It is well-known that the marginal distributions in a turbulent flow are not Gaussian.

Let $F = (F_1, \ldots, F_d)$ be a vector field on $\mathbb{R}^d$. The curl $\nabla \times F$ is defined as $\nabla \times F = \partial F_2/\partial x_1 - \partial F_1/\partial x_2$ if $d = 2$ and as

$$\nabla \times F = \left(\frac{\partial F_3}{\partial x_2} - \frac{\partial F_2}{\partial x_3} \right) e_1 + \left(\frac{\partial F_1}{\partial x_3} - \frac{\partial F_3}{\partial x_1} \right) e_2 + \left(\frac{\partial F_2}{\partial x_1} - \frac{\partial F_1}{\partial x_2} \right) e_3$$

if $d = 3$. Here, the e_i are the canonic unit vectors. The divergence $\nabla \cdot F$ is given as

$$\nabla \cdot F = \sum_{i=1}^{d} \frac{\partial F_i}{\partial x_i}.$$

An important statement, not only for the theory of turbulence, is the Helmholtz decomposition: Under mild regularity assumptions, a vector field F in three dimensions can be uniquely decomposed into a curl free vector field F^c, i.e.,

$$\nabla \times F^c = 0,$$

and a divergence free vector field F^d, i.e.,

$$\nabla \cdot F^d = 0$$

such that $F = F^c + F^d$ (Sagaut and Cambon 2008). A curl free vector field F^c can either be interpreted as the flux of an incompressible liquid that is verbally driven by sinks and sources, or it describes a fluid that is compressible and liquid 'disappears' if it is compressed. The absolute value of the divergence $\nabla \cdot F$ gives the strength of the sinks and sources, where the sign determines whether it is a sink or a source. Divergence free vector fields describe vortices and magnetic fields. The direction of the curl $\nabla \times F$ gives the axis of the vortex and the norm of the curl its strength.

Stationary and non-stationary Gaussian random fields that are divergence free or curl free (in the mean square sense, hence pathwise) can be easily constructed from a sufficiently often differentiable univariate covariance function, C_0 say, cf. Matheron (1972, 1973), for instance. For simplicity, let C_0 be translation invariant. Then,

$$C^d(h) = (-\Delta E + \nabla \nabla^\top) C_0(h)$$

is the cross-covariance function of a stationary divergence free random field and

$$C^c(h) = (-\nabla \nabla^\top) C_0(h)$$

is the cross-covariance function of a curl free random field. Here, E denotes the identity matrix and Δ denotes the Laplace operator. Scheuerer and Schlather (2012) show that if a vector-valued random field on $\mathbb{R}^2$ is divergence free or curl free or a vector-valued random field on $\mathbb{R}^3$ is curl free, then their cross-covariance functions have been obtained by essentially the above construction.

The curl free models, the divergence free models, and Kolmogorov's cross-variogram have been implemented in **RandomFields**. The covariance model `RMcurlfree` defines random fields in $\mathbb{R}^2$ that have four components: the first component is the (random) potential field given by C_0, the next two components give the derived (curl free) vector field, whereas the last component gives the field of sinks and sources of the vector field, i.e., the divergence of the vector field. In fact, due to the included field of sinks and sources, whose covariance function equals $\Delta^2 C_0$, the covariance function C_0 must be at least four times differentiable. Random fields simulated with `RMdivfree` have also four components that can be interpreted in a similar way, the last component being here the curl of the respective divergence free vector field. Figure 7 illustrates such a four-variate random field. The code is

```
R> model <- RMcurlfree(RMmatern(nu = 5), scale = 4)
R> simu <- RFsimulate(model, x, y)
R> plot(simu, select.variables = list(1, 2 : 3, 4))
R> plot(model, dim = 2, xlim = c(-3, 3), main = "", cex = 2.3, col = "brown")
```

Here, `plot` has the additional argument `select.variables`, by which the user chooses the components to be plotted and the way they are plotted. If `select.variables` is not given then `plot` plots all components separately. If `select.variables` is a list, then any component of the list should be a scalar or a vector and is plotted in a separate figure. If an element of the list is a scalar, an `image` is created. If the element is a bivariate vector a vector field is created and if the element is a trivariate vector, an `image` is plotted for the first component, and, additionally, a vector field is shown given by the subsequent two components.

The model `RMvector` allows combinations of a divergence free and a curl free spatial or space-time random field

$$C(h, t) = (-0.5(a + 1)\Delta + a\nabla\nabla^\top)C_0(h, t), \quad h \in \mathbb{R}^d, \ t \in \mathbb{R}, \ d = 2, 3,$$

