

Supplementary materials: Large Scale Mapping of Indoor Magnetic Field by Local and Sparse Gaussian Processes

Anonymous Author(s)

Affiliation

Address

email

1 A Addendum on Gaussian Identities

2 This section will be merged into the "Gaussian Identities" appendix in the camera-ready version.

3 Let $p(x) = \mathcal{N}(x|\mu_0, \Sigma_0)$ and $q(x) = \mathcal{N}(x|\mu_1, \Sigma_1)$ be the densities of two multivariate Gaussian
4 vector of the same dimension, then their quotient admits also a Gaussian expression

$$\frac{p(x)}{q(x)} \propto \exp\left(-\frac{1}{2}(x - \mu_3)^\top \Sigma_3^{-1}(x - \mu_3)\right), \quad (1)$$

where

$$\mu_3 = \Sigma_3(\Sigma_0^{-1}\mu_0 - \Sigma_1^{-1}\mu_1), \quad \Sigma_3 = (\Sigma_0^{-1} - \Sigma_1^{-1})^{-1}.$$

5 Notice that Σ_3 can be non-positive, making this quotient unsuitable for a random variable density.
6 However, if Σ_3 is definite positive, this expression can be normalized into a proper Gaussian density
7 $\mathcal{N}(\mu_3, \Sigma_3)$.

8 B Definite Positive Symmetric Matrices

9 Here, we introduce some well-known results about definite positive symmetric matrices that will be
10 useful to manipulate covariances and derive the LBCM posterior in appendix C. We use the standard
11 notation $S \succ 0$ and $S_1 \succ S_2$ to indicate that the symmetric matrices S and $S_1 - S_2$ are definite
12 positive.

13 **Theorem 1** *Let S_1, S_2 be two symmetric positive matrices with at least one definite positive, and*
14 *α_1, α_2 two strictly positive scalars, then $\alpha_1 S_1 + \alpha_2 S_2 \succ 0$.*

15 **Theorem 2** *Let S be a symmetric definite positive matrix, then $S^{-1} \succ 0$.*

16 **Theorem 3** *Let S_1, S_2 be two symmetric definite positive matrices, then $S_1 \succ S_2$ if and only if*
17 *$S_2^{-1} \succ S_1^{-1}$.*

18 **Theorem 4** *Let S be a symmetric definite positive matrix and V a (possibly rectangular) matrix*
19 *with full column rank, then $A^\top S A \succ 0$.*

20 C Derivation of LBCM Posterior

21 In this section, we detail the derivation of the LBMC posterior. It is based on an approximation of
22 the BCM that we can also quickly derive here for completeness.

23 In general, if we have J experts trained on a partition $\mathcal{D}_1, \dots, \mathcal{D}_J$ of the dataset \mathcal{D} , then from Bayes
24 theorem:

$$p(f_* | \mathcal{D}) \stackrel{\text{Bayes}}{\propto} p(\mathcal{D} | f_*) p(f_*) = p(\mathcal{D}_1, \dots, \mathcal{D}_J | f_*) p(f_*). \quad (2)$$

25 The BCM introduces a conditional independence approximation between the \mathcal{D}_i , which we inject in
 26 (2), and then we use Bayes a second time on each expert posterior

$$p(\mathcal{D}_1, \dots, \mathcal{D}_J | f_*) p(f_*) \stackrel{\text{BCM}}{\propto} p(f_*) \prod_{i=1}^J p(\mathcal{D}_i | f_*) \stackrel{\text{Bayes}}{\propto} p(f_*) \prod_{i=1}^J \frac{p(f_* | \mathcal{D}_i)}{p(f_*)} = \frac{\prod_{i=1}^J p(f_* | \mathcal{D}_i)}{p(f_*)^{J-1}} \quad (3)$$

27 Then, we recall that we inject our geometric approximation in (3) to get the LBCM posterior

$$p_{\text{LBCM}}(f_* | \mathcal{D}) \propto \frac{\prod_{i=1}^J p(f_* | \mathcal{D}_i)^{\beta_i} p(f_*)^{1-\beta_i}}{p(f_*)^{J-1}} = \frac{\prod_{i \in A(x_*)} p(f_* | \mathcal{D}_i)^{\beta_i}}{p(f_*)^{-1 + \sum_{i \in A(x_*)} \beta_i}}. \quad (4)$$

28 where $A(x_*) = \{i | \beta_i \neq 0\}$. This expression comprises a quotient, products, and powers of Gaussian
 29 densities. We straightforwardly combine all the Gaussian identities from appendix A. With a zero
 30 mean Gaussian prior, the LBCM posterior has the following exponential form

$$p_{\text{LBCM}}(f_* | \mathcal{D}) \propto \exp \left(-\frac{1}{2} (x - \mu)^\top \Lambda (x - \mu) \right), \quad (5)$$

