Gray-box Inference for Structured Gaussian Process Models: *Supplementary Material*

Pietro Galliani, Amir Dezfouli, Edwin V. Bonilla and Novi Quadrianto

February 28, 2017

1 Proof of Theorem 2

Here we proof the result that we can estimate the expected log likelihood and its gradients using expectations over low-dimensional Gaussians, that is that

Theorem. *For the structured* GP *model defined in our paper the expected log likelihood over the given variational distribution and its gradients can be estimated using expectations over T_n-dimensional Gaussians and $|\mathcal{V}|^2$-dimensional Gaussians, where T_n is the length of each sequence and $|\mathcal{V}|$ is the vocabulary size.*

1.1 Estimation of $\mathcal{L}_{\text{ell}}$ in the full (non-sparse) model

For the $\mathcal{L}_{\text{ell}}$ we have that:

$$\mathcal{L}_{\text{ell}} = \left\langle \sum_{n=1}^{N_{\text{seq}}} \log p(\mathbf{y}_n|\mathbf{f}_n) \right\rangle_{q(\mathbf{f}^{\text{un}})q(\mathbf{f}^{\text{bin}})} \tag{1}$$

$$= \sum_{n=1}^{N_{\text{seq}}} \int_{\mathbf{f}^{\text{bin}}} \int_{\mathbf{f}^{\text{un}}} q(\mathbf{f}^{\text{un}})q(\mathbf{f}^{\text{bin}}) \log p(\mathbf{y}_n|\mathbf{f}_n)\ d\mathbf{f}^{\text{un}} d\mathbf{f}^{\text{bin}} \tag{2}$$

$$= \sum_{n=1}^{N_{\text{seq}}} \int_{\mathbf{f}^{\text{bin}}} \int_{\mathbf{f}^{\text{un}}_n} \int_{\mathbf{f}^{\text{un}}_{\setminus n}} q(\mathbf{f}^{\text{un}}_{\setminus n}|\mathbf{f}^{\text{un}}_n)q(\mathbf{f}^{\text{un}}_n)q(\mathbf{f}^{\text{bin}}) \log p(\mathbf{y}_n|\mathbf{f}_n)\ d\mathbf{f}^{\text{un}}_{\setminus n} d\mathbf{f}^{\text{un}}_n d\mathbf{f}^{\text{bin}} \tag{3}$$

$$= \sum_{n=1}^{N_{\text{seq}}} \left\langle \log p(\mathbf{y}_n|\mathbf{f}_n) \right\rangle_{q(\mathbf{f}^{\text{un}}_n)q(\mathbf{f}^{\text{bin}})} \tag{4}$$

$$= \sum_{n=1}^{N_{\text{seq}}} \sum_{k=1}^{K} \pi_k \left\langle \log p(\mathbf{y}_n|\mathbf{f}_n) \right\rangle_{q_{kn}(\mathbf{f}^{\text{un}}_n)q(\mathbf{f}^{\text{bin}})}, \tag{5}$$

where $q_{kn}(\mathbf{f}^{\text{un}}_n)$ is a $(T_n \times |\mathcal{V}|)$-dimensional Gaussian with block-diagonal covariance $\mathbf{\Sigma}_{k(n)}$, each block of size $T_n \times T_n$. Therefore, we can estimate the above term by sampling from T_n-dimensional Gaussians independently. Furthermore, $q(\mathbf{f}^{\text{bin}})$ is a $|\mathcal{V}|^2$-dimensional Gaussian, which can also be sampled independently. In practice, we can assume that the covariance of $q(\mathbf{f}^{\text{bin}})$ is diagonal and we only sample from unary Gaussians for the pairwise functions. ■

1.2 Gradients

Taking the gradients of the kth term for the nth sequence in the $\mathcal{L}_{\text{ell}}$:

$$\begin{aligned}
\mathcal{L}_{\text{ell}}^{(k,n)} &= \langle \log p(\mathbf{y}_n|\mathbf{f}_n) \rangle_{q_{kn}(\mathbf{f}_n^{\text{un}})q(\mathbf{f}^{\text{bin}})} && (6)\\
&= \int_{\mathbf{f}^{\text{bin}}} \int_{\mathbf{f}_n^{\text{un}}} q_{kn}(\mathbf{f}_n^{\text{un}}) q(\mathbf{f}^{\text{bin}}) \log p(\mathbf{y}_n|\mathbf{f}_n) \; \mathrm{d}\mathbf{f}_n^{\text{un}} \mathrm{d}\mathbf{f}^{\text{bin}} && (7)\\
\nabla_{\boldsymbol{\lambda}_k^{\text{un}}} \mathcal{L}_{\text{ell}}^{(k,n)} &= \int_{\mathbf{f}^{\text{bin}}} \int_{\mathbf{f}_n^{\text{un}}} q_{kn}(\mathbf{f}_n^{\text{un}}) q(\mathbf{f}^{\text{bin}}) \log p(\mathbf{y}_n|\mathbf{f}_n) \nabla_{\boldsymbol{\lambda}_k^{\text{un}}} \log q_{kn}(\mathbf{f}_n^{\text{un}}) \; \mathrm{d}\mathbf{f}_n^{\text{un}} \mathrm{d}\mathbf{f}^{\text{bin}} && (8)\\
&= \left\langle \log p(\mathbf{y}_n|\mathbf{f}_n) \nabla_{\boldsymbol{\lambda}_k^{\text{un}}} \log q_{kn}(\mathbf{f}_n^{\text{un}}) \right\rangle_{q_{kn}(\mathbf{f}_n^{\text{un}})q(\mathbf{f}^{\text{bin}})}, && (9)
\end{aligned}$$

where we have used the fact that $\nabla_{\mathbf{x}} f(\mathbf{x}) = f(\mathbf{x}) \nabla_{\mathbf{x}} \log f(\mathbf{x})$ for any nonnegative function $f(\mathbf{x})$ Similarly. the gradients of the parameters of the distribution over binary functions can be estimated using:

$$\nabla_{\boldsymbol{\lambda}^{\text{bin}}} \mathcal{L}_{\text{ell}}^{(k,n)} = \left\langle \log p(\mathbf{y}_n|\mathbf{f}_n) \nabla_{\boldsymbol{\lambda}^{\text{bin}}} \log q(\mathbf{f}^{\text{bin}}) \right\rangle_{q_{kn}(\mathbf{f}_n^{\text{un}})q(\mathbf{f}^{\text{bin}})}. \qquad (10)$$

■

2 KL terms in the sparse model

The KL term ($\mathcal{L}_{\text{kl}}$) in the variational objective ($\mathcal{L}_{\text{elbo}}$) is composed of a KL divergence between the approximate posteriors and the priors over the inducing variables and pairwise functions:

$$\mathcal{L}_{\text{kl}} = \underbrace{-\text{KL}(q(\mathbf{u}) \| p(\mathbf{u}))}_{\mathcal{L}_{\text{kl}}^{\text{un}}} \underbrace{-\text{KL}(q(\mathbf{f}^{\text{bin}}) \| p(\mathbf{f}^{\text{bin}}))}_{\mathcal{L}_{\text{kl}}^{\text{bin}}}, \qquad (11)$$

where, as the approximate posterior and the prior over the pairwise functions are Gaussian, the KL over pairwise functions can be computed analytically:

$$\begin{aligned}
\mathcal{L}_{\text{kl}}^{\text{bin}} &= -\text{KL}(q(\mathbf{f}^{\text{bin}}) \| p(\mathbf{f}^{\text{bin}})) = \text{KL}(\mathcal{N}(\mathbf{f}^{\text{bin}}; \mathbf{m}^{\text{bin}}, \mathbf{S}^{\text{bin}}) \| \mathcal{N}(\mathbf{f}^{\text{bin}}; \mathbf{0}, \mathbf{K}^{\text{bin}})) && (12)\\
&= -\frac{1}{2} \left(\log \left|\mathbf{K}^{\text{bin}}\right| - \log \left|\mathbf{S}^{\text{bin}}\right| + (\mathbf{m}^{\text{bin}})^T (\mathbf{K}^{\text{bin}})^{-1} \mathbf{m}^{\text{bin}} + \text{ tr } (\mathbf{K}^{\text{bin}})^{-1} \mathbf{S}^{\text{bin}} - |\mathcal{V}| \right). && (13)
\end{aligned}$$

