Publication IV

Jarno Vanhatalo, Ville Pietiläinen, and Aki Vehtari. 2010. Approximate inference for disease mapping with sparse Gaussian processes. Statistics in Medicine, volume 29, number 15, pages 1580-1607.

© 2010 John Wiley & Sons

Reprinted by permission of John Wiley & Sons.

Research Article

Statistics in Medicine

Received 12 March 2009, Accepted 4 February 2010 Published online 29 March 2010 in Wiley Interscience

(www.interscience.wiley.com) DOI: 10.1002/sim.3895

Approximate inference for disease mapping with sparse Gaussian processes

Jarno Vanhatalo,*† Ville Pietiläinen and Aki Vehtari

Gaussian process (GP) models are widely used in disease mapping as they provide a natural framework for modeling spatial correlations. Their challenges, however, lie in computational burden and memory requirements. In disease mapping models, the other difficulty is inference, which is analytically intractable due to the non-Gaussian observation model. In this paper, we address both these challenges. We show how to efficiently build fully and partially independent conditional (FIC/PIC) sparse approximations for the GP in two-dimensional surface, and how to conduct approximate inference using expectation propagation (EP) algorithm and Laplace approximation (LA). We also propose to combine FIC with a compactly supported covariance function to construct a computationally efficient additive model that can model long and short length-scale spatial correlations simultaneously. The benefit of these approximations is computational. The sparse GPs speed up the computations and reduce the memory requirements. The posterior inference via EP and Laplace approximation is much faster and is practically as accurate as via Markov chain Monte Carlo. Copyright © 2010 John Wiley & Sons, Ltd.

Keywords: sparse Gaussian process; compact support covariance; expectation propagation; Laplace approximation

1. Introduction

Latent Gaussian models are widely used in disease mapping. Such models assume a latent Gaussian field, which is observed indirectly from noisy areally referenced data. The data are usually realizations of mortality or morbidity counts that are modeled with Poisson distribution. The Poisson rate is a function of a relative risk which is given a spatial prior via a Gaussian field, or a Gaussian process (GP) as it is called here.

There are two major challenges while using latent Gaussian models for disease mapping: (i) the inference is analytically intractable due to the non-Gaussian observation model, which leads to the need for approximate inference schemes and (ii) the inference requires inversion of the covariance matrix, which scales as $O(n^3)$ as a function of data size n. In this paper, we will show how the inference can be conducted efficiently and accurately by combining several distinct approximations.

The techniques considered in this paper are inspired by the approximate inference via expectation propagation (EP) and Laplace approximation in GP classification [1–3], and by the sparse GP approximations in machine learning [4–6]. The aim of the paper is to analyze the techniques in the spatial modeling context and show how they can be used for a detailed statistical analysis. We will consider the Laplace approximation and EP with an approximate integration over the hyperparameters as an alternative to Markov chain Monte Carlo (MCMC), and use the sparse GP approximations to speed up computations. We also describe the practical details of implementation.

GP is a stochastic process that is completely specified by its mean function and covariance function [7–9]. The covariance function governs the smoothness properties of the process, and the effective range of correlation and the variability of the process are governed explicitly by the covariance function parameters. A common choice for reducing the computational cost in

Department of Biomedical Engineering and Computational Science, Aalto University, P.O. Box 12200, FI-00076 Aalto, Finland

**Correspondence to: Jarno Vanhatalo, Department of Biomedical Engineering and Computational Science, Aalto University, P.O. Box 12200, FI-00076 Aalto, Finland.*

†E-mail: jarno.vanhatalo@tkk.fi

Contract/grant sponsor: Academy of Finland
Contract/grant sponsor: Finnish Funding Agency for Technology and Innovation (TEKES)
Contract/grant sponsor: Graduate School in Electronics and Telecommunications and Automation (GETA)
Contract/grant sponsor: Finnish Foundation for Economic and Technology Sciences—KAUTE
Contract/grant sponsor: Finnish Cultural Foundation
Contract/grant sponsor: Emil Aaltonen Foundation

Copyright © 2010 John Wiley & Sons, Ltd.

spatial models has been to restrict the GP to a Gaussian Markov random field (GMRF) [10–13]. A GMRF is specified by assuming a conditional independence (Markov property) between a set of locations. That is, the full conditional of a latent variable depends only on a small number of other latent variables, called the neighbors. The conditional independence assumption leads to a computationally efficient model since the resulting precision matrix is sparse. However, in general, the GMRF models have the feature that the prior is tied to a specified set of locations rather than being defined in a spatially continuous way, and hence, they cannot be used directly to make continuous predictions [8].

An alternative to GMRF is to give computationally efficient approximations for the covariance matrix. In the machine learning literature, these approximations are called *sparse GP approximations*, where the full rank covariance matrix is formed as a function of low rank matrices, and possibly a (block)diagonal matrix [4, 5, 14, 15]. In the spatial modeling context, a similar approach is achieved from different starting points, in which, for example, the spatial process is projected in a lower dimensional subspace, or approximated with a kernel convolution or basis functions [16–19]. Instead of approximating the covariance, one can also use compact support (CS) covariance functions. CS covariance functions give zero correlation among data points that are far enough from each other and form naturally sparse covariance matrices that are computationally efficient to handle [7, 20, 21]. In this work, we will combine sparse approximations and the CS covariance function to construct a computationally efficient additive model. The sparse approximations work well on long length-scale phenomena but fail with short length scale (see Section 4.2), and CS covariance functions are fast only with short length scale. Thus, the proposed model combines the good global properties of sparse approximations and the good local properties of CS covariance functions, keeping the computation time reasonable at the same time.

The traditional approach for an approximate inference in spatial modeling has been MCMC. MCMC methods are computationally very intensive thus motivating the search for alternative techniques. Recently, there has been much work on analytical approximations for GP models. For example, in GP classification with a logit or probit link function, the widely studied methods include variational Bayes approximations [22, 23], Laplace approximation [1, 3], and EP [2, 3]. In these settings, the covariance function parameters are fixed at posterior mode, and the conditional posterior of latent values is given a Gaussian approximation. Nickisch and Rasmussen [24] give an extensive comparison of the different approximation schemes in classification. Recently, in the spatial modeling context, Rue *et al.* [13] have worked on integrated nested Laplace approximation (INLA) for GMRF models. Here, we will show that in disease mapping Laplace approximation and EP give practically identical results with moderate size MCMC sample and that their results can be improved with approximately integrating over the hyperparameters.

This paper is organized as follows. In Section 2, we build the disease mapping model and describe the GP prior. In Section 3, we review the Laplace method and the EP algorithm and discuss the methods to marginalize over the hyperparameters. In Section 4, we review the fully (FIC) and partially (PIC) independent conditional sparse approximations for the GP prior. We also describe how the approximations should be built in the spatial case, and discuss their limitations. The FIC sparse GP model with EP is similar to the work by Cornford *et al.* [17], who considered the sequential method for choosing the inducing inputs, and variational Bayes and EP for the approximate inference. Later, Snelson [25] showed that, in certain circumstances, FIC and sequential design are the same. However, Cornford *et al.* [17] excluded detailed analysis of the sparse approximations and the EP implementation. The use of PIC for spatial data is new to our knowledge. In Section 5, we propose to add FIC with CS covariance function to construct a computationally efficient additive model that is able to capture simultaneously long and short length-scale spatial correlations. In Sections 6 and 7, we confirm our theoretical discussion with experiments, and in Section 8 we discuss the results.

2. Gaussian processes in disease mapping

The disease mapping model constructed in this work follows the general approach discussed, for example, by Best *et al.* [11].

2.1. Disease mapping model

The starting point is to aggregate the data into n areas, with co-ordinates $\mathbf{X}=\{\mathbf{x}_i\}_{i=1}^n$, counts of deaths $\mathbf{y}=\{y_i\}_{i=1}^n$, and background populations $\mathbf{p}=\{p_i\}_{i=1}^n$. The observed mortality is modeled as a Poisson process

$$y_i \sim \text{Poisson}(e_i \mu_i), \tag{1}$$

where e_i is the standardized expected number of deaths and μ_i stands for the relative risk in the area i. The expected number of deaths is evaluated from the explanatory variables of the population and the mortality in the area [26]. Inside each area, the number of deaths and the background population are divided into R groups $\{y_{i,r}, p_{i,r}\}_{r=1}^R$ according to the explanatory covariates (age, gender, and scholarly degree). The standardized expected number of deaths is then evaluated as

$$e_i = \sum_{r=1}^{R} \left(\frac{\sum_{i=1}^n y_{i,r}}{\sum_{i=1}^n p_{i,r}} \right) p_{i,r}. \tag{2}$$

The object in the inference is the relative risk, which is assumed to be similar in areas close to each other. As the relative risk is bounded to be positive, the common practice is to introduce a set of latent variables $\mathbf{f}=\{f_i\}_{i=1}^n$, which are related to the relative risk through a log transformation $f_i=\log(\mu_i)$. The spatial prior is then formalized in the latent space. We will also write

Copyright © 2010 John Wiley & Sons, Ltd.

the likelihood as a function of **f**

$$p(\mathbf{y}|\mathbf{f}) = \prod_{i=1}^{n} p(y_i|f_i) = \prod_{i=1}^{n} \text{Poisson}(y_i|e_i \exp(f_i)), \tag{3}$$

where we have left out **p** for brevity.

2.2. Gaussian process prior

GP is a type of a continuous stochastic process, which defines a probability distribution over functions [7]. Thus, we can model the latent variables as realizations of a latent function $f(\mathbf{x}):\Re^2 \to \Re$ at input locations **X**, and conduct the inference directly in the function space. Giving a GP prior for the latent function implies that the set of latent variables $\mathbf{f} = \{f(\mathbf{x}_i)\}_{i=1}^n$, indexed by the co-ordinates **X**, have a multivariate Gaussian distribution,

$$p(\mathbf{f}|\mathbf{X}) = \text{N}(\mathbf{m}, \mathbf{K}_{\text{f,f}}), \tag{4}$$

where **m** is the mean and $\mathbf{K}_{\text{f,f}}$ the covariance. Here, the prior mean is set to zero, $\mathbf{m} = \mathbf{0}$, which induces prior mean one for the relative risk. Setting $\mathbf{m} = \mathbf{0}$ is justified since the data are empirically standardized with (2), for which reason the relative risk will be one if there are no spatial variations. The assumption in the empirical standardization is that there are enough data so that the uncertainty in the empirical estimate for the country wide risk in different groups is negligible.

The covariance matrix is constructed from a covariance function k, which represents the prior assumptions of the smoothness of the latent function. Each element of the covariance matrix is given as $[\mathbf{K}_{\text{f,f}}]_{ij} = k(\mathbf{x}_i, \mathbf{x}_j, \theta)$, where θ denotes the hyperparameters. We are free in our choice of covariance functions, as long as the covariance matrices produced are symmetric and positive semidefinite ($\mathbf{v}^{\mathrm{T}}\mathbf{K}_{\text{f,f}}\mathbf{v} \geqslant 0, \forall \mathbf{v} \in \Re^n$). An example of a stationary covariance function is the *squared exponential*

$$k_{\text{se}}(\mathbf{x}_i, \mathbf{x}_j) = \sigma_{\text{se}}^2 \exp(-r^2/l^2), \tag{5}$$

where $\theta = \{\sigma_{\text{se}}^2, l\}$, and $r = |\mathbf{x}_i - \mathbf{x}_j|$ is the Euclidean distance between the inputs $\mathbf{x}_i$ and $\mathbf{x}_j$. Here, σ_{se}^2 is the scaling parameter, and l is the length scale, which governs how fast the correlation decreases as the distance increases. The squared exponential is very smooth and, thus, suits well for modeling global (long length-scale) phenomena. Also in spatial statistics, widely used are Matérn covariance functions [7, 8], such as

$$k_{ma}(x_i, x_j) = \sigma_{\text{m}}^2 (1 + \sqrt{3}r) \exp(-\sqrt{3}r). \tag{6}$$

This covariance function is only once mean squared differentiable and thus much rougher than the squared exponential.

An interesting class of covariance functions is the class of compactly supported functions, which form naturally sparse covariance matrices. When used to model short length-scale phenomena, these lead to computational and memory savings since most of the elements in their covariance matrices are zero. A full support covariance function cannot be cut arbitrarily to obtain a CS since the resulting function would not, in general, be positive definite. One class of CS covariance functions are piecewise polynomials, such as

$$k_{\text{pp}} = \sigma_{\text{pp}}^2 (1 - r/l)_+^{j+2} ((j^2 + 4j + 3)(r/l)^2 + (3j + 6)r/l + 3)/3, \tag{7}$$

where $j = \lfloor D/2 \rfloor + 3$ (see for example [7, 20]). The function k_{pp} is positive definite up to the input dimensions D. Gneiting [27] has proposed a method to construct CS covariance functions that preserve the smoothness properties of powered exponential covariance functions, and thus can be used to approximate full support functions.

In many practical situations, a spatial prior with a covariance function with only one length scale might be too restrictive since such a construction can effectively model only one phenomenon. For example, the overall relative risk might vary rather smoothly across the whole area of interest, but at the same time it can have fast local variations. In this case, a more reasonable model would be

$$f(\mathbf{x}) = g(\mathbf{x}) + h(\mathbf{x}), \tag{8}$$

where the latent value function is a sum of two functions, of which one is slowly varying, whereas the other is fast varying. We can place GP prior for both the functions g and h with different covariance functions. The prior for the additive model becomes

$$p(\mathbf{f}|\mathbf{X}) = \text{N}(\mathbf{f}|\mathbf{0}, \mathbf{K}_{\text{g,g}} + \mathbf{K}_{\text{h,h}}). \tag{9}$$

The multiple length-scale model could also be formed using specific covariance functions. For example, the rational quadratic covariance function can be seen as a scale mixture of squared exponential covariance functions [7], and could fit well for data that contain both local and global phenomena. However, using sparse approximations with the rational quadratic would prevent it from modeling local phenomena (see Section 4.2). For this reason, we consider the additive model (9), as it makes possible an accurate and computationally efficient sparse approximation for GP.

In Figure 1, we have plotted the correlation structure of k_{se}, k_{pp}, and $k_{\text{se+pp}}$ covariance functions with one-dimensional input. The figure shows also the latent value functions drawn from GP with these covariance functions.

Copyright © 2010 John Wiley & Sons, Ltd.

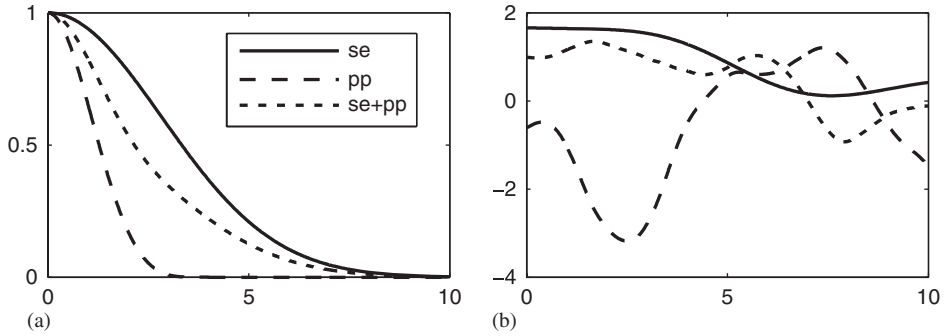

Figure 1. The squared exponential (k_{se}), the piecewise polynomial (k_{pp}), and the additive ($k_{se}+k_{pp}$) covariance function together with samples drawn from GPs using these. The length scale in k_{se} and k_{pp} are four units and the magnitudes are one. In the additive model, the magnitude of the k_{se} part is 0.6 and the magnitude of the k_{pp} part is 0.4: (a) Covariance functions and (b) latent functions drawn from GP with different covariance functions.

2.3. Hyperprior

To finish the model construction, one has to define a prior for the hyperparameters. It is *a priori* plausible that the process variance is zero or very small and that there are no spatial phenomena. Thus, the prior for the covariance function parameters should be such that it enables both the length scale and the magnitude to reach zero. To obtain these characteristics, we give a half-Student's *t*-prior for the covariance function parameters

$$\text{half-}t(\theta_k|\nu,A)\propto\begin{cases}0 & \text{if } \theta_k<0\\ \left(1+\frac{1}{\nu}\left(\frac{\theta_k}{A}\right)^2\right)^{-(\nu+1)/2} & \text{otherwise}\end{cases}$$

where A is the scale, ν the degrees of freedom, and θ_k is either the scaling parameter or the length scale. In case of length scale, this is related to the choice of width of population prior in the hierarchical normal model discussed by Gelman [28].

The half-Student *t*-distribution works as a weakly informative prior. We assume that the spatial correlations are *a priori* short and can adjust the range of our beliefs with the scale parameter A. However, when using small degrees of freedom, for example $\nu=4$, we allow the posterior of the length scale to concentrate into larger values if data justify this. Similarly, the process variance that is mainly governed by σ^2 is assumed to be *a priori* small, but by using small degrees of freedom we allow it to grow if necessary in light of data.

3. Conducting the posterior inference

In this section, we discuss computationally efficient methods to conduct the posterior inference. The treatment will be easier if we concentrate on the latent values. The posterior of relative risk is straightforward to compute afterward. The posterior of the latent values is

$$p(\mathbf{f}|\mathscr{D})=\int p(\mathbf{f},\theta|\mathscr{D})\,\mathrm{d}\theta=\frac{1}{Z}\int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|X,\theta)p(\theta)\,\mathrm{d}\theta, \tag{10}$$

where the normalization constant $Z=\int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|X,\theta)p(\theta)\,\mathrm{d}\theta\,\mathrm{d}\mathbf{f}$, and we have denoted the data by $\mathscr{D}=\{\mathbf{X},\mathbf{y},\mathbf{p}\}$. Neither the normalization constant nor the marginalization are analytically tractable due to the Poisson observation model, and one needs to use approximate methods.

The traditional approach has been to use MCMC methods to sample from the joint posterior of $\mathbf{f}$ and θ. For this, one needs to evaluate the unnormalized joint posterior density, or its logarithm $\log p(\mathbf{f},\theta|\mathscr{D})$, at every Monte Carlo step. The computationally heavy term in that is the log of the GP prior

$$\log p(\mathbf{f}|\mathbf{X},\theta)=-\frac{1}{2}\log|\mathbf{K}_{\mathrm{f,f}}|-\frac{1}{2}\mathbf{f}^{\mathrm{T}}\mathbf{K}_{\mathrm{f,f}}^{-1}\mathbf{f}-\frac{n}{2}\log(2\pi), \tag{11}$$

where the inversion and the determinant of the covariance matrix scale as $O(n^3)$. Even with a perfect MCMC sampler that would produce almost i.i.d. samples one would need to evaluate the log posterior hundreds of times, and in practice the most effective samplers require thousands of evaluations, which effectively restricts the MCMC approach for only moderate size data sets.

In this work, we consider first an empirical Bayes approach, where we give an analytic approximation for the conditional posterior of the latent variables $p(\mathbf{f}|\mathscr{D},\hat{\theta})$ at the posterior mode of the hyperparameters

$$\hat{\theta}=\arg\max_{\theta} p(\theta|\mathscr{D}). \tag{12}$$

Copyright © 2010 John Wiley & Sons, Ltd.

The posterior mode $\hat{\theta}$ is found by giving an analytic approximation $q(\theta|\mathscr{D})$ for $p(\theta|\mathscr{D})$, which is then optimized to its (local) maxima using gradient-based optimization algorithms. Evaluating the density of $q(\theta|\mathscr{D})$ scales also as $O(n^3)$, and it has to be evaluated at each optimization iteration. However, finding $\hat{\theta}$ requires typically only around 10–30 optimization steps, and as will be shown in Section 6.4 this is much faster than MCMC. Two analytic approximations for $p(\mathbf{f}|\mathscr{D},\theta)$ are discussed in Sections 3.1 and 3.2.

We can also (approximately) marginalize over θ by taking a weighted average of the approximate conditional posteriors $q(\mathbf{f}|\mathscr{D},\theta_i)$ at locations θ_i that represent the marginal posterior of hyperparameters $p(\theta|\mathscr{D})$ well enough. The resulting posterior $p(\mathbf{f}|\mathscr{D}) \approx \sum w_i q(\mathbf{f}|\mathscr{D},\theta_i)$ is a mixture of normal distributions with weights w_i. If the evaluation points θ_i are chosen appropriately, we need only a few of them and the computational complexity remains sensible. This kind of an approach has been used, for example, by Rue *et al.* [13] with GMRF models and will be discussed in Section 3.3.

3.1. Laplace method

In this section, we describe an analytic approximation for the conditional posterior of latent values that is based on the Taylor expansion at the mode. Motivated by Williams and Barber [1] and Rasmussen and Williams [7] the approximation scheme is called the Laplace method, but essentially the same approximation is named Gaussian approximation by Rue *et al.* [13].