where the ∇-operators and the Laplace operator act on the spatial components only and $a \in [-1, 1]$. The model is parametrized in a way such that $a = -1$ corresponds to a curl free field and $a = 1$ to a divergence free field. The corresponding random field is bivariate and contains the vectors only. Here, the covariance function C_0 must be only twice differentiable in the spatial component since the field of sources and sinks or curls is not included. Finally, Figure 8 shows an intrinsically stationary Gaussian vector field with Kolmogorov's variogram, which has been obtained by ■results=tex■ x <- y <- seq(-2, 2, len = 20) model <- RMkolmogorov() simu <- RFsimulate(model, x, y, z = 1) plot(simu, select.variables = list(1 : 2), col = c("red")) plot(model, dim = 3, xlim = c(-3, 3), MARGIN = 1 : 2, cex = 2.3, fixed.MARGIN = 1.0, main = "", col = "brown") Here, `MARGIN` gives the two coordinate directions that should be plotted if the dimension d is greater than 2. The argument `fixed.MARGIN` gives the values of the remaining coordinates, here one coordinate, that are kept fix. In fact, `fixed.MARGIN = 0.0`, which is the default, does not yield very illuminating results, cf. Equation 4, so that we have chosen `fixedMARGIN = 1.0`.

5. Example: weather data

The data set considered here gives the difference between both the pressure and the temperature at 157 locations in the United States measured at a certain time instance and their predictions 48 hours ago. As in Gneiting *et al.* (2010) we assume that the expectation of the differences is zero, and refer to `data(weather)` for details.

We consider first a simplified version of the bivariate Whittle-Matérn model from Section 4.1 where ν_{red} is set to 1 and the scale $s_{11} = s_{12} = s_{22}$ is the same for all entries in the cross covariance. A bivariate nugget effect with independent components is added:

```
R> nug <- RMmatrix(M = matrix(nc = 2, c(NA, 0, 0, NA)), RMnugget())
R> pars.model <- nug + RMbiwm(nudiag = c(NA, NA), scale = NA,
+                             cdiag = c(NA, NA), rhored = NA)
```

