

Generalized probabilistic principal component analysis of correlated data

Mengyang Gu

*Department of Statistics and Applied Probability
University of California, Santa Barbara
5511 South Hall
Santa Barbara, CA 93106-3110*

MENGYANG@PSTAT.UCSB.EDU

Weining Shen

*Department of Statistics
University of California, Irvine
2206 Bren Hall
Irvine, CA 92697-1250*

WEININGS@UCI.EDU

Editor: Manfred Opper

Abstract

Principal component analysis (PCA) is a well-established tool in machine learning and data processing. The principal axes in PCA were shown to be equivalent to the maximum marginal likelihood estimator of the factor loading matrix in a latent factor model for the observed data, assuming that the latent factors are independently distributed as standard normal distributions. However, the independence assumption may be unrealistic for many scenarios such as modeling multiple time series, spatial processes, and functional data, where the outcomes are correlated. In this paper, we introduce the generalized probabilistic principal component analysis (GPPCA) to study the latent factor model for multiple correlated outcomes, where each factor is modeled by a Gaussian process. Our method generalizes the previous probabilistic formulation of PCA (PPCA) by providing the closed-form maximum marginal likelihood estimator of the factor loadings and other parameters. Based on the explicit expression of the precision matrix in the marginal likelihood that we derived, the number of the computational operations is linear to the number of output variables. Furthermore, we also provide the closed-form expression of the marginal likelihood when other covariates are included in the mean structure. We highlight the advantage of GPPCA in terms of the practical relevance, estimation accuracy and computational convenience. Numerical studies of simulated and real data confirm the excellent finite-sample performance of the proposed approach.

Keywords: Gaussian process, maximum marginal likelihood estimator, kernel method, principal component analysis, Stiefel manifold

1. Introduction

Principal component analysis (PCA) is one of the oldest and most widely known approaches for dimension reduction. It has been used in many applications, including exploratory data analysis, regression, time series analysis, image processing, and functional data analysis. The most common solution of the PCA is to find a linear projection that transforms the

set of original correlated variables onto a projected space of new uncorrelated variables by maximizing the variation of the projected space (Jolliffe, 2011). This solution, despite its wide use in practice, lacks a probabilistic description of the data.

A probabilistic formulation of the PCA was first introduced by Tipping and Bishop (1999), where the authors considered a Gaussian latent factor model, and then obtained the PCA (principal axes) as the solution of a maximum marginal likelihood problem, where the latent factors were marginalized out. This approach, known as the probabilistic principal component analysis (PPCA), assumes that the latent factors are independently distributed following a standard normal distribution. However, the independence assumption of the factors is usually too restrictive for many applications, where the variables of interest are correlated between different inputs, e.g. times series, images, and spatially correlated data. The latent factor model was extended to incorporate the dependence structure in previous studies. For example, the linear model of coregionalization (LMC) was studied in modeling multivariate outputs of spatially correlated data (Gelfand et al., 2004, 2010), where each factor is modeled by a Gaussian process (GP) to account for the spatial correlation in the data. When the factor loading matrix is shared, the LMC becomes a semiparametric latent factor model, introduced in machine learning literature (Seeger et al., 2005; Alvarez et al., 2012), and was widely applied in emulating computationally expensive computer models with multivariate outputs (Higdon et al., 2008; Fricker et al., 2013), where each factor is modeled by a GP over a set of inputs, such as the physical parameters of the partial differential equations. However, the estimation of factor loading matrix in PCA is not the maximum marginal likelihood estimator when the factors at two inputs are correlated.

In this work, we propose a new approach called the generalized probabilistic principal component analysis (GPPCA), as an extension of the PPCA for the correlated output data. We assume each column of the factor loading matrix is orthonormal for the identifiability purpose. Based on this assumption, we obtain a closed-form solution for the maximum marginal likelihood estimation of the factor loading matrix when the covariance function of the factor processes is shared. This result is an extension of the PPCA for the correlated factors, and the connection between these two approaches is studied. When the covariance functions of the factor processes are different, the maximum marginal likelihood estimation of the factor loading matrix is equivalent to an optimization problem with orthogonal constraints, sometimes referred as the Stiefel manifold. A fast numerical search algorithm on the Stiefel manifold is introduced by Wen and Yin (2013) for the optimization problem.

There are several approaches for estimating the factor loading matrix for the latent factor model and semiparametric latent factor model in the Frequentist and Bayesian literature. One of the most popular approaches for estimating the factor loading matrix is PCA (see e.g., Bai and Ng (2002); Bai (2003); Higdon et al. (2008)). Under the orthonormality assumption for the factor loading vectors, the PCA can be obtained from the maximum likelihood estimator of the factor loading matrix. However, the correlation structure of each factor is not incorporated for the estimation. In Lam et al. (2011) and Lam and Yao (2012), the authors considered estimating the factor loading matrix based on the sample covariance of the output data at the first several time lags when modeling high-dimensional time series. We will numerically compare our approach to the aforementioned Frequentist approaches.

Bayesian approaches have also been widely studied for factor loading matrix estimation. West (2003) points out the connection between PCA and a class of generalized singular

g-priors, and introduces a spike-and-slab prior that induces the sparse factors in the latent factor model assuming the factors are independently distributed. When modeling spatially correlated data, priors are also discussed for the spatially varying factor loading matrices (Gelfand et al., 2004) in LMC. The closed-form marginal likelihood obtained in this work is more computationally feasible than the previous results, as the inverse of the covariance matrix is shown to have an explicit form.

Our proposed method is also connected to the popular kernel approach, which has been used for nonlinear component analysis (Schölkopf et al., 1998) by mapping the output data to a high-dimensional feature space through a kernel function. This method, known as the kernel PCA, is widely applied in various problems, such as the image analysis (Mika et al., 1999) and novelty detection (Hoffmann, 2007). However, we focus on applying the kernel function for capturing the correlation of the outputs at different inputs (e.g. the time point, the location of image pixels or the physical parameters in the PDEs) in this work.

We highlight a few contributions of this paper. First of all, we derive the closed-form maximum marginal likelihood estimator (MMLE) of the factor loading matrix, when the factors are modeled by GPs. Note that our expression of the marginal likelihood (after integrating out the factor processes) is computationally more feasible than the previous result, because the inverse of the covariance matrix is shown to have an explicit form, which makes the computational complexity linear to the number of output variables. Based on this closed-form marginal likelihood, we are able to obtain the MMLE of the other parameters, such as the variance of the noise and kernel parameters, and the predictive distribution of the outcomes. Our second contribution is that we provide a fully probabilistic analysis of the mean and other regression parameters, when some covariates are included in the mean structure of the factor model. The empirical mean of data was often subtracted before applying PPCA and LMC (Tipping and Bishop, 1999; Higdon et al., 2008), which does not quantify the uncertainty when the output is linearly dependent on some covariates. Here we manage to marginalize out the regression parameters in the mean structure explicitly without increasing the computational complexity. Our real data application examples demonstrate the improvements in out-of-sample prediction when the mean structure is incorporated in the data analysis. Lastly, the proposed estimator of the factor loading matrix in GPPCA are closely connected to the PCA and PPCA, and we will discuss how the correlation in the factors affects the estimators of the factor loading matrix and predictive distributions. Both the simulated and real examples show the improved accuracy in estimation and prediction, when the output data are correlated.

The rest of the paper is organized as follows. The main results of the closed-form marginal likelihood and the maximum marginal likelihood estimator of the factor loading matrix are introduced in Section 2.1. In Section 2.2, we provide the maximum marginal likelihood estimator for the noise parameter and kernel parameters, after marginalizing out the factor processes. Section 2.3 discusses the estimators of the factor loading matrix and other parameters when some covariates are included in the model. The comparison between our approach and other approaches in estimating the factor loading matrix is studied in Section 3, with a focus on the connection between GPPCA and PPCA. Simulation results are provided in Section 4 and real data examples are shown in Section 5 and Section 6, respectively. We conclude this work in Section 6. The data and code used in this paper are publicly available (<https://github.com/MengyangGu/GPPCA>).

2. Main results

We state our main results in this section. In section 2.1, we derive a computationally feasible expression of the marginal distribution for the latent factor model after marginalizing out the factor processes, based on which we show the maximum marginal likelihood estimator of the factor loading matrix. In Section 2.2, we discuss the parameter estimation and predictive distribution. We extend our method to study the factor model by allowing the intercept and additional covariates in the mean structure in Section 2.3.

2.1. Generalized probabilistic principal component analysis

To begin with, let $\mathbf{y}(\mathbf{x}) = (y_1(\mathbf{x}), \dots, y_k(\mathbf{x}))^T$ be a k -dimensional real-valued output vector at a p -dimensional input vector \mathbf{x} . Let $\mathbf{Y} = [\mathbf{y}(\mathbf{x}_1), \dots, \mathbf{y}(\mathbf{x}_n)]$ be a $k \times n$ matrix of the observations at inputs $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. In this subsection and the next subsection, we assume that each row of the \mathbf{Y} is centered at zero.

Consider the following latent factor model

$$\mathbf{y}(\mathbf{x}) = \mathbf{A}\mathbf{z}(\mathbf{x}) + \boldsymbol{\epsilon}, \quad (1)$$

where $\boldsymbol{\epsilon} \sim N(0, \sigma_0^2 \mathbf{I}_k)$ is a vector of independent Gaussian noises, with \mathbf{I}_k being the $k \times k$ identity matrix. The $k \times d$ factor loading matrix $\mathbf{A} = [\mathbf{a}_1, \dots, \mathbf{a}_d]$ relates the k -dimensional output to d -dimensional factor processes $\mathbf{z}(\mathbf{x}) = (z_1(\mathbf{x}), \dots, z_d(\mathbf{x}))^T$, where $d \leq k$.

In many applications, each output is correlated. For example, model (1) is widely used in analyzing multiple time series, where $y_l(x)$'s are correlated across different time points for every $l = 1, \dots, k$. Model (1) is also used for multivariate spatially correlated outputs, often referred as the linear model of coregionalization (LMC) (Gelfand et al., 2010). In these studies, each factor is modeled by a zero-mean Gaussian process (GP), meaning that for any set of inputs $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, $\mathbf{Z}_l^T = (z_l(\mathbf{x}_1), \dots, z_l(\mathbf{x}_n))^T$ follows a multivariate normal distribution

$$\mathbf{Z}_l^T \sim \text{MN}(\mathbf{0}, \boldsymbol{\Sigma}_l), \quad (2)$$

where the (i, j) entry of $\boldsymbol{\Sigma}_l$ is parameterized by a covariance function $\sigma_l^2 K_l(\mathbf{x}_i, \mathbf{x}_j)$ for $l = 1, \dots, d$ and $1 \leq i, j \leq n$. We defer the discussion of the kernel to Section 2.2.

Note that the model (1) is unchanged if one replaces the pair $(\mathbf{A}, \mathbf{z}(\mathbf{x}))$ by $(\mathbf{A}\mathbf{E}, \mathbf{E}^{-1}\mathbf{z}(\mathbf{x}))$ for any invertible matrix \mathbf{E} . As pointed out in Lam et al. (2011), only the d -dimensional linear subspace of \mathbf{A} , denoted as $\mathcal{M}(\mathbf{A})$, can be uniquely identified, since $\mathcal{M}(\mathbf{A}) = \mathcal{M}(\mathbf{A}\mathbf{E})$ for any invertible matrix \mathbf{E} . Due to this reason, we assume the columns of \mathbf{A} in model (1) are orthonormal for identifiability purpose (Lam et al., 2011; Lam and Yao, 2012).

Assumption 1

$$\mathbf{A}^T \mathbf{A} = \mathbf{I}_d. \quad (3)$$

Note the Assumption 1 can be relaxed by assuming $\mathbf{A}^T \mathbf{A} = c\mathbf{I}_d$ where c is a positive constant which can potentially depend on k , e.g. $c = k$. As each factor process has the variance σ_l^2 , typically estimated from the data, we thus derive the results based on Assumption 1 herein.

Denote the vectorization of the output $\mathbf{Y}_v = \text{vec}(\mathbf{Y})$ and the $d \times n$ latent factor matrix $\mathbf{Z} = (\mathbf{z}(\mathbf{x}_1), \dots, \mathbf{z}(\mathbf{x}_n))$ at inputs $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$. We first give the marginal distribution of \mathbf{Y}_v (after marginalizing out \mathbf{Z}) with an explicit inverse of the covariance matrix in Lemma 1.

Lemma 1 *Under Assumption 1, the marginal distribution of \mathbf{Y}_v in model (1) is the multivariate normal distribution as follows,*

$$\mathbf{Y}_v \mid \mathbf{A}, \sigma_0^2, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_d \sim \text{MN} \left(\mathbf{0}, \sum_{l=1}^d \boldsymbol{\Sigma}_l \otimes (\mathbf{a}_l \mathbf{a}_l^T) + \sigma_0^2 \mathbf{I}_{nk} \right) \quad (4)$$

$$\sim \text{MN} \left(\mathbf{0}, \sigma_0^2 \left(\mathbf{I}_{nk} - \sum_{l=1}^d (\sigma_0^2 \boldsymbol{\Sigma}_l^{-1} + \mathbf{I}_n)^{-1} \otimes (\mathbf{a}_l \mathbf{a}_l^T) \right)^{-1} \right). \quad (5)$$

The form in (4) appeared in the previous literature (e.g. Gelfand et al. (2004)) and its derivation is given in Appendix B. However, directly computing the marginal likelihood by expression (4) may be expensive, as the covariance matrix is of size $nk \times nk$. Our expression (5) of the marginal likelihood is computationally more feasible than the expression (4), as the inverse of the covariance matrix of \mathbf{Y}_v is derived explicitly in (5). Based on the marginal likelihood in (5), we derive the maximum marginal estimation of \mathbf{A} where the covariance matrix for each latent factor is assumed to be the same as in Theorem 2 below.

Theorem 2 *For model (1), assume $\boldsymbol{\Sigma}_1 = \dots = \boldsymbol{\Sigma}_d = \boldsymbol{\Sigma}$. Under Assumption 1, after marginalizing out \mathbf{Z} , the likelihood function is maximized when*

$$\hat{\mathbf{A}} = \mathbf{U}\mathbf{R}, \quad (6)$$

where \mathbf{U} is a $k \times d$ matrix for the first d principal eigenvectors of $\mathbf{G} = \mathbf{Y}(\sigma_0^2 \boldsymbol{\Sigma}^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T$, and \mathbf{R} is an arbitrary $d \times d$ orthogonal rotation matrix.

By Theorem 2, the solution $\hat{\mathbf{A}}$ is not unique because of the arbitrary rotation matrix. However, the linear subspace of the column space of the estimated factor loading matrix, denoted by $\mathcal{M}(\hat{\mathbf{A}})$, is uniquely determined by (6).

In general, the covariance function of each factor can be different. We are able to express the maximum marginal likelihood estimator as the solution to an optimization problem with the orthogonal constraints, stated in Theorem 3.

Theorem 3 *Under Assumption 1, after marginalizing out \mathbf{Z} , the maximum marginal likelihood estimator of \mathbf{A} in model (1) is*

$$\hat{\mathbf{A}} = \underset{\mathbf{A}}{\operatorname{argmax}} \sum_{l=1}^d \mathbf{a}_l^T \mathbf{G}_l \mathbf{a}_l, \quad \text{s.t.} \quad \mathbf{A}^T \mathbf{A} = \mathbf{I}_d, \quad (7)$$

where $\mathbf{G}_l = \mathbf{Y}(\sigma_0^2 \boldsymbol{\Sigma}_l^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T$.

The subset of matrices \mathbf{A} that satisfies the orthogonal constraint $\mathbf{A}^T \mathbf{A} = \mathbf{I}_d$ is often referred as the *Stiefel manifold*. Unlike the case where the covariance of each factor processes is shared, no closed-form solution of the optimization problem in (7) has been found. A numerical optimization algorithm that preserves the orthogonal constraints in (3) is introduced in Wen and Yin (2013). The main idea of their algorithm is to find the gradient

of the objective function in the tangent space at the current step, and iterates by a curve along the projected negative descent on the manifold. The curvilinear search is applied to find the appropriate step size that guarantees the convergence to a stationary point. We implement this approach to numerically optimize the marginal likelihood to obtain the estimated factor loading matrix in Theorem 3.

We call the method of estimating \mathbf{A} in Theorem 2 and Theorem 3 the generalized probabilistic principal component analysis (GPPCA) of correlated data, which is a direct extension of the PPCA in Tipping and Bishop (1999). Although both approaches obtain the maximum marginal likelihood estimator of the factor loading matrix, after integrating out the latent factors, the key difference is that in GPPCA, the latent factors at different inputs are allowed to be correlated, whereas the latent factors in PPCA are assumed to be independent. A detailed numerical comparison between our method and other approaches including the PPCA will be given in Section 3.