Doing a second-order Taylor expansion of $\log p(\mathbf{f}|\mathscr{D},\theta)$ around the mode $\hat{\mathbf{f}}$, gives us a Gaussian approximation

$$p(\mathbf{f}|\mathscr{D},\theta) \approx q(\mathbf{f}|\mathscr{D},\theta) = N(\mathbf{f}|\hat{\mathbf{f}},\mathbf{H}),$$

where $\hat{\mathbf{f}} = \arg\max_{\mathbf{f}} p(\mathbf{f}|\mathscr{D},\theta)$ and $\mathbf{H}^{-1} = -\nabla\nabla \log p(\mathbf{f}|\mathscr{D},\theta)|_{\mathbf{f}=\hat{\mathbf{f}}}$ is the Hessian of the negative log conditional posterior at the mode [7, 29]. The Hessian is given by

$$\nabla\nabla \log p(\mathbf{f}|\mathscr{D},\theta) = -\mathbf{K}_{\mathrm{f,f}}^{-1} - \mathbf{W}, \tag{13}$$

where $\mathbf{W}_{ii} = e_i \exp(f_i)$ and $\mathbf{W}_{ji} = 0$ if $i \neq j$.

The mode $\hat{\mathbf{f}}$ is found by minimizing $-\log p(\mathbf{f}|\mathscr{D},\theta)$ with respect to the latent values. This can be done efficiently, for example, via Newton's method [1, 7] using the gradient information $\nabla \log p(\mathbf{f}|\mathscr{D},\theta)$ and the Hessian (13). After evaluating the Hessian at the mode the conditional posterior can be written as

$$N(\mathbf{f}|\hat{\mathbf{f}},(\mathbf{K}_{\mathrm{f,f}}^{-1}+\mathbf{W})^{-1}). \tag{14}$$

To find the (approximate) posterior mode of the hyperparameters the marginal posterior of the hyperparameters is written as $p(\theta|\mathscr{D}) \propto p(\mathscr{D}|\theta)p(\theta)$, where

$$p(\mathscr{D}|\theta) = \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X},\theta)\,\mathrm{d}\mathbf{f}, \tag{15}$$

is the marginal likelihood. The marginal likelihood is analytically intractable, but one can use the Laplace method to find an approximation, $q(\mathscr{D}|\theta)$, for it. This is obtained by doing a Taylor expansion for the log of the integrand in (15) around $\hat{\mathbf{f}}$, which gives a Gaussian integral over $\mathbf{f}$ multiplied by a constant (for details see Appendix A). The hyperparameters can then be optimized by maximizing the approximate log marginal posterior

$$\log p(\theta|\mathscr{D}) \approx \log q(\theta|\mathscr{D}) \propto -\tfrac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}\mathbf{K}_{\mathrm{f,f}}^{-1}\hat{\mathbf{f}} + \log p(\mathbf{y}|\hat{\mathbf{f}}) - \tfrac{1}{2}\log|\mathbf{B}| + \log p(\theta), \tag{16}$$

where

$$|\mathbf{B}| = |\mathbf{K}_{\mathrm{f,f}}||\mathbf{K}_{\mathrm{f,f}}^{-1} + W| = |I + \mathbf{W}^{1/2}\mathbf{K}_{\mathrm{f,f}}\mathbf{W}^{1/2}|.$$

The reason for modifying matrix $\mathbf{B}$ by multiplying $\mathbf{K}_{\mathrm{f,f}}$ by $\mathbf{W}^{1/2}$ from the left and right is to ensure the numerical stability [7]. The gradients of the approximate log marginal posterior (16) can be solved analytically, which enables the use of gradient-based optimization to find $\hat{\theta} = \arg\max_{\theta} q(\theta|\mathscr{D})$. The gradient evaluation is summarized in Appendix B.

3.2. Expectation propagation algorithm

The EP algorithm is a general method for approximating integrals over functions that factor into simple terms [2]. The Laplace method constructs normal approximation at the posterior mode and approximates the posterior covariance via the curvature of the density at that point. EP, for its part, tries to minimize the Kullback–Leibler divergence from the true posterior to its approximation.

We again concentrate on the conditional posterior of the latent values

$$p(\mathbf{f}|\mathscr{D},\theta) = \frac{1}{Z}p(\mathbf{f}|X,\theta)\prod_{i=1}^{N} p(y_i|f_i), \tag{17}$$

which consist of a Gaussian prior, n Poisson terms, and the normalizing constant. This is approximated by EP as

$$p(\mathbf{f}|\mathscr{D},\theta) \approx \frac{1}{Z_{\mathrm{EP}}}p(\mathbf{f}|X,\theta)\prod_{i=1}^{N} t_i(f_i|\tilde{Z}_i,\tilde{\mu}_i,\tilde{\sigma}_i^2) = N(\mathbf{b},\Sigma), \tag{18}$$

Copyright © 2010 John Wiley & Sons, Ltd.

where the Poisson likelihood terms have been replaced by site functions $t_i(f_i|\tilde{Z}_i,\tilde{\mu}_i,\tilde{\sigma}_i^2)=\tilde{Z}_i\mathrm{N}(f_i|\tilde{\mu}_i,\tilde{\sigma}_i^2)$ and the normalizing constant by Z_{EP}.

With fixed hyperparameters, EP algorithm proceeds as follows. First, we initialize the site parameters $\tilde{Z}_i$, $\tilde{\mu}_i$, and $\tilde{\sigma}_i^2$, after which each of them is updated sequentially. At each iteration, first the particular site term is left out of the posterior and the others marginalized out. This results in a cavity distribution

$$q_{-i}(f_i)\propto\int p(\mathbf{f}|X,\theta)\prod_{j\neq i}t_j(f_j|\tilde{Z}_j,\tilde{\mu}_j,\tilde{\sigma}_j^2)\,\mathrm{d}f_j. \tag{19}$$

The second step is to find a Gaussian that best approximates the cavity distribution multiplied by the exact likelihood for that site. The distribution $q_{-i}(f_i)p(y_i|f_i)$ is called tilted posterior marginal. The fit is measured with respect to the Kullback–Leibler divergence, $\mathrm{KL}(q_{-i}(f_i)p(y_i|f_i)||\hat{q}(f_i))$, where $\hat{q}(f_i)$ is the approximating Gaussian. In the case of a Gaussian approximating distribution, the KL divergence is minimized by matching the first and second moments of the approximation to those of the tilted posterior marginal [30]. After the moments of $\hat{q}(f_i)$ are solved, the parameters of the local approximation t_i are updated so that the match with the desired moments is achieved. Finally, the parameters of the approximate posterior (18) are updated: $\Sigma=(\mathbf{K}_{\mathrm{f,f}}^{-1}+\tilde{\Sigma}^{-1})^{-1}$ and $\mathbf{b}=\Sigma\tilde{\Sigma}^{-1}\tilde{\mu}$, where $\tilde{\Sigma}=\mathrm{diag}[\tilde{\sigma}_1^2,\ldots,\tilde{\sigma}_n^2]$. As both the site term and the approximate posterior are Gaussian, the cavity distribution is also Gaussian. Thus, only the posterior marginal variance $\sigma_i^2=\Sigma_{ii}$ and the mean b_i, and the site parameters $\tilde{\sigma}_i^{-2}$ and $\tilde{\mu}_i$ are needed to find the parameters of the cavity distribution at the iteration i (for equations see [7]).

The normalizing constant $Z_{\mathrm{EP}}=\int p(\mathbf{f}|X,\theta)\prod_{i=1}^{N}t(f_i|\tilde{Z}_i,\tilde{\mu}_i,\tilde{\sigma}_i^2)df_i$ is EP's approximation for the marginal likelihood. This has an analytic solution, and its logarithm [7]

$$\log Z_{\mathrm{EP}}=-\frac{1}{2}\log|K+\tilde{\Sigma}|-\frac{1}{2}\tilde{\mu}^{\mathrm{T}}(K+\tilde{\Sigma})^{-1}\tilde{\mu}+\sum_{i=1}^{n}\log z_{-i}+\frac{1}{2}\sum_{i=1}^{n}\log(\sigma_{-i}^2+\tilde{\sigma}_i^2)+\sum_{i=1}^{n}\frac{(\mu_{-i}-\tilde{\mu}_i)^2}{2(\sigma_{-i}^2+\tilde{\sigma}_i^2)}, \tag{20}$$

where z_{-i} is the zeroth moment, μ_{-i} the mean, and σ_{-i}^2 the variance of q_{-i}. This has a fixed solution at the convergence of the EP algorithm and can be differentiated with respect to the hyperparameters [30]. Thus, we are able to find the (approximate) maximum a posterior (MAP) solution for the hyperparameters in similar manner as in the Laplace approximation by maximizing $\log Z_{\mathrm{EP}}+\log p(\theta)$. For exact equations refer to [7].

3.2.1. Computation. The dominant part in the computational time in EP is the update of the posterior covariance. As only one site term is updated at each round of iteration, the covariance can be updated by a rank-one update in time $O(n^2)$. However, since there are n local approximations to be updated this results in a total of $O(n^3)$ computational time [7], in addition to computing the marginal likelihood approximation (20) and its derivatives scale as $O(n^3)$ due to the $n\times n$ matrix $\mathbf{K}_{\mathrm{f,f}}$.

In general, the moments of the tilted posterior marginal are analytically unsolvable, but can be evaluated, for example, by quadrature integration. Gaussian quadrature (also used by Zoeter and Heskes [31]) proved to be the most efficient method since the distributions $q_{-i}(f_i)p(y_i|f_i)$ are very close to Gaussian. In this work, the integration was done with the adaptive Gauss-Kronrod quadrature, which gives the solution approximately in user defined absolute and relative error accuracy [32]. The absolute accuracy was set to 10^{-8} and the integration limits to $[\mu-6\sigma,\mu+6\sigma]$, where μ and σ are the mean and standard deviation of the normal approximation of $q_{-i}(f_i)p(y_i|f_i)$. The normal approximation was obtained by using a Laplace approximation for the likelihood term and evaluating the variance and the mean of the product of two Gaussians. This is fast to compute since all the parameters can be solved analytically.

3.3. Marginalization over the hyperparameters

The Laplace method and the EP algorithm give an approximation for the conditional posterior $p(\mathbf{f}|\mathscr{D},\theta)$ (equations (14) and (18)) and up to a normalization for the marginal posterior $p(\theta|\mathscr{D})$ (see equations (16) and (20)). Using the approximations $q(\theta|\mathscr{D})$ we can explore the log marginal posterior of the hyperparameters to conduct a numerical integration over θ. In this section, we will consider three approaches for this, a grid search similar to the approach used by Rue *et al.* [13], an approximate Monte Carlo integration, and a central composite design (CCD) [13].

With every method, we first transform the hyperparameters through a logarithm and conduct the integration over $\gamma=\log\theta$. The reason for the transformation is that the posterior $p(\gamma|\mathscr{D})$ is usually closer to Gaussian than $p(\theta|\mathscr{D})$ and that $\gamma\in\Re$, whereas the length scale and the magnitude are bounded to be positive.

3.3.1. Grid search. The marginal posterior of the latent variables can be approximated with a finite sum

$$p(\mathbf{f}|\mathscr{D})\approx\sum_{i=1}^{M}p(\mathbf{f}|\mathscr{D},\gamma_i)p(\gamma_i|\mathscr{D})\Delta_i, \tag{21}$$

over the values γ_i with area weights Δ_i. The first step in exploring $\log p(\gamma|\mathscr{D})$ is to find its posterior mode $\hat{\gamma}$ using the Laplace method or EP. After this we evaluate the negative Hessian of $\log p(\gamma|\mathscr{D})$ at the mode, which would be the inverse covariance

Copyright © 2010 John Wiley & Sons, Ltd.

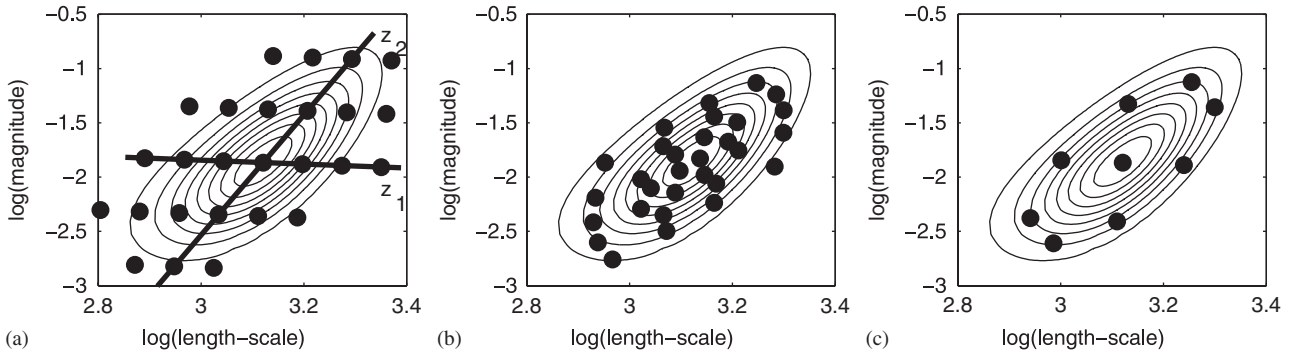

Figure 2. Illustration of the grid-based, the Monte Carlo and the central composite design integration over the hyperparameters. The contour shows the posterior density $q(\gamma|\mathcal{D})$ evaluated with Laplace approximation. On the left is a grid of points over which $q(\mathbf{f}|\mathcal{D},\gamma)$ is integrated. The left figure also shows the transformed parameter vectors $\mathbf{z}$ along which the grid points are searched. In the middle are plotted the proposal samples from the normal approximation for the posteriors that were used in the importance sampling scheme. On the right are the design points from the central composite design scheme. The posterior surface in the figures is from the simul200 data set discussed in Section 6.1. As can be seen the posterior of the log hyperparameters $p(\gamma|\mathcal{D})$ is very close to normal and thus the integration methods described here should presumably work rather well: (a) Grid-based integration; (b) Monte Carlo integration using importance sampling; and (c) integration using central composite design.

matrix for γ if the density were Gaussian. The exploration is aided using standardized variables $\mathbf{z}$. If $\mathbf{P}$ is the inverse Hessian (the approximate covariance) with eigendecomposition $\mathbf{P}=\mathbf{U}\mathbf{C}\mathbf{U}^{\mathrm{T}}$ then γ can be defined via $\mathbf{z}$ as

$$\gamma(\mathbf{z})=\hat{\gamma}+\mathbf{U}\mathbf{C}^{1/2}\mathbf{z}. \tag{22}$$

If $p(\gamma|\mathcal{D})$ were a Gaussian density, then $\mathbf{z}$ would be zero mean normal distributed. This re-parameterization corrects for scale and rotation and simplifies the integration [13].

The exploration of $\log p(\gamma|\mathcal{D})$ is started from the mode $\hat{\gamma}$ and continued so that the bulk of the posterior mass is included in the integration. The grid points are set evenly along the directions $\mathbf{z}$, for which reason the area weights Δ_i are equal. This is illustrated in Figure 2. The parameters to tune in the grid search are the step size δ_i along direction i and the difference $\delta_p=\log p(\hat{\gamma}|\mathcal{D})-p(\gamma(\mathbf{z})|\mathcal{D})$ which controls the area included in the integration. Rue *et al.* [13] suggest values $\delta_i=1$ and $\delta_p=2.5$, which seem to provide accurate enough integration.

The numerical integration using the grid search is feasible only for a small number of hyperparameters since the number of grid points grows exponentially with the dimension of the hyperparameters space d. For example, the number of nearest neighbors of the mode increases to $O(3^d)$, which results in 728 grid points already for $d=6$. If we take also the second neighbors, the number of grid points increases as $O(5^d)$, which results in 15 624 grid points for six hyperparameters.

3.3.2. Monte Carlo integration. We can use Laplace method's and EP's marginal likelihood estimate to sample the hyperparameters from their (approximate) marginal posterior $q(\gamma|\mathcal{D})$ with standard MCMC methods like, for example, hybrid Monte Carlo [33]. During the sampling, we can save the conditional posterior approximations $q(\mathbf{f}|\mathcal{D},\gamma)$ and make the final inference by taking the average over several normal distributions. As the latent values are marginalized out the sample chain will converge and mix considerably faster than when sampling the full posterior $p(\mathbf{f},\gamma|\mathcal{D})$. This approach is reasonable with a moderate data size, n, but for large data sets this will be computationally too heavy since the Markov chain samples are dependent on each other. Thus, in order to speed up the marginalization, we need more efficient ways than Markov chain to sample from the posterior. In optimal situations we could draw independent samples, but since this is impossible we suggest to use importance sampling with quasi Monte Carlo samples (see, for example [34, 35]).

We use a Normal or Student t-distribution centered at the mode $\hat{\gamma}$ and covariance $\mathbf{P}$ approximated with the negative Hessian of the log posterior as a proposal distribution. We sample from the proposal distribution $g(\gamma)\sim\mathrm{N}(\hat{\gamma},\mathbf{P})$ (or $g(\gamma)\sim t_\nu(\hat{\gamma},\mathbf{P})$) and evaluate the integral as follows:

$$p(\mathbf{f}|\mathcal{D})\approx\frac{1}{\sum_{i=1}^{M}w_i}\sum_{i=1}^{M}q(\mathbf{f}|\mathcal{D},\gamma_i)w_i, \tag{23}$$

where $w_i=q(\gamma^{(i)})/g(\gamma^{(i)})$ are the importance weights. The normal proposal distribution is also illustrated in Figure 2(b). Importance sampling is adequate only if the importance weights do not vary substantially and, thus, the goodness of the Monte Carlo integration can be monitored using the importance weights. The worst scenario occurs when the importance weights are small with high probability and with small probability get very large values (that is the tails of q are much wider than those of g). We monitor the cumulative normalized weights and the estimate of the effective sample size $M_{\mathrm{eff}}=1/\sum_{i=1}^{M}\hat{w}_i^2$, where $\hat{w}_i=w_i/\sum w_i$ [29, 36, 37].

In some situations, the naive Gaussian or Student-t proposal distribution is not adequate since the posterior distribution $q(\gamma|\mathcal{D})$ may be nonsymmetric or the covariance estimate $\mathbf{P}$ is poor. In these situations, we found the scaled Student-t proposal distribution, proposed by Geweke [36], to be efficient. In this approach, the scale of the proposal distribution is adjusted along

Copyright © 2010 John Wiley & Sons, Ltd.

each main direction defined by **P** so that the importance weights are maximized. In the experiments, this scaling more than doubled the number of effective samples.

The key property of a random sequence is its uniformity, so that any contiguous sub-sequence is well spread according to the distribution. Drawing independent and identically distributed samples from the Normal or Student-*t* distribution is straightforward but not the most optimal way in the sense of uniformity. For this reason, we suggest to use Quasi Monte Carlo algorithms [35], which utilize quasi random samples like Hammersley sequences [38] that are distributed more uniformly in the parameter space.

The rate at which the error in Monte Carlo integration decreases, is independent of the dimensionality of the hyperparameter vector and, thus, the importance sampling procedure is more efficient than grid search in high parameter dimensions. However, as discussed earlier one still needs hundreds of i.i.d. samples from the posterior to get reliable results. In the next section, we discuss one more approach to approximate the integrals.

3.3.3. Central composite design. Rue *et al.* [13] suggest a CCD for choosing the representative points from the posterior of the hyperparameters when the dimensionality d is moderate or high. In this setting, the integration is considered as a quadratic design problem in a d dimensional space with the aim in finding points that allow for estimating the curvature of the posterior distribution around the mode. Here, we use fractional factorial design points augmented with a center point and a group of $2d$ star points. The design points are all on the surface of a d-dimensional sphere with radius $\sqrt{d}$ and the star points consist of $2d$ points along each axis at a distance $\pm\sqrt{d}$. This is illustrated in Figure 2(c). The number of the design points grows very moderately, for example for $d=6$ we need only 45 points. The fractional factorial design is discussed in detail by Sanchez and Sanchez [39].

The design points are searched after transforming γ into $\mathbf{z}$ space, which is assumed to be a standard Gaussian variable. The integration weights can then be determined from the statistics of a standard Gaussian variable $E[\mathbf{z}^{\mathrm{T}}\mathbf{z}]=d$, $E[\mathbf{z}]=\mathbf{0}$, and $E[1]=1$. This results in integration weights

$$\Delta=\left[(n_p-1)\exp\left(-\frac{df_0^2}{2}\right)(f_0^2-1)\right]^{-1} \tag{24}$$

for the points on the sphere and $\Delta_0=1$ for the central point (see Appendix D for a more-detailed derivation). The CCD integration speeds up the computations considerably compared to the grid search or Monte Carlo integration. The accuracy is between the empirical Bayes estimate and the full integration with grid search or Monte Carlo. Rue *et al.* [13] and Martino [40] report good results with this integration scheme, and it also worked well in our experiments. For a more-detailed treatment of the method, see [40].

