Abstract

This paper presents a novel distributed variational inference framework that unifies many parallel sparse Gaussian process regression (SGPR) models for scalable hyperparameter learning with big data. To achieve this, our framework exploits a structure of correlated noise process model that represents the observation noises as a finite realization of a high-order Gaussian Markov random process. By varying the Markov order and covariance function for the noise process model, different variational SGPR models result. This consequently allows the correlation structure of the noise process model to be characterized for which a particular variational SGPR model is optimal. We empirically evaluate the predictive performance and scalability of the distributed variational SGPR models unified by our framework on two real-world datasets.

1. Introduction

The rich class of Bayesian non-parametric Gaussian process (GP) models has recently established itself as a leading approach to probabilistic non-linear regression due to its capability of representing highly complex correlation structure underlying the data. However, the full-rank GP regression (FGPR) model incurs a cost of cubic time in the data size for computing the predictive distribution and in each iteration of gradient ascent to refine the estimate of its hyperparameters to improve the log-marginal likelihood (Rasmussen & Williams, 2006), hence limiting its usage to only small datasets in practice.

To boost its scalability, a wealth of sparse GPR (SGPR) models (Lázaro-Gredilla et al., 2010; Low et al., 2015; Quiñonero-Candela & Rasmussen, 2005; Snelson & Ghahramani, 2007) utilizing varying low-rank approximate representations of FGPR have been proposed, many of which are spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005) based on the notion of inducing variables (Section 3) such as the subset of regressors (SoR) (Smola & Bartlett, 2001), deterministic training conditional (DTC) (Seeger et al., 2003), fully independent training conditional (FITC) (Snelson & Ghahramani, 2005), fully independent conditional (FIC), partially independent training conditional (PITC) (Schwaighofer & Tresp, 2003), and partially independent conditional (PIC) (Snelson & Ghahramani, 2007) approximations. Consequently, they incur linear time in the data size, which is still prohibitively expensive for training with million-sized datasets. To scale up these SGPR models further for performing real-time predictions necessary in many time-critical applications and decision support systems (e.g., environmental sensing (Cao et al., 2013; Dolan et al., 2009; Ling et al., 2016; Low et al., 2008; 2009; 2011; 2012; Podnar et al., 2010; Zhang et al., 2016), traffic monitoring (Chen et al., 2012; 2013b; 2015; Hoang et al., 2014a;b; Low et al., 2014a;b; Ouyang et al., 2014; Xu et al., 2014; Yu et al., 2012)), a number of these models have been parallelized (e.g., FITC, FIC, PITC, and PIC (Chen et al., 2013a), and low-rank-cum-Markov approximation (LMA) (Low et al., 2015) unifying a spectrum of SGPR models with PIC and FGPR at the two extremes), but the resulting parallel SGPR models do not readily extend to include hyperparameter learning. The work of Deisenroth & Ng (2015) has recently introduced a practical product-of-expert (PoE) paradigm for GP which imposes a factorized structure on the marginal likelihood that allows it to be optimized effectively in a parallel/distributed fashion.

However, the main criticism of the above approximation paradigms is their lack of a rigorous approximation since they do not require optimizing some loss criterion incurred by an approximation model (Titsias, 2009b). To resolve this, the work of Titsias (2009a) has introduced an alterna-
tive formulation of variational inference for DTC approximation that involves minimizing the Kullback-Leibler (KL) distance between the variational DTC approximation and the posterior distribution of some latent variables (including the inducing variables) induced by the FGPR model given the data/observations, or equivalently maximizing a lower bound of the log-marginal likelihood. Hyperparameter learning can then be achieved by maximizing this variational lower bound as a function of the hyperparameters. Its incurred time per iteration of gradient ascent is still linear in the data size but can be significantly reduced by parallelization on multiple distributed machines/cores, as demonstrated by Gal et al. (2014). Despite their theoretical rigor and scalability, it has been shown by Hoang et al. (2015) that DTC has utilized the most crude approximation among all SGPR models (except SoR) spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005), thus severely compromising its predictive performance. As such, it remains an open question whether efficient and scalable hyperparameter learning of more refined SGPR models (e.g., PIC, LMA) for big data can be achieved through distributed variational inference.

To address this question, we first observe that variational DTC (Titsias, 2009a) and its distributed variant (Gal et al., 2014) have implicitly assumed the observation noises to be independently distributed with constant variance, which is often violated in practice (Huizenga & Molenaar, 1995; Koochakzadeh et al., 2015; Rasmussen & Williams, 2006). This strong assumption has been relaxed slightly by Titsias (2009b) to that of input-dependent noise variance which allows a variational inference formulation for FITC approximation to be derived. This seems to suggest a possibility of deriving variational inference formulations for the more refined sparse GP approximations and, perhaps surprisingly, their distributed variants by exploiting more sophisticated noise process models such as those being used by existing GP works. Such GP works, however, suffer from poor scalability to big data: Notably, the work of Goldberg et al. (1997) has proposed a heteroscedastic GPR (HGPR) model that extends the FGPR model by representing the noise variance with a log-GP (in the original GP modeling the noise-free latent measurements), hence allowing it to vary across the input space; the observation noises remain independently distributed though. But, the exact HGPR model cannot be computed tractably while approximate HGPR models (Kersting et al., 2007; Lázaro-Gredilla & Titsias, 2011) still incur cubic time in the data size, thus scaling poorly to big data (i.e., million-sized datasets). This is similarly true for FGPR models (Murray-Smith & Girard, 2001; Rasmussen & Williams, 2006) that represent correlation of observation noises with an additional covariance function. Unfortunately, variational DTC (Titsias, 2009a), its distributed variant (Gal et al., 2014), and variational FITC (Titsias, 2009b) cannot readily accommodate such heteroscedastic or correlated noise process models without sacrificing their time efficiency. So, the key challenge remains in being able to specify some structure of the noise process model that can be exploited for efficient and scalable hyperparameter learning of more refined SGPR models (e.g., PITC, PIC, LMA) through distributed variational inference, which is the focus of our work here.

To tackle this challenge, this paper presents a novel variational inference framework (Section 3) for deriving sparse GP approximations to a new FGPR model with observation noises that vary as a finite realization of a high-order Gaussian Markov random process (Section 2), thus enriching the expressiveness of HGPR models by correlating the noises across the input space. Interestingly, our proposed framework can unify many SGPR models via specific choices of the Markov order and covariance function for the noise process model (Section 4), which include variational DTC and FITC (Titsias, 2009a,b) and the more refined PITC, PIC, and LMA. This then enables the characterization of the correlation structure of the noise process model for which a particular sparse GP approximation is (variationally) optimal and explains why PIC and LMA tend to outperform DTC and FITC in practice despite not being characterized as optimal when independently distributed observation noises are assumed. More importantly, our framework is amenable to parallelization by distributing its computational load of hyperparameter learning on multiple machines/cores (Section 5), hence reducing its incurred linear time per iteration of gradient ascent by a factor close to the number of machines/cores. We empirically evaluate the predictive performance and scalability of the distributed variational SGPR models (e.g., state-of-the-art distributed variational DTC (Gal et al., 2014)) unified by our framework on two real-world datasets (Section 6).

2. Gaussian Processes with Correlated Noises

Let \( \mathcal{X} \) be a set representing the input domain such that each \( d \)-dimensional input feature vector \( \mathbf{x} \in \mathcal{X} \) is associated with a latent output variable \( f_{\mathbf{x}} \) and its corresponding noisy output \( y_{\mathbf{x}} \triangleq f_{\mathbf{x}} + \epsilon_{\mathbf{x}} \) differing by an additive noise \( \epsilon_{\mathbf{x}} \). Let \( \{ f_{\mathbf{x}} \}_{\mathbf{x} \in \mathcal{X}} \) denote a Gaussian process (GP), that is, every fi-

---

\( 1 \)The work of Campbell et al. (2015) has separately developed a distributed variational inference framework for Bayesian non-parametric models that are limited to only clustering processes (e.g., Dirichlet, Pitman-Yor, and their variants) not including GPs. 

\( 2 \)Such a characterization, which is important to many real-world applications of GP involving different noise structures, cannot be realized from the unifying framework of Hoang et al. (2015) relying on reverse variational inference to obtain the variational lower bound for a SGPR model. Furthermore, it is unclear or at least non-trivial to determine whether it is amenable to parallelization for learning the hyperparameters of LMA which does not meet its assumed decomposability conditions.
nite subset of \( \{f_x\}_{x \in X} \) follows a multivariate Gaussian distribution. Then, the GP is fully specified by its prior mean \( \mu_x \) and covariance \( k_{xx'} \triangleq \text{cov}[f_x, f_{x'}] \) for all \( x, x' \in X \), the latter of which can be defined, for example, by the widely-used squared exponential covariance function \( k_{xx'} \triangleq \exp(-0.5(x - x')^T \Lambda^{-2}(x - x')) \) where \( \Lambda \triangleq \text{diag}[\epsilon_1, \ldots, \epsilon_d] \) and \( \epsilon_k^2 \) are its length-scale and signal variance hyperparameters, respectively. Similarly, let \( \{\epsilon_x\}_{x \in X} \) denote another GP with prior mean \( \mathbb{E}[\epsilon_x] = 0 \) and covariance \( k_{\epsilon\epsilon'} \triangleq \text{cov}[\epsilon_x, \epsilon_{x'}] \) for all \( x, x' \in X \), the latter of which is defined by a covariance function like that used for \( k_{xx'} \) (albeit with different hyperparameter values).

Supposing a column vector \( y_D \triangleq (y_x)_{x \in D}^T \) of noisy outputs is observed for some set \( D \subset X \) of training inputs, a FGPR model with correlated observation noises (Murray-Smith & Girard, 2001; Rasmussen & Williams, 2006) can perform probabilistic regression by providing a GP posterior/predictive distribution \( p(f_u|\mu_u + K_U(K_D + S_{DD})^{-1}(y_D - \mu_D)) \) of the unobserved outputs \( f_u \triangleq (f_x)_{x \in U} \) for any set \( U \subset X \setminus D \) of test inputs where \( \mu_u(\mu_D) \) is a column vector with mean components \( \mu_x \) for all \( x \in U \) \((x \in D)\), \( K_U \) \((K_D + S_{DD})\) is a matrix with covariance components \( k_{xx'} \) for all \( x, x' \in U \) \((x, x' \in D)\), \( K_U \) \((K_D + S_{DD})\) is a matrix with covariance components \( k_{\epsilon\epsilon'} \) for all \( x, x' \in U \) \((x, x' \in D)\), \( K_U \) \((K_D + S_{DD})\) is a matrix with covariance components \( k_{\epsilon\epsilon'} \) for all \( x, x' \in U \) \((x, x' \in D)\), \( K_{\epsilon\epsilon} \) is a block comprising components of value 0. Though the matrix \( S_{DD} \) (and hence its submatrix \( S_{DD}^B \)) is dense, its structure (1) interestingly guarantees that \( P_{DD} \triangleq S_{DD}^{-1} \) is block-banded (Low et al., 2015). That is, if \(|i - j| > B\), then \( P_{DD} = 0 \). Such sparsity of \( P_{DD} \) is the main ingredient to improving the scalability of the FGPR model with correlated observation noises, as revealed in Section 3.