This model is called 'parsimonious model' in Gneiting *et al.* (2010). We transform the earth coordinates to Euclidean distances before performing the analysis:
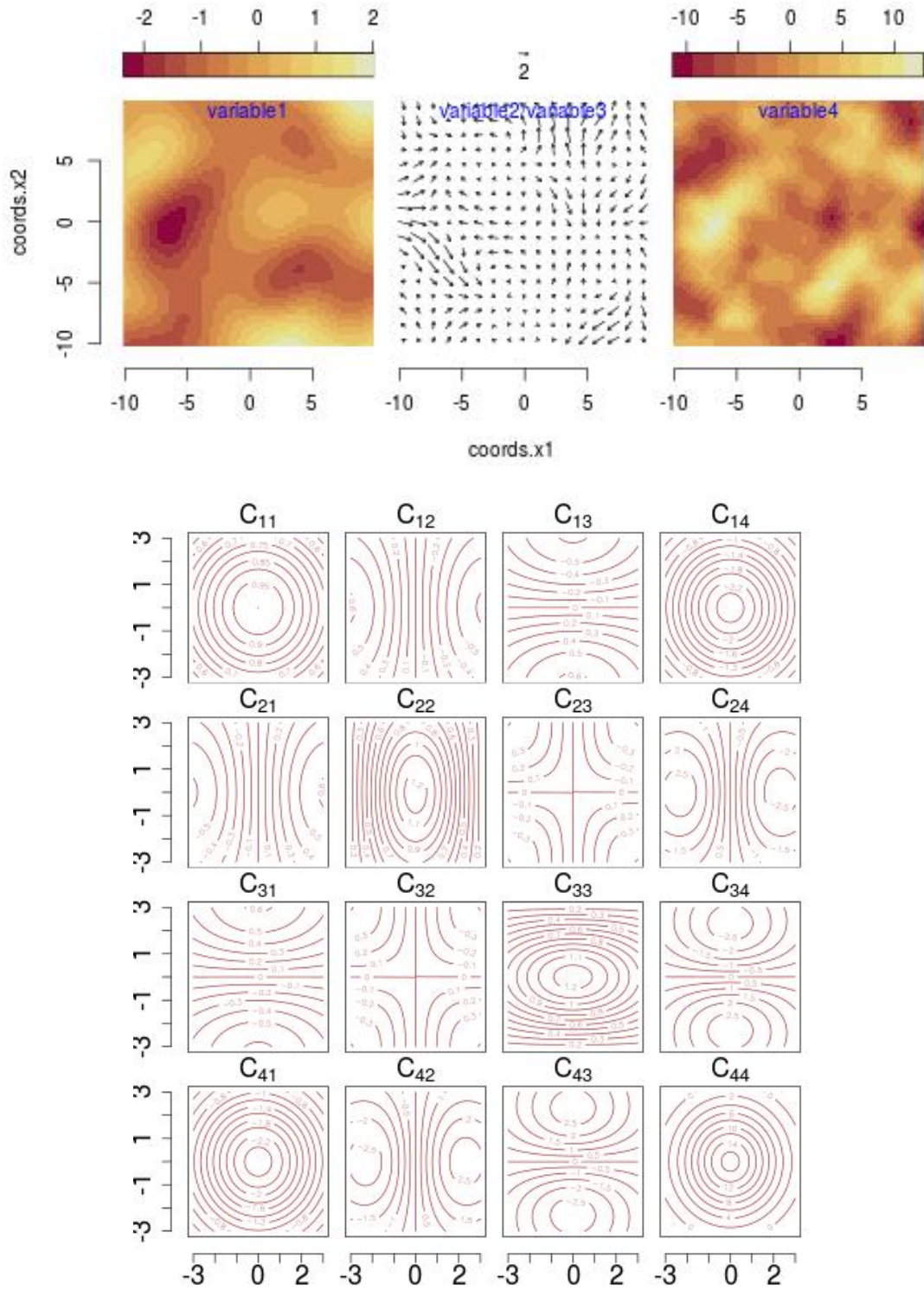
Figure 7: Simulation of a curl free vector field. Top left: potential field (variable 1); top center: the corresponding curl free vector field (variables 2/3) is given by the arrows whose length can be visibly inspected by comparing them to the above arrow which represents a vector of length 2; top right: the field of sinks and sources (variable 4), i.e., the divergence of the vector field in the top center; bottom: contour plots of the covariance function for these variables 1-4.
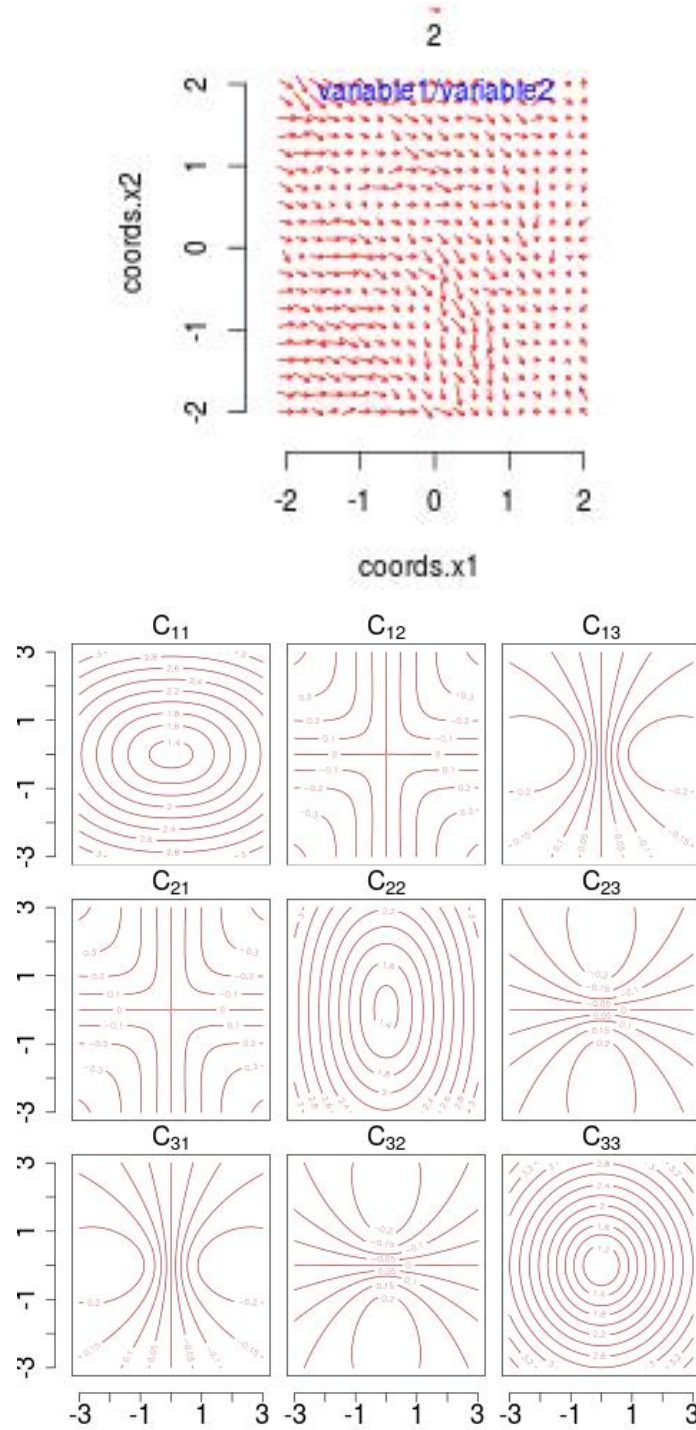
Figure 8: Kolmogorov's model of turbulence. The top picture shows the vector field of the first two components of a simulated three-dimensional vector field at the locations where the third coordinate is kept fixed. Their length can be visibly inspected by comparing them to the above arrow which represents a vector of length 2. The bottom pictures show the variogram values between all components at the locations where the third component is fixed to the value 1.

```
R> data(weather)
R> Dist.mat <- as.vector(RFearth2dist(weather[ , 3 : 4]))
```

