Online Variational Bayesian Inference: Algorithms for Sparse Gaussian Processes and Theoretical Bounds

Cuong V. Nguyen [1] **Thang D. Bui** [1] **Yingzhen Li** [1] **Richard E. Turner** [1]

Abstract

Sparse approximations for Gaussian process models provide a suite of methods that enable these models to be deployed in large data regime and enable analytic intractabilities to be sidestepped. However, the field lacks a principled method to handle streaming data, which are important for time-series analysis. The small number of existing approaches either use suboptimal hand-crafted heuristics for hyperparameter learning, or suffer from catastrophic forgetting or slow updating when new data arrive. This paper develops a new principled framework for deploying Gaussian process probabilistic models in the streaming setting, providing principled methods for learning hyperparameters and optimising pseudo-input locations. New theoretical bounds for general online variational Bayesian inference are also given and discussed in the paper.

1. Introduction

Probabilistic models employing Gaussian processes (GPs) have become a standard approach to solving many machine learning tasks, thanks largely to the modelling flexibility, robustness to overfitting, and well-calibrated uncertainty estimates afforded by the approach (Rasmussen & Williams, 2006). One of the pillars of the modern GP probabilistic modelling approach is a set of sparse approximation schemes that allow the prohibitive computational cost of GP methods, typically $\mathcal{O}(N^3)$ for training and $\mathcal{O}(N^2)$ for prediction where N is the number of training points, to be substantially reduced whilst still retaining accuracy. Arguably the most important and influential approximations of this sort are pseudo-point approximation schemes that employ a set of $M \ll N$ pseudo-points to summarise the observational data thereby reducing computational costs to $\mathcal{O}(NM^2)$ and $\mathcal{O}(M^2)$ for training and prediction, respectively (Snelson & Ghahramani, 2006; Titsias, 2009). Stochastic optimisation methods that employ mini-batches of training data can be used to further reduce computational costs (Hensman et al., 2013; 2015; Dezfouli & Bonilla, 2015; Hernández-Lobato & Hernández-Lobato, 2016), allowing GPs to be scaled to datasets comprising millions of data points.

The focus of this paper is to provide a comprehensive framework for deploying the GP probabilistic modelling approach to streaming data, which arrive sequentially in an online fashion, possibly in small batches, and whose number are not known a priori. The vast majority of previous work has focussed exclusively on the batch setting and there is not a satisfactory framework that supports learning and approximation in the streaming setting. A naïve approach might simply incorporate each new datum as they arrived into an ever growing dataset and retrain the GP model from scratch each time. With infinite computational resources, this approach is optimal, but in the majority of practical settings it is intractable. A feasible alternative would train on just the most recent K training data points, but this completely ignores potentially large amounts of informative training data and it does not provide a method for incorporating the old model into the new one which would save computation (except through initialisation of the hyperparameters). Existing, sparse approximation schemes could be applied in the same manner, but they merely allow K to be increased, rather than allowing all previous data to be leveraged, and again do not utilise intermediate approximate fits.

What is needed is a method for performing learning and sparse approximation that incrementally updates the previously fit model using the new data. Such an approach would utilise all the previous training data (as they will have been incorporated into the previously fit model) as well as leverage as much of the previous computation as possible at each stage (since the algorithm only requires access to the data at the current time point). This paper provides a new principled framework for deploying GP probabilistic models in the streaming setting. The framework subsumes Csató's seminal approach to online regression (Csató, 2002) that was based upon the variational free energy (VFE) approach to approximate inference. In the new framework this algorithm is recovered as a special case. We

[1]Department of Engineering, University of Cambridge, UK. Correspondence to: Cuong V. Nguyen <vcn22@cam.ac.uk>.

ICML 2017 Time Series Workshop, Sydney, Australia, 2017.

also provide principled methods for learning hyperparameters and optimising pseudo-input locations. The approach also relates to the streaming variational Bayes framework (Broderick et al., 2013).

The paper also proves novel theoretical bounds for general online variational Bayesian inference. Previous theoretical work for variational Bayesian inference mainly focused on the batch (offline) setting (Seeger, 2002; Alquier et al., 2016), and a theory for the online setting is needed. Our paper presents the first attempt towards establishing theoretical guarantees for online variational Bayesian inference.