For the distributions over the unary functions we need to compute a KL divergence between a mixture of Gaussians and a Gaussian. For this we consider the decomposition of the KL divergence as follows:

$$\mathcal{L}_{\text{kl}}^{\text{un}} = -\text{KL}(q(\mathbf{u}) \| p(\mathbf{u})) = \underbrace{\mathbb{E}_q[-\log q(\mathbf{u})]}_{\mathcal{L}_{\text{ent}}} + \underbrace{\mathbb{E}_q[\log p(\mathbf{u})]}_{\mathcal{L}_{\text{cross}}}, \qquad (14)$$

where the entropy term ($\mathcal{L}_{\text{ent}}$) can be lower bounded using Jensen's inequality:

$$\mathcal{L}_{\text{ent}} \geq -\sum_{k=1}^{K} \pi_k \log \sum_{\ell=1}^{K} \pi_\ell \mathcal{N}(\mathbf{m}_k; \mathbf{m}_\ell, \mathbf{S}_k + \mathbf{S}_\ell) \overset{\text{def}}{=} \hat{\mathcal{L}}_{\text{ent}}. \qquad (15)$$

and the negative cross-entropy term ($\mathcal{L}_{\text{cross}}$) can be computed exactly:

$$\mathcal{L}_{\text{cross}} = -\frac{1}{2} \sum_{k=1}^{K} \pi_k \sum_{j=1}^{|\mathcal{V}|} [M \log 2\pi + \log |\kappa(\mathbf{Z}_j, \mathbf{Z}_j)| + \mathbf{m}_{kj}^T \kappa(\mathbf{Z}_j, \mathbf{Z}_j)^{-1} \mathbf{m}_{kj} + \text{ tr } \kappa(\mathbf{Z}_j, \mathbf{Z}_j)^{-1} \mathbf{S}_{kj}]. \qquad (16)$$

3 Proof of Theorem 3

To prove Theorem 3 we will express the expected log likelihood term in the same form as that given in Equation (5), showing that the resulting $q_{kn}(\mathbf{f}_n^{\text{un}})$ is also a $(T_n \times |\mathcal{V}|)$-dimensional Gaussian with block-diagonal covariance, having $|\mathcal{V}|$ blocks each of dimensions $T_n \times T_n$. We start by taking the given $\mathcal{L}_{\text{ell}}$, where the expectations are over the joint posterior $q(\mathbf{f}, \mathbf{u}|\boldsymbol{\lambda}) = p(\mathbf{f}^{\text{un}}|\mathbf{u})q(\mathbf{u})q(\mathbf{f}^{\text{bin}})$:

$$\mathcal{L}_{\text{ell}} = \left\langle \sum_{n=1}^{N_{\text{seq}}} \log p(\mathbf{y}_n|\mathbf{f}_n) \right\rangle_{p(\mathbf{f}^{\text{un}}|\mathbf{u})q(\mathbf{u})q(\mathbf{f}^{\text{bin}})} \tag{17}$$

$$= \int_{\mathbf{f}} \log p(\mathbf{y}|\mathbf{f}) \underbrace{\int_{\mathbf{u}} q(\mathbf{u})p(\mathbf{f}^{\text{un}}|\mathbf{u})\text{d}\mathbf{u}}_{q(\mathbf{f}^{\text{un}})} q(\mathbf{f}^{\text{bin}})\text{d}\mathbf{f}, \tag{18}$$

where our our approximating distribution is:

$$q(\mathbf{f}) = q(\mathbf{f}^{\text{un}})q(\mathbf{f}^{\text{bin}}) \tag{19}$$

$$q(\mathbf{f}^{\text{un}}) = \int_{\mathbf{u}} q(\mathbf{u})p(\mathbf{f}^{\text{un}}|\mathbf{u})\text{d}\mathbf{u}, \tag{20}$$

which can be computed analytically:

$$q(\mathbf{f}^{\text{un}}) = \sum_{k=1}^{K} \pi_k q_k(\mathbf{f}^{\text{un}}) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{|\mathcal{V}|} \mathcal{N}(\mathbf{f}_j^{\text{un}}; \mathbf{b}_{kj}, \boldsymbol{\Sigma}_{kj}) \tag{21}$$

$$\mathbf{b}_{kj} = \mathbf{A}_j \mathbf{m}_{kj} \tag{22}$$

$$\boldsymbol{\Sigma}_{kj} = \widetilde{\mathbf{K}}_j + \mathbf{A}_j \mathbf{S}_{kj} \mathbf{A}_j^T. \tag{23}$$

We note in Equation (21) that $q_k(\mathbf{f}^{\text{un}})$ has a block diagonal structure, which implies that we have the same expression for the $\mathcal{L}_{\text{ell}}$ as in Equation (5). Therefore, we obtain analogous estimates:

$$\mathcal{L}_{\text{ell}} = \sum_{n=1}^{N_{\text{seq}}} \sum_{k=1}^{K} \pi_k \left\langle \log p(\mathbf{y}_n|\mathbf{f}_n) \right\rangle_{q_{kn}(\mathbf{f}_n^{\text{un}})q(\mathbf{f}^{\text{bin}})}, \tag{24}$$