Another nice feature of the proposed GPPCA method is that the estimation of the factor loading matrix can be applied to any covariance structure of the factor processes. In this paper, we use kernels to parameterize the covariance matrix as an illustrative example. There are many other ways to specify the covariance matrix or the inverse of the covariance matrix, such as the Markov random field and the dynamic linear model, and these approaches are readily applicable in our latent factor model (1).

For a function with a p -dimensional input, we use a product kernel to model the covariance for demonstration purposes (Sacks et al., 1989), meaning that for the l th factor,

$$\sigma_l^2 K_l(\mathbf{x}_a, \mathbf{x}_b) = \sigma_l^2 \prod_{m=1}^p K_{lm}(x_{am}, x_{bm}), \quad (8)$$

for any input $\mathbf{x}_a = (x_{a1}, \dots, x_{ap})$ and $\mathbf{x}_b = (x_{b1}, \dots, x_{bp})$, where $K_{lm}(\cdot, \cdot)$ is a one-dimensional kernel function of the l th factor that models the correlation of the m th coordinate of any two inputs.

Some widely used one-dimensional kernel functions include the power exponential kernel and the Matérn kernel. For any two inputs $\mathbf{x}_a, \mathbf{x}_b \in \mathcal{X}$, the Matérn kernel is

$$K_{lm}(x_{am}, x_{bm}) = \frac{1}{2^{\nu_{lm}-1} \Gamma(\nu_{lm})} \left(\frac{|x_{am} - x_{bm}|}{\gamma_{lm}} \right)^{\nu_{lm}} \mathcal{K}_{\nu_{lm}} \left(\frac{|x_{am} - x_{bm}|}{\gamma_{lm}} \right), \quad (9)$$

where $\Gamma(\cdot)$ is the gamma function and $\mathcal{K}_{\nu_{lm}}(\cdot)$ is the modified Bessel function of the second kind with a positive roughness parameter ν_{lm} and a nonnegative range parameter γ_{lm} for $l = 1, \dots, d$ and $m = 1, \dots, p$. The Matérn kernel contains a wide range of different kernel functions. In particular, when $\nu_{lm} = 1/2$, the Matérn kernel becomes the exponential kernel, $K_l(x_{am}, x_{bm}) = \exp(-|x_{am} - x_{bm}|/\gamma_{lm})$, and the corresponding factor process is the Ornstein-Uhlenbeck process, which is a continuous autoregressive process with order 1. When $\nu_{lm} \rightarrow \infty$, the Matérn kernel becomes the Gaussian kernel, i.e., $K_l(x_{am}, x_{bm}) = \exp(-|x_{am} - x_{bm}|^2/\gamma_{lm}^2)$, where the factor process is infinitely differentiable. The Matérn kernel has a closed-form expression when $(2\nu_{lm} + 1)/2 \in \mathbb{N}$. For example, the Matérn kernel with $\nu_{lm} = 5/2$ has the following form

$$K_{lm}(x_{am}, x_{bm}) = \left(1 + \frac{\sqrt{5}|x_{am} - x_{bm}|}{\gamma_{lm}} + \frac{5|x_{am} - x_{bm}|^2}{3\gamma_{lm}^2} \right) \exp \left(-\frac{\sqrt{5}|x_{am} - x_{bm}|}{\gamma_{lm}} \right), \quad (10)$$

for any inputs \mathbf{x}_a and \mathbf{x}_b with $l = 1, \dots, d$ and $m = 1, \dots, p$. In this work, we use the Matérn kernel in (10) for the simulation and real data analysis for demonstration purposes. Specifying a sensible kernel function depends on real applications and our results in this work apply to all commonly used kernel functions. We will also numerically compare different approaches when the kernel function is misspecified in Appendix C.

2.2. Parameter estimation and predictive distribution

The probabilistic estimation of the factor loading matrix depends on the variance of the noise and the covariances of the factor processes. We assume that the covariances of the factors are parameterized by a product of the kernel functions for demonstration purposes. We discuss the parameter estimation and the predictive distribution of the data in this subsection. The probabilistic estimation of the factor loading matrix in the GPPCA can be also applied when the covariances of the factors are specified or estimated in other ways.

We denote $\tau_l := \frac{\sigma_l^2}{\sigma_0^2}$ as the signal's variance to noise ratio (SNR) for the l th factor process, as a transformation of σ_l^2 in (8). The maximum likelihood estimator of σ_0^2 has a closed form expression using this parameterization. Furthermore, let the correlation matrix of the k th factor process be \mathbf{K}_l with the (i, j) th term being $K_l(\mathbf{x}_i, \mathbf{x}_j)$. After this transformation, the estimator of \mathbf{A} in Theorems 2 and 3 becomes a function of the parameters $\boldsymbol{\tau} = (\tau_1, \dots, \tau_d)$ and $\boldsymbol{\gamma} = (\gamma_1, \dots, \gamma_d)$. Under Assumption 1, after marginalizing out \mathbf{Z} , the maximum likelihood estimator of σ_0^2 becomes a function of \mathbf{A} , $\boldsymbol{\tau}$ and $\boldsymbol{\gamma}$ as

$$\hat{\sigma}_0^2 = \frac{\hat{S}^2}{nk}, \quad (11)$$

where $\hat{S}^2 = \text{tr}(\mathbf{Y}^T \mathbf{Y}) - \sum_{l=1}^d \mathbf{a}_l^T \mathbf{Y} (\tau_l^{-1} \mathbf{K}_l^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T \mathbf{a}_l$. Ignoring the constants, the likelihood of $\boldsymbol{\tau}$ and $\boldsymbol{\gamma}$ by plugging $\hat{\mathbf{A}}$ and $\hat{\sigma}_0^2$ satisfies

$$L(\boldsymbol{\tau}, \boldsymbol{\gamma} \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\sigma}_0^2) \propto \left\{ \prod_{l=1}^d |\tau_l \mathbf{K}_l + \mathbf{I}_n|^{-1/2} \right\} |\hat{S}^2|^{-nk/2}. \quad (12)$$

The derivation of Equation (12) is given in the Appendix B. Since there is no closed-form expression for the parameter estimates in the kernels, one often numerically maximizes the Equation (12) to estimate these parameters

$$(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\gamma}}) := \underset{(\boldsymbol{\tau}, \boldsymbol{\gamma})}{\text{argmax}} L(\boldsymbol{\tau}, \boldsymbol{\gamma} \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\sigma}_0^2). \quad (13)$$

After obtaining $\hat{\sigma}_0^2$ and $\hat{\boldsymbol{\tau}}$ from (11) and (13), respectively, we transform the expressions back to get the estimator of σ_l^2 as

$$\hat{\sigma}_l^2 = \hat{\tau}_l \hat{\sigma}_0^2,$$

for $l = 1, \dots, d$. Since both the estimator of $\hat{\sigma}_0^2$ and $\hat{\mathbf{A}}$ in Theorem 2 and 3 can be expressed as a function of $(\boldsymbol{\tau}, \boldsymbol{\gamma})$, in each iteration, one can use the Newton's method (Nocedal, 1980) to find $(\boldsymbol{\tau}, \boldsymbol{\gamma})$ based on the likelihood in (12), after plugging the estimator of $\hat{\sigma}_0^2$ and $\hat{\mathbf{A}}$.

We have a few remarks regarding the expressions in (11) and (13). First, under Assumption 1, the likelihood of $(\boldsymbol{\tau}, \boldsymbol{\gamma})$ in (12) can also be obtained by marginalizing out σ_0^2 using the objective prior $\pi(\sigma_0^2) \propto 1/\sigma_0^2$, instead of maximizing over σ_0^2 .

Second, consider the first term at the right hand side of (11). As each row of \mathbf{Y} has a zero mean, let $\mathbf{S}_0 := \mathbf{Y}\mathbf{Y}^T/n = \sum_{i=1}^n \mathbf{y}(\mathbf{x}_i)\mathbf{y}(\mathbf{x}_i)^T/n$, be the sample covariance matrix for $\mathbf{y}(\mathbf{x}_i)$. One has $\text{tr}(\mathbf{Y}\mathbf{Y}^T) = n \sum_{i=1}^k \lambda_{0i}$, where λ_{0i} is the i th eigenvalue of \mathbf{S}_0 . The second term at the right hand side of (11) is the variance explained by the projection. In particular, when the conditions in Theorem 2 hold, i.e. $\Sigma_1 = \dots = \Sigma_d$, one has $\sum_{l=1}^d \hat{\mathbf{a}}_l^T \mathbf{Y} (\tau_l^{-1} \mathbf{K}_l^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T \hat{\mathbf{a}}_l = n \sum_{l=1}^d \hat{\lambda}_l$, where $\hat{\lambda}_l$ is the l th largest eigenvalues of $\mathbf{Y} (\sigma_0^2 \Sigma^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T/n$. The estimation of the noise is then the average variance lost in the projection. Note that the projection in the GPPCA takes into account the correlation of the factor processes, whereas the projection in the PPCA assumes the independent factors. This difference makes the GPPCA more accurate in estimating the subspace of the factor loading matrix when the factors are correlated, as shown in various numerical examples in Section 4.

Thirdly, although the model in (1) is regarded as a nonseparable model (Fricker et al., 2013), the computational complexity of our algorithm is the same with that for the separable model (Gu and Berger, 2016; Conti and O’Hagan, 2010). Instead of inverting an $nk \times nk$ covariance matrix, the expression of the likelihood in (12) allows us to proceed in the same way when the covariance matrix for each factor has a size of $n \times n$. The number of computational operations of the likelihood is at most $\max(O(dn^3), O(kn^2))$, which is much smaller than the $O(n^3k^3)$ for inverting an $nk \times nk$ covariance matrix, because one often has $d \ll k$. When the input is one-dimensional and the Matérn kernel in (9) is used, the computational operations are only $O(dkn)$ for computing the likelihood in (12) without any approximation (see e.g. Hartikainen and Sarkka (2010); Gu and Xu (2019)). We implement this algorithm in the `FastGaSP` package available on CRAN.

Note that the estimator in (12) is known as the Type II maximum likelihood estimator, which is widely used in estimating the kernel parameters. When the number of the observations is small, the estimator in (12) is not robust, in the sense that the estimated range parameters can be very small or very large, which makes the covariance matrix either a diagonal matrix or a singular matrix. This might be unsatisfactory in certain applications, such as emulating computationally expensive computer models (Oakley, 1999). An alternative way is to use the maximum marginal posterior estimation that prevents the two unsatisfying scenarios of the estimated covariance matrix. We refer to Gu et al. (2018) and Gu (2019) for the theoretical properties of the maximum marginal posterior estimation and an R package is available on CRAN (Gu et al. (2019)).

Given the parameter estimates, we can also obtain the predictive distribution for the outputs. Let $\hat{K}_l(\cdot, \cdot)$ be the l th kernel function after plugging the estimates $\hat{\gamma}_l$ and let $\hat{\Sigma}_l$ be the estimator of the covariance matrix for the l th factor, where the (i, j) element of $\hat{\Sigma}_l$ is $\hat{\sigma}_l^2 \hat{K}_l(\mathbf{x}_i, \mathbf{x}_j)$, with $1 \leq i, j \leq n$ and $l = 1, \dots, d$. We have the following predictive distribution for the output at any given input.

Theorem 4 *Under the Assumption 1, for any \mathbf{x}^* , one has*

$$\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\gamma}, \hat{\sigma}^2, \hat{\sigma}_0^2 \sim \text{MN} \left(\hat{\boldsymbol{\mu}}^*(\mathbf{x}^*), \hat{\boldsymbol{\Sigma}}^*(\mathbf{x}^*) \right),$$

where

$$\hat{\boldsymbol{\mu}}^*(\mathbf{x}^*) = \hat{\mathbf{A}}\hat{\mathbf{z}}(\mathbf{x}^*), \tag{14}$$

with $\hat{\mathbf{z}}(\mathbf{x}^*) = (\hat{z}_1(\mathbf{x}^*), \dots, \hat{z}_d(\mathbf{x}^*))^T$, with $\hat{z}_l(\mathbf{x}^*) = \hat{\Sigma}_l^T(\mathbf{x}^*)(\hat{\Sigma}_l + \hat{\sigma}_0^2 \mathbf{I}_n)^{-1} \mathbf{Y}^T \hat{\mathbf{a}}_l$, $\hat{\Sigma}_l(\mathbf{x}^*) = \hat{\sigma}_l^2 (\hat{K}_l(\mathbf{x}_1, \mathbf{x}^*), \dots, \hat{K}_l(\mathbf{x}_n, \mathbf{x}^*))^T$ for $l = 1, \dots, d$, and

$$\hat{\Sigma}^*(\mathbf{x}^*) = \hat{\mathbf{A}} \hat{\mathbf{D}}(\mathbf{x}^*) \hat{\mathbf{A}}^T + \hat{\sigma}_0^2 (\mathbf{I}_k - \hat{\mathbf{A}} \hat{\mathbf{A}}^T), \quad (15)$$

with $\hat{\mathbf{D}}(\mathbf{x}^*)$ being a diagonal matrix, and its l th diagonal term, denoted as $\hat{D}_l(\mathbf{x}^*)$, has the following expression

$$\hat{D}_l(\mathbf{x}^*) = \hat{\sigma}_l^2 \hat{K}_l(\mathbf{x}^*, \mathbf{x}^*) + \hat{\sigma}_0^2 - \hat{\Sigma}_l^T(\mathbf{x}^*) \left(\hat{\Sigma}_l + \hat{\sigma}_0^2 \mathbf{I}_n \right)^{-1} \hat{\Sigma}_l(\mathbf{x}^*),$$

for $l = 1, \dots, d$.

Next we give the posterior distribution of \mathbf{AZ} in Corollary 5.

Corollary 5 (Posterior distribution of \mathbf{AZ}) *Under the Assumption (1), the posterior distribution of \mathbf{AZ} is*

$$(\mathbf{AZ} \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\gamma}, \hat{\sigma}^2, \hat{\sigma}_0^2) \sim \text{MN} \left(\hat{\mathbf{A}} \hat{\mathbf{Z}}, \hat{\sigma}_0^2 \sum_{l=1}^d \hat{\mathbf{D}}_l \otimes \hat{\mathbf{a}}_l \hat{\mathbf{a}}_l^T \right),$$

where $\hat{\mathbf{Z}} = (\hat{\mathbf{Z}}_1^T, \dots, \hat{\mathbf{Z}}_d^T)^T$, $\hat{\mathbf{Z}}_l^T = \hat{\Sigma}_l (\hat{\Sigma}_l + \hat{\sigma}_0^2 \mathbf{I}_n)^{-1} \mathbf{Y}^T \hat{\mathbf{a}}_l$, and $\hat{\mathbf{D}}_l = \left(\sigma_0^2 \hat{\Sigma}_l^{-1} + \mathbf{I}_n \right)^{-1}$, for $l = 1, \dots, d$.

The Corollary 5 is a direct consequence of Theorem 4, so the proof is omitted. Note that the uncertainty of the parameters and the factor loading matrix are not taken into consideration for predictive distribution of $\mathbf{Y}(\mathbf{x}^*)$ in Theorem 4 and the posterior distribution of \mathbf{AZ} in Corollary 5, because of the use of the plug-in estimator for $(\mathbf{A}, \sigma_0^2, \sigma^2, \gamma)$. The resulting posterior credible interval may be narrower than it should be when the sample size is small to moderate. The uncertainty in \mathbf{A} and other model parameters could be obtained by Bayesian analysis with a prior placed on these parameters for these scenarios.

2.3. Mean structure

In many applications, the outputs are not centered at zero. For instance, Bayarri et al. (2009) and Gu and Berger (2016) studied emulating the height of the pyroclastic flow generated from TITAN2D computer model, where the flow volume in the chamber is positively correlated to height of the flow at each spatial coordinate. Thus, modeling the flow volume as a covariate in the mean function typically improves the accuracy of the emulator. When \mathbf{Y} is not centered around zero, one often subtracts the mean of each row of \mathbf{Y} before the inference (Higdon et al., 2008; Paulo et al., 2012). The full Bayesian analysis of the regression parameters are discussed in coregionalization models of multivariate spatially correlated data (see e.g. Gelfand et al. (2004)) using the Markov Chain Monte Carlo (MCMC) algorithm, but the computation may be too complex to implement in many studies.

Consider the latent factor model with a mean structure for a k -dimensional output vector at the input \mathbf{x} ,

$$\mathbf{y}(\mathbf{x}) = (\mathbf{h}(\mathbf{x})\mathbf{B})^T + \mathbf{Az}(\mathbf{x}) + \epsilon, \quad (16)$$

where $\mathbf{h}(\mathbf{x}) := (h_1(\mathbf{x}), \dots, h_q(\mathbf{x}))$ is $1 \times q$ known mean basis function related to input \mathbf{x} and possibly other covariates, $\mathbf{B} = (\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_k)$ is a $q \times k$ matrix of the regression parameters. The regression parameters could be different for each row of the outcomes, and $\boldsymbol{\epsilon} \sim N(0, \sigma_0^2 \mathbf{I}_k)$ is a vector of the independent Gaussian noises, with \mathbf{I}_k being the $k \times k$ identity matrix.