2. Online Variational Free Energy Inference and Learning for Sparse GP

2.1. Sparse Gaussian Process for Regression

Given N input and real-valued output pairs $\{\mathbf{x}_n, y_n\}_{n=1}^N$, a standard GP regression model assumes $y_n = f(\mathbf{x}_n) + \epsilon_n$, where f is an unknown function that is corrupted by Gaussian observation noise $\epsilon_n \sim \mathcal{N}(0, \sigma_y^2)$. Typically, f is assumed to be drawn from a zero-mean GP prior $f \sim \mathcal{GP}(\mathbf{0}, k(\cdot,\cdot|\theta))$ whose covariance function depends on hyperparameters θ. In this paper, we focus on the variational free energy (VFE) sparse approximation scheme (Titsias, 2009; Matthews et al., 2016) which lower bounds the marginal likelihood of the data using a variational distribution $q(f)$ over the latent function:

$$\log p(\mathbf{y}|\theta) = \log \int \mathrm{d}f\ p(\mathbf{y}, f|\theta) \geq \int \mathrm{d}f\ q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}_{\mathrm{vfe}}(q, \theta).$$

This variational bound approximates the marginal likelihood and can be used for learning the hyperparameters θ.

In order to arrive at a computationally tractable method, the approximate posterior is parameterized via a set of pseudo-points $\mathbf{u}$ that are a subset of the function values $f = \{f_{\neq \mathbf{u}}, \mathbf{u}\}$ and which will summarise the data. Specifically, the approximate posterior is assumed to be $q(f) = p(f_{\neq \mathbf{u}}|\mathbf{u}, \theta)q(\mathbf{u})$, where $q(\mathbf{u})$ is a variational distribution over $\mathbf{u}$ and $p(f_{\neq \mathbf{u}}|\mathbf{u}, \theta)$ is the prior distribution of the remaining latent function values $f_{\neq \mathbf{u}}$. This assumption allows the variational lower bound to be computationally tractable. We also use $\mathbf{z}$ to denote the input locations of $\mathbf{u}$.

For the GP model considered here closed-form expressions for the optimal variational approximation $q_{\mathrm{vfe}}(f)$ and the optimal variational bound $\mathcal{F}_{\mathrm{vfe}}(\theta) = \max_{q(\mathbf{u})} \mathcal{F}_{\mathrm{vfe}}(q(\mathbf{u}), \theta)$ are available. In order for this method to perform well, it is necessary to adapt the pseudo-point input locations, e.g. by optimising the variational free energy, so that the pseudo-data distribute themselves over the training data.

2.2. Online VFE Inference and Learning for Sparse GP

This paper assumes data arrive sequentially so that at each step new data points $\mathbf{y}_{\mathrm{new}}$ are added to the old dataset $\mathbf{y}_{\mathrm{old}}$. The goal is to approximate the marginal likelihood and the posterior of the latent process at each step, which can be used for anytime prediction. The hyperparameters will also be adjusted online. Importantly, we assume that we can only access the current data points $\mathbf{y}_{\mathrm{new}}$ directly for computational reasons (it might be too expensive to hold $\mathbf{y}_{\mathrm{old}}$ and $\mathbf{x}_{1:N_{\mathrm{old}}}$ in memory, for example, or approximations made at the previous step must be reused to reduce computational overhead). So the effect of the old data on the current posterior must be propagated through the previous posterior. We will now develop a new sparse VFE approximation for this purpose, that compactly summarises the old data via pseudo-points. The pseudo-inputs will also be adjusted online since this is critical as new parts of the input space will be revealed over time. The framework is easily extensible to more complex non-linear models.[1]