3.4. Assessing the approximation error

There are three sources for approximation error: the normal approximation of the conditional posterior $q(\mathbf{f}|\mathscr{D},\theta)$, the approximate marginal posterior $q(\theta|\mathscr{D})$, and the numerical integration over θ. We will show in Section 6.1 that in practice all these approximations are very accurate. However, we need a way to asses the approximations so that we can detect possible problems.

The Laplace approximation for $q(\theta|\mathscr{D})$ (16) is up to a normalization formally equivalent to the Laplace approximation of the marginal posterior density that was proposed by Tierney and Kadane [41], which has error rate $O(n^{-1})$. The asymptotics of the error of the Laplace approximation is also discussed by Rue *et al.* [13]. They conclude that the accuracy of $q(\theta|\mathscr{D})$ in (16) is related to the effective number of parameters $p_D(\theta)$ in $\mathbf{f}$ conditionally on θ so that the approximation is the better when the smaller $p_D(\theta)$ is compared to the number of data points, n. The effective number of parameters can be approximated with [42]

$$p_D(\theta)\approx n-\mathrm{tr}(\mathbf{K}_{\mathrm{f,f}}^{-1}(\mathbf{K}_{\mathrm{f,f}}^{-1}+\mathbf{W})^{-1}), \tag{25}$$

which can be used to monitor the accuracy of the normal approximation for the conditional posterior. For EP approximation this can be evaluated by replacing $\mathbf{W}$ with $\tilde{\Sigma}$. The number of effective latent variables also measures to what extent the prior correlations are preserved in the posterior of the latent variables given θ. For non-informative data $p_D(\theta)=0$ and the posterior approximation is exact. In order to assess the goodness of the conditional posteriors we can evaluate $p_D(\theta_i)$ for each θ_i in the integration set. In our experiments, $p_D(\theta_i)$ has been small relative to the number of data points n (typically $0.04n$–$0.1n$). The EP's approximation for $p(\theta|\mathscr{D})$ is studied in length by Nickisch and Rasmussen [24] in classification setting and more generally by Paquet *et al.* [43] who also provide the means to evaluate correction terms for Z_{EP}. The results reported so far suggest that the EP estimate for the marginal likelihood $q(\theta|\mathscr{D})$ should be more accurate than the one from the Laplace approximation, since EP matches the first two moments of the approximation with the real posterior. However, if one doubts the accuracy of the approximation it is possible to evaluate few of the correction terms to see if the results are altered.

The error due to the numerical integration over θ can be monitored by comparing results with the increasing number of integration points. We can monitor whatever statistics we think is important, for example mean or variance or the KL divergence. In practice different statistics converge at different rates, for example estimating the mean $E[\mathbf{f}|\mathscr{D}]$ requires less integration points than the variance $\mathrm{Var}[\mathbf{f}|\mathscr{D}]$.

The most crucial approximation is the assumption of the Gaussian form for $p(f|\mathscr{D},\theta)$. If this does not hold we will be able to infer well only the posterior mean and variance of the latent variables at best. To check for the normality of the marginal posteriors $p(f_i|\mathscr{D},\theta)$ we can again use the perturbation corrections for the EP solution [43]. The corrections do not change the first and

Copyright © 2010 John Wiley & Sons, Ltd.

second moments of the approximation since these are consistent in EP but they correct the higher cumulants. An easy validation of the normality is to evaluate the first-order correction, since this can be evaluated with terms that are already computed during the EP algorithm. One might be tempted to use the correction terms in any situation, since this would provide more accurate results. The drawback, however, is that the corrected posterior approximation is no longer Gaussian and, for example, prediction to new point becomes harder.

All the methods to assess the approximations can be checked routinely. In our implementation these are checked automatically and the user is warned if something seems to be wrong.

4. Sparse approximations to Gaussian process

In the previous section, we discussed how the inference can be sped up from MCMC via the analytic approximations. There, the speedup was obtained by the reduction in the number of function evaluations, but the methods still scaled as $O(n^3)$ in number of data n. Next, we review two sparse approximations for the GP prior that reduce the dependency of the computational time to data from $O(n^3)$ to $O(nm^2)$, where $m \ll n$, and analyze their properties from the spatial analysis point of view.

4.1. Fully and partially independent conditional

Here, we give a short review of the FIC and PIC sparse approximations. Readers interested in detailed derivations of the approximations should refer to the original papers [4, 44] or for a more general perspective on the unifying view on sparse approximations [5].

The approximations are based on introducing an additional set of latent values $\mathbf{u}=\{u_i\}_{i=1}^m$, called *inducing variables*, that correspond to a set of input locations $\mathbf{X}_u$, *inducing inputs*. The latent value prior is approximated as

$$p(\mathbf{f}|\mathbf{X}) \approx q(\mathbf{f}|\mathbf{X},\mathbf{X}_u) = \int q(\mathbf{f}|\mathbf{X},\mathbf{X}_u,\mathbf{u})p(\mathbf{u}|\mathbf{X}_u)\,\mathrm{d}\mathbf{u}, \tag{26}$$

where $\mathbf{f}$ is interpreted to be conditional on $\mathbf{u}$ through the inducing conditional $q(\mathbf{f}|\mathbf{X},\mathbf{X}_u,\mathbf{u})$. The above decomposition does not in itself entail any approximation if we were to use the exact conditional $p(\mathbf{f}|\mathbf{u})=\mathrm{N}(\mathbf{K}_{\mathrm{f,u}}\mathbf{K}_{\mathrm{u,u}}^{-1}\mathbf{u}, \mathbf{K}_{\mathrm{f,f}}-\mathbf{Q}_{\mathrm{f,f}})$, where $\mathbf{Q}_{\mathrm{a,b}}=\mathbf{K}_{\mathrm{a,u}}\mathbf{K}_{\mathrm{u,u}}^{-1}\mathbf{K}_{\mathrm{u,b}}$ and $[\mathbf{K}_{\mathrm{f,u}}]_{ij}=\mathbf{k}(\mathbf{x}_i,[\mathbf{X}_u]_j)$. However, in FIC framework the latent values are assumed to be conditionally independent given $\mathbf{u}$, in which case the inducing conditional factorizes into $q(\mathbf{f}|\mathbf{X},\mathbf{X}_u,\mathbf{u})=\prod q_i(f_i|\mathbf{X},\mathbf{X}_u,\mathbf{u})$. In PIC, instead of treating all the latent values conditionally independent of each other, they are set into blocks and the blocks are conditionally independent of each other. The latent values within a block have a multivariate normal distribution with original covariance, but the correlation between the blocks is zero given the inducing variables. The approximate conditional of both FIC and PIC can be summarized as

$$q(\mathbf{f}|\mathbf{X},\mathbf{X}_u,\mathbf{u})=\mathrm{N}(\mathbf{K}_{\mathrm{f,u}}\mathbf{K}_{\mathrm{u,u}}^{-1}\mathbf{u}, \mathrm{mask}(\mathbf{K}_{\mathrm{f,f}}-\mathbf{Q}_{\mathrm{f,f}},\mathbf{M})),$$

where the function $\mathrm{mask}(\mathbf{K}_{\mathrm{f,f}}-\mathbf{Q}_{\mathrm{f,f}},\mathbf{M})$, with matrix $\mathbf{M}$ of ones and zeros, returns a matrix Λ of size $\mathbf{M}$ and elements

$$\Lambda_{ij}=\begin{cases}[\mathbf{K}_{\mathrm{f,f}}-\mathbf{Q}_{\mathrm{f,f}}]_{ij} & \text{if } \mathbf{M}_{ij}=1\\ 0 & \text{otherwise}\end{cases}$$

An approximation with $\mathbf{M}=\mathbf{I}$ corresponds to FIC and an approximation where $\mathbf{M}$ is block diagonal and corresponds to PIC.

The inducing inputs are given a zero-mean Gaussian prior $\mathbf{u}\sim\mathrm{N}(\mathbf{0},\mathbf{K}_{\mathrm{u,u}})$ so that the approximate prior over the latent values is

$$q(\mathbf{f}|\mathbf{X},\mathbf{X}_u)=\mathrm{N}(\mathbf{0},\mathbf{Q}_{\mathrm{f,f}}+\Lambda), \tag{27}$$

where the matrix $\mathbf{Q}_{\mathrm{f,f}}$ is of rank m and $\Lambda=\mathrm{mask}(\mathbf{K}_{\mathrm{f,f}}-\mathbf{Q}_{\mathrm{f,f}},\mathbf{M})$ is a sparse matrix of size $n\times n$. This leads to computational savings since the crucial computations only require $O(nm^2)$. The other advantage is the reduction in memory requirements from $O(n^2)$ to $O(nm)$.

Intuitively, PIC approaches FIC in the limit of a block size one and the exact GP in the limit of a block size n. A formal treatment of this is given by Snelson [25]. We have made explicitly visible the dependence of the approximate prior $q(\mathbf{f}|\mathbf{X},\mathbf{X}_u)$ on the inducing inputs because, although the inducing variables are marginalized out from the approximate prior, the approximation still depends on the choice of the locations of the inducing inputs.

4.2. Properties of the approximations

The prior covariance of FIC and PIC can be seen as a non-stationary covariance function of its own

$$k_{\mathrm{FIC/PIC}}(k(x_i,x_j),\mathbf{M},\mathbf{X}_u)=\begin{cases}[\mathbf{K}_{\mathrm{f,f}}]_{ij}=k(\mathbf{x}_i,\mathbf{x}_j) & \text{if } \mathbf{M}_{ij}=1\\ [\mathbf{Q}_{\mathrm{f,f}}]_{ij}=[\mathbf{K}_{\mathrm{f,u}}\mathbf{K}_{\mathrm{u,u}}^{-1}\mathbf{K}_{\mathrm{u,f}}]_{ij} & \text{otherwise}\end{cases} \tag{28}$$

where the inducing inputs $\mathbf{X}_u$ and the matrix $\mathbf{M}$ are free parameters similar to hyperparameters. In the works by Snelson and Ghahramani [4, 44] and Lawrence [45], the inducing inputs were treated as extra hyperparameters, which were optimized alongside

Copyright © 2010 John Wiley & Sons, Ltd.

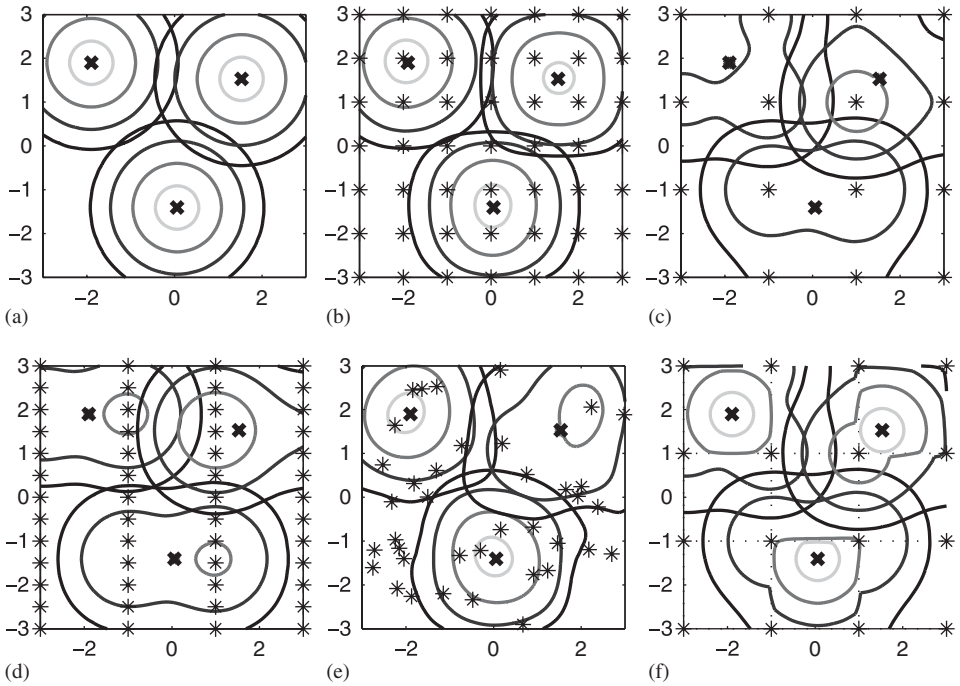

Figure 3. The correlation has been computed for three locations marked with x. Locations of inducing inputs are marked with * and in subplot 3(f) dotted lines denote the PIC block edges. The hyperparameters θ are the same in all the figures. See text for discussion: (a) The full GP; (b) the FIC approximation; (c) the FIC approximation; (d) the FIC approximation; (e) the FIC approximation; and (f) the PIC approximation.

θ. For PIC, Snelson and Ghahramani [44] constructed the block structure with simple clustering scheme. The effect of these extra parameters is discussed by Snelson [25] for high dimensional regression problems, and his results show that the approximations are efficient in many cases. In this paper, FIC and PIC are analyzed for spatial modeling. We are interested in approximating the full GP as close as possible to the approximations and consider how this can be done by keeping the computations feasible at the same time.

Figure 3 illustrates the approximations compared to the exact correlation structure, with hyperparameters θ fixed to the same values for all models. Figure 3(a) shows the exact correlation computed for 3 locations with the squared exponential covariance function. Figure 3(b) shows that if the inducing inputs are located appropriately, FIC produces correlation similar to the exact one. Figure 3(c) shows that if the distance between inducing inputs is too large compared to the characteristic length scale of the covariance function, FIC approximation fails and the effective correlation structure has larger length scale than the original and the highest correlation might be somewhere else than in the closest neighborhood. Furthermore, additional simulation studies showed that the induced effect to the marginal likelihood makes the hyperparameter values corresponding to the longer length scales more likely. Figure 3(d),(e) shows that not only the number of inducing inputs, but also their locations are important. In both cases, there are as many inducing inputs as in Figure 3(b). In Figure 3(d), the horizontal distance between the inducing inputs is too large compared to the actual length scale, increasing the effective length scale in that direction. In Figure 3(e), the locations of the inducing inputs have been chosen randomly and the goodness of the approximation varies in different areas. These examples illustrate also that if the locations of the inducing inputs were optimized, the interpretation of the produced effective correlation structure may be difficult and may lead to overfitting. Figure 3(f) shows that PIC can be used to improve the approximation locally. If the blocks in PIC are chosen in a sensible way, most of the high correlations will be approximated exactly and, thus, PIC also produces better marginal likelihood approximation than FIC.

Exact results would be obtained in FIC by having $m=n$ and inducing input in every data point location, or in PIC by having a single block with size $b=n$. In practice, we have to choose the maximum limit for m and b so that we can still compute the result. Reducing m increases the average distance from the data points to the inducing inputs limiting the smallest useful length scale. If the inducing inputs have been spread evenly, it is easy to estimate the approximate lower limit for the length scale and this can be taken into account in the overall data analysis. As a rule of thumb, with evenly spread inducing inputs the shortest length scale that FIC is able to infer is the distance between two adjacent inducing inputs. Increasing b improves the approximation locally allowing also shorter length scales although the approximation can still be poor at the edges of the blocks. As the correlation information passes through inducing inputs, it is useful to put the inducing inputs at the corners of the blocks and if the block edges are long, it is useful to add inducing inputs also along the edge. This was verified with a simulation study. In spatial cases, it is easy to put inducing inputs in an even grid and form even square blocks. Alternative even grids could be triangular or hexagonal, but based on a simulation study the differences to square grid are small.

The covariance function used has also some effect on the behavior of the approximations. For example, exponential covariance has thicker tail than squared exponential and thus inducing inputs can be slightly further apart before approximation fails on tails.

Copyright © 2010 John Wiley & Sons, Ltd.

Above we have considered the covariance structure of a model with one covariance function. The two covariance function case is more difficult as will be shown in the experiments. With FIC all the correlations are induced through the inducing inputs and, thus, modeling very short length-scale phenomena becomes unpractical since one would need infeasibly many inducing inputs to capture high variability. PIC on the other hand can capture short length scales within a block, but its shortcomings are the discontinuities in the correlation structure.

4.3. Computations with FIC and PIC

The computational savings with FIC/PIC are obtained by using the Woodbury-Sherman-Morrison lemma to invert the covariance matrix in (27) as

$$(\mathbf{Q}_{\mathrm{f,f}}+\Lambda)^{-1}=\Lambda^{-1}-\mathbf{V}\mathbf{V}^{\mathrm{T}}, \tag{29}$$

where $\mathbf{V}=\Lambda^{-1}\mathbf{K}_{\mathrm{f,u}}\mathrm{chol}[(\mathbf{K}_{\mathrm{u,u}}+\mathbf{K}_{\mathrm{u,f}}\Lambda^{-1}\mathbf{K}_{\mathrm{f,u}})^{-1}]$ (for proof, see [46]). A similar result for the determinant is given as

$$|\mathbf{Q}_{\mathrm{f,f}}+\Lambda|=|\Lambda||\mathbf{K}_{\mathrm{u,u}}^{-1}||\mathbf{K}_{\mathrm{u,u}}+\mathbf{K}_{\mathrm{u,f}}\Lambda^{-1}\mathbf{K}_{\mathrm{f,u}}|. \tag{30}$$

With FIC the computational time is dominated by the matrix multiplications, which need time $O(m^2n)$. With PIC the cost also depends on the sizes of the blocks in Λ. If the blocks were of equal size $b\times b$, the time for inversion of Λ would be $O(n/b\times b^3)=O(nb^2)$. With blocks at most the size of the number of inducing inputs, that is $b=m$, the computational cost in PIC and FIC are similar. However, in practice, PIC is faster since already with a small block size we can save in the number of inducing inputs.

The inference with Laplace approximation relies on evaluating the approximate log marginal posterior (16) and its derivatives, which can be evaluated efficiently using the above lemmas. The implementation is, in principle, straightforward manipulation of the Woodbury–Sherman–Morrison lemma (29), but in practice this requires care in implementation. For this reason, we have included the implementation in Appendix C.

In EP, the problematic part is the update of the posterior covariance. Recently, it was presented how the posterior update can be done in $O(m^2)$ for each site with FIC, resulting in total $O(nm^2)$ computation time [47] . The approach is called a generalized FITC. The basic idea is to write the posterior covariance as

$$((\mathbf{Q}_{\mathrm{f,f}}+\Lambda)^{-1}+\tilde{\Sigma}^{-1})^{-1}=\mathbf{D}+\mathbf{P}\mathbf{R}^{\mathrm{T}}\mathbf{R}\mathbf{P}^{\mathrm{T}}, \tag{31}$$

where $\mathbf{D}$ is diagonal, $\mathbf{P}$ is an $n\times m$ matrix, and $\mathbf{R}$ is an $m\times m$ upper triangular matrix. At the beginning $\mathbf{D}=\Lambda$, $\mathbf{P}=\mathbf{K}_{\mathrm{f,u}}$, and $\mathbf{R}=\mathrm{chol}_{\mathrm{upper}}[\mathbf{K}_{\mathrm{u,u}}^{-1}]$ (where $\mathrm{chol}_{\mathrm{upper}}$ is the upper triangular Cholesky decomposition). At each iteration the update of $\mathbf{D}$ and $\mathbf{P}$ are, respectively, $O(1)$ and $O(m)$ operations, and the update of $\mathbf{R}$ is $O(m^2)$ operation via rank 1 Cholesky update. The posterior marginals needed for the cavity distribution can be solved as $\sigma_i^2=\mathbf{D}_{ii}+\|\mathbf{R}\mathbf{P}_{i.}^{\mathrm{T}}\|$ and $b_i=v_i+\mathbf{P}_{i.}^{\mathrm{T}}\gamma$ in time $O(m)$. Here, the parameters $v=\mathbf{D}\tilde{\Sigma}^{-1}\tilde{\mu}$ and $\gamma=\mathbf{R}^{\mathrm{T}}\mathbf{R}\mathbf{P}^{\mathrm{T}}\tilde{\Sigma}^{-1}\tilde{\mu}$ are updated in $O(m)$ time at every iteration (for details see [47]).

The extension of the generalized FITC to generalized PIC is straightforward by following the equations in [47]. Only matrix $\mathbf{D}$ is changed to block diagonal, which affects on the updates of $\mathbf{P}$, $\mathbf{D}$, v, and γ that become at maximum $O(bm)$ or $O(b^2)$ operations. The update of $\mathbf{R}$ remains exactly the same as in FIC. Thus, if the block size in PIC is close to the number of inducing inputs the total cost in time is similar to FIC.

5. Combining the sparse approximation with compact support covariance

The FIC and PIC approximations work well for phenomena, whose length scale is long enough compared to the distances between the inducing inputs. In this section, we propose a new sparse GP model that combines the good global properties of FIC sparse approximation and the good local properties of CS covariance function.

