Supplementary Material for: Scalable Gaussian Process Classification via Expectation Propagation

Daniel Hernández-Lobato
Universidad Autónoma de Madrid
Francisco Tomás y Valiente 11
28049, Madrid, Spain
daniel.hernandez@uam.com

José Miguel Hernández-Lobato
Harvard University
33 Oxford street
Cambridge, MA 02138, USA
jmhl@seas.harvard.edu

1 Introduction

In this document we give all the necessary details to implement the EP algorithm for the proposed method described in the main manuscript, *i.e.* SEP. In particular, we describe how to compute the EP posterior approximation from the product of all approximate factors and how to implement the EP updates to refine each approximate factor. We also give an intuitive idea about how to compute the EP approximation to the marginal likelihood and its gradients. Note that the updates described are very similar to the ones in [3].

2 Reconstruction of the posterior approximation

In this section we show how to obtain the posterior approximation as the normalized product of the approximate factors $\tilde{\phi}_i(\overline{\mathbf{f}})$ and the prior $p(\overline{\mathbf{f}}|\overline{\mathbf{X}})$. From the main manuscript, we know that these factors have the following form:

$$\tilde{\phi}_i(\overline{\mathbf{f}}) = \tilde{s}_i \exp\left\{ -\frac{\tilde{\nu}_i}{2} \overline{\mathbf{f}}^{\mathrm{T}} \boldsymbol{\upsilon}_i \boldsymbol{\upsilon}_i^{\mathrm{T}} \overline{\mathbf{f}} + \tilde{\mu}_i \overline{\mathbf{f}}^{\mathrm{T}} \boldsymbol{\upsilon}_i \right\}, \tag{1}$$

$$p(\overline{\mathbf{f}}|\overline{\mathbf{X}}) = \mathcal{N}(\overline{\mathbf{f}}|\mathbf{0}, \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}), \tag{2}$$

where $\boldsymbol{\upsilon}_i = \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{K}_{\overline{\mathbf{f}} f_i}$ and $\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}$ is a covariance matrix of size $m \times m$ with the prior covariance among the values associated to the inducing points $\overline{\mathbf{X}}$. Both the approximate factors and the prior are Gaussian, a family of distributions that is closed under product and division. The consequence is that $q(\overline{\mathbf{f}}) = \prod_{i=1}^{n} \tilde{\phi}_i(\overline{\mathbf{f}}) p(\overline{\mathbf{f}}|\overline{\mathbf{X}})/Z_q$ is also Gaussian. In particular, $q(\mathbf{f}) = \mathcal{N}(\overline{\mathbf{f}}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$. To obtain the parameters of q we can use the formulas given in the Appendix of [1]. This gives,

$$\boldsymbol{\Sigma} = \left(\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} + \boldsymbol{\Upsilon} \boldsymbol{\Delta} \boldsymbol{\Upsilon}^{\mathrm{T}} \right)^{-1}, \tag{3}$$

$$\boldsymbol{\mu} = \boldsymbol{\Sigma} \boldsymbol{\Upsilon} \tilde{\boldsymbol{\mu}} \tag{4}$$

where $\boldsymbol{\Delta}$ is a diagonal matrix with diagonal entries equal to $\tilde{\nu}_i$, $\boldsymbol{\Upsilon}$ is a matrix whose i-th column is equal to $\boldsymbol{\upsilon}_i$, and $\tilde{\boldsymbol{\mu}}$ is a vector whose i-th component is equal to $\tilde{\mu}_i$. These computations have a cost $\mathcal{O}(nm^2)$, under the assumption that $m \ll n$. Otherwise the cost is $\mathcal{O}(m^3)$.