Consider an approximation to the true posterior at the previous step, $q_{\mathrm{old}}(f)$, which must be updated to form the new approximation $q_{\mathrm{new}}(f)$,

$$q_{\mathrm{old}}(f) \approx p(f|\mathbf{y}_{\mathrm{old}}) = \frac{1}{\mathcal{Z}_1(\theta_{\mathrm{old}})} p(f|\theta_{\mathrm{old}}) p(\mathbf{y}_{\mathrm{old}}|f), \quad (1)$$

$$q_{\mathrm{new}}(f) \approx p(f|\mathbf{y}_{\mathrm{old}}, \mathbf{y}_{\mathrm{new}}) = \frac{1}{\mathcal{Z}_2(\theta_{\mathrm{new}})} p(f|\theta_{\mathrm{new}}) p(\mathbf{y}_{\mathrm{old}}|f) p(\mathbf{y}_{\mathrm{new}}|f). \quad (2)$$

Whilst the updated exact posterior $p(f|\mathbf{y}_{\mathrm{old}}, \mathbf{y}_{\mathrm{new}})$ balances the contribution of old and new data through their likelihoods, the new approximation cannot access $p(\mathbf{y}_{\mathrm{old}}|f)$ directly. Instead, we can find an approximation of $p(\mathbf{y}_{\mathrm{old}}|f)$ by inverting (1), that is $p(\mathbf{y}_{\mathrm{old}}|f) \approx \mathcal{Z}_1(\theta_{\mathrm{old}}) q_{\mathrm{old}}(f) / p(f|\theta_{\mathrm{old}})$. Substituting this into (2) yields

$$\hat{p}(f|\mathbf{y}_{\mathrm{old}}, \mathbf{y}_{\mathrm{new}}) = \frac{\mathcal{Z}_1(\theta_{\mathrm{old}})}{\mathcal{Z}_2(\theta_{\mathrm{new}})} p(f|\theta_{\mathrm{new}}) p(\mathbf{y}_{\mathrm{new}}|f) \frac{q_{\mathrm{old}}(f)}{p(f|\theta_{\mathrm{old}})}.$$

Although it is tempting to use this as the new posterior, $q_{\mathrm{new}}(f) = \hat{p}(f|\mathbf{y}_{\mathrm{old}}, \mathbf{y}_{\mathrm{new}})$, this recovers exact GP regression with fixed hyperparameters and it is intractable. So, instead, we consider a variational update that projects the distribution back to a tractable form using pseudo-data. At this stage we allow the pseudo-data input locations in the new approximation to differ from those in the old one. This is required if new regions of input space are gradually revealed, as for example in typical time-series applications. Let $\mathbf{a} = f(\mathbf{z}_{\mathrm{old}})$ and $\mathbf{b} = f(\mathbf{z}_{\mathrm{new}})$ be the function values at the pseudo-inputs before and after seeing new data. Note that the number of pseudo-

[1] Due to space constraints, we only include key results here. Full results and derivations can be found in https://arxiv.org/abs/1705.07131.

points, $M_{\mathbf{a}} = |\mathbf{a}|$ and $M_{\mathbf{b}} = |\mathbf{b}|$ are not necessarily restricted to be the same. The form of the approximate posterior mirrors that in the batch case, that is, the previous approximate posterior, $q_{\text{old}}(f) = p(f_{\neq \mathbf{a}}|\mathbf{a}, \theta_{\text{old}})q_{\text{old}}(\mathbf{a})$ where we assume $q_{\text{old}}(\mathbf{a}) = \mathcal{N}(\mathbf{a}; \mathbf{m}_{\mathbf{a}}, \mathbf{S}_{\mathbf{a}})$. The new posterior approximation takes the same form, but with the new pseudo-points and new hyperparameters: $q_{\text{new}}(f) = p(f_{\neq \mathbf{b}}|\mathbf{b}, \theta_{\text{new}})q_{\text{new}}(\mathbf{b})$. Similar to the batch case, this approximate inference problem can be turned into an optimisation problem using variational inference. Specifically, consider the KL-divergence between the approximate posterior and the running posterior:

$$\begin{aligned}\text{KL} &= \left\langle \log \frac{p(f_{\neq \mathbf{b}}|\mathbf{b}, \theta_{\text{new}})q_{\text{new}}(\mathbf{b})}{\frac{\mathcal{Z}_1(\theta_{\text{old}})}{\mathcal{Z}_2(\theta_{\text{new}})} p(f|\theta_{\text{new}})p(\mathbf{y}_{\text{new}}|f)\frac{q_{\text{old}}(f)}{p(f|\theta_{\text{old}})}} \right\rangle_{q_{\text{new}}(f)} \\ &= \log \frac{\mathcal{Z}_2(\theta_{\text{new}})}{\mathcal{Z}_1(\theta_{\text{old}})} + \left\langle \log \frac{p(\mathbf{a}|\theta_{\text{old}})q_{\text{new}}(\mathbf{b})}{p(\mathbf{b}|\theta_{\text{new}})q_{\text{old}}(\mathbf{a})p(\mathbf{y}_{\text{new}}|f)} \right\rangle_{q_{\text{new}}(f)}\end{aligned}$$

