Regularization for Sparse Latent Gaussian Process

Rui Meng*
Department of Statistics
University of California
Santa Cruz, CA 95060
rmeng1@ucsc.edu

Herbert Lee
Department of Statistics
University of California
Santa Cruz, CA 95060
herbie@ucsc.edu

Braden Soper
Lawrence Livermore National Laboratory
Livermore, CA
soper3@llnl.gov

Abstract

Gaussian Process Latent Variable Model (GPLVM) is a flexible unsupervised bayesian nonparametric modelling approach which has been applied to many learning tasks such as Facial Expression Recognition, Image Reconstruction, Human pose estimation. Due to poor scaling properties of exact inference methods on GPLVMs, approximation methods based on sparse Gaussian process (SGP) and stochastic variational inference (SVI) are necessary for inference on large data sets. One problem in SGP is that the distribution of inducing inputs may exhibit overdispersion which may lead to inefficient inference and poor model fit. In this paper, we first propose regularization approach for latent sparse Gaussian process in SVI, which balance the distribution of inducing inputs and latent inputs without the need to tune hyper-parameters. We justify the use of this regularization term by proving that performing variational inference (VI) on a GPLVM with this regularization term is equivalent to perform VI on a related empirical Bayes model with a prior on inducing inputs. Second, we extend categorical latent Gaussian process to model categorical time series and illustrate that our regularization is particularly important in this latent dynamic model with a synthetic data set. Finally, we apply our model to a real data set of stock indices and visualize the latent dynamics.

*Use footnote for providing further information about author (webpage, alternative address)—*not* for acknowledging funding agencies.

Preprint. Under review.

1 Introduction

2 Related Work

3 Sparse Gaussian Process

4 Gaussian Process Latent Variable Model

5 Regularization for Sparse Latent Gaussian Process

6 Regularization Theory

7 Temporal Categorical Latent Gaussian Process

8 Experiments

9 Conclusion

Latent variables are introduced as unknown inputs into GPs by [8] using the Gaussian Process Latent Variable Model (GPLVM). It considers a Gaussian process mapping from a latent space to a data space. The embedding inputs are optimized by maximizing the Gaussian process likelihood with respect to the latent input $\boldsymbol{X}$. Mathematically, the data $\boldsymbol{Y}$ are modeled as $\boldsymbol{Y} \sim \mathrm{GP}(\boldsymbol{X}, C(\boldsymbol{\theta}))$ where C is a covariance function and $\boldsymbol{\theta}$ are corresponding hyper-parameters. Then Titsias discusses the same model from a Bayesian perspective [14]. It gives a robust Bayesian training algorithm, which automatically selects the effective dimensionalities through the usage of automatic relevance determination (ARD) kernels [10].

Mathematically, let $Y \in \mathbb{R}^{N \times D}$ be the observed data where N is the number of observations and D the dimensionality of each data vector. Observations have corresponding latent variables $X \in \mathbb{R}^{N \times Q}$ where Q is the dimensionality on latent space. Assume the independence across features, the GPLVM is

$$\begin{aligned} y_{nd}|f_{nd} &\sim \mathcal{N}(y_{nd}|f_{nd}, \sigma^2 = \beta^{-1}) \\ f_{nd} &= \mathcal{F}_d(\boldsymbol{x}_n) \\ \mathcal{F}_d &\overset{iid}{\sim} \mathrm{GP}(0, C(\boldsymbol{\theta})) \end{aligned}$$

with normal prior of latent variables X, $p(X) = \prod_{n=1}^{N} \mathcal{N}(\boldsymbol{x}_n|\boldsymbol{0}, I_Q)$ where $\boldsymbol{x}_n$ is the nth row of X. [13, 14] apply variational sparse GP formulation by introducing a separate set of M inducing variable $\boldsymbol{u}_d \in \mathbb{R}^M$ for each dimension d evaluated at a set of inducing input locations $Z \in \mathbb{R}^{M \times Q}$. Then [14, 4] propose the variational structure

$$q(\boldsymbol{f}, \boldsymbol{u}, \boldsymbol{X}) = \prod_{d=1}^{D} \left(p(\boldsymbol{f}_d|\boldsymbol{u}_d, X) q(\boldsymbol{u}_d)\right) q(\boldsymbol{X})\,. \tag{1}$$

