Ensembles of Gaussian process latent variable models

Marzieh Ajirak,[◇] Yuhao Liu,[†] Petar M. Djurić[◇]
[◇] Department of Electrical and Computer Engineering
[†]Department of Applied Mathematics and Statistics
Stony Brook University, NY 11794

***Abstract*—In this paper, we address the classification and dimensionality reduction via ensembles of Gaussian Process Latent Variable Models (GPLVMs). The underlying idea is to have a diverse representation of latent spaces represented by an ensemble of GPLVMs. Each GPLVM of the ensemble has its own projections of the high dimensional observed data on a low dimensional latent space. These models are weighted using importance sampling. Since in practical settings, neither the kernel of the GPLVM nor the dimension of the latent space is known, it is logical to engage an ensemble of GPLVMs based on different kernels and for each of them estimate the dimension of the lower dimensional space. We demonstrate the advantage of working with ensembles for classification and show the performance of dimensionality reduction of our method with numerical simulations.**

I. Introduction

Gaussian process latent variable models (GPLVMs) are machine learning (ML) methods that combine latent variable models and GPs [1]. In these models, the input variables are not observed and are of a much lower dimension than the output variables. Thus, one important characteristic of these models is that they provide a compressed representation of high-dimensional data. Further, they can work with different data types as well as with missing data and can take advantage of the availability of prior information [2]. They have often been used in tasks like classification of high dimensional vectors, clustering, and regression and have found a wide range of applications, from neuroscience [3] and bioinformatics [4] to robotics [5] and finance [6].

The framework of variational inference for integration of the latent variables was adopted in [7], where the concept of an expanded probability model with auxiliary inducing variables was exploited. The approach requires the joint maximization of Jensen's lower bound over the variational parameters that include the parameters of the approximated posterior of the latent variables and the model hyperparameters. In [8], the method from [7] was expanded by using auxiliary inducing variables of the GP prior, and where the latent variables are marginalized with respect to the variational posterior.

Monte Carlo-based methods have also been explored for inference of GPs and GPLVMs. In [9], the authors apply a Markov chain Monte Carlo method [10] as opposed to the Gibbs sampling approach presented in [11]. They show that their scheme is more efficient for regression and classification than Gibbs sampling because the latter suffers considerably from high correlations of the high-dimensional variables. In their method, they propose to use control variables for generating better samples, and their meaning is analogous to that of the inducing variables of sparse GP models [12]. In [13], a hybrid Monte Carlo sampling method was proposed that simultaneously addresses the approximation of the posterior of a GP, the estimation of the parameters of the covariance function, and scalability. With this approach, the drawn samples of inducing points and covariance parameters are used for computing integrals. In [14], the Metropolis-within Gibbs construction was adopted as a sampling tool across the levels of a hierarchy of deep GPs. Another MCMC-based approach for learning deep GPLVMs was proposed in [15], where the variational approximation is used to initialize the Markov chains and thereby speed up the convergence. It is also shown that the posterior approximation with the proposed method is better than that of the underlying variational approximation. More recently, stochastic gradient Hamiltonian Monte Carlo was employed for inference of deep GPs [16].

In this paper, we aim at working with an ensemble of GPLVMs and exploiting their diversity to get improved performance in classification and dimension estimation of the latent space. We can construct the GPLVMs in different ways, e.g., by sampling from the variational posteriors and computing the weights of the samples using the principle of importance sampling (or adaptive importance sampling) or by using variational inference many times with different initial conditions, which leads to different projections on the lower-dimensional space and again applying importance sampling. By constructing different GPLVMs with distinct types of kernels and operating in latent spaces of different dimensions, we improve the overall capacity for exploring the lower dimensional spaces and eventually enhancing the performance of the ensemble of GPLVMs.