Since the KL divergence is non-negative, the second term in the expression above is the negative approximate lower bound of the online log marginal likelihood (as $\mathcal{Z}_2/\mathcal{Z}_1 \approx p(\mathbf{y}_{\text{new}}|\mathbf{y}_{\text{old}})$), or the variational free energy $\mathcal{F}(q_{\text{new}}(f), \theta_{\text{new}})$. By setting the derivative of $\mathcal{F}$ w.r.t. $q(\mathbf{b})$ equal to 0, the optimal approximate posterior can be obtained for the regression case,[2]

$$\begin{aligned}q_{\text{vfe}}(\mathbf{b}) &\propto p(\mathbf{b}) \exp\Big(\int \mathrm{d}\mathbf{a}\, p(\mathbf{a}|\mathbf{b}) \log \frac{q_{\text{old}}(\mathbf{a})}{p(\mathbf{a}|\theta_{\text{old}})} \\ &\quad + \int \mathrm{d}\mathbf{f}\, p(\mathbf{f}|\mathbf{b}) \log p(\mathbf{y}_{\text{new}}|\mathbf{f}) \Big) \\ &\propto p(\mathbf{b}) \mathcal{N}(\hat{\mathbf{y}}; \mathbf{K}_{\hat{\mathbf{f}}\mathbf{b}} \mathbf{K}_{\mathbf{bb}}^{-1} \mathbf{b}, \Sigma_{\hat{\mathbf{y}},\text{vfe}}),\end{aligned}$$

where $\mathbf{f}$ is the latent function values at the new training points, $\mathbf{D}_{\mathbf{a}} = (\mathbf{S}_{\mathbf{a}}^{-1} - \mathbf{K}_{\mathbf{aa}}'^{-1})^{-1}$ and,

$$\hat{\mathbf{y}} = \begin{bmatrix} \mathbf{y}_{\text{new}} \\ \mathbf{D}_{\mathbf{a}} \mathbf{S}_{\mathbf{a}}^{-1} \mathbf{m}_{\mathbf{a}} \end{bmatrix}, \; \mathbf{K}_{\hat{\mathbf{f}}\mathbf{b}} = \begin{bmatrix} \mathbf{K}_{\mathbf{fb}} \\ \mathbf{K}_{\mathbf{ab}} \end{bmatrix}, \; \Sigma_{\hat{\mathbf{y}},\text{vfe}} = \begin{bmatrix} \sigma_y^2 \mathrm{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_{\mathbf{a}} \end{bmatrix}.$$

The negative variational free energy is also analytically available and can be used for hyperparameter learning. The computational complexity and memory overhead of the new method is of the same order as the uncollapsed stochastic variational inference approach. The procedure is demonstrated on a toy regression example as shown in fig. 1. These results can be extended to handle non-Gaussian likelihoods, as shown in fig. 2.

3. Theoretical Bounds for Online VFE Inference

In this section, we give some novel theoretical bounds for online VFE inference in general. We shall assume the hyperparameters are fixed and thus suppress the dependence on the hyperparameters in our notations. Thus, the considered online VFE inference algorithm can be written as:

$$q_n = \arg\min_{q \in \Omega} \text{KL}\left(q(f) \parallel \frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f) \right), \quad (3)$$

which approximates the current posterior by minimising the KL-divergence to the posterior computed from the previous variational distribution q_{n-1}. In the above equation, $\mathbf{y}_n$ is the batch of data received at the current iteration n, $Z_n = \int p(\mathbf{y}_n|f) q_{n-1}(f) df$ is the normalization factor, Ω is the space of all distributions that we use to approximate the posteriors, and $q_0 = p$ (the prior).

