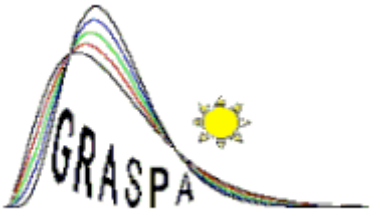
Web Working Papers
by

The Italian Group of Environmental Statistics



Gruppo di Ricerca per le Applicazione della Statistica
ai Problemi Ambientali

www.graspa.org

A general spatio-temporal model for environmental data

Alessandro Fassò and Michela Cameletti

GRASPA Working paper n.27, February 2007

A general spatio-temporal model for environmental data

Alessandro Fassó, Michela Cameletti

Department of Information Technology and Mathematical Methods

University of Bergamo *

Keywords: Separability, Spatial covariance, Kalman filter, Environmental statistics, Distributed computing.

Abstract

Statistical models for spatio-temporal data are increasingly used in environmetrics, climate change, epidemiology, remote sensing and dynamical risk mapping. Due to the complexity of the relationships among the involved variables and dimensionality of the parameter set to be estimated, techniques for model definition and estimation which can be worked out stepwise are welcome. In this context, hierarchical models are a suitable solution since they make it possible to define the joint dynamics and the full likelihood starting from simpler conditional submodels. Moreover, for a large class of hierarchical models, the maximum likelihood estimation procedure can be simplified using the Expectation-Maximization (EM) algorithm.

In this paper, we define the EM algorithm for a rather general three-stage spatio-temporal hierarchical model, which includes also spatio-temporal covariates. In particular, we show that most of the parameters are updated using closed forms and this guarantees stability of the algorithm unlike the classical optimization techniques of the Newton-Raphson type for maximizing the full likelihood function. Moreover, we illustrate how the EM algorithm can be combined with a spatio-temporal parametric bootstrap for evaluating the parameter accuracy through standard errors and non Gaussian confidence intervals.

To do this a new software library in form of a standard `R` package has been developed. Moreover, realistic simulations on a distributed

*Viale Marconi, 5, 24044 Dalmine (BG). E-mail: `alessandro.fasso@unibg.it`

computing environment allows us to discuss the algorithm properties and performance also in terms of convergence iterations and computing times.

1 Software availability

Name: `R` package Stem

Developer: Michela Cameletti

E-mail: michela.cameletti@unibg.it

Software required: `R`

Availability: downloadable from
`http://www.graspa.org/Stem/Stem_1.1.zip`

2 Introduction

Statistical modelling of spatio-temporal data has to take into account various sources of variability and correlation arising from time at various frequencies, from space at various scales, their interaction and other covariates which may be purely spatial quantities or pure time-series without a spatial dimension, or even dynamical fields on space and time. Hierarchical models for spatio-temporal process can cope with this complexity in a straightforward and flexible way. For this reason they are receiving more and more attention from both the Bayesian and frequentist point of view (see, for example, Wikle et al. (1998), Wikle (2003) and **?**), the latter being the approach adopted in this paper.

A hierarchical model can be constructed by putting together conditional submodels which are defined hierarchically at different levels. At the first level the observation variability is modelled by the so-called measurement equation, which is essentially given by a signal plus an error. In the classical approach the true signal or trend is a deterministic function; here, for the sake of flexibility, the trend is a stochastic process which is defined at the subsequent levels of the hierarchy, where the inherent complex dynamics is split into sub-dynamics which, in turn, are modelled hierarchically.

In addition to flexibility, a second advantage of this approach is that we can apportion the total uncertainty to the various components or levels. Moreover, from the likelihood point of view, this corresponds to take a

conditional viewpoint for which the joint probability distribution of a spatio-temporal process can be expressed as the product of some simpler conditional distributions defined at each hierarchical stage.

When the spatio-temporal covariance function satisfies the so-called separability property, these models can be easily represented in state-space form. Hence Kalman filtering and smoothing techniques can be used for reconstructing the temporal component of the unobserved trend (Wikle and Cressie, 1999). For example in environmental statistics, Brown et al. (2001) consider the calibration of radar rainfall data by means of a ground-truth monitoring network and Fassó et al. (2007b) study airborne particulate matter and the calibration of a heterogeneous monitoring network.

Moreover, a separable hierarchical model easily provides a spatial estimator of the Kriging type (Cressie, 1993, Ch. 3) so that a spatio-temporal process, together with its uncertainty, can be mapped in time. For example, Stroud et al. (2001), Sahu et al. (2007), Fassó et al. (2007a), Fassó and Cameletti (2008) propose mapping methods for spatio-temporal data, such as rainfall, tropospheric ozone or airborne particulate matters, which are continuous in space and measured by a monitoring network irregularly distributed in the considered areas.

The Expectation-Maximization (EM) algorithm has been originally proposed for maximum likelihood estimation in presence of structural missing data, see e.g. McLachlan and Krishnan (1997). In spatio-temporal modelling, the EM has been recently used by Xu and Wikle (2007) for estimating certain parameterizations and by Amisigo and Van De Giesen (2005) for the concurrent estimation of model parameters and missing data in river runoff series.

In this paper we propose EM estimation and bootstrap uncertainty assessment for a separable hierarchical spatio-temporal model which generalizes Xu and Wikle (2007) and Amisigo and Van De Giesen (2005) as it covers the case of spatio-temporal covariates. This model class is used for air quality applications in Fassó et al. (2007a) and Fassó and Cameletti (2008), which consider also dynamical mapping and introduce some sensitivity analysis techniques for assessing the mapping performance and understanding the model components. In this framework, using the state-space representation, it is easily seen that temporal prediction is an immediate consequence of Kalman filtering for this model, see e.g. Durbin and Koopman (2001).

The rest of the paper is organized as follows. In Section 3, the above separable spatio-temporal model with covariates is formally introduced.

In Section 4, the EM algorithm is discussed extensively. In particular, we show that the maximization step is based on closed form formulas for all the parameters except for the spatial covariance ones, which are obtained by the

Newton-Raphson (NR) algorithm. Hence, we avoid the inversion of the large Hessian matrix which would arise in performing numerical maximization of the full likelihood.

In Section 5, the spatio-temporal parametric bootstrap is introduced for computing standard errors of the parameter estimates and their confidence intervals. This method turns out to be particularly useful for assessing estimate accuracy, especially in our case which is characterized by asymmetric estimate distributions.

Section 6 is devoted to a simulation study that discusses the performances of the EM algorithm in terms of estimate precision and computing time. This is done using realistic data which are generated on the basis of the airborne particulate matter data set discussed by Cameletti (2007), Fassó et al. (2007a) and Fassó and Cameletti (2008). In particular, subsection 6.1 focuses on the implementation issues with special reference to `R` software and the distributed computing environment while the discussion of the results is provided in subsections 6.2 and 6.3.

The conclusions are drawn in Section 7, while the paper ends with appendixes A and B which contain computational details regarding EM and NR algorithm.