3 Computation of the cavity distribution

Before the update of each $\tilde{\phi}_i$, the first step is to compute the cavity distribution $q^{\setminus i} \propto q/\tilde{\phi}_i$. Because q and $\tilde{\phi}_i$ are Gaussians, so it is $q^{\setminus i}$. In particular, $q^{\setminus i}(\overline{\mathbf{f}}) = \mathcal{N}(\overline{\mathbf{f}}|\boldsymbol{\mu}^{\setminus i}, \boldsymbol{\Sigma}^{\setminus i})$. The parameters of $q^{\setminus i}$ can also be obtained using the formulas given in the Appendix of [1]. That is,

$$\boldsymbol{\Sigma}^{\setminus i} = \left(\boldsymbol{\Sigma}^{-1} - \tilde{\nu}_i \boldsymbol{\upsilon}_i \boldsymbol{\upsilon}_i^{\mathrm{T}} \right)^{-1} = \boldsymbol{\Sigma} + (\tilde{\nu}_i^{-1} - \boldsymbol{\upsilon}_i^{\mathrm{T}} \boldsymbol{\Sigma} \boldsymbol{\upsilon}_i)^{-1} \boldsymbol{\Sigma} \boldsymbol{\upsilon}_i \boldsymbol{\upsilon}_i^{\mathrm{T}} \boldsymbol{\Sigma}, \tag{5}$$

$$\boldsymbol{\mu}^{\setminus i} = \boldsymbol{\Sigma}^{\setminus i} \left(\boldsymbol{\Sigma}^{-1} \boldsymbol{\mu} - \tilde{\mu}_i \boldsymbol{\upsilon}_i \right) = \boldsymbol{\mu} + \boldsymbol{\Sigma}^{\setminus i} \boldsymbol{\upsilon}_i \left(\tilde{\nu}_i \boldsymbol{\upsilon}_i^{\mathrm{T}} \boldsymbol{\mu} - \tilde{\mu}_i \right), \tag{6}$$

where we have used the Woodbury matrix identity and that $\mathbf{\Sigma}^{-1} = (\mathbf{\Sigma}^{\setminus i})^{-1} + \tilde{\nu}_i \boldsymbol{v}_i \boldsymbol{v}_i^{\mathrm{T}}$. These computations have a cost that is $\mathcal{O}(m^2)$.

4 Update of the approximate factors

In this section we show how to find the approximate factors $\tilde{\phi}_i$. For that we consider that the corresponding cavity distribution $q^{\setminus i}$ has already been computed. From the main manuscript, we know that the exact factor to be approximated is:

$$\phi_i(\overline{\mathbf{f}}) = \int \Phi(y_i f_i) \mathcal{N}(f_i | m_i, s_i) df_i = \Phi\left(\frac{y_i m_i}{\sqrt{s_i + 1}}\right), \tag{7}$$

where $\Phi(\cdot)$ is the c.d.f. of a standard Gaussian, $m_i = \mathbf{K}_{f_i \overline{\mathbf{f}}} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \overline{\mathbf{f}}$ and $s_i = \mathbf{K}_{f_i f_i} - \mathbf{K}_{f_i \overline{\mathbf{f}}} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{K}_{\overline{\mathbf{f}} f_i}$. We compute Z_i, *i.e.*, the normalization constant of $\phi_i q^{\setminus i}$, as follows:

$$Z_i = \int \Phi\left(\frac{y_i m_i}{\sqrt{s_i + 1}}\right) \mathcal{N}(\overline{\mathbf{f}} | \boldsymbol{\mu}^{\setminus i}, \mathbf{\Sigma}^{\setminus i}) d\overline{\mathbf{f}} = \Phi\left(\frac{y_i a_i}{\sqrt{b_i}}\right), \tag{8}$$

where $a_i = \mathbf{K}_{f_i \overline{\mathbf{f}}} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \boldsymbol{\mu}^{\setminus i}$ and $b_i = 1 + \mathbf{K}_{f_i f_i} - \mathbf{K}_{f_i \overline{\mathbf{f}}} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{K}_{\overline{\mathbf{f}} f_i} + \mathbf{K}_{f_i \overline{\mathbf{f}}} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{\Sigma}^{\setminus i} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{K}_{\overline{\mathbf{f}} f_i}$. By using the equations given in the Appendix of [1] it is possible to obtain the moments, *i.e.*, the mean $\hat{\boldsymbol{\mu}}$ and the covariances $\hat{\mathbf{\Sigma}}$ of $\phi_i q^{\setminus i}$, from the derivatives of $\log Z_i$ with respect to the parameters of $q^{\setminus i}$. Namely,