Our contributions in this paper include the following: (1) introduction of the concept of an ensemble of GPLVMs for tasks like classification and dimensionality reduction, (2) use of importance sampling for evaluating the members of the ensemble and for proposing new members with better capacity for performing the intended task, and (3) proposal of methods for leveraging the ensembles of GPLVMs for improved performance.

The rest of the paper is organized as follows. In Section

The authors thank the support of NSF under Award 2021002.

II, we briefly review the GPLVMs. Section III introduces the concept of an ensemble of GPLVMs and elaborates on how we construct the member GPLVMs. We propose a method for classification based on an ensemble of GPLVMs in Section IV and a method for estimating the dimension of the latent space and types of kernels in Section V. In Section VI we provide numerical results that demonstrate the performance of the proposed methodology, and in Section VII we conclude our paper with our final remarks.

II. The inference problem

Consider a set of observations $\mathbf{Y} \in \mathbb{R}^{N\times D}$ that are assumed to be generated according to the following equation:

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{d=1}^{D} p\left(\mathbf{y}_d|\mathbf{X}\right), \tag{1}$$

where $\mathbf{y}_d \in \mathbb{R}^{N\times 1}$, $\mathbf{X} \in \mathbb{R}^{N\times Q}$, with $Q << D$, and where the probability density function of $\mathbf{y}_d$ conditioned on the variables $\mathbf{X}$ is given by

$$p\left(\mathbf{y}_d|\mathbf{X}\right) = \mathcal{N}\left(\mathbf{y}_d|\mathbf{0}, \mathbf{K}_{NN} + \beta^{-1}\mathbf{I}_N\right). \tag{2}$$

Here the notation $\mathcal{N}\left(\mathbf{y}_d \mid \mathbf{0}, \mathbf{K}_{NN} + \beta^{-1}\mathbf{I}_N\right)$ means that the vector $\mathbf{y}_d$ is distributed according to the multivariate Gaussian distribution with zero mean and a covariance matrix $\mathbf{K}_{NN} + \beta^{-1}\mathbf{I}_N$, where $\mathbf{K}_{NN} \in \mathbb{R}^{N\times N}$ is constructed by using specific kernel(s) and $\mathbf{X}$. For example, in the case of the radial-basis function kernel, the elements of $\mathbf{K}_{NN}$ are

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_{q=1}^{Q}\alpha_q\left(x_q - x_q'\right)^2\right) \tag{3}$$

where σ_f^2 and $\alpha_{1:Q}$ are hyperparameters. The matrix $\mathbf{I}_N$ in (2) is the identity matrix of size $N \times N$, and $\beta \in \mathbb{R}^+$ is a precision parameter. Further, it is assumed that the probability density function (pdf) of $\mathbf{X}$ is

$$p(\mathbf{X}) = \prod_{n=1}^{N}\mathcal{N}\left(\mathbf{x}_n|\mathbf{0}, \mathbf{I}_Q\right), \tag{4}$$

where $\mathbf{x}_n$ is the nth row of $\mathbf{X}$. In this model, the only variables that are observed are the elements of the matrix $\mathbf{Y}$. Thus, if we use $\mathbf{K}_{NN}$ with entries defined by (3), $\mathbf{X}, \sigma_f^2, \alpha_{1:Q}, \beta$ are unknown. The main objective is to find the unknown matrix $\mathbf{X}$. The true posterior $p(\mathbf{X}|\mathbf{Y})$ is difficult to obtain, and thus, we approximate it by

$$q(\mathbf{X}|\mathbf{Y}) = \prod_{n=1}^{N}\mathcal{N}\left(\mathbf{x}_n|\boldsymbol{\mu}_n, \mathbf{S}_n\right), \tag{5}$$