For any set of inputs $\{\mathbf{x}_1, \dots, \mathbf{x}_n\}$, we assume that $\mathbf{Z}_l = (z_l(\mathbf{x}_1), \dots, z_l(\mathbf{x}_n))$ follows a multivariate normal distribution

$$\mathbf{Z}_l^T \sim \text{MN}(\mathbf{0}, \boldsymbol{\Sigma}_l), \quad (17)$$

where the (i, j) th entry of $\boldsymbol{\Sigma}_l$ is parameterized by $K_l(\mathbf{x}_i, \mathbf{x}_j)$ for $l = 1, \dots, d$ and $1 \leq i, j \leq n$.

Denote \mathbf{H} the $n \times q$ matrix with (i, j) th term being $h_j(\mathbf{x}_i)$ for $1 \leq i \leq n$ and $q < n$. We let $n > q$ and assume \mathbf{H} is a full rank matrix. Further denote $\mathbf{M} = \mathbf{I} - \mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T$. We apply a Bayesian approach for the regression parameters by assuming the objective prior $\pi(\mathbf{B}) \propto 1$ (Berger et al., 2001, 2009). We first marginalize out \mathbf{B} and then marginalize out \mathbf{Z} to obtain the marginal likelihood for estimating the other parameters.

Lemma 6 *Let the prior of the regression parameters be $\pi(\mathbf{B}) \propto 1$. Under Assumption 1, after marginalizing out \mathbf{B} and \mathbf{Z} , the maximum likelihood estimator for σ_0^2 is*

$$\hat{\sigma}_0^2 = \frac{S_M^2}{k(n-q)}, \quad (18)$$

where $S_M^2 = \text{tr}(\mathbf{Y} \mathbf{M} \mathbf{Y}^T) - \sum_{l=1}^d \mathbf{a}_l^T \mathbf{Y} \mathbf{M} (\mathbf{M} + \tau_l^{-1} \mathbf{K}_l^{-1})^{-1} \mathbf{M} \mathbf{Y}^T \mathbf{a}_l$. Moreover, the marginal density of the data satisfies

$$p(\mathbf{Y} \mid \mathbf{A}, \boldsymbol{\tau}, \boldsymbol{\gamma}, \hat{\sigma}_0^2) \propto \left\{ \prod_{l=1}^d |\tau_l \mathbf{K}_l + \mathbf{I}_n|^{-1/2} |\mathbf{H}^T (\tau_l \mathbf{K}_l + \mathbf{I}_n)^{-1} \mathbf{H}|^{-\frac{1}{2}} \right\} |S_M^2|^{-\binom{k(n-q)}{2}}. \quad (19)$$

Remark 7 *Under Assumption 1, the MLE of $(\boldsymbol{\tau}, \boldsymbol{\gamma})$ in (19) is equivalent to the maximum marginal likelihood estimator by marginalizing out both \mathbf{B} and σ_0^2 using the objective prior $\pi(\mathbf{B}, \sigma_0^2) \propto 1/\sigma_0^2$, instead of maximizing over σ_0^2 .*

Since there is no closed-form expression for the parameters $(\boldsymbol{\tau}, \boldsymbol{\gamma})$ in the kernels, one can numerically maximize the Equation (19) to estimate \mathbf{A} and other parameters.

$$\hat{\mathbf{A}} = \underset{\mathbf{A}}{\text{argmax}} \sum_{l=1}^d \mathbf{a}_l^T \mathbf{G}_{l,M} \mathbf{a}_l, \quad \text{s.t.} \quad \mathbf{A}^T \mathbf{A} = \mathbf{I}_d, \quad (20)$$

$$(\hat{\boldsymbol{\tau}}, \hat{\boldsymbol{\gamma}}) = \underset{(\boldsymbol{\tau}, \boldsymbol{\gamma})}{\text{argmax}} p(\mathbf{Y} \mid \hat{\mathbf{A}}, \boldsymbol{\tau}, \boldsymbol{\gamma}). \quad (21)$$

When $\boldsymbol{\Sigma}_1 = \dots = \boldsymbol{\Sigma}_d$, the closed-form expression of $\hat{\mathbf{A}}$ can be obtained similarly in Theorem 2. In general, we can use the approach in Wen and Yin (2013) for solving the optimization problem in (20). After obtaining $\hat{\boldsymbol{\tau}}$ and $\hat{\sigma}_0^2$, we transform them to get $\hat{\sigma}_l^2 = \hat{\tau}_l \hat{\sigma}_0^2$ for $l = 1, \dots, d$.

Let $\hat{\boldsymbol{\Sigma}}_l$ be a matrix with the (i, j) -term as $\hat{\sigma}_l^2 \hat{K}_l(\mathbf{x}_i, \mathbf{x}_j)$, where $\hat{K}_l(\mathbf{x}_i, \mathbf{x}_j)$ is the kernel function after plugging the estimator $\hat{\gamma}_l$ for $1 \leq l \leq d$. We first marginalize out \mathbf{B} and then marginalize out \mathbf{Z} . The rest of the parameters are estimated by the maximum marginal likelihood estimator by (18), (20) and (21) in the predictive distribution given below.

Theorem 8 *Under the Assumption 1 and assume the objective prior $\pi(\mathbf{B}) \propto 1$. After marginalizing out \mathbf{B} , \mathbf{Z} , and plugging in the maximum marginal likelihood estimator of $(\mathbf{A}, \boldsymbol{\gamma}, \boldsymbol{\sigma}^2, \sigma_0^2)$, the predictive distribution of model (16) for any \mathbf{x}^* is*

$$\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\sigma}}^2, \hat{\sigma}_0^2 \sim \text{MN} \left(\hat{\boldsymbol{\mu}}_M^*(\mathbf{x}^*), \hat{\boldsymbol{\Sigma}}_M^*(\mathbf{x}^*) \right).$$

Here

$$\hat{\boldsymbol{\mu}}_M^*(\mathbf{x}^*) = (\mathbf{h}(\mathbf{x}^*)\hat{\mathbf{B}})^T + \hat{\mathbf{A}}\hat{\mathbf{z}}_M(\mathbf{x}^*), \quad (22)$$

$$\hat{\boldsymbol{\Sigma}}_M^*(\mathbf{x}^*) = \hat{\mathbf{A}}\hat{\mathbf{D}}_M(\mathbf{x}^*)\hat{\mathbf{A}}^T + \hat{\sigma}_0^2(1 + \mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}^T(\mathbf{x}^*))(\mathbf{I}_k - \hat{\mathbf{A}}\hat{\mathbf{A}}^T), \quad (23)$$

where $\hat{\mathbf{B}} = (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T(\mathbf{Y} - \hat{\mathbf{A}}\hat{\mathbf{Z}}_M)^T$, $\hat{\mathbf{Z}}_M = (\hat{\mathbf{Z}}_{1,M}^T, \dots, \hat{\mathbf{Z}}_{d,M}^T)^T$ with $\hat{\mathbf{Z}}_{l,M} = \mathbf{a}_l^T\mathbf{Y}\mathbf{M}(\hat{\boldsymbol{\Sigma}}_l\mathbf{M} + \hat{\sigma}_0^2\mathbf{I}_n)^{-1}\hat{\boldsymbol{\Sigma}}_l$, $\hat{\mathbf{z}}_M(\mathbf{x}^*) = (\hat{z}_{1,M}(\mathbf{x}^*), \dots, \hat{z}_{d,M}(\mathbf{x}^*))^T$ with $\hat{z}_{l,M}(\mathbf{x}^*) = \hat{\boldsymbol{\Sigma}}_l^T(\mathbf{x}^*)(\hat{\boldsymbol{\Sigma}}_l\mathbf{M} + \hat{\sigma}_0^2\mathbf{I}_n)^{-1}\mathbf{M}\mathbf{Y}\mathbf{a}_l$, for $l = 1, \dots, d$, and $\hat{\mathbf{D}}_M(\mathbf{x}^*)$ is a diagonal matrix with the l th term:

$$\begin{aligned} \hat{D}_{l,M}(\mathbf{x}^*) &= \hat{\sigma}_l^2\hat{K}_l(\mathbf{x}^*, \mathbf{x}^*) + \hat{\sigma}_0^2 - \hat{\boldsymbol{\Sigma}}_l^T(\mathbf{x}^*)\tilde{\boldsymbol{\Sigma}}_l^{-1}\hat{\boldsymbol{\Sigma}}_l(\mathbf{x}^*) \\ &\quad + (\mathbf{h}^T(\mathbf{x}^*) - \mathbf{H}^T\tilde{\boldsymbol{\Sigma}}_l^{-1}\hat{\boldsymbol{\Sigma}}_l(\mathbf{x}^*))^T(\mathbf{H}^T\tilde{\boldsymbol{\Sigma}}_l^{-1}\mathbf{H})^{-1}(\mathbf{h}^T(\mathbf{x}^*) - \mathbf{H}^T\tilde{\boldsymbol{\Sigma}}_l^{-1}\hat{\boldsymbol{\Sigma}}_l(\mathbf{x}^*)), \end{aligned}$$

with $\tilde{\boldsymbol{\Sigma}}_l = \hat{\boldsymbol{\Sigma}}_l + \hat{\sigma}_0^2\mathbf{I}_n$ for $l = 1, \dots, d$.

In Theorem 8, the estimated mean parameters are $\hat{\mathbf{B}} = \mathbb{E}[\mathbf{B} \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\sigma}}^2, \hat{\sigma}_0^2]$, which could be used for inferring the trend of some given covariates (e.g. the gridded temperature example in Section 5.2).

Denote $\mathbf{Y}(\mathbf{x}^*) = (\mathbf{Y}_1^T(\mathbf{x}^*), \mathbf{Y}_2^T(\mathbf{x}^*))^T$, where $\mathbf{Y}_1(\mathbf{x}^*)$ and $\mathbf{Y}_2(\mathbf{x}^*)$ are two vectors of dimensions k_1 and k_2 ($k_1 + k_2 = k$), respectively. Assuming the same conditions in Theorem 8 hold, if one observes both $\mathbf{Y}_1(\mathbf{x}^*)$ and \mathbf{Y} , the predictive distribution of $\mathbf{Y}_2(\mathbf{x}^*)$ follows

$$\mathbf{Y}_2(\mathbf{x}^*) \mid \mathbf{Y}_1(\mathbf{x}^*), \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\sigma}}^2, \hat{\sigma}_0^2 \sim \text{MN} \left(\hat{\boldsymbol{\mu}}_{M,2|1}^*(\mathbf{x}^*), \hat{\boldsymbol{\Sigma}}_{M,2|1}^*(\mathbf{x}^*) \right), \quad (24)$$

where $\hat{\boldsymbol{\mu}}_{M,2|1}^*(\mathbf{x}^*) = \hat{\boldsymbol{\mu}}_{M,2}^*(\mathbf{x}^*) + \hat{\boldsymbol{\Sigma}}_{M,12}^*(\mathbf{x}^*)^T\hat{\boldsymbol{\Sigma}}_{M,11}^*(\mathbf{x}^*)^{-1}(\mathbf{Y}_1(\mathbf{x}^*) - \hat{\boldsymbol{\mu}}_{M,1}^*(\mathbf{x}^*))$ with $\hat{\boldsymbol{\mu}}_{M,1}^*(\mathbf{x}^*)$ and $\hat{\boldsymbol{\mu}}_{M,2}^*(\mathbf{x}^*)$ being the first k_1 and last k_2 entries of $\hat{\boldsymbol{\mu}}_M^*(\mathbf{x}^*)$; $\hat{\boldsymbol{\Sigma}}_{M,2|1}^*(\mathbf{x}^*) = \hat{\boldsymbol{\Sigma}}_{M,22}^*(\mathbf{x}^*) - \hat{\boldsymbol{\Sigma}}_{M,12}^*(\mathbf{x}^*)^T\hat{\boldsymbol{\Sigma}}_{M,11}^*(\mathbf{x}^*)^{-1}\hat{\boldsymbol{\Sigma}}_{M,12}^*(\mathbf{x}^*)$ with $\hat{\boldsymbol{\Sigma}}_{M,11}$, $\hat{\boldsymbol{\Sigma}}_{M,22}$ and $\hat{\boldsymbol{\Sigma}}_{M,12}$ being the first $k_1 \times k_1$, last $k_2 \times k_2$ entries in the diagonals and $k_1 \times k_2$ entries in the off-diagonals of $\hat{\boldsymbol{\Sigma}}_M^*$, respectively.

3. Comparison to other approaches

In this section, we compare our method to various other frequently used approaches and discuss their connections and differences using examples. First of all, note that the maximum likelihood estimator (MLE) of the factor loading matrix \mathbf{A} under the Assumption 1 is $\mathbf{U}_0\mathbf{R}$ (without marginalizing out \mathbf{Z}), where \mathbf{U}_0 is the first d ordered eigenvectors of $\mathbf{Y}\mathbf{Y}^T$ and \mathbf{R} is an arbitrary orthogonal rotation matrix. This corresponds to the solution of principal component analysis, which is widely used in the literature for the inference of the latent factor model. For example, Bai and Ng (2002) and Bai (2003) assume that $\mathbf{A}^T\mathbf{A} = k\mathbf{I}_d$ and estimate \mathbf{A} by $\sqrt{k}\mathbf{U}_0$ in modeling high-dimensional time series. The estimation of factor loading matrix by the PCA is also applied in emulating multivariate outputs from a

computer model (Higdon et al., 2008), where the factor loading matrix is estimated by the singular value decomposition of the standardized output matrix.

The principal axes of the PCA are the same with those obtained from the PPCA, in which the factor loading matrix is estimated by the maximum marginal likelihood, after marginalizing out the independent and normally distributed factors (Tipping and Bishop, 1999). The estimator of the factor loadings is found to be the first d columns of $\tilde{\mathbf{U}}_0(\tilde{\mathbf{D}}_0 - \sigma_0^2 \mathbf{I}_d) \mathbf{R}$, where $\tilde{\mathbf{D}}_0$ is a diagonal matrix whose l th diagonal term is the l th largest eigenvalues of $\mathbf{Y}\mathbf{Y}^T/n$ and \mathbf{R} is an arbitrary $d \times d$ orthogonal rotation matrix.

The PPCA gives a probabilistic model of the PCA by modeling \mathbf{Z} via independent normal distributions. However, when outputs are correlated across different inputs, modeling the factor as independent normal distributions may not be sensible in some applications. In comparison, the factors are allowed to be correlated in GPPCA; and we marginalize the factors out to estimate \mathbf{A} to account for the uncertainty. This is why our approach can be viewed as a generalized approach of the PPCA for the correlated data.

The second observation is that the estimation of the factor loading matrix in the PCA or PPCA typically assumes the data are standardized. However, the standardization process could cause a loss of information and the uncertainty in the standardization is typically not considered. This problem is also resolved by GPPCA, where the intercept and other covariates can be included in the model and the mean parameters can be marginalized out in estimating the factor loading matrix, as discussed in Section 2.3.

Next we numerically illustrate the difference between the GPPCA and PCA using Example 1.

Example 1 *The data is sampled from the model (1) with the shared covariance matrix $\Sigma_1 = \Sigma_2 = \Sigma$, where x is equally spaced from 1 to n and the kernel function is assumed to follow (10) with $\gamma = 100$ and $\sigma^2 = 1$. We choose $k = 2$, $d = 1$ and $n = 100$. Two scenarios are implemented with $\sigma_0^2 = 0.01$ and $\sigma_0^2 = 1$, respectively. The parameters $(\sigma_0^2, \sigma^2, \gamma)$ are assumed to be unknown and estimated from the data.*

Note the linear subspace spanned from the column space of estimated loading matrix by the PCA or PPCA is the same, which is $\mathcal{M}(\mathbf{U}_0)$. Thus we only compare the GPPCA to the PCA in Figure 1 where \mathbf{A} is a two-dimensional vector generated from a uniform distribution on the Stiefel manifold (Hoff, 2013). The signal to noise ratio (SNR) is $\tau = 10^2$ and $\tau = 1$ for the upper and lower panels in Figure 1, respectively.

From Figure 1, we observe that when the SNR is large, two rows of the outputs are strongly correlated, as shown in the upper left panel, with the empirical correlation being around -0.83 between two rows of the output \mathbf{Y} . The estimated subspaces by the PCA and GPPCA both match the true \mathbf{A} equally well in this scenario, shown in the upper right panel. When the variance of the noise gets large, the outputs are no longer very correlated. For example, the empirical correlation between two simulated output variables is only around -0.18 . As a result, the angle between the estimated subspace and the column space of \mathbf{A} by the PCA is large, as shown in the lower right panel.