3 The spatio-temporal model

Let $Z(s,t)$ be the observed scalar spatio-temporal process at time t and geographical location s. Let $Z_t = \{Z(s_1,t),\ldots,Z(s_n,t)\}$ [1] be the network data at time t and at n geographical locations $s_1,\ldots,s_n$. Moreover let $Y_t = \{Y_1(t),\ldots,Y_p(t)\}$ be a $p-$dimensional vector for the unobserved temporal process at time t with $p \leq n$. The three-stage hierarchical model is defined by the following equations for $t = 1,\ldots,T$

$$Z_t = U_t + \varepsilon_t \tag{1}$$
$$U_t = X_t\beta + KY_t + \omega_t \tag{2}$$
$$Y_t = GY_{t-1} + \eta_t \tag{3}$$

In equation (1) the measurement error ε_t is introduced so that U_t can be seen as a smoothed version of the spatio-temporal process Z_t. In the second stage the unobserved spatio-temporal process U_t is defined as the sum of three components: a function of the $(n \times d)$-dimensional matrix X_t of d covariates observed at time t at the n locations, the latent space-constant

[1] Here and in the sequel, braces are used for column stacking the vectors involved. Brackets will be used for row stacking instead.

temporal process Y_t and the model error ω_t. It should be noted that the $(n \times p)$-dimensional matrix K is known and accounts for the weights of the p components of Y_t for each spatial location s_i, $i = 1, \ldots, n$. A common choice for K is given by the loadings of a principal component decomposition (see Fassó et al. (2007b) and Wikle and Cressie (1999)). Then in equation (3), the temporal dynamics of Y_t is modelled as a p-dimensional autoregressive process where G is the transition matrix and η_t is the innovation error.

The three error components, namely ε_t, η_t and ω_t, are zero mean and independent over time as well as mutually independent. In particular, the pure measurement error ε_t is a Gaussian white noise process with variance and covariance matrix given by $\sigma_\varepsilon^2 I_n$, where I_n is a n-dimensional identity matrix. The measurement instrument precision σ_ε^2 is supposed constant over space and time as it is the case of a homogeneous network. The case of different instruments belonging to a heterogeneous network is discussed in Fassó et al. (2007b). The innovation η_t of equation (3) is a p-dimensional Gaussian white noise process with variance-covariance matrix Σ_η. Finally, the pure spatial component ω_t of equation (2) is a n-dimensional Gaussian spatial process. It is uncorrelated with ε_t and η_t for each t and its variance-covariance matrix is given by a time-constant spatial covariance function

$$Cov\left[\omega\left(s,t\right),\omega\left(s',t\right)\right] = \sigma_\omega^2 C_\theta\left(h\right)$$

where $h = \|s - s'\|$ is the Euclidean distance between sites s and s'. As the covariance function depends only on h, the spatial process is second-order stationary and isotropic. Moreover, the function $C_\theta\left(h\right)$ depends on the parameter θ to be estimated and is continuous at $h = 0$ with $\lim_{h\to 0} C_\theta\left(h\right) = 1$. A simple example of covariance function is the exponential which is given by

$$C_\theta\left(h\right) = \exp\left(-\theta h\right) \tag{4}$$

Other covariance functions defining isotropic second-order stationary spatial processes are discussed, for example, in Banerjee et al. (2004, Ch. 1).

Substitution of equation (2) into equation (1) yields the following two-stage hierarchical model

$$Z_t = X_t\beta + KY_t + e_t \tag{5}$$
$$Y_t = GY_{t-1} + \eta_t \tag{6}$$

which can be considered as a classical *state-space model* (Durbin and Koopman, 2001), where (5) is the measurement equation and (6) is the state equation. If all the parameters are known, the unobserved temporal process Y_t is estimated for each time point t using the *Kalman filter* and *Kalman smoother*

techniques with initial conditions Y_0 given by a p-dimensional Gaussian vector with mean μ_0 and variance-covariance matrix Σ_0. Note that essentially μ_0 and Σ_0 are nuisance parameters to be considered only as starting values for the Kalman filter algorithm. In the sequel, the Kalman smoother outputs are denoted by y_t^T, P_t^T and $P_{t,t-1}^T$, which are, respectively, the mean, the variance and the lag-one covariance of the Y_t conditional on the complete observation matrix $z = (z_1, \ldots, z_T)$, as defined in Appendix A.

In equation (5) the error $e_t = \omega_t + \varepsilon_t$ has a zero-mean Gaussian distribution with variance-covariance matrix $\Sigma_e = \sigma_\omega^2 \Gamma \left(\| s_i - s_j \| \right)_{i,j=1,\ldots,n}$, where Γ is the following scaled spatial covariance function

$$\Gamma_{\gamma,\theta}(h) = \begin{cases} 1+\gamma & h = 0 \\ C_\theta(h) & h > 0 \end{cases} \tag{7}$$

and $\gamma = \sigma_\varepsilon^2 / \sigma_\omega^2$. It should be pointed out that the measurement error variance σ_ε^2 can be interpreted in geostatistical terms as the so-called "nugget effect" of the spatial process $e(s,t)$ for fixed t. For positive definiteness reasons, it is preferable to estimate $\log(\gamma)$ instead of σ_ε^2.

Hence, the parameter vector to be estimated is given by

$$\Psi = \left\{ \beta, \sigma_\omega^2, vec(G), vecLT(\Sigma_\eta), \mu_0, \log(\gamma), \theta \right\} \tag{8}$$

where vec is the column stacking operator, and $vecLT$ is the vec operator applied to the lower triangular submatrix including the diagonal. It should be noted that $\{\Psi, vecLT(\Sigma_0)\}$ identifies the model (5) and (6). As shown in De Jong (1988), the corresponding loglikelihood function is

$$\log L(\Psi; z) = -\frac{nT}{2} \log(2\pi) + \tag{9}$$
$$-\frac{1}{2} \sum_{t=1}^{T} \left[\log|\Omega_t| + (z_t - \mu_t)' \Omega_t^{-1} (z_t - \mu_t) \right]$$

where $\mu_t = \left(X_t \beta + K y_t^{t-1} \right)$, $\Omega_t = \left(K P_t^{t-1} K' + \Sigma_e \right)$, $y_1^0 = \mu_0$, $P_1^0 = \Sigma_0$ and the symbol $|.|$ is used for matrix determinant. The above given likelihood is a complex and non linear function of the unknown parameter vector and its numerically maximization by means of the classical algorithms of the Newton-Raphson type could be problematic. The adoption of the EM algorithm, which is described in the next section, copes with this problem.

4 The EM algorithm

The maximum likelihood (ML) estimation of the unknown parameter vector Ψ defined by (8) is here performed using the iterative procedure given by the

EM algorithm (McLachlan and Krishnan (1997), Little and Rubin (2002)). This method is particularly useful for missing-data problems including the model defined by (5) and (6), where the missing-data component is given by the latent variable Y_t.