where the parameters $\{\boldsymbol{\mu}_n, \mathbf{S}_n\}_{n=1}^N$ are obtained by the variational inference principle [7]. We proceed by obtaining a closed form expression of the lower bound on $p(\mathbf{Y}|\mathbf{X})$, followed by jointly maximizing it over the variational parameters $\{\boldsymbol{\mu}_n, \mathbf{S}_n\}_{n=1}^N$ and the hyperparameters. We point out that the variational posterior distribution $q(\mathbf{X}|\mathbf{Y})$ is only approximate because it is immediately clear that the true posterior $p(\mathbf{X}|\mathbf{Y})$ cannot be Gaussian. We also observe that the posterior in (5) is conditioned on a conjectured dimension Q of $\mathbf{x}_n$. In [7], it is first assumed that the dimension of $\mathbf{x}_n$ is larger than the true dimension, and one selects the dimension Q by using the principle of automatic relevance determination (ARD), where only dimensions of large inverse length-scales (small values of α_q in (3)) are kept.

III. Generation of ensembles of GPLVMs

As a prior of $\mathbf{X}$ we start with the following pdf:

$$\begin{aligned} p(\mathbf{X}|\sigma^2) &= \prod_{n=1}^{N}\mathcal{N}(\mathbf{x}_n|\mathbf{0}, \sigma^2\mathbf{I}) \\ &= \frac{1}{\left(2\pi\sigma^2\right)^{\frac{NQ}{2}}} e^{-\frac{\sum_{n=1}^{N}\mathbf{x}_n^\top\mathbf{x}_n}{2\sigma^2}}, \end{aligned} \tag{6}$$

where σ^2 is a parameter that in principle has to be set and that will play a significant role in the estimation of Q. To avoid the problem of selecting specific values of σ^2, we assume that σ^2 comes from its own prior $p(\sigma^2) \propto \frac{1}{\sigma^2}$. When we use it to marginalize σ^2 from (6), the prior for $\mathbf{X}$ becomes

$$\begin{aligned} p(\mathbf{X}) &\propto \int_0^\infty p(\mathbf{X}|\sigma^2)p(\sigma^2)\mathrm{d}\sigma^2 \\ &= \frac{1}{2}\left(\frac{1}{2\pi}\right)^{\frac{NQ}{2}}\Gamma\left(\frac{NQ}{4}\right)\left(\frac{\sum_{n=1}^{N}\mathbf{x}_n^\top\mathbf{x}_n}{2}\right)^{-\frac{NQ}{2}}, \end{aligned} \tag{7}$$

where for the integration of σ^2 we used the integral [17]

$$\int_0^\infty \left(\sigma^2\right)^{-(k+1)} e^{-\frac{a}{\sigma^2}}\mathrm{d}\sigma^2 = \frac{1}{2}a^{-\frac{k}{2}}\Gamma\left(\frac{k}{2}\right). \tag{8}$$

The Gaussian prior from (6) is averaged over all σ^2, and it encourages/discourages models based on the proposed $\mathbf{X}$ and dimension Q. Typically, higher values of Q are discouraged (penalized more).

Next, suppose we obtain a variational posterior $q(\mathbf{X}|\mathbf{Y})$ as in (5). We can use this posterior to draw samples of the latent variables, that is,

$$\boldsymbol{X}^{(m)} \sim q(\boldsymbol{X}|\mathbf{Y}), \quad m = 1, \ldots, M, \tag{9}$$

where $\boldsymbol{X}^{(m)} \in \mathbb{R}^{N\times Q}$. We then compute the weights of the drawn samples according to

$$\tilde{w}^{(m)} = \frac{p\left(\mathbf{Y}|\boldsymbol{X}^{(m)}\right)p(\mathbf{X}^{(m)})}{q\left(\boldsymbol{X}^{(m)}|\mathbf{Y}\right)}, \quad m = 1, \ldots, M, \tag{10}$$

where $p(\mathbf{X})$ is given by (7), and where we exploit the principle of importance sampling [10]. The samples are then normalized by $w^{(m)} = \tilde{w}^{(m)}/\sum_{k=1}^{M}\tilde{w}^{(k)}$, and we have an approximation of the posterior of $\mathbf{X}$ given by

$$p^M(\mathbf{X}|\mathbf{Y}) = \sum_{m=1}^{M} w^{(m)}\delta\left(\mathbf{X} - \mathbf{X}^{(m)}\right), \tag{11}$$

where $\delta(\cdot)$ stands for the Dirac delta function. We note that we can use the set $\{\mathbf{X}^{(m)}, w^{(m)}\}$ to construct an improved

proposal for $\mathbf{X}$ in the spirit of adaptive importance sampling [18].