$$\hat{\mathbf{m}} = \boldsymbol{\mu}^{\setminus i} + \mathbf{\Sigma}^{\setminus i} \frac{\partial \log Z_i}{\partial \boldsymbol{\mu}^{\setminus i}} = \boldsymbol{\mu}^{\setminus i} + \alpha_i \mathbf{\Sigma}^{\setminus i} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{K}_{\overline{\mathbf{f}} f_i}, \tag{9}$$

$$\begin{aligned} \hat{\mathbf{\Sigma}} &= \mathbf{\Sigma}^{\setminus i} - \mathbf{\Sigma}^{\setminus i} \left(\left(\frac{\partial \log Z_i}{\partial \boldsymbol{\mu}^{\setminus i}} \right) \left(\frac{\partial \log Z_i}{\partial \boldsymbol{\mu}^{\setminus i}} \right)^{\mathrm{T}} - 2 \frac{\partial \log Z_i}{\partial \mathbf{\Sigma}^{\setminus i}} \right) \mathbf{\Sigma}^{\setminus i} \\ &= \mathbf{\Sigma}^{\setminus i} - \mathbf{\Sigma}^{\setminus i} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{K}_{\overline{\mathbf{f}} f_i} \mathbf{K}_{f_i \overline{\mathbf{f}}} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{\Sigma}^{\setminus i} \left(\alpha_i^2 + \frac{\alpha_i a_i}{b_i} \right), \end{aligned} \tag{10}$$

where

$$\alpha_i = \frac{\mathcal{N}(y_i a_i / \sqrt{b_i} | 0, 1)}{\Phi(y_i a_i / \sqrt{b_i})} \frac{y_i}{b_i}. \tag{11}$$

These are very similar to the EP updates described in [3].

Given the previous updates, it is possible to find the parameters of the corresponding approximate factor $\tilde{\phi}_i$, which is simply obtained as $\tilde{\phi}_i = Z_i q^{\mathrm{new}} / q^{\setminus i}$, where q^{new} is a Gaussian distribution with the mean and the covariances of $\phi_i q^{\setminus i}$. We show here that the precision matrix of the approximate factor $\tilde{\phi}_i$ has a low rank form. Denote with $\tilde{\mathbf{V}}_i$ to such matrix. Let also $\tilde{\mathbf{m}}_i$ be the precision matrix of $\tilde{\phi}_i$ times the mean vector. Define $\boldsymbol{v}_i = \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \mathbf{K}_{\overline{\mathbf{f}} f_i}$. Then, by using the equations given in the Appendix of [1] we have that

$$\tilde{\mathbf{V}}_i = \hat{\mathbf{\Sigma}}^{-1} - \left(\mathbf{\Sigma}^{\setminus i}\right)^{-1} = \left(\mathbf{\Sigma}^{\setminus i}\right)^{-1} + \boldsymbol{v}_i \boldsymbol{v}_i^{\mathrm{T}} \tilde{\nu}_i - \left(\mathbf{\Sigma}^{\setminus i}\right)^{-1} = \boldsymbol{v}_i \boldsymbol{v}_i^{\mathrm{T}} \tilde{\nu}_i \tag{12}$$

$$\tilde{\mathbf{m}}_i = \hat{\mathbf{\Sigma}}^{-1} \hat{\mathbf{m}} - \left(\mathbf{\Sigma}^{\setminus i}\right)^{-1} \boldsymbol{\mu}^{\setminus i} = \left(\alpha_i + a_i \tilde{\nu}_i + \alpha_i \boldsymbol{v}_i^{\mathrm{T}} \mathbf{\Sigma}^{\setminus i} \boldsymbol{v}_i \tilde{\nu}_i\right) \boldsymbol{v}_i = \tilde{\mu}_i \boldsymbol{v}_i \tag{13}$$

where we have used the Woodbury matrix identity, the definition of $\hat{\mathbf{m}}$ and $\hat{\mathbf{\Sigma}}$, and

$$\tilde{\nu}_i = \left[\left(\alpha_i^2 + \frac{\alpha_i a_i}{b_i} \right)^{-1} + \boldsymbol{v}_i^{\mathrm{T}} \mathbf{\Sigma}^{\setminus i} \boldsymbol{v}_i \right]^{-1} \qquad \tilde{\mu}_i = \alpha_i + a_i \tilde{\nu}_i + \alpha_i \boldsymbol{v}_i^{\mathrm{T}} \mathbf{\Sigma}^{\setminus i} \boldsymbol{v}_i \tilde{\nu}_i. \tag{14}$$

Thus, we see that the approximate factor has the form described in (1).