Equation (3) can also be interpreted as minimising an approximation of the KL-divergence between $q(f)$ and $p(f|\mathbf{y}_{1:n})$. Specifically, note that for all $q \in \Omega$,

$$\begin{aligned}\text{KL}(q(f) \parallel p(f|\mathbf{y}_{1:n})) &= \text{KL}(q(f) \parallel p(f|\mathbf{y}_{1:n-1})) \\ &\quad - \int q(f) \log p(\mathbf{y}_n|f) df + \log p(\mathbf{y}_n|\mathbf{y}_{1:n-1}),\end{aligned}$$

and minimising the RHS of this equation with $p(f|\mathbf{y}_{1:n-1})$ replaced by the variational distribution $q_{n-1}(f)$ is equivalent to (3).

We now give the following bounds for the algorithm. These bounds are general and not specific to GP models.

3.1. Bounds for Optimal Variational Distributions

The first quantity of interest is the optimal KL-divergence of the approximated posteriors. Formally, for $n \geq 1$, we define the optimal variational distribution q_n^* as:

$$q_n^* = \arg\min_{q \in \Omega} \text{KL}(q(f) \parallel p(f|\mathbf{y}_{1:n})), \quad (4)$$

and we are interested in upper bounding the optimal KL-divergence $\text{KL}(q_n^*(f) \parallel p(f|\mathbf{y}_{1:n}))$. We will give two upper bounds for this divergence: a one-step and an n-step bound.

Lemma 1 (One-step bound). *For all $n \geq 1$,*

$$\begin{aligned}KL(q_n^*(f) \parallel p(f|\mathbf{y}_{1:n})) &\leq KL(q_{n-1}^*(f) \parallel p(f|\mathbf{y}_{1:n-1})) \\ &\quad + \log p(\mathbf{y}_n|\mathbf{y}_{1:n-1}) - \mathbb{E}_{f \sim q_{n-1}^*}[\log p(\mathbf{y}_n|f)].\end{aligned}$$

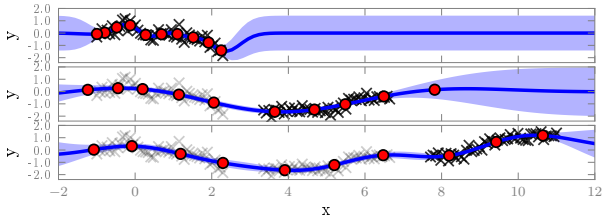

Figure 1. Online VFE inference and learning on a toy time-series. Black crosses are data points (past points are greyed out), red circles are pseudo-points, and blue lines and shaded areas are the marginal predictive means and confidence intervals at test points.

[2] Note that we have dropped θ_{new} from $p(\mathbf{b}|\theta_{\text{new}})$, $p(\mathbf{a}|\mathbf{b}, \theta_{\text{new}})$ and $p(\mathbf{f}|\mathbf{b}, \theta_{\text{new}})$ to lighten the notation.

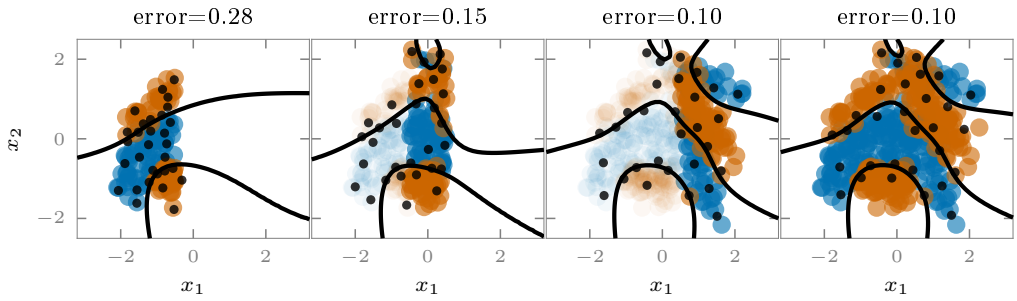

Figure 2. Inference and learning on a binary classification task in a non-iid streaming setting. The right-most plot shows the prediction made by using sparse variational inference on full training data (Hensman et al., 2015). Past observations are greyed out. The pseudo-points are shown as black dots and the black curves show the decision boundary.