An approach for improving the set of samples $\mathbf{X}^{(m)}$ is to use first a proposal that is a mixture of variational posteriors and given by

$$q_1(\mathbf{X}) = \frac{1}{L}\sum_{l=1}^{L} q_{1,l}(\mathbf{X}|\mathbf{Y}), \tag{12}$$

where the mixands $q_{1,l}(\mathbf{X}|\mathbf{Y})$ are the variational posteriors obtained by *different initialization* of the posterior. In the next step, we cluster the vectors $\mathbf{x}_{1,n}^{(m)}$ into K groups, and from the obtained groups we construct Gaussians for drawing another generation of vectors $\mathbf{x}_n$. They are generated from

$$q_2(\mathbf{X}) = \sum_{k=1}^{K} w_{2,k} q_{2,k}(\mathbf{X}|\mathbf{Y}), \tag{13}$$

where $w_{2,k}$ is the weight given to the kth mixand $q_{2,k}(\mathbf{X}|\mathbf{Y})$ and which is obtained by adding the weights of $\mathbf{x}_{1,n}^{(m)}$ that belong to the kth cluster. Once the parameters of the Gaussians are constructed from the cluster members, we draw the next generation of samples by

$$\mathbf{x}_{2,n}^{(m)} \sim \mathcal{N}(\boldsymbol{\mu}_{2,n}, \mathbf{S}_{2,n}), \quad m = 1, 2, \cdots, M. \tag{14}$$

It is well known that the choice of the kernel of the GPLVM is crucial for the performance of the GPLVM. It is also known that there are many basic types of kernels to choose from and innumerable possibilities to combine them. In this case, we can view the step of selecting kernels as another form of sampling from the space of kernels. Once a kernel is picked, we proceed with determining the best dimension of the latent space for its operation. When the dimensions of the GPLVMs are determined, the GPLVMs are assigned weights. These weights suggest which types of kernels are better for the analyzed data.

IV. Classification using ensemble GPLVMs

The problem of classification amounts to assigning an observed vector $\mathbf{y}$ to one of the predetermined classes c_k, $k = 1, 2, \cdots, K$. A Bayesian approach to this problem is based on computing the posterior probability $P(c_k|\mathbf{y})$, $\forall k$, and assigning $\mathbf{y}$ to c_k that has the maximum posterior probability.

Suppose that we have labeled data $\mathbf{Y}$ and have constructed many copies of the latent variables $\mathbf{X}^{(m)}$, $m = 1, 2, \cdots, M$. Then we observe a new vector $\mathbf{y}_*$ and have to classify it into one of the K classes c_k. The first step is, to draw samples $\mathbf{x}_*^{(j)}$ according to

$$\mathbf{x}_*^{(j,m)} \sim q(\mathbf{x}_*|\mathbf{Y}, \mathbf{X}^{(m)}, \mathbf{y}_*), \tag{15}$$

where $j = 1, 2, \cdots, J$ and $m = 1, 2, \cdots, M$.

We write this posterior as

$$p^J(\mathbf{x}_*|\mathbf{Y}, \mathbf{X}^{(m)}, \mathbf{y}_*) = \sum_{j=1}^{J} \lambda^{(j,m)} \delta(\mathbf{x}_* - \mathbf{x}^{(j,m)}), \tag{16}$$

where the $\lambda^{(j,m)}$s are weights obtained analogously as the weights in (10).