5.1. Additive sparse GP model

As discussed in Section 4.2, the covariance matrix in FIC approximate prior (28) can be interpreted as a realization of a special kind of covariance function k_{FIC}. By adding up the FIC with CS covariance function, for example (7), we are able to construct a sparse GP model for the two-component problem (8) with prior

$$p(\mathbf{f}\,|\,\mathbf{X},\mathbf{X}_u,\theta)=\mathrm{N}(\mathbf{0},\mathbf{Q}_{\mathrm{f,f}}+\Lambda+\mathbf{K}_{\mathrm{f,f}}^{\mathrm{CS}})=\mathrm{N}(\mathbf{0},\mathbf{Q}_{\mathrm{f,f}}+\hat{\Lambda}). \tag{32}$$

We refer to this later as the CS+FIC model. Here, the matrix $\hat{\Lambda}=\Lambda+\mathbf{K}_{\mathrm{f,f}}^{\mathrm{CS}}$ is sparse with the same sparsity structure as in $\mathbf{K}_{\mathrm{f,f}}^{\mathrm{CS}}$. The sparse $\hat{\Lambda}$ is fast to use in computations and cheap to store since only the non-zero elements need to be saved and used in calculations.

The implementation of the CS+FIC model follows the guidelines given in Section 4.3. The inverse and determinant of the covariance matrix $\mathbf{Q}_{\mathrm{f,f}}+\hat{\Lambda}$ will be evaluated using equations (29) and (30) by plugging $\hat{\Lambda}$ in place of Λ. However, the matrix $\hat{\Lambda}$ is sparse without a (block)diagonal structure and, thus, its inverse will not be sparse. In the following section, we show, how to overcome this problem.

Copyright © 2010 John Wiley & Sons, Ltd.

5.2. Computation with sparse additive model

The key role in the computations with CS+FIC model is played by the sparse Cholesky factorization. In this section, we will consider Cholesky factorization $\mathbf{L}$ of symmetric positive definite matrix $\mathbf{A}$ which is a lower triangular matrix such that $\mathbf{A}=\mathbf{L}\mathbf{L}^{\mathrm{T}}$. For full matrix it can be found in time $O(n^3)$, but for sparse matrices this is faster since the sparsity is preserved in the Cholesky factorization. The factorization time depends on the sparsity structure of the matrix $\mathbf{A}$. It can be reduced by permuting the columns and rows of the matrix so that the number of the non-zero elements in $\mathbf{L}$ are minimized (see, for example [48, 49]). After finding the Cholesky decomposition of the matrix $\hat{\Lambda}$, we can efficiently evaluate the terms needed in the log posterior cost function and its derivatives. For example $\log|\hat{\Lambda}|=2\sum_{i=1}^{n}\log\mathbf{L}_{ii}$ and $\mathbf{f}^{\mathrm{T}}\hat{\Lambda}^{-1}\mathbf{f}$ is evaluated by first solving the linear equation $\mathbf{L}\mathbf{L}^{\mathrm{T}}\mathbf{v}=\mathbf{f}$ and then evaluating $\mathbf{f}^{\mathrm{T}}\mathbf{v}$. The terms, where we multiply an $n\times m$ matrix with $\hat{\Lambda}^{-1}$, for example $\hat{\Lambda}^{-1}\mathbf{K}_{\mathrm{f,u}}$, are analogous to solving m linear equations.

The terms that need most concern are of the form $\mathrm{tr}(\hat{\Lambda}^{-1}\partial\hat{\Lambda}/\partial\theta)$, which occur, for example, in the gradients of the approximate log marginal likelihoods (16) and (20) (see Appendix C). The matrix $\mathbf{B}=\partial\hat{\Lambda}/\partial\theta$ has non-zero elements at most for all $\mathbf{B}_{ij}$, for which $[\hat{\Lambda}]_{ij}\neq 0$. Thus, if we denote $\mathbf{Z}=\hat{\Lambda}^{-1}$ we can write $\mathrm{tr}(\mathbf{Z}\mathbf{B})=\mathrm{tr}(\mathbf{Z}_{\mathrm{sp}}\mathbf{B})$, where $\mathbf{Z}_{\mathrm{sp}}$ is a sparse representation of $\mathbf{Z}$, in which $[\mathbf{Z}_{\mathrm{sp}}]_{ij}\neq 0$ only if $\mathbf{B}_{ij}\neq 0$. We can obtain such a matrix by using an algorithm introduced by Takahashi *et al.* [50] (see also [51, 52]).

The algorithm is derived as follows. First, determine the Cholesky decomposition of $\hat{\Lambda}$. Then, we can write

$$\mathbf{L}^{\mathrm{T}}\mathbf{Z}=\mathbf{L}^{-1}. \tag{33}$$

Next, we take the diagonal of the Cholesky triangle, $\mathbf{A}=\mathrm{mask}(\mathbf{L},\mathbf{I})$, and write the equation (33) as

$$\mathbf{A}\mathbf{Z}+(\mathbf{L}^{\mathrm{T}}-\mathbf{A})\mathbf{Z}=\mathbf{L}^{-1}. \tag{34}$$

Subtracting the second term on the left-hand side, and multiplying by $\mathbf{A}^{-1}$, we obtain

$$\mathbf{Z}=\mathbf{A}^{-1}\mathbf{L}^{-1}-\mathbf{A}^{-1}(\mathbf{L}^{\mathrm{T}}-\mathbf{A})\mathbf{Z}. \tag{35}$$

Now, we can give a recursive formula for the elements of the inverse as

$$\mathbf{Z}_{ij}=\frac{\delta_{ij}}{\mathbf{A}_{ii}^{2}}-\frac{1}{\mathbf{A}_{ii}}\sum_{k=i+1}^{n}\mathbf{L}_{ki}\mathbf{Z}_{kj},\quad j\geqslant i, i=n,\ldots,1. \tag{36}$$

We could find the dense inverse by looping i from n to 1 and j from n to i and filling the lower triangular of $\mathbf{Z}$ according to symmetry. To find the sparse inverse we evaluate only a small fraction of the elements.

Let us denote by $\mathbf{C}$ an adjacency matrix, which has $\mathbf{C}_{ij}=1$ if $\mathbf{L}_{ij}\neq 0$ or $\mathbf{L}_{ij}^{\mathrm{T}}\neq 0$ and zero otherwise. Here we consider $\mathbf{L}_{ij}$ non-zero even if its numerical value was zero, but if it is symbolically nonzero. That is, it has to be evaluated when solving for sparse Cholesky factorization (see, for example [49]). To find the sparse inverse, we need to evaluate exactly the elements $\mathbf{Z}_{ij}$ such that $\mathbf{C}_{ij}\neq 0$. To justify this, consider the recursive formula (36). The algorithm needs at least all the elements $\{\mathbf{Z}_{ij}:\mathbf{C}_{ij}\neq 0\}$ since these are the elements that are needed for the diagonal of $\mathbf{Z}$. From the Cholesky factorization it follows that $\mathbf{C}_{ij}$ is nonzero if for any $k<i,j$ the elements $\mathbf{C}_{ik}$ and $\mathbf{C}_{kj}$ are nonzero [49]:

$$k<i,j\ \ \mathbf{C}_{ik}\neq 0\ \ \mathbf{C}_{kj}\neq 0\Rightarrow\mathbf{C}_{ij}\neq 0.$$

Thus, we can find the sparse inverse by looping i from n to 1 and at each i evaluating only the elements $\mathbf{Z}_{ij}, j\in\{j\geqslant i, C_{ij}\neq 0\}$ using equation (36).

Now, the Laplace approximation is straightforward to implement using the above sparse matrix routines. In EP, we need one more trick to keep the computations reasonable. The updates of the posterior covariance and mean are done similarly to FIC and PIC with the exception of updating matrix $\mathbf{D}$ in (31). Updating matrix $\mathbf{D}$ itself would force it to become a dense matrix as the EP algorithm proceeds through all the data points. For this reason we never evaluate the matrix $\mathbf{D}$ explicitly, but we evaluate all the elements we need from it implicitly. Using the matrix inversion lemma (29) we can write

$$\begin{aligned}\mathbf{D}=(\hat{\Lambda}^{-1}+\tilde{\Sigma}^{-1})^{-1}&=\hat{\Lambda}-\hat{\Lambda}(\hat{\Lambda}+\tilde{\Sigma})^{-1}\hat{\Lambda}\\&=\hat{\Lambda}-\hat{\Lambda}\tilde{\Sigma}^{-1/2}(\tilde{\Sigma}^{-1/2}\hat{\Lambda}\tilde{\Sigma}^{-1/2}+\mathbf{I})^{-1}\tilde{\Sigma}^{-1/2}\hat{\Lambda}\\&=\hat{\Lambda}-\hat{\Lambda}\tilde{\Sigma}^{-1/2}(\mathbf{L}_{\mathrm{LDL}}\mathbf{D}_{\mathrm{LDL}}\mathbf{L}_{\mathrm{LDL}}^{\mathrm{T}})^{-1}\tilde{\Sigma}^{-1/2}\hat{\Lambda}.\end{aligned} \tag{37}$$

There are two things to note in the above formulation. First, the term $\tilde{\Sigma}$ might have arbitrary large terms and, thus, we do not want to use it explicitly in calculations but rather work with $\tilde{\Sigma}^{-1/2}$ [7]. This leads to the modification from line 1 to line 2. In the last line we have written $\tilde{\Sigma}^{-1/2}\hat{\Lambda}\tilde{\Sigma}^{-1/2}+\mathbf{I}$ as its LDL Cholesky decomposition, which consists of a lower diagonal matrix $\mathbf{L}_{\mathrm{LDL}}$,

Copyright © 2010 John Wiley & Sons, Ltd.

which has ones in the diagonal, and a diagonal matrix $\mathbf{D}_{\mathrm{LDL}}$. The terms we need from $\mathbf{D}$ at the iteration i are at the i-th column of the matrix. The i-th column can be evaluated as

$$\mathbf{D}_{:,i} = \hat{\Lambda}_{:,i} - \hat{\Lambda}\tilde{\Sigma}^{-1/2}(\mathbf{L}_{\mathrm{LDL}}\mathbf{D}_{\mathrm{LDL}}\mathbf{L}_{\mathrm{LDL}}^{\mathrm{T}})^{-1}\tilde{\Sigma}^{-1/2}\hat{\Lambda}_{:,i},$$

where we have to solve a linear equation, multiply the result by a sparse matrix from the left, and subtract two sparse vectors.

At each EP iteration, there is change in only one column of $\hat{\Lambda}\tilde{\Sigma}^{-1/2}$ and in one row and one column of $\tilde{\Sigma}^{-1/2}\hat{\Lambda}\tilde{\Sigma}^{-1/2}+\mathbf{I}$. The former is cheap to update since we need to change only the non-zero elements in the particular column. The change in the latter term, on the other hand, also affects the LDL Cholesky decomposition, which would lead to heavy computations if it was evaluated from scratch each time. However, the change in a row and column i of $\tilde{\Sigma}^{-1/2}\hat{\Lambda}\tilde{\Sigma}^{-1/2}+\mathbf{I}$ affects only the row i and the columns $k \geqslant i$ of the matrices $\mathbf{L}_{\mathrm{LDL}}$ and $\mathbf{D}_{\mathrm{LDL}}$, and thus we can find the new LDL Cholesky decomposition by updating only these terms. This can be done efficiently with the algorithm introduced by Davis and Hager [53]. After the first round of EP iteration, all the diagonal elements of $\tilde{\Sigma}$ are nonzero and, thus, the non-zero pattern of the LDL Cholesky decomposition remains the same throughout EP iterations. For this reason we have to find the symbolic factorization only once at the beginning of EP algorithm. This leads to considerable speedup compared to the general case [53].

5.3. *Computational complexity and memory requirements of* CS+FIC

The computational complexity and memory requirements of CS+FIC depend on the number of non-zero elements in $\mathbf{L} = \mathrm{chol}[\hat{\Lambda}]$ ($\mathrm{nnz}(\mathbf{L}) \geqslant \mathrm{nnz}(\hat{\Lambda})$, where $\mathrm{nnz}(\mathbf{L})$ is the number of non-zero elements in $\mathbf{L}$). The fill-in of the Cholesky factorization can be reduced using permutation algorithms such as (approximate) minimum degree ordering (AMD) or nested dissection [49]. The time needed for permutation is, in general, negligible compared to the time needed for other evaluations. Thus, the essential part of the implementation is to use good permutation algorithms.

An estimate for the $\mathrm{nnz}(\mathbf{L})$ is given, for example, by Davis [49] or George *et al.* [54], who consider nested dissection algorithm in case of two dimensional $N \times N$ lattice with $n = N^2$ nodes. In this case, the nested dissection leads to asymptotically optimal ordering with $O(n \log n)$ nonzeros in $\mathbf{L}$ and $O(n^{3/2})$ time for computing the Cholesky factorization. After finding the Cholesky factorization of $\hat{\Lambda}$, evaluating its determinant is $O(n)$ operation and $\mathbf{f}^{\mathrm{T}}\hat{\Lambda}^{-1}\mathbf{f}$ is $O(n \log n)$ operation.

The computational complexity of evaluating the sparse inverse depends on the number of non-zero elements in the columns of $\mathbf{L}$. If γ_k denotes the number of non-zero elements outside the diagonal of column k of $\mathbf{L}$, then the computational cost for finding the sparse inverse is $O(\sum_{k=1}^{n} \gamma_k(\gamma_k+1))$. To get an intuition of the computation time we can consider banded and dense covariance matrices as limiting cases. Finding the sparse inverse of a banded covariance matrix requires time $O(\sum_{k=1}(b/2)((b/2)+1) = O(n(b/2)^2)$, where b denotes the band width, and finding the full inverse is $O(\sum_{k=1}^{n}(k-1)k) = O(n^3/3)$ operation. In the two-dimensional lattice, the average γ_k is $O(\log n)$, which gives an estimate for the recursion time $O(n \log(n)^2)$.

The modifications for the LDL Cholesky decomposition in EP are done using the row modification algorithm of the sparse Cholesky decomposition [53]. The update is optimal in the sense that it requires time proportional to the number of non-zero elements in $\mathbf{L}_{\mathrm{LDL}}$ that change. Since at iteration i only the row k and the columns $k \geqslant i$ of $\mathbf{L}_{\mathrm{LDL}}$ are updated, the average time for updating LDL decomposition is $O(1/n \sum_{l=1}^{n} \sum_{k=l}^{n} \gamma_k) \approx O(1/n \sum_{l=1}^{n}(n-l+1)\log n) = O(n \log n)$. Thus, the total time for EP with CS+FIC model scales as $O(n^2 \log n)$.

To conclude the CS+FIC model scales with Laplace approximation as $O(n \log(n)^2)$ and with EP as $O(n^2 \log n)$. The memory requirement in both cases is $O(\max(nm, n \log n))$.

6. Simulation studies

So far, we have described how the posterior inference with the GP model can be sped up by using approximations for integration and sparse GPs. In the present section, we give experimental results that confirm our theoretical discussion. First, in Section 6.1, we compare the Laplace method and EP to an MCMC solution. In Section 6.2, we compare the predictive performance of the FIC and PIC sparse approximations with data sets that have different length scales in their spatial effects. In Section 6.3, we compare the performances of sparse GP models for data sets with two spatial phenomena, one with short and the other with long length scale. In Section 6.4 we consider the computational speed.

All the experiments in this section use simulated data. The expected number of deaths is evaluated from real mortality data from Finland aggregated into lattice with 10 km × 10 km grid cells, which results in 3206 data points. The log relative risk for the cells, f_i, is drawn from a (dense) GP with squared exponential covariance function, whose parameters are varied in the experiments. The numbers of deaths, y_i, are drawn from Poisson distribution with mean $e_i\mu_i = e_i \exp(f_i)$.

We will use two predictive performance tests to compare the different inference schemes and sparse approximations. The first test is based on the expected log predictive density (ELPD). The ELPD at cell i is given as

$$\mathrm{ELPD}_i = \mathrm{E}_{y_i}\left[\log\left(\int p(y_i|f_i)p(f_i|\mathcal{D})\,\mathrm{d}f_i\right)\right],$$

where $p(f_i|\mathcal{D})$ is the marginal posterior density of latent value f_i, and $\mathrm{E}_{y_i}[\cdot]$ is the expectation over $p(y_i|\bar{f}_i)$, where $\bar{f}_i$ is the simulated latent value at data point i. In case of MCMC solution, the integration over the latent values is done by averaging over the latent

Copyright © 2010 John Wiley & Sons, Ltd.

value samples. In case of EP, and Laplace approximation, the integration was conducted by Gaussian quadrature. The ELPD of the true model at cell i is

$$\mathrm{ELPD}_i^{\mathrm{true}} = \mathrm{E}_{y_i}[\log p(y_i|\bar{f}_i)].$$

The ELPD values are compared to the true values by evaluating the mean of the differences between the true and inferred ELPD values

$$\Delta\mathrm{ELPD} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{ELPD}_i^{\mathrm{true}} - \mathrm{ELPD}_i). \tag{38}$$

The other test statistic evaluated was the root mean squared error between the predictive mean of the latent value and the real simulated latent value

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\mathrm{E}[f_i|\mathscr{D}] - \bar{f}_i)^2}. \tag{39}$$

6.1. Comparing the inference methods

In this section, we assess the goodness of the methods discussed in Section 3. We will consider long MCMC simulations as the golden standard and compare the EP and Laplace approximation to them. For this we compare the following solutions:

- MCMC solution for the conditional posterior of latent variables $p(\mathbf{f}|\hat{\theta},\mathscr{D})$.
- EP and Laplace approximation for the conditional posterior of latent variables $q(\mathbf{f}|\hat{\theta},\mathscr{D})$.
- Full MCMC solution, in which we sample from $p(\theta,\mathbf{f}|\mathscr{D})$, and from which we can extract the marginals $p(f_i|\mathscr{D})$, $p(l|\mathscr{D})$, and $p(\sigma_{\mathrm{se}}^2|\mathscr{D})$.
- The integrated EP and Laplace approximation for the marginal posterior of the latent variables, $q(\mathbf{f}|\mathscr{D})$, where the marginalization over the hyperparameters is conducted via grid search, importance sampling, and CCD method.
- The marginal posterior of the hyperparameters evaluated using EP and Laplace approximations for the marginal likelihood. This is obtained by evaluating $q(\theta|\mathscr{D})$ (equations (16) and (20)) in a grid of hyperparameter values, after which the marginal distribution is evaluated by summing over all but one dimension of θ and normalizing the answer.
- The log marginal likelihood $\log p(\mathscr{D}|\theta)$ approximated by EP, Laplace approximation, and MCMC. This is evaluated for a 7×9 grid of hyperparameter values around the mode.

The MCMC sampling from the posterior distribution is described in [55]. We sampled 2000 approximately independent samples from both $p(\theta,f|\mathscr{D})$ and $p(f|\hat{\theta},\mathscr{D})$ using 10 distinct chains (this was obtained by saving only every 300th sample from the original chain). The convergence of the sample chains was analyzed with a potential scale reduction factor (PSRF) [56], and a Kolmogorov–Smirnov (KS) test [57] between different chains. The sample autocorrelation was estimated using Geyer's initial monotone sequence estimator [58]. We discarded all the latent variables whose autocorrelation was over 2 to ensure the reliability of the results. In the sample chain from the conditional posterior there were 36 and in the sample chain from the joint posterior 91 such latent variables out of the total 3206. The sample autocorrelation of hyperparameters was about 5. The MCMC solution for marginal likelihood $p(\mathscr{D}|\theta)$ was evaluated using annealed importance sampling [24, 59]. The temperature changes were taken with step size 0.005 and the latent variables were whitened before sampling as described in [55]. We sampled 100 realizations for each hyperparameter configuration and took the average of these to be the final estimate for $p(\mathscr{D}|\theta)$.

The data studied are simulated as described above using the squared exponential covariance function with magnitude $\sigma_{\mathrm{se}}^2 = 0.1$ (which corresponds in approximately 0.12 variance for the relative risk) and length scale 200 km (the maximum width and height of Finland are approximately 600 and 1200 km, respectively). The data set will later be called *simul200*. We analyze the data with FIC with squared exponential covariance function. The hyperparameters are given priors $l_{\mathrm{se}} \sim \text{half-}t(4, 50)$ and $\sigma_{\mathrm{se}}^2 \sim \text{half-}t(4, 0.3)$. The length scale is given rather narrow prior to verifying that the half t-distribution really enables the posterior mass to concentrate in larger values if prior and data are in conflict. As will be seen below the prior does not overly restrict the posterior with this data. We placed the inducing inputs in 100 km $\times$ 100 km lattice and call the model *FIC100*. In this construction, the length scale of the spatial phenomenon is so long that FIC represents very closely full GP (see the discussion in Section 4.2 and the results in [55]).