Using the joint approach required by the algorithm, apart for an additive constant, the complete loglikelihood is given by

$$\begin{aligned}
&\log L_c\left(\Psi;\bar{z}\right) \propto -\frac{T}{2}\log\left|\Sigma_e\right| + \\
&-\frac{1}{2}\sum_{t=1}^{T}\left(z_t - X_t\beta - Ky_t\right)'\Sigma_e^{-1}\left(z_t - X_t\beta - Ky_t\right) + \\
&-\frac{1}{2}\log\left|\Sigma_0\right| - \frac{1}{2}\left(y_0-\mu_0\right)'\Sigma_0^{-1}\left(y_0-\mu_0\right) + \\
&-\frac{T}{2}\log\left|\Sigma_\eta\right| - \frac{1}{2}\sum_{t=1}^{T}\left(y_t - Gy_{t-1}\right)'\Sigma_\eta^{-1}\left(y_t - Gy_{t-1}\right)
\end{aligned} \tag{10}$$

where $\bar{z} = \{y_0, \ldots, y_T, z_1, \ldots, z_T\}$ is the complete-data vector.

At each iteration $k = 1, 2, \ldots$ the EM algorithm consists of an expectation (E) and a maximization (M) step. Given the current values of the parameters $\Psi^{(k)}$, the E-step computes the expected value of the complete likelihood function $L_c\left(\Psi;\bar{z}\right)$ conditional on the observation matrix z and $\Psi^{(k)}$ which is given by

$$Q\left(\Psi;\Psi^{(k)}\right) = E_{\Psi^{(k)}}\left[L_c\left(\Psi;\bar{z}\right) \mid z\right]$$

At the M-step the function $Q\left(\Psi;\Psi^{(k)}\right)$ has to be maximized, that is $\Psi^{(k+1)}$ is chosen so that $Q\left(\Psi^{(k+1)};\Psi^{(k)}\right) \geq Q\left(\Psi;\Psi^{(k)}\right)$ for each Ψ.

4.1 E-step

With reference to the complete loglikelihood (10), it is easy to implement the E-step and to compute function $Q\left(\Psi;\Psi^{(k)}\right)$ which is reported in the following equation

$$\begin{aligned}
- \quad & 2Q\left(\Psi;\Psi^{(k)}\right) = -2E_{\Psi^{(k)}}\left[\log L_c\left(\Psi;\bar{z}\right) \mid z\right] \\
= \quad & \tilde{Q} + \log\left|\Sigma_0\right| + tr\left\{\Sigma_0^{-1}\left[\left(y_0^T - \mu_0\right)\left(y_0^T - \mu_0\right)' + P_0^T\right]\right\} \\
+ \quad & T\log\left|\Sigma_\eta\right| + tr\left\{\Sigma_\eta^{-1}\left[S_{11} - S_{10}G' - GS_{10}' + GS_{00}G'\right]\right\}
\end{aligned} \tag{11}$$

where

$$\tilde{Q} = \tilde{Q}\left(\Psi;\Psi^{(k)}\right) = T\log\left|\Sigma_e\right| + tr\left[\Sigma_e^{-1}W\right] \tag{12}$$

and

$$W = \sum_{t=1}^{T} \Big[\left(z_t - X_t\beta - Ky_t^T\right)\left(z_t - X_t\beta - Ky_t^T\right)' + + KP_t^T K' \Big] \quad (13)$$

Moreover $S_{00} = S_{00}^{(k)} = \frac{\sum_{t=1}^{T}\left(y_{t-1}^T y_{t-1}^{T\prime} + P_{t-1}^T\right)}{T}$, $S_{10} = S_{10}^{(k)} = \frac{\sum_{t=1}^{T}\left(y_t^T y_{t-1}^{T\prime} + P_{t,t-1}^T\right)}{T}$ and $S_{11} = S_{11}^{(k)} = \frac{\sum_{t=1}^{T}\left(y_t^T y_t^{T\prime} + P_t^T\right)}{T}$ with the Kalman smoother outputs y_t^T, P_t^T and $P_{t,t-1}^T$ computed using equations of Appendix A and $\Psi^{(k)}$ as the "true" value.

4.2 M-step

Using the so-called conditional maximization approach (McLachlan and Krishnan, 1997, Ch. 5), the solution of $\frac{\partial Q\left(\Psi;\Psi^{(k)}\right)}{\partial\Psi} = 0$ is approximated by partitioning $\Psi = \left\{\check{\Psi}, \tilde{\Psi}\right\}$. The first result is a closed form solution for the component

$$\check{\Psi} = \left\{\beta, \sigma_\omega^2, vec\left(G\right), vecLT\left(\Sigma_\eta\right), \mu_0\right\}$$

holding fixed at its current value the second component $\tilde{\Psi} = \{\theta, \log(\gamma)\}$ and Σ_0 constant. In particular the closed forms are given by

$$\beta^{(k+1)} = \left[\sum_{t=1}^{T}\left(X_t'\Sigma_e^{-1}X_t\right)\right]^{-1}\sum_{t=1}^{T}\left[X_t'\Sigma_e^{-1}\left(z_t - Ky_t^T\right)\right] \quad (14)$$

$$\sigma_\omega^{2\,(k+1)} = \frac{\sigma_\omega^{2\,(k)}}{Tn} tr\left[\Sigma_e^{-1}W\right] \quad (15)$$

$$G^{(k+1)} = S_{10}S_{00}^{-1} \quad (16)$$

$$\Sigma_\eta^{(k+1)} = S_{11} - S_{10}S_{00}^{-1}S_{10}' \quad (17)$$

$$\mu_0^{(k+1)} = y_0^T \quad (18)$$

where $\Sigma_e = \Sigma_e^{(k)}$ and W is given by (13) with $\beta = \beta^{(k+1)}$.

Since there are no closed forms for the remaining parameters $\tilde{\Psi} = \{\theta, \log(\gamma)\}$, the Newton Raphson (NR) algorithm is used for minimizing the quantity $\tilde{Q}$ given by equation (12), considered as a function of $\tilde{\Psi}$ only, that is $\tilde{Q}\left(\tilde{\Psi}\right) = \tilde{Q}\left(\left\{\check{\Psi}^{(k+1)}, \tilde{\Psi}\right\}; \Psi^{(k)}\right)$. So at the generic k^{th} iteration of the EM algorithm, setting $\Psi = \Psi^{(k)}$, the updating formula for the i^{th} iteration of the inner NR

algorithm is given by

$$\tilde{\Psi}_{(i+1)} = \tilde{\Psi}_{(i)} - H^{-1}_{\tilde{\Psi}=\tilde{\Psi}_{(i)}} \times \Delta_{\tilde{\Psi}=\tilde{\Psi}_{(i)}} \tag{19}$$

where H and Δ are, respectively, the Hessian matrix and the gradient vector of $\tilde{Q}\left(\tilde{\Psi}\right)$ evaluated in $\tilde{\Psi} = \tilde{\Psi}_{(i)}$. In Appendix B the complete calculations required for H and Δ are reported together with the details for the exponential covariance function. The formula (19) is repeated until the NR algorithm converges. Hence the obtained root, say $\tilde{\Psi}^{(k+1)}$, is used for the next outer EM iteration based on $\Psi^{(k+1)} = \left\{\check{\Psi}^{(k+1)}, \tilde{\Psi}^{(k+1)}\right\}$.