In the next step, for each m, the samples $\mathbf{x}_*^{(j,m)}$ are classified in one of the classes c_k by using a favorite ML method. For simplicity, let us first assume that this method provides a hard decision that $\mathbf{x}^{(j,m)}$ belongs to one of the classes. Then, from all the classifications $\mathbf{x}_*^{(j,m)}$, $j = 1, 2 \cdots, J$, we estimate the probability $P(c_k|\mathbf{y}_*, \mathbf{Y}, \mathbf{X})$ by

$$\widehat{P}^{(m)}(c_k|\mathbf{y}_*, \mathbf{Y}, \mathbf{X}^{(m)}) = \sum_{j=1}^{J} \lambda^{(j,m)} \mathcal{I}_{c=c_k}, \tag{17}$$

where $\mathcal{I}_{c=c_k}$ is the indicator function.

V. Estimating the dimension of the latent space

In practice, we have no knowledge of the actual dimension of the latent states Q, and we do not know the type of kernel that is good for modeling the data. Let k denote the index of a kernel, e.g., an RBF kernel or an Exponential kernel. This index comes from an index set $\mathcal{K}$, where each index in the set corresponds to a different kernel type. In this subsection, we address the problem of finding the posterior $P(Q, k|\mathbf{Y})$, where $Q = 1, 2, ...D$ and $k \in \mathcal{K}$ simultaneously. For the posterior of k and Q we write

$$P(Q, k|\mathbf{Y}) \propto p(\mathbf{Y}|Q, k)P(Q)P(k), \tag{18}$$

where we assume that Q and k are independent, and

$$p(\mathbf{Y}|Q, k) = \int p(\mathbf{Y}|\mathbf{X}, Q, k)p(\mathbf{X}|Q, k)\mathrm{d}\mathbf{X}. \tag{19}$$

We solve this integral by Monte Carlo integration where we sample $\mathbf{X}$ as before from $q(\mathbf{X}|\mathbf{Y}, Q, k)$, i.e.,

$$\mathbf{X}^{(m)} \sim q(\mathbf{X}|\mathbf{Y}, Q, k). \tag{20}$$

We then have

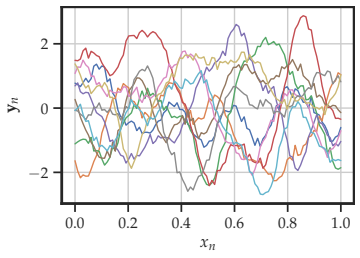
$$p(\mathbf{Y}|Q, k) \approx \sum_{m=1}^{M} \frac{p(\mathbf{Y}|\mathbf{X}^{(m)}, Q, k)p(\mathbf{X}^{(m)}|Q, k)}{q(\mathbf{X}^{(m)}|\mathbf{Y}, Q, k)}. \tag{21}$$

We note that in the above approach the penalty from overestimating the dimension comes from the prior $P(\mathbf{X})$.

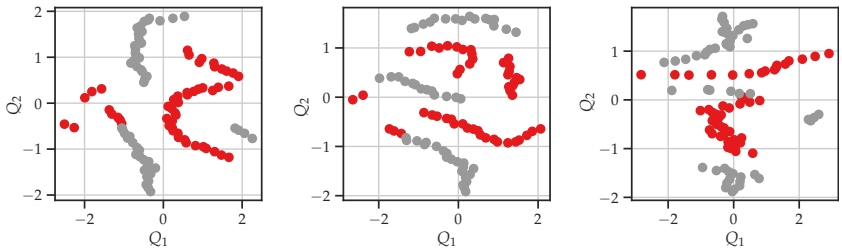
In computing the factors in (21) we need the hyperparameters of the GP and we use the ones obtained while computing $q(\mathbf{X}^{(m)}|\mathbf{Y}, Q, k)$. Another approach would be to have them drawn from their priors and compute the required integral using the Monte Carlo method in the same way as we compute (19) by (21). If the prior $p(Q)$ and $p(k)$ are both uniform, we see that $p(Q, k|\mathbf{Y})$ is proportional to $p(\mathbf{Y}|Q, k)$ only. We start the computations with $Q = 1$, then $Q = 2$, and so on. After some Q, these probabilities will start to drop significantly in value, which will be a sign to stop the process of exploring higher dimensions of $\mathbf{X}$.