Here, as before, $q_{kn}(\mathbf{f}_n^{\text{un}})$ is a $(T_n \times |\mathcal{V}|)$–dimensional Gaussian with block-diagonal covariance $\boldsymbol{\Sigma}_{k(n)}$, each block of size $T_n \times T_n$. The main difference in this (sparse) case is that $\mathbf{b}_{k(n)}$ and $\boldsymbol{\Sigma}_{k(n)}$ are constrained by the expressions in Equations (22) and (23). Hence, the proof for the gradients follows the same derivation as in §1.2 above. ■

4 Gradients of $\mathcal{L}_{\text{elbo}}$ for sparse model

Here we give the gradients of the variational objective wrt the parameters for the variational distributions over the inducing variables, pairwise functions and hyper-parameters.

4.1 Inducing variables

4.1.1 KL term

As the structured likelihood does not affect the KL divergence term, the gradients corresponding to this term are similar to those in the non-structured case (Dezfouli and Bonilla, 2015). Let $\mathbf{K}_{zz}$ be the block-diagonal covariance with $|\mathcal{V}|$ blocks $\kappa(\mathbf{Z}_j, \mathbf{Z}_j)$, $j = 1, \ldots Q$. Additionally, lets assume the following definitions:

$$\mathbf{C}_{kl} \overset{\text{def}}{=} \mathbf{S}_k + \mathbf{S}_\ell, \tag{25}$$

$$\mathcal{N}_{k\ell} \overset{\text{def}}{=} \mathcal{N}(\mathbf{m}_k; \mathbf{m}_\ell, \mathbf{C}_{kl}), \tag{26}$$

$$z_k \overset{\text{def}}{=} \sum_{\ell=1}^{K} \pi_\ell \mathcal{N}_{k\ell}. \tag{27}$$

The gradients of $\mathcal{L}_{\text{kl}}$ wrt the posterior mean and posterior covariance for component k are:

$$\nabla_{\mathbf{m}_k} \mathcal{L}_{\text{cross}} = -\pi_k \mathbf{K}_{zz}^{-1} \mathbf{m}_k, \tag{28}$$

$$\nabla_{\mathbf{S}_k} \mathcal{L}_{\text{cross}} = -\frac{1}{2} \pi_k \mathbf{K}_{zz}^{-1} \tag{29}$$

$$\nabla_{\pi_k} \mathcal{L}_{\text{cross}} = -\frac{1}{2} \sum_{j=1}^{|\mathcal{V}|} [M \log 2\pi + \log |\kappa(\mathbf{Z}_j, \mathbf{Z}_j)| + \mathbf{m}_{kj}^T \kappa(\mathbf{Z}_j, \mathbf{Z}_j)^{-1} \mathbf{m}_{kj} + \text{ tr } \kappa(\mathbf{Z}_j, \mathbf{Z}_j)^{-1} \mathbf{S}_{kj}], \tag{30}$$

where we note that we compute $\mathbf{K}_{zz}^{-1}$ by inverting the corresponding blocks $\kappa(\mathbf{Z}_j, \mathbf{Z}_j)$ independently. The gradients of the entropy term wrt the variational parameters are:

$$\nabla_{\mathbf{m}_k} \hat{\mathcal{L}}_{\text{ent}} = \pi_k \sum_{\ell=1}^{K} \pi_\ell \left(\frac{\mathcal{N}_{k\ell}}{z_k} + \frac{\mathcal{N}_{k\ell}}{z_\ell} \right) \mathbf{C}_{kl}^{-1} (\mathbf{m}_k - \mathbf{m}_\ell), \tag{31}$$

$$\nabla_{\mathbf{S}_k} \hat{\mathcal{L}}_{\text{ent}} = \frac{1}{2} \pi_k \sum_{\ell=1}^{K} \pi_\ell \left(\frac{\mathcal{N}_{k\ell}}{z_k} + \frac{\mathcal{N}_{k\ell}}{z_\ell} \right) \left[\mathbf{C}_{kl}^{-1} - \mathbf{C}_{kl}^{-1} (\mathbf{m}_k - \mathbf{m}_\ell)(\mathbf{m}_k - \mathbf{m}_\ell)^T \mathbf{C}_{kl}^{-1} \right], \tag{32}$$

$$\nabla_{\pi_k} \hat{\mathcal{L}}_{\text{ent}} = -\log z_k - \sum_{\ell=1}^{K} \pi_\ell \frac{\mathcal{N}_{k\ell}}{z_\ell}.$$