Then the evidence lower bound (ELBO) is

$$\begin{aligned} \mathrm{ELBO} &= E_{q(\boldsymbol{f},\boldsymbol{u},\boldsymbol{X})} \log \left(\frac{\left(\prod_{d=1}^{D} p(\boldsymbol{y}_d|\boldsymbol{f}_d) p(\boldsymbol{f}_d|\boldsymbol{u}_d, X) p(\boldsymbol{u}_d)\right) p(X)}{\left(\prod_{d=1}^{D} p(\boldsymbol{f}_d|\boldsymbol{u}_d, X) q(\boldsymbol{u}_d)\right) q(X)} \right) \\ &= E_{q(\boldsymbol{f},\boldsymbol{u},X)} \log \left(\frac{\left(\prod_{d=1}^{D} p(\boldsymbol{y}_d|\boldsymbol{f}_d) p(\boldsymbol{u}_d)\right) p(X)}{\left(\prod_{d=1}^{D} q(\boldsymbol{u}_d)\right) q(X)} \right) \\ &= \sum_{d=1}^{D} E_{q(\boldsymbol{f},\boldsymbol{u},X)} p(\boldsymbol{y}_d|\boldsymbol{f}_d) - \mathrm{KL}(q(\boldsymbol{u})||p(\boldsymbol{u})) - \mathrm{KL}(q(X)||p(X)) \end{aligned} \tag{2}$$

[14] marginalize the optimal variational distribution $q(\boldsymbol{u})$ and then maximize the ELBO with respect to $q(X)$ and Z, while [4] directly maximize the ELBO in terms of $q(\boldsymbol{u})$, $q(X)$ and Z. The time complexity of the computation of ELBO is reduced from $O(M^2N)$ in [14] to $O(M^3)$ in [4]. Specifically, ELBO in [14] is

$$\begin{aligned}\text{ELBO} &= \sum_{d=1}^{D} E_{q(X)} \log p(\boldsymbol{y}_d|X) - \text{KL}(q(X)||p(X)) \\ &\geq \left(\sum_{d=1}^{D} \log\left(\frac{(\beta)^{\frac{N}{2}}|K_{MM}|^{\frac{1}{2}}}{(2\pi)^{\frac{N}{2}}|\beta\Psi_2 + K_{MM}|^{\frac{1}{2}}} e^{-\frac{1}{2}\boldsymbol{y}_d^T W \boldsymbol{y}_d}\right)\right) - \frac{D\beta\psi_0}{2} \\ &\quad + \frac{D\beta}{2}\text{tr}(K_{MM}^{-1}\Psi_2) - \text{KL}(q(X)||p(X))\end{aligned}$$

where $W = \beta I_N - \beta^2\Psi_1(\beta\Psi_2 + K_{MM})^{-1}\Psi_1^T$, $\psi_0 = \text{tr}(\langle K_{NN}\rangle_{q(X)})$, $\Psi_1 = \langle K_{NM}\rangle_{q(X)}$ and $\Psi_2 = \langle K_{MN}K_{NM}\rangle_{q(X)}$. while ELBO in [4] is

$$\begin{aligned}\text{ELBO} &= \sum_{d=1}^{D} E_q \log p(\boldsymbol{y}_d|\boldsymbol{f}_d) - \text{KL}(q(\boldsymbol{u})||p(\boldsymbol{u})) - \text{KL}(q(X)||p(X)) \\ &= -\frac{\beta}{2}\left(\text{tr}(Y^TY) - 2\text{tr}(Y^T\Psi_1 K_{MM}^{-1}\boldsymbol{m}) + \text{tr}(\boldsymbol{m}^T K_{MM}^{-1}\Psi_2 K_{MM}^{-1}\boldsymbol{m}) + \text{tr}(K_{MM}^{-1}\Psi_2 K_{MM}^{-1}S)\right) \\ &\quad -\frac{ND}{2}\log(2\pi\beta^{-1}) + \frac{D\beta}{2}\left(-\psi_0 + \text{tr}(K_{MM}^{-1}\Psi_2)\right) - \text{KL}(q(\boldsymbol{u})||p(\boldsymbol{u})) - \text{KL}(q(X)||p(X))\,,\end{aligned}$$

where $S = \sum_{d=1}^{D} S_d$. However, if the likelihood $y_{nd}|fnd$ is non-Gaussian,

10 Regularization for Inducing Inputs

This regularization approach is proposed for inducing inputs to address an over-fitting issue caused by embedding inputs. One common approach in the literature for inference is to maximize the evidence lower boundary (ELBO) of CLGP/TCLGP. We find it helpful to put a penalty term on the ELBO, which accounts for the dissimilarity between the distribution of inducing inputs and the distribution of embedding inputs, as described in the next section. Since a TCLGP can be treated as a CLGP with Gaussian processes priors on embedding inputs, we introduce the regularization approach based on a CLGP.