Once we have the parameters of the approximate factor $\tilde{\phi}_i$, we can compute the value of $\tilde{s}_i$ in (1) which guarantees that the approximate factor integrates the same as the exact factor with respect to $q^{\setminus i}$. Let $\boldsymbol{\theta}$ be the natural parameters of q after the update. Similarly, let $\boldsymbol{\theta}^{\setminus i}$ be the natural parameters of $q^{\setminus i}$. Then,

$$\tilde{s}_i = \log Z_i + g(\boldsymbol{\theta}^{\setminus i}) - g(\boldsymbol{\theta}), \tag{15}$$

where $g(\boldsymbol{\theta})$ is the log-normalizer of a multi-variate Gaussian with natural parameters $\boldsymbol{\theta}$).

5 Parallel EP updates and damping

The updates described for the approximate factors are done in parallel. That is, we compute the required quantities to update each factor $\tilde{\phi}_i$ at the same time using (14). Then, the new parameters of each approximate factor $\tilde{\nu}_i$ and $\tilde{\mu}_i$ are computed based on the previous ones. Finally, after the parallel update, we recompute q as indicated in Section 2. All these operations have a closed-form and involve only matrix multiplications with cost $\mathcal{O}(nm^2)$, where n is the number of samples and m is the number of inducing points.

Parallel EP updates were first proposed in [7] and have been also used in the context of Gaussian process classification in [2]. Parallel EP updates are much faster than sequential updates because they avoid having to code loops over the training instances. All operations simply involve matrix multiplications which are significantly faster as a consequence of using the BLAS library (available in most scientific programming languages such as R, matlab or Python) that has been significantly optimized.

Parallel updates may deteriorate EP convergence in some situations. Thus, we also use damped EP updates. Damping is a standard approach in EP algorithms which significantly improves convergence. The idea is to avoid large changes in the parameters $\tilde{\nu}_i$ and $\tilde{\mu}_i$ of the approximate factors $\tilde{\phi}_i$. For this, the parameters after the EP updates are set to be a linear combination of the old and the new parameters. In particular,

$$\tilde{\nu}_i = \rho\tilde{\nu}_i^{\text{new}} + (1-\rho)\tilde{\nu}_i^{\text{old}}\,, \qquad\qquad \tilde{\mu}_i = \rho\tilde{\mu}_i^{\text{new}} + (1-\rho)\tilde{\mu}_i^{\text{old}}\,, \tag{16}$$

where $\rho \in [0,1]$ is a parameter controlling the amount of damping. If $\rho = 1$ there is no damping and if $\rho = 0$ the parameters of each $\tilde{\phi}_i$ are not updated at all. In our experiments we set $\rho = 0.5$ when doing batch training and we set $\rho = 0.99$ when the training process is done in a stochastic fashion using minibatches (in this case we do more frequent reconstructions of q, *i.e.*, after processing each minibatch and less damping is needed). Damping does not change the fixed points of EP.

6 Estimate of the marginal likelihood

As indicated in the main manuscript, the estimate of the marginal likelihood is given by

$$\log Z_q = g(\boldsymbol{\theta}) - g(\boldsymbol{\theta}_{\text{prior}}) + \sum_{i=1}^{n} \log \tilde{Z}_i \qquad\qquad \log \tilde{Z}_i = \log Z_i + g(\boldsymbol{\theta}^{\setminus i}) - g(\boldsymbol{\theta})\,, \tag{17}$$

where $\boldsymbol{\theta}$, $\boldsymbol{\theta}^{\setminus i}$ and $\boldsymbol{\theta}_{\text{prior}}$ are the natural parameters of q, $q^{\setminus i}$ and $p(\overline{\mathbf{f}}|\overline{\mathbf{X}})$, respectively; and $g(\boldsymbol{\theta})$ is the log-normalizer of a multivariate Gaussian distribution with natural parameters $\boldsymbol{\theta}$. Let $\boldsymbol{m}$ and $\boldsymbol{S}$ be the variance and the mean, respectively, of a Gaussian distribution over m dimensions with natural parameters $\boldsymbol{\theta}'$. Then,

$$g(\boldsymbol{\theta}') = \frac{m}{2}\log 2\pi + \frac{1}{2}\log|\boldsymbol{S}| + \frac{1}{2}\boldsymbol{m}^{\text{T}}\boldsymbol{S}^{-1}\boldsymbol{m}\,. \tag{18}$$