4.1.2 Expected log likelihood term

Retaking the gradients in the full model in Equation (9), we have that:

$$\nabla_{\boldsymbol{\lambda}_k^{\text{un}}} \mathcal{L}_{\text{ell}}^{(k,n)} = \left\langle \log p(\mathbf{y}_n | \mathbf{f}_n) \nabla_{\boldsymbol{\lambda}_k^{\text{un}}} \log q_{kn}(\mathbf{f}_n^{\text{un}}) \right\rangle_{q_{kn}(\mathbf{f}_n^{\text{un}}) q(\mathbf{f}^{\text{bin}})}, \tag{33}$$

where the variational parameters $\boldsymbol{\lambda}_k^{\text{un}}$ are the posterior means and covariances ($\{\mathbf{m}_{kj}\}$ and $\{\mathbf{S}_{kj}\}$) of the inducing variables. As given in Equation (21), $q_k(\mathbf{f}^{\text{un}})$ factorizes over the latent process ($j = 1, \ldots, |\mathcal{V}|$), so do the marginals $q_{kn}(\mathbf{f}_n^{\text{un}})$, hence:

$$\nabla_{\boldsymbol{\lambda}_k^{\text{un}}} \log q_{kn}(\mathbf{f}_n^{\text{un}}) = \nabla_{\boldsymbol{\lambda}_k^{\text{un}}} \sum_{j=1}^{|\mathcal{V}|} \log \mathcal{N}(\mathbf{f}_{nj}^{\text{un}}; \mathbf{b}_{kjn}, \mathbf{\Sigma}_{kjn}), \tag{34}$$

where each of the distributions in Equation (34) is a T_n–dimensional Gaussian. Let us assume the following definitions:

$$\mathbf{X}_n : \text{all feature vectors corresponding to sequence } n \tag{35}$$

$$\mathbf{A}_{jn} \overset{\text{def}}{=} \kappa(\mathbf{X}_n, \mathbf{Z}_j) \kappa(\mathbf{Z}_j, \mathbf{Z}_j)^{-1} \tag{36}$$

$$\widetilde{\mathbf{K}}_j^n \overset{\text{def}}{=} \kappa_j(\mathbf{X}_n, \mathbf{X}_n) - \mathbf{A}_{jn} \kappa(\mathbf{Z}_j, \mathbf{X}_n), \text{ therefore:} \tag{37}$$

$$\mathbf{b}_{kjn} = \mathbf{A}_{jn} \mathbf{m}_{kj}, \tag{38}$$

$$\mathbf{\Sigma}_{kjn} = \widetilde{\mathbf{K}}_j^n + \mathbf{A}_{jn} \mathbf{S}_{kj} \mathbf{A}_{jn}^T. \tag{39}$$

Hence, the gradients of $\log q_k(\mathbf{f}^{\text{un}})$ wrt the the variational parameters of the unary posterior distributions over the inducing points are:

$$\nabla_{\mathbf{m}_{kj}} \log q_{kn}(\mathbf{f}_n^{\text{un}}) = \mathbf{A}_{jn}^T \boldsymbol{\Sigma}_{kjn}^{-1} \left(\mathbf{f}_{nj}^{\text{un}} - \mathbf{b}_{kjn}\right), \tag{40}$$

$$\nabla_{\mathbf{S}_{kj}} \log q_{kn}(\mathbf{f}_n^{\text{un}}) = \frac{1}{2} \mathbf{A}_{jn}^T \left[\boldsymbol{\Sigma}_{kjn}^{-1} (\mathbf{f}_{nj}^{\text{un}} - \mathbf{b}_{kjn})(\mathbf{f}_{nj}^{\text{un}} - \mathbf{b}_{kjn})^T \boldsymbol{\Sigma}_{kjn}^{-1} - \boldsymbol{\Sigma}_{kjn}^{-1}\right] \mathbf{A}_{jn} \tag{41}$$