The GPPCA by Theorem 2 essentially transforms the output by $\tilde{\mathbf{Y}} = \mathbf{Y}\mathbf{L}$, graphed in the middle panels, where $\mathbf{L} = \mathbf{U}\mathbf{D}^{1/2}$ with \mathbf{U} and \mathbf{D} being a matrix of eigenvectors and a diagonal matrix of the eigenvalues from the eigendecomposition of $(\hat{\sigma}_0^2 \hat{\Sigma}^{-1} + \mathbf{I}_n)^{-1}$, respectively, where variance parameter and kernel parameter are estimated by the MMLE

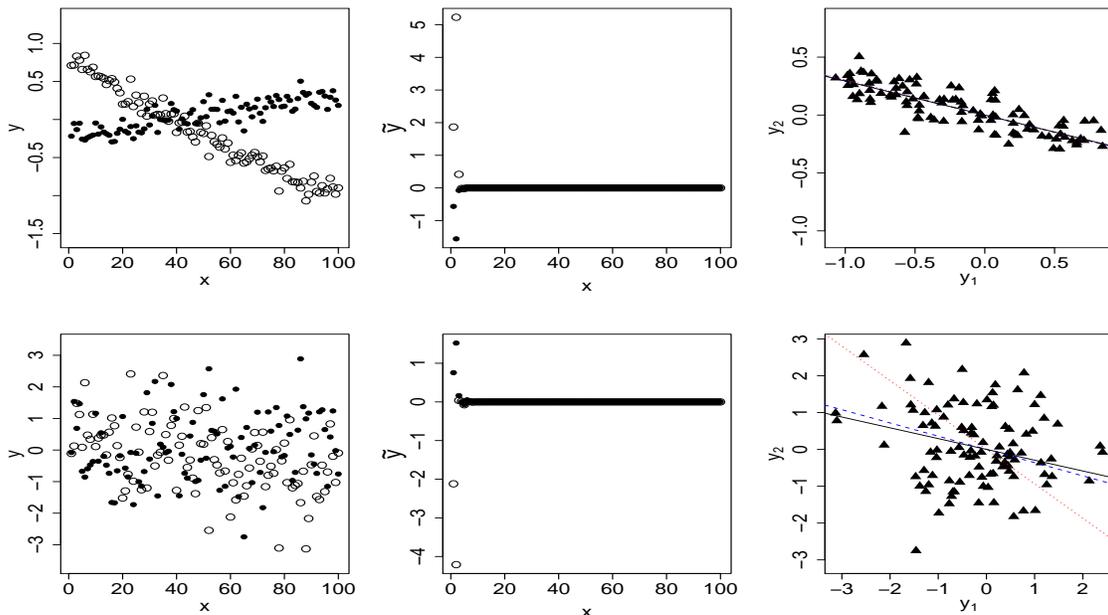


Figure 1: Estimation of the factor loading matrix by the PCA and GPPCA for Example 1 with the variance noise being $\sigma_0^2 = 0.01$ and $\sigma_0^2 = 1$, graphed in the upper panels and lower panels, respectively. The circles and dots are the first and second rows of \mathbf{Y} in the left panel, and of the transformed output $\tilde{\mathbf{Y}} = \mathbf{Y}\mathbf{L}$ in the middle panels, where $\mathbf{L} = \mathbf{U}\mathbf{D}^{1/2}$ with \mathbf{U} being the eigenvectors and the diagonals of \mathbf{D} being the eigenvalues of the eigendecomposition of $(\hat{\sigma}_0^2 \hat{\Sigma}^{-1} + \mathbf{I}_n)^{-1}$, where the (i, j) -term of $\hat{\Sigma}$ is $\hat{\sigma}^2 \hat{K}(\mathbf{x}_i, \mathbf{x}_j)$ by plugging the estimated range parameter $\hat{\gamma}$. The circles and dots in the middle panels almost overlap when x is slightly larger than 0. In the right panels, the black solid lines, red dotted lines and blue dash lines are the subspace of \mathbf{A} , the first eigenvector of \mathbf{U}_0 and the first eigenvector of \mathbf{G} in Theorem 2, respectively, with the black triangles being the outputs. The black, blue and red lines almost overlap in the upper right panel.

discussed in Section 2.2. The two rows of the transformed outputs are strongly correlated, shown in the middle panels. The empirical correlation between two rows of the transformed outputs graphed in the lower panel is about -0.99 , even though the variance of the noise is as large as the variance of the signal. The subspace by the GPPCA is equivalent to the first eigenvector of the transformed output for this example, and it is graphed as the blue dashed curves in the right panels. The estimated subspace by the GPPCA is close to the truth in both scenarios, even when the variance of the noise is large in the second scenario.

For PCA, the mean of the outputs is typically estimated by the maximum likelihood estimator $\hat{\mathbf{A}}_{pca} \hat{\mathbf{A}}_{pca}^T \mathbf{Y}$, where $\hat{\mathbf{A}}_{pca} = \mathbf{U}_0$ (Bai and Ng, 2002). In Figure 2, the PCA estimation of the mean for Example 1 is graphed as the red curves and the posterior mean of the output in the GPPCA in Corollary 5 is graphed as the blue curves. The PCA underesti-

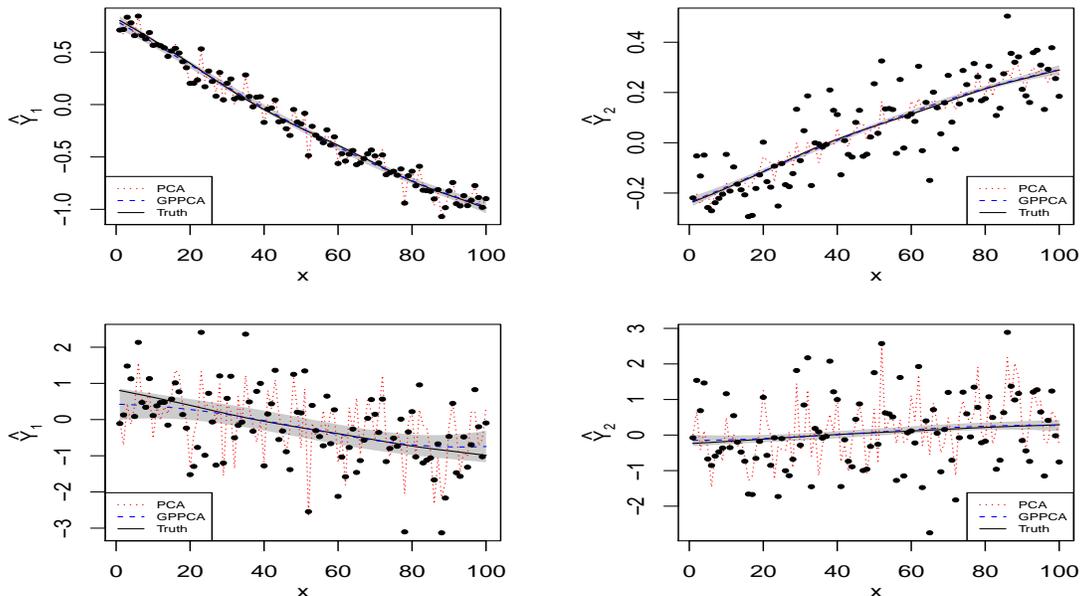


Figure 2: Estimation of the mean of the output \mathbf{Y} for Example 1 with the variance of the noise being $\sigma_0^2 = 0.01$ and $\sigma_0^2 = 1$, graphed in the upper panels and lower panels, respectively. The first row and second row of \mathbf{Y} are graphed as the black curves in the left panels and right panels, respectively. The red dotted curves and the blue dashed curves are the prediction by the PCA and GPPCA, respectively. The grey region is the 95% posterior credible interval from GPPCA. The black curves, blue curves and grey regions almost overlap in the upper panels.

mates the variance of the noise and hence has a large estimation error. In comparison, the estimated mean of the output by the GPPCA is more accurate, as the correlation in each output variable is properly modeled through the GPs of the latent factors.

Note that we restrict \mathbf{A} to satisfy $\mathbf{A}^T \mathbf{A} = \mathbf{I}_d$ when simulating data examples in Figure 1. In practice, we find this constraint only affects the estimation of the variance parameter σ_l^2 in the kernel, $l = 1, \dots, d$, because the meaning of this parameter changes.

There are some other estimators of the factor loading matrix in modeling high-dimensional time series. For example, Lam et al. (2011); Lam and Yao (2012) estimate the factor loading matrix of model (1) by $\hat{\mathbf{A}}_{LY} := \sum_{q=1}^{q_0} \hat{\Sigma}_y(q) \hat{\Sigma}_y^T(q)$, where $\hat{\Sigma}_y(q)$ is the $k \times k$ sample covariance at lag q of the output and q_0 is fixed to be a small positive integer. This approach is sensible, because $\mathcal{M}(\mathbf{A})$ is shown to be spanned from $\sum_{q=1}^{q_0} \Sigma_y(q) \Sigma_y^T(q)$ under some reasonable assumptions, where $\Sigma_y(q)$ is the underlying lag- q covariance of the outputs. It is also suggested in Lam and Yao (2012) to estimate the latent factor by $\hat{\mathbf{Z}}_{LY} = \hat{\mathbf{A}}_{LY}^T \mathbf{Y}$, meaning that the mean of the output is estimated by $\hat{\mathbf{A}}_{LY} \hat{\mathbf{Z}}_{LY} = \hat{\mathbf{A}}_{LY} \hat{\mathbf{A}}_{LY}^T \mathbf{Y}$. This estimator and the PCA are both included for comparison in Section 4.

4. Simulated examples

In this section, we numerically compare different approaches studied before. We use several criteria to examine the estimation. The first criterion is the largest principal angle between the estimated subspace $\mathcal{M}(\hat{\mathbf{A}})$ and the true subspace $\mathcal{M}(\mathbf{A})$. Let $0 \leq \phi_1 \leq \dots \leq \phi_d \leq \pi/2$ be the principal angles between $\mathcal{M}(\mathbf{A})$ and $\mathcal{M}(\hat{\mathbf{A}})$, recursively defined by

$$\phi_i = \arccos \left(\max_{\mathbf{a} \in \mathcal{M}(\mathbf{A}), \hat{\mathbf{a}} \in \mathcal{M}(\hat{\mathbf{A}})} |\mathbf{a}^T \hat{\mathbf{a}}| \right) = \arccos(|\mathbf{a}_i^T \hat{\mathbf{a}}_i|),$$

subject to

$$\|\mathbf{a}\| = \|\hat{\mathbf{a}}\| = 1, \mathbf{a}^T \mathbf{a}_i = 0, \hat{\mathbf{a}}^T \hat{\mathbf{a}}_i = 0, i = 1, \dots, d-1,$$

where $\|\cdot\|$ denotes the L_2 norm. The largest principal angle is ϕ_d , which quantifies how close two linear subspaces are. When two subspaces are identical, all principal angles are zero. When the columns of the \mathbf{A} and $\hat{\mathbf{A}}$ form orthogonal bases of the $\mathcal{M}(\mathbf{A})$ and $\mathcal{M}(\hat{\mathbf{A}})$, the cosine of the largest principal angle is equal to the smallest singular value of $\mathbf{A}^T \hat{\mathbf{A}}$ (Björck and Golub, 1973; Absil et al., 2006). Thus the largest principal angle can be calculated efficiently through the singular value decomposition of $\mathbf{A}^T \hat{\mathbf{A}}$.

We numerically compare four approaches for estimating \mathbf{A} . The first approach is the PCA, which estimates \mathbf{A} by \mathbf{U}_0 , where \mathbf{U}_0 is the first d eigenvectors of $\mathbf{Y}\mathbf{Y}^T/n$. Note the other version of the PCA and the PPCA have the same largest principal angle between the estimated subspace of \mathbf{A} and the true subspace of \mathbf{A} , so the results are omitted. The GPPCA is the second approach. When the covariance of the factor processes is the same, the closed-form expression of the estimator of the factor loading matrix is given in Theorem 2. When the covariance of the factor processes is different, we implement the optimization algorithm in Wen and Yin (2013) that preserves the orthogonal constraints to obtain the maximum marginal likelihood estimation of the factor loading matrix in Theorem 3. In both cases, the estimator $\hat{\mathbf{A}}$ can be written as a function of $(\gamma, \tau, \sigma_0^2)$ which are estimated by maximizing the marginal likelihood after integrating out \mathbf{Z} and plugging $\hat{\mathbf{A}}$. The third approach, denoted as LY1, estimates \mathbf{A} by $\hat{\Sigma}_y(1)\hat{\Sigma}_y^T(1)$, where $\hat{\Sigma}_y(1)$ is the sample covariance of the output at lag 1 and the fourth approach, denoted as LY5, estimates \mathbf{A} by $\sum_{q=1}^{q_0} \hat{\Sigma}_y(q)\hat{\Sigma}_y^T(q)$ with $q_0 = 5$, used in Lam and Yao (2012) and Lam et al. (2011), respectively.

We also compare the performance of different approaches by the average mean squared errors (AvgMSE) in predicting the mean of the output over N experiments as follows

$$\text{AvgMSE} = \sum_{l=1}^N \sum_{j=1}^k \sum_{i=1}^n \frac{(\hat{Y}_{j,i}^{(l)} - \mathbb{E}[Y_{j,i}^{(l)}])^2}{knN}, \quad (25)$$

where $\mathbb{E}[Y_{j,i}^{(l)}]$ is the (j, i) element of the mean of the output matrix at the l th experiment, and $\hat{Y}_{j,i}^{(l)}$ is the estimation. As discussed before, the estimated mean of the output matrix by the PCA, LY1 and LY5 is $\hat{\mathbf{A}}\hat{\mathbf{A}}^T\mathbf{Y}$, where $\hat{\mathbf{A}}$ is the estimated factor loading matrix in each approach (Bai and Ng (2002); Lam et al. (2011); Lam and Yao (2012)). In GPPCA, we use the posterior mean of \mathbf{AZ} in Corollary 5 to estimate mean of the output matrix.

The cases of the shared covariance and the different covariances of the factor processes are studied in Example 2 and Example 3, respectively. we assume that \mathbf{A} is sampled from

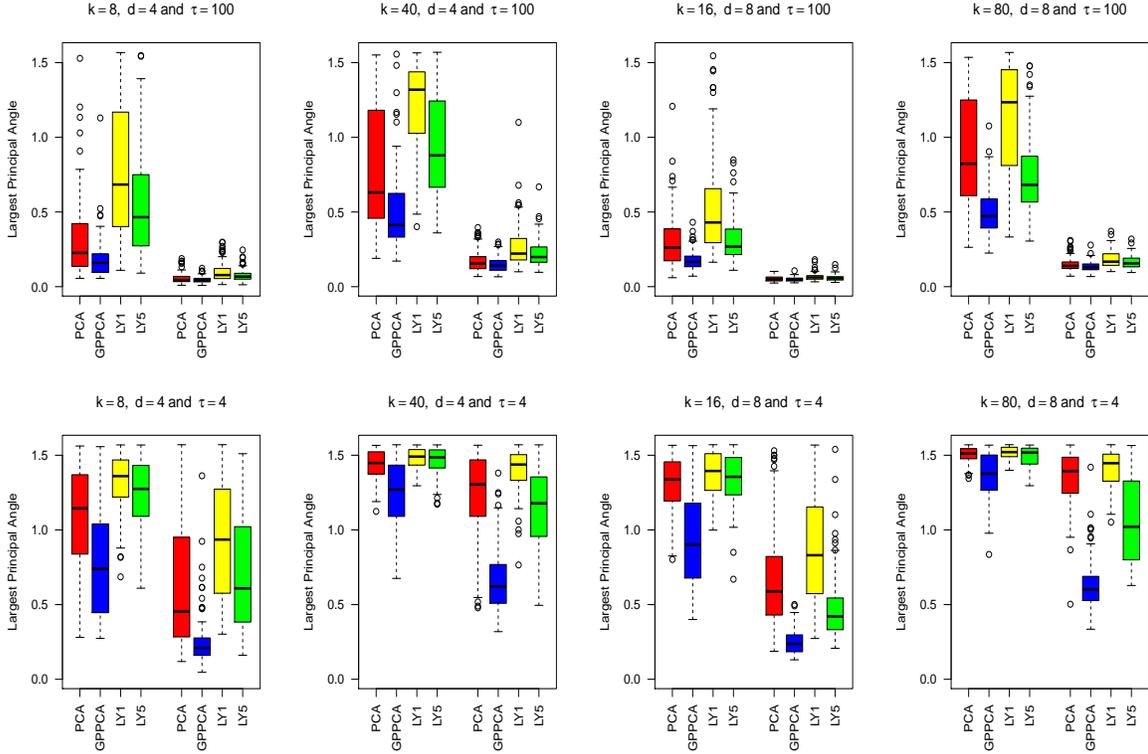


Figure 3: The largest principal angle between the true subspace of the factor loading matrix and the estimation from the four approaches for Example 2 (ranging from $[0, \pi/2]$, the smaller the better). In the first row, the number of the observations of each output variable is assumed to be $n = 200$ and $n = 400$ for the left four boxplots and right four boxplots in each panel, respectively. In the second row, the number of observations is assumed to be $n = 500$ and $n = 1000$ for the left four boxplots and right four boxplots in each panel, respectively.

the uniform distribution on the Stiefel manifold (Hoff, 2013), and the kernels are correctly specified with unknown parameters in these examples. In Appendix C, we compare different approaches when the factor loading matrix, kernel functions or the factors are misspecified.