As for a CLGP [3], a general model is defined as

$$\begin{aligned}y_{nd} &\sim \text{Cat}(\text{Softmax}(\boldsymbol{f}_{nd}))\,, \\ f_{ndk} &= \mathcal{F}_{dk}(\boldsymbol{x}_n)\,, \qquad u_{mdk} = \mathcal{F}_{dk}(\boldsymbol{z}_m) \\ \mathcal{F}_{dk}(\cdot) &\overset{iid}{\sim} \text{GP}(0, C(\boldsymbol{\theta}_d))\,.\end{aligned} \tag{3}$$

The corresponding ELBO is a lower bound of the log likelihood of the model derived by Jensen inequality and shown as

$$\begin{aligned}\log p(\boldsymbol{Y}) &= \log\left(\int \frac{p(\boldsymbol{Y},\boldsymbol{F},\boldsymbol{X},\boldsymbol{U})}{q(\boldsymbol{X},\boldsymbol{U},\boldsymbol{F})} q(\boldsymbol{X},\boldsymbol{U},\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\right) \\ &\geq \int \log\left(\frac{p(\boldsymbol{Y},\boldsymbol{F},\boldsymbol{X},\boldsymbol{U})}{q(\boldsymbol{X},\boldsymbol{U},\boldsymbol{F})}\right) q(\boldsymbol{X},\boldsymbol{U}\,\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} = \text{ELBO}\,.\end{aligned} \tag{4}$$

The modified ELBO (MELBO) is defined as

$$\text{MELBO} = \text{ELBO} - \lambda \text{R} \tag{5}$$

where λ is a regularization weight and R is a regularization term, as described in the next section.

10.1 Regularization using KL Divergence

The CLGP model introduces inducing inputs to simplify the computation in Gaussian processes and inducing inputs are optimized by maximizing the ELBO. However, for a low dimensional categorical dataset where the data size is significantly larger than the dimension size, there is a significant difference between the distribution of embedding inputs and the distribution of inducing inputs, as can be seen in Figure **??**. This is because the prior of embedding inputs forces them to concentrate around 0 while the inducing inputs cannot learn enough from maximizing the ELBO. From another aspect, because CLGP does not guarantee the local distribution preservation mentioned in [9], it is difficult for the inducing inputs $\boldsymbol{Z}$ to cover the important locations. So $\boldsymbol{Z}$ provides useless information for Gaussian processes prediction. Hence, it is difficult to assign suitable positions for inducing inputs in Gaussian processes through the maximization of the ELBO.

We introduce a regularization term in the ELBO of CLGP to motivate similar distributions between inducing inputs and embedding inputs. Hence, it guides the assignment for both inducing inputs and embedding inputs. The regularization term is proposed as

$$\mathrm{R} = \mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X})$$

where $\tilde{q}(\cdot)$ denotes a variational empirical distribution function . We utilize a Gaussian distribution class for the variational distribution $\tilde{q}$. Then distribution $\tilde{q}(\boldsymbol{Z})$ and $\tilde{q}(\boldsymbol{X})$ are naturally estimated using the sample mean and sample variance-covariance matrix as follows:

$$\tilde{q}(\boldsymbol{Z}) = \mathcal{N}(\hat{\boldsymbol{\mu}}_Z, \hat{\Sigma}_Z)\,, \tag{6}$$

$$\tilde{q}(\boldsymbol{X}) = \mathcal{N}(\hat{\boldsymbol{\mu}}_X, \hat{\Sigma}_X)\,, \tag{7}$$

where $\hat{\boldsymbol{\mu}}_Z = \bar{Z} = \frac{1}{M}\sum_{m=1}^{M} Z_m$, $\hat{\boldsymbol{\Sigma}}_Z = \frac{1}{M}\sum_{m=1}^{M}(Z_m - \bar{Z})^2$, $\hat{\boldsymbol{\mu}}_X = \bar{\boldsymbol{m}} = \frac{1}{N}\sum_{n=1}^{N}\boldsymbol{m}_{n\cdot}$ and $\hat{\Sigma}_X = \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{m}_{n\cdot} - \bar{\boldsymbol{m}})^2$.