### 3. Variational Sparse GP Regression Models

This section introduces a novel variational inference framework for deriving sparse GP approximations to the FGPR model with correlated observation noises that vary as a finite realization of a \( B \)-th order Gaussian Markov random process (Section 2). Similar to the variational inference formulations for DTC and FITC approximations (Titsias, 2009a;b), we exploit a vector \( f_S \) of inducing output variables for some small set \( S \subset X \) of inducing inputs to construct sufficient statistics for \( y_D \) such that \( y_D \perp f_D | f_S \) (i.e., \( p(f_D, f_S | y_D) = p(f_D | f_S)p(f_S | y_D) \)). However, choosing \( S \) for which this conditional independence property holds may not be possible in practice. Instead, it is approximated by some (heuristic) choice of \( S \) as follows:

\[
p(f_D, f_S | y_D) \approx p(f_D | f_S) q(f_S)
\]

where \( q(f_S) \) is a free-form distribution to be optimized by minimizing the KL distance \( D_{KL}(q) \triangleq \text{KL}(p(f_D | f_S)q(f_S)\|p(f_D, f_S | y_D)) \) between \( p(f_D | f_S)q(f_S) \) and \( p(f_D, f_S | y_D) \) on either sides of (2) with respect to \( q(f_S) \) and the hyperparameters defining prior covariances \( k_{xx'} \) and \( k_{\epsilon\epsilon'} \) (Section 2). To do this, note that

\[
\log p(y_D) = \mathcal{L}(q, Z) + D_{KL}(q)
\]

where \( Z \) denotes a set of hyperparameters \(^3\) defining \( k_{xx'} \) and \( k_{\epsilon\epsilon'} \) and, as derived in Appendix A,

\[
\mathcal{L}(q, Z) \triangleq \mathbb{E}_{q(f_S)}[\log \mathcal{N}(y_D | \nu, S_{DD}) + \log(p(f_S) / q(f_S))] - 0.5\text{Tr}[P_{DD}^{-1} S_{DD}]
\]

where \( \nu \triangleq \text{mu} + K_{DS} K_{SS}^{-1} (f_S - \mu_S) \) and \( K_{SS} \triangleq K_{YY}^{-1} \) for all \( Y, Y' \subset X \). Supposing \( Z \) is fixed/know, since \( D_{KL}(q) \geq 0 \) and \( \log p(y_D) \) is independent of \( q(f_S) \), minimizing \( D_{KL}(q) \) is equivalent to maximizing \( \mathcal{L}(q, Z) \) which entails solving for the optimal choice of \( q(f_S) \) that satisfies \( \partial \mathcal{L}(q, Z) / \partial q = 0 \) as a PDF parameterized by \( Z \). As derived in Appendix B, this yields

\[
\log q(f_S) = \log [\mathcal{N}(y_D | \nu, S_{DD}) p(f_S)] + \text{const}
\]

where \( \text{const} \) is used to absorb all terms independent of \( f_S \). By completing the quadratic form for \( f_S \) (Appendix C), it can be derived from (5) that

\[
q(f_S) = \mathcal{N}(f_S | \mu_S + K_{SS} \Gamma_{SS}^{-1} V_{SD}, K_{SS} \Gamma_{SS}^{-1} K_{SS})
\]

\(^3\)All the terms in (3) and the equations to follow depend on \( Z \).

To ease notational clutter, their dependence on \( Z \) are omitted from their expressions, unless otherwise needed for clarity reasons.
where \( \Gamma_{SS} \triangleq K_{SS} + K_{SD}P_{DD}K_{DS}, \) \( P_{DD} = S_{DD}^{-1}, \) and \( V_{SD} \triangleq K_{SD}P_{DD}(y_d - \mu_d). \) By plugging (6) into (4), \( R(Z) \triangleq \max_q \mathcal{L}(q, Z) \) can be derived analytically. Then, maximizing \( R(Z) \) with respect to \( Z \) gives the optimal hyperparameters\(^4\). The details of this maximization are deferred to Section 5 for the sake of clarity. In the rest of this section, we will instead focus on deriving an equivalent reformulation of \( q(f_S) \) (6) that can be computed efficiently in linear time in the data size \( |D| \) since computing \( q(f_S) \) directly using (6) generally incurs \( \mathcal{O}(|D|^3) \) time. Moreover, its computation can be distributed among parallel machines/cores to achieve scalability, as described next.

Our first result (Lemma 1) below derives the structure of each non-zero constituent block \( P_{DD,i} \) of the \( B \)-block banded matrix \( P_{DD} \) (i.e., \( |i - j| \leq B \)), which is the key ingredient to be exploited for computing \( q(f_S) \) (6) efficiently and scalably (Theorem 1). Note that it suffices to simply compute \( P_{DD,i} \) for \( i \leq j \leq i_B^+ \) because \( P_{DD} \) is symmetric (i.e., \( P_{DD,i} = P_{DD,i}^T \) for \( i < j \leq j_B^+ \)) and \( B \)-block banded (i.e., \( P_{DD,i} = 0 \) for \( |i - j| > B \)) (Section 2):

**Lemma 1** Let

\[
G^k = \begin{bmatrix}
G^k_{D_kD_k} & G^k_{D_kD_kD_k} \\
G_{D_kD_kD_k} & G_{D_kD_kD_kD_k}
\end{bmatrix} \triangleq \begin{bmatrix}
S_{D_kD_k} & S_{D_kD_kD_k} \\
S_{D_kD_kD_k} & S_{D_kD_kD_kD_k}
\end{bmatrix}^{-1}
\]

for \( k = 1, \ldots, M \). Then, for \( i, j = 1, \ldots, M \) such that \( i \leq j \leq i_B^+ \) and \( j_B^+ \triangleq \max(1, 1 - B), \)

\[
P_{DD,i} = \sum_{i' = i_B^+}^{j_B^+} G^k_{D_kD_k}G^k_{D_kD_kD_k}G^k_{D_kD_kD_k}G^k_{D_kD_kD_k}.
\]

See Appendix D for its proof. From Lemma 1, constructing \( G^k \) of size \( \mathcal{O}(B^3|D|/M) = \mathcal{O}(B^4|S|^3) \) incurs \( \mathcal{O}(B^3|S|^3) \) time. \( G^1, \ldots, G^M \) can be computed independently and hence in parallel on \( M \) distributed machines/cores; otherwise, they can be constructed sequentially in \( \mathcal{O}(|D|^2B^3)^2 = \mathcal{O}(MB^3|S|^3) \) time.

Our next result exploits Lemma 1 and the \( B \)-block-banded structure of \( P_{DD} \) for decomposing \( \Gamma_{SS} \) and \( V_{SD} \) in (6) into linear sums of independent terms whose computations can therefore be distributed among parallel machines/cores:

**Theorem 1** Let \( B^+(k) \triangleq \{k, k + 1, \ldots, k_B^+\}. \) Then,

\[
\Gamma_{SS} = K_{SS} + \sum_{k=1}^M \beta_k \quad \text{and} \quad V_{SD} = \sum_{k=1}^M \alpha_k
\]

\[\alpha_k \triangleq \sum_{i,j \in B^+(k)} K_{SD,i}G^k_{D_kD_k}G^k_{D_kD_kD_k}G^k_{D_kD_kD_kD_k}(y_d,i - \mu_d), \quad \beta_k \triangleq \sum_{i,j \in B^+(k)} K_{SD,i}G^k_{D_kD_k}G^k_{D_kD_kD_k}G^k_{D_kD_kD_kD_k}K_{DS,j}.
\]

Its proof is in Appendix E. From (8), \( \alpha_1, \ldots, \alpha_M \) and \( \beta_1, \ldots, \beta_M \) can again be computed independently and hence in parallel in \( \mathcal{O}(|B^+(k)|^2|S|^3) = \mathcal{O}(B^2|S|^3) \) time on \( M \) distributed machines/cores; alternatively, they can be computed sequentially in \( \mathcal{O}(|\{D\}|^2B^3)^2 = \mathcal{O}(MB^3|S|^3) \) time. \( \Gamma_{SS} \) and \( V_{SD} \) can then be constructed in \( \mathcal{O}(M|S|^2) \) time and computing \( q(f_S) \) (6) in turn incurs \( \mathcal{O}(|S|^3) \) time. So, the overall time complexity of deriving \( q(f_S) \) (6) is linear in \( |D| \) which can be further reduced via parallelization by a factor close to the number of machines/cores.

**Remark** The efficiency in deriving \( q(f_S) \) (6) can be exploited for constructing a linear-time approximation \( q(f_M, y_D) \) to the GP predictive distribution \( p(f_M, y_D) \) for any subset \( \mathcal{U}_k = Y_k \setminus D_k \) of test inputs and its distributed variant, the details of which are not needed to understand our distributed variational inference framework for hyperparameter learning in Section 5 (but required for predictions in our experiments in Section 6) and hence deferred to Appendices F and G, respectively. By varying the choices of the Markov order \( B \), the covariance function \( k_{xx'} \) for the noise process model, the number \( M \) of data partitions, and the approximation \( q(f_M, f_S, y_D) \) to the test conditional \( p(f_M, y_D) \), they recover the predictive distribution of various SGPR models spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005) (i.e., SoR, DTC, FITC, FIC, PIFC, PIC) and LMA, as detailed in Section 4.4.1.

### 4. Unifying Sparse GP Regression Models

#### 4.1. SGPR models spanned by unifying view of Quiñonero-Candela & Rasmussen (2005)

We will first demonstrate how our variational inference framework (Section 3) can unify SoR, DTC, FITC, FIC, PIFC, PIC. As discussed in Appendix F, it suffices to show how the covariance function \( k_{xx'} \) for the noise process model and the Markov order \( B \) can be set such that the resulting variationally optimal distribution \( q(f_S) \) (6) coincides with that of these SGPR models. Let us consider the following covariance function for the noise process model:

\[
k_{xx'} \triangleq k_{xx'} - k_{xS}k_{Sx'}^{-1}k_{Sx} + \sigma_n^2I(x = x')
\]

where \( \sigma_n^2 \) is a noise variance hyperparameter. It follows that \( k_{xx'}^{DD} = K_{DD} - K_{DS}K_{SS}^{-1}K_{SD} + \sigma_n^2I \). Then, imposing the \( B \)-th order Markov property on the observation noise process \( \{x_k\}_{k \in D} \) through (1) with \( B = 0 \) gives \( S_{DD} = \text{blkdiag}(K_{DD} - Q_{DD}) + \sigma_n^2I \) which implies \( S_{DD,i,j} = I(i = j)(K_{DD,i,j} - Q_{DD} + \sigma_n^2I) \).

Plugging this expression of \( S_{DD} \) into (6) yields the same \( q(f_S) \) induced by PIC, as detailed in (Hoang et al., 2015)**.\(^6\)

\(^4\)The formulation described thus far is reminiscent of variational expectation-maximization (EM) (Wainwright & Jordan, 2008) whose approximate E step involves maximizing the variational lower bound \( \mathcal{L}(q, Z) \) with respect to \( q \) given optimal \( Z \) from \( M \) step while its M step follows exactly that of EM by maximizing \( \mathcal{L}(q, Z) \) with respect to \( Z \) given optimal \( q \) from E step. The implication is that, like variational EM, variational DTC and FITC (Titsias, 2009a,b), and distributed variational DTC (Gal et al., 2014), though the log-marginal likelihood log \( p(y_D) \) does not necessarily increase in each iteration, our formulation has an attractive interpretation of maximizing its lower bound.