31 where

$$\begin{aligned} \mu &= \Lambda^{-1} \left(\sum_{i \in A(x_*)} \beta_i \text{cov}(f_* | \mathcal{D}_i)^{-1} \mathbb{E}(f_* | \mathcal{D}_i) \right), \\ \Lambda &= (1 - \sum_{i \in A(x_*)} \beta_i) \text{cov}(f_*)^{-1} + \sum_{i \in A(x_*)} \beta_i \text{cov}(f_* | \mathcal{D}_i)^{-1}. \end{aligned} \quad (6)$$

32 For p_{LBCM} to be a proper Gaussian density $\mathcal{N}(\mu, \Lambda^{-1})$, all there is left is to check that Λ^{-1} is
 33 symmetric definite positive. Symmetry is stable by sum, inverse, and multiplication by a scalar,
 34 so Λ^{-1} is symmetric. Showing positive definiteness is slightly more cumbersome. According to
 35 theorem 2, we can directly consider Λ . We start by rewriting it

$$\Lambda = \text{cov}(f_*)^{-1} + \sum_{i \in A(x_*)} \beta_i (\text{cov}(f_* | \mathcal{D}_i)^{-1} - \text{cov}(f_*)^{-1}). \quad (7)$$

36 which is a sum of matrices weighted by positive coefficients that would be positive definite if each
 37 matrix is positive definite as well (theorem 1). Using the inversion results (theorem 2), we have
 38 $\text{cov}(f_*)^{-1} \succ 0$. And according to 3, the matrix $\text{cov}(f_* | \mathcal{D}_i)^{-1} - \text{cov}(f_*)^{-1}$ is positive definite if
 39 and only if $\text{cov}(f_*) \succ \text{cov}(f_* | \mathcal{D}_i)$.

40 Intuitively, the prior covariance is larger than the posterior one. Thus, $\text{cov}(f_*) \succ \text{cov}(f_* | \mathcal{D}_i)$ should
 41 be valid for any sensible expert model, but to finish the formal proof, we need to limit ourself to
 42 specific examples. For instance, using full GP experts with kernel κ_{SE} ,

$$\text{cov}(f_*) - \text{cov}(f_* | \mathcal{D}_i) = K_{\mathbf{f}, \mathbf{f}_*}^\top \Sigma K_{\mathbf{f}, \mathbf{f}_*}, \quad (8)$$

43 where $\Sigma = (K_{\mathbf{f}, \mathbf{f}} + \sigma_{\text{noise}}^2 I_{nd})^{-1}$. The matrix $K_{\mathbf{f}, \mathbf{f}_*}$ has full column rank and using theorem 1 and
 44 2 we have $\Sigma \succ 0$. From theorem 4, it follows that $\text{cov}(f_*) \succ \text{cov}(f_* | \mathcal{D}_i)$ holds for each GP expert
 45 and LBCM is well defined. If, instead, we use the DTC or the G-DTC sparse approximation defined
 46 from this GP

$$\text{cov}(f_*) - \text{cov}(f_* | \mathcal{D}_i) = K_{\mathbf{u}, \mathbf{f}_*}^\top (K_{\mathbf{u}, \mathbf{u}}^{-1} - S) K_{\mathbf{u}, \mathbf{f}_*}, \quad (9)$$

47 where $S = (\sigma_{\text{noise}}^{-2} K_{\mathbf{u}, \mathbf{f}} K_{\mathbf{f}, \mathbf{u}} + K_{\mathbf{u}, \mathbf{u}})^{-1}$. We can see that $S^{-1} = \sigma_{\text{noise}}^{-2} K_{\mathbf{u}, \mathbf{f}} K_{\mathbf{f}, \mathbf{u}} + K_{\mathbf{u}, \mathbf{u}} \succ K_{\mathbf{u}, \mathbf{u}}$,
 48 thus from theorem 3, we have $K_{\mathbf{u}, \mathbf{u}}^{-1} - S \succ 0$. It follows from theorem 4 that $\text{cov}(f_*) \succ \text{cov}(f_* | \mathcal{D}_i)$
 49 holds for each experts. Therefore, LBCM is again well-defined.

50 D Partial Grid

51 G-DTC is based on extracting a partial grid of latent inputs near the training dataset. Here, we give
 52 efficient computation techniques for the partial grid, describe some subtleties about latent inputs
 53 near the subdomains border, and motivate the chosen value for the parameter R .