The consequence is that

$$\log Z_q = \frac{1}{2}\log|\boldsymbol{\Sigma}| + \frac{1}{2}\boldsymbol{\mu}^{\text{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \frac{1}{2}\log|\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}| + \sum_{i=1}^{n}\log\tilde{Z}_i\,, \tag{19}$$

with

$$\begin{aligned}
\tilde{Z}_i &= \log Z_i + \frac{1}{2}\log|\boldsymbol{\Sigma}^{\setminus i}| + \frac{1}{2}(\boldsymbol{\mu}^{\setminus i})^{\text{T}}\left(\boldsymbol{\Sigma}^{\setminus i}\right)^{-1}\boldsymbol{\mu}^{\setminus i} - \frac{1}{2}\log|\boldsymbol{\Sigma}| - \frac{1}{2}\boldsymbol{\mu}^{\text{T}}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} \\
&= \log Z_i - 2\tilde{\mu}_i\boldsymbol{v}_i^{\text{T}}\boldsymbol{\mu} + \tilde{\mu}_i^2\boldsymbol{v}_i^{\text{T}}\boldsymbol{\Sigma}\boldsymbol{v}_i + \left(\boldsymbol{\mu}^{\text{T}}\boldsymbol{v}_i\right)^2 C_i - 2\boldsymbol{\mu}^{\text{T}}\boldsymbol{v}_i\boldsymbol{v}_i^{\text{T}}\boldsymbol{\Sigma}\boldsymbol{v}_i\tilde{\mu}_i C_i \\
&\quad + \tilde{\mu}_i^2 C_i\left(\boldsymbol{v}_i^{\text{T}}\boldsymbol{\Sigma}\boldsymbol{v}_i\right)^2 + \frac{1}{2}\log(1 - \tilde{v}_i\boldsymbol{v}_i\boldsymbol{\Sigma}\boldsymbol{v}_i)\,,
\end{aligned} \tag{20}$$

where we have used that $\left(\boldsymbol{\Sigma}^{\setminus i}\right)^{-1} = \boldsymbol{\Sigma}^{-1} - \tilde{\nu}_i\boldsymbol{v}_i\boldsymbol{v}_i^{\text{T}}$, the Woodbury matrix identity, the matrix determinant lemma, that $\boldsymbol{\mu}^{\setminus i} = \boldsymbol{\Sigma}^{\setminus i}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu} - \tilde{\mu}_i\boldsymbol{v}_i)$, and set $C_i = (\tilde{\nu}_i^{-1} - \boldsymbol{v}_i\boldsymbol{\Sigma}\boldsymbol{v}_i)^{-1}$. The consequence is that the computation of $\log Z_q$ can be done with cost $\mathcal{O}(nm^2)$ if $m \ll n$.

7 Gradient of $\log Z_q$ after convergence

In this section we show that the gradient of $\log Z_q$, after convergence, is given by the expression given in the main manuscript. For that, we extend the results of [5]. Denote by ξ_j to one hyper-parameter of the model. That is, a parameter of the covariance function k or a component of the inducing points. Then, the gradient of $\log Z_q$ with respect to this parameter is:

$$\frac{\partial \log Z_q}{\partial \xi_j} = \left(\frac{\partial g(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} - \left(\frac{\partial g(\boldsymbol{\theta}_{\text{prior}})}{\partial \boldsymbol{\theta}_{\text{prior}}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j} + \sum_{i=1}^{n} \left(\frac{\partial g(\boldsymbol{\theta}^{\setminus i})}{\partial \boldsymbol{\theta}^{\setminus i}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} - \sum_{i=1}^{n} \left(\frac{\partial g(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j}, \tag{21}$$