\(^3\)When \( B = 0, G^k = S_{DD,i,k}^{-1} \) for \( k = 1, \ldots, M \).

\(^5\)We choose the size of \( S \) such that \( |D_i| = |D|/M = \mathcal{O}(|S|). \)

\(^6\)The work of Hoang et al. (2015) assumes a GP with zero prior mean which is equivalent to setting \( \mu_S = \mu_D = 0 \) in (6).
Furthermore, if \(|D_i| = 1\) for \(i = 1, \ldots, M\) (i.e., \(M = |D|\)), then \(S_{DD} = \text{diag}[K_{DD} - Q_{DD}] + \sigma_n^2 I\) which can be plugged into (6) to recover the same \(q(f_s)\) induced by FIC. Supposing \(k_{xx}^2 \triangleq \sigma_n^2 I(x = x')\) instead of using (9), \(S_{DD} = \sigma_n^2 I\) that can be plugged into (6) to recover the same \(q(f_s)\) induced by DTC. Finally, the predictive distributions \(q(fu_i|yp)\) of the SGPR models spanned by the unifying view of Quiñoneiro-Candela & Rasmussen (2005) such as PIC, FIC and DTC can be recovered by integrating the resulting \(q(f_s)\) with their approximations \(q(fu_i|fs, yp)\) to the test conditional \(p(fu_i|fs; yp)\), as discussed in Appendix F. We omit details of unifying PITC, FITC, and SoR with our framework since they induce the same \(q(f_s)\) as that of PIC, FIC and DTC, respectively.

4.2. Low-rank-cum-Markov approximation (LMA)

To unify LMA with our variational inference framework, we have to derive its induced \(q(f_s)\) since it is not explicitly given in (Low et al., 2015). To do this, let us first derive an equivalent reformulation of the predictive distribution of the FGPR model with independently distributed noises of constant variance, as shown in Appendix F:

\[
p(fu_i|fs; yp) \propto N(fu_i|\mu_s, K_{ss}^{-1}V_{SD}, \sigma_n^2 K_{s}\Gamma_{SS}^{-1}K_{SS})
\]

where \(\Gamma_{SS} \triangleq K_{SS} + K_{SD}R_{DD}^{-1}K_{PS} + V_{SD} = K_{DD}R_{DD}^{-1}(yp - \mu_s)\) such that \(R_{DD} \triangleq K_{DD} - Q_{DD} + \sigma_n^2 I\). But, computing \(p(fu_i|yp)\) generally incurs \(O(|D|^3)\) time. To reduce to linear time and achieve further scalability via parallelization, the work of Low et al. (2015) has proposed LMA by approximating \(R_{DD}\) with \(S_{DD}\) (1) based on the covariance function in (9), which interestingly induces the same \(q(f_s)\) and predictive distribution (Appendix I) as that produced by our variational inference framework (see, respectively, (6) in Section 3 and (48) in Appendix F, the latter of which relies on a specific choice of approximation \(q(fu_i|fs, yp)\) (47) to the test conditional \(p(fu_i|fs; yp)\)). So, LMA is unified with our framework.

4.3. Key insight: Different variational SGPR models evolve from varying noise correlation structures

It is well-known that, among existing SGPR models, DTC and FITC are deemed variationally optimal as they minimize the KL distance to a FGPR model with independently distributed noises (i.e., \(S_{DD} = \sigma_n^2 I\) (Titsias, 2009a) or \(S_{DD} = \text{diag}[K_{DD} - Q_{DD}] + \sigma_n^2 I\) (Titsias, 2009b)). However, empirical results in the existing literature show that FITC and DTC often predict more poorly than PIC (Hoang et al., 2015), and LMA (Low et al., 2015), which are usually explained by empirically inspecting how well these SGPR models approximate the prior covariance matrix of the FGPR model (Low et al., 2015; Quiñoneiro-Candela & Rasmussen, 2005; Snelson, 2007). But, to the best of our knowledge, it has not been explicitly explained why SGPR models like PIC and LMA, despite being able to provide more refined approximations of the prior covariance matrix of the FGPR model, do not minimize their KL distance to the FGPR model.

Our unification results (Sections 4.1 and 4.2) have explained this by giving a theoretical justification based on the correlation structure of the noise process model: Specifically, FITC and DTC can only minimize their KL distances to a FGPR model under the assumption of independently distributed noises, which is often violated in many real-world datasets where observation noises tend to be correlated (Huizenga & Molenaar, 1995; Kooshakzadeh et al., 2015; Rasmussen & Williams, 2006). However, they do not necessarily minimize their KL distances to a FGPR model in such cases. In light of this fact, our unification results reveal that different SGPR models minimize their KL distances to a FGPR model under varying assumptions of correlation structure of the noise process model: For example, Section 4.1 (4.2) shows that when our variational inference framework minimizes the KL distance to a FGPR model (Section 3) under the assumption of \(B = 0\) (\(B > 0\)) and the covariance function \(k_{xx}^2\) in (9) for the noise process model, it recovers the same \(q(f_s)\) and predictive distribution \(q(fu_i|yp)\) induced by PIC (LMA).

5. Distributed Variational Inference for Hyperparameter Learning

Recall from Section 3 that plugging the derived \(q(f_s)\) (6) into the variational lower bound \(L(q, Z)\) (4) gives

\[
R(Z) = \max_q L(q, Z) = \log N(yp|\mu_p, S_{DD} + Q_{DD}) - 0.5\text{Tr}[S_{DD}^{-1}(K_{DD} - Q_{DD})]
\]

where the last equality is derived in Appendix J. Hyperparameter learning then involves iteratively refining the estimate of hyperparameters \(Z\) via gradient ascent to improve the value of \(R(Z)\). Its time complexity depends on the efficiency of computing the gradient \(\partial R/\partial Z\) per iteration of gradient ascent. As shown in Appendix K, differentiating both sides of (10) with respect to \(z \in Z\) yields

\[
\partial R/\partial z = -\frac{1}{2} \text{Tr}\left[\partial P_{DD}/\partial z W_{DD} + P_{DD}/\partial z W_{DD} - S_{DD}/\partial z P_{DD}\right] - \frac{1}{2} \text{Tr}\left[\Gamma_{SS}/\partial z - K_{SS}/\partial z \right] + 2 \text{Tr}\left[V_{SD}/\partial z \Gamma_{SS}^{-1}V_{SD}\right]
\]

where \(W_{DD} \triangleq K_{DD} - Q_{DD} + (yp - \mu_p)(yp - \mu_p)^\top\). However, computing \(\partial R/\partial z\) directly using (11) is prohibitively expensive as it involves computing the matrix derivative \(\partial P_{DD}/\partial z = -P_{DD} (\partial S_{DD}/\partial z) P_{DD}\) which

\[\sqrt{\text{in}}\]
A Distributed Variational Inference Framework for Unifying Parallel Sparse Gaussian Process Regression Models

... generally incurs $O(|D|^3)$ time. In the rest of this section, we will exploit the $B$-block-banded structure of $P_{PTD}$ (Section 2) for deriving an efficient reformulation of $\partial R/\partial z$ (11) that can be computed in linear time in the data size $|D|$. More importantly, it is amenable to parallelization on distributed machines/cores by constructing and communicating summaries of data clusters to be described next:

**Definition 1 (B-th order Markov Cluster)** A B-th order Markov cluster $k$ is defined as a tuple $P_k \triangleq (\alpha_k, \beta_k, \{\alpha_k^{-1}, \beta_k^{-1}, \psi_k, \psi_k^z\}_{z \in \mathcal{Z}})$ where $\alpha_k$ and $\beta_k$ are previously defined in Theorem 1, $\alpha_k \triangleq \delta \alpha_k/\delta z$, $\beta_k \triangleq \delta \beta_k/\delta z$, and $\psi_k \triangleq \delta \psi_k/\delta z$.

**Definition 2 (Local Summary)** The local summary of a B-th order Markov cluster $k$ is defined as a tuple $P_k \triangleq (\alpha_k, \beta_k, \{\alpha_k^{-1}, \beta_k^{-1}, \psi_k, \psi_k^z\}_{z \in \mathcal{Z}})$ where $\alpha_k$ and $\beta_k$ are previously defined in Theorem 1, $\alpha_k \triangleq \delta \alpha_k/\delta z$, $\beta_k \triangleq \delta \beta_k/\delta z$, and $\psi_k \triangleq \delta \psi_k/\delta z$.

The local summaries $P_1, \ldots, P_M$ can be computed independently and hence in parallel on M distributed machines/cores: To construct local summary $P_k$, each machine/core $k$ only needs access to its Markov cluster $M_k$, the corresponding noisy outputs $y_{M_k}$, and a common set $S_\alpha$ of inducing inputs known to all machines/cores. Evaluating the derivative terms in local summary $P_k$ essentially requires computing terms such as $\partial G_{k,i}^Z/\partial z$ for $i \in B^+(k)$. To compute the latter terms, we exploit the result of Lemma 1 for computing $\partial G_k^Z/\partial z$ in terms of $\partial S_{D_k}^Z/\partial z$, $\partial D_k^Z/\partial z$, and $\partial S_{D_k}^Z/\partial z$. For each Markov cluster $M_k$, computing $G_k$ (and hence $\partial G_k^Z/\partial z$) incurs only $O(3^3|S|^3)$ time, as discussed in the paragraph after Lemma 1. Its corresponding summary $P_k$ can then be computed in $O(3^2|S|^3)$ time. So, the overall time complexity of computing local summary $P_k$ of Markov cluster $M_k$ in each parallel machine/core $k$ is $O(3^3|S|^3)$ which is independent of the data size $|D|$. Alternatively, $P_1, \ldots, P_M$ can be constructed sequentially in $O(|D|3^3|S|^3) = O(M3^3|S|^3)$ time that is linear in $|D|$. Every machine/core $k$ then communicates the local summary $P_k$ of its assigned Markov cluster $M_k$ to a central machine that will assimilate these local summaries into a global summary defined as follows:

**Definition 3 (Global Summary)** The global summary is defined as a tuple $P_\ast \triangleq (\alpha, \beta, \{\alpha^z, \beta^z, \psi^z\}_{z \in \mathcal{Z}})$ where

\[
\alpha \triangleq \sum_{k=1}^{M} \alpha_k, \quad \beta \triangleq K_{SS} + \sum_{k=1}^{M} \beta_k, \quad \alpha^z \triangleq \sum_{k=1}^{M} \alpha_k^z,
\]

\[
\beta^z \triangleq \frac{\partial K_{SS}}{\partial z} + \sum_{k=1}^{M} \beta_k^z, \quad \text{and} \quad \psi^z \triangleq \sum_{k=1}^{M} \psi_k^z - \psi_k^z.
\]

Our main result to follow presents an efficient reformulation of $\partial R/\partial z$ (11) by exploiting the global summary $P_\ast$:

**Theorem 2** $\partial R/\partial z$ (11) can be re-expressed in terms of global summary $P_\ast = (\alpha, \beta, \{\alpha^z, \beta^z, \psi^z\}_{z \in \mathcal{Z}})$ as follows:

\[
\frac{\partial R}{\partial z} = \frac{1}{2} \mathbf{Tr}\left[ K_{SS}^{-1} \frac{\partial K_{SS}}{\partial z} \right] + \alpha^{zT} \beta^{-1} \alpha - \frac{1}{2} \mathbf{Tr}\left[ \beta^{-1} \beta^z - \frac{1}{2} \alpha^{zT} \beta^{-1} \beta^{zT} \right] - \frac{1}{2} \alpha^{zT} \beta^{-1} \beta^{zT} \beta^{-1} \alpha - \frac{1}{2} \phi^z.
\]

(12)

Its proof is in Appendix L. Since $\alpha$ and $\alpha^z$ are column vectors of size $|S|$, $\beta$ and $\beta^z$ are matrices of size $|S|$ by $|S|$, and $\phi^z$ is a scalar, computing $\partial R/\partial z$ using (12) only incurs $O(|S|^3)$ time given the global summary $P_\ast$. So, the overall time complexity of computing $\partial R/\partial z$ (12) in a distributed manner on $M$ parallel machines/cores (sequentially) is still $O(M3^3|S|^3)$.