The EM algorithm converges when the following two convergence criteria are jointly met:

$$\frac{\left\|\Psi^{(k+1)} - \Psi^{(k)}\right\|}{\left\|\Psi^{(k)}\right\|} < \pi$$

and

$$\frac{\left\|\log L\left(\Psi^{(k+1)}; z\right) - \log L\left(\Psi^{(k)}; z\right)\right\|}{\left\|\log L\left(\Psi^{(k)}; z\right)\right\|} < \pi$$

where $\|.\|$ is the Euclidean distance and π is a small positive a priori fixed quantity. The use of these relative criteria instead of some other absolute ones makes it possible to correct for the different parameter scales.

4.3 Closed forms discussion

The closed form (14), which requires the matrix $\sum_{t=1}^{T}\left(X_t'\Sigma_e^{-1}X_t\right)$ to be invertible, corresponds to the generalized least squares estimator for β of the model $Z_t - Ky_t^T = X_t\beta + e_t$. Moreover, using the well known result $E\left(XX'\right) = Var\left(X\right) + \mu\mu'$ that holds for any random vector X with mean vector μ and variance-covariance matrix $Var\left(X\right)$, it can be shown that

$$S_{00} = \frac{1}{T}\sum_{t=1}^{T} E\left(Y_{t-1}Y_{t-1}' \mid z\right)$$

$$S_{10} = \frac{1}{T}\sum_{t=1}^{T} E\left(Y_tY_{t-1}' \mid z\right)$$

So the closed form given by (16) can be seen as an extension of the ML estimate of the transition matrix of a vector autoregressive model of order one (Hamilton, 1994, Ch. 11), where the corresponding *a posteriori* expectation

is used instead of the (unobserved) data product $Y_t Y_{t-1}^{'}$. In the same way, the closed form of the innovation variance estimator can be written as

$$S_{11} - S_{10} S_{00} S_{10}' = \frac{\sum_{t=1}^{T} E\left[(Y_t - GY_{t-1}) (Y_t - GY_{t-1})' \mid z \right]}{T}$$

which can be interpreted as the *a posteriori* innovation variance.

Finally $\sigma_{\omega}^{2\,(k+1)}$ in (15) is positive by definition and is the update of the previous value $\sigma_{\omega}^{2\,(k)}$ by means of the ratio $\Sigma_e^{-1} \frac{W}{T}$, where $\frac{W}{T}$ can be seen as an empirical estimate at step $(k+1)$ of the residual variance-covariance matrix of the measurement equation (5) and Σ_e is the corresponding estimate at the k^{th} previous step.

5 Bootstrapping spatio-temporal data

In this section we discuss a simulation technique for spatio-temporal models which is conditional on the observed data only through $\hat{\Psi}$, $X = (X_1, \ldots, X_T)$ and K. Since the EM algorithm does not use loglikelihood Hessian matrix, it does not provide ready-to-use standard errors of the parameter estimates, as instead the Newton-Raphson type algorithms do. Hence the parametric bootstrap is primarily used in EM estimation for approximating the above standard errors. Moreover, it is used here for parameter uncertainty assessment, including non-Gaussian confidence intervals and analysis of the estimate distributions. In some cases, with reference to a Kriging spatial interpolator, it can be applied for computing map uncertainty and data roughness (Fassó et al., 2007a; Fassó and Cameletti, 2008).

In literature, only purely spatial or temporal bootstrap techniques have been discussed (see, for example, Solow (1985) or the review of Buhlmann (2002)). The spatio-temporal bootstrap introduced here is a sequence of B simulations based on the assumption that the estimated parametric model of Section 3 is the correct one. In particular, we simulate directly from the involved Gaussian distributions and use equations (5) and (6), with Ψ replaced by its ML estimate $\hat{\Psi}$ and the covariates X kept fixed for all the B simulations. In detail, the $b-th$ single bootstrap simulation, for fixed $b = 1, ..., B$ returns one year of spatial data through the following steps:

1. simulate the initial random vector $y_0^{\star}$ from the p-dimensional Gaussian distribution with mean $\hat{\mu}_0$ and variance-covariance matrix Σ_0.

2. For $t = 1, \ldots, T$ repeat the following sub-steps from a) to d):

(a) simulate the random vector $\eta_t^\star$ from the p-dimensional Gaussian distribution with zero mean and variance-covariance matrix given by $\hat{\Sigma}_\eta$.

(b) Use equation (6) to update the latent process, that is

$$y_t^\star = \hat{G} + K y_{t-1}^\star + \eta_t^\star.$$

(c) Simulate the random vector $e_t^\star$ from the d-dimensional Gaussian distribution with zero mean and variance-covariance matrix given by $\hat{\Sigma}_e = \hat{\sigma}_\omega^2 \Gamma_{\hat{\gamma},\hat{\theta}} \left(\| s_i - s_j \| \right)_{i,j=1,...,n}$.

(d) Define the bootstrap observation vector at time t with realistic input X_t by

$$z_t^\star = X_t \hat{\beta} + y_t^\star + e_t^\star.$$

3. Define the generic $b-th$ bootstrap sample by

$$Z_b^\star = \{ z_1^\star, ..., z_T^\star \} = Z\left(\eta^\star, e^\star, X, \hat{\Psi}, K \right)$$

4. Compute the $b-th$ bootstrap estimate replication using data $Z_b^\star$, that is

$$\hat{\Psi}_b^\star = \hat{\Psi}\left(Z_b^\star, X, K \right)$$

using the EM estimation of Section 4.

By repeating this procedure for $b = 1, ..., B$, we get a large sample $\hat{\Psi}_1^\star, \ldots, \hat{\Psi}_B^\star$, which is informative on the sampling distribution of $\hat{\Psi}$. In particular, we have the bootstrap standard errors of each estimate ${}_i\hat{\Psi}$, $i = 1, ..., \dim(\Psi)$, namely ${}_i\hat{s} = \sqrt{\frac{1}{B-1} \sum_{b=1}^{B} \left({}_i\hat{\Psi}_b^\star - {}_i \bar{\Psi}^\star \right)^2}$, where $\bar{\Psi}^\star$ is the bootstrap average estimate.

The choice of the number of replications B is based on the evaluation of two opposite factors: the required computational effort and the precision of the bootstrap approximations. A criterion for choosing B is given by Andrews and Buchinskym (2000) that provide a design-of-experiment method to achieve a desired level of *a priori* accuracy. Along this line, we evaluate B using the *a posteriori* bootstrap accuracy of the standard error estimates given by the length δ of the 95% confidence intervals of the "true"standard error of the parameter estimate, that is ${}_i\sigma = \sqrt{E\left({}_i s^2 \right)}$. These confidence intervals are computed by applying standard sampling theory to the bootstrap

sample $\hat{\Psi}^{\star}_1, \ldots, \hat{\Psi}^{\star}_B$. In particular, for the i-th parameter we have that δ is given by

$$\delta =_i \hat{s}^2 \sqrt{(B-1)} \left(\sqrt{\frac{1}{\chi^2_{\{0.025\}}}} - \sqrt{\frac{1}{\chi^2_{\{0.975\}}}} \right)$$

where the quantities at denominator are, respectively, the quantile of order 0.025 and 0.975 of the χ^2 distribution with $B-1$ degrees of freedom. Therefore the number B of replications is appropriate when δ is small enough.