This one-step bound states that the optimal KL-divergence at each iteration is upper bounded by the optimal KL-divergence at the previous iteration plus the difference between the exact log-likelihood and the expected log-likelihood (w.r.t. the previous optimal variational distribution q^*_{n-1}) of the current batch of data.

Asymptotically, if $q^*_{n-1}(f) \to p(f|\mathbf{y}_{1:n-1})$, i.e. we can exactly estimate $p(f|\mathbf{y}_{1:n-1})$ with the variational distribution $q^*_{n-1}(f)$, the RHS of Lemma 1 becomes $\log p(\mathbf{y}_n|\mathbf{y}_{1:n-1}) - \mathbb{E}_{f\sim p(f|\mathbf{y}_{1:n-1})}[\log p(\mathbf{y}_n|f)]$, the difference in the Jensen's inequality $\log p(\mathbf{y}_n|\mathbf{y}_{1:n-1}) \geq \mathbb{E}_{f\sim p(f|\mathbf{y}_{1:n-1})}[\log p(\mathbf{y}_n|f)]$.

Using Lemma 1, we can prove the following n-step bound.

Theorem 1 (n-step bound)**.** *For all $n \geq 1$,*

$$KL(q^*_n(f) \parallel p(f|\mathbf{y}_{1:n})) \leq KL(q^*_0(f) \parallel p(f)) + \log p(\mathbf{y}_{1:n}) - \sum_{i=1}^{n} \mathbb{E}_{f\sim q^*_{i-1}}[\log p(\mathbf{y}_i|f)].$$

This n-step bound states that the optimal KL-divergence at each iteration is upper bounded by the optimal KL-divergence at the first iteration plus the difference between the data log-likelihood and the total expected log-likelihood of each batch (w.r.t. the optimal variational distribution obtained from the previous batch).

If $p \in \Omega$ and $q^*_0 = p$, then $\mathrm{KL}(q^*_0(f) \parallel p(f)) = 0$. In this case, the RHS of Theorem 1 becomes $\log p(\mathbf{y}_{1:n}) - \sum_{i=1}^{n} \mathbb{E}_{f\sim q^*_{i-1}}[\log p(\mathbf{y}_i|f)]$. Since $\mathrm{KL}(q^*_n(f) \parallel p(f|\mathbf{y}_{1:n})) \geq 0$, we also have $\sum_{i=1}^{n} \mathbb{E}_{f\sim q^*_{i-1}}[\log p(\mathbf{y}_i|f)] \leq \log p(\mathbf{y}_{1:n})$.

3.2. Bounds for Approximate Variational Distributions

Another quantity of interest is the KL-divergence of the variational distribution q_n in (3). To bound this quantity, we consider the *divergence regret* of q_n compared to the optimal q^*_n, which is defined as:

$$R(q_n) = \mathrm{KL}(q_n(f) \parallel p(f|\mathbf{y}_{1:n})) - \mathrm{KL}(q^*_n(f) \parallel p(f|\mathbf{y}_{1:n})).$$

The following theorem gives an upper bound for $R(q_n)$.

Theorem 2. *For all $n \geq 1$, $R(q_n) \leq \int |\log \frac{q_{n-1}(f)}{p(f|\mathbf{y}_{1:n-1})}| df \sqrt{2\, KL(q^*_n(f) \parallel \frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f))}$.*

The bound in this theorem depends both on $\mathrm{KL}(q^*_n(f) \parallel \frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f))$ and the integral $\int |\log(q_{n-1}(f)/p(f|\mathbf{y}_{1:n-1}))| df$. This integral is a form of distance between $q_{n-1}(f)$ and $p(f|\mathbf{y}_{1:n-1})$, which specifies how well we have done up to the previous step.