So the MELBO is expressed as

$$\mathrm{MELBO} = \mathrm{ELBO} - \lambda \mathrm{KL}(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X})\,, \tag{8}$$

where λ is a regularization weight. Usually, λ is set to be equal to the number of inducing points. Under this setting, the MELBO can be treated equivalently as an empirical Bayesian approach with a prior on inducing inputs. The detail is further discussed in the next section.

11 Regularization Bayesian Theory

This section discusses the underlying relationship between MELBO in the CLGP model and ELBO in an empirical Bayesian model. First, we display the empirical Baysian model with a prior for inducing inputs $\boldsymbol{Z}$. Then we illustrate that maximizing MELBO is equivalent to maximizing a lower bound of the empirical Bayesian model.

11.1 Empirical Bayesian Model with a Prior of Inducing Inputs

Since predictive accuracy in sparse Gaussian processes strongly depends on optimal locations of inducing inputs, we force the distributions of the inducing inputs and the embedding inputs to be on the same scale. Therefore, we put an empirical prior on inducing inputs $\boldsymbol{Z}$. For the ease of computation, we assume the prior to be a Gaussian distribution and utilize the sample mean and sample covariance matrix of the embedding inputs $\boldsymbol{X}$ as the mean and covariance matrix of this prior. Mathematically, it implies that

$$\boldsymbol{z}_m \sim \mathcal{N}(\boldsymbol{z}_m|\hat{\boldsymbol{\mu}}_X, \hat{\Sigma}_X)\,, \tag{9}$$

$$q(\boldsymbol{z}_m) \sim \mathcal{N}(\boldsymbol{z}_m|\boldsymbol{\nu}_m, \Upsilon_m)\,, \tag{10}$$

for $m = 1, \ldots, M$, where the empirical prior depends on the variational distribution $q(\boldsymbol{X})$. Moreover, the empirical distribution is a normal approximation of the distribution of the mean of the embedding inputs' variational distribution $\boldsymbol{m}$. In order to guarantee a finite variation on inducing inputs $\boldsymbol{Z}$, we put a constraint on the mean $\boldsymbol{v}$ such that $\hat{\Sigma}_Z = |\frac{\sum_{m=1}^{M}(\boldsymbol{\nu}_m - \bar{\boldsymbol{\nu}})^2}{M}| < K$, where $\bar{\boldsymbol{\nu}} = \frac{\sum_{m=1}^{M}\boldsymbol{\nu}_m}{M}$.

Under this setting, the corresponding ELBO is

$$\begin{aligned}\log p(\boldsymbol{Y}) &\geq \ell_{eblo} = \int q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z},\boldsymbol{X},\boldsymbol{U})\log\frac{p(\boldsymbol{Z})p(\boldsymbol{X})p(\boldsymbol{U})p(\boldsymbol{Y}|\boldsymbol{F})}{q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})}d\boldsymbol{Z}\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \\ &= -\mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z})) - \mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\ &\quad + \int q(\boldsymbol{Z})q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{Z},\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{Z}\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\end{aligned}$$

Moreover, $\mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z}))$ can be rewritten as

$$\begin{aligned}\mathrm{KL}(q(\boldsymbol{Z})||p(\boldsymbol{Z})) &= \sum_{m=1}^{M}\mathrm{KL}(q(\boldsymbol{z}_m)||p(\boldsymbol{z}_m)) \\ &= \sum_{m=1}^{M}\frac{1}{2}\left[\log\frac{|\hat{\Sigma}_X|}{|\Upsilon_m|} - Q + \mathrm{tr}(\hat{\Sigma}_X^{-1}\Upsilon_m) + (\hat{\boldsymbol{\mu}}_X - \boldsymbol{\nu}_m)^T\hat{\Sigma}_X^{-1}(\hat{\boldsymbol{\mu}}_X - \boldsymbol{\nu}_m)\right]\end{aligned} \tag{11}$$