### 6. Experiments and Discussion

This section empirically evaluates the predictive performance and scalability of distributed variational SGPR models unified by our framework (Section 5) such as the distributed variants of PIC and LMA and the current state-of-the-art distributed variant of DTC (Gal et al., 2014), which we respectively call $d$PIC, $d$LMA, and $d$DTC.

(a) The AIMPEAK dataset (Chen et al., 2013a) contains 41850 observations of traffic speeds (km/h) along 775 road segments of an urban road network during morning peak hours on April 20, 2011. Each observation comprises a 5-dimensional input vector featuring the length, number of lanes, speed limit, direction of the road segment as well as its recording time which is discretized into 54 five-minute time slots. The outputs correspond to the traffic speeds.

(b) The AIRLINE dataset (Hensman et al., 2013; Hoang et al., 2015) contains 2055733 information records of commercial flights in the USA from January to April 2008. The input denotes a 8-dimensional feature vector of the age of the aircraft (year), travel distance (km), airtime, departure and arrival time (min), day of the week, day of the month, and month. The output is the delay time (min) of the flight.

Both datasets are modeled using GP with correlated noises (Section 2) whose prior covariance matrix is defined using the squared exponential covariance function described in

---

---

---
Section 2. For \( d \text{PIC} \) and \( d \text{LMA} \) models, the prior covariance of the noise process is constructed using (1) (respectively, with Markov order \( B = 0 \) and \( 1 \)) and the covariance function in (9). Likewise, for \( d \text{TDC} \), the prior covariance of the noise process is constructed using (1) with \( B = 0 \) and covariance function \( k^{\epsilon}_x \Leftrightarrow \sigma_n^2 I \). Both the training and test data are then partitioned evenly into \( M \) blocks using \( k \)-means (i.e., \( k = M \)). All experiments are run on a Linux system with Intel® Xeon® E5-2670 at 2.6GHz with 96 GB memory and 32 processing cores. Our distributed variational SGPRs are implemented using Armadillo linear algebra library for C++ (Sanderson, 2010). For each tested model, we report the (a) root mean square error (RMSE), \( \sqrt{\frac{1}{|\mathcal{D}|} \sum_{x \in \mathcal{D}} (y_x - \hat{y}_x)^2} \), of its predictions \( \{\hat{y}_x\}_{x \in \mathcal{D}} \), (b) timing vs. no. of iterations, and (c) training time vs. no. of parallel cores. The results for each model are evaluated on 5% of the dataset with respect to its best configuration (e.g., learning rate, no. of inducing inputs, etc.).

**AIMPEAK Dataset.** We randomly remove 50 data points from the original dataset so that the experimented data can be partitioned unevenly into \( M = 100 \) blocks. The empirical results, observations, and analysis are described below:

![Graphs of RMSEs and total incurred times of \( d \text{TDC}, d \text{PIC}, \) and \( d \text{LMA} \) vs. number \( M' \) of parallel cores for AIMPEAK dataset.](image)

(a) Figs. 1 and 2 show results of RMSEs, incurred times, and parallel efficiencies of \( d \text{PIC}, d \text{TDC}, \) and \( d \text{LMA} \) averaged over 5 random instances with varying number \( t \) of iterations. In particular, it can be observed from both Figs. 1 and 2 that the RMSEs of all tested methods decrease rapidly (see Figs 1a, 1c, and 1e) while their total incurred training times only increase linearly in the number \( t \) of iterations (see Fig. 2b), which shows the effectiveness of our distributed hyperparameter learning framework (Section 5) for these variational SGPR models.

(b) Fig. 1 also shows results of the total training times incurred by \( d \text{PIC}, d \text{LMA,} \) and \( d \text{TDC} \) over 10 iterations with varying number \( M' = 2, 4, 8, 16 \) of parallel cores. As expected, it can be observed from Figs. 1b, 1d, and 1f that the training times of the tested models decrease gradually when the number of computing cores increases, which corroborates our complexity analysis of distributed SGPR models in Section 5. More specifically, our experiment reveals that using \( M' = 16 \) cores, \( d \text{PIC} \) (116 seconds), \( d \text{LMA} \) (193 seconds), and \( d \text{TDC} \) (97 seconds) can achieve speedups (i.e., parallel efficiencies) of 5.5 to 7.2 over their centralized counterparts (i.e., \( M' = 1 \)); PIC (836 seconds), LMA (1266 seconds), and TDC (537 seconds).

(c) Fig. 2b further shows that \( d \text{TDC} \) consistently incurs less training time than both \( d \text{PIC} \) and \( d \text{LMA} \) with varying number \( t \) of learning iterations. This is expected because the primary cost of computing a local summary for \( d \text{TDC} \), which involves computing \( G^k \) in Lemma 1, is constant\(^{10}\) while that of \( d \text{PIC} \) and \( d \text{LMA} \) grows cubically in the block size \( |\mathcal{D}|/M \). On the other hand, Fig. 2a, however, reveals that the RMSEs achieved by \( d \text{PIC} \) (3.75) and \( d \text{LMA} \) (3.54) are both significantly lower than that achieved by \( d \text{TDC} \) (10.70). This inferior performance of \( d \text{TDC} \) is also expected because of its more restrictive assumption of deterministic relation between the training and inducing outputs (Hoang et al., 2015).

**AIRLINE Dataset.** We extract 2M data points from the original dataset for experiment to guarantee an even par-

\(^{10}\)For \( d \text{TDC} \), since \( S_{DD} = \sigma_n^2 I \), it follows straightforwardly that \( G^k = \sigma_n^{-2} I \) which can be constructed in constant time.
Figure 3. Graphs of RMSEs achieved by (a) dDTC, (c) dPIC, and (e) dLMA vs. number $t$ of iterations, and graphs of the total training times incurred by (b) dDTC, (d) dPIC, and (f) dLMA vs. number $M'$ of parallel cores for the AIRLINE dataset.

Figure 4. Graphs of (a) RMSEs and (b) total incurred times of $d$DTC, $d$PIC, and $d$LMA vs. number $t$ of iterations with $M' = 32$ computing cores and $M = 2000$ blocks for AIRLINE dataset.

dPIC, are further evaluated on two benchmark settings of the AIRLINE dataset to compare with the previously best reported RMSE results of the existing state-of-the-art distributed methods such as Dist-VGP (Gal et al., 2014) and rBCM (Deisenroth & Ng, 2015). All results are reported in Table 1 below, which essentially shows that $d$PIC and $d$LMA significantly outperform Dist-VGP and rBCM in both settings. Notably, $d$LMA (16.50 and 13.20) manages to reduce the previously best reported RMSEs (27.10 and 34.40) by 39.11% and 61.62%, respectively.

<table>
<thead>
<tr>
<th></th>
<th>Dist-VGP</th>
<th>rBCM</th>
<th>dPIC</th>
<th>dLMA</th>
</tr>
</thead>
<tbody>
<tr>
<td>700K/100K</td>
<td>33.00</td>
<td>27.10</td>
<td>21.09</td>
<td>16.50</td>
</tr>
<tr>
<td>2M/100K</td>
<td>35.30</td>
<td>34.40</td>
<td>17.88</td>
<td>13.20</td>
</tr>
</tbody>
</table>

Table 1. RMSEs achieved by existing distributed parallel methods on two standard benchmark settings (training/test data sizes) of AIRLINE dataset: (a) 700K/100K and (b) 2M/100K. The results (RMSEs) of Dist-VGP (Gal et al., 2014) and rBCM (Deisenroth & Ng, 2015) are reported from their respective papers.

7. Conclusion

This paper describes a novel distributed variational inference framework that unifies many parallel SGPR models (e.g., DTC, FITC, FIC, PITC, PIC, LMA) for scalable hyperparameter learning, consequently reducing their incurred linear time per iteration of gradient ascent significantly. To achieve this, our framework exploits a structure of correlated noise process model that represents the observation noises as a finite realization of a $B$-th order Gaussian Markov random process. By varying the Markov order and covariance function for the noise process model, different variational SGPR models result. This consequently allows the correlation structure of the noise process model to be characterized for which a particular variational SGPR model is optimal; in other words, different SGPR models minimize their KL distance to a FGPR model under varying characteristics of the noise correlation structure. Empirical evaluation on two real-world datasets show that our proposed framework can achieve significantly better predictive performance than the state-of-the-art distributed variational DTC (Gal et al., 2014) and distributed GPs (Deisenroth & Ng, 2015) while preserving scalability to big data.
Acknowledgments. This research is supported by the National Research Foundation, Prime Minister’s Office, Singapore under its International Research Centre in Singapore Funding Initiative.