VI. Numerical results

In this section, we present the results of classification and dimension estimation.



(a) $D = 10$, $N = 100$, RBF kernel with $l = 0.2$ and $\beta = 0.01$.



(b) Three different members, $q^{(m)}(\mathbf{x})$ of the ensemble GPLVM.

Fig. 1. One-dimensional latent space.

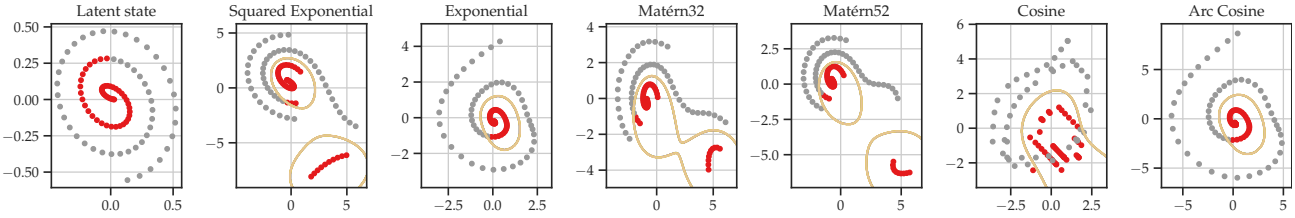


Fig. 2. Two-dimensional latent space, $N = 100$, $D = 10$, $M = 6$, RBF kernel with lengthscales $l = 0.1$ and $\beta = 0.01$

A. Example with classification

In order to depict how different initializations of the variational inference result in different projections of GPLVMs, we created the latent data $\mathbf{x}$, which came from a one-dimensional space. Then we generated the observed data $\mathbf{Y} = \{y_d(\mathbf{X})\}_{d=1}^{D}$ by using a multivariate Gaussian distribution with a covariance matrix constructed using the generated latent vector $\mathbf{x}$ and an RBF kernel. Figure 1(a) shows the one-dimensional latent space and realizations of the generated observations. Each column of $\mathbf{Y}$, $\mathbf{y}_d$, was sampled from the multivariate Gaussian distribution with zero mean and covariance matrix based on $\mathbf{x}$. In the experiment, the latent data $\mathbf{x}$ were 100 equally distanced points with $x_n \leq 0.5$ representing class one and $x_n > 0.5$, class two. Fig. 1(b) demonstrates three posterior means, each obtained by optimizing the log-likelihood of the 10-dimensional data with different initializations.

For the evaluation of the classification performance, we generated latent variables in a two-dimensional space. They followed the spiral shape as shown in Fig. 2 (leftmost plot). These latent variables were converted to 10-dimensional observations. The different colors of the dots represent different classes. We carried out the classification of the observed data by $M = 100$ GPLVMs with random initialization and kernel selection and fused the obtained results using a majority vote. The performances in the presence of low and high noises are summarized in Table 1 with different noise levels. Figure 2 shows six of these projections and their decision boundary obtained by SVM.

B. Example with dimension decision

In this experiment, we generated the training and testing set for latent states and observations with $\sigma_y^2 = 0.1$, the true dimension of the latent states was $Q_0 = 3$, the dimension of observations was $D = 5$, and $\mathbf{K}_{NN,d}$ was an RBF kernel-based covariance matrix with different hyper-parameters $\boldsymbol{\theta}_d$ for each d. The training and testing sets had $N = 200$ samples. Our candidate latent dimensions were from the set $\mathcal{Q} = \{1, 2, 3, 4\}$, and the candidate types of kernels were from the set $\mathcal{K} =\{$ RBF, Exponentiated Quadratic, Matérn32, Matérn52, Exponential$\}$. The ensemble of GPLVMs had $M = 10$ members.