As there is no easy way to compare the full posterior, we examine the marginal distributions. Figure 4 shows scatter plots of the mean and the variance of the posterior of the latent variables for MCMC vs the Laplace approximation and Figure 5 shows the comparison of the MCMC with the Laplace solution for the marginals of the hyperparameters. The EP solution looked exactly the same as the Laplace approximation. From the Figure 4(a) we can see that the Gaussian distribution, obtained either with the Laplace method or EP, is a very accurate approximation for the conditional posterior $p(\mathbf{f}|\hat{\theta},\mathscr{D})$. The Laplace and EP approximation at the posterior mode of hyperparameters, $q(\mathbf{f}|\hat{\theta},\mathscr{D})$, also estimate very accurately the mean of the marginal posterior $p(\mathbf{f}|\mathscr{D})$ but underestimates slightly its variance as described in the Figure 4(b). The integrated Laplace approximation and the full MCMC are compared in the Figure 4(d), from which it is seen that the grid-based integration gives very accurate results as well. The importance sampling and CCD integration methods gave similar results to the grid-based integration. The accuracy of the grid-based integration and the marginal likelihood estimate of Laplace approximation and EP is further confirmed by Figure 5, which shows the marginal posterior of hyperparameters. The MCMC solution and Laplace approximation give identical results here also.

Copyright © 2010 John Wiley & Sons, Ltd.

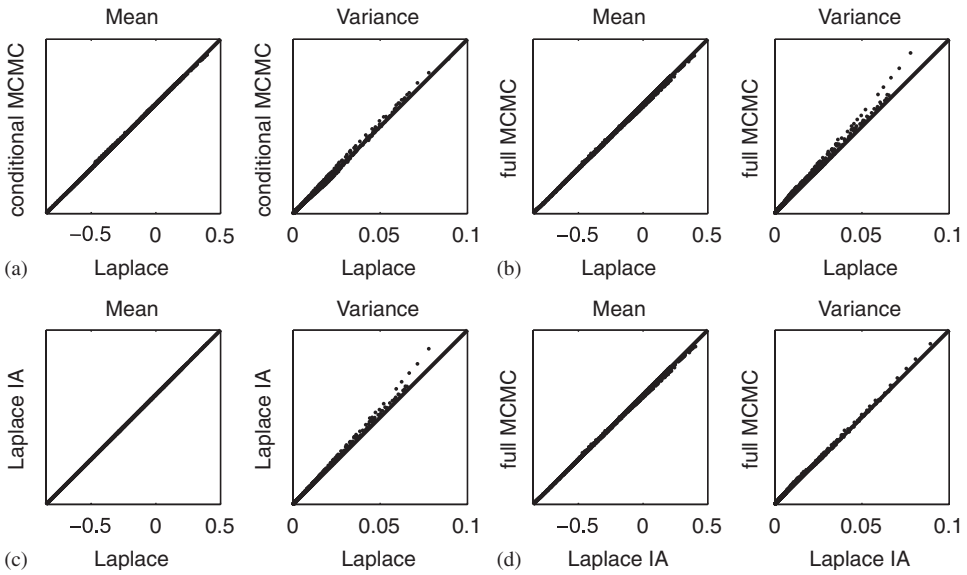

Figure 4. Scatter plots of the mean and variance of the posterior of latent variables for MCMC vs Laplace approximation: (a) Laplace approximation $q(\mathbf{f}|\hat{\theta},\mathscr{D})$ vs MCMC for $p(\mathbf{f}|\hat{\theta},\mathscr{D})$; (b) Laplace approximation $q(\mathbf{f}|\hat{\theta},\mathscr{D})$ vs MCMC for $p(\mathbf{f}|\mathscr{D})$; (c) Laplace approximation $q(\mathbf{f}|\hat{\theta},\mathscr{D})$ vs integrated Laplace approximation $q(\mathbf{f}|\mathscr{D})$; and (d) integrated Laplace approximation $q(\mathbf{f}|\mathscr{D})$ vs MCMC for $p(\mathbf{f}|\mathscr{D})$.

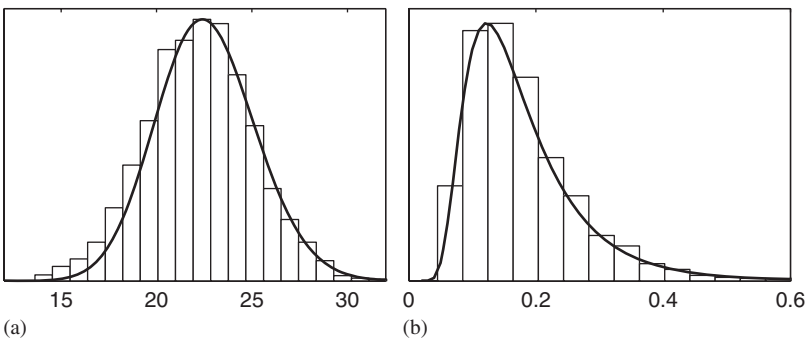

Figure 5. The marginal posterior of the hyperparameters for simul200 data set. The histogram shows the full MCMC solution and the full line integrated Laplace approximation. The joint posterior of the log hyperparameters, $\gamma=\log(\theta)$, is shown in Figure 2: (a) Marginal posterior of length scale $p(l|\mathscr{D})$ and (b) marginal posterior of magnitude $p(\sigma^2_{\text{se}}|\mathscr{D})$.

The log marginal likelihood $\log p(\mathscr{D}|\theta)$ is equal to all the methods, EP, Laplace approximation, and annealed importance sampling, down to the first decimal place in all the studied grid cells. The log marginal likelihood for the mode was -4477.6. Thus, both EP and Laplace approximation give very accurate approximations for the marginal likelihood and thus their approximation for the posterior surface is nearly exact.

We studied the latent value posteriors more carefully with the KS test [57], which compares the empirical cumulative density functions of two sample sets, s_1 and s_2, as

$$\sqrt{n}\sup_{s_1}|F_n(s_1)-F_n(s_2)|,$$

where n is the number of samples in the chain, and $F_n(s)$ is an empirical distribution function. The statistics from the KS test can be compared to the statistics of a repeated test for i.i.d. sample chains to get, for example, 95 per cent confidence limit [57]. By comparing the latent value samples from MCMC to the samples drawn from the posterior distribution of EP (18) and the Laplace approximation (14) with the KS statistics, we get an estimate of how many of the marginal posteriors are similar with certain confidence level. We conducted the KS test with 95 per cent confidence level, which was found by 10 000 tests between two identically normally distributed samples. The KS test makes no assumptions of the distribution, but it assumes that the samples are independent. Thus the MCMC sample chain has to be well converged and thinned.

The KS statistics was under the 95 per cent confidence level in over 93 per cent of the cells with both EP and the Laplace approximation. The statistics are visualized in Figure 6. In Figure 7, we show the marginals for three cells with different amount of population and death cases (see the figure for details).

Copyright © 2010 John Wiley & Sons, Ltd.

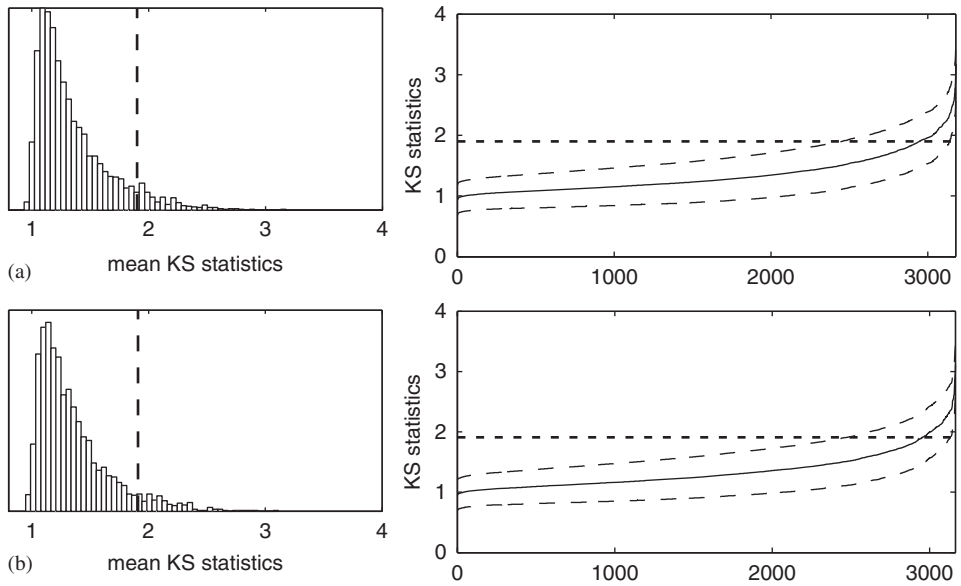

Figure 6. The results from KS test between MCMC, EP, and Laplace approximation. The histograms show the distribution of the mean KS statics for MCMC samples compared 200 times with posterior samples drawn from EP or Laplace approximation. The figures on the right show the mean and standard deviation of these tests plotted in ascending order. The straight dashed line in all the figures is the limiting value for the 95 per cent confidence level: (a) EP compared with MCMC and (b) Laplace approximation compared with MCMC.

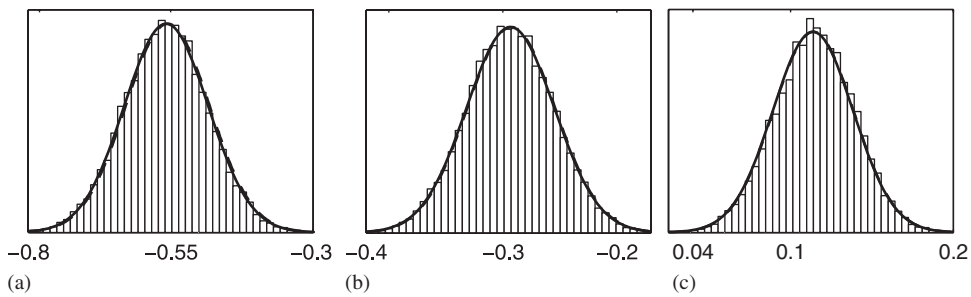

Figure 7. The posterior densities of the latent values in three cells in data simul200 and model FIC100. The histogram shows the MCMC samples and the full line of the EP approximation. The Laplace approximation is invisible as it is under the EP's line: (a) $y=0$, $p=4$; (b) $y=5$, $p=314$; and (c) $y=40$, $p=10964$.

We also conducted the predictive performance tests but did not find any significant difference between any of the above mentioned inference schemes. The mean ΔELPD statistics was 0.005 ± 0.025 and the RMSE was around 0.091 for all the models.

6.2. Comparing the FIC and PIC approximations

The performance of the FIC and PIC approximation are compared with four simulated data sets. These are *simul200* (Section 6.1), and data sets whose Gaussian fields have magnitude $\sigma^2_{\text{se}}=0.1$ and length scales 20, 50, and 100 km. These will be called, respectively, *simul20*, *simul50*, and *simul100*. We constructed 50 realizations of each data and averaged the results over them. FIC has either 43 inducing inputs in $100\,\text{km}\times100\,\text{km}$, or 152 inducing inputs in $50\,\text{km}\times50\,\text{km}$ lattice. The corresponding model names are *FIC100* and *FIC50*. The blocks in PIC are of size $100\,\text{km}\times100\,\text{km}$, and the inducing inputs are placed either in the corners of the blocks ($m=43$), or in the corners and in the middle of the edges ($m=100$). The models are called *PIC100A* and *PIC100B*, respectively. The hyperparameters are given priors $l_{\text{se}}\sim\text{half-}t(4,50)$ and $\sigma^2_{\text{se}}\sim\text{half-}t(4,0.3)$, with the same justifications as in the previous section.

The inference is conducted via Laplace method and EP and the performance is evaluated by ΔELPD and RMSE statistics. The results are shown for EP in Figures 10 and 11. The Laplace approximations performed similarly to EP. In RMSE statistics there was no difference. In Figure 8, we show the latent value posterior surfaces when the FIC and PIC models work well and poorly.

The posterior modes of the hyperparameters in simul200 were approximately $l=225\,\text{km}$ and $\sigma^2=0.15$ with all the models. This is reasonable since the sparse approximations have to induce the correlations through the inducing inputs, which means that the effective distances between data points are longer than in dense GP.

With simul100 the posterior mode of hyperparameters of FIC100 model was approximately $l=140\,\text{km}$ and $\sigma^2=0.2$, which is too large compared to the length scale of the simulating process and leads to oversmoothing shown in Figure 8(c). The effective distance of FIC50 is still short enough for simul100 but it also oversmooths simul50. With simul20 the posterior mode of hyperparameters of FIC models were approximately $l=18\,\text{km}$ and $\sigma^2=0.09$. With this short length scale, FIC is able to find

Copyright © 2010 John Wiley & Sons, Ltd.

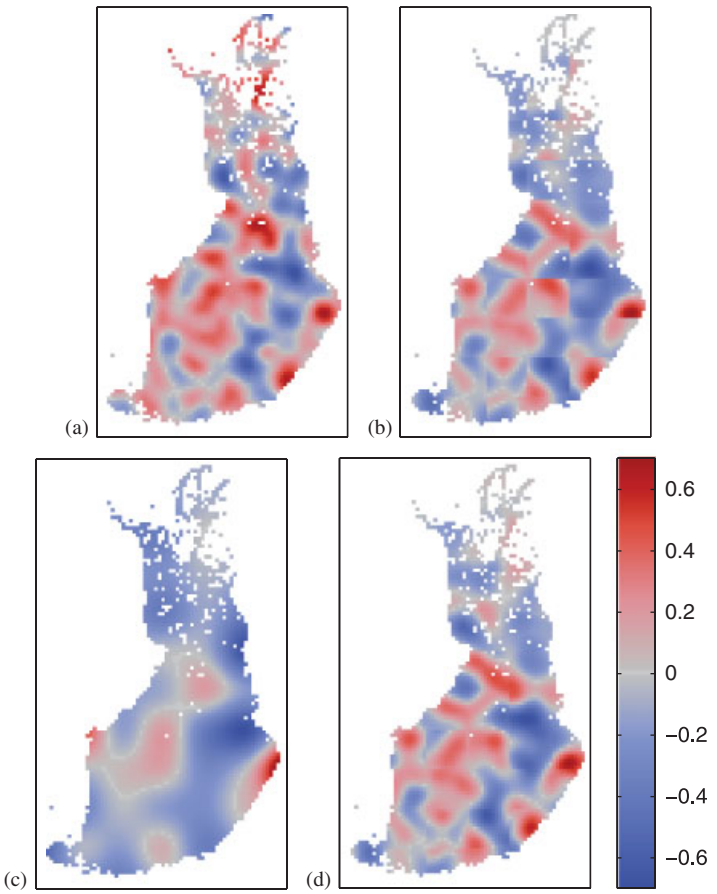

Figure 8. Examples of latent value posterior surfaces for simul100 data set. The real latent surface is shown in Figure 8(a). In Figure 8(b) is a prediction with PIC, where the block structure is clearly visible as the model has too few inducing inputs. Figure 8(c) is a prediction with FIC that has too few inducing inputs, which leads to oversmoothing. Figure 8(d) shows prediction with FIC and PIC models that are constructed correctly. The white areas are either outside Finland's borders or are uninhabited: (a) Simulated latent surface; (b) prediction of PIC100A; (c) prediction of FIC100; and (d) prediction of PIC100B (FIC50 was similar).

correlations only between data points that are next to an inducing input and the models broke down completely resulting in a spotty map.

For PIC, it is favorable to model the correlation structure exactly inside the block, and thus, the posterior modes of the hyperparameters in PIC models were very close to the hyperparameter values of the simulating process in simul100, simul50, and simul20 data sets. With PIC, the problem is that the correlation between blocks gets weaker as the number of inducing inputs is reduced, which results in sharp edges in the predicted latent value surface as illustrated in Figure 8(b). Placing inducing inputs in the middle of the block edges improved PIC as shown in Figure 8(d).

From Figure 9, it can be seen that PIC models work equally well deep inside the blocks. Thus, the average model performance over the cells does not necessarily change much even if we add more inducing inputs on the block boundaries. This explains why the difference between PIC100A and PIC100B is rather small in Figures 10(b) and 11(b). However, adding inducing inputs also on the block boundaries reduces discontinuities in the posterior surface significantly.

6.3. Simulated data with two spatial phenomena

In this section, we apply additive sparse GP models to simulated data set with two spatial phenomena. The data were created similarly to the data sets discussed in Section 6.2. The underlying latent function was constructed with zero mean GP with covariance function $k_{\text{se}}(\cdot,\cdot)+k_{\text{pp}}(\cdot,\cdot)$. The hyperparameters for the covariance functions were $\sigma^2_{\text{se}}=0.06$ and $l_{\text{se}}=200\,\text{km}$ for squared exponential, and $\sigma^2_{\text{pp}}=0.04$ and $l_{\text{pp}}=100\,\text{km}$ for piecewise polynomial. The models constructed were FIC and PIC with $k_{\text{se}}(\cdot,\cdot)+k_{\text{pp}}(\cdot,\cdot)$ covariance function, and CS+FIC where the FIC part used squared exponential and the CS part the piecewise polynomial function. The hyperparameters were given priors $l_{\text{se}}\sim\text{half-}t(4,20)$, $\sigma^2_{\text{se}}\sim\text{half-}t(4,0.3)$, $l_{\text{pp}}\sim\text{half-}t(4,5)$ and $\sigma^2_{\text{pp}}\sim\text{half-}t(4,0.3)$, which encompass the prior belief that the piecewise polynomial should adapt to the short and the squared exponential to the long length-scale phenomena. The inducing inputs and block structure for FIC and PIC models were the same as in Section 6.2. The inducing inputs in CS+FIC model were set to a regular $100\,\text{km}\times 100\,\text{km}$ grid.

The results for EP are presented in Figure 12. CS+FIC was the best model in both the performance tests. We have left out the FIC100 because it was clearly the worst model, and would have spread the limits in the figures too much. As the deviations in

Copyright © 2010 John Wiley & Sons, Ltd.

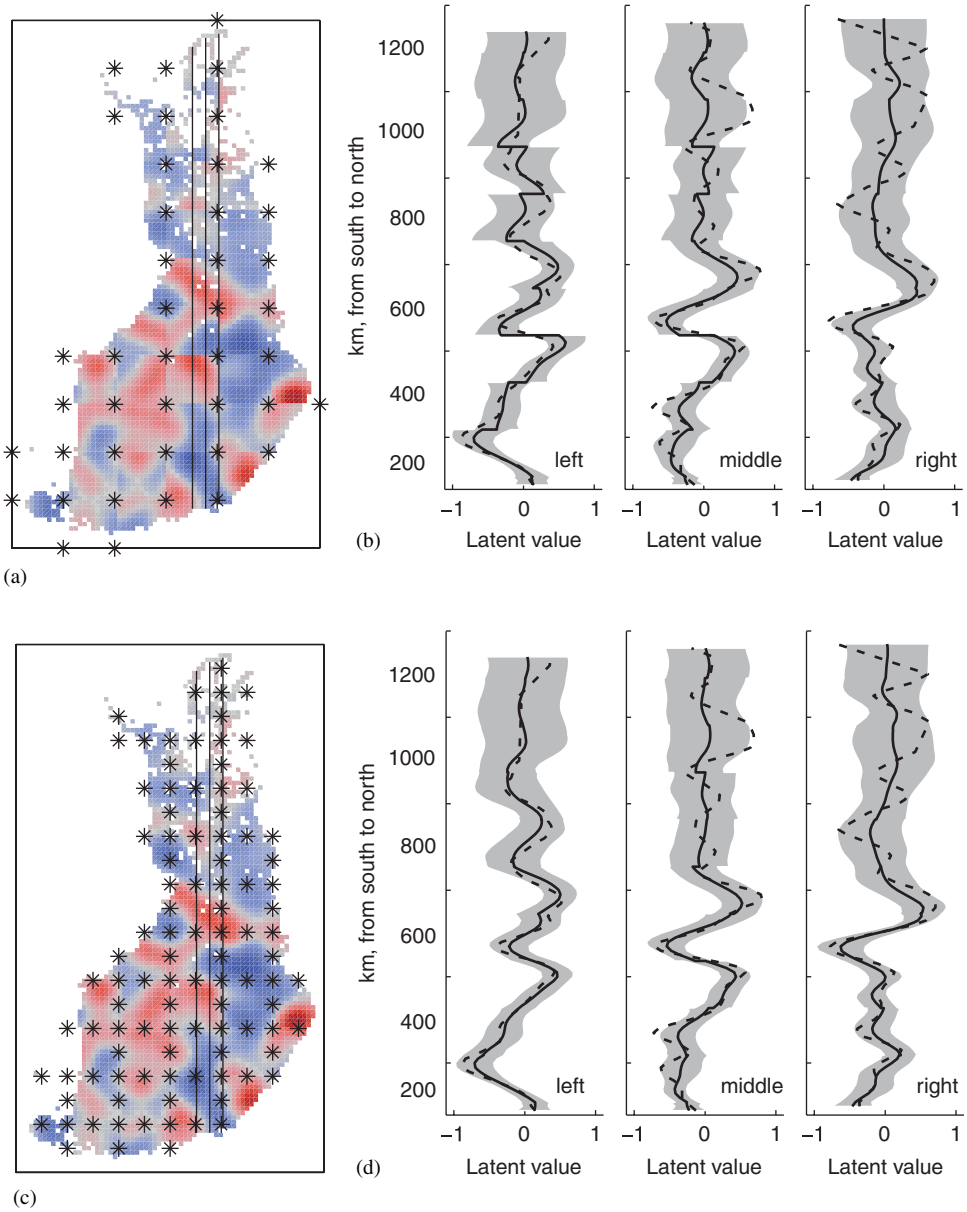

Figure 9. Examples of latent value posterior surfaces for simul100 data set and PIC100A and PIC100B models. The maps show the inducing inputs and the vertical cross sections. The plot next to the map shows the predictive mean (full line), 95 per cent posterior interval (shaded area), and the true latent value (dashed line) on the cross sections. In the leftmost cross section of the PIC100A model, the discontinuities are most clearly visible. Comparing that to the leftmost cross section in PIC100B shows how the inducing inputs make the correlation continuous. From the cross sections, it can also be seen that the uncertainty increases toward north as less people live there than in the south: (a) Prediction with PIC100A; (b) cross sections of PIC100A; (c) prediction with PIC100B; and (d) cross sections of PIC100B.

the performance of PIC and CS+FIC model overlap slightly, we conducted pairwise comparison to see the relative order of the models. CS+FIC was the best in each of the 50 simulated data sets. The difference in ΔELPD between CS+FIC and the second best model PIC100B was in average 2×10^{-3}, which is also the difference in the medians in Figure 12(a).