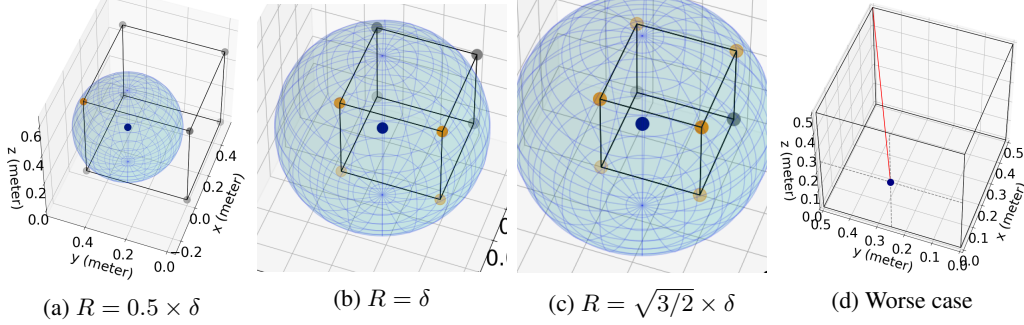


Figure 1: **Vertices selection from one cell of the cubic grid.** The selected vertices (orange) are extracted from a cubic grid (gray) at a distance R or less than an input x (blue). We represent one grid cell only for clarity. (a) R is too small, and only one vertex is selected. (b) R is still too small because the four selected vertices are coplanar (beware, the 4 gray vertices on the back are outside the sphere)(c) R is larger, and the six selected vertices are not coplanar anymore. (d) In the worst case, the input is located in the middle of a face, and R should be greater than the red segment length $\sqrt{\delta^2 + (\delta/2)^2 + (\delta/2)^2} = \sqrt{3/2} \times \delta$.

54 Given inputs x_1, \dots, x_n and a radius R , we retrieve all the vertices z_1, \dots, z_M from a cubic grid
 55 of step δ , that are at a distance R or less from at least one x_i . To do so efficiently, for each x_i ,
 56 we generate all the vertices z in a box of center x_i and side length $2R$ (the generation process is
 57 described below). Then, we loop through all the vertices of this box to remove those at a distance
 58 greater than R from x_i . Each remaining z is inserted in $\mathcal{O}(\log(M))$ into a *set* data structure to avoid
 59 duplicates. Thus, the partial grid is created in $\mathcal{O}((R/\delta)^3 N \log(M))$ operations, where $(R/\delta)^3$ is
 60 proportional to the number of vertices in a ball of radius R .

61 All that is left is to describe the vertices generation process in a box of center $x = (\alpha_1, \dots, \alpha_{d'})$ and
 62 side length $2R$. Let $p_0 = (\gamma_1, \dots, \gamma_{d'})$ be the origin of the cubic grid of step δ , then it is possible to
 63 describe any vertex z by its *grid macro coordinates* $v \in \mathbb{Z}^{d'}$:

$$z = \delta \times v + p_0 \quad (10)$$

64 The lowest and highest macro coordinate values of all the vertices inside the box are respectively

$$\begin{aligned} v_{\min} &= \left(\left\lceil \frac{\alpha_1 - R - \gamma_1}{\delta} \right\rceil, \dots, \left\lceil \frac{\alpha_{d'} - R - \gamma_{d'}}{\delta} \right\rceil \right) \\ v_{\max} &= \left(\left\lfloor \frac{\alpha_1 + R - \gamma_1}{\delta} \right\rfloor, \dots, \left\lfloor \frac{\alpha_{d'} + R - \gamma_{d'}}{\delta} \right\rfloor \right) \end{aligned} \quad (11)$$

65 where $\lceil a \rceil$ and $\lfloor a \rfloor$ denotes the ceiling and integer part of a scalar a respectively. Then
 66 $\llbracket v_{\min,1}, v_{\max,1} \rrbracket \times \dots \times \llbracket v_{\min,d'}, v_{\max,d'} \rrbracket$ is the list of *grid macro coordinates* of the vertices inside
 67 the box.

68 As a side note, when we split the space Ω in $\Omega_1, \dots, \Omega_J$ to create the data partition, we allow the
 69 partial grid of expert i to go outside Ω_i . In other words, each expert's partial grid can overlap at the
 70 subdomain boundaries.

71 Now we can focus on the ideal value of R in the case of 3D inputs ($d' = 3$). In particular, we
 72 want R large enough such that any input x has neighbor vertices spread in 3D, and we say that
 73 $R = \sqrt{\delta^2 + (\delta/2)^2 + (\delta/2)^2}$ is the smallest of such values that work for any input x . Fig. 1
 74 illustrates it.