TABLE I
CLASSIFICATION ACCURACY

	$\sigma^2 = 0.01$	$\sigma^2 = 0.1$	$\sigma^2 = 0.2$
GPLVM	0.952	0.734	0.625
EnGPLVM	0.981	0.812	0.773

Figure 3 shows the posterior of $\log p(Q, k|\mathbf{Y})$ under each candidate pair (Q, k). Note that the pair (Q, k) of the latent dimension and kernel type achieved the largest posterior when $Q = 3$ and the applied kernel was the RBF. Thus, the method found the correct kernel of the generative model and the dimension of the latent space. In Fig. 4 we plotted the cumulative log-likelihood under the RBF kernel when the data kept arriving sequentially. The cumulative log-likelihood was defined by

$$CL_N(Q, k) = \sum_{n=1}^{N} \log p(\mathbf{y}_n|Q, k). \tag{22}$$

From the figure, the cumulative log-likelihood under $Q = 3$ gradually dominated the remaining log-likelihoods, which demonstrates that our method can also be applied sequentially.

In another experiment, we compared our approach to estimating the dimension of the latent space with the ARD method. The mechanism of the ARD is to remove the specific

dimensions whose length scales l are larger than a threshold γ. We generated data with the exponential kernel ($k = 5$), and the true latent dimension was $Q_0 = 2$. We found that the maximum of the posterior by our approach was achieved for $Q = Q_0$, whereas the ARD method obtained for the length scales l the following values:

- $Q = 1$: l = [3.0];
- $Q = 2$: l = [5.5, 5.1];
- $Q = 3$: l = [11.0, 12.0, 12.1];
- $Q = 4$: l = [10.4, 11.9, 11.5, 13.0].

From the obtained results, it is not clear how the ARD would pick the dimensions. From the obtained values, it appears that under all Q, there is not much evidence to remove any of the dimensions. Further, the ARD approach is somewhat subjective and can be inaccurate, as shown in this example. By contrast, the proposed approach does not require the setting of subjective thresholds and is based on fundamental principles.

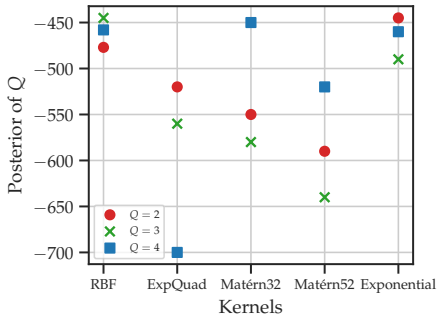


Fig. 3. Log posterior $\log p(Q, k|\mathbf{Y})$ under different Q and k. The correct kernel is RBF and the correct dimension is $Q_0 = 3$.

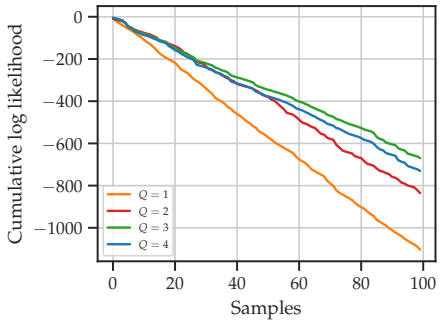


Fig. 4. Cumulative log likelihood for each Q under the RBF kernel.