where $\boldsymbol{\theta}$, $\boldsymbol{\theta}^{\setminus i}$ and $\boldsymbol{\theta}_{\text{prior}}$ are the natural parameters of q, $q^{\setminus i}$, and the prior $p(\bar{\mathbf{f}}|\overline{\mathbf{X}})$, respectively. Importantly, the term $\log Z_i$ depends on ξ_j in a direct way, *i.e.*, because the exact likelihood factor $\phi_i(\bar{\mathbf{f}}) = \int \Phi(y_i f_i)\mathcal{N}(f_i|m_i, s_i)df_i = \Phi(y_i m_i/\sqrt{s_i+1})$, with $m_i = \mathbf{K}_{f_i\bar{\mathbf{f}}}\mathbf{K}_{\bar{\mathbf{f}}\bar{\mathbf{f}}}^{-1}\bar{\mathbf{f}}$ and $s_i = \mathbf{K}_{f_i f_i} - \mathbf{K}_{f_i\bar{\mathbf{f}}}\mathbf{K}_{\bar{\mathbf{f}}\bar{\mathbf{f}}}^{-1}\mathbf{K}_{\bar{\mathbf{f}}f_i}$, depends on ξ_j, and in an indirect way, *i.e.*, because the natural parameters of the cavity distribution $q^{\setminus i}$, $\boldsymbol{\theta}^{\setminus i}$, depend on ξ_j. In particular,

$$Z_i = \int \phi_i(\bar{\mathbf{f}}) \exp\left\{\left(\boldsymbol{\theta}^{\setminus i}\right)^{\mathrm{T}} h(\bar{\mathbf{f}}) - g(\boldsymbol{\theta}^{\setminus i})\right\} d\bar{\mathbf{f}}, \tag{22}$$

where $h(\bar{\mathbf{f}})$ are the sufficient statistics of $q^{\setminus i}$. The consequence is that

$$\begin{aligned}\frac{\partial \log Z_i}{\partial \xi_j} &= \overbrace{\frac{\partial \log Z_i}{\partial \xi}}^{\text{Only } \phi_i(\bar{\mathbf{f}}) \text{ changes}} + \left(\frac{\partial \log Z_i}{\partial \boldsymbol{\theta}^{\setminus i}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} \\ &= \overbrace{\frac{\partial \log Z_i}{\partial \xi_j}}^{\text{Only } \phi_i(\bar{\mathbf{f}}) \text{ changes}} + \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} - \left(\boldsymbol{\eta}^{\setminus i}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j},\end{aligned} \tag{23}$$

where $\boldsymbol{\eta}$ and $\boldsymbol{\eta}^{\setminus i}$ are the expected sufficient statistics under the posterior approximation q and the cavity distribution $q^{\setminus i}$. Recall that we have assumed convergence which leads to a match of the moments between $Z_i^{-1}\phi_i q^{\setminus i}$ and q.

If we substitute (23) in (21) we have that:

$$\begin{aligned}\frac{\partial \log Z_i}{\partial \xi_j} &= \left(\frac{\partial g(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} - \left(\frac{\partial g(\boldsymbol{\theta}_{\text{prior}})}{\partial \boldsymbol{\theta}_{\text{prior}}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j} + \sum_{i=1}^{n} \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} \\ &\quad - \sum_{i=1}^{n} \left(\boldsymbol{\eta}^{\setminus i}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} + \sum_{i=1}^{n} \left(\frac{\partial g(\boldsymbol{\theta}^{\setminus i})}{\partial \boldsymbol{\theta}^{\setminus i}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} - \sum_{i=1}^{n} \left(\frac{\partial g(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} \\ &= \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} - (\boldsymbol{\eta}_{\text{prior}})^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j} + \sum_{i=1}^{n} \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} \\ &\quad - \sum_{i=1}^{n} \boldsymbol{\eta}^{\setminus i} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi} + \sum_{i=1}^{n} \left(\boldsymbol{\eta}^{\setminus i}\right)^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} - \sum_{i=1}^{n} \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} \\ &= \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} - (\boldsymbol{\eta}_{\text{prior}})^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j} + \sum_{i=1}^{n} \boldsymbol{\eta}^{\mathrm{T}} \left(\frac{\partial \boldsymbol{\theta}^{\setminus i}}{\partial \xi_j} - \frac{\partial \boldsymbol{\theta}}{\partial \xi_j}\right) \\ &= \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} - (\boldsymbol{\eta}_{\text{prior}})^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j} - \sum_{i=1}^{n} \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_i}{\partial \xi_j} \\ &= \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}}{\partial \xi_j} - (\boldsymbol{\eta}_{\text{prior}})^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j} - \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}^{\setminus \text{prior}}}{\partial \xi_j} \\ &= \boldsymbol{\eta}^{\mathrm{T}} \left(\frac{\partial \boldsymbol{\theta}}{\partial \xi_j} - \frac{\partial \boldsymbol{\theta}^{\setminus \text{prior}}}{\partial \xi_j}\right) - (\boldsymbol{\eta}_{\text{prior}})^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j} \\ &= \boldsymbol{\eta}^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} - (\boldsymbol{\eta}_{\text{prior}})^{\mathrm{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} + \sum_{i=1}^{n} \frac{\partial \log Z_i}{\partial \xi_j},\end{aligned} \tag{24}$$

where $\boldsymbol{\eta}_{\text{prior}}$ are the expected sufficient statistics of the prior and we have used that $\boldsymbol{\theta} = \boldsymbol{\theta}_{\text{prior}} + \sum_{i=1}^{n} \boldsymbol{\theta}_i$, with $\boldsymbol{\theta}_i$ the natural parameters of the approximate factor $\tilde{\phi}_i$, and that $\boldsymbol{\theta}^{\setminus \text{prior}} = \sum_{i=1}^{n} \boldsymbol{\theta}_i$. Thus, at convergence the approximate factors can be considered to be fixed. In particular, (24) is the gradient obtained under the assumption that all $\tilde{\phi}_i$ remain fixed and do not change with the model hyper-parameters.

The chain rule of derivatives has to be taken with care in the previous expression. Since the natural parameters and the expected sufficient statistics are often expressed in the form of matrices, the chain rule for matrix derivatives has to be employed in practice (see [4, Sec. 2.8.1]). The consequence is that

$$\boldsymbol{\eta}^{\text{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} - \left(\boldsymbol{\eta}_{\text{prior}}\right)^{\text{T}} \frac{\partial \boldsymbol{\theta}_{\text{prior}}}{\partial \xi_j} = -0.5 \text{tr} \left(\mathbf{M}^{\text{T}} \frac{\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}}{\partial \xi_j} \right) , \tag{25}$$

where

$$\mathbf{M} = \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} - \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \boldsymbol{\Sigma} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} - \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} \boldsymbol{\mu}\boldsymbol{\mu}^{\text{T}} \mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1} . \tag{26}$$

In the case of computing the derivatives with respect to the inducing points several contractions occur, as indicated in [6]. The computational cost of obtaining these derivatives is $\mathcal{O}(m^3)$.

The derivatives with respect to each $\log Z_i$ can be computed also efficiently using the chain rule for matrix derivatives indicated in [4, Sec. 2.8.1]. The computational cost of obtaining these derivatives is $\mathcal{O}(nm^2)$. Furthermore, several standard properties of the trace can be employed to simplify the computations. In particular, the trace is invariant to cyclic rotations. Namely, $\text{tr}(\mathbf{ABCD}) = \text{tr}(\mathbf{DABC})$.

By using the gradients described, it is possible to maximize $\log Z_q$ to find good values for the model hyper-parameters. However, as stated in the main manuscript, we do not wait until EP converges for doing the update. In particular, we perform an update of the hyper-parameters considering the $\tilde{\phi}_i$ as fixed, after each parallel refinement of the approximate factors. Because we are updating the approximate factors too, we cannot simply expect that such steps always improve on the objective $\log Z_q$, but in practice they seem to work very well. In our experiments we use an adaptive learning rate that is different for each hyper-parameter. In particular, we increase the learning rate by 2% if the sign of the estimate of the gradient for that hyper-parameter does not change between two consecutive iterations. If a change is observed, we reduce we multiply the learning rate by 1/2. If an stochastic approximation of the estimate of the gradient is employed, we use the ADADELTA method to estimate the learning rate [8].