6 Simulation study

In this section we illustrate the implementation of the EM algorithm together with the spatio-temporal bootstrap and discuss the accuracy of the parameter estimates as well as the algorithm performance under the assumption of correct model specification.

To do this, we start from a set of simulated realistic data. This approach, based on simulated data, makes it possible to control "true"parameters and distributional assumptions and focus more on estimation accuracy and algorithm performances rather than on empirical model interpretation and validation.

The real data set used as a basis for generating the realistic simulated data considers log-transformed particulate matter concentrations (PM_{10}) measured in Piemonte, Italy, during year 2004. These data are discussed by Cameletti (2007), Fassó et al. (2007a) and Fassó and Cameletti (2008). In particular, we have $T = 366$ days and a monitoring network composed of $n = 22$ spatial stations. Moreover, the dynamical multiple field X contains daily data for *Particulate Emissions* both with diameter not exceeding 2.5 μm ($PE_{2.5}$) and between 2.5 and 10 μm (PE_{10}) and *Mixing Height* (MH). The covariate set X also contains the static *Altitude* (A), a dummy variable for *Urban Land use* (UL) and an intercept β_0.

For simulating the realistic data we use the above described X component together with the parameter value, named $\dot{\Psi}$, and the principal component matrix K both estimated using the PM_{10} data. In particular, the simulating reference model used is defined by the set of parameters $\dot{\Psi}$ reported in the first column of Table 1. A four-dimensional latent variable model is used with diagonal persistence matrix $G = diag\,(g_1, \ldots, g_4)$ and incorrelated innovations $\Sigma_\eta = diag\left(\sigma^2_{\eta_1}, \ldots, \sigma^2_{\eta_4}\right)$. It is worth noting that, as the coefficients of the G matrix are lower than one, the model is stationary. Moreover, the spatial covariance function is of the exponential type, defined by equation (4), with $\theta = 0.01$, which corresponds to a spatial correlation of about 0.6 at

a distance of 50 Km.

6.1 Methods and software

The code for estimation and bootstrap has been organized in an object-oriented software library which runs under the open-source statistical platform `R`, see e.g. R Development Core Team (2006), `?` and `?`.

The result is an `R` package called `Stem` from the acronym of Spatio-Temporal EM. This package contains functions for defining and manipulating objects from the class `Stem` which implements on a single computer all the properties of a spatio-temporal model as defined in Section 3. For example, we have function `Stem.Model` for initializing a `Stem` object, function `Stem.Estimation` which implements EM algorithm, function `Stem.Bootstrap` for computing the estimate standard errors and function `Stem.Kriging` for performing dynamical mapping.

The structure of the implemented code is schematically displayed as a flow chart in Fig. 1. It can be seen that each bootstrap iteration (indexed by b) is divided in two principal modules: a) simulation and b) estimation. In particular, the estimation section involves the EM algorithm and requires a variable number of iterations which, in turn, are made up of three sub-modules: a) Kalman filtering and smoothing , b) updating closed forms and c) Newton-Raphson iterations for the spatial covariance parameters.

The stopping threshold, used for the EM and NR algorithm convergence (see subsection 4.2), is $\pi = 10^{-3}$, which is sufficiently small because the corresponding statistical precision given by the parameter empirical coefficients of variation, ${}_i\hat{s}/{}_i\hat{\Psi}$ is not smaller than 10^{-2}.

Actually we performed the spatio-temporal bootstrap using a computer cluster with 4 cpu's of the quad core Intel Xeon processor running at 2.66 GHz and a Linux environment. Parallelizing a bootstrap task is a typical example of the so-called *embarrassingly parallel problems* and the related distributed computing procedure is given by a cluster implementation of `Stem` package together with the `R` packages called `RMPI` and `SNOW`. The first package is an interface to MPI system library (Message-Passing Interface) which is a standardized and portable message-passing system for defining the cluster and the coordination of the node work. The second package, `SNOW`, provides a high-level interface for delivering the job through the cluster.

In order to perform the algorithm analysis and to analyze the computational load involved, for each bootstrap replication, we saved the computing time and iteration numbers of each modules in which the code is divided. The analysis of the next section is performed for different latent process dimensions, namely $p = 4$, 6 and 8.

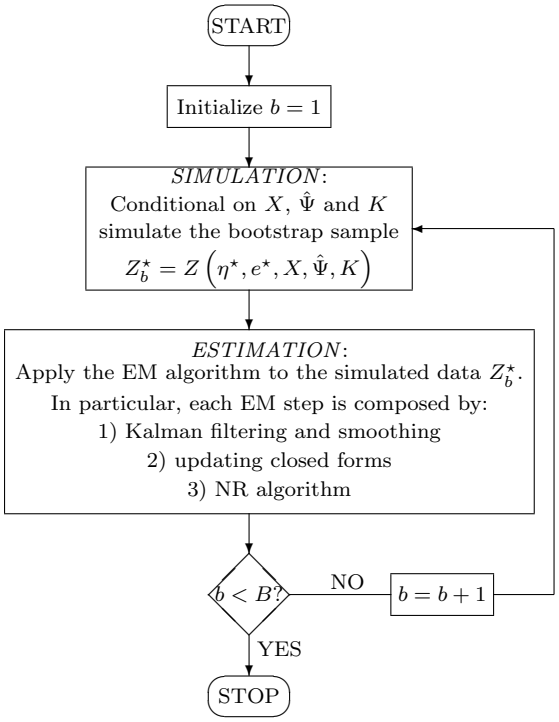


Figure 1: Schematic flowchart of the spatio-temporal bootstrap and EM algorithm code

6.2 Estimate performance

Table 1 reports the obtained ML parameter estimates $\hat{\Psi}$, which are also used as a basis of the $B = 500$ bootstrap replications, together with the corresponding standard errors $\hat{s}$ and 95% confidence interval bounds. It can be observed that, as expected, the ML estimates given by the EM algorithm are close to the true parameters $\dot{\Psi}$ and are characterized by a high level of accuracy. An exception is given by the initial vector μ_0 that, however, can be considered as a purely nuisance parameter and could be discarded from a top level model presentation. This can be seen by recalling that $|\hat{g}_j| < 1$, $j = 1, \ldots, 4$, and y_t is stationary, hence the initial value y_0 and its average μ_0 have an influence on y_t and z_t which decays exponentially fast with t.

For testing the estimate distribution normality, we use the Jarque-Bera test (**?**) whose p-values are reported in Table 1, together with a selection of normal probability plots, reported in Figure 2.