12 Relation between CLGP and Empirical Bayesian Model

Under the CLGP model (3), for the ease of notation, we rewrite $\boldsymbol{z}_m$ by $\boldsymbol{\nu}_m$ for all $m = 1, \ldots, M$. Then, we have

$$\begin{aligned}\hat{\boldsymbol{\mu}}_Z &= \frac{\sum_{m=1}^{M}\boldsymbol{\nu}_m}{M} \\ \hat{\Sigma}_Z &= \frac{\sum_{m=1}^{M}(\boldsymbol{\nu}_m - \hat{\boldsymbol{\mu}}_Z)(\boldsymbol{u}_m - \hat{\boldsymbol{\mu}}_Z)^T}{M}\,.\end{aligned}$$

Then to show that maximizing the MELBO is equivalent to maximizing a lower boundary of $\log p(\boldsymbol{Y})$ under the empirical Bayesian model, we introduce two lemmas and one theorem in appendix **??**.

Lemma 1 shows a lower bound of the empirical Bayesian model when choosing a specific distribution family of $q(\boldsymbol{Z})$ and Lemma 2 shows the calculation for the regularization term in CLGP. Then based on Lemma 1 and Lemma 2, we prove that maximizing the MELBO in CLGP when $\lambda = M$ is equivalent to maximizing the lower bound of $\log p(\boldsymbol{Y})$ in Lemma 1 under the empirical Bayesian model.

13 Variational Inference

The ELBO of the TCLGP model is expressed as

$$\begin{aligned}\log p(\boldsymbol{Y}) &\geq -\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}|\tilde{\boldsymbol{T}})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\ &\quad + \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})\log\left(\sum_{n=1}^{N}\sum_{d=1}^{D}\sum_{t=1}^{T}p(\boldsymbol{y}_{ndt}|\boldsymbol{f}_{ndt})\right)d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,,\end{aligned} \tag{12}$$

where $p(\boldsymbol{X}|\tilde{\boldsymbol{T}})$ is a product of densities of multivariate Gaussian distributions over all time series.

The variational distributions of $\boldsymbol{U}$ and $\boldsymbol{X}$ are constructed using independent Gaussian distributions such that

$$\begin{aligned}q(\boldsymbol{U}) &= \prod_{d=1}^{D}\prod_{k=1}^{K}\mathcal{N}(\boldsymbol{u}_{dk}|\boldsymbol{\mu}_{dk},\Sigma_d)\,, \\ q(\boldsymbol{X}) &= \prod_{n=1}^{N}\prod_{q=1}^{Q}\prod_{t=1}^{T}\mathcal{N}(x_{nqt}|m_{nqt},s_{nqt}^2)\,.\end{aligned}$$

Both $q(\boldsymbol{X})$ and $p(\boldsymbol{X}|\tilde{\boldsymbol{T}})$ have a multivariate Gaussian distribution. Their KL divergence has a closed-form expression such that:

$$\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}|\tilde{\boldsymbol{T}})) \quad = \quad \sum_{n=1}^{N}\sum_{q=1}^{Q}\mathrm{KL}\left(q(\boldsymbol{x}_{nq})||p(\boldsymbol{x}_{nq}|\tilde{\boldsymbol{T}}_n)\right)$$

where each $\mathrm{KL}\left(q(\boldsymbol{x}_{nq})||p(\boldsymbol{x}_{nq}|\tilde{\boldsymbol{T}}_n)\right)$ has a closed-form expression using the results of KL divergence of multivariate Gaussian distributions as follows.

Assume the dimension size of a multivariate variable is D, and $p \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$ and $q \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}, \tilde{\Sigma})$. Then the KL divergence between p and q is

$$\mathcal{KL}[p||q] \quad = \quad \frac{1}{2}\left(\log\frac{|\tilde{\Sigma}|}{|\Sigma|} - D + \mathrm{tr}(\tilde{\Sigma}^{-1}\Sigma) + (\tilde{\boldsymbol{\mu}}-\boldsymbol{\mu})^T\tilde{\Sigma}^{-1}(\tilde{\boldsymbol{\mu}}-\boldsymbol{\mu})\right). \tag{13}$$