Example 2 (Factors with the same covariance matrix) *The data are sampled from model (1) with $\Sigma_1 = \dots = \Sigma_d = \Sigma$, where $x_i = i$ for $1 \leq i \leq n$, and the kernel function in (10) is used with $\gamma = 100$ and $\sigma^2 = 1$. In each scenario, we simulate the data from 16 different combinations of σ_0^2 , k , d and n . We repeat $N = 100$ times for each scenario. The parameters $(\sigma_0^2, \sigma^2, \gamma)$ are treated as unknown and estimated from the data.*

In Figure 3, we present the largest principal angle between the true subspace $\mathcal{M}(\mathbf{A})$ and estimated subspace $\mathcal{M}(\hat{\mathbf{A}})$ at different settings of Example 1. The red, blue, yellow and green boxplots are the results from the PCA, GPPCA, LY1 and LY5. In each panel, the

$d = 4$ and $\tau = 100$	k=8		k=40	
	$n = 200$	$n = 400$	$n = 200$	$n = 400$
PCA	5.3×10^{-3}	5.1×10^{-3}	1.4×10^{-3}	1.1×10^{-3}
GPPCA	3.3×10^{-4}	2.6×10^{-4}	2.2×10^{-4}	1.3×10^{-4}
LY1	4.6×10^{-2}	5.8×10^{-3}	1.5×10^{-2}	2.1×10^{-3}
LY5	3.2×10^{-2}	5.5×10^{-3}	1.1×10^{-2}	1.8×10^{-3}
$d = 8$ and $\tau = 100$	k=16		k=80	
	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
PCA	5.2×10^{-3}	5.0×10^{-3}	1.3×10^{-3}	1.1×10^{-3}
GPPCA	2.9×10^{-4}	2.4×10^{-4}	1.9×10^{-4}	1.1×10^{-4}
LY1	1.4×10^{-2}	5.1×10^{-3}	5.4×10^{-3}	1.2×10^{-3}
LY5	8.8×10^{-3}	5.1×10^{-3}	3.9×10^{-3}	1.2×10^{-3}
$d = 4$ and $\tau = 4$	k=8		k=40	
	$n = 200$	$n = 400$	$n = 200$	$n = 400$
PCA	1.4×10^{-1}	1.3×10^{-1}	4.2×10^{-2}	3.4×10^{-2}
GPPCA	5.8×10^{-3}	4.4×10^{-3}	5.3×10^{-3}	3.0×10^{-3}
LY1	2.2×10^{-1}	1.7×10^{-1}	7.2×10^{-2}	6.4×10^{-2}
LY5	2.2×10^{-1}	1.5×10^{-1}	4.8×10^{-2}	4.1×10^{-2}
$d = 8$ and $\tau = 4$	k=16		k=80	
	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
PCA	1.4×10^{-1}	1.3×10^{-1}	3.9×10^{-2}	3.2×10^{-2}
GPPCA	5.1×10^{-3}	3.9×10^{-3}	4.3×10^{-3}	2.4×10^{-3}
LY1	1.8×10^{-1}	1.4×10^{-1}	5.1×10^{-2}	3.4×10^{-2}
LY5	1.7×10^{-1}	1.3×10^{-1}	4.6×10^{-2}	3.1×10^{-2}

Table 1: AvgMSE for Example 2.

sample size gets doubled from the left four boxplots to the right four. The SNR $\tau = \sigma^2/\sigma_0^2$ is assumed to be 100 and 4 in the upper panels and lower panels, respectively.

Since the covariance of the factor processes is the same in Example 2, the estimated \mathbf{A} by the GPPCA has a closed-form solution given in Theorem 2. For all 16 different scenarios, the GPPCA outperforms the other three methods in terms of having the smallest largest principal angle between $\mathcal{M}(\mathbf{A})$ and $\mathcal{M}(\hat{\mathbf{A}})$. Both PCA and GPPCA can be viewed as maximum likelihood type of approaches under the orthonormality assumption of the factor loading matrix. The difference is that the estimator of \mathbf{A} by the GPPCA maximizes the marginal likelihood after integrating out the factor processes, whereas the PCA maximizes the likelihood without modeling the factor processes. The principal axes by the PCA are the same as the PPCA which assumes the factors are independently distributed. As discussed before, the model with independent factors, however, is not a sensible sampling model for the correlated data, such as the multiple time series or multivariate spatial processes.

The performance of all methods improves when the sample size increases or when the SNR increases, as shown in Figure 3. The LY5 estimator (Lam et al., 2011) seems to perform slightly better than the PCA when the SNR is smaller. This method is sensible because the factor loading space $\mathcal{M}(\mathbf{A})$ is spanned by the eigenvectors of $\mathbf{M} := \sum_{i=1}^{q_0} \Sigma_y(q) \Sigma_y^T(q)$

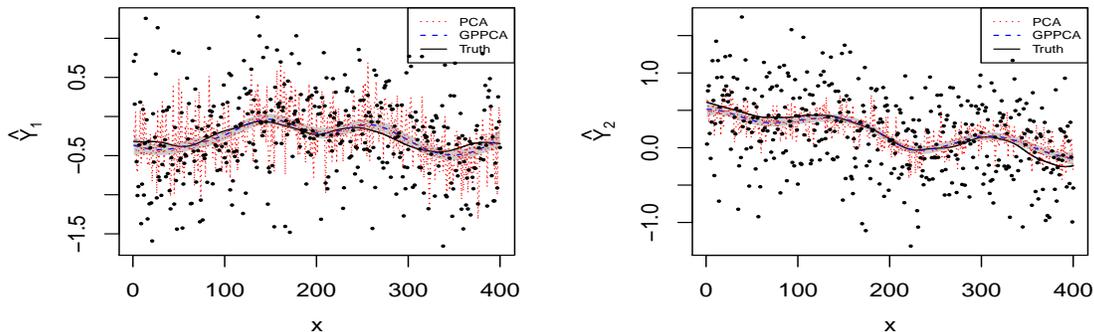


Figure 4: Prediction of the mean of the first two output variables in one experiment with $k = 8$, $d = 4$, $n = 400$ and $\tau = 4$. The observations are plotted as black circles and the truth is graphed as the black curves. The estimation by the PCA and GPPCA is graphed as the red dotted curves and blue dashed curves, respectively. The shaded area is the 95% posterior credible interval by the GPPCA.

under some conditions. However, this may not be the unique way to represent the subspace of the factor loading matrix. Thus the estimator based on this argument may not be as efficient as the maximum marginal likelihood approach by the GPPCA, shown in Figure 3.

The AvgMSE of the different approaches for Example 2 is summarized in Table 1. The mean squared error of the estimation by the GPPCA is typically a digit or two smaller than those obtained by other approaches. This is because the correlation of the factor processes in the GPPCA is properly modeled, and the kernel parameters are estimated based on the maximum marginal likelihood estimation.

We plot the first two rows of the estimated mean of the output in one experiment from the Example 2 in Figure 4. The estimation of the GPPCA approach is graphed as the blue dashed curves, which is very close to the truth, graphed as the black curves, whereas the estimation by the PCA is graphed as the red dotted curves, which are less smooth and less accurate in predicting the mean of the outputs, because of the noise in the data. The estimators by LY1 and LY5 are similar to those of PCA so we omit them in Figure 4. The problem of the PCA (and PPCA) is that the estimation assumes that the factors are independently distributed, which makes the likelihood too concentrated. Hence the variance of the noise is underestimated as indicated by the red curves in Figure 4. In comparison, the variance of the noise estimated by the GPPCA is more accurate, which makes predictions by the GPPCA closer to the truth.

Example 3 (Factors with different covariance matrices) *The data are sampled from model (1) where $x_i = i$ for $1 \leq i \leq n$. The variance of the noise is $\sigma_0^2 = 0.25$ and the kernel function is assumed to follow from (10) with $\sigma^2 = 1$. The range parameter γ of each factor is uniformly sampled from $[10, 10^3]$ in each experiment. We simulate the data from 8 different*

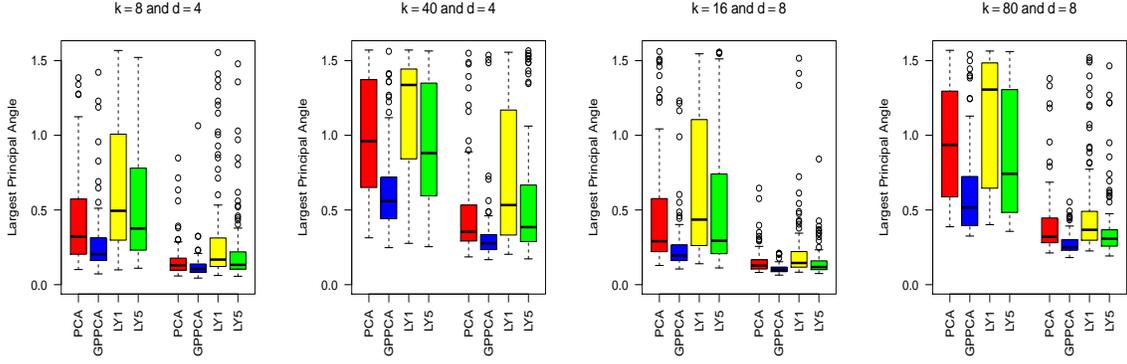


Figure 5: The largest principal angle between the true subspace and the estimated subspace of the four approaches for Example 3. The number of observations of each output variable is $n = 200$ and $n = 400$ for left 4 boxplots and right 4 boxplots in 2 left panels, respectively. The number of observations is $n = 500$ and $n = 1000$ for left 4 boxplots and right 4 boxplots in 2 right panels, respectively.

$d = 4$ and $\tau = 4$	k=8		k=40	
	$n = 200$	$n = 400$	$n = 200$	$n = 400$
PCA	1.3×10^{-1}	1.3×10^{-1}	3.8×10^{-2}	3.0×10^{-2}
GPPCA	1.4×10^{-2}	4.0×10^{-2}	7.1×10^{-3}	1.1×10^{-2}
LY1	1.6×10^{-1}	1.4×10^{-1}	4.9×10^{-2}	3.4×10^{-2}
LY5	1.5×10^{-1}	1.3×10^{-1}	4.4×10^{-2}	3.2×10^{-2}
$d = 8$ and $\tau = 4$	k=16		k=80	
	$n = 500$	$n = 1000$	$n = 500$	$n = 1000$
PCA	1.3×10^{-1}	1.3×10^{-1}	3.5×10^{-2}	2.9×10^{-2}
GPPCA	1.3×10^{-2}	3.3×10^{-2}	6.0×10^{-3}	8.0×10^{-3}
LY1	1.4×10^{-1}	1.3×10^{-1}	3.7×10^{-2}	2.9×10^{-2}
LY5	1.4×10^{-1}	1.3×10^{-1}	3.4×10^{-2}	2.8×10^{-2}

Table 2: AvgMSE for Example 3.

combinations of k , d and n . In each scenario, we repeat $N = 100$ times. The parameters in the kernels and the variance of the noise are all estimated from the data.

Since the covariance matrices are different in Example 3, we implement the numerical optimization algorithm on the Stiefel manifold (Wen and Yin, 2013) to estimate \mathbf{A} in Theorem 3. The largest principal angle between $\mathcal{M}(\mathbf{A})$ and $\mathcal{M}(\hat{\mathbf{A}})$ and the AvgMSE in estimating the mean of the output matrix by different approaches for Example 3 is given in Figure 5 and Table 2, respectively. The estimation by the GPPCA outperforms the other methods based on both criteria.

5. Real Data Examples

We apply the proposed GPPCA approach and compared its performance with other approaches on two real data applications in this section.

5.1. Emulating multivariate output of the computer models

We first apply GPPCA for emulating computer models with multivariate output. Computer models or simulators have been developed and used in various scientific, engineering and social applications. Some simulators are computationally expensive (as the numerical solution of a system of the partial differential equations (PDEs) is often required and is slow), and some contain multivariate outputs at a set of the input parameters (see e.g. Higdon et al. (2008); Paulo et al. (2012); Fricker et al. (2013); Gu and Berger (2016)). Thus, a statistical emulator is often required to approximate the behavior of the simulator.

We consider the testbed called the ‘diplomatic and military operations in a non-warfighting domain’ (DIAMOND) simulator (Taylor and Lane, 2004). The DIAMOND simulator models the number of casualties during the second day to sixth day after the earthquake and volcanic eruption in Giarre and Catania. The simulator has 13 input variables, such as the helicopter cruise speed, engineer ground speed, shelter and food supply capacity at the two places (see Table 1 in Overstall and Woods (2016) for a complete list of the input variables).

We use the same $n = 120$ training and $n^* = 120$ test outputs in Overstall and Woods (2016) to compare different methods. We focus on the out-of-sample prediction criteria:

$$\text{RMSE} = \sqrt{\frac{\sum_{j=1}^k \sum_{i=1}^{n^*} (\hat{Y}_j^*(\mathbf{x}_i^*) - Y_j^*(\mathbf{x}_i^*))^2}{kn^*}}, \quad (26)$$

$$P_{CI}(95\%) = \frac{1}{kn^*} \sum_{j=1}^k \sum_{i=1}^{n^*} \mathbf{1}\{Y_j^*(\mathbf{x}_i^*) \in CI_{ij}(95\%)\}, \quad (27)$$

$$L_{CI}(95\%) = \frac{1}{kn^*} \sum_{j=1}^k \sum_{i=1}^{n^*} \text{length}\{CI_{ij}(95\%)\}, \quad (28)$$

where $Y_j^*(\mathbf{x}_i^*)$ is the j th coordinate of the held-out test output vector at the i th test input \mathbf{x}_i^* for $1 \leq i \leq n^*$ and $1 \leq j \leq k^*$, $CI_{ij}(95\%)$ is the 95% predictive credible interval and $\text{length}\{CI_{ij}(95\%)\}$ is the length of the 95% predictive credible interval of the $Y_j^*(\mathbf{x}_i^*)$. A method with a small out-of-sample RMSE, $P_{CI}(95\%)$ being close to nominal 95% level, and a small $L_{CI}(95\%)$ is considered precise in prediction and uncertainty quantification.

We compare the prediction performance of the GPPCA, the independent Gaussian processes (Ind GP) and multivariate Gaussian process (Multi GP) on the held-out test output. The Ind GP builds a GP to emulate each coordinate of the output vector separately. The Multi GP in Overstall and Woods (2016) proposes a separable model, where the covariance of the output is a Kronecker product of the covariance matrix of the output vector at the same input, and the correlation matrix of the any output variable at different inputs. The parameters of Multi GP are estimated by the MLE using the code provided in Overstall and Woods (2016) and the parameters in Ind GP are estimated by the posterior mode using RobustGaSP R package (Gu et al., 2019).

Method	Mean function	Kernel	RMSE	$PCI(95\%)$	$LCI(95\%)$
GPPCA	Intercept	Gaussian kernel	3.33×10^2	0.948	1.52×10^3
GPPCA	Selected covariates	Gaussian kernel	3.18×10^2	0.957	1.31×10^3
GPPCA	Intercept	Matérn kernel	2.82×10^2	0.962	1.22×10^3
GPPCA	Selected covariates	Matérn kernel	2.74×10^2	0.957	1.18×10^3
Ind GP	Intercept	Gaussian kernel	3.64×10^2	0.918	1.18×10^3
Ind GP	Selected covariates	Gaussian kernel	4.04×10^2	0.918	1.17×10^3
Ind GP	Intercept	Matérn kernel	3.40×10^2	0.930	0.984×10^3
Ind GP	Selected covariates	Matérn kernel	3.31×10^2	0.927	0.967×10^3
Multi GP	Intercept	Gaussian kernel	3.63×10^2	0.975	1.67×10^3
Multi GP	Selected covariates	Gaussian kernel	3.34×10^2	0.963	1.54×10^3
Multi GP	Intercept	Matérn kernel	3.01×10^2	0.962	1.34×10^3
Multi GP	Selected covariates	Matérn kernel	3.05×10^2	0.970	1.50×10^3

Table 3: Emulation of the DIAMOND simulator by different models. The first four rows show the predictive performance by the GPPCA with different mean structure and kernels. The middle four rows give the predictive performance by Ind GP with the same mean structure and kernels, as used in the GPPCA. The 9th and 10th rows show the emulation result of two best models in Overstall and Woods (2016) using Gaussian kernel for the same held-out test output, whereas the last two rows give the result of the same model with the Matérn kernel in (10). The RMSE is 1.08×10^5 using the mean of the training output to predict.

We use a product kernel for all models where each kernel is assumed the same for each input dimension. The Gaussian kernel is assumed in Overstall and Woods (2016) and we also include results using the Matérn kernel in (10) for comparison. In Overstall and Woods (2016), the model with the least RMSE is the one using the Gaussian kernel and a set of selected covariates. We find the 11th input (food capacity in Catania) is positively correlated with the outputs. Thus for the GPPCA and Ind GP, we explore the predictive performance of the models with the mean basis function being $\mathbf{h}(\mathbf{x}) = (1, x_{11})$. For GPPCA, we assume the range parameters in the kernels are shared for the latent factor processes, while the variance parameters are allowed to be different.