On the one hand there are some parameters, especially among the regression coefficients β that satisfy the normality assumption. On the other hand, a number of other estimates do not fit the normal assumption. Note that,

although maximum likelihood estimates are expected to be asymptotically Gaussian distributed, nongaussianity for fixed sample size may arise partly because the information given by $N \times T = 365 \times 22$ space-time strongly autocorrelated data may be quite smaller than the corresponding information given by the same amount of independent data. This means that spatial and temporal correlation partly reduce sample information and slow down convergence to the asymptotic normality. Moreover, nongaussianity is especially motivated for those parameters which are close to the parameter space borders; in this paper we have the instability border given by $g_j = 1$, for any $j = 1, ..., 4$ and the improper error border given by $\sigma_j^2 = 0$, for any $j = \varepsilon, \omega, \eta_1, ..., \eta_4$.

In such a situation the use of bootstrap confidence intervals, instead of the classical ones, is appropriate and returns informative asymmetric intervals. Considering for example $\hat{g}_1 = 0.95992$, Table 1 gives a meaningful 95% confidence interval. On the other side, if we computed Gaussian based 95% confidence intervals, from Table 1, we would have the unpleasant fact that $\hat{g}_1 \pm 1.96\hat{s} = 0.95992 \pm 1.96 \times 0.02163$ exceeds $g_1 = 1$ and would give the possibility of a non stable model.

From the algorithm reliability point of view, above discussed departure from gaussianity is seen to arise more from skewness and long tails rather than single outliers. This means that no "strange local maxima"have been obtained, that is EM and bootstrap together give a reliable algorithm.

Note that $B = 500$ bootstrap iterations took about an hour with the computer cluster and give satisfactory standard error estimates. This is confirmed by the last column of Table 1 which shows that the a posteriori bootstrap accuracy δ are small enough for our purposes.

6.3 Computational performance

Table 2 and Figure 3 further focus on algorithm performance. First of all, the percentage of cases for which the estimation procedure does not converge is low for all considered p. Note that the rare failures encountered were caused by the non convergence of the NR module. Moreover, as expected, the number of iterations required for the EM convergence increases approximately linearly with p. On the contrary, the NR module iterations required for each EM step does not increase with p because the NR dimension is invariant with p.

Observe that the total time of each EM iteration increases with p and the EM algorithm spends most part of the time (about 60%) for Kalman smoothing computation which is hard to parallelize and can be reduced only by a `C` language compiled routine. Finally, the time required for data sim-

Table 1: Simulation results: "true"parameter vector $\dot{\Psi}$, ML estimate $\hat{\Psi}$, parameter standard error $\hat{s}$, 95% bootstrap confidence interval (CI) bounds, Jarque-Bera test p-value (JBp), a posteriori bootstrap accuracy δ.

	$\dot{\Psi}$	$\hat{\Psi}$	$\hat{s}$	95% CI bounds		JBp	δ
σ^2_ω	0.10	0.09822	0.00518	0.09003	0.10932	0.00338	0.00064
θ	0.01	0.01026	0.00072	0.00888	0.01171	0.00003	0.00009
σ^2_ε	0.01	0.00955	0.00073	0.00831	0.01104	0.29876	0.00009
β_0	3.90191	3.92176	0.04544	3.82501	4.00270	0.20689	0.00566
$PE_{2.5}$	0.00331	0.00312	0.00011	0.00292	0.00333	0.02227	0.00001
PE_{10}	0.05080	0.05225	0.00155	0.04933	0.05559	0.47062	0.00019
MH	-1.09887	-1.12573	0.04007	-1.21955	-1.05404	0.08181	0.00499
UL	-0.29236	-0.28341	0.00972	-0.30232	-0.26355	0.26641	0.00121
A	-0.62618	-0.65517	0.03499	-0.71378	-0.58916	0.00684	0.00436
g_1	0.97	0.95992	0.02163	0.90575	0.98126	0.00000	0.00269
g_2	0.94	0.95955	0.01775	0.91197	0.97825	0.00000	0.00221
g_3	0.72	0.69868	0.04257	0.61156	0.77008	0.00001	0.00530
g_4	0.93	0.95449	0.01782	0.89981	0.97508	0.00000	0.00222
$\sigma^2_{\eta_1}$	0.05	0.08022	0.02134	0.04716	0.12966	0.00121	0.00266
$\sigma^2_{\eta_2}$	0.14	0.15302	0.01575	0.12233	0.18702	0.06784	0.00196
$\sigma^2_{\eta_3}$	0.20	0.19784	0.01622	0.16269	0.23065	0.36686	0.00202
$\sigma^2_{\eta_4}$	0.15	0.15813	0.01518	0.13166	0.19034	0.18700	0.00189
μ_{0_1}	0	1.87361	1.17414	-2.42291	2.10978	0.02022	0.14621
μ_{0_2}	0	2.46252	1.09549	-2.09878	2.16162	0.00199	0.13642
μ_{0_3}	0	0.01116	1.22953	-2.31542	2.55819	0.17976	0.15311
μ_{0_4}	0	0.57224	1.09835	-2.38474	2.06554	0.06271	0.13677

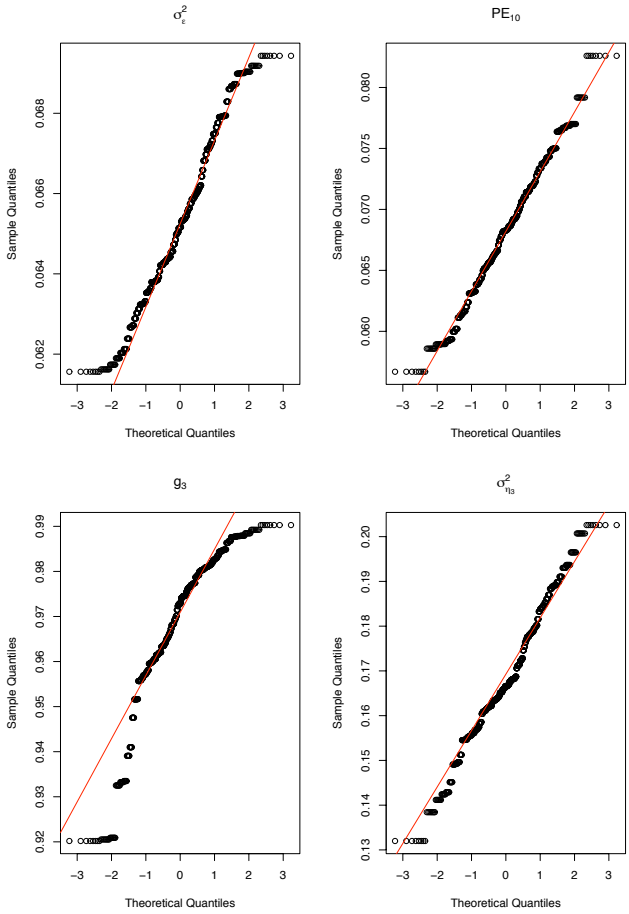


Figure 2: Normal probability plots for some parameters. According to the Jarque-Bera test p-values reported in Table 1, the normal distribution hypothesis can be accepted for σ^2_ε, PE_{10} and $\sigma^2_{\eta_3}$ and refused for g_1.

ulation is negligible being less than 2%, hence the parametric bootstrap is seen to be an efficient simulation method for spatio-temporal data.