As a consequence of Theorem 2, if the integral $\int |\log(q_{n-1}(f)/p(f|\mathbf{y}_{1:n-1}))| df$ is bounded and $\frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f)) \to q^*_n(f)$, then $R(q_n) \to 0$. On the other hand, if $\mathrm{KL}(q^*_n(f) \parallel \frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f))$ is bounded and $q_{n-1}(f) \to p(f|\mathbf{y}_{1:n-1})$, we also have $R(q_n) \to 0$. In these cases, $\mathrm{KL}(q_n(f) \parallel p(f|\mathbf{y}_{1:n})) \to \mathrm{KL}(q^*_n(f) \parallel p(f|\mathbf{y}_{1:n}))$.

We note that Theorem 2 can be slightly modified to give another bound: $R(q_n) \leq \int |\log(\frac{p(\mathbf{y}_n|f) q_{n-1}(f)/Z_n}{p(f|\mathbf{y}_{1:n})})| df \sqrt{2\, \mathrm{KL}(q^*_n(f) \parallel \frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f))}$, which contains the integral $\int |\log(\frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f)/p(f|\mathbf{y}_{1:n}))| df$ instead of the integral $\int |\log(q_{n-1}(f)/p(f|\mathbf{y}_{1:n-1}))| df$.

Combining Theorems 1 and 2, we can obtain the following bound for $\mathrm{KL}(q_n(f) \parallel p(f|\mathbf{y}_{1:n}))$:

$$\mathrm{KL}(q_n(f) \parallel p(f|\mathbf{y}_{1:n})) \leq \mathrm{KL}(q^*_0(f) \parallel p(f)) + \log p(\mathbf{y}_{1:n}) - \textstyle\sum_{i=1}^{n} \mathbb{E}_{f\sim q^*_{i-1}}[\log p(\mathbf{y}_i|f)] + \sqrt{2\mathrm{KL}(q^*_n(f) \parallel \frac{1}{Z_n} p(\mathbf{y}_n|f) q_{n-1}(f))} \int |\log \frac{q_{n-1}(f)}{p(f|\mathbf{y}_{1:n-1})}| df.$$

4. Conclusion

We have introduced a novel online inference and learning framework for GP models. The framework unifies disparate methods in the literature and greatly extends them, allowing sequential updates of the approximate posterior and online hyperparameter optimisation in a principled manner. We also proved new bounds as a preliminary step towards establishing theoretical guarantees for online variational Bayesian inference.

Acknowledgements

Cuong V. Nguyen is supported by EPSRC grant EP/M0269571. Thang D. Bui is supported by Google European Doctoral Fellowship. Yingzhen Li thanks the Schlumberger Foundation FFTF fellowship. Richard Turner thanks EPSRC grants EP/M0269571 and EP/L000776/1 as well as Google for funding.

References

Alquier, Pierre, Ridgway, James, and Chopin, Nicolas. On the properties of variational approximations of Gibbs posteriors. *Journal of Machine Learning Research*, 2016.

Broderick, Tamara, Boyd, Nicholas, Wibisono, Andre, Wilson, Ashia C, and Jordan, Michael I. Streaming variational Bayes. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1727–1735, 2013.

Csató, Lehel. *Gaussian Processes – Iterative Sparse Approximations*. PhD thesis, Aston University, 2002.

Dezfouli, Amir and Bonilla, Edwin V. Scalable inference for Gaussian process models with black-box likelihoods. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1414–1422, 2015.

Hensman, James, Fusi, Nicolo, and Lawrence, Neil D. Gaussian processes for big data. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 282–290, 2013.

Hensman, James, Matthews, Alexander G. D. G., and Ghahramani, Zoubin. Scalable variational Gaussian process classification. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

Hernández-Lobato, Daniel and Hernández-Lobato, José Miguel. Scalable Gaussian process classification via expectation propagation. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2016.

Matthews, Alexander G. D. G., Hensman, James, Turner, Richard E, and Ghahramani, Zoubin. On sparse variational methods and the Kullback-Leibler divergence between stochastic processes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2016.

Rasmussen, Carl E. and Williams, Christopher K. I. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

Seeger, Matthias. PAC-Bayesian generalisation error bounds for Gaussian process classification. *Journal of Machine Learning Research*, 2002.

Snelson, Edward and Ghahramani, Zoubin. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems (NIPS)*, pp. 1257–1264, 2006.

Titsias, Michalis K. Variational learning of inducing variables in sparse Gaussian processes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 567–574, 2009.