FIC was able to model well the long length-scale phenomenon, but it oversmoothed the relative risk too much. The PIC100B works better than PIC100A because it also has inducing inputs in the edges of the blocks. The main advantage of CS+FIC is that it has a continuous correlation structure. PIC and CS+FIC model equally well the long length-scale phenomenon and the short length-scale phenomenon inside the blocks. However, PIC is worse than CS+FIC near the block edges.

6.4. Computational speed

In this section, we will consider the speed of the different inference schemes and sparse GPs. All the results are recorded with Intel(R) Pentium(R) D CPU 3.00 GHz computer with 2 GB memory and the code is implemented with Matlab 7.4.

Copyright © 2010 John Wiley & Sons, Ltd.

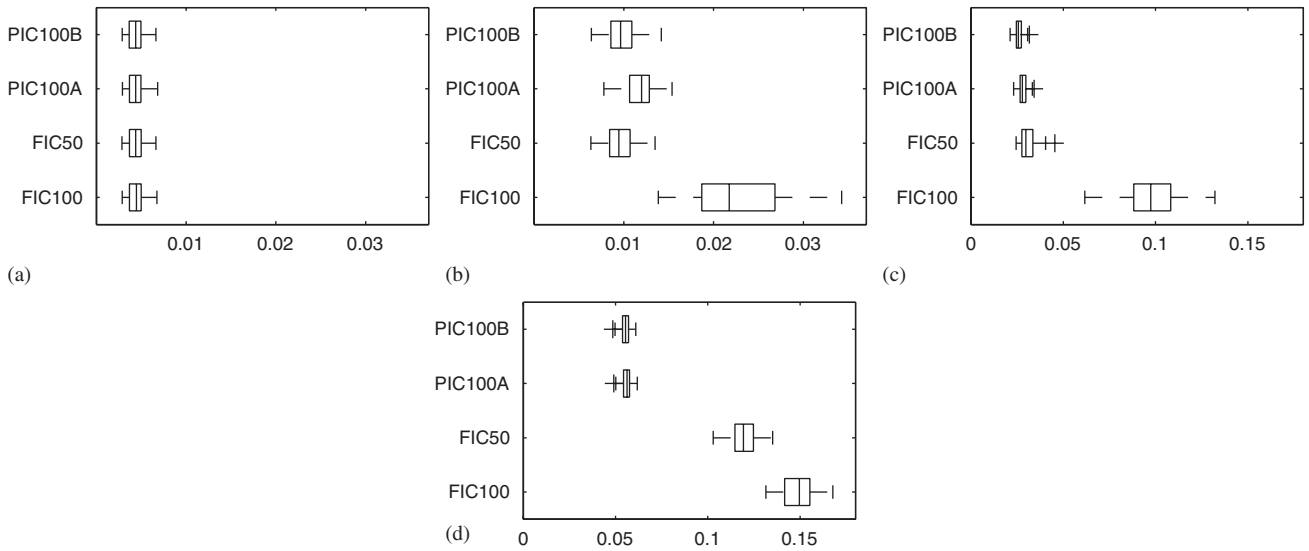

Figure 10. Box plots of ΔELPD of different models and data sets. The results are averaged over 50 simulated data sets. Box ends are the lower and upper quartiles, the vertical line inside the box is the median and the whiskers show the extent of the data. Note the difference in scale in (a)–(b) and (c)–(d): (a) The simul200 data; (b) the simul100 data; (c) the simul50 data; and (d) the simul20 data.

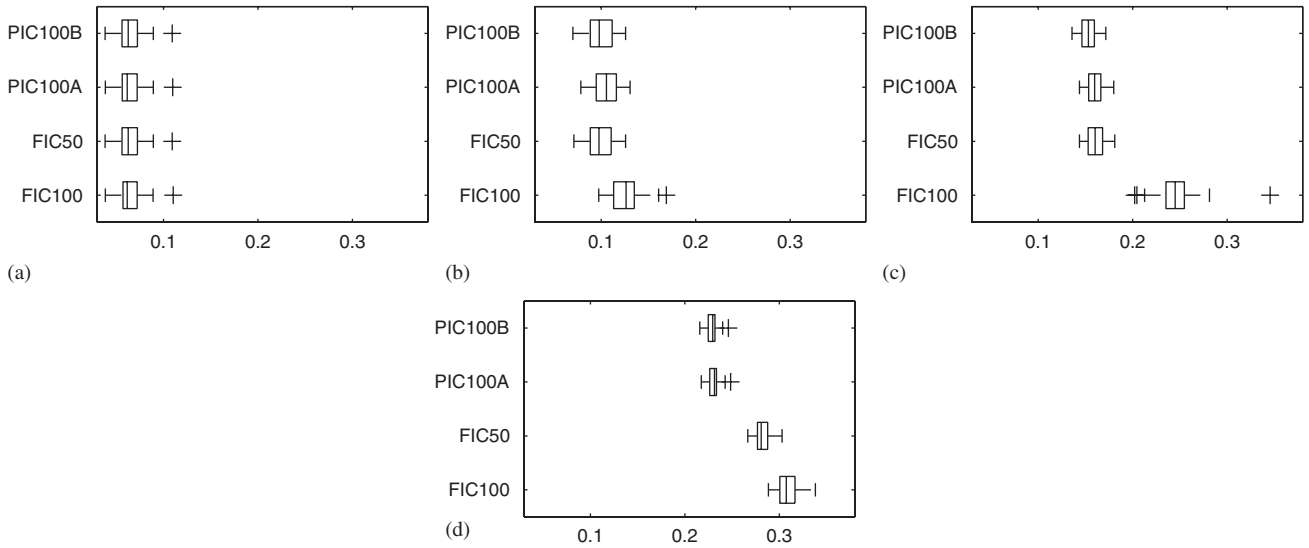

Figure 11. Box plots of RMSE of different models and data sets. The results are averaged over 50 simulated data sets. Box ends are the lower and upper quartiles, the vertical line inside the box is the median and the whiskers show the extent of the data: (a) The simul200 data; (b) the simul100 data; (c) the simul50 data; and (d) the simul20 data.

In Section 6.1, the time needed to find the EP and Laplace approximation for $p(\mathbf{f}|\hat{\theta},\mathscr{D})$ and one approximately independent MCMC sample from the same distribution (that is optimization of θ is excluded from the times) were, respectively, 14, 3, and 17 cpu seconds. In the experiments in Sections 6.2 and 6.3, the hyperparameters were initialized so that finding the mode $\hat{\theta}$ with EP and Laplace approximation took about 10–25 iterations with all the models. The time needed for one optimization step ranged from 5 to 20 s with Laplace approximation and from 15 to 900 s with EP depending on the model and data. The optimization took less than 5 min for all other models but CS+FIC, if it were optimized with EP. We also run full MCMC (that is, we sampled from $p(\mathbf{f},\theta|\mathscr{D})$) for some of the simulated data sets above from the same initial hyperparameter values as with Laplace approximation and EP. The MCMC sampler needed about 500–800 steps for convergence and it took approximately 100 cpu seconds to draw one approximately independent sample from $p(\mathbf{f},\theta|\mathscr{D})$. The problem was the slow convergence and mixing of the hyperparameters.

There are a number of parameters that affect the computational speed of the sparse GPs (the number of inducing inputs, block size, the sparsity of $\hat{\Lambda}$, etc.). In general we can summarize that FIC and PIC are equally fast, if they have the same number of inducing inputs and the block size of PIC is about the same as the number of inducing inputs ($m \approx b$). As we need to add more inducing inputs into FIC and PIC to model short length scales it is interesting to compare the speed of CS+FIC with the speed of FIC and PIC with the same predictive performance. For this, we constructed such PIC and FIC models that they worked

Copyright © 2010 John Wiley & Sons, Ltd.

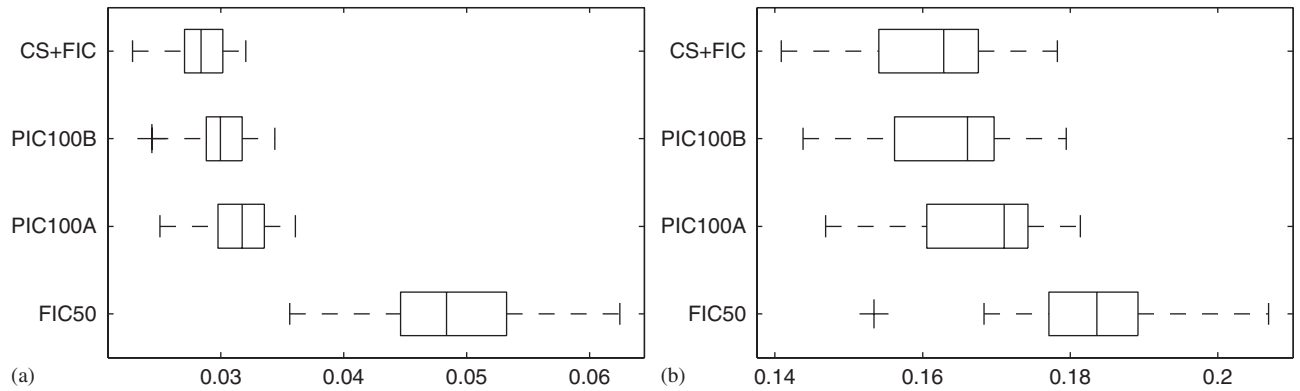

Figure 12. Box plots of ΔLPD and RMSE of different models in case of relative risk with two spatial phenomena. The results are averaged over 50 simulated data sets. Box ends are the lower and upper quartiles, the vertical line inside the box is the median and the whiskers show the extent of the data. The model FIC100 was left out of the figures because it was clearly the worst model (median ΔELPD 0.11 and median RMSE 0.23): (a) The ΔELPD averaged over 50 simulated data sets and (b) the RMSE averaged over 50 simulated data sets.

(almost) equally well as CS+FIC in the data with two spatial phenomena (Section 6.3). In this setting, PIC had blocksize of 50 km and inducing inputs in the block corners and two inducing inputs at each block edge resulting in a total of 616 inducing inputs. FIC had 871 inducing inputs on a 20 km $\times$ 20 km lattice. The times for inference with Laplace approximation were 440 (FIC), 220 (PIC) and 130 (CS+FIC) seconds and with EP 8410 (FIC), 1490 (PIC), and 4130 (CS+FIC).

7. Experiment with real data: alcohol-related diseases in Finland

In the previous section we examined the properties of the approximate inference schemes and the sparse GPs with simulated data. Here, we apply the methods for the real disease mapping problem. We studied the deaths on alcohol-related diseases in Finland during 2001–2005. The data are aggregated into a lattice with 5 km $\times$ 5 km cells. This leads to spatially sparse data since large areas of Finland are uninhabited, and only 10 608 out of 13 946 cells (that is 76 per cent) contain any observations.

We analyzed the data with several GP models and chose the best among them. The first step was to use only one covariance function and FIC. This model was followed by additive models with two and three covariance functions and PIC sparse approximation. The best model had an additive covariance function $k_{\text{se}}(\cdot,\cdot)+k_{\text{ma}}(\cdot,\cdot)+k_{\text{pp}}(\cdot,\cdot)$. The squared exponential covariance function is used for long length-scale correlations for which reason its length scale is given a wide prior $l_{\text{se}} \sim$ half-$t(4, 50)$. The Mátern covariance function is used for moderate length correlations and its length scale is given little narrower prior $l_{\text{ma}} \sim$ half-$t(4, 20)$. The piecewise polynomial covariance function is used for local correlations and presumably its length scale should be very short so we gave it $l_{\text{ppcs2}} \sim$ half-$t(4, 5)$ prior. The magnitude of each of the covariance functions was given a $\sigma^2 \sim$ half-$t(4, 0.3)$ prior.

As the covariance function is additive the final analysis was conducted using CS+FIC, which outperformed the PIC model slightly. The inducing inputs were set in a 25 km $\times$ 25 km lattice resulting in 560 of them. The inference is conducted via EP and Laplace approximation with a MAP estimate for the hyperparameters and by marginalizing over them with the CCD integration described in Section 3.3.3.

As the most common spatial prior in disease mapping is the conditional autoregressive (CAR) model, it is interesting to compare our GP prior with it. A CAR model closest to our approach is the INLA [13], where the posterior of the relative risk is evaluated using Laplace approximation for the conditional posterior of the latent variables and the integration over hyperparameters is conducted using the same CCD approximation as here. When with GP the correlation between two cells is defined by the covariance function and the distance between them, with CAR the correlation is induced by the neighborhood structure. The crucial choices then are the size of the neighborhood, which affects the smoothness, and whether the empty cells are included into the model or not. In the latter case the effective distance between two cells will grow if there are empty areas between them, and thus would be very different in northern Finland from southern Finland (see Figure 13). However, if the empty cells are included into the model, the correlation structure will be the same throughout the country but the size of the data set will increase. Along with the present data set including empty cells the size of the data will grow to 31 per cent.

We tested CAR model with two neighborhoods, one with 8 nearest neighbors, and the other with 20 nearest neighbors. The best model out of these was the smaller neighborhood so that all the empty cells were included into the model. The same model performed the worst if the empty cells were excluded, and the model with the larger neighborhood placed between these two models. The results are reported for the best and worst CAR model to illustrate the importance of keeping also the empty cells.

In Figure 13 we have plotted the posterior median of the relative risk of the GP model and the two CAR models. The GP model and the CAR model with empty cells seem to give similar results for the overall shape of the relative risk surface, but the CAR model without empty cells predicts the relative risk much differently from the other two. Figure 14 shows the posterior probability that the risk has increased, and again there is only a small difference between GP and CAR with empty cells. To compare the models in more detail, we plotted the mean and variance of the latent variables into scatter plots shown in Figure 15(a). Now, we can observe a clear difference between the CAR and GP models. CAR model gives certain cells much higher variance

Copyright © 2010 John Wiley & Sons, Ltd.

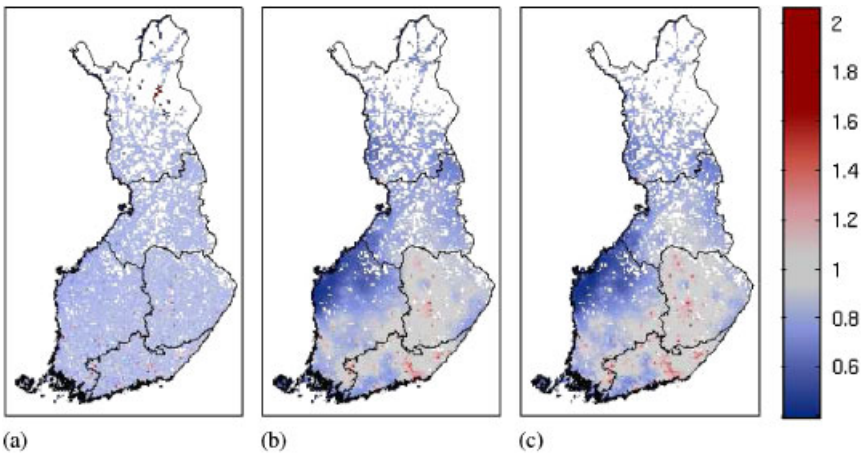

Figure 13. The posterior median of the relative risk of alcohol-related diseases in Finland in 2001–2005 with CAR and GP models. Notice also the large areas with no population in the northern parts of Finland and how including the empty cells improve the resolution of the CAR model. (Borderlines © Affecto Finland Oy, Karttakeskus, License L8573/10): (a) CAR model without empty cells; (b) CAR model with empty cells; and (c) GP model.

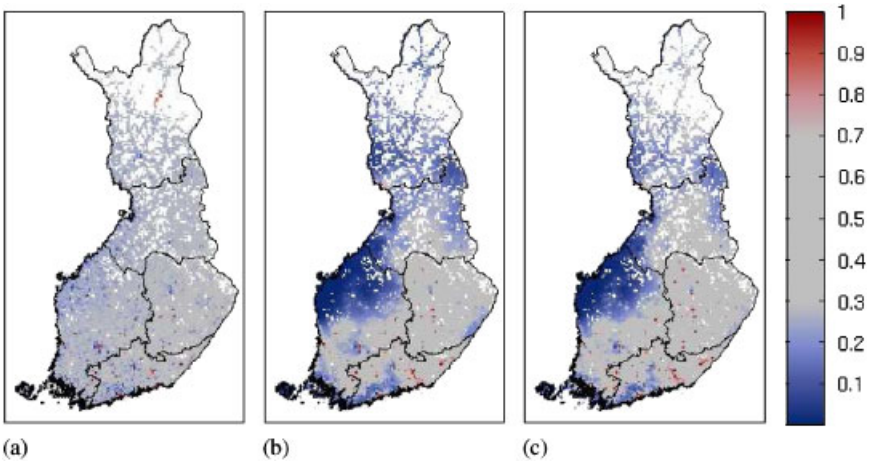

Figure 14. The posterior probability that the relative risk is increased. The data are the alcohol-related diseases in Finland in 2001–2005. (Borderlines © Affecto Finland Oy, Karttakeskus, License L8573/10): (a) CAR model without empty cells; (b) CAR model with empty cells; and (c) GP model.

than GP, but the posterior mean does not differ that dramatically between the models. The cells that CAR gives much higher variance than GP are located on the borders of Finland, and in case of CAR model without empty cells also in Lapland. Thus, the posterior precision of latent variables in CAR models is highly altered by the number of the neighbors. Similar results have also been reported, for example, by Rue and Held [10, page 105]. Overall, the CAR model gives certain cells very awkward values, since the expectation of the latent variables ranges from -20 to 0.7 and the variance from 4×10^{-4} to 3×10^{4}, whereas with GP the expectations are between -0.9 and 0.7, and the variances between 0.004 and 0.23.

The predictive performance of the different models is evaluated using 10-fold cross-validation [60]. The data are divided into 10 groups so that $s(i)$ is the set of data points in the group where the i-th data point belongs. The comparison is done using the log predictive density diagnostics

$$\frac{1}{n}\sum_{i=1}^{n}\log p(y_i^{\mathrm{rep}}=y_i\,|\,\mathbf{y}^{\setminus s(i)},\mathbf{X}^{\setminus s(i)}),$$

where y_i^{rep} denotes the posterior predictive replicate of the number of deaths in the cell i given the data in the groups where data point y_i is excluded, $\{\mathbf{y}^{\setminus s(i)},\mathbf{X}^{\setminus s(i)}\}$, and n is the number of data points. This is equivalent to conditional predictive ordinate diagnostics [61]. The statistics were -0.4666 (GP with MAP estimate), -0.4665 (GP with integration), -0.4838 (CAR without empty cells), and -0.4669 (CAR with empty cells). The standard deviation of the diagnostics were 0.0055. In pairwise comparison the CAR model without empty cells was significantly worse than others but there was no statistically significant difference between the other models. The small difference in the predictive densities results from the fact that the variance of the Poisson observation model increases at the same rate with its mean. For this reason the differences in the posterior of the relative risk have small influence on the predictive density.

Copyright © 2010 John Wiley & Sons, Ltd.