7 Conclusions

In this paper we presented the use of the EM algorithm for performing ML estimation of a rather general three-stage hierarchical spatio-temporal model. The application to realistic environmental data shows that the estimation procedure is quite fast and returns accurate estimates. In particular, the estimate precision is obtained by combining the EM algorithm with a spatio-temporal parametric bootstrap. It is worth noting that the resulting boot-

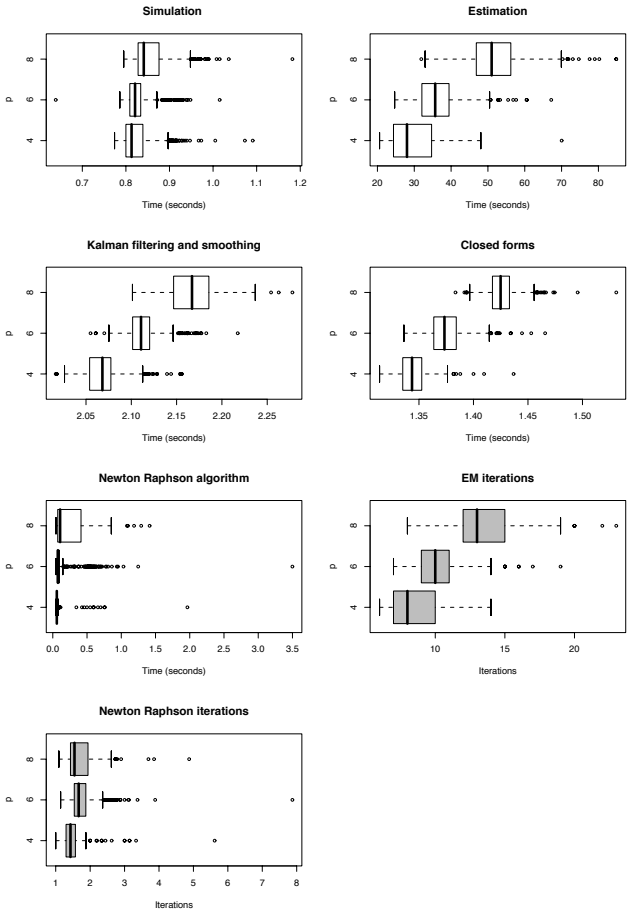


Figure 3: Computational time and iteration box plots.

Table 2: Computational load analysis for $p = 4$, 6 and 8 with B=500.

		$p = 4$	$p = 6$	$p = 8$
Non convergence (%)		0.6	2.2	3.0
	Number of iterations			
EM algorithm	Mean	8.60	9.96	13.50
	75th percentile	10.00	11.00	15.00
	Max	14.00	19.00	23.00
NR algorithm	Mean per EM iteration	1.45	1.76	1.70
	Time (seconds)			
Simulation		0.83	0.83	0.86
Total EM algorithm		30.03	36.37	51.80
Average EM step	Kalman smoothing	2.07	2.11	2.17
	Closed forms	1.34	1.38	1.43
	NR algorithm (total per EM step)	0.08	0.17	0.23
	Total per EM step	3.49	3.66	3.83

strap confidence intervals improve those based on the asymptotic normality because skewed distributions arise as a consequence of the parameter space constraints.

Moreover, it is shown that all the parameters, except those related to the spatial covariance, are updated by closed-form formulas. This means that the stability of the algorithm is enhanced with respect to the classical Newton-Raphson likelihood optimization methods.

We encourage the use of the maximum spatial information available in form of covariates together with a parsimoniously parametrized spatial covariance function. This is because the non convergence percentage, which is quite low, is caused by the failure of the NR algorithm used for updating the parameters of the spatial covariance. Hence if the number of these parameters increases, convergence of the NR module is likely to be problematic.

Computations are made easy by an `R` package called `Stem`, which is object-oriented and has been freshly developed by one of the authors. Moreover, the high computational load can be easily distributed on an IP-based computer cluster with all computers running under Linux and MPI.

8 Acknowledgments

The work is partially supported by PRIN project n.2006131039 *Statistical analysis of spatial and temporal dynamics and health impact of particulate matters* and Regione Piemonte project CIPE 2004 *Statistical methods and spatio-temporal models for atmospheric pollution monitoring*.
The authors thank Ing. Marco Salvi for helping us with the implementation of the distributed computing environment. The authors also thank three anonymous reviewers whose comments greatly improved the quality of this work.

A The Kalman filter and smoother equations

As described in Section 4, the generic k^{th} iteration of the EM algorithm requires the Kalman smoother outputs; here the general updating equations of Durbin and Koopman (2001) are adapted for this computation.
Let y_t^r, P_t^r and $P_{t,t-1}^r$ denote, respectively, the mean, the variance and the lag-one covariance of Y_t $(t = 1, \ldots, T)$ conditional on the observation matrix $(z_1, \ldots, z_r)$ up to time r. The Kalman filter recursion equations for computing the *predicted* values are given by $y_t^{t-1} = Gy_{t-1}^{t-1}$ and $P_t^{t-1} = GP_{t-1}^{t-1}G' + \Sigma_\eta$; the ones for obtaining the *filtered* values are $y_t^t = y_t^{t-1} + A_t\left(z_t - X_t\beta - Ky_t^{t-1}\right)$ and $P_t^t = P_t^{t-1} - A_tKP_t^{t-1}$ where $A_t = P_t^{t-1}K'\left(KP_t^{t-1}K' + \Sigma_e\right)^{-1}$. The initial values are $y_0^0 = \mu_0$ and $P_0^0 = \Sigma_0$.
To get the *smoothed* values the following Kalman smoother recursion formulas are defined backward for $t = T, \ldots, 1$: $y_{t-1}^T = y_{t-1}^{t-1} + B_{t-1}\left(y_t^T - y_t^{t-1}\right)$ and $P_{t-1}^T = P_{t-1}^{t-1} + B_{t-1}\left(P_t^T - P_t^{t-1}\right)B_{t-1}'$ where $B_{t-1} = P_{t-1}^{t-1}G'\left(P_t^{t-1}\right)^{-1}$. The initial values y_T^T and P_T^T are the output of the previously defined Kalman filter equations.
Finally the smoothed lag-one covariance is computed using the following equation $P_{t,t-1}^T = P_t^tB_{t-1}' + B_t\left(P_{t+1,t}^T - GP_t^t\right)B_{t-1}'$ for $t = T-1, \ldots, 1$ where for $t = T$ $P_{T,T-1}^T = (I - A_TK)GP_{T-1}^{T-1}$ holds.