References


A Distributed Variational Inference Framework for Unifying Parallel Sparse Gaussian Process Regression Models


A. Derivation of (4)

For all \( f_S \) and \( f_D \),

\[
p(y_D) = p(f_S, f_D, y_D) / p(f_S, f_D | y_D)
\]

which directly implies

\[
\log p(y_D) = \log(p(f_S, f_D, y_D) / p(f_S, f_D | y_D)).
\]

Let \( q(f_S, f_D) \) be an arbitrary probability density function (PDF) over \( (f_S, f_D) \). Integrating both sides of the above equation with respect to \( q(f_S, f_D) \) gives

\[
\log p(y_D) = \int q(f_S, f_D) \log \frac{p(f_S, f_D, y_D)}{p(f_S, f_D | y_D)} \, df_S df_D.
\]  
\[
(13)
\]

Then, using the identity \( \log(ab) = \log a + \log b \), it follows that

\[
\log \frac{p(f_S, f_D, y_D)}{p(f_S, f_D | y_D)} = \log \frac{q(f_S, f_D)}{p(f_S, f_D | y_D)} + \log \frac{q(f_S, f_D)}{p(f_S, f_D | y_D)}
\]

which is plugged into (13) to yield

\[
\log p(y_D) = KL(q(f_S, f_D)||p(f_S, f_D | y_D)) + \mathcal{L}(q, Z)
\]

where \( Z \) denotes the set of hyperparameters defining prior covariances \( k_{xx} \) and \( k'_{xx} \) (Section 2) and

\[
\mathcal{L}(q, Z) = \int q(f_S, f_D) \log \frac{p(f_S, f_D, y_D)}{p(f_S, f_D | y_D)} \, df_S df_D.
\]  
\[
(16)
\]

Choosing \( q(f_S, f_D) \) \( \propto p(f_S) \) \( q(f_D) \) and factorizing \( p(f_S, f_D, y_D) = p(y_D | f_D) p(f_D | s_S) p(s_S) \),

\[
\mathcal{L}(q, Z) = \int q(f_S) h(f_S) \, df_S
\]

where

\[
h(f_S) \triangleq \mathbb{E}_{f_D | f_S} \left[p \left( \log p(y_D | f_D) \right) \right] + \log \frac{q(f_S)}{p(f_S)}.
\]  
\[
(18)
\]

Then, to derive (4), note that, by definition, \( p(y_D | f_D) = \mathbb{N} (y_D | f_D, S_{DD}) \) and \( p(f_D | f_S) = \mathbb{N} (f_D | \nu, \mathbf{K}_{DD} - Q_{DD}) \) with \( \nu \triangleq \mu_D + K_{DS} k^{-1}_{SS} (f_S - \mu_S) \) and \( Q_{DD} \triangleq K_{DS} k^{-1}_{SS} K_{SD} \). This implies

\[
\log p(y_D | f_D) = -\frac{1}{2} |D| \log(2\pi) - \frac{1}{2} \log |S_{DD}| - \frac{1}{2} (y_D - f_D)^\top S^{-1}_{DD} (y_D - f_D).
\]  
\[
(19)
\]

Taking expectations on both sides of (19) with respect to \( p(f_D | f_S) = \mathbb{N}(f_D | \nu, \mathbf{K}_{DD} - Q_{DD}) \) yields

\[
\mathbb{E}_{f_D | f_S} [\log p(y_D | f_D)] = -\frac{1}{2} |D| \log(2\pi) - \frac{1}{2} \log |S_{DD}| - \frac{1}{2} y_D^\top S^{-1}_{DD} y_D
\]

\[
+ y_D^\top S^{-1}_{DD} \mathbb{E}_{f_D | f_S} [f_D] - \frac{1}{2} \mathbb{E}_{f_D | f_S} [f_D] \mathbb{E}_{f_D | f_S} [f_D] \cdot
\]

Then, using the Gaussian identity \( \mathbb{E}_{f_D | f_S} [f_D^\top S^{-1}_{DD} f_D] = \text{Tr}[S^{-1}_{DD} (K_{DD} - Q_{DD})] + \nu^\top S^{-1}_{DD} \nu \), the above equation can be further simplified to

\[
\mathbb{E}_{f_D | f_S} [\log p(y_D | f_D)] = \log \mathbb{N}(y_D | \nu, S_{DD})
\]

\[
- \frac{1}{2} \text{Tr}[S^{-1}_{DD} (K_{DD} - Q_{DD})].
\]

Finally, plugging this into (17) and (18) results in (4).

B. Derivation of (5)

Supposing \( Z \) is fixed/known, optimizing \( q(f_S) \) to maximize \( \mathcal{L}(q, Z) \) can generally be achieved by setting the variational derivative \( \partial \mathcal{L}(q, Z) / \partial q = 0 \) and solving for \( q(f_S) \) subject to the constraint \( \int q(f_S) \, df_S = 1 \). Solving for \( q(f_S) \) under such a constraint appears rather tedious. So, without loss of generality, let us redefine \( q(f_S) \) in terms of an arbitrary, finitely integrable function \( w(f_S) \) as follows:

\[
q(f_S) \triangleq \frac{w(f_S)}{\int w(f_S) \, df_S} = C \cdot w(f_S)
\]  
\[
(20)
\]

where \( C \triangleq 1 / (\int w(f_S) \, df_S) \). Plugging (20) into (4), \( \mathcal{L}(q, Z) \) can be rewritten as

\[
\mathcal{L}(q, Z) = C \int w(f_S) (r(f_S) - \log w(f_S)) \, df_S
\]

\[
- \frac{1}{2} \text{Tr}[S^{-1}_{DD} (K_{DD} - Q_{DD})]
\]

\[
\triangleq \mathcal{L}(w, Z)
\]  
\[
(21)
\]

where \( r(f_S) \triangleq \log \mathbb{N}(y_D | \nu, S_{DD}) / C \). According to (21), \( \mathcal{L}(q, Z) \) can be optimized by deriving and solving \( \partial \mathcal{L}(w, Z) / \partial w = 0 \). To achieve this, let \( \phi(f_S) \) be an arbitrary function that vanishes on the boundary of the region of integration. Then, \( \partial \mathcal{L}(w, Z) / \partial w \) can be derived from the definition of functional derivative:

\[
\int \frac{\partial \mathcal{L}(w, Z)}{\partial w} \phi(f_S) \, df_S = \int \left[ \frac{dQ(\varepsilon)}{d\varepsilon} \right]_{\varepsilon=0} \, df_S
\]  
\[
(22)
\]

where

\[
Q(\varepsilon) \triangleq C (w(f_S) + \varepsilon \phi(f_S)) (r(f_S) - \log(w(f_S) + \varepsilon \phi(f_S))).
\]  
\[
(23)
\]

Intuitively, (23) characterizes the perturbation of \( Cw(f_S) (r(f_S) - \log w(f_S)) \) when a vanishingly small function \( \varepsilon \phi(f_S) \) is added to \( w(f_S) \). To derive the RHS of (22), \( dQ(\varepsilon)/d\varepsilon \) is evaluated at \( \varepsilon = 0 \):

\[
\left[ \frac{dQ(\varepsilon)}{d\varepsilon} \right]_{\varepsilon=0} = C (r(f_S) - \log w(f_S) - 1) \phi(f_S).
\]  
\[
(24)
\]

Since \( \phi(f_S) \) is an arbitrary function, plugging (24) into (22) gives

\[
\frac{\partial \mathcal{L}(w, Z)}{\partial w} = C (r(f_S) - \log w(f_S) - 1).
\]  
\[
(25)
\]
By setting the RHS of (25) to 0 and solving for \( w(f_S) \),
\[
    w(f_S) = \frac{1}{e} \exp(v(f_S)) = \frac{1}{e} \mathcal{N}(y_D|\nu, S_{DD}) p(f_S) .
\]  
(26)
By plugging (26) into (20),
\[
    q(f_S) = \frac{\mathcal{N}(y_D|\nu, S_{DD}) p(f_S)}{\int \mathcal{N}(y_D|\nu, S_{DD}) p(f_S) \, df_S} .
\]  
(27)
Finally, taking the logarithms on both sides of (27) gives
\[
    \log q(f_S) = \log[\mathcal{N}(y_D|\nu, S_{DD}) p(f_S)] + E
\]  
(28)
where \( E \triangleq -\log(\int \mathcal{N}(y_D|\nu, S_{DD}) p(f_S) \, df_S) \) clearly does not depend on \( f_S \) and can thus be considered a constant, thus yielding (5).

C. Derivation of (6)

To derive (6), (28) is first rewritten as a quadratic function of \( f_S \):
\[
    \log q(f_S) = -\frac{1}{2} |D| \log(2\pi) - \frac{1}{2} \log |S_{DD}|
    - \frac{1}{2} (y_D - \nu)^T S_{DD}^{-1} (y_D - \nu)
    - \frac{1}{2} |S| \log(2\pi) - \frac{1}{2} \log |K_{SS}|
    - \frac{1}{2} (f_S - \mu_S)^T K_{SS}^{-1} (f_S - \mu_S) + E
    - \frac{1}{2} f_S^T K_{SS}^{-1} f_S + f_S^T K_{SS}^{-1} \mu_S
    - \frac{1}{2} \nu^T S_{DD}^{-1} \nu + \nu^T S_{DD}^{-1} y_D + \text{const}
\]  
(29)
where \text{const} is used to absorb all terms independent of \( f_S \). Then, plugging the definition of \( \nu \triangleq \mu_D + K_{DS} K_{SS}^{-1} (f_S - \mu_S) \) into (29) gives
\[
    \log q(f_S) = -\frac{1}{2} f_S^T \Sigma_{SS|D}^{-1} f_S + f_S^T \Sigma_{SS|D}^{-1} \mu_{SS|D} + \text{const}
\]  
(30)
where
\[
    \Sigma_{SS|D} \triangleq K_{SS} \Gamma_{SS}^{-1} K_{SS}
\]  
(31)
and
\[
    \mu_{SS|D} \triangleq \mu_S + \Sigma_{SS|D} K_{SS}^{-1} K_{SD} S_{DD}^{-1} (y_D - \mu_D)
    = \mu_S + \Sigma_{SS|D} K_{SS}^{-1} K_{SD} P_{DD} (y_D - \mu_D) .
\]  
(32)
Consequently, \( q(f_S) = \mathcal{N}(f_S|\mu_{SS|D}, \Sigma_{SS|D}) \).

Finally, by plugging \( \Sigma_{SS|D} = K_{SS} \Gamma_{SS}^{-1} K_{SS} \) and \( V_{SD} = K_{SD} P_{DD} (y_D - \mu_D) \) into (32),
\[
    \mu_{SS|D} = \mu_S + K_{SS} \Gamma_{SS}^{-1} V_{SD} ,
\]  
(33)
thus yielding (6).

D. Proof of Lemma 1

Let \( U_{DD} \) denote the Cholesky factor of \( P_{DD} \). Then, it follows that \( U_{DD} \) is upper-triangular and \( P_{DD} = U_{DD}^T U_{DD} \).

Thus, for each constituent block \( P_{D_iD_j} \), that lies above or is one of the diagonal blocks of \( P_{DD} \) (i.e., \( i \leq j \)),
\[
    P_{D_iD_j} = \sum_{k=1}^{M} U_{D_kD_i}^T U_{D_kD_j} ,
\]  
(34)
where \( U_{D_kD_i} \) corresponds to the constituent block on the \( i \)-th row and \( k \)-th column of \( U_{DD} \). Since \( U_{DD} \) is upper-triangular, each summation term on the RHS of (34) is non-zero only if \( k \leq i \). So, for \( i, j = 1, \ldots, M \) such that \( i \leq j \), (34) can be rewritten as
\[
    P_{D_iD_j} = \sum_{k=1}^{i} U_{D_kD_i}^T U_{D_kD_j} .
\]  
(35)
Furthermore, by exploiting the fact that \( P_{DD} \) is \( B \)-block-banded, it can be shown that \( U_{D_kD_i} = 0 \) if \( t > k + B \) (see Lemma 1.1 of Asif & Moura (2005)). Hence, it follows that \( U_{D_kD_i} U_{D_kD_j} \) can only be non-zero if \( j \leq k + B \), which implies \( k \geq \max(1, j - B) \triangleq j_B \).