Therefore, the gradients of $\mathcal{L}_{\text{ell}}$ wrt the parameters of the distributions over unary functions are:

$$\nabla_{\mathbf{m}_{kj}} \mathcal{L}_{\text{ell}} = \frac{\pi_k}{S} \kappa(\mathbf{Z}_j, \mathbf{Z}_j)^{-1} \sum_{n=1}^{N_{\text{seq}}} \kappa(\mathbf{Z}_j, \mathbf{X}_n) (\boldsymbol{\Sigma}_{kjn})^{-1} \sum_{i=1}^{S} (\mathbf{f}_{nkij}^{\text{un}} - \mathbf{b}_{kjn}) \log p(\mathbf{y}_n | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}}), \tag{42}$$

$$\begin{aligned}\nabla_{\mathbf{S}_{kj}} \mathcal{L}_{\text{ell}} = \frac{\pi_k}{2S} \sum_{n=1}^{N_{\text{seq}}} \mathbf{A}_{jn}^T \Big\{ \sum_{i=1}^{S} \big[& (\boldsymbol{\Sigma}_{kjn})^{-1} (\mathbf{f}_{nkij}^{\text{un}} - \mathbf{b}_{kjn})((\mathbf{f}_{nj}^{\text{un}})^{(k,i)} - \mathbf{b}_{kjn})^T (\boldsymbol{\Sigma}_{kjn})^{-1} \\ & - (\boldsymbol{\Sigma}_{kjn})^{-1} \big] \log p(\mathbf{y}_n | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}}) \Big\} \mathbf{A}_{jn} \end{aligned} \tag{43}$$

4.1.3 Pairwise functions

The gradients of the $\mathcal{L}_{\text{kl}}^{\text{bin}}$ wrt the parameters of the posterior over pairwise functions are given by:

$$\nabla_{\mathbf{m}^{\text{bin}}} \mathcal{L}_{\text{kl}}^{\text{bin}} = -(\mathbf{K}^{\text{bin}})^{-1} \mathbf{m}^{\text{bin}} \tag{44}$$

$$\nabla_{\mathbf{S}^{\text{bin}}} \mathcal{L}_{\text{kl}}^{\text{bin}} = \frac{1}{2} \left((\mathbf{S}^{\text{bin}})^{-1} - (\mathbf{K}^{\text{bin}})^{-1} \right) \tag{45}$$

The gradients of the $\mathcal{L}_{\text{ell}}$ wrt the parameters of the posterior over pairwise functions are given by:

$$\nabla_{\mathbf{m}^{\text{bin}}} \mathcal{L}_{\text{ell}} = \frac{1}{S} \sum_{n=1}^{N_{\text{seq}}} \sum_{k=1}^{K} \pi_k \sum_{i=1}^{S} (\mathbf{S}^{\text{bin}})^{-1} (\mathbf{f}_i^{\text{bin}} - \mathbf{m}^{\text{bin}}) \log p(\mathbf{y}_n | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}}) \tag{46}$$

$$\nabla_{\mathbf{S}^{\text{bin}}} \mathcal{L}_{\text{ell}} = \frac{1}{2S} \sum_{n=1}^{N_{\text{seq}}} \sum_{k=1}^{K} \pi_k \sum_{i=1}^{S} [(\mathbf{S}^{\text{bin}})^{-1} (\mathbf{f}_i^{\text{bin}} - \mathbf{m}^{\text{bin}})(\mathbf{f}_i^{\text{bin}} - \mathbf{m}^{\text{bin}})^T (\mathbf{S}^{\text{bin}})^{-1} - (\mathbf{S}^{\text{bin}})^{-1}] \log p(\mathbf{y}_n | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}}) \tag{47}$$

5 Piecewise pseudolikelihood

Piecewise pseudolikelihood (Sutton and McCallum, 2007) approximates the likelihood $p(\mathbf{y}_n | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}})$ of sequence n given the latent functions $\mathbf{f}_{nki}^{\text{un}}$ and $\mathbf{f}_i^{\text{bin}}$ by computing the product,[1] for every single factor and every variable occurring in it, of the conditional probability of the variable given its neighbours with respect to that factor.