B Derivatives for the Newton-Raphson algorithm

With reference to the Newton-Raphson updating equation (19), the first and second derivatives of the function $\tilde{Q}$ defined by equation (12) with respect to θ and $\log(\gamma)$ are reported here. For notation simplicity let $\Gamma = \Gamma_{\log(\gamma),\theta}(h)$ (see equation (7)) and use ${}_i\tilde{\Psi}$ both for θ $(i = 1)$ and for $\log(\gamma)$ $(i = 2)$.
Using the standard matrix differential rules (see, for example, Harville (1997) and Wand (2002)), it can be shown that

$$\frac{\partial\tilde{Q}}{\partial_i\tilde{\Psi}} = Ttr\left(\Gamma^{-1}\frac{\partial\Gamma}{\partial_i\tilde{\Psi}}\right) - \frac{1}{\sigma_\omega^2}tr\left(\Gamma^{-1}\frac{\partial\Gamma}{\partial_i\tilde{\Psi}}\Gamma^{-1}W\right) \tag{20}$$

where W is given by equation (13). The quantity $\frac{\partial\Gamma}{\partial\theta} = [1 - I_0(h)] \times \frac{\partial C_\theta(h)}{\partial\theta}$ while $\frac{\partial\Gamma}{\partial\log(\gamma)} = I_0(h) \times \exp[\log(\gamma)]$; $I_0(h)$ is the indicator function of zero with $I_0(h) = 1$ for $h = 0$ and zero elsewhere.

For the second order derivatives of $\tilde{Q}$ the following result is obtained

$$\begin{aligned}\frac{\partial^2 \tilde{Q}}{\partial_i \tilde{\Psi} \partial_i \tilde{\Psi}'} = & Ttr\left(\Gamma^{-1}\frac{\partial^2 \Gamma}{\partial_i \tilde{\Psi} \partial_i \tilde{\Psi}'}\right) + \\ & - Ttr\left(\Gamma^{-1}\frac{\partial \Gamma}{\partial_i \tilde{\Psi}}\Gamma^{-1}\frac{\partial \Gamma}{\partial_i \tilde{\Psi}}\right) + \\ & - \frac{1}{\sigma_\omega^2} tr\left(\Gamma^{-1}\frac{\partial^2 \Gamma}{\partial^2{}_i \tilde{\Psi}}\Gamma^{-1} W\right) + \\ & + \frac{2}{\sigma_\omega^2} tr\left(\Gamma^{-1}\frac{\partial \Gamma}{\partial_i \tilde{\Psi}}\Gamma^{-1}\frac{\partial \Gamma}{\partial_i \tilde{\Psi}}\Gamma^{-1} W\right)\end{aligned} \tag{21}$$

where $\frac{\partial^2 \Gamma}{\partial\theta\partial\theta'} = [1 - I_0(h)] \times \frac{\partial^2 C_\theta(h)}{\partial\theta\partial\theta'}$; for $\log(\gamma)$ it holds that $\frac{\partial^2 \Gamma}{\partial^2 \log(\gamma)} = \frac{\partial \Gamma}{\partial \log(\gamma)}$. Finally, considering that $\frac{\partial^2 \Gamma}{\partial \log(\gamma)\partial\theta}$ is a null matrix, the second order mixed derivative is

$$\begin{aligned}\frac{\partial^2 \tilde{Q}}{\partial\theta \partial \log(\gamma)} &= -Ttr\left(\Gamma^{-1}\frac{\partial \Gamma}{\partial\theta}\Gamma^{-1}\frac{\partial \Gamma}{\partial \log(\gamma)}\right) + \\ &+ \frac{2}{\sigma_\omega^2} tr\left(\Gamma^{-1}\frac{\partial \Gamma}{\partial\theta}\Gamma^{-1}\frac{\partial \Gamma}{\partial \log(\gamma)}\Gamma^{-1} W\right)\end{aligned}$$

B.1 The exponential covariance function example

Considering the particular case of equation (4), it is found that the first and second derivatives of Γ with respect to θ are given by the following expressions

$$\frac{\partial \Gamma}{\partial\theta} = [1 - I_0(h)] \times \exp(-\theta h)(-h)$$

$$\frac{\partial^2 \Gamma}{\partial^2\theta} = [1 - I_0(h)] \times \exp(-\theta h)\left(h^2\right)$$

References

Amisigo, B. A., Van De Giesen, N. C., 2005. Using a spatio-temporal dynamic state-space model with the EM algorithm to patch gaps in daily riverflow series. Hydrology and Earth System Sciences 9, 209–224.

Andrews, D., Buchinskym, M., 2000. A three-step method for choosing the number of boostrap repetitions. Econometrica 68 (1), 23–51.

Banerjee, S., Carlin, B., Gelfand, A., 2004. Hierarchical Modeling and Analysis for Spatial Data. Monographs on Statistics and Applied Probability. Chapman and Hall, New York.

Brown, P. E., Diggle, P. J., Lord, M. E., Young, P., 2001. Space-time calibration of radar rainfall data. Journal of the Royal Statistical Society, Series C 50, 221–241.

Buhlmann, P., 2002. Bootstraps for time-series. Statistical Science 17 (1), 52–72.

Cameletti, M., 2007. Modelli spazio-temporali per dati ambientali. Ph.D. thesis, University of Milano Bicocca.

Cressie, N., 1993. Statistics for Spatial Data. Wiley, New York.

De Jong, P., 1988. The likelihood for a state space model. Biometrika 75, 165–169.

Durbin, J., Koopman, S., 2001. Time Series Analysis by State Space Methods. Oxford University Press, New York.

Fassó, A., Cameletti, M., 2008. A unified statistical approach for simulation, modelling, analysis and mapping of environmental data. Submitted.

Fassó, A., Cameletti, M., Bertaccini, P., 2007a. Uncertainty decompositions in environmental modelling and mapping. In: Proceedings of Summer Computer Simulation Conference, San Diego (CA-USA), 15-18 July 2007. pp. 867–874.

Fassó, A., Cameletti, M., Nicolis, O., 2007b. Air quality monitoring using heterogeneous networks. Environmetrics 18, 245–264.

Hamilton, J. D., 1994. Time series analysis. Princeton University Press, New Jersey.

Harville, D., 1997. Matrix Algebra From a Statistician's Perspective. Springer Verlag, New York.

Little, R., Rubin, D., 2002. Statistical Analysis with Missing Data. Wiley, New York.

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

R Development Core Team, 2006. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria.
URL `http://www.R-project.org`

Sahu, S. S., Gelfand, A. E., Holland, D. M., 2007. High-resolution space-time ozone modeling for assessing trends. Journal of the American Statistical Association 102, 1221–1234.

Solow, A., 1985. Bootstrapping correlated data. Mathematical Geology 17 (7), 769–775.

Stroud, J., Muller, P., Sansò, B., 2001. Dymanic models for spatiotemporal data. Journal of the Royal Statistical Society, Series B 63, 673–689.

Wand, M., 2002. Vector differential calculus in statistics. The American Statistician 56 (1), 55–62.

Wikle, C. K., 2003. Hierarchical models in environmental science. International Statistical Review 71, 181–199.

Wikle, C. K., Berliner, L., Cressie, N., 1998. Hierarchical bayesian space–time models. Journal of Environmental and Ecological Statistics 5, 117–154.

Wikle, C. K., Cressie, N., 1999. A dimension-reduced approach to space–time Kalman filtering. Biometrika 86, 812–829.

Xu, K., Wikle, C. K., 2007. Estimation of parameterized spatio-temporal dynamic models. Journal of Statistical Inference and Planning 137, 567–588.