So,
\[
    P_{D_iD_j} = \sum_{k=j_B}^{i} U_{D_kD_i}^T U_{D_kD_j} ,
\]  
(36)
where \( H_{D_kD_k}^B \triangleq U_{D_kD_k}^T U_{D_kD_k} \) and \( H_{D_kD_k}^k \triangleq U_{D_kD_k}^T U_{D_kD_k} \). Finally, since \( P_{DD} \triangleq S_{DD}^{-1} \) is \( B \)-block-banded, it follows from Theorem 1 of Asif & Moura (2005) that, for \( k = 1, \ldots, M \),
\[
    \begin{bmatrix}
        S_{D_kD_k} & S_{D_kD_k}^b \\
        S_{D_k^bD_k} & S_{D_k^bD_k}^b
    \end{bmatrix}
    \begin{bmatrix}
        U_{D_kD_k}^T \cr
        U_{D_kD_k}^b
    \end{bmatrix}
    =
    \begin{bmatrix}
        U_{D_kD_k}^T \\
        0
    \end{bmatrix} .
\]  
(37)
Multiplying both sides of (37) with \( U_{D_kD_k} \) gives
\[
    \begin{bmatrix}
        S_{D_kD_k} & S_{D_kD_k}^b \\
        S_{D_k^bD_k} & S_{D_k^bD_k}^b
    \end{bmatrix}
    \begin{bmatrix}
        H_{D_kD_k}^k \\
        H_{D_kD_k}^b
    \end{bmatrix}
    =
    \begin{bmatrix}
        I \\
        0
    \end{bmatrix} .
\]  
(38)
where \( H_{D_kD_k}^k \triangleq H_{D_kD_k}^T \), \( k^*_B \triangleq \min(M, k + B) \), and \( I \) is an identity matrix of size \( |D|/M \). From (38),
\[
    \begin{bmatrix}
        H_{D_kD_k}^k \\
        H_{D_kD_k}^b
    \end{bmatrix}
    =
    \begin{bmatrix}
        S_{D_kD_k} & S_{D_kD_k}^b \\
        S_{D_k^bD_k} & S_{D_k^bD_k}^b
    \end{bmatrix}^{-1}
    \begin{bmatrix}
        I \\
        0
    \end{bmatrix} .
\]  
(39)
From (39), it is straightforward to see that $H_{D_k D_k}^k = G_{D_k D_k}^k$ and $H_{D_k D_k}^{k-1} = G_{D_k D_k}^{k-1}$, the latter of which implies $H_{D_k D_k}^k = G_{D_k D_k}^k$ for $i = k + 1, \ldots, k_B$. Plugging these into (36) yields (7).

**E. Proof of Theorem 1**

Since $P_{D_j} = \emptyset$ for $|i - j| > B$,

$$V_{SD} = K_{SD} P_{D'D} (y_D - \mu_D) = \sum_{i=1}^{M} V_i$$  

(40)

where $V_i = \sum_{j \in B(i)} K_{SD} P_{D_j} (y_{D_j} - \mu_{D_j})$ such that $B(i) = B^+(i) \cup B^-(i)$ and $B^-(i) \triangleq \{i_B, \ldots, i - 1\}$.

Since $P_{D'D}$ is symmetric,

$$P_{D_j} = P_{D_j}^T = \sum_{k=i_B}^{i} \left( G_{D_k D_k}^k G_{D_k D_k}^{k-1} G_{D_k D_k}^k \right)$$  

(41)

for $j = i_B, \ldots, i - 1$ such that the second equality follows from (7).

Plugging (7) and (41) into the expression of $V_i$ yields

$$V_i = \sum_{j \in B^+(i)} \sum_{k=j}^{i} v_{ij}^k + \sum_{j \in B^-(i)} \sum_{k=i}^{j} v_{ij}^k$$

where $v_{ij}^k = K_{SD} G_{D_k D_k} G_{D_k D_k}^{k-1} G_{D_k D_k}^k (y_{D_j} - \mu_{D_j})$. Therefore, $V_{SD}$ (40) can be equivalently expressed as

$$V_{SD} = \sum_{i=1}^{M} \sum_{j \in B^+(i)} \sum_{k=j}^{i} v_{ij}^k + \sum_{i=1}^{M} \sum_{j \in B^-(i)} \sum_{k=i}^{j} v_{ij}^k .$$  

(42)

To simplify (42), note that there is a one-to-one correspondence from $\{v_{ij}^k\}_{i=1, \ldots, M, j \in B^+(i)}$ to $\{v_{ij}^k\}_{i=1, \ldots, M, k \leq i \leq j}$. Similarly, there is a one-to-one correspondence from $\{v_{ij}^k\}_{i=1, \ldots, M, j \in B^-(i)}$ to $\{v_{ij}^k\}_{i=1, \ldots, M, k \leq i \leq j}$. Hence, (42) can be (more) concisely written as

$$V_{SD} = \sum_{k=1}^{M} \sum_{i \in B^+(k)} \sum_{j \in B^+(k)} v_{ij}^k = \sum_{k=1}^{M} \alpha_k$$  

(43)

where the last equality is due to the definition of $\alpha_k$ (8). By following a similar argument as above, it can be shown that

$$\Gamma_{SS} = K_{SS} + \sum_{i=1}^{M} u_i$$

where

$$u_i = \sum_{j \in B^+(i)} \sum_{k=j}^{i} u_{ij}^k + \sum_{j \in B^-(i)} \sum_{k=i}^{j} u_{ij}^k$$  

(44)

such that $u_{ij}^k = K_{SD} G_{D_k D_k} G_{D_k D_k}^{k-1} G_{D_k D_k}^k (y_{D_j} - \mu_{D_j})$. Then,

$$\Gamma_{SS} = K_{SS} + \sum_{i=1}^{M} \sum_{j \in B^+(i)} u_{ij}^k + \sum_{i=1}^{M} \sum_{j \in B^-(i)} u_{ij}^k$$

(45)

where the last equality is due to the definition of $\beta_k$ (8).

**F. Approximating the GP Predictive Distribution $p(f_u | y_D)$**

Given $q(f_S)$ (6) that maximizes the lower bound $L(q, Z)$ of the log-marginal likelihood $\log p(y_D)$ (see (3) and (4)), this section describes how it can be used to derive a predictive distribution efficiently. Specifically, we will construct an efficient approximation to the GP predictive distribution $p(f_u | y_D)$ for any subset $U_i = V_i \setminus D_i$ of test inputs by exploiting the fact that evaluating $q(f_S)$ incurs only linear time in the data size $|D|$. Let us first express the predictive distribution $p(f_u | y_D)$ of the FGPR model by marginalizing out the inducing output variables $f_S$ and approximating $p(f_S | y_D)$ with $q(f_S)$ to yield $p(f_u | y_D) \simeq q(f_u | y_D)$:

$$p(f_u | y_D) = \int p(f_u | f_S, y_D) p(f_S | y_D) \, df_S$$

$$\simeq \int p(f_u | f_S, y_D) q(f_S) \, df_S$$

$$\triangleq q(f_u | y_D) .$$

However, the above approximation is not sufficient to guarantee that $q(f_u | y_D)$ can be evaluated in linear time since the exact test conditional $p(f_u | f_S, y_D)$ generally incurs $O(|D|^2)$ time.

To overcome this computational bottleneck, a strategy, which is widely (albeit implicitly) used by the existing literature (Titsias, 2009a; Hensman et al., 2013; Hoang et al., 2015), is to impose an additional approximation $q(f_k | f_S, y_D)$ to the test conditional $p(f_u | f_S, y_D)$ such that

$$p(f_u | y_D) \simeq \int q(f_u | f_S, y_D) q(f_S) \, df_S \triangleq q(f_u | y_D) .$$  

(46)

can be integrated in linear time. For example, the work of Titsias (2009a) derives $q(f_S)$ (6) using the covariance matrix $S_{DD} \triangleq \sigma_n^2 I$ of observation noises given
some constant noise variance hyperparameter $\sigma_n^2$, which is equivalent to choosing the covariance function $k_{xx'} \triangleq \sigma_n^2 \delta(x = x')$ for the noise process model, and implicitly sets $q(f_{u,l}|f_{s},y_{p}) \triangleq p(f_{u,l}|f_{s}) = N(f_{u,l}|K_{u,s}K_{s,s}^{-1}(f_{s} - \mu_s), K_{u,u} - K_{u,s}K_{s,s}^{-1}K_{u,s})$. As a result, $q(f_{u,l}|y_{p})$ can be derived efficiently in linear time using (46), which interestingly recovers the predictive distribution of DTC (Seeger et al., 2003).

More generally, the recent work of Hoang et al. (2015) has shown that the predictive distributions of the other SGPR models (i.e., SoR (Smola & Bartlett, 2001), FITC (Snelson & Ghahramani, 2005), FIC, PITC (Schwaighofer & Tresp, 2003), and PIC (Snelson & Ghahramani, 2007)) spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005) can also be recovered by varying $q(f_{s})$ and $q(f_{u,l}|f_{s},y_{p})$. In particular, if the covariance of the observation noises and the Markov order $B$ are set such that the resulting variationally optimal distribution $q(f_{s})$ (6) coincides with that induced by the SGPR models, then their predictive distributions can be recovered by integrating $q(f_{s})$ with the induced $q(f_{u,l}|f_{s},y_{p})$ of these SGPR models. Consequently, our variational inference framework can unify the SGPR models spanned by the unifying view of Quiñonero-Candela & Rasmussen (2005), as detailed in Section 4.1, and establish the noise covariance function $k_{xx'}$ for the noise process model for which the SGPR models are expected to perform best.