VII. Conclusion

In this paper, we proposed a latent variable model based on an ensemble of GPLVMs. The ensemble can be constructed in many ways, and a few have been outlined. All the GPLVMs have the same type of kernel in one construction, but they are initialized in different ways. This, in turn, leads to various proposal functions that are used for generating the hidden variables. Another construction is based on distinct kernels, where each GPLVM has a kernel defined by a unique functional form different from the kernel functional forms of the other GPLVMs in the ensemble. In that case, we are sampling from a set of functional kernel forms. Obviously, we can have a mix of the two approaches. Further, we allow for the improvement of the GPLVMs by using the concept of importance sampling. In the paper, we also proposed methods for the classification and estimation of the dimension of the latent space. With numerical experiments, we demonstrated the performance of the proposed methodology and compared it with the performance of single GPLVMs. The results suggest that much can be gained by working with ensembles of GPLVMs.

References

[1] N. Lawrence and A. Hyvärinen, "Probabilistic non-linear principal component analysis with Gaussian process latent variable models." *Journal of Machine Learning Research*, vol. 6, no. 11, 2005.

[2] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT press, 2012.

[3] G. Gundersen, M. Zhang, and B. Engelhardt, "Latent variable modeling with random features," in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 1333–1341.

[4] S. Ahmed, M. Rattray, and A. Boukouvalas, "GrandPrix: scaling up the Bayesian GPLVM for single-cell data," *Bioinformatics*, vol. 35, no. 1, pp. 47–54, 2019.

[5] J. A. Delgado-Guerrero, A. Colomé, and C. Torras, "Sample-efficient robot motion learning using Gaussian process latent variable models," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 314–320.

[6] R. S. Nirwan and N. Bertschinger, "Applications of Gaussian process latent variable models in finance," in *Proceedings of SAI Intelligent Systems Conference*. Springer, 2019, pp. 1209–1221.

[7] M. Titsias and N. D. Lawrence, "Bayesian Gaussian process latent variable model," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 2010, pp. 844–851.

[8] A. C. Damianou, M. K. Titsias, and N. D. Lawrence, "Variational inference for latent variables and uncertain inputs in Gaussian processes," *Journal of Machine Learning Research*, vol. 17, pp. 1–62, 2016.

[9] M. K. Titsias, N. D. Lawrence, and M. Rattray, "Efficient sampling for Gaussian process inference using control variables." in *Advances in Neural Information Processing Systems*. Citeseer, 2008, pp. 1681–1688.

[10] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. Springer, 2004, vol. 2.

[11] R. M. Neal, "Monte Carlo implementation of Gaussian process models for Bayesian regression and classification," *arXiv preprint physics/9701026*, 1997.

[12] E. Snelson and Z. Ghahramani, "Sparse Gaussian processes using pseudo-inputs," *Advances in Neural Information Processing Systems*, vol. 18, p. 1257, 2006.

[13] J. Hensman, A. G. d. G. Matthews, M. Filippone, and Z. Ghahramani, "MCMC for variationally sparse Gaussian processes," *arXiv preprint arXiv:1506.04000*, 2015.

[14] M. M. Dunlop, M. A. Girolami, A. M. Stuart, and A. L. Teckentrup, "How deep are deep Gaussian processes?" *Journal of Machine Learning Research*, vol. 19, no. 54, pp. 1–46, 2018.

[15] M. D. Hoffman, "Learning deep latent Gaussian models with Markov chain Monte Carlo," in *International Conference on Machine Learning*. PMLR, 2017, pp. 1510–1519.

[16] M. Havasi, J. M. Hernández-Lobato, and J. J. Murillo-Fuentes, "Inference in deep Gaussian processes using stochastic gradient hamiltonian monte carlo," *arXiv preprint arXiv:1806.05490*, 2018.

[17] G. E. Box and G. C. Tiao, *Bayesian Inference in Statistical Analysis*. John Wiley & Sons, 2011, vol. 40.

[18] M. F. Bugallo, V. Elvira, L. Martino, D. Luengo, J. Miguez, and P. M. Djuric, "Adaptive importance sampling: The past, the present, and the future," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 60–79, 2017.