The same is for the KL divergence between $q(\boldsymbol{U})$ and $p(\boldsymbol{U})$ such that

$$\begin{aligned}\mathrm{KL}\left(q(\boldsymbol{U})||p(\boldsymbol{U})\right) &= \sum_{d=1}^{D}\sum_{k=1}^{K}\mathrm{KL}(q(\boldsymbol{u}_{dk})||p(\boldsymbol{u}_{dk})) \\ &= \sum_{d=1}^{D}\sum_{k=1}^{K}\frac{1}{2}\left(\log\frac{|C(\boldsymbol{Z};\boldsymbol{\theta}_d)|}{|\Sigma_d|} - m + \mathrm{tr}(C(\boldsymbol{Z};\boldsymbol{\theta}_d)^{-1}\Sigma_d) + \boldsymbol{\mu}_{dk}^T C(\boldsymbol{Z};\boldsymbol{\theta}_d)^{-1}\boldsymbol{\mu}_{dk}\right).\end{aligned}$$

Here, $C(\boldsymbol{Z};\boldsymbol{\theta}_d)$ denotes the covariance matrix under the dth Gaussian process with respect to inputs $\boldsymbol{Z}$ and $C(\boldsymbol{Z},\boldsymbol{Z}^*;\boldsymbol{\theta}_d)$ denotes the covariance matrix under the dth Gaussian process with respect to inputs $\boldsymbol{Z}$ and $\boldsymbol{Z}^*$.

In the machine learning literature, especially in the auto-encoder literature, $\mathrm{KL}(q(\boldsymbol{X})||p(\boldsymbol{X}))$ and $\mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U}))$ are called regularization terms. They are used to minimize the distance between the variational distributions and their prior distributions. $\int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}$ is called the reconstruction term. It is used to describe the likelihood to reconstruct the observations. Therefore, maximizing the ELBO means maximizing the reconstruction term and at the same time minimizing the distance between the variational distributions and their prior distributions for $\boldsymbol{X}$ and $\boldsymbol{U}$.

As for the reconstruction term, directly computing the expectation is intractable. Thus, the expectation term is approximated using a Monte Carlo integration method [3]. Mathematically, the integration is approximated as

$$\int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})\log p(\boldsymbol{Y}|\boldsymbol{F})d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F} \quad = \quad \frac{1}{S}\sum_{s=1}^{S}\log p(\boldsymbol{Y}|\boldsymbol{F}^{(s)}) \tag{14}$$

where S denotes the number of samples in the Monte Carlo integration. $\boldsymbol{F}^{(s)}$ is sampled from $p(\boldsymbol{F}|\boldsymbol{X}^{(s)},\boldsymbol{U}^{(s)})$ where both $\boldsymbol{X}^{(s)}$ and $\boldsymbol{U}^{(s)}$ are sampled from $q(\boldsymbol{X})$ and $q(\boldsymbol{U})$ respectively. Because $q(\boldsymbol{X}), q(\boldsymbol{U})$, and $p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U})$ are all Gaussian distributions, generating the sample $\boldsymbol{F}$ is tractable.

On the other hand, $p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U}) = \prod_{d=1}^{D}\prod_{k=1}^{K}p(\boldsymbol{f}_{dk}|\boldsymbol{X},\boldsymbol{u}_{dk})$. It is still expensive to generate $\boldsymbol{f}_{dk}$ from $p(\boldsymbol{f}_{dk}|\boldsymbol{X},\boldsymbol{u}_{dk})$ because it costs a computation time of $O(nm^2)$. [3] implicitly assumes that $\boldsymbol{f}_{dk}$ are independent conditional on inducing variables $\boldsymbol{u}_{dk}$ in training processes, which is the same as the assumption in the Fully Independent Training Conditional Approximation (FITC) [11]. It means that

$$\begin{aligned}p(\boldsymbol{F}|\boldsymbol{X},\boldsymbol{U}) &= \prod_{n=1}^{N}\prod_{d=1}^{D}\prod_{t=1}^{T}\prod_{k=1}^{K}p(f_{ndtk}|\boldsymbol{x}_{nt},\boldsymbol{u}_{dk}) \\ &= \prod_{n=1}^{N}\prod_{d=1}^{D}\prod_{t=1}^{T}\prod_{k=1}^{K}\mathcal{N}(f_{ndtk}|a_{ndtk},b_{ndtk}^2)\,.\end{aligned} \tag{15}$$

where $a_{ntdk} = \boldsymbol{v}_{ndt}^T\Sigma_{Zd}^{-1}\boldsymbol{\mu}_{dk}$ and $b_{ntdk}^2 = \sigma_n^2 - \boldsymbol{v}_{ndt}^T\Sigma_{Zd}^{-1}\boldsymbol{v}_{ndt}$ and $\Sigma_{Zd} = C(\boldsymbol{Z};\boldsymbol{\theta}_d), \boldsymbol{v}_{ndt} = C(\boldsymbol{Z},\boldsymbol{x}_{nt};\boldsymbol{\theta}_d), \sigma_n^2 = C(\boldsymbol{x}_{nt},\boldsymbol{\theta}_d)$.