In our linear model, this yields the following expression for the log pseudolikelihood $\tilde{p}(\mathbf{y}_n | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}})$:

$$\log \tilde{p}(\mathbf{y}_n | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}}) = \sum_{w=1}^{W_n} \log p((\mathbf{y}_n)_w | \mathbf{f}_{nki}^{\text{un}}) + \sum_{|w_1 - w_2| = 1} \log p((\mathbf{y}_n)_{w_1} | (\mathbf{y}_n)_{w_2}, \mathbf{f}_i^{\text{bin}}) = \sum_{w=1}^{W_n} \log \tilde{p}((\mathbf{y}_n)_w | \mathbf{f}_{nki}^{\text{un}}, \mathbf{f}_i^{\text{bin}})$$

where W_n is the number of words in sentence n and

$$p((\mathbf{y}_n)_w | \mathbf{f}_{nki}^{\text{un}}) \propto \exp(\mathbf{f}_{nki\mathbf{y}_n}^{\text{un}}(w)); \tag{48}$$

$$p((\mathbf{y}_n)_w | (\mathbf{y}_n)_{w+1}, \mathbf{f}_i^{\text{bin}}) \propto \exp(\mathbf{f}_i^{\text{bin}}((\mathbf{y}_n)_w, (\mathbf{y}_n)_{w+1})); \tag{49}$$

$$p((\mathbf{y}_n)_{w+1} | (\mathbf{y}_n)_w, \mathbf{f}_i^{\text{bin}}) \propto \exp(\mathbf{f}_i^{\text{bin}}((\mathbf{y}_n)_w, (\mathbf{y}_n)_{w+1})). \tag{50}$$

[1] Here i represents the index of the specific samples $\mathbf{f}_{nki}^{\text{un}}$ and $\mathbf{f}_i^{\text{bin}}$ taken from our distributions $q_{kn}(\mathbf{f}_n^{\text{un}})$ and $q(\mathbf{f}^{\text{bin}})$.

6 Experiments

6.1 Experimental set-up

Before starting all experiments, our program selected the positions of the 500 inducing points by performing K-means clustering on the training data. The initial values of the means (one mean value for every inducing point p and for every possible label l) are set as the fraction of the points of the training set in the cluster whose centroid is p that belong to label l.

As described in Algorithm 1, we adaptively choose the correct learning rates for all parameters by searching (beginning from an initial guess, and doubling or dividing it by two as needed) for the biggest step size that causes the objective to decrease over twenty stochastic optimization steps. The step size to use for the given parameter is taken as one fourth of this value, to avoid instability. The initial step sizes were 0.5 for unary mean parameters, 0.0005 for binary mean parameters, 0.005 for unary covariance parameeters, 0.0005 for binary covariance parameters and 1.0 for (linear) kernel hyperparameters (these values were selected only to speed up, insofar as possible, the process of parameter search); and the optimal step sizes were found in the order unary mean $\rightarrow$ unary covariance $\rightarrow$ binary mean $\rightarrow$ binary covariance $\rightarrow$ kernel hyperparameter. Of course this is not an exhaustive grid search; but it is less computationally expensive and works well in practice.

6.2 Optimization

Then we optimize the three sets of parameters (unaries, binaries, hyperparameters) in a global loop, in the same order as mentioned earlier, through standard Stochastic Gradient Descent, until the time limit is reached, using 4000 new random samples each step for the estimation of the relevant gradients and averages.[2]

For the small-scale experiments, the variational parameters for unary nodes are optimized for 500 iterations, variational parameters for pairwise nodes are optimized for 100 iterations, and hyper-parameters are updated for 20 iterations.

We keep a tight bound (10 in the unary case, 20 in the binary one) on the maximum possible absolute values that covariance parameters can take. If an update would bring their value beyond it, we recompute the gradients with new samples: oftentimes, this resolves the issue (in brief, because the faulty update was due to a bad estimation of the gradients). If the problem persists, we disregard the current sentence and move on to the next one; and if after ten attempts the problem still persists, we move back to the latest "safe" position that did not cause out-of-bound errors. In this way, we can use a relatively small number of samples and maintain rather aggressive step sizes while recovering neatly from out-of-bounds errors.

The setting for the large-scale experiments was similar, except that the optimization schedule was $(12500, 2500, 500)$ (that is 12500 unary optimization steps, 2500 binary ones and 500 hyperparameter ones) for the big BASE NP and CHUNKING experiments. All the experiments were run for four hours, not counting initial clustering or final prediction (but counting step size selection). For comparison, the (small-scale) GP-ESS experiments, which were run for 250,000 elliptical slice sampling steps, took on average 11.34 hours for CHUNKING, 21.19 hours for BASE NP, 2.07 hours for SEGMENTATION and 15.75 for JAPANESE NE.