Alternatively, we propose to approximate $p(f_{u,l}|f_{s},y_{p}) \approx q(f_{u,l}|f_{s},y_{p}) \triangleq N(\mu_{u,l}|\mathbb{K}_{u,s}P_{DD}(y_{p} - \mu_{D}), \Sigma_{u|u,s,D})$ for any sub-set $\mathcal{U}_l = V_l \setminus D_l$ of test inputs where

$$
\mu_{u,l|D} \triangleq \mu_{u,l} + \mathbb{K}_{u,D}P_{DD}(y_{p} - \mu_{D}) - \mathbb{K}_{u,D}P_{DD}K_{DS}K_{SS}^{-1}(f_{s} - \mu_{S})
$$

$$
\Sigma_{u|u,s,D} \triangleq \mathbb{K}_{u,u} - \mathbb{K}_{u,D}P_{DD}\mathbb{K}_{DS}K_{SS}^{-1}
$$

(47)

such that $\mathbb{K}_{u,D} \triangleq \{\mathbb{K}_{u,D}\}_{j=1,...,M}$ with $\mathbb{K}_{u,D,j}$ being a submatrix of $\mathbb{K}_{V,V}$, $\mathbb{Q}_{V,V} \triangleq Q_{V,V} + S_{V,V}$, where $S_{V,V}$ is computed using (1), $\mathbb{K}_{u,D,j}$ is a submatrix of $\mathbb{K}_{V,V} \triangleq Q_{V,V} + S_{V,V}$, where $S_{V,V}$ is computed using (1), and $\mathbb{K}_{D,V} \triangleq \mathbb{K}_{V,D}$. Then, $q(f_{u,l}|y_{p}) \triangleq N(f_{u,l}|\mu_{u,l|D},\Sigma_{u|u,s,D})$ can be obtained by integrating the test conditional $q(f_{u,l}|f_{s},y_{p}) = N(f_{u,l}|\mu_{u,l|s,D},\Sigma_{u|u,s,D})$ in (47) with $q(f_{s}) = N(f_{s}|\mu_{s,D},\Sigma_{s|D})$ in (31) and (33) over $f_{s}$:

$$
\mu_{u,l|D} \triangleq \mu_{u,s|D} + b
$$

$$
\Sigma_{u|u,s,D} \triangleq A\Sigma_{s|D}A^T + \Sigma_{u|u,s,D}
$$

(48)

where

$$
A \triangleq -\mathbb{K}_{u,D}P_{DD}K_{DS}K_{SS}^{-1}
$$

$$
b \triangleq -\mu_{u,l} - \mathbb{K}_{u,D}P_{DD}(y_{p} - \mu_{D}) + K_{DS}K_{SS}^{-1}\mu_{S}
$$

(49)

Intuitively, the use of (47) to approximate the test conditional $p(f_{u,l}|f_{s},y_{p})$ yields two advantages: (a) It can be shown that computing $q(f_{u,l}|y_{p})$ (48) incurs only linear time in the data size $|D|$ due to the sparsity (i.e., $B$-block-banded structure) of $P_{DD}$ (Section 2) and its computation can be distributed among parallel machines/cores to achieve scalability (Appendix G), and (b) by setting $k_{xx'} \triangleq k_{xx} - K_{SS}K_{x}^{-1}K_{x,x'} + \sigma_n^2 \delta(x = x')$ (9), Section 4.2 shows that (48) coincides with the predictive distribution of a recently developed low-rank-cum-Markov approximation (LMA) which has been empirically demonstrated in (Low et al., 2015) to significantly outperform the existing state-of-the-art SGPR models in terms of predictive performance and time efficiency.

**G. Distributed Computation of (48)**

Plugging the expressions of $A$ and $b$ (49) into that of $\mu_{u,l|D}$, (48) yields

$$
\mu_{u,l|D} - \mu_{u,l} = \mathbb{K}_{u,D}P_{DD}z_{D}
$$

(50)

where $z_{D} \triangleq y_{p} - \mu_{D} - K_{D,S}K_{SS}^{-1}(\mu_{S,D} - \mu_{S})$.

Then, for $P_{D_{i}D_{j}} = 0$ for $|\ell - j| > B$ (Section 2),

$$
\begin{align*}
\mu_{u,l|D} - \mu_{u,l} &= \sum_{\ell=1}^{M} \sum_{j \in B(\ell)} \mathbb{K}_{u,D_{\ell}}P_{D_{\ell}D_{j}}z_{D_{j}} \\
&= \sum_{k=1}^{M} \sum_{l,j \in B(k)} \mathbb{K}_{u,D_{k}}G_{D_{k}D_{\ell}}^k G_{D_{k}D_{j}}^{k-1}z_{D_{j}}
\end{align*}
$$

(51)

where $B(.)$ is previously defined in Appendix E, $z_{D_{j}} = y_{p} - \mu_{D_{j}} - K_{D_{j},S}K_{SS}^{-1}(\mu_{S,D_{j}} - \mu_{S})$, and (51) can be derived in a similar manner as that in the proof of Theorem 1 (Appendix E). Since each summation term in (51) can be computed independently and only involves a small subset of the training and test data that can be pre-assigned to a particular machine/core, (51) (hence, the predictive mean $\mu_{u,l|D}$) can be computed in a distributed manner. Note that prior to computing the predictive mean $\mu_{u,l|D}$, $\mathbb{K}_{u,D_{i}}$ for $i, \ell = 1, \ldots, M$ can be pre-computed using the distributed procedure described in Appendix C of Low et al. (2015) while the submatrices $G_{D_{k}D_{\ell}}^k$ are zero for $k = 1, \ldots, M$ and $\ell, j \in B(k)$ and $\mu_{S,D_{j}}$ (33) of $q(f_{s})$ (6) can also be pre-computed in a distributed manner, as detailed in the respective paragraphs below Lemma 1 and Theorem 1.

On the other hand, $U_{SS} \triangleq \text{cholesky}(\Sigma_{S|D})$ can be pre-computed in $O(|S|^3)$ time given $\Sigma_{S|D}$ (31) of $q(f_{s})$ (6) which can be pre-computed in a distributed manner, as detailed in the paragraph below Theorem 1. Then, from (48),

$$
\Sigma_{u|u,s,D} = (U_{SS}A^T)^T(U_{SS}A^T) + \Sigma_{u|u,s,D}
$$

(52)
Since $P_{D_iD_j} = 0$ for $|\ell - j| > B$ (Section 2),

$$U_{SS} A^T = -U_{SS} K_{SS}^{-1} K_{SD} P_{DD} \mathbf{K}_{DU_i} = -U_{SS} K_{SS}^{-1} \sum_{\ell = 1}^M \sum_{j \in B(\ell)} K_{SD} P_{D_iD_j} \mathbf{K}_{DU_i}$$

$$= -U_{SS} K_{SS}^{-1} \sum_{k = 1}^{K} \sum_{\ell, j \in B(k)} K_{SD} G_{D_iD_k} G_{D_kD_j} G_{D_jD_k}^{-1} \mathbf{K}_{DU_i}$$

(53)

that can be derived in a similar manner as that in the proof of Theorem 1 (Appendix E). Again, each summation term in (53) can be computed independently and only involves a small subset of the training and test data. So, (53) can be computed in a distributed manner. Note that prior to computing $U_{SS} A^T$, $\mathbf{K}_{DU_i}$ for $i, j = 1, \ldots, M$ can be pre-computed using the distributed procedure described in Appendix C of Low et al. (2015) while the submatrices $G_{D_iD_k}$, $G_{D_kD_j}$, and $G_{D_jD_k}$ of $G^k$ for $k = 1, \ldots, M$ and $\ell, j \in B(k)$ can also be pre-computed in a distributed manner, as detailed in the paragraph below Lemma 1.

Likewise, from (47),

$$\Sigma_{U_i U_i|S \cup D} = \mathbf{K}_{U_i U_i} + \sum_{\ell = 1}^M \sum_{j \in B(\ell)} \mathbf{K}_{U_i D_j} P_{D_j D_j} \mathbf{K}_{D_j U_i}$$

$$= \mathbf{K}_{U_i U_i} + \sum_{k = 1}^{K} \sum_{\ell, j \in B(k)} \mathbf{K}_{U_i D_k} G_{D_k D_k} G_{D_k D_j} G_{D_j D_k}^{-1} \mathbf{K}_{D_j U_i}$$

(54)

that can be derived in a similar manner as that in the proof of Theorem 1 (Appendix E). Again, each summation term in (54) can be computed independently and only involves a small subset of the training and test data. So, (54) can be computed in a distributed manner. Note that prior to computing $\Sigma_{U_i U_i|S \cup D}$, $\mathbf{K}_{U_i D_j}$ and $\mathbf{K}_{D_j U_i}$ for $i, j, \ell = 1, \ldots, M$ can be pre-computed using the distributed procedure described in Appendix C of Low et al. (2015) while the submatrices $G_{D_k D_k}$, $G_{D_k D_j}$, and $G_{D_j D_k}$ of $G^k$ for $k = 1, \ldots, M$ and $\ell, j \in B(k)$ can also be pre-computed in a distributed manner, as detailed in the paragraph below Lemma 1.

Finally, after the distributed computations of (53) and (54), $\Sigma_{U_i U_i|D}$ can be constructed efficiently in $O(|U|^2 |S|) = O(|S|^3)^{11}$ time using (52) since $U_{SS} A^T$ and $\Sigma_{U_i U_i|S \cup D}$ are $|S| \times |U|$ and $|U| \times |U|$ matrices, respectively.

---

**H. Predictive Distribution $p(f_S|y_D)$ of FGPR Model with Independently Distributed Observation Noises of Constant Variance**

By the definition of a GP, the joint prior $p(f_S, f_D)$ can be expressed as

$$\begin{bmatrix} f_S \\ f_D \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \mu_S \\ \mu_D \end{bmatrix}, \begin{bmatrix} K_{SS} & K_{SD} \\ K_{DS} & K_{DD} \end{bmatrix} \right)$$

Then, since $p(y_D|f_D) = \mathcal{N}(y_D|f_D, \sigma_n^2 I)$ due to the assumption of independently distributed observation noises with constant noise variance $\sigma_n^2$, the joint prior $p(f_S, y_D)$ (Rasmussen & Williams, 2006) is given by

$$\begin{bmatrix} f_S \\ y_D \end{bmatrix} \sim \mathcal{N} \left( \begin{bmatrix} \mu_S \\ \mu_D \end{bmatrix}, \begin{bmatrix} K_{SS} & K_{SD} \\ K_{DS} & Q_{DD} + R_{DD} \end{bmatrix} \right)$$

(55)

where $R_{DD} = \sum_{\ell = 1}^M \mathbf{K}_{D_D}^\ell$. Using (55),

$$\mu_{FGR}^{\mathcal{S}|\mathcal{D}} = \mu_S + K_{SD}(Q_{DD} + R_{DD})^{-1}(y_D - \mu_D)$$

$$\sigma_{\mathcal{S}|\mathcal{D}}^{\mathcal{S}|\mathcal{D}} = K_{SS} - K_{SD}(Q_{DD} + R_{DD})^{-1}K_{DS}$$

(56)

Note that

$$Q_{DD} + R_{DD}^{-1} = R_{DD}^{-1} - R_{DD}^{-1}K_{DS}G_{SS}^{-1}K_{SD}R_{DD}^{-1}$$

which follows directly from the matrix inversion lemma where $\Gamma_{SS} \equiv K_{SS} + K_{SD} R_{DD}^{-1} K_{DS}$. Using (56) and (57),

$$\mu_{FGR}^{\mathcal{S}|\mathcal{D}} - \mu_S = K_{SD}(Q_{DD} + R_{DD})^{-1}(y_D - \mu_D)$$

$$= K_{SD} (R_{DD}^{-1} - R_{DD}^{-1}K_{DS}G_{SS}^{-1}K_{SD}R_{DD}^{-1})(y_D - \mu_D)$$

$$= (\Gamma_{SS} - K_{SD} R_{DD}^{-1} K_{DS}) \Gamma_{SS}^{-1} K_{SD} R_{DD}^{-1} (y_D - \mu_D)$$

$$= K_{SS} \Gamma_{SS}^{-1} K_{SD} R_{DD}^{-1} (y_D - \mu_D)$$

$$= K_{SS} \Gamma_{SS}^{-1} V_{SD}$$

where $V_{SD} \equiv K_{SD} R_{DD}^{-1} (y_D - \mu_D)$. Similarly,

$$\sigma_{\mathcal{S}|\mathcal{D}}^{\mathcal{S}|\mathcal{D}} = K_{SS} - K_{SD}(Q_{DD} + R_{DD})^{-1}K_{DS}$$

$$= K_{SS} - K_{SS} \Gamma_{SS}^{-1} K_{SD} R_{DD}^{-1} K_{DS}$$

$$= K_{SS} \Gamma_{SS}^{-1} (\Gamma_{SS} - K_{SD} R_{DD}^{-1} K_{DS})$$

$$= K_{SS} \Gamma_{SS}^{-1} K_{SS}$$

**I. Unifying with LMA**

Supposing a column vector $y_D$ of noisy outputs is observed for some set $\mathcal{D} \subset \chi$ of training inputs, LMA (Low et al., 2015) provides a predictive distribution $\mathcal{N}(f_U, \lambda_{\mathcal{U}|\mathcal{D}}, \Sigma_{\mathcal{U}|\mathcal{D}})$ for any subset $\mathcal{U} = \mathcal{V} \setminus \mathcal{D}$ of test inputs where

$$\mu_{\mathcal{U}|\mathcal{D}}^{\mathcal{L}} = \mu_U + \mathbf{K}_{U_D} \mathbf{K}_{D_D}^{-1} (y_D - \mu_D)$$

$$\Sigma_{\mathcal{U}|\mathcal{D}}^{\mathcal{L}} = \mathbf{K}_{U_U} - \mathbf{K}_{U_D} \mathbf{K}_{D_D}^{-1} \mathbf{K}_{D_U}$$