The predictive RMSEs of different models are shown in Table 3. Overall, all three approaches are precise in prediction, as the predictive RMSE is less than 1% of the RMSE using the mean to predict. Compared to the other two approaches, the GPPCA has the smallest out-of-sample RMSE on each combination of the kernel function and mean function among three approaches. The nominal 95% predictive interval covers around 95% of held-out test output with relatively short average length of the predictive interval. The predictive interval from Multi GP covers more than 95% of the held-out test output, but the average length of the interval is the highest. The Ind GP has the shortest length of the predictive interval, but it covers less than 95% of the held-out test output using any kernel or mean function. The held-out test output on the fifth and sixth day and the prediction by Ind GP and GPPCA are graphed in the right panel in Figure 6, both of which seem to be accurate.

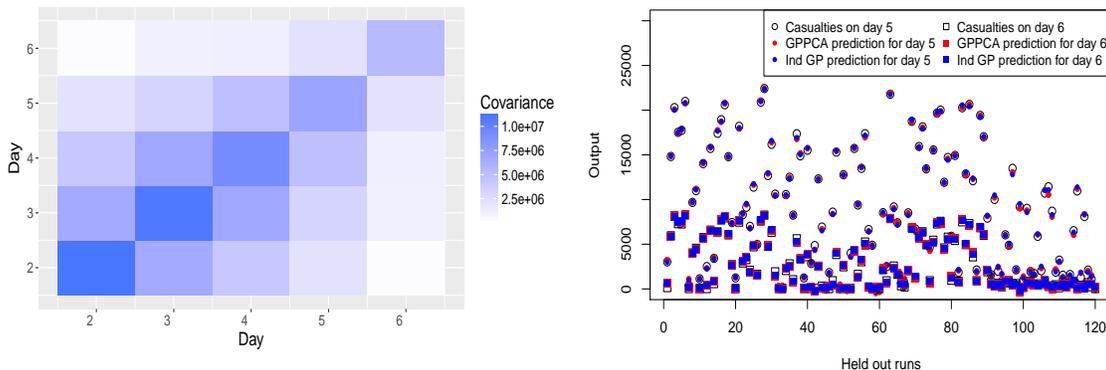


Figure 6: The estimated covariance of the casualties at the different days after the catastrophe by the GPPCA is graphed in the left panel. The held-out test output, the prediction by the GPPCA and Independent GPs with the mean basis $\mathbf{h}(\mathbf{x}) = (1, x_{11})$ and Matérn kernel for the fifth day and sixth day are graphed in the right panel.

In GPPCA, the estimated covariance matrix of the casualties at the different days is $\hat{\mathbf{A}}\hat{\mathbf{\Lambda}}\hat{\mathbf{A}} + \hat{\sigma}_0^2\mathbf{I}_k$, where $\hat{\mathbf{A}}$ is a diagonal matrix where the i th term is $\hat{\sigma}_i^2$ (the estimated variance of the i th factor). This covariance matrix is shown in the left panel in Figure 6. We find that the estimated covariance between any two days is positive. This is sensible as the short food capacity, for example, is associated with the high casualties for all following days after the catastrophe. We also notice that the estimated correlation of the output at the two consecutive days is larger, though we do not enforce a time-dependent structure (such as the autoregressive model in Liu and West (2009); Farah et al. (2014)). The GPPCA is a more general model as the output does not have to be time-dependent, and the estimated covariance between the output variables captures the time dependence in the example.

5.2. Gridded temperature

In this subsection, we consider global gridded temperature anomalies from U.S. National Oceanic and Atmospheric Administration (NOAA), available at:

<ftp://ftp.ncdc.noaa.gov/pub/data/noaaglobaltemp/operational>

This dataset records the global gridded monthly anomalies of the air and marine temperature from Jan 1880 to near present with $5^\circ \times 5^\circ$ latitude-longitude resolution (Shen, 2017).

A proportion of the temperature measurements is missing in the data set, which is also a common scenario in other climate data sets. As many scientific studies may rely on the full data set, we first compare different approaches on interpolation, using the monthly temperature anomalies at $k = 1, 639$ spatial coordinates in the past 20 years. We hold out the 24,000 randomly sampled measurements on $k^* = 1, 200$ spatial coordinates in $n^* = 20$ months as the test data set. The rest 15, 336 measurements are used as the training data. We

Method	measurement error	RMSE	$P_{CI}(95\%)$	$L_{CI}(95\%)$
GPPCA, $d = 50$	estimated	0.386	0.870	1.02
GPPCA, $d = 100$	estimated	0.320	0.772	0.563
GPPCA, $d = 50$	fixed	0.385	0.933	1.33
GPPCA, $d = 100$	fixed	0.314	0.977	1.44
PPCA, $d = 50$	estimated	0.620	0.677	1.08
PPCA, $d = 100$	estimated	0.602	0.525	0.803
PPCA, $d = 50$	fixed	0.617	0.765	1.32
PPCA, $d = 100$	fixed	0.585	0.819	1.400
Temporal model	estimated	0.937	0.944	2.28
Spatial model	estimated	0.560	0.942	2.23
Spatio-temporal model	estimated	0.492	0.957	2.10
Temporal regression by RF	estimated	0.441	/	/
Spatial regression by RF	estimated	0.391	/	/

Table 4: Out-of-sample prediction of the temperature anomalies by different approaches. The first four rows give the predictive performance by the GPPCA with different latent factors, estimated and fixed variance of the measurement error, whereas the latter four rows record the results by the PPCA. The predictive performance by the temporal, spatial and spatio-temporal smoothing methods are given in the 9th and 10th rows. The last two rows give the predictive RMSE by regression using the random forest (RF) algorithm.

evaluate the interpolation performance of different methods based on the RMSE, $P_{CI}(95\%)$, and $L_{CI}(95\%)$ on the test data set.

The predictive performance by the GPPCA using the predictive distribution in (24) is shown in the first four rows of Table 4. Here the number of grid boxes is $k = 1639$, and the temporal correlation of the temperature measurements at different months are parameterized by the Matérn kernel in (10). We model the intercept and monthly change rate at each location by assuming the mean basis function $\mathbf{h}(x) = (1, x)$, where x is an integer from 1 to 240 to denote the month of an observation. We explore the cases with $d = 50$ and $d = 100$ latent factor processes where the covariance in each latent process is assumed to be the same. In this dataset, the average recorded variance of the measurement error is around 0.1. We implement the scenarios with an estimated variance or a fixed variance of the measurements. In the fifth to the eighth rows, we show the predictive performance of the PPCA with the same number of latent factors. In the ninth and tenth rows, we show the results by a spatial model and a temporal model both based on the Matérn kernel, separately for the observations in each spatial coordinate and in each month, respectively. The `RobustGaSP` R package (Gu et al., 2019) is used to fit the GP regression with the estimated nuggets, and the mean basis function is assumed to be $\mathbf{h}(x) = (1, x)$ when fitting GP regression for the monthly measurements. The predictive performance by a spatio-temporal model that uses a product Matérn kernel function is shown in the eleventh row. In the last two rows in Table 4, we consider two regression schemes based

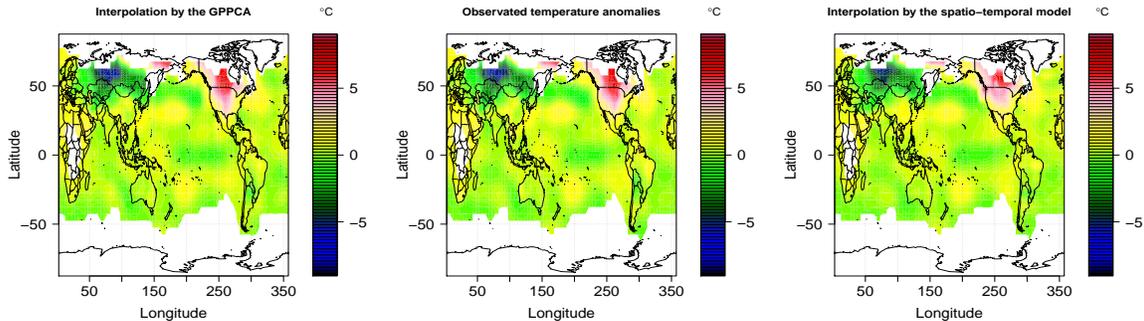


Figure 7: Interpolation of the temperature anomalies in November 2016. The real temperature anomalies in November 2016 is graphed in the middle panel. The interpolated temperature anomalies by the GPPCA and spatio-temporal model are graphed in the left and right panels, respectively. The number of training and test observations are 439 and 1200, respectively. The out-of-sample RMSE of the GPPCA and spatio-temporal model is 0.314 and 0.747, respectively.

on the random forest algorithm (Breiman (2001)). The first scheme treats the observations in each spatial coordinate as independent measurements, whereas the second scheme treats the observations in each month as independent measurements. The modeling fitting details of these approaches are given in Appendix D.

First, we find that GPPCA has the lowest out-of-sample RMSE among all the methods we considered. When the number of factors increases, both the PPCA and GPPCA seem to perform better in terms of RMSEs. However, the estimation by the GPPCA is more precise. This is because the temporal correlation and linear trend are modeled and estimated in the GPPCA, whereas the PPCA is a special case of GPPCA with the independent monthly measurements. This result is achieved by the simplest setting in GPPCA, that is when the covariance of the factor processes is assumed to be the same. In this case, the estimation of the factor loadings has a closed-form expression. Assuming different parameters in the factor processes and use other kernel functions may further improve the precision in prediction. Furthermore, when the variance of the measurement error is estimated, the predictive credible interval by either the PPCA or GPPCA is too short, resulting in less than 95% of the data covered by 95% predictive interval. When the variance of the noise is fixed at 0.1 (the variance of the measurement error), around 95% of the held-out data are covered in the nominal 95% predictive interval in the GPPCA, but not in the PPCA.

The spatial smoothing approach by GP and spatial regression by RF have smaller predictive errors than its temporal counterparts, indicating the spatial correlation may be larger than the temporal correlation in the data. Combining both the spatial and temporal information seems to be more accurate than using only the spatial or temporal information. However, the spatio-temporal model is not as accurate as the GPPCA. We plot the interpolated temporal anomalies in November 2016 by the GPPCA (with the variance of the measurement error fixed to be 0.1) and the spatio-temporal model in the left and right

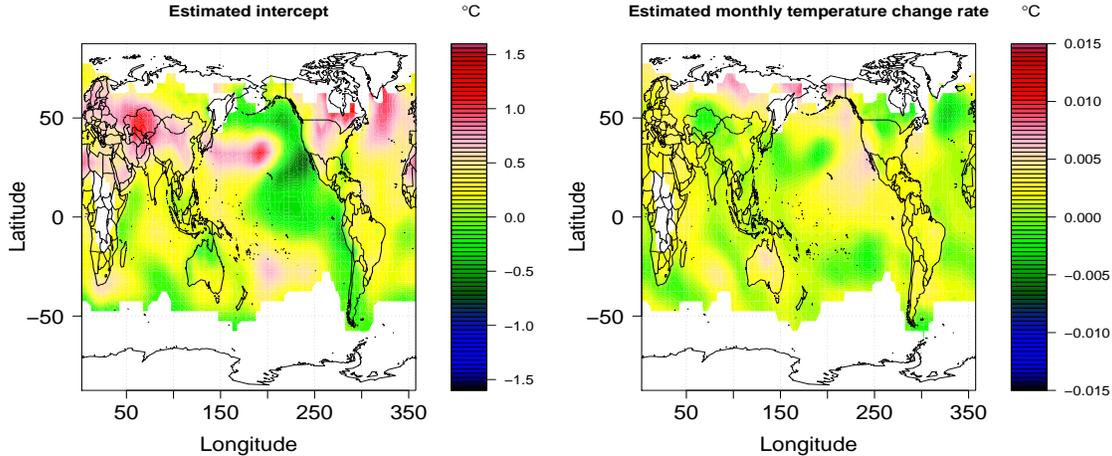


Figure 8: Estimated intercept and monthly change rate of the temperature anomalies by the GPPCA using the monthly temperature anomalies between January 1999 and December 2018.

panels in Figure 7, respectively. Compared with the observed temperature anomalies shown in the middle panel, the GPPCA interpolation is more precise than the spatial smoothing method at the locations where the temperature anomalies changes rapidly, e.g. the region between the U.S. and Canada, and the east region in Russia. We should acknowledge that the implemented spatio-temporal model is not the only choice. Other spatio-temporal models may be applicable, yet fitting these models may be more computationally expensive.

Note that the missing values are typically scattered in different rows and columns of the observation matrix in practice. One of the future directions is to extend the GPPCA to include the columns of the data matrix with missing values to improve the estimation of the factor loading matrix and the predictive distribution of the missing values, based on expectation-maximization algorithm, or the Markov chain Monte Carlo algorithm if one can specify the full posterior distributions of the factor loading matrix and the parameters. Besides, We should also emphasize that we do not utilize the spatial distance in the GPPCA. This makes the GPPCA suitable for other interpolation and matrix completion tasks when there is no distance information between the output variables.

The estimated trend parameters $\hat{\Theta}$ by the GPPCA are shown in Figure 8. Based on the last twenty years' data, the average annual increase of the temperature is at the rate of around $0.02 \text{ }^{\circ}\text{C}$. The areas close to the north pole seems to have the most rapid increase rate. Among the rest of the areas, the south west part and the north east part of the U.S. also seem to increase slightly faster than the other areas. Note we only use the observations from the past 20 years for demonstration purpose. A study based on a longer history of measurements may give a clearer picture of the change in global temperature.

6. Concluding remarks

In this paper, we have introduced the GPPCA, as an extension of the PPCA for the latent factor model with the correlated factors. By allowing data to infer the covariance structure of the factors, the estimation of the factor loading matrix and the predictive distribution of the output variables both become more accurate by the GPPCA, compared to the ones by the PCA and other approaches. This work also highlights the scalable computation achieved by a closed-form expression of the inverse covariance matrix in the marginal likelihood. In addition, we extend our approach to include additional covariates in the mean function and we manage to marginalize out the regression parameters to obtain a closed-form expression of the marginal likelihood when estimating the factor loading matrix.

There are several future directions related to this work. First of all, the factor loading matrix, as well as other parameters in the kernel functions and the variance of the noise, are estimated by the maximum marginal likelihood estimator, where the uncertainty in the parameter estimation is not expressed in the predictive distribution of the output variables. A full Bayesian approach may provide a better way to quantify the uncertainty in the predictive distribution. Secondly, we assume the number of the latent factors is known in this work. A consistent way to identify the number of latent factors is often needed in practice. Thirdly, the convergence rate of the predictive distribution and the estimation of the subspace of the factor loading matrix of the GPPCA is needed to be explored. The numerical results shown in this work seem to be encouraging towards this direction. Furthermore, when the covariances of the factor processes are not the same, the numerical optimization algorithm that preserves the orthogonal constraints (Wen and Yin, 2013) is implemented for the marginal maximum likelihood estimator of the factor loading matrix. The convergence of this algorithm is an interesting direction to explore. A fast algorithm for the optimization problem in Theorem 3 will also be crucial for some computationally intensive applications. Finally, here we use kernels to parameterize the covariance of the factor processes for demonstrative purposes. The GPPCA automatically apply to many other models of the latent factors, as long as the likelihood of a factor follows a multivariate normal distribution. It is interesting to explore the GPPCA in other factor models and applications.

Acknowledgments

The authors thank the editor and three anonymous referees for their comments that substantially improve this article. Shen's research is partially supported by the Simons Foundation Award 512620 and the National Science Foundation (NSF DMS 1509023). This material was based upon work partially supported by the National Science Foundation under Grant DMS-1638521 to the Statistical and Applied Mathematical Sciences Institute.