The function `RFfit` recognizes all parameters that have the value `NA`, and estimates them through ML. Fitting the parsimonious model yields

```
R> PT <- weather[ , 1 : 2]
```

```
R> pars <- RFfit(pars.model, distances = Dist.mat, dim = 3, data = PT)
```

```
Note: There are strong indications that the model might be overparametrised
or that the bounds for the variables are too wide. Try narrower lower and
upper bounds for the variables in the latter case. One of the critical
parameters is 'lowerbound_var_factor' whose value (10000)
might be reduced.

Note: There are very strong indications that the model might be overparametrised
or that the bounds for the variables are too wide. Try narrower lower and
upper bounds for the variables in the latter case. One of the critical
parameters is 'lowerbound_var_factor' whose value (10000)
might be reduced.

Note: There are strong indications that the model might be overparametrised
or that the bounds for the variables are too wide. Try narrower lower and
upper bounds for the variables in the latter case. One of the critical
parameters is 'lowerbound_var_factor' whose value (10000)
might be reduced.
```

```
R> print(pars)
```

```
Internal variables:
      matrix.M.1 matrix.M.2       biwm.s biwm.nudiag.1 biwm.nudiag.2
value  69.633411 0.01922079   96.108534     1.5602533    0.58651829
sd      7.125320 0.02518448   26.930517     0.5263959    0.08361192
lower   1.377946 0.01922079    0.399673     0.1000000    0.10000000
upper 616.236201 8.59579842 4668.895578     4.0000000    4.00000000
      biwm.cdiag.1 biwm.cdiag.2 biwm.rhored
value 6.716103e+04 6.957985e+00  -0.5822896
sd    1.986175e+04 1.339287e+00   0.1117433
lower 1.898735e+00 3.694388e-04  -0.9900000
upper 3.797471e+05 7.388775e+01   0.9900000

User's variables:
      matrix.M.1 matrix.M.2   biwm.s biwm.nudiag.1 biwm.nudiag.2
value  69.633411 0.01922079 96.10853     1.5602533    0.58651829
sd      7.102884 0.18158245 26.59207     0.5051598    0.08366761
      biwm.cdiag.1 biwm.cdiag.2 biwm.rhored
```

```
value      67161.03      6.957985  -0.5822896
sd               Inf      1.297093   0.1051021


       #variab loglikelihood           AIC
         8.000     -1265.737      2547.474
```

Note that the (internal) fitted variable 'matrix.M.2' is close to or on the effective lowe
Hence the gradient of the likelihood function might not be zero and none of the
reported 'sd' values might be reliable.

The main part is table below `User's variables`. Here, the rows deliver the estimated values
and their standard deviations, respectively. Each column name indicates which parameter
is estimated and consists of three parts separated by dots. Let us consider the first name
'`matrix.M.1`'. The left part '`matrix`' refers to the model '`RMmatrix`', the middle part 'M' to
the parameter M of the model and the right part '1' numbers consecutively the entries of the
parameter. In case the parameter is a scalar, no number is attached. The names `matrix.M.1`
and `matrix.M.2` denote the nugget effect of the first and second component, `biwm.s` denotes
the scale s, `biwm.nudiag.1` and `biwm.nudiag.2` refer to ν_{11} and ν_{22}, `biwm.cdiag.1` and
`biwm.cdiag.2` denote c_{11} and c_{22}, and `biwm.rhored` refers to ρ_{red} of the bivariate Whittle-
Matérn model in Section 4.1. As `matrix.M.2` is close to zero, a note is displayed at the end
and the boundaries used in `optim` are given, see the above table after `Internal variables`.

Note that the values found in Gneiting *et al.* (2010) deviate slightly, as the former and the
current optimization algorithm differ. Note also that due to internal representation issues the
values for the estimated standard deviation reported above differ slightly for the '`Internal
variables`'.

As mentioned in Section 3 not only the parsimonious model is fitted, but a sequence of models.
All models within this sequence can be made visible by

```
R> print(pars, full=TRUE)
```

```
indep. multivariate
===================
model 2 , Internal variables:
       matrix2.M      biwm.s biwm.nudiag    biwm.cdiag
value 0.01922079   90.421047   0.5968900 6.744233e+00
sd    0.04061979   28.289452   0.0877261 1.354158e+00
lower 0.01922079    0.399673   0.1000000 3.694388e-04
upper 8.59579842 4668.895578   4.0000000 7.388775e+01


User's variables:
     matrix.M.1 matrix.M.2   biwm.s biwm.nudiag.1 biwm.nudiag.2
value  68.811879       - 88.81451      1.712965             -
sd      7.854676       - 76.65762      1.664701             -
value          - 0.01922079 90.42105             -   0.59689001
sd             - 0.18176523 28.28979             -   0.08779084
     biwm.cdiag.1 biwm.cdiag.2 biwm.rhored
```

```
value      69802.37               -              -
sd          6337.83               -              -
value             -        6.744233              -
sd                -        1.354161              -

        #variab loglikelihood         AIC
          8.000     -1276.746     2569.491
```