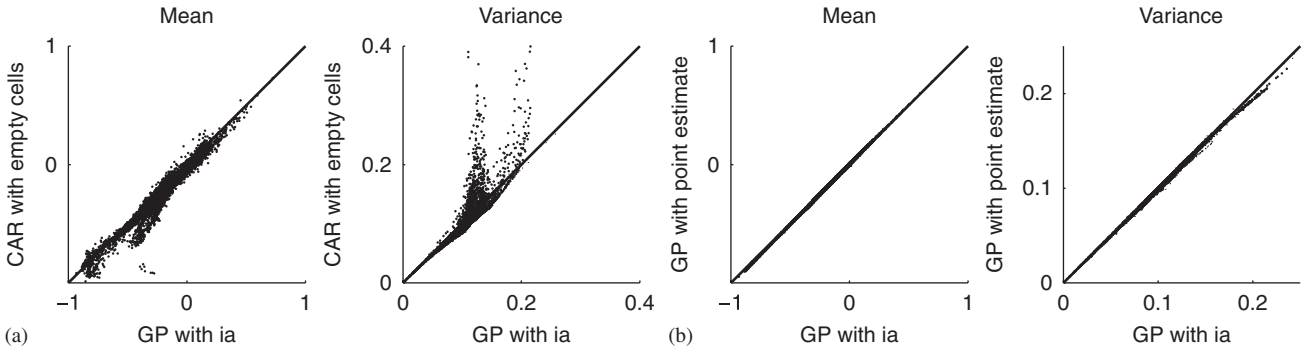

Figure 15. Scatter plots of the mean and the variance of the latent variable comparing the CAR model with empty cells, GP with integration over hyperparameters and GP with point estimate for the hyperparameters: (a) GP versus CAR with integration over hyperparameters and (b) GP with and without integration over hyperparameters.

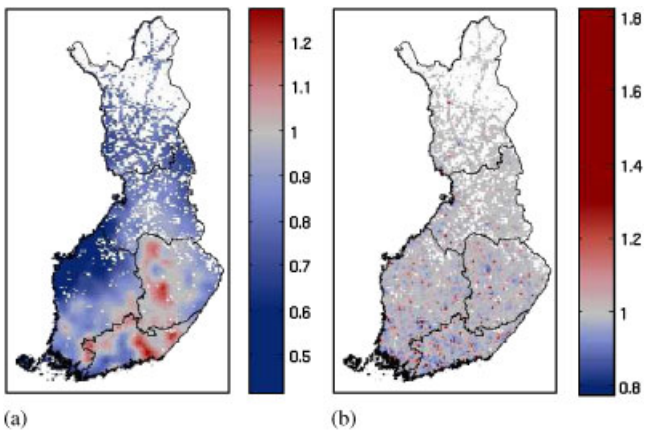

Figure 16. The posterior median of the long length-scale components and the short length-scale component of the relative risk in alcohol-related diseases: (a) The $k_{se}(\cdot,\cdot)+k_{ma}(\cdot,\cdot)$ component and (b) the $k_{pp}(\cdot,\cdot)$ component.

An advantage in using the GP model instead of CAR model is that we can investigate the long and short length-scale phenomena separately. This is illustrated in Figure 16, which shows the $k_{pp}(\cdot,\cdot)$ and the $k_{se}(\cdot,\cdot)+k_{ma}(\cdot,\cdot)$ components of the relative risk surface. The posterior modes of the hyperparameters were $l_{pp}=11.5$ km, $\sigma^2_{pp}=0.059$, $l_{se}=217$ km, $\sigma^2_{se}=0.097$, $l_{ma}=17$ km, and $\sigma^2_{ma}=0.036$. When the long and short length-scale phenomena are separated the country wide differences are more clearly visible than when plotting the overall relative risk, since the sharp local changes are filtered which improves the contrast between different parts of Finland. Similarly, when we examine only the short length-scale phenomenon we can better see the local changes in the relative risk and study, for example, the differences inside cities or counties.

Optimizing the hyperparameters of CS+FIC into their mode took about 10 min (12 optimization steps) with Laplace approximation and the integration required an extra 50 min with CCD (this time comprises evaluating the Hessian with finite differences and the 45 design points). INLA gave the results in about 4 min for the model without empty cells and in about 10 min for the model with empty cells. In theory, the inference time with INLA scales between $O(n\log(n)^2)$ and $O(n^2\log(n))$ with spatial data [13] and, thus, the 150 per cent increase in inference time is more than expected in this case. Supposedly INLA convergences slower if there are cells without data in the model. With more empty cells in the data INLA would slow down even more drastically, whereas the inference time with GP would remain the same. Much of the difference between the actual inference times between GP and CAR model comes from INLA's more efficient implementation. Our GP models have been implemented with Matlab scripts whereas INLA is a standalone program written with C. However, even with the present implementation GP is only slightly slower than INLA.

8. Discussion

Here, we discuss the results presented in the experiments. We also discuss briefly the possible future research directions motivated by this work. The code, with demos, used in this paper is publicly available at http://www.lce.hut.fi/research/mm/gpstuff/.

Copyright © 2010 John Wiley & Sons, Ltd.

8.1. The approximate inference

The results in Section 6.1 show that the Laplace method and the EP algorithm are effective approximations for the conditional posterior $p(\mathbf{f}|\mathscr{D},\hat{\theta})$ and for the marginal likelihood $p(\mathscr{D}|\theta)$. From Figures 4 and 7, we can see that EP and Laplace approximation give practically the same posterior marginals as MCMC. The Poisson likelihood is very close to normal in case of large p and notably smaller y (Figures 7(b),(c)), and the Gaussian approximation should presumably work well in that case. However, the posterior density is very close to Gaussian also when both p and y are small (Figure 7(a)) since then the normal prior dominates the latent value posterior.

The difference between EP and the Laplace approximation is that EP fits the posterior moments better. In classification, this is crucial since the posterior of the latent values is rather far from normal [3, 24]. With the data and model used in this work, the posterior seems to be very close to normal, for which reason the performance of the Laplace approximation and EP are practically equal. The benefit from using Laplace approximation, however, is that it is much faster than EP (Section 6.4).

The accuracy of the approximate integration over hyperparameters is dominated by the choice of the integration method and its parameters. The approximation for the marginal likelihood from EP and the Laplace approximation are so accurate that their influence on the integration is negligible. Overall, the results seem to be rather insensitive to the marginalization over the hyperparameters. The posterior variance of the latent variables increases as we integrate over the hyperparameters but their expectation remains practically unaltered (see Figures 4 and 15). The predictive performance of the model is as good with point estimate as with the integrated results. The most accurate integration method is the grid search with small step size. However, with a large number of hyperparameters (over 4) this is inefficient in which case importance sampling and CCD give good approximations.

Based on the above discussion we propose the following for an inference procedure: (1) find the mode $\hat{\theta}_{\text{La}}$ with Laplace approximation, (2) find the mode $\hat{\theta}_{\text{EP}}$ with EP starting from $\hat{\theta}_{\text{La}}$, (3) compare the results from the Laplace approximation and EP, (4) if the results are far apart check the model, the optimization, and the first-order correction terms for EP, (5) for the final results conduct the integration using the grid search, the importance sampling or CCD depending on the number of hyperparameters.

8.2. The sparse Gaussian processes

The results in Section 6.2 show that the predictive performance of the FIC and PIC decreases as the length scale of the spatial phenomenon decreases. With FIC this results first in oversmoothing and when the model breaks down completely in a spotty map. With PIC the decrease in the model performance leads to sharp discontinuities between the blocks that are visible also in the map. In Section 6.3, it was illustrated how CS+FIC enhances the analysis over PIC since its correlation structure is continuous.

The computational speed of the sparse approximations depend on, among others, the number of inducing inputs, the block size, the sparsity of $\hat{\Lambda}$ and the initial values of hyperparameters. Thus, the results in Section 6.4 are only indicative on that respect. However, we can rather safely summarize that in case of one long length-scale spatial phenomenon FIC and PIC are equally fast. In case of both short and long length-scale spatial phenomena, FIC is the slowest, CS+FIC is the fastest with Laplace approximation, and PIC with EP.

Different sparse approximations fit for different kinds of data, for which reason we suggest the following: (1) conduct a crude analysis with FIC and PIC with sparsely located inducing inputs and one covariance function, (2) conduct analysis with PIC with two covariance functions, (3) if there seems to be additive phenomena conduct final analysis with CS+FIC, (4) otherwise tune PIC or FIC so that the block size and the inducing inputs are set well enough for one length scale.

8.3. Hyperprior

The reviewers were worried that the heavy tails of the Student-t distribution could allow for extreme values of the length scale, which would lead to very long range spatial correlation that is nonzero far beyond the extent of the study region. However, this would happen only with extreme amounts of data. The number of background population and death cases are usually so small that the data do not contain enough information to completely overrule the prior. We could also use a uniform prior on the range of data, but this would give equal probability for all length scales in its range. A Gamma prior on the other hand would give much more mass for small values than the half t-distribution.

To study the sensitivity of the models to the hyperprior, we run simulation experiments with different half-t priors as well as with uniform prior. Overall the results were not very sensitive to the prior specification. If the half-t prior was too narrow compared to the true value of the length scale, the posterior of the length scale tends to concentrate on smaller values than with uniform prior. As the length scale increased the magnitude tends to increase as well and the predictions remained alike. With very extreme true length-scale values the half-t prior restricted the posterior better than the uniform prior. In general, the predictive accuracy of the model is not very sensitive to the hyperparameter values as will be discussed next.

8.4. Robustness of the models and methods

The reviewers were concerned about the robustness of the approximations in case of model misfit. The model misfit is related more to the observation model than to the GP prior or the normal approximation. If the data contained outliers or observations with clearly higher variance than what the Poisson observation model predicts, the posterior of the relative risk would be forced to shift towards the observations. However, the posterior distribution would still be proportional to the Poisson likelihood times the log Gaussian prior, and thus the Normal approximation for the latent value posterior would be as justified as without outliers.

Copyright © 2010 John Wiley & Sons, Ltd.

In the presence of outliers, the model would try to fit to the data by shortening the length scale and increasing the magnitude of the covariance function since this enables rougher latent value surface. At the extreme the length scale would converge to zero and the latent field would become isotropic with covariance matrix $\sigma^2\mathbf{I}$.

If the data contain only few outliers, the Gaussian prior will most likely overrule the Poisson likelihood and oversmooth the relative risk on these areas. On the other hand, if the bulk of the data is overdispersed the latent field can become practically isotropic and the relative risks uncorrelated. This is, however, easy to observe from the hyperparameters. In our experiments with real data sets we sometimes observe that the length scale converges to very small value, practically smaller than the cell size. There are two main reasons for this. The more common reason is that a GP model with only one covariance function is overly simple. Many times the data contain long correlations together with sharper local effects, in which case overly flexible model fits the data better than overly smooth. This can be easily fixed by adding more components to GP. The other, and more troublesome reason is the overdispersion. In this case, an additive model adapts to data by pulling the shorter length scale close to zero and modeling the long length-scale effect with the other covariance function. In these situations we could use a more robust negative-Binomial observation model. However, its implementation for the framework discussed in this paper is still under construction.

The reviewers also pointed out that the hyperparameters may not be well identified from the data. The nonidentifiability of the hyperparameters in spatial modeling is well treated in the literature. For example, recently Zhang [62] considered the asymptotics in estimating the parameters of Matérn class covariance functions. He shows that in a fixed region two GPs with Matérn covariance function cannot be correctly distinguished from each other with probability one no matter how many sample data are observed. Also the length scale, l, or the magnitude, σ^2, are not consistently estimable, since these two parameters are tied together. A quantity σ^2/l can be consistently estimated from the data, and it is more important to interpolation (and thus for predictive performance) than individual parameters as long as the parameters are in a sensible range. These results are asymptotic and, do not necessarily hold with finite data. However, they still show that identifying the hyperparameters in GP models can be problematic. The object of the inference in our work is, however, the relative risk that is a function of the latent variable and, thus, not affected that severely by the identification problems in hyperparameters. In general, the predictive performance of the model seems to be rather insensitive to the choice of the hyperparameters, which is also seen from the experiments in this paper.

8.5. Gaussian process or CAR model

Our findings on the performance of the GP and CAR models in disease mapping context were rather different from the earlier results given by Best *et al.* [11]. They compared several different spatial models with a simulated data set and concluded that a multivariate normal prior (same as GP prior for fixed set of locations) smooths the relative risk surface too much whereas CAR model (or Besag York Mollié model in their terminology) worked better. The results in Section 7, however, show that the predictive performance of GP and CAR model are practically equal but GP is more consistent in its results. CAR model might give unnaturally large or small expectations or variances for relative risk in cells that have only few neighbors. The CAR model also suffers from empty areas either in model performance, if these areas are left out, or in computational time, if they are included in the model. Presumably, the reason for the differing results is that Best *et al.* [11] used the GP model with single covariance function and their data did not have empty areas, whereas our model was additive and data areally very sparse. Our experience is also that GP models with only one covariance function tend to give smooth results, but with several components it can adapt to very fast changes as well. In addition the choice of the covariance functions matter. Best *et al.* [11] also criticize GP for its slowness. However, with the techniques described in this work we can overcome this deficiency and with areally very sparse data sets even gain speedup compared to CAR since we can exclude all the empty cells from the analysis, which would degrade the performance of the CAR model.

Acknowledgements

This research was funded by the Academy of Finland, Finnish Funding Agency for Technology and Innovation (TEKES), and the Graduate School in Electronics and Telecommunications and Automation (GETA). The first author also thanks the Finnish Foundation for Economic and Technology Sciences—KAUTE, Finnish Cultural Foundation, and Emil Aaltonen Foundation for supporting his post graduate studies. We also want to thank the reviewers for their comments that helped to improve the paper.

Appendix A: Laplace approximation for the marginal likelihood

In this section, we summarize the derivation of the Laplace approximation for the marginal likelihood. Earlier treatment of the approximation with GP prior is given in [1, 7].

The Taylor series of the log conditional posterior of the latent values around $\hat{\mathbf{f}}$ is

$$\log p(\mathbf{f}|\mathscr{D},\theta) \approx \log p(\hat{\mathbf{f}}|\mathscr{D},\theta) + \frac{1}{2}(\mathbf{f}-\hat{\mathbf{f}})^T[\nabla\nabla \log p(\mathbf{f}|\mathscr{D},\theta)]_{\mathbf{f}=\hat{\mathbf{f}}}(\mathbf{f}-\hat{\mathbf{f}}) + \cdots,$$

Copyright © 2010 John Wiley & Sons, Ltd.

where the linear term has vanished, because $\hat{\mathbf{f}}$ is the mode of log posterior [7, 29]. Close by $\hat{\mathbf{f}}$ the higher terms become negligible and the approximation is effectively a sum of constant and a logarithm of Gaussian term. Thus we can approximate the conditional posterior as

$$p(\mathbf{f}|\mathscr{D},\theta)\approx q(\mathbf{f}|\mathscr{D},\theta)=\mathrm{N}(\mathbf{f}|\hat{\mathbf{f}},H).$$

To approximate the marginal likelihood by Laplace method, we write it as follows:

$$p(\mathscr{D}|\theta)=\int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{X},\theta)\,\mathrm{d}\mathbf{f}=\int \exp(g(\mathbf{f}))\,\mathrm{d}\mathbf{f}, \tag{A1}$$

and make a Taylor expansion of $g(\mathbf{f})$ around $\hat{\mathbf{f}}$. This gives us

$$g(\mathbf{f})\approx g(\hat{\mathbf{f}})-\tfrac{1}{2}(\mathbf{f}-\hat{\mathbf{f}})^{\mathrm{T}}(\mathbf{K}_{\mathrm{f,f}}^{-1}+W)(\mathbf{f}-\hat{\mathbf{f}})+\cdots.$$

Taking only the first two terms and plugging them into (A1) gives

$$\begin{aligned} p(\mathscr{D}|\theta) &\approx q(\mathscr{D}|\theta)\\ &= \exp(g(\hat{\mathbf{f}}))\int \exp\left(-\frac{1}{2}(\mathbf{f}-\hat{\mathbf{f}})^{\mathrm{T}}(\mathbf{K}_{\mathrm{f,f}}^{-1}+W)(\mathbf{f}-\hat{\mathbf{f}})\right)\mathrm{d}\mathbf{f}\\ &= p(\mathbf{y}|\hat{\mathbf{f}})\frac{1}{(2\pi)^{n/2}}|\mathbf{K}_{\mathrm{f,f}}|^{-1/2}\exp\left(-\frac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}\mathbf{K}_{\mathrm{f,f}}^{-1}\hat{\mathbf{f}}\right)(2\pi)^{n/2}|\mathbf{K}_{\mathrm{f,f}}^{-1}+W|^{-1/2} \end{aligned} \tag{A2}$$

where the last two terms are the result from the integral. Now by taking the logarithm of this we obtain the approximate log marginal likelihood

$$\begin{aligned} \log q(\mathscr{D}|\theta) &= \log p(\mathbf{y}|\hat{\mathbf{f}})-\frac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}\mathbf{K}_{\mathrm{f,f}}^{-1}\hat{\mathbf{f}}-\frac{1}{2}\log|\mathbf{K}_{\mathrm{f,f}}|-\frac{1}{2}\log|\mathbf{K}_{\mathrm{f,f}}^{-1}+W|\\ &= \log p(\mathbf{y}|\hat{\mathbf{f}})-\frac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}\mathbf{K}_{\mathrm{f,f}}^{-1}\hat{\mathbf{f}}-\frac{1}{2}\log|B|. \end{aligned} \tag{A3}$$

where $B=|\mathbf{K}_{\mathrm{f,f}}||\mathbf{K}_{\mathrm{f,f}}^{-1}+W|=|I+W^{1/2}\mathbf{K}_{\mathrm{f,f}}W^{1/2}|$.

Appendix B: Gradients of the Laplace approximation of the marginal likelihood

The gradients of the approximate marginal likelihood with respect to the hyperparameters are derived in [1, 7]. Here we summarize the derivation so that we can use the results in Appendix C.

The gradient of the approximate marginal likelihood at the posterior mode $\hat{\mathbf{f}}$ is

$$\frac{\partial\log q(\mathscr{D}|\theta)}{\partial\theta_j}=\frac{\partial\log q(\mathscr{D}|\theta)}{\partial\theta_j}|_{\mathrm{explicit}}+\sum_{i=1}^{n}\frac{\partial\log q(\mathscr{D}|\theta)}{\partial\hat{f}_i}\frac{\partial\hat{f}_i}{\partial\theta_j},$$

which is a sum of explicit and implicit terms. The explicit term can be written as

$$\frac{\partial\log q(\mathscr{D}|\theta)}{\partial\theta_j}|_{\mathrm{explicit}}=\frac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}\mathbf{K}_{\mathrm{f,f}}^{-1}\frac{\partial\mathbf{K}_{\mathrm{f,f}}}{\partial\theta_j}\mathbf{K}_{\mathrm{f,f}}^{-1}\hat{\mathbf{f}}-\frac{1}{2}\mathrm{tr}\left((\mathbf{W}^{-1}+\mathbf{K}_{\mathrm{f,f}})^{-1}\frac{\partial\mathbf{K}_{\mathrm{f,f}}}{\partial\theta_j}\right)$$

The terms in the implicit derivative are

$$\frac{\partial\log q(\mathscr{D}|\theta)}{\partial\hat{f}_i}=-\frac{1}{2}[(\mathbf{K}_{\mathrm{f,f}}^{-1}+\mathbf{W})^{-1}]_{ii}\frac{\partial^3}{\partial\hat{f}_i^3}\log p(\mathbf{y}|\hat{\mathbf{f}}) \tag{B1}$$

$$\frac{\partial\hat{\mathbf{f}}}{\partial\theta_j}=(\mathbf{I}+\mathbf{K}_{\mathrm{f,f}}\mathbf{W})^{-1}\frac{\partial\mathbf{K}_{\mathrm{f,f}}}{\partial\theta_j}\nabla\log p(\mathbf{y}|\hat{\mathbf{f}}). \tag{B2}$$

Setting $\mathbf{w}_1=\mathbf{K}_{\mathrm{f,f}}^{-1}\hat{\mathbf{f}}$, $\mathbf{w}_2=[\partial\log q(\mathscr{D}|\theta)/\partial\hat{\mathbf{f}}](\mathbf{I}+\mathbf{K}_{\mathrm{f,f}}\mathbf{W})^{-1}$ and denoting $\mathbf{w}_3=[\nabla\log p(\mathbf{y}|\hat{\mathbf{f}})]^{\mathrm{T}}$ we can write the gradient as

$$\frac{\partial\log q(\mathscr{D}|\theta)}{\partial\theta_j}=\frac{1}{2}\mathbf{w}_1^{\mathrm{T}}\frac{\partial\mathbf{K}_{\mathrm{f,f}}}{\partial\theta_j}\mathbf{w}_1-\frac{1}{2}\mathrm{tr}\left((\mathbf{W}^{-1}+\mathbf{K}_{\mathrm{f,f}})^{-1}\frac{\partial\mathbf{K}_{\mathrm{f,f}}}{\partial\theta_j}\right)+\mathbf{w}_2\frac{\partial\mathbf{K}_{\mathrm{f,f}}}{\partial\theta_j}\mathbf{w}_3. \tag{B3}$$

For numerical stability we use $(\mathbf{W}^{-1}+\mathbf{K}_{\mathrm{f,f}})^{-1}=\mathbf{W}^{1/2}(\mathbf{I}+\mathbf{W}^{1/2}\mathbf{K}_{\mathrm{f,f}}\mathbf{W}^{1/2})^{-1}\mathbf{W}^{1/2}$.

Copyright © 2010 John Wiley & Sons, Ltd.