Appendix A: Auxiliary facts

1. Let \mathbf{A} and \mathbf{B} be matrices,

$$(\mathbf{A} \otimes \mathbf{B})^T = (\mathbf{A}^T \otimes \mathbf{B}^T);$$

further assuming \mathbf{A} and \mathbf{B} are invertible,

$$(\mathbf{A} \otimes \mathbf{B})^{-1} = \mathbf{A}^{-1} \otimes \mathbf{B}^{-1}.$$

2. Let \mathbf{A} , \mathbf{B} , \mathbf{C} and \mathbf{D} be the matrices such that the products \mathbf{AC} and \mathbf{BD} are matrices,

$$(\mathbf{A} \otimes \mathbf{B})(\mathbf{C} \otimes \mathbf{D}) = (\mathbf{AC}) \otimes (\mathbf{BD}).$$

3. For matrices \mathbf{A} , \mathbf{B} and \mathbf{C} ,

$$(\mathbf{C}^T \otimes \mathbf{A})\text{vec}(\mathbf{B}) = \text{vec}(\mathbf{ABC});$$

further assuming $\mathbf{A}^T\mathbf{B}$ is a matrix,

$$\text{tr}(\mathbf{A}^T\mathbf{B}) = \text{vec}(\mathbf{A})^T \text{vec}(\mathbf{B}).$$

4. For any invertible $n \times n$ matrix \mathbf{C} ,

$$|\mathbf{C} + \mathbf{AB}| = |\mathbf{C}| |\mathbf{I}_n + \mathbf{BC}^{-1}\mathbf{A}|.$$

Appendix B: Proofs

We first give some notations for the vectorization used in the proofs. Let $\mathbf{A}_v = [\mathbf{I}_n \otimes \mathbf{a}_1, \dots, \mathbf{I}_n \otimes \mathbf{a}_d]$ and $\mathbf{Z}_{vt} = \text{vec}(\mathbf{Z}^T)$. Let $\boldsymbol{\Sigma}_v$ be a $nd \times nd$ block diagonal matrix where the l th diagonal block is $\boldsymbol{\Sigma}_l$, for $l = 1, \dots, d$.

Proof [Proof of Equation (4)] Vectorize the observations in model (1), one has

$$\mathbf{Y}_v = \mathbf{A}_v \mathbf{Z}_{vt} + \boldsymbol{\epsilon}_v$$

where $\mathbf{Z}_{vt} \sim N(\mathbf{0}, \boldsymbol{\Sigma}_v)$ and $\boldsymbol{\epsilon}_v \sim N(\mathbf{0}, \sigma^2 \mathbf{I}_{nk})$. Using the fact 1 and fact 2, $\mathbf{A}_v \mathbf{Z}_{vt} \sim \text{MN}(\mathbf{0}, \boldsymbol{\Sigma}_{A_v Z_{vt}})$, where

$$\boldsymbol{\Sigma}_{A_v Z_{vt}} = \mathbf{A}_v \boldsymbol{\Sigma}_v \mathbf{A}_v^T = [\boldsymbol{\Sigma}_1 \otimes \mathbf{a}_1, \dots, \boldsymbol{\Sigma}_d \otimes \mathbf{a}_d] \mathbf{A}_v^T = \sum_{l=1}^d \boldsymbol{\Sigma}_l \otimes (\mathbf{a}_l \mathbf{a}_l^T) \quad (29)$$

for $l = 1, \dots, d$. Marginalizing out \mathbf{Z}_{vt} , one has Equation (4). ■

Proof [Proof of Lemma 1] By (4) and (29), one has

$$\mathbf{Y}_v \mid \mathbf{A}, \sigma_0^2, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_d \sim \text{MN}(\mathbf{0}, \mathbf{A}_v \boldsymbol{\Sigma}_v \mathbf{A}_v^T + \sigma_0^2 \mathbf{I}_{nk}).$$

The precision matrix is

$$\begin{aligned} & (\mathbf{A}_v \boldsymbol{\Sigma}_v \mathbf{A}_v^T + \sigma_0^2 \mathbf{I}_{nk})^{-1} \\ &= \sigma_0^{-2} \mathbf{I}_{nk} - \mathbf{A}_v \frac{(\sigma_0^2 \boldsymbol{\Sigma}_v^{-1} + \mathbf{A}_v^T \mathbf{A}_v)^{-1}}{\sigma_0^2} \mathbf{A}_v^T \\ &= \sigma_0^{-2} \mathbf{I}_{nk} - \mathbf{A}_v \frac{(\sigma_0^2 \boldsymbol{\Sigma}_v^{-1} + \mathbf{I}_{nd})^{-1}}{\sigma_0^2} \mathbf{A}_v^T \\ &= \sigma_0^{-2} \left\{ \mathbf{I}_{nk} - [(\sigma_0^2 \boldsymbol{\Sigma}_1^{-1} + \mathbf{I}_n)^{-1} \otimes \mathbf{a}_1, \dots, (\sigma_0^2 \boldsymbol{\Sigma}_d^{-1} + \mathbf{I}_n)^{-1} \otimes \mathbf{a}_d] \mathbf{A}_v^T \right\} \\ &= \sigma_0^{-2} \left(\mathbf{I}_{nk} - \sum_{l=1}^d (\sigma_0^2 \boldsymbol{\Sigma}_l^{-1} + \mathbf{I}_n)^{-1} \otimes \mathbf{a}_l \mathbf{a}_l^T \right) \end{aligned}$$

where the first equality follows from the Woodbury identity; the second equality is by Assumption 1; the third equality is by fact 2; and the fourth equality is by fact 1 and fact 2, from which the results follow immediately. \blacksquare

Proof [Proof of Theorem 2]

When $\boldsymbol{\Sigma}_1 = \dots = \boldsymbol{\Sigma}_d = \boldsymbol{\Sigma}$, by the fact 3, the likelihood of \mathbf{A} is

$$\begin{aligned} L(\mathbf{A} \mid \mathbf{Y}, \sigma_0^2, \boldsymbol{\Sigma}) &\propto \exp \left(- \frac{\mathbf{Y}_v^T \left(\mathbf{I}_{nk} - (\sigma_0^2 \boldsymbol{\Sigma}^{-1} + \mathbf{I}_n)^{-1} \otimes \sum_{l=1}^d \mathbf{a}_l \mathbf{a}_l^T \right) \mathbf{Y}_v}{2\sigma_0^2} \right) \\ &\propto \exp \left(- \frac{\mathbf{Y}_v^T \mathbf{Y}_v - \mathbf{Y}_v^T \text{vec}(\mathbf{A} \mathbf{A}^T \mathbf{Y} (\sigma_0^2 \boldsymbol{\Sigma}^{-1} + \mathbf{I}_n)^{-1})}{2\sigma_0^2} \right) \\ &\propto \text{etr} \left(- \frac{\mathbf{Y}^T \mathbf{Y} - \mathbf{Y}^T \mathbf{A} \mathbf{A}^T \mathbf{Y} (\sigma_0^2 \boldsymbol{\Sigma}^{-1} + \mathbf{I}_n)^{-1}}{2\sigma_0^2} \right) \\ &\propto \text{etr} \left(- \frac{\mathbf{Y}^T \mathbf{Y} - \mathbf{A}^T \mathbf{Y} (\sigma_0^2 \boldsymbol{\Sigma}^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T \mathbf{A}}{2\sigma_0^2} \right), \end{aligned}$$

where $\text{etr}(\cdot) := \exp(\text{tr}(\cdot))$.

Maximizing the likelihood as a function of \mathbf{A} is equivalent to the optimization problem:

$$\hat{\mathbf{A}} = \underset{\mathbf{A}}{\text{argmax}} \text{tr}(\mathbf{A}^T \mathbf{G} \mathbf{A}) \quad \text{s.t.} \quad \mathbf{A}^T \mathbf{A} = \mathbf{I}_d, \quad (30)$$

where $\mathbf{G} = \mathbf{Y} (\mathbf{I}_n + \sigma_0^2 \boldsymbol{\Sigma}^{-1})^{-1} \mathbf{Y}^T$. This optimization in (30) is a trace optimization problem (Kokopoulou et al. (2011)). By the Courant-Fischer-Weyl min-max principal (Saad (1992)), $\text{tr}(\mathbf{A}^T \mathbf{G} \mathbf{A})$ is maximized when $\hat{\mathbf{A}} = \mathbf{U} \mathbf{R}$, with \mathbf{U} being the orthonormal basis of the eigenspace associated with the d largest eigenvalue of \mathbf{G} and \mathbf{R} is any arbitrary rotation matrix. In this case, $\text{tr}(\hat{\mathbf{A}}^T \mathbf{G} \hat{\mathbf{A}}) = \text{tr}(\mathbf{U} \boldsymbol{\Lambda} \mathbf{U}^T) = \sum_{l=1}^d \lambda_l$, where $\boldsymbol{\Lambda}$ is a diagonal matrix of the d largest eigenvalue λ_l of \mathbf{G} , for $l = 1, \dots, d$. \blacksquare

Proof [Proof of Theorem 3] Under Assumption 1, by fact 3, the likelihood for \mathbf{A} is

$$\begin{aligned} L(\mathbf{A} \mid \mathbf{Y}, \sigma_0^2, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_d) &\propto \exp \left(-\frac{\mathbf{Y}_v^T \left(\mathbf{I}_{nk} - \sum_{l=1}^d (\sigma_0^2 \boldsymbol{\Sigma}_l^{-1} + \mathbf{I}_n)^{-1} \otimes \mathbf{a}_l \mathbf{a}_l^T \right) \mathbf{Y}_v}{2\sigma_0^2} \right) \\ &\propto \text{etr} \left(-\frac{\mathbf{Y}^T \mathbf{Y} - \mathbf{Y}^T \sum_{l=1}^d \mathbf{a}_l \mathbf{a}_l^T \mathbf{Y} (\sigma_0^2 \boldsymbol{\Sigma}_l^{-1} + \mathbf{I}_n)^{-1}}{2\sigma_0^2} \right) \\ &\propto \text{etr} \left(-\frac{\mathbf{Y}^T \mathbf{Y} - \sum_{l=1}^d \mathbf{a}_l^T \mathbf{Y} (\sigma_0^2 \boldsymbol{\Sigma}_l^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T \mathbf{a}_l}{2\sigma_0^2} \right), \end{aligned}$$

from which the result follows. ■

Proof [Proof of Equation (12)]

From the proof of Theorem 3, one has

$$L(\sigma_0^2 \mid \mathbf{Y}, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_d, \mathbf{A}) \propto (\sigma_0^2)^{-nk/2} \text{etr} \left(-\frac{\mathbf{Y}^T \mathbf{Y} - \sum_{l=1}^d \mathbf{a}_l^T \mathbf{Y} (\sigma_0^2 \boldsymbol{\Sigma}_l^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T \mathbf{a}_l}{2\sigma_0^2} \right). \quad (31)$$

Equation (11) follows immediately by maximizing (31).

We now turn to show the profile likelihood in (12). Under Assumption 1

$$\begin{aligned} &p(\mathbf{Y} \mid \boldsymbol{\tau}, \boldsymbol{\gamma}, \mathbf{A}, \sigma_0^2) \\ &= \int p(\mathbf{Y} \mid \mathbf{A}, \sigma_0^2, \mathbf{Z}) p(\mathbf{Z} \mid \boldsymbol{\tau}, \boldsymbol{\gamma}) d\mathbf{Z} \\ &= \int (2\pi\sigma_0^2)^{-\frac{nk}{2}} \text{etr} \left(-\frac{(\mathbf{Y} - \mathbf{AZ})^T (\mathbf{Y} - \mathbf{AZ})}{\sigma_0^2} \right) (2\pi)^{-\frac{nd}{2}} \prod_{l=1}^d |\boldsymbol{\Sigma}_l|^{-\frac{1}{2}} \exp \left(-\frac{1}{2} \sum_{l=1}^d \mathbf{z}_l \boldsymbol{\Sigma}_l^{-1} \mathbf{z}_l^T \right) d\mathbf{Z} \\ &= (2\pi\sigma_0^2)^{-\frac{nk}{2}} \prod_{l=1}^d |\boldsymbol{\Sigma}_l / \sigma_0^2 + \mathbf{I}_k|^{-1/2} \exp \left(-\frac{S^2}{2\sigma_0^2} \right) \end{aligned} \quad (32)$$

where $S^2 = \text{tr}(\mathbf{Y}^T \mathbf{Y}) - \sum_{l=1}^d \mathbf{a}_l^T \mathbf{Y} (\tau_l^{-1} \mathbf{R}_l^{-1} + \mathbf{I}_n)^{-1} \mathbf{Y}^T \mathbf{a}_l$. Equation (12) follows by plugging $\hat{\mathbf{A}}$ and $\hat{\sigma}_0^2$ into (32). ■

Proof [Proof of Theorem 4] Denote the parameters $\hat{\boldsymbol{\theta}} = (\hat{\boldsymbol{\gamma}}, \hat{\mathbf{A}}, \hat{\boldsymbol{\sigma}}^2, \hat{\sigma}_0^2)$. Denote $\hat{\boldsymbol{\Sigma}}_v$ as the estimated $\boldsymbol{\Sigma}_v$ by plugging the estimated parameters. We first compute the posterior distribution of $(\mathbf{Z}_{vt} \mid \mathbf{Y}_v, \hat{\boldsymbol{\theta}})$. From Equation (4),

$$\begin{aligned} p(\mathbf{Z}_{vt} \mid \mathbf{Y}_v, \hat{\boldsymbol{\theta}}) &\propto \exp \left(\frac{(\mathbf{Y}_v - \hat{\mathbf{A}}_v \mathbf{Z}_{vt})^T (\mathbf{Y}_v - \hat{\mathbf{A}}_v \mathbf{Z}_{vt})}{2\hat{\sigma}_0^2} \right) \exp \left(-\frac{1}{2} \mathbf{Z}_{vt}^T \hat{\boldsymbol{\Sigma}}_v^{-1} \mathbf{Z}_{vt} \right) \\ &\propto \exp \left\{ -\frac{1}{2} (\mathbf{Z}_{vt} - \hat{\mathbf{Z}}_{vt})^T \left(\frac{\hat{\mathbf{A}}_v^T \hat{\mathbf{A}}_v}{\hat{\sigma}_0^2} + \hat{\boldsymbol{\Sigma}}_v^{-1} \right) (\mathbf{Z}_{vt} - \hat{\mathbf{Z}}_{vt}) \right\}, \end{aligned}$$

where $\hat{\mathbf{Z}}_{vt} = (\hat{\mathbf{A}}_v^T \hat{\mathbf{A}}_v + \hat{\sigma}_0^2 \hat{\Sigma}_v^{-1})^{-1} \hat{\mathbf{A}}_v^T \mathbf{Y}_v$ from which we have

$$\mathbf{Z}_{vt} \mid \mathbf{Y}_v, \hat{\boldsymbol{\theta}} \sim \text{MN} \left(\hat{\mathbf{Z}}_{vt}, \left(\frac{\hat{\mathbf{A}}_v^T \hat{\mathbf{A}}_v}{\hat{\sigma}_0^2} + \hat{\Sigma}_v^{-1} \right)^{-1} \right). \quad (33)$$

Note $\hat{\mathbf{A}}_v^T \hat{\mathbf{A}}_v = \mathbf{I}_{nd}$. Using fact 2 and fact 3, one has

$$\begin{aligned} \hat{\mathbf{Z}}_{vt} &= \begin{pmatrix} \left(\hat{\sigma}_0^2 \hat{\Sigma}_1^{-1} + \mathbf{I}_n \right)^{-1} \otimes \hat{\mathbf{a}}_1^T \\ \vdots \\ \left(\hat{\sigma}_0^2 \hat{\Sigma}_d^{-1} + \mathbf{I}_n \right)^{-1} \otimes \hat{\mathbf{a}}_d^T \end{pmatrix} \text{vec}(\mathbf{Y}) = \begin{pmatrix} \text{vec} \left(\hat{\mathbf{a}}_1^T \mathbf{Y} \left(\hat{\sigma}_0^2 \hat{\Sigma}_1^{-1} + \mathbf{I}_n \right)^{-1} \right) \\ \vdots \\ \text{vec} \left(\hat{\mathbf{a}}_d^T \mathbf{Y} \left(\hat{\sigma}_0^2 \hat{\Sigma}_d^{-1} + \mathbf{I}_n \right)^{-1} \right) \end{pmatrix} \\ &= \text{vec} \begin{pmatrix} \hat{\mathbf{a}}_1^T \mathbf{Y} \left(\hat{\sigma}_0^2 \hat{\Sigma}_1^{-1} + \mathbf{I}_n \right)^{-1} \\ \vdots \\ \hat{\mathbf{a}}_d^T \mathbf{Y} \left(\hat{\sigma}_0^2 \hat{\Sigma}_d^{-1} + \mathbf{I}_n \right)^{-1} \end{pmatrix}^T := \text{vec}(\hat{\mathbf{Z}}^T). \end{aligned} \quad (34)$$

Now we are ready to derive the predictive mean and predictive variance. First

$$\begin{aligned} \mathbb{E}[\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}] &= \mathbb{E}[\mathbb{E}[\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{Z}(\mathbf{x}^*), \hat{\boldsymbol{\theta}}]] = \mathbb{E}[\hat{\mathbf{A}}\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}] \\ &= \hat{\mathbf{A}} \mathbb{E}[\mathbb{E}[\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{Z}, \hat{\boldsymbol{\theta}}]] = \hat{\mathbf{A}} \hat{\mathbf{Z}}(\mathbf{x}^*) \end{aligned}$$

with the l th term of $\hat{\mathbf{Z}}(\mathbf{x}^*)$

$$\begin{aligned} \hat{Z}_l(\mathbf{x}^*) &= \hat{\Sigma}_l(\mathbf{x}^*) \Sigma_l^{-1} \mathbb{E}[\mathbf{Z}_l^T \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}] \\ &= \hat{\Sigma}_l^T(\mathbf{x}^*) \hat{\Sigma}_l^{-1} (\hat{\Sigma}_l^{-1} + \hat{\sigma}_0^2 \mathbf{I}_n)^{-1} \mathbf{Y}^T \hat{\mathbf{a}}_l \\ &= \hat{\Sigma}_l^T(\mathbf{x}^*) (\hat{\sigma}_0^2 \mathbf{I}_n + \hat{\Sigma}_l)^{-1} \mathbf{Y}^T \hat{\mathbf{a}}_l. \end{aligned}$$

where the first equality is from the property of multivariate normal distribution and the second equality is from (34).