(58)
Then, as defined previously in Appendix F,
\[
\begin{align*}
\overline{K}_{DD}^{-1} &= (S_{DD} + Q_{DD})^{-1} \\
&= (S_{DD} + K_{DS}K_{SS}^{-1}K_{SD})^{-1} \\
&= S_{DD}^{-1} - S_{DD}^{-1}K_{DS}\Gamma_{SS}^{-1}K_{SD}S_{DD}^{-1}
\end{align*}
\]
where the last equality follows from the definition of \( \Gamma_{SS} \approx K_{SS} + K_{SD}S_{DD}^{-1}K_{PS} \) and the matrix inversion lemma. Since \( S_{DD}^{-1} = P_{DD} \) (Section 2), (59) can be rewritten as
\[
\overline{K}_{DD}^{-1} = P_{DD} - P_{DD}K_{DS}\Gamma_{SS}^{-1}K_{SD}P_{DD} \tag{60}
\]
Plugging (60) into (58) gives (after some algebraic manipulations)
\[
\begin{align*}
\mu_{I,D}^{\text{LMA}} &= A\mu_{S,D} + b \\
\Sigma_{I,D}^{\text{LMA}} &= A\Sigma_{SS,D}A^T + \Sigma_{I,U,S|D}
\end{align*}
\]
where (a) \( A \triangleq -\overline{K}_{U,D}P_{DD}K_{DS}K_{SS}^{-1}, \) (b) \( b \triangleq \mu_{U,D} + \frac{1}{2}\mu_{U,D}P_{DD}(y_D - \mu_D + K_{DS}K_{SS}\mu_S), \) (c) \( \Sigma_{I,U,S|D} \triangleq K_{U,D}K_{DD}^{-1}K_{DU} - K_{U,D}P_{DD}K_{DU}, \) and (d) \( \Sigma_{SS,D} \) and \( \mu_{S,D} \) are previously defined in (31) and (33), respectively. It can be observed that the predictive distribution of LMA (61) coincides with \( \mathbb{P}(f|y_D) \) (48) produced by our variational inference framework (Appendix F). LMA (Low et al., 2015) is therefore unified with our framework.

**J. Derivation of (10)**

To derive (10), note that (28) directly implies
\[
-E = \log N(y_D|\nu, S_{DD}) + \log \frac{p(f_S)}{q(f_S)} \tag{62}
\]
where \( E \) is a constant independent of \( f_S \) (see Appendix B). Taking expectations on both sides of (62) with respect to \( q(f_S) \) yields
\[
-E = \mathbb{E}_{q(f_S)} \left[ \log N(y_D|\nu, S_{DD}) + \log \frac{p(f_S)}{q(f_S)} \right]. \tag{63}
\]
Plugging (63) into (4) (Section 3) gives
\[
\mathcal{R}(Z) = -E - 0.5\text{Tr}[S_{DD}^{-1}(K_{DD} - Q_{DD})]. \tag{64}
\]
On the other hand, by the definition of \( E \) (see Appendix B),
\[
E = -\log \int f_S N(y_D|\nu, S_{DD}) p(f_S) \, df_S
= -\log \int f_S N(y_D|\Phi_{DS}f_S + \tau_D, S_{DD}) N(f_S|\mu_S, K_{SS}) \, df_S
\]
where \( \Phi_{DS} \triangleq K_{DS}K_{SS}^{-1}, \tau_D \triangleq \mu_D - \Phi_{DS}\mu_S \), and the last equality follows directly from the definition of \( \nu \) in (4).

Since \( \nu = \Phi_{DS}f_S + \tau_D \) is an affine transformation of \( f_S \), (65) can be reduced to
\[
\begin{align*}
E &= -\log \mathcal{N}(y_D|\Phi_{DS}\mu_S + \tau_D, S_{DD} + \Phi_{DS}K_{SS}\Phi_{DS}^T) \\
&= -\log \mathcal{N}(y_D|\mu_D, S_{DD} + K_{DS}K_{SS}^{-1}K_{SD}) \\
&= -\log \mathcal{N}(y_D|\mu_D, S_{DD} + Q_{DD}) \tag{66}
\end{align*}
\]
where the second and third equalities follow from the definitions of \( \Phi_{DS}, \tau_D \), and \( Q_{DD} \) in (4). Plugging (66) into (64) gives (10).

**K. Derivation of (11)**

To derive (11), note that
\[
\begin{align*}
&\mathcal{N}(y_D|\mu_D, S_{DD} + Q_{DD}) \\
&= -\frac{1}{2} \mathcal{N}(y_D - \mu_D)^T (S_{DD} + Q_{DD})^{-1}(y_D - \mu_D) \\
&= -\frac{1}{2} \log |S_{DD} + Q_{DD}| - \frac{1}{2} |D| \log(2\pi). \tag{67}
\end{align*}
\]
By plugging (67) into (10) and using the matrix determinant lemma:
\[
\begin{align*}
&\mathcal{N}(y_D|\mu_D, S_{DD} + Q_{DD}) \\
&= \log |S_{DD} + Q_{DD}| \\
&= \log |\Gamma_{SS}| - \log |K_{SS}| - \log |P_{DD}|
\end{align*}
\]
and the matrix inversion lemma:
\[
(S_{DD} + Q_{DD})^{-1} = P_{DD} - P_{DD}K_{DS}\Gamma_{SS}^{-1}K_{SD}P_{DD},
\]
\[
\mathcal{R}(Z) = \frac{1}{2} \log |K_{SS}| + \frac{1}{2} \log |P_{DD}| - \frac{1}{2} \log |\Gamma_{SS}| \\
+ \frac{1}{2} V_{SD}^T \Gamma_{SS}^{-1}V_{SD} - \frac{1}{2} \text{Tr}[P_{DD}W_{DD}] + F \tag{68}
\]
where \( V_{SD} \triangleq V_{SD}^T \) and \( F \triangleq -(1/2)|D| \log(2\pi). \) Finally, differentiating both sides of (68) with respect to \( z \in Z \) yields (11).

**L. Proof of Theorem 2**

Note that
\[
V_{SD} = \sum_{k=1}^{M} \alpha_k = \alpha \tag{69}
\]
where the first and second equalities follow directly from Theorem 1 and Definition 3, respectively. This consequently implies
\[
\frac{\partial V_{SD}}{\partial z} = \sum_{k=1}^{M} \frac{\partial \alpha_k}{\partial z} = \sum_{k=1}^{M} \alpha^z_k = \alpha^z \tag{70}
\]
where the second and last equalities follow directly from Definitions 2 and 3, respectively.
Similarly,
\[
\Gamma_{SS} = K_{SS} + \sum_{k=1}^{M} \beta_k = \beta
\]  
(71)
where the first and second equalities follow directly from Theorem 1 and Definition 3, respectively. This also implies
\[
\frac{\partial \Gamma_{SS}}{\partial z} = \frac{\partial K_{SS}}{\partial z} + \sum_{k=1}^{M} \frac{\partial \beta_k}{\partial z} = \frac{\partial K_{SS}}{\partial z} + \sum_{k=1}^{M} \beta_k^z
\]  
(72)
where the second and last equalities follow directly from Definitions 2 and 3, respectively.

It follows from (69), (70), (71), and (72) that
\[
\alpha^z \beta^{-1} \alpha = \frac{\partial V_{SD}^T \Gamma_{SS}^{-1} V_{SD}}{\partial z} - \frac{1}{2} \alpha^T \beta \beta^{-1} \alpha = \frac{1}{2} V_{SD}^T \Gamma_{SS}^{-1} \frac{\partial \Gamma_{SS}}{\partial z} V_{SD}
\]
\[
-\frac{1}{2} \text{Tr} \left[ \beta^{-1} \beta^z \right] = -\frac{1}{2} \text{Tr} \left[ \Gamma_{SS}^{-1} \frac{\partial \Gamma_{SS}}{\partial z} \right]
\]  
(73)
Lastly, let
\[
\Omega \triangleq -\frac{1}{2} \text{Tr} \left[ \frac{\partial P_{DD} W_{DD}}{\partial z} - P_{DD} \frac{\partial W_{DD}}{\partial z} - S_{DD} \frac{\partial P_{DD}}{\partial z} \right]
\]
\[
= -\frac{1}{2} \text{Tr} \left[ (P_{DD} W_{DD}) - S_{DD} P_{DD} \right]
\]
which corresponds to the first term on the RHS of (11). By exploiting the $B$-block-banded structure of $P_{DD}$, $\Omega$ can be rewritten as
\[
\Omega = -\frac{1}{2} \sum_{i=1}^{M} \sum_{j \in B^+(i)} \sum_{k=1}^{i} \omega_{ij}^k
\]
\[
-\frac{1}{2} \sum_{i=1}^{M} \sum_{j \in B^-(i)} \sum_{k=1}^{i} \omega_{ij}^k
\]  
(74)
where (74) can be derived in a similar manner as that in the proof of Theorem 1 (Appendix E) and
\[
\omega_{ij}^k \triangleq \text{Tr} \left[ \frac{\partial}{\partial z} \left( G_{D_i D_j}^k G_{D_k D_i}^{k-1} G_{D_k D_j}^k W_{D_i D_j} \right) \right]
\]
\[
-\text{Tr} \left[ \frac{\partial}{\partial z} \left( G_{D_i D_j}^k G_{D_k D_i}^{k-1} G_{D_k D_j}^k S_{D_i D_j} \right) \right]
\]  
(75)
Using (75) and the definitions of $\delta_k^z$ and $\psi_k^z$ (see Definition 2), (74) can be reduced to
\[
\Omega = -\frac{1}{2} \sum_{k=1}^{M} \delta_k^z - \psi_k^z = -\frac{1}{2} \phi^z
\]  
(76)
such that the last equality follows from Definition 3. Finally, plugging (73) and (76) into (11) yields (12).