Appendix C: Laplace approximation with sparse Gaussian processes

With FIC, PIC, and CS+FIC, the covariance matrix $\mathbf{K}_{\mathrm{f,f}}$ in the approximate marginal likelihood (A3) is changed to $\mathbf{Q}_{\mathrm{f,f}}+\hat{\Lambda}$, where $\hat{\Lambda}$ is sparse. Now, the second term in the marginal likelihood can be evaluated efficiently from

$$-\frac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}(\mathbf{Q}_{\mathrm{f,f}}+\hat{\Lambda})^{-1}\hat{\mathbf{f}}=-\frac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}\hat{\Lambda}^{-1}\hat{\mathbf{f}}+\frac{1}{2}\hat{\mathbf{f}}^{\mathrm{T}}\mathbf{V}\mathbf{V}^{\mathrm{T}}\hat{\mathbf{f}},$$

where $\mathbf{V}=\hat{\Lambda}^{-1}\mathbf{K}_{\mathrm{f,u}}\mathrm{chol}[(\mathbf{K}_{\mathrm{u,u}}+\mathbf{K}_{\mathrm{u,f}}\hat{\Lambda}^{-1}\mathbf{K}_{\mathrm{f,u}})^{-1}]$. The determinant of the matrix B can also be evaluated more efficiently from the relation

$$\begin{aligned}|B| &= |I+W^{1/2}\mathbf{K}_{\mathrm{f,u}}\mathbf{K}_{\mathrm{u,u}}^{-1}\mathbf{K}_{\mathrm{u,f}}W^{1/2}+W^{1/2}\hat{\Lambda}W^{1/2}|\\ &= |\hat{\Lambda}_2||\mathbf{K}_{\mathrm{u,u}}^{-1}||\mathbf{K}_{\mathrm{u,u}}+W^{1/2}\mathbf{K}_{\mathrm{f,u}}\hat{\Lambda}_2^{-1}\mathbf{K}_{\mathrm{u,f}}W^{1/2}|,\end{aligned}\tag{C1}$$

since here $\hat{\Lambda}_2=I+W^{1/2}\hat{\Lambda}W^{1/2}$ has the same sparsity structure as $\hat{\Lambda}$.

The second term in the gradient of the marginal likelihood is:

$$\mathrm{tr}\left((\mathbf{W}^{-1}+\hat{\Lambda}+\mathbf{Q}_{\mathrm{f,f}})^{-1}\frac{\partial(\hat{\Lambda}+\mathbf{Q}_{\mathrm{f,f}})}{\partial\theta_j}\right)=\mathrm{tr}\left(\mathbf{W}^{1/2}(\hat{\Lambda}_2^{-1}-\mathbf{V}_2\mathbf{V}_2^{\mathrm{T}})\mathbf{W}^{1/2}\frac{\partial(\hat{\Lambda}+\mathbf{Q}_{\mathrm{f,f}})}{\partial\theta_j}\right),$$

where $\mathbf{V}_2=\hat{\Lambda}_2W^{1/2}\mathbf{K}_{\mathrm{f,u}}\mathrm{chol}[(\mathbf{K}_{\mathrm{u,u}}+\mathbf{K}_{\mathrm{u,f}}W^{1/2}\hat{\Lambda}_2^{-1}W^{1/2}\mathbf{K}_{\mathrm{f,u}})^{-1}]$. With FIC and PIC this can be evaluated as described in [55]. With CS+FIC the evaluation is similar to FIC but we need to use the sparse inverse algorithm for all the terms that contain $\hat{\Lambda}_2^{-1}$.

To evaluate the first and the third terms in the gradient we need the vectors $\mathbf{w}_1$, $\mathbf{w}_2$, and $\mathbf{w}_3$. After this the first term in the gradient of the marginal likelihood is evaluated as described in [55] and the third term similar to that. The vectors $\mathbf{w}_1$ and $\mathbf{w}_3$ are straightforward to evaluate, but the vector $\mathbf{w}_2$ needs more consideration. The first term in the implicit derivative (B1) can be written as

$$\begin{aligned}\frac{\partial\log q(\mathscr{D}|\theta)}{\partial\hat{\mathbf{f}}} &= -\frac{1}{2}\mathrm{mask}((\hat{\Lambda}^{-1}-\mathbf{V}\mathbf{V}^{\mathrm{T}}+\mathbf{W})^{-1},\mathbf{I})\frac{\partial^3}{\partial\hat{\mathbf{f}}^3}\log p(\mathbf{y}|\hat{\mathbf{f}})\\ &= -\frac{1}{2}\mathrm{mask}(\hat{\Lambda}_3^{-1}+\mathbf{V}_3\mathbf{V}_3^{\mathrm{T}},\mathbf{I})\frac{\partial^3}{\partial\hat{\mathbf{f}}^3}\log p(\mathbf{y}|\hat{\mathbf{f}})\\ &= -\frac{1}{2}\mathrm{mask}(\hat{\Lambda}_3^{-1},\mathbf{I})\frac{\partial^3}{\partial\hat{\mathbf{f}}^3}\log p(\mathbf{y}|\hat{\mathbf{f}})-\frac{1}{2}\mathrm{mask}(\mathbf{V}_3\mathbf{V}_3^{\mathrm{T}},\mathbf{I})\frac{\partial^3}{\partial\hat{\mathbf{f}}^3}\log p(\mathbf{y}|\hat{\mathbf{f}}),\end{aligned}\tag{C2}$$

where $\hat{\Lambda}_3=\hat{\Lambda}^{-1}+W$ and $\mathbf{V}_3=\hat{\Lambda}_3^{-1}\mathbf{V}\mathrm{chol}[(\mathbf{I}-\mathbf{V}^{\mathrm{T}}\hat{\Lambda}_3^{-1}\mathbf{V})^{-1}]$. For FIC and PIC this is easy to evaluate, since the $\hat{\Lambda}$ is (block)diagonal. With CS+FIC we have to evaluate the inverse of $\hat{\Lambda}_3$ as follows:

$$\begin{aligned}\hat{\Lambda}_3^{-1} &= \hat{\Lambda}(\hat{\Lambda}+W^{-1})^{-1}W^{-1}\\ &= \hat{\Lambda}W^{1/2}(W^{1/2}\hat{\Lambda}W^{1/2}+\mathbf{I})^{-1}W^{-1/2}.\end{aligned}\tag{C3}$$

This is semi-optimal since we have the inverse of W in the equation, and the elements in W can be very close to zero. However, taking the square root first improves the numerical stability compared to the inversion of W. Using the above equation we can efficiently evaluate $\mathbf{V}_3$ using sparse matrix routines, after which the diagonal of $\mathrm{mask}(\mathbf{V}_3\mathbf{V}_3^{\mathrm{T}},\mathbf{I})$ is obtained with $O(nm)$ operations. The term $\mathrm{mask}(\hat{\Lambda}_3^{-1},\mathbf{I})$ can be evaluated using the sparse inverse algorithm. First, we evaluate the sparse inverse of $W^{1/2}\hat{\Lambda}W^{1/2}+\mathbf{I}$, and take the inner product between each row of $\hat{\Lambda}W^{1/2}$ and the corresponding row of the sparse inverse. After this each inner product is multiplied by the corresponding diagonal of $W^{-1/2}$. Now, we can evaluate $\mathbf{w}_2=\partial\log q(\mathscr{D}|\theta)/\partial\hat{\mathbf{f}}(I+\hat{\Lambda}W+\mathbf{Q}_{\mathrm{f,f}}W)^{-1}$ by using matrix inversion lemma for $(I+\hat{\Lambda}W+\mathbf{Q}_{\mathrm{f,f}}W)^{-1}$.

Appendix D: The weights for CCD method

To determine the integration weights for the design points, we assume $p(\theta|\mathscr{D})$ to be standard Gaussian after re-parameterization. Thus $E[\theta^{\mathrm{T}}\theta]=d$ and $\int p(\theta)\,d\theta=1$, where d is the dimensionality of θ. This gives the integration weights for the points on the sphere with radius $\sqrt{d}f_0$. Owing to the symmetry, the integration weights are equal for the points on the sphere.

$$E[\theta^{\mathrm{T}}\theta]=\sum_k\theta_k^{\mathrm{T}}\theta_k p(\theta_k)\Delta_k=(n_p-1)df_0^2(2\pi)^{-d/2}\exp\left(-\frac{df_0^2}{2}\right)\Delta=d,\tag{D1}$$

Copyright © 2010 John Wiley & Sons, Ltd.

where n_p is the number of the points on the sphere. From (D1) it follows that

$$\Delta = \left[(n_p - 1) f_0^2 (2\pi)^{-d/2} \exp\left(-\frac{df_0^2}{2} \right) \right]^{-1}. \tag{D2}$$

Integral over $p(\theta)$ is approximated with $p(\mathbf{0})\Delta_0 + \sum_k p(\theta_k)\Delta_k$. Thus the weight of the central point is

$$\Delta_0 = \frac{1 - \sum_k p(\theta_k)\Delta_k}{p(\mathbf{0})} = \frac{1 - (n_p - 1)(2\pi)^{-d/2} \exp\left(-\frac{df_0^2}{2} \right) \Delta}{(2\pi)^{-d/2}}$$

$$= (2\pi)^{d/2} \left(1 - \frac{1}{f_0^2} \right) \tag{D3}$$

The weights can be rescaled so that

$$\Delta_0 = 1 \tag{D4}$$

$$\Delta = \left[(n_p - 1) \exp\left(-\frac{df_0^2}{2} \right) (f_0^2 - 1) \right]^{-1}. \tag{D5}$$

The more-detailed discussion on the CCD method is provided by Martino [40]. There is, however, a typographical error in the case of integration weights (equation on page 18 of paper IV).

References

1. Williams CKI, Barber D. Bayesian classification with Gaussian processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 1998; **20**(12):1342–1351. DOI: 10.1109/34.735807.
2. Minka T. A family of algorithms for approximate Bayesian inference. *Ph.D. Thesis*, Massachusetts Institute of Technology, Cambridge, MA, 2001.
3. Kuss M, Rasmussen CE. Assessing approximate inference for binary Gaussian process classification. *Journal of Machine Learning Research* 2005; **6**:1679–1704.
4. Snelson E, Ghahramani Z. Sparse Gaussian process using pseudo-inputs. In *Advances in Neural Information Processing Systems 18*, Weiss Y, Schölkopf B, Platt J (eds). The MIT Press: Cambridge, MA, 2006.
5. Quiñonero-Candela J, Rasmussen CE. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research* 2005; **6**(3):1939–1959.
6. Vanhatalo J, Vehtari A. Modelling local and global phenomena with sparse Gaussian processes. *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*. AUAI Press: Corvallis, OR, 2008.
7. Rasmussen CE, Williams CKI. *Gaussian Processes for MachineLearning*. The MIT Press: Cambridge, MA, 2006.
8. Diggle PJ, Ribeiro PJ. *Model-based Geostatistics*. Springer: Berlin, 2007.
9. Möller J, Syversveen AR, Waagepetersen RP. Log Gaussian Cox processes. *Scandinavian Journal of Statistics* 1998; **25**:451–482. DOI: 10.1111/1467-9469.00115.
10. Rue H, Held L. *Gaussian Markov Random Fields Theory and Applications*. Chapman & Hall/CRC: London/Boca Raton, 2005.
11. Best N, Richardson S, Thomson A. A comparison of Bayesian spatial models for disease mapping. *Statistical Methods in Medical Research* 2005; **14**:35–59. DOI: 10.1191/0962280205sm388oa.
12. Elliot P, Wakefield J, Best N, Briggs D (eds). *Spatial Epidemiology Methods and Applications*. Oxford University Press: Oxford, 2001.
13. Rue H, Martino S, Chopin N. Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of Royal Statistical Society B* 2009; **71**(2):1–35. DOI: 10.1111/j.1467-9868.2008.00700.x.
14. Csató L, Opper M. Sparse online Gaussian processes. *Neural Computation* 2002; **14**(3):641–669. DOI: 10.1162/089976602317250933.
15. Seeger M, Williams CKI, Lawrence N. Fast forward selection to speed up sparse Gaussian process regression. In *Ninth International Workshop on Artificial Intelligence and Statistics*, Bishop CM, Frey BJ (eds). Society for Artificial Intelligence and Statistics: New Jersey, U.S.A., 2003.
16. Wikle CK, Cressie N. A dimension-reduced approach to space–time Kalman filtering. *Biometrica* 1999; **86**:815–829. DOI: 10.1093/biomet/86.4.815.
17. Cornford D, Csató L, Opper M. Sequential, Bayesian geostatistics: a principle method for large data sets. *Geographical Analysis* 2005; **37**:183–199. DOI: 10.1111/j.1538-4632.2005.00635.x.
18. Paciorek CJ. Computational techniques for spatial logistic regression with large datasets. *Computational Statistics and Data Analysis* 2007; **51**:3631–3653. DOI: 10.1016/j.csda.2006.11.008.
19. Banerjee S, Gelfand AE, Finley AO, Sang H. Gaussian predictive process models for large spatial data sets. *Journal of Royal statistical Society B* 2008; **70**(4):825–848. DOI: 10.1111/j.1467-9868.2008.00663.x.
20. Wendland H. *Scattered Data Approximation*. Cambridge University Press: Cambridge, 2005.
21. Storkey A. Efficient covariance matrix methods for Bayesian Gaussian processes and Hopfield neural networks. *Ph.D. Thesis*, University of London, U.K., 1999.
22. Gibbs MN, Mackay DJC. Variational Gaussian process classifiers. *IEEE Transactions on Neural Networks* 2000; **11**(6):1458–1464.

Copyright © 2010 John Wiley & Sons, Ltd.

23. Csató L, Fokoué E, Opper M, Schottky B. Efficient approaches to Gaussian process classification. *Neural Information Processing Systems*. MIT Press: Cambridge, MA, 2000; 251–257.
24. Nickisch H, Rasmussen CE. Approximations for binary Gaussian process classification. *Journal of Machine Learning Research* 2008; **9**:2035–2078.
25. Snelson E. Flexible and efficient Gaussian process models for machine learning. *Ph.D. Thesis*, University College London, London, U.K., 2007.
26. Ahmad OB, Boschi-Pinto C, Lopez AD, Murray CJ, Lozano R, Inoue M. Age standardization of rates: a new WHO standard. *GPE Discussion Paper Series*, World Health Organization, 2000; 31.
27. Gneiting T. Compactly supported correlation functions. *Journal of Multivariate Analysis* 2002; **83**:493–508. DOI: 10.1006/jmva.2001.2056.
28. Gelman A. Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis* 2006; **1**(3):515–533. DOI: 10.1214/06-BA117A.
29. Gelman A, Carlin JB, Stern HS, Rubin DB. *Bayesian Data Analysis, Second Edition*. Chapman and Hall/CRC: London/Boca Raton, 2004.
30. Seeger M. Expectation propagation for exponential families. *Technical Report*, Max Planck Institute for Biological Cybernetics, Tübingen, Germany, 2005.
31. Zoeter O, Heskes T. Gaussian quadrature based expectation propagation. In *Proceedings of the Ninth International Workshop on Artificial Intelligence and Statistics*, Cowell RG, Ghahramani Z (eds). Society for Artificial Intelligence and Statistics: New Jersey, U.S.A., 2005; 445–452.
32. Shampine LF. Vectorized adaptive quadrature in MATLAB. *Journal of Computational and Applied Mathematics* 2008; **211**:131–140. DOI: 10.1016/j.cam.2006.11.021.
33. Neal RM. *Bayesian Learning for Neural Networks*. Springer: Berlin, 1996.
34. Niederreiter H. *Random Number Generation and Quasi-Monte Carlo Methods*. Society for Industrial and Applied Mathematics: Philadelphia, U.S.A., 1992.
35. Morokoff WJ, Caflisch RE. Quasi-Monte Carlo integration. *Journal of Computational Physics* 1995; **122**:218–230. DOI: 10.1006/jcph.1995.1209.
36. Geweke J. Bayesian inference in econometric models using Monte Carlo integration. *Econometrica* 1989; **57**(6):721–741.
37. Vehtari A. Bayesian model assessment and selection using expected utilities. *Ph.D. Thesis*, Helsinki University of Technology, Finland, 2001.
38. Hammersley JM. Monte Carlo methods for solving multivariate problems. *Annals of the New York Academy of Sciences* 1960; **86**(3):844–874. DOI: 10.1111/j.1749-6632.1960.tb42846.x.
39. Sanchez SM, Sanchez PJ. Very large fractional factorials and central composite designs. *ACM Transactions on Modeling and Computer Simulation* 2005; **15**:362–377. DOI: 10.1145/1113316.1113320.
40. Martino S. Approximate Bayesian inference for latent Gaussian models. *Ph.D. Thesis*, Norwegian University of Science and Technology, Trondheim, 2007.
41. Tierney L, Kadane JB. Accurate approximations for posterior moments and marginal densities. *Journal of the American Statistical Association* 1986; **81**(393):82–86.
42. Spiegelhalter DJ, Best NG, Carlin BP, van der Linde A. Bayesian measures of model complexity and fit. *Journal of Royal Statistical Society B* 2002; **64**(4):583–639. DOI: 10.1111/1467-9868.00353.
43. Paquet U, Winther O, Opper M. Perturbation corrections in approximate inference: mixture modelling applications. *Journal of Machine Learning Research* 2009; **10**:1263–1304.
44. Snelson E, Ghahramani Z. Local, global sparse Gaussian process approximations. In *Proceedings of the Eleventh International Workshop on Artificial Intelligence and Statistics*, Meila M, Shen X (eds). Omnipress: New Jersey, 2007.
45. Lawrence N. Learning for larger datasets with the Gaussian process latent variable model. In *Proceedings of the Eleventh International Workshop on Artificial Intelligence and Statistics*, Meila M, Shen X (eds). Omnipress: New Jersey, U.S.A., 2007.
46. Harville DA. *Matrix Algebra from a Statistician's Perspective*. Springer: Berlin, 1997.
47. Naish-Guzman A, Holden S. The generalized FITC approximation. In *Advances in Neural Information Processing Systems 20*, Platt J, Koller D, Singer Y, Roweis S (eds). MIT Press: Cambridge, MA, 2008.
48. Amestoy P, Davis TA, Duff IS. Algorithm 837: AMD, an approximate minimum degree ordering algorithm. *ACM Transactions on Mathematical Software* 2004; **30**(3):381–388. DOI: 10.1145/1024074.1024081.
49. Davis TA. *Direct Methods for Sparse Linear Systems*. SIAM: Philadelphia, PA, 2006.
50. Takahashi K, Fagan J, Chen MS. Formation of a sparse bus impedance matrix and its application to short circuit study. *Power Industry Computer Application Conference Proceedings*. IEEE Power Engineering Society: Minneapolis, U.S.A., 1973.
51. Niessner H, Reichert K. On computing the inverse of a sparse matrix. *International Journal for Numerical Methods in Engineering* 1983; **19**:1513–1526. DOI: 10.1002/nme.1620191009.
52. Rue H, Martino S. Approximate Bayesian inference for hierarchical Gaussian Markov random field models. *Journal of Statistical Planning and Inference* 2007; **137**:3177–3192. DOI: 10.1016/j.jspi.2006.07.016.
53. Davis TA, Hager WW. Row modifications of a sparse Cholesky factorization. *SIAM Journal on Matrix Analysis and Applications* 2005; **26**(3):621–639. DOI: 10.1137/S089547980343641X.
54. George A, William G, Poole J, Voigt RG. Incomplete nested dissection for solving *n* by *n* grid problems. *SIAM Journal on Numerical Analysis* 1978; **15**(4):662–673. DOI: 10.1137/0715044.
55. Vanhatalo J, Vehtari A. Sparse log Gaussian processes via MCMC for spatial epidemiology. *JMLR Workshop and Conference Proceedings* 2007; **1**:73–89.
56. Brooks SP, Gelman A. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics* 1998; **7**(4):434–455.
57. Robert CP, Casella G. *Monte Carlo Statistical Methods* (2nd edn). Springer: Berlin, 2004.
58. Geyer CJ. Practical Markov chain Monte Carlo. *Statistical Science* 1992; **7**(4):473–511. DOI: 10.1214/ss/1177011137.
59. Neal RM. Annealed importance sampling. *Statistics and Computing* 2001; **11**:125–139. DOI: 10.1023/A:1008923215028.
60. Vehtari A, Lampinen J. Bayesian model assessment and comparison using cross-validation predictive densities. *Neural Computation* 2002; **14**(10):2439–2468.
61. Gelfand AE, Dey DK, Chang H. Model determination using predictive distributions with implementation via sampling-based methods (with Discussion). In *Bayesian Statistics 4*, Bernardo JM, Berger JO, Dawid AP, Smith AFM (eds). Oxford University Press: Oxford, 1992; 147–167.
62. Zhang H. Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics. *Journal of the American Statistical Association* 2004; **99**(465):250–261. DOI: 10.1198/016214504000000241.

Copyright © 2010 John Wiley & Sons, Ltd.