6.3 Performance profiles

Figure 1 shows the performance of our algorithm as a function of time. We see that the test likelihood decreases very regularly in all the folds and so does overall the error rate, albeit with more variability. The bulk of the optimization, both with respect to the test likelihood and with respect to the error rate, occurs during the first 120 minutes. This suggests that the kind of approach described in this paper might be particularly suited for cases in which expected loglikelihood of the prediction and speed of convergence are priorities.

[2] When computing the gradient of the kernel hyperparameter, 8000 samples were used instead to insure greater stability.

Algorithm 1 Method for computing the step size

```
function CHECK_STEPSIZE(parameters, step_size, num_to_check=20)
    to_check ← num_to_check sentences at random from the training set;
    old_obj ← current value of the objective function;
    for $i \leftarrow 0 \ldots$ num_to_check do
        grad ← gradient of objective wrt parameters;
        parameters ← parameters - step_size · grad;
        if parameters are out of bounds then
            return False;
    new_obj ← current value of the objective function;
    if new_obj < old_obj then
        return True;
    else
        return False;
function SEARCH_STEPSIZE_UP(parameters, initial_step_size)
    step_size ← initial_step_size;
    old_params ← current parameter values of the model;
    while True do
        is_good ← CHECK_STEPSIZE(parameters, step_size);
        if is_good = False then
            parameters ← old_params;
            return $\frac{\texttt{step\_size}}{\texttt{factor}^2}$;
        step_size ← step_size · factor;
function SEARCH_STEPSIZE_DOWN(parameters, initial_step_size)
    step_size ← initial_step_size;
    old_params ← current parameter values of the model;
    while True do
        is_good ← CHECK_STEPSIZE(parameters, step_size);
        if is_good = True then
            parameters ← old_params;
            return $\frac{\texttt{step\_size}}{\texttt{factor}}$;
        step_size ← step_size/factor;
function CHOOSE_STEPSIZE(parameters, initial_step_size, factor = 2)
    step_size ← SEARCH_STEPSIZE_UP(parameters, initial_step_size, factor);
    if step_size == $\frac{\texttt{initial\_step\_size}}{\texttt{factor}^2}$ then
        step_size ← SEARCH_STEPSIZE_DOWN(parameters, initial_step_size, factor);
    return step_size/factor;
```

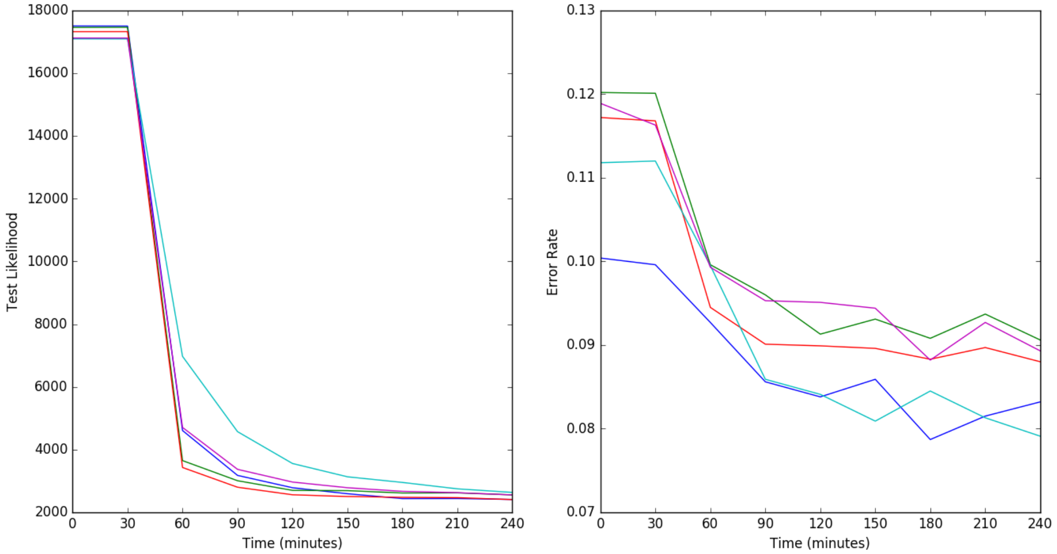

Figure 1: The test performance of GP-VAR-T on CHUNKING for the large scale experiment as a function of time.

References

Amir Dezfouli and Edwin V Bonilla. Scalable inference for Gaussian process models with black-box likelihoods. In *NIPS*. 2015.

Charles Sutton and Andrew McCallum. Piecewise pseudolikelihood for efficient training of conditional random fields. In *ICML*, 2007.