Note that the (internal) fitted variable 'matrix.M.2' is close to or on the effective lowe
Hence the gradient of the likelihood function might not be zero and none of the
reported 'sd' values might be reliable.


```
simple multivariate (biwm.rhored = 0)
======================================
Internal variables:
        matrix.M.1 matrix.M.2      biwm.s biwm.nudiag.1 biwm.nudiag.2
value   68.713145 0.01922079   90.325696      1.6805615    0.59704196
sd       6.954941 0.02509845   26.655742      0.6066646    0.08512939
lower    1.377946 0.01922079    0.399673      0.1000000    0.10000000
upper  616.236201 8.59579842 4668.895578      4.0000000    4.00000000
        biwm.cdiag.1 biwm.cdiag.2
value   6.988096e+04 6.740859e+00
sd      2.150969e+04 1.329089e+00
lower   1.898735e+00 3.694388e-04
upper   3.797471e+05 7.388775e+01


User's variables:
        matrix.M.1 matrix.M.2   biwm.s biwm.nudiag.1 biwm.nudiag.2
value   68.713145 0.01922079 90.32570      1.680561    0.59704196
sd       6.952666 0.18173424 26.40195      0.586658    0.08478152
        biwm.cdiag.1 biwm.cdiag.2 biwm.rhored
value      69880.964     6.740859           0
sd          4320.755     1.325465           -

        #variab loglikelihood         AIC
          7.000     -1276.746     2567.491
```

Note that the (internal) fitted variable 'matrix.M.2' is close to or on the effective lowe
Hence the gradient of the likelihood function might not be zero and none of the
reported 'sd' values might be reliable.


```
user's model
============
Internal variables:
        matrix.M.1 matrix.M.2      biwm.s biwm.nudiag.1 biwm.nudiag.2
```

```
value   69.633411 0.01922079    96.108534       1.5602533    0.58651829
sd       7.125320 0.02518448    26.930517       0.5263959    0.08361192
lower    1.377946 0.01922079     0.399673       0.1000000    0.10000000
upper  616.236201 8.59579842  4668.895578       4.0000000    4.00000000
        biwm.cdiag.1 biwm.cdiag.2 biwm.rhored
value  6.716103e+04 6.957985e+00   -0.5822896
sd     1.986175e+04 1.339287e+00    0.1117433
lower  1.898735e+00 3.694388e-04   -0.9900000
upper  3.797471e+05 7.388775e+01    0.9900000


User's variables:
       matrix.M.1 matrix.M.2    biwm.s biwm.nudiag.1 biwm.nudiag.2
value   69.633411 0.01922079  96.10853     1.5602533    0.58651829
sd       7.102884 0.18158245  26.59207     0.5051598    0.08366761
        biwm.cdiag.1 biwm.cdiag.2 biwm.rhored
value      67161.03     6.957985   -0.5822896
sd              Inf     1.297093    0.1051021


        #variab loglikelihood          AIC
          8.000     -1265.737     2547.474
```

Note that the (internal) fitted variable 'matrix.M.2' is close to or on the effective lowe
Hence the gradient of the likelihood function might not be zero and none of the
reported 'sd' values might be reliable.

The first model refers to the independent model. The first two lines of the table after User's
variables correspond to the first component, the last two lines to the second component.
Parameters that have a dash are not estimated for the respective component. In the second
model, the components are also assumed to be independent ($\rho_{\mathrm{red}} = 0$), but share the same
scale parameter. For the fixed parameter ρ_{red}, reporting the standard deviation is not ap-
propriate, indicated by a dash. Here, the independent model, which is the simplest model
concerning parameter estimation, has been fitted first. The fitted parameters serve as initial
values for estimating the parameters of the second model, called simple multivariate model
here. The parameters of the latter are finally used as starting values for the parsimonious
model.

As the parsimonious model has the smallest AIC, it should be chosen among the three models.
This finding is supported by the likelihood ratio test which uses a χ^2 approximation:

R> RFratiotest(pars)

```
Approx. likelihood ratio test
=============================
Null model: (matrix.M.1, matrix.M.2, biwm.s, biwm.nudiag.1, biwm.nudiag.2, biwm.cdiag.1, b
Alt. model: user's model
loglikelihood test: 'Alt. model' against 'Null model': p=2.702e-06 (df=1)
```

Here, the null model is the simple multivariate model and the alternative model is the par-
simonious model. The RFratiotest reports the test statistic only for these two models, as

the parsimonious model and the independent model do not include each other. The latter is true since the parsimonious model is more flexible with regard to the correlation structure while the independent model is more flexible with regard to scaling, cf. the arguments `c` and `s`, respectively.

Following **geoR**, the cross validation

```
R> RFcrossvalidate(pars, x = weather[ , 3 : 4], data = PT, full = TRUE)
```

fits the model only to the complete data set and then the value at each of the 157 locations is predicted by kriging given the data at all the other locations. Again, the argument `full=TRUE` produces information about the internal sequence. The reported deviations are about the same for all three models. Hence, cross validation does not give a clear preferance to any of the three models.