A linear transformation trick [7] is introduced for sampling to improve the inference efficiency. It re-parameterizes a random variables as a function of hyper-parameters and a random variable which does not depend on any hyper-parameter. Therefore, it is tractable to compute the derivative of the random variable with respect to its corresponding hyper-parameters. A general re-parameterization for multivariate normal distribution is discussed. That is involved in the computation of the ELBO.

For example, suppose a random variable $\boldsymbol{x}$ follows a multivariate Gaussian distribution $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$. Since the covariance matrix Σ is positive definite, it can be decomposed as $\Sigma = LL^T$ where L is a lower triangular matrix. Then $\boldsymbol{x}$ can be re-parameterized as $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{L\epsilon}$, where $\epsilon \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The corresponding derivatives are derived as

$$\begin{aligned} \frac{\partial x_i}{\partial \mu_j} &= \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}, \\ \frac{\partial x_i}{\partial l_{jk}} &= \begin{cases} \epsilon_k & i = j \\ 0 & i \neq j \end{cases} \qquad i \geq j\,. \end{aligned} \tag{16}$$

Generally, if $\boldsymbol{Y}$ has missing data, we denote observable $\boldsymbol{Y}$ as $\tilde{\boldsymbol{Y}} = \{Y_{ndt}\}_{(n,d,t)\in\Theta}$ with corresponding latent variables $\tilde{\boldsymbol{F}} = \{f_{ndt}\}_{(n,d,t)\in\Theta}$ and we denote the corresponding embedding inputs as $\tilde{\boldsymbol{X}}$. The ELBO is expressed as

$$\begin{aligned} \log p(\tilde{\boldsymbol{Y}}) \quad \geq \quad & -\mathrm{KL}(q(\tilde{\boldsymbol{X}})||p(\tilde{\boldsymbol{X}}|\tilde{\boldsymbol{T}})) - \mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) \\ & + \int q(\tilde{\boldsymbol{X}})q(\boldsymbol{U})p(\tilde{\boldsymbol{F}}|\tilde{\boldsymbol{X}}, \boldsymbol{U}) \log \left(\sum_{(n,d,t)\in\Theta} p(\boldsymbol{y}_{ndt}|\boldsymbol{f}_{ndt}) \right) d\tilde{\boldsymbol{X}}\boldsymbol{U}\tilde{\boldsymbol{F}}\,, \end{aligned} \tag{17}$$

Considering the regularization for inducing inputs, the MELBO is expressed as

$$\mathrm{MELBO} = \mathrm{ELBO} - \lambda KL(\tilde{q}(\boldsymbol{Z})||\tilde{q}(\boldsymbol{X}))\,, \tag{18}$$

where λ is a regularization weight. It is usually chosen as the number of inducing points.

Sometimes hyper-parameters of a Gaussian process across time are difficult to learn through optimization. Putting priors on those parameters can get rid of this issue. For example, if the RFB kernel is used as $Cov(t_i, t_j) = \phi_0 \exp(-\frac{|t_j - t_i|^2}{2\phi_1})$, then we can set $\phi_0 \sim \mathrm{Gamma}(2, 1)$ and $\phi_1 \sim \mathrm{Gamma}(2, 1)$ to guarantee that both scale-variance and scale-length parameters are reasonable in optimization.