8 Predictive distribution

Once the training process is complete, we can use the posterior approximation q for making predictions about the class label $y_\star \in \{-1, 1\}$ of a new instance $\mathbf{x}_\star$. In that case, we compute first an approximate posterior for the Gaussian process evaluated at the target location, *i.e.*, $f(\mathbf{x}_\star)$, which is summarized as $f_\star$:

$$\begin{aligned} p(f_\star|\mathbf{y}, \overline{\mathbf{X}}) &\approx \int p(f_\star|\overline{\mathbf{f}}) q(\overline{\mathbf{f}}) d\overline{\mathbf{f}} \\ &\approx \int \mathcal{N}(f_\star|\mathbf{K}_{f_\star\overline{\mathbf{f}}}\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1}\overline{\mathbf{f}}, \mathbf{K}_{f_\star f_\star} - \mathbf{K}_{f_\star\overline{\mathbf{f}}}\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1}\mathbf{K}_{\overline{\mathbf{f}}f_\star}) \mathcal{N}(\overline{\mathbf{f}}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) d\overline{\mathbf{f}} \\ &\approx \mathcal{N}(f_\star|m_\star, s_\star) , \end{aligned} \tag{27}$$

where $m_\star = \mathbf{K}_{f_\star\overline{\mathbf{f}}}\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1}\boldsymbol{\mu}$ and $s_\star = \mathbf{K}_{f_\star f_\star} - \mathbf{K}_{f_\star\overline{\mathbf{f}}}\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1}\mathbf{K}_{\overline{\mathbf{f}}f_\star} + \mathbf{K}_{f_\star\overline{\mathbf{f}}}\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1}\boldsymbol{\Sigma}\mathbf{K}_{\overline{\mathbf{f}}\overline{\mathbf{f}}}^{-1}\mathbf{K}_{\overline{\mathbf{f}}f_\star}$. $\mathbf{K}_{f_\star f_\star}$ and $\mathbf{K}_{f_\star\overline{\mathbf{f}}}$ contain the prior variance of $f_\star$ and the prior covariances between $f_\star$ and $\overline{\mathbf{f}}$, respectively. The approximate predictive distribution for the class label $y_\star$ is simply:

$$p(y_\star|\mathbf{y}, \overline{\mathbf{X}}) = \int p(y_\star|f_\star) p(f_\star|\mathbf{y}, \overline{\mathbf{X}}) df_\star = \int \Phi(y_\star f_\star) \mathcal{N}(f_\star|m_\star, s_\star) df_\star = \Phi\left(\frac{y_\star m_\star}{\sqrt{s_\star + 1}} \right) , \tag{28}$$

where $\Phi(\cdot)$ is the c.d.f of a standard Gaussian distribution.

References

[1] D. Hernández-Lobato. *Prediction Based on Averages over Automatically Induced Learners: Ensemble Methods and Bayesian Techniques.* PhD thesis, Universidad Autónoma de Madrid, 2009.

[2] D. Hernández-Lobato, J. M. Hernández-Lobato, and Pierre Dupont. Robust multi-class Gaussian process classification. In *Advances in Neural Information Processing Systems 24*, 2011.

[3] T. Minka. *A Family of Algorithms for Approximate Bayesian Inference*. PhD thesis, MIT, 2001.

[4] K. B. Petersen and M. S. Pedersen. The matrix cookbook, 2012. Version 20121115.

[5] M. Seeger. Expectation propagation for exponential families. Technical report, Department of EECS, University of California, Berkeley, 2006.

[6] E. Snelson. *Flexible and efficient Gaussian process models for machine learning*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London, 2007.

[7] M. Van Gerven, B. Cseke, R. Oostenveld, and T. Heskes. Bayesian source localization with the multivariate Laplace prior. In *Advances in Neural Information Processing Systems 22*, pages 1901–1909, 2009.

[8] M.D. Zeiler. ADADELTA: An adaptive learning rate method. *ArXiv e-prints*, 2012. arXiv:1212.5701.