In Gneiting *et al.* (2010), also the whole bivariate Whittle-Matérn model has been considered, cf. Section 4.1. Re-analysis of this model can be done in a similar way as for the parsimonious model,

```
R> whole.model <- nug + RMbiwm(nudiag = c(NA, NA), nured = NA,
+                              s = rep(NA, 3), cdiag = c(NA, NA), rhored = NA)

R> whole <- RFfit(whole.model, distances = Dist.mat, dim = 3, data = PT)

Note: There are some indications that the model might be overparametrised
or that the bounds for the variables are too wide. Try narrower lower and
upper bounds for the variables in the latter case. One of the critical
parameters is 'lowerbound_var_factor' whose value (10000)
might be reduced.

Note: There are very strong indications that the model might be overparametrised
or that the bounds for the variables are too wide. Try narrower lower and
upper bounds for the variables in the latter case. One of the critical
parameters is 'lowerbound_var_factor' whose value (10000)
might be reduced.
```

Now, the likelihood ratio test between the parsimonious and the whole model,

```
R> RFratiotest(nullmodel = pars, alternative = whole)

Approx. likelihood ratio test
=============================
null model: df=8  loglik=-1265.737
alt. model: df=11  loglik=-1265.415
p= 0.9558062
```

yields that the parsimonious model should be preferred.

Finally, Figure 9 shows the kriged field for the parsimonious model that interpolates pressure and temperature. To this end, the earth coordinates have been projected onto the plane.
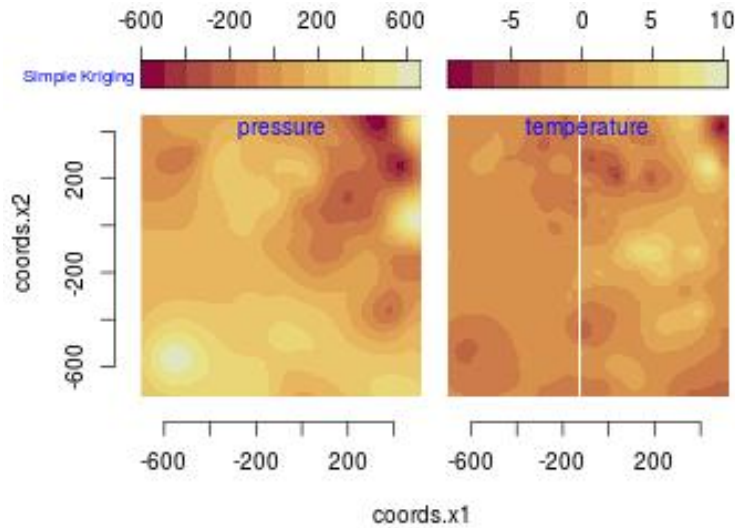
Figure 9: The kriged bivariate random field for the atmospheric data considered in Section 5.

```
R> a <- colMeans(weather[ , 3 : 4]) * pi / 180
R> P <- cbind(c(-sin(a[1]), cos(a[1]), 0),
+              c(-cos(a[1]) * sin(a[2]), -sin(a[1]) * sin(a[2]), cos(a[2])),
+              c(cos(a[1]) * cos(a[2]), sin(a[1]) * cos(a[2]), sin(a[2])))
R> x <- RFearth2cartesian(weather[ , 3 : 4])
R> plane <- (x %*% P)[ , 1 : 2]
```

Subsequently, kriging has been performed on a 200×200 grid in the plane,

```
R> n <- 200
R> r <- apply(plane, 2, range)
R> data <- cbind(plane, weather[ , 1 : 2])
R> z <- RFinterpolate(pars, data = data, dim = 2,
+                     x = seq(r[1, 1], r[2, 1], length = n),
+                     y = seq(r[1, 2], r[2, 2], length = n),
+                     varunits = c("Pa", "K"))
R> plot(z)
```

6. Concluding remarks

Here, the multivariate features of **RandomFields** version 3.0 have been presented. Besides the extension of the tool box for analyzing and simulating random fields, a fundamental goal of version 3.0 of **RandomFields** has been the improvement of the accessibility of the package. To this end, the number of functions and the number of visible arguments have been reduced, at the same time as the functionality of the package has been extended. Models are now defined as functions, and even the "tilde" definition (~) of models can be used, in accordance with the formula specification of linear models. The number of man pages has been largely increased,

offering a separate man page for each model. Further, several summary pages are included, and the documentation of more advanced features is shifted to secondary pages.

Currently, **RandomFields** offers more than 35 simple models and more than 30 functionals operating on simple models to construct more complex ones. Besides multivariate models also space-time models are included. **RandomFields** offers a modular construction system that allows for any combination of models, functionals operating on models, and functions, whenever the combination is mathematically sound. Validity checks of the model are performed by **RandomFields**.

The inference on Gaussian random fields will be extended towards mixed models with a spatial Gaussian component. The package will also advance into the analysis and the modeling of non-Gaussian fields, in particular max-stable random fields.

In Section 5 all the bivariate data have been co-located. This is certainly not the case for many data sets. Now, co-located multivariate data ease algorithms for estimation and spatial prediction. **RandomFields** is being developed towards arbitrary data where missing values are coded by NAs.

Acknowledgment

References

Apanasovich TV, Genton MG (2010). "Cross-Covariance Functions for Multivariate Random Fields Based on Latent Dimensions." *Biometrika*, **97**, 15–30.