Generally, after introducing a variation distribution on $\boldsymbol{\phi}$ denoted as $q(\phi_{qj}), \forall q = 1, \ldots, Q, \quad j = 1, \ldots J$, where J is the number of hyper-parameters in the Gaussian process, the ELBO is redefined as

$$\begin{aligned} \log p(\boldsymbol{Y}) \quad \geq \quad & -\mathrm{KL}(q(\boldsymbol{\phi})||p(\boldsymbol{\phi})) + \int q(\boldsymbol{\phi})q(\boldsymbol{X}) \log \left(\frac{p(\boldsymbol{X}|\boldsymbol{\phi}, \boldsymbol{t})}{q(\boldsymbol{X})} \right) d\boldsymbol{\phi}\boldsymbol{X} \\ & -\mathrm{KL}(q(\boldsymbol{U})||p(\boldsymbol{U})) + \int q(\boldsymbol{X})q(\boldsymbol{U})p(\boldsymbol{F}|\boldsymbol{X}, \boldsymbol{U}) \log p(\boldsymbol{Y}|\boldsymbol{F}) d\boldsymbol{X}\boldsymbol{U}\boldsymbol{F}\,. \end{aligned}$$

Then we utilize the stochastic gradient descent (SGD) method to maximize the ELBO with respect to the hyper-parameters in the GPs and the variational parameters. The details of SGD are discussed as follows.

Stochastic Gradient Descent is one class of optimization methods. Those methods are pretty popular in the deep learning literature because of their properties of cheap computation and scalability. Through learning an optimized learning rate with respect to gradients, Duchi proposed AdaGrad [2]. Then AdaDelta [15] and RMSPROP [5] are proposed to solve the aggressive learning rate issues. Then Adaptive Moment Estimation is proposed by [6] which takes the momentum into consideration and has a great performance for a large dataset.

In the context of TCLGP, we compare AdaGrad, AdaDelta, RMSPROP and Adam for synthetic datasets. AdaGrad has a significantly worse performance than others, and Adam is slightly better than AdaDelta and RMSPROP. Hence, we adopt Adam for all experiments.

On the other hand, the KL-vanishing problem sometimes happens depending on specific datasets. From the auto-encoder perspective [12], TCLGP gains more from ignoring the regularization of latent

variables and utilizing the reconstruction term. So optimization is biased toward reconstruction and ignoring regularization of latent variables. We utilize the KL-annealing [1] for an easier optimization. The KL-annealing suggests adding a small weight to KL divergences initially and increasing this weight gradually to 1 in the whole optimization process. This approach prevents the TCLGP model from zeroing out the KL divergences in the early training stage.

The details of stochastic variational inference algorithm for large datasets are discussed in the appendix **??**.

References

[1] Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A. M., Jozefowicz, R., and Bengio, S. (2015). Generating Sentences from a Continuous Space. *ArXiv e-prints*.

[2] Duchi, J., Hazan, E., and Singer, Y. (2010). Adaptive subgradient methods for online learning and stochastic optimization. Technical Report UCB/EECS-2010-24, EECS Department, University of California, Berkeley.

[3] Gal, Y., Chen, Y., and Ghahramani, Z. (2015). Latent Gaussian Processes for Distribution Estimation of Multivariate Categorical Data. *ArXiv e-prints*.

[4] Hensman, J., Fusi, N., and Lawrence, N. D. (2013). Gaussian Processes for Big Data. *ArXiv e-prints*.

[5] Hinton, G. (2012). Lecture 6e of his coursera class. *Lecture in his Coursera*.

[6] Kingma, D. P. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. *ArXiv e-prints*.

[7] Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *CoRR*, abs/1312.6114.

[8] Lawrence, N. D. (2003). Gaussian process latent variable models for visualisation of high dimensional data. In *NIPS*.

[9] Lawrence, N. D. and Moore, A. J. (2007). Hierarchical gaussian process latent variable models. In *Proceedings of the 24th International Conference on Machine Learning*, ICML '07, pages 481–488, New York, NY, USA. ACM.

[10] MacKay, D. J. C. (1996). *Bayesian Methods for Backpropagation Networks*, pages 211–254. Springer New York, New York, NY.

[11] Rasmussen, C. E. and Williams, C. K. I. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press.

[12] Shen, X., Su, H., Niu, S., and Demberg, V. (2018). Improving Variational Encoder-Decoders in Dialogue Generation. *ArXiv e-prints*.

[13] Titsias, M. (2009). Variational learning of inducing variables in sparse gaussian processes. In van Dyk, D. and Welling, M., editors, *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA. PMLR.

[14] Titsias, M. and Lawrence, N. D. (2010). Bayesian gaussian process latent variable model. In Teh, Y. W. and Titterington, M., editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 844–851, Chia Laguna Resort, Sardinia, Italy. PMLR.

[15] Zeiler, M. D. (2012). ADADELTA: An Adaptive Learning Rate Method. *ArXiv e-prints*.