Bakar KS, Sahu SK (2013). *spTimer: Spatio-Temporal Bayesian Modelling Using R*. R package version 0.9, URL http://CRAN.R-project.org/package=spTimer.

Berg C, Christensen JPR, Ressel P (1984). *Harmonic Analysis on Semigroups. Theory of Positive Definite and Related Functions*. Springer-Verlag, New York.

Bevilacqua M, Daley DJ, Porcu E, Schlather M (2013). "Classes of Compactly Supported Correlation Functions for Multivariate Random Fields." *Technical report*, Department of Mathematics, Universidad Tecnica Federico Santa Maria, Valparaiso, Chile.

Chan G, Wood ATA (1999). "Simulation of Stationary Gaussian Vector Fields." *Statistics and Computing*, **9**, 265 – 268.

Chilès JP, Delfiner P (1999). *Geostatistics. Modeling Spatial Uncertainty.* John Wiley & Sons, New York, Chichester.

Chorin AJ, Marsden JE (1990). *A Mathematical Introduction to Fluid Mechanics*, volume 2. Springer-Verlag, New York.

Coeurjolly JF (2009). ***dvfBm**: Discrete Variations of a Fractional Brownian Motion.* R package version 1.0, URL http://CRAN.R-project.org/package=dvfBm.

Cressie NAC (1993). *Statistics for Spatial Data.* John Wiley & Sons, New York, Chichester.

De Marsily G (1984). "Spatial Variability of Properties in Porous Media: A Stochastic Approach." In J Bear, M Corapcioglu (eds.), *Fundamentals of Transport Phenomena in Porous Media*, volume 82 of *NATO Science Series*, pp. 719–769. Springer-Verlag.

Dietrich CR, Newsam GN (1993). "A Fast and Exact Method for Multidimensional Gaussian Stochastic Simulations." *Water Resources Research*, **29**, 2861–2869.

Dietrich CR, Newsam GN (1996). "A Fast and Exact Method for Multidimensional Gaussian Stochastic Simulations: Extensions to Realizations Conditioned on Direct and Indirect Measurement." *Water Resources Research*, **32**, 1643–1652.

Finley AO, Banerjee S (2013). ***spBayes**: Univariate and Multivariate Spatial-Temporal Modeling.* R package version 0.3-8, URL http://CRAN.R-project.org/package=spBayes.

French J (2014). ***SpatialTools**: Tools for Spatial Data Analysis.* R package version 0.5.5, URL http://CRAN.R-project.org/package=SpatialTools.

Furrer R, Sain SR (2010). "**spam**: A Sparse Matrix R Package with Emphasis on MCMC Methods for Gaussian Markov Random Fields." *Journal of Statistical Software*, **36**(10), 1–25. URL http://www.jstatsoft.org/v36/i10/.

Gabriel E, Rowlingson B, Diggle P (2013). "**stpp**: An R Package for Plotting, Simulating and Analyzing Spatio-Temporal Point Patterns." *Journal of Statistical Software*, **53**(2), 1–29. URL http://www.jstatsoft.org/v53/i02/.

Gneiting T (1999). "Correlation Functions for Atmospheric Data Analysis." *Quarterly Journal of the Royal Meteorological Society*, **125**, 2449–2464.

Gneiting T, Kleiber W, Schlather M (2010). "Matérn Cross-Covariance Functions for Multivariate Random Fields." *Journal of the American Statistical Association*, **105**, 1167–1177.

Gneiting T, Schlather M (2004). "Stochastic Models That Separate Fractal Dimension and the Hurst Effect." *SIAM Review*, **46**, 269–282. MR2114455.

Goulard M, Voltz M (1992). "Linear Coregionalization Model: Tools for Estimation and Choice of Cross-Variogram Matrix." *Mathematical Geology*, **24**, 269–282.

Guttorp P, Gneiting T (2006). "On the Matérn Correlation Family." *Biometrika*, **93**(4), 989. ISSN 0006-3444.

Hankin RKS (2012). "Introducing **multivator**: A Multivariate Emulator." *Journal of Statistical Software*, **46**(8), 1–20. URL http://www.jstatsoft.org/v46/i08/.

Hofer C, Papritz A (2011). "**constrainedKriging**: An R-package for Customary, Constrained and Covariance-Matching Constrained Point or Block Kriging." *Computers & Geosciences*, **37**(10), 1562–1569.

Kellog OD (1953). *Foundations of Potential Theory*. Dover, New York.

Lantuéjoul C (2002). *Geostatistical Simulation*. Springer-Verlag, Berlin.

Lindsay BG (1988). "Composite Likelihood Methods." In N Prabhu (ed.), *Statistical Inference from Stochastic Processes*, volume 80 of *Contemporary Mathematics*, pp. 221–239.

Matheron G (1972). "Leçon sur les Fonctions Aléatoire d'Ordre 2." *Technical Report C-53*, Ecole des Mines de Paris. URL cg.ensmp.fr/bibliotheque/public/MATHERON_Cours_00302.pdf.

Matheron G (1973). "The Intrinsic Random Functions and Their Applications." *Advances in Applied Probability*, **5**, 439–468.

Melo C, Santacruz A, Melo O, others (2012). *geospt: Spatial Geostatistics; Some Geostatistical and Radial Basis Functions, Prediction and Cross Validation; Design of Optimal Spatial Aampling Networks Based on Geostatistical Modelling*. R package version 0.5-0, URL http://CRAN.R-project.org/package=geospt.

Nychka D, Furrer R, Sain S (2013). *fields: Tools for Spatial Data*. R package version 6.9.1, URL http://CRAN.R-project.org/package=fields.

Oakley J (1999). *Bayesian Uncertainty Analysis for Complex Computer Codes*. Ph.D. thesis, Department of Probability and Statistics, University of Sheffield.

Padoan S, Bevilacqua M (2013). *CompRandFld: Composite-Likelihood Based Analysis of Random Fields*. R package version 1.0.3, URL http://CRAN.R-project.org/package=CompRandFld.

Patankar SV (1980). *Numerical Heat Transfer and Fluid Flow*. Taylor & Francis Group.

Pebesma EJ (2004). "Multivariable Geostatistics in S: The **gstat** Package." *Computers & Geosciences*, **30**, 683–691.

Pope SB (2000). *Turbulent Flows*. Cambridge University Press.

Porcu E, Mateu J, Christakos G (2009). "Quasi-Arithmetic Means of Covariance Functions with Potential Applications to Space-Time Data." *Journal of Multivariate Analysis*, **100**, 1830–1844.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Ribeiro Jr PJ, Diggle PJ (2001). "**geoR**: A Package for Geostatistical Analysis." *R-NEWS*, **1**(2), 14–18. ISSN 1609-3631, URL http://CRAN.R-project.org/doc/Rnews/.

Roustant O, Ginsbourger D, Deville Y (2012). "**DiceKriging**, **DiceOptim**: Two R Packages for the Analysis of Computer Experiments by Kriging-Based Metamodeling and Optimization." *Journal of Statistical Software*, **51**(1), 1–55. URL http://www.jstatsoft.org/v51/i01/.

Rue H (2001). "Fast Sampling of Gaussian Markov Random Fields." *Journal of the Royal Statistical Society B*, **63**(2), 325–338.

Sagaut P, Cambon C (2008). *Homogeneous Turbulence Dynamics*. Cambridge University Press, Cambridge.

Scheuerer M (2011). "An Alternative Procedure for Selecting a Good Value for the Parameter c in RBF-Interpolation." *Advances in Computational Mathematics*, **34**, 105–126.

Scheuerer M, Schlather M (2012). "Covariance Models for Divergence-Free and Curl-Free Random Vector Fields." *Stochastic Models*, **28**, 433–451.

Schilling R, Song R, Vondracek Z (2010). *Bernstein Functions*. De Gruyter, Berlin, New York.

Schlather M (2012). "Construction of Covariance Functions and Unconditional Simulation of Random Fields." In E Porcu, J Montero, M Schlather (eds.), *Advances and Challenges in Space-time Modelling of Natural Events*. Springer-Verlag, New York.

Schlather M, Malinowski A, Menck P, Oesting M, Strokorb K (2014a). "Analysis, Simulation and Prediction of Multivariate Random Fields with Package RandomFields." *J. Statist. Software*. Submitted.

Schlather M, Malinowski A, Oesting M, Boecker D, Strokorb K, Engelke S, Martini J, Ballani F, Menck P, Gross S, Ober U, Burmeister K, Manitz J, Ribeiro P, Singleton R, Pfaff B, R Core Team (2014b). **RandomFields**: *Simulation and Analysis of Random Fields*. R package version 3.0.35, URL http://CRAN.-project.org/package=RandomFields.

Simpson D, Lindgren F, Rue H (2011). "Fast Approximate Inference with **INLA**: The Past, the Present and the Future." *arXiv:1105.2982*.

Stein M (1999). *Interpolation of Spatial Data*. Springer-Verlag, Heidelberg, New York. MR1697409.

Stratton JA (2007). *Electromagnetic Theory*, volume 33. IEEE Computer Society Press.

Wackernagel H (2003). *Multivariate Geostatistics*. 3rd edition. Springer-Verlag, Berlin.

Wendland H (2005). *Scattered Data Approximation*. Cambridge University Press, Cambridge.

Wood ATA, Chan G (1994). "Simulation of Stationary Gaussian Processes in $[0,1]^d$." *Journal of Computational and Graphical Statistics*, **3**, 409–432.

Xie X (2013). **sspline**: *Smoothing Splines on the Sphere*. R package version 0.1-6, URL http://CRAN.R-project.org/package=sspline.

Affiliation:

Martin Schlather
Institute for Mathematics
School of Business Informatics and Mathematics

Universität Mannheim
68131 Mannheim, Germany
E-mail: schlather@math.uni-mannheim.de
URL: http://ms.math.uni-mannheim.de/