Secondly, we have

$$\begin{aligned} &\mathbb{V}[\mathbf{Y}^* \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}] \\ &= \mathbb{E}[\mathbb{V}[\mathbf{Y}^* \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}(\mathbf{x}^*)]] + \mathbb{V}[\mathbb{E}[\mathbf{Y}^* \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}(\mathbf{x}^*)]] \\ &= \hat{\sigma}_0^2 \mathbf{I}_k + \mathbb{V}[\hat{\mathbf{A}}\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}] \\ &= \hat{\sigma}_0^2 \mathbf{I}_k + \hat{\mathbf{A}} [\mathbb{E}[\mathbb{V}[\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}]] + \mathbb{V}[\mathbb{E}[\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}]]] \hat{\mathbf{A}}^T = \hat{\sigma}_0^2 \mathbf{I}_k + \hat{\sigma}_0^2 \hat{\mathbf{A}} \hat{\mathbf{D}}(\mathbf{x}^*) \hat{\mathbf{A}}^T \end{aligned}$$

with $\hat{\mathbf{D}}(\mathbf{x}^*) = \frac{1}{\hat{\sigma}_0^2} (\mathbb{E}[\mathbb{V}[\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}]] + \mathbb{V}[\mathbb{E}[\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}]])$.

Note that $\mathbb{E}[\mathbb{V}[\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}]]$ is $k \times k$ diagonal matrix where the l th diagonal term is $\sigma_l^2 \hat{K}_l(\mathbf{x}^*, \mathbf{x}^*) - \hat{\Sigma}_l^T(\mathbf{x}^*) \hat{\Sigma}_l^{-1} \hat{\Sigma}_l(\mathbf{x}^*)$, and $\mathbb{V}[\mathbb{E}[\mathbf{Z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\theta}}, \mathbf{Z}]]$ is another $k \times k$ diagonal matrix where the i th diagonal term is $\sigma_l^2 \hat{K}_l(\mathbf{x}^*, \mathbf{x}^*) - \hat{\Sigma}_l^T(\mathbf{x}^*) (\hat{\sigma}_0^2 \mathbf{I}_n + \hat{\Sigma}_l)^{-1} \hat{\Sigma}_l(\mathbf{x}^*)$. Thus, by the

Woodbury matrix identity, $\hat{\mathbf{D}}(\mathbf{x}^*)$ is a diagonal matrix where the l th term is $\hat{\sigma}_l^2 K_l(\mathbf{x}^*, \mathbf{x}^*) - \hat{\Sigma}_l^T(\mathbf{x}^*)(\hat{\sigma}_0^2 \mathbf{I}_n + \hat{\Sigma}_l)^{-1} \hat{\Sigma}_l(\mathbf{x}^*)$ for $l = 1, \dots, d$. ■

Proof [Proof of Lemma 6] Denote $\tilde{\mathbf{M}} = \mathbf{M}/\sigma_0^2$. Using the prior $\pi(\mathbf{B}) \propto 1$, we first marginalizing out \mathbf{B} and the marginal density becomes

$$p(\mathbf{Y} | \mathbf{Z}, \mathbf{A}, \sigma_0^2, \Sigma_1, \dots, \Sigma_d) \propto (\sigma_0^2)^{-k(n-q)/2} \text{etr} \left(-\frac{(\mathbf{Y} - \mathbf{AZ})\tilde{\mathbf{M}}(\mathbf{Y} - \mathbf{AZ})^T}{2} \right).$$

Denote $\mathbf{Y}_{vt} = \text{vec}(\mathbf{Y}^T)$. By Fact 3, we have

$$\begin{aligned} & p(\mathbf{Y}, \mathbf{Z} | \mathbf{A}, \sigma_0^2, \Sigma_{1:d}) \\ & \propto (\sigma_0^2)^{-\frac{k(n-q)}{2}} \prod_{l=1}^d |\Sigma_l|^{-\frac{1}{2}} \text{etr} \left(-\frac{(\mathbf{Y} - \mathbf{AZ})\tilde{\mathbf{M}}(\mathbf{Y} - \mathbf{AZ})^T + \sum_{l=1}^d \mathbf{Z}_l \Sigma_l^{-1} \mathbf{Z}_l^T}{2} \right). \\ & \propto (\sigma_0^2)^{-\frac{k(n-q)}{2}} \prod_{l=1}^d |\Sigma_l|^{-\frac{1}{2}} \text{etr} \left(-\frac{\mathbf{Y}\tilde{\mathbf{M}}\mathbf{Y}^T}{2} \right) \\ & \quad \times \exp \left\{ -\frac{\mathbf{Z}_{vt}^T (\mathbf{I}_d \otimes \tilde{\mathbf{M}}) \mathbf{Z}_{vt} - 2\mathbf{Z}_{vt}^T (\mathbf{A}^T \otimes \tilde{\mathbf{M}}) \mathbf{Y}_{vt} + \mathbf{Z}_{vt}^T \Sigma_v^{-1} \mathbf{Z}_{vt}}{2} \right\} \end{aligned} \quad (35)$$

where $\mathbf{Z}_{vt} = \text{vec}(\mathbf{Z}^T)$ and Σ_v is an $nd \times nd$ block diagonal matrix, where the l th diagonal block is Σ_l , $l = 1, \dots, d$. Marginalizing out \mathbf{Z} , one has

$$\begin{aligned} & p(\mathbf{Y} | \mathbf{A}, \sigma_0^2, \Sigma_{1:d}) \\ & \propto (\sigma_0^2)^{-\frac{k(n-q)}{2}} \prod_{l=1}^d |\tilde{\mathbf{M}}\Sigma_l + \mathbf{I}_n|^{-\frac{1}{2}} \text{etr} \left(-\frac{\mathbf{Y}\tilde{\mathbf{M}}\mathbf{Y}^T}{2} \right) \\ & \quad \times \exp \left\{ -\frac{1}{2} \mathbf{Y}_{vt}^T (\mathbf{A}^T \otimes \tilde{\mathbf{M}})^T (\mathbf{I}_d \otimes \tilde{\mathbf{M}} + \Sigma_v^{-1})^{-1} (\mathbf{A}^T \otimes \tilde{\mathbf{M}}) \mathbf{Y}_{vt} \right\} \\ & \propto (\sigma_0^2)^{-\frac{k(n-q)}{2}} \prod_{l=1}^d |\tilde{\mathbf{M}}\Sigma_l + \mathbf{I}_n|^{-\frac{1}{2}} \text{etr} \left(-\frac{\mathbf{Y}\tilde{\mathbf{M}}\mathbf{Y}^T}{2} \right) \\ & \quad \times \exp \left\{ -\frac{1}{2} \mathbf{Y}_{vt}^T \left(\sum_{l=1}^d (\mathbf{a}_l \otimes \tilde{\mathbf{M}}) (\tilde{\mathbf{M}} + \Sigma_l^{-1})^{-1} (\mathbf{a}_l^T \otimes \tilde{\mathbf{M}}) \right) \mathbf{Y}_{vt} \right\} \\ & \propto (\sigma_0^2)^{-\frac{k(n-q)}{2}} \prod_{l=1}^d |\tilde{\mathbf{M}}\Sigma_l + \mathbf{I}_n|^{-\frac{1}{2}} \text{etr} \left(-\frac{\mathbf{Y}\tilde{\mathbf{M}}\mathbf{Y}^T}{2} \right) \\ & \quad \times \exp \left\{ -\frac{1}{2} \mathbf{Y}_{vt}^T \left(\sum_{l=1}^d (\mathbf{a}_l \mathbf{a}_l^T) \otimes \tilde{\mathbf{M}} (\tilde{\mathbf{M}} + \Sigma_l^{-1})^{-1} \tilde{\mathbf{M}} \right) \mathbf{Y}_{vt} \right\} \\ & \propto (\sigma_0^2)^{-\frac{k(n-q)}{2}} \prod_{l=1}^d |\mathbf{M}\tau_l \mathbf{K}_l + \mathbf{I}_n|^{-\frac{1}{2}} \exp \left\{ -\frac{\text{tr}(\mathbf{Y}\mathbf{M}\mathbf{Y}^T) - \sum_{l=1}^d \mathbf{a}_l^T \mathbf{Y}\mathbf{M}(\mathbf{M} + \tau_l^{-1} \mathbf{K}_l^{-1})^{-1} \mathbf{M}\mathbf{Y}^T \mathbf{a}_l}{2\sigma_0^2} \right\} \end{aligned}$$

Note for any $l = 1, \dots, d$

$$\begin{aligned}
 |\tau_l \mathbf{M} \mathbf{K}_l + \mathbf{I}_n| &= |\tau_l \mathbf{K}_l + \mathbf{I}_n| |\mathbf{I}_n - (\tau_l \mathbf{K}_l + \mathbf{I}_n)^{-1} \tau_l \mathbf{K}_l \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T| \\
 &= |\tau_l \mathbf{K}_l + \mathbf{I}_n| |\mathbf{H}^T \mathbf{H}|^{-1} |\mathbf{H}^T \mathbf{H} - \mathbf{H}^T ((\tau_l \mathbf{K}_l)^{-1} + \mathbf{I}_n)^{-1} \mathbf{H}| \\
 &= |\tau_l \mathbf{K}_l + \mathbf{I}_n| |\mathbf{H}^T \mathbf{H}|^{-1} |\mathbf{H}^T (\tau_l \mathbf{K}_l + \mathbf{I}_n)^{-1} \mathbf{H}|,
 \end{aligned}$$

where the first equation is by the definition of \mathbf{M} ; the second equation is based on Fact 4; the third equation is by the Woodbury matrix identity. Further maximizing over σ_0^2 and we have the result. \blacksquare

The following lemma is needed to prove Theorem 8.

Lemma 9 *Let $\tilde{\mathbf{M}} = \frac{1}{\sigma_0^2} (\mathbf{I}_n - \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T)$, where \mathbf{H} is a $n \times q$ matrix with $n > q$, and $\mathbf{H}^T \mathbf{H}$ is a $q \times q$ matrix with rank q . Further let $\tilde{\Sigma} = \Sigma + \sigma_0^2 \mathbf{I}_n$, where both Σ and $\tilde{\Sigma}$ have full rank. One has*

$$(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\mathbf{I}_n - \Sigma (\tilde{\mathbf{M}} \Sigma + \mathbf{I}_n)^{-1} \tilde{\mathbf{M}}) = (\mathbf{H}^T \tilde{\Sigma}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \tilde{\Sigma}^{-1} \quad (36)$$

Proof Denote $\Sigma_0 = \frac{1}{\sigma_0^2} \Sigma$. We start from the right hand side:

$$\begin{aligned}
 & (\mathbf{H}^T \tilde{\Sigma}^{-1} \mathbf{H})^{-1} \mathbf{H}^T \tilde{\Sigma}^{-1} \\
 &= (\mathbf{H}^T \mathbf{H} - \mathbf{H}^T (\Sigma_0^{-1} + \mathbf{I}_n)^{-1} \mathbf{H})^{-1} \mathbf{H}^T (\Sigma_0 + \mathbf{I}_n)^{-1} \\
 &= \{ (\mathbf{H}^T \mathbf{H})^{-1} - (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T - \Sigma_0^{-1} - \mathbf{I}_n)^{-1} \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \} \mathbf{H}^T (\Sigma_0 + \mathbf{I}_n)^{-1} \\
 &= (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\Sigma_0 + \mathbf{I}_n)^{-1} + (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\mathbf{M} + \Sigma_0^{-1})^{-1} \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\Sigma_0 + \mathbf{I}_n)^{-1} \\
 &= (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \{ \mathbf{I}_n - (\Sigma_0^{-1} + \mathbf{I}_n)^{-1} + (\mathbf{M} + \Sigma_0^{-1})^{-1} \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\Sigma_0 + \mathbf{I}_n)^{-1} \} \\
 &= (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \{ \mathbf{I}_n - (\mathbf{M} + \Sigma_0^{-1})^{-1} ((\mathbf{M} + \Sigma_0^{-1}) (\Sigma_0^{-1} + \mathbf{I}_n)^{-1} - \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\Sigma_0 + \mathbf{I}_n)^{-1}) \} \\
 &= (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \{ \mathbf{I}_n - (\mathbf{M} + \Sigma_0^{-1})^{-1} (\mathbf{I}_n - \mathbf{H} (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T) \} \\
 &= (\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\mathbf{I}_n - \Sigma (\tilde{\mathbf{M}} \Sigma + \mathbf{I}_n)^{-1} \tilde{\mathbf{M}}),
 \end{aligned}$$

where we repeatedly use the Woodbury matrix identity. \blacksquare

Proof [Proof of Theorem 8] Denote $\hat{\Theta} = (\hat{\mathbf{A}}, \hat{\gamma}, \hat{\sigma}^2, \hat{\sigma}_0^2)$. From Equation (35) in the proof of Lemma 6, one has

$$\mathbf{Z}_{vt} | \mathbf{Y}, \hat{\Theta} \sim \text{MN}(\hat{\mathbf{Z}}_{vt}, \hat{\Sigma}_{\mathbf{Z}_{vt}}), \quad (37)$$

where $\hat{\mathbf{Z}}_{vt} = \text{vec}(\hat{\Sigma}_1 (\mathbf{M} \hat{\Sigma}_1 + \hat{\sigma}_0^2 \mathbf{I}_n)^{-1} \mathbf{M} \mathbf{Y}^T \hat{\mathbf{a}}_1, \dots, \hat{\Sigma}_d (\mathbf{M} \hat{\Sigma}_d + \hat{\sigma}_0^2 \mathbf{I}_n)^{-1} \mathbf{M} \mathbf{Y}^T \hat{\mathbf{a}}_d)$ and $\hat{\Sigma}_{\mathbf{Z}_{vt}}$ is a $dn \times dn$ block diagonal matrix where the l th $n \times n$ diagonal block is $\hat{\sigma}_0^2 \hat{\Sigma}_l (\mathbf{M} \hat{\Sigma}_l + \hat{\sigma}_0^2 \mathbf{I}_n)^{-1}$.

It is also easy to obtain

$$\mathbf{B} | \mathbf{Y}, \mathbf{Z}, \sigma_0^2 \sim N((\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\mathbf{Y}^T - \mathbf{Z}^T \mathbf{A}^T), \sigma_0^2 \mathbf{I}_k \otimes (\mathbf{H}^T \mathbf{H})^{-1}). \quad (38)$$

Denote $z(\mathbf{x}^*) = (z_1(\mathbf{x}^*), \dots, z_d(\mathbf{x}^*))^T$ the factors at input \mathbf{x}^* . First the mean

$$\begin{aligned}
 \hat{\boldsymbol{\mu}}_M^*(\mathbf{x}^*) &= \mathbb{E}[\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\Theta}}] \\
 &= \mathbb{E}[\mathbb{E}[\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{B}, \mathbf{z}(\mathbf{x}^*), \hat{\boldsymbol{\Theta}}]] \\
 &= \mathbb{E}[(\mathbf{h}(\mathbf{x}^*)\mathbf{B})^T + \hat{\mathbf{A}}\mathbf{z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\Theta}}] \\
 &= \mathbb{E}[\mathbb{E}[(\mathbf{h}(\mathbf{x}^*)\mathbf{B})^T + \hat{\mathbf{A}}\mathbf{z}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\Theta}}, \mathbf{Z}]] \\
 &= \mathbb{E}[(\mathbf{Y} - \hat{\mathbf{A}}\mathbf{Z})\mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}^T(\mathbf{x}^*) + \hat{\mathbf{A}}\tilde{\mathbf{z}}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\Theta}}]
 \end{aligned}$$

where $\tilde{\mathbf{z}}(\mathbf{x}^*)$ is a d -dimensional vector where the each term is $\hat{\boldsymbol{\Sigma}}_l^T(\mathbf{x}^*)\hat{\boldsymbol{\Sigma}}_l^{-1}\mathbf{Z}_l^T$ for $l = 1, \dots, d$. From (37), noting $\mathbf{Z}_{vt} = \text{vec}(\mathbf{Z}^T)$, one has $\mathbb{E}[\mathbf{Z} \mid \mathbf{Y}, \hat{\boldsymbol{\Theta}}] = (\hat{\mathbf{Z}}_{1,M}^T, \dots, \hat{\mathbf{Z}}_{d,M}^T)^T$, with $\hat{\mathbf{Z}}_{l,M} = \mathbf{a}_l^T \mathbf{Y} \mathbf{M} (\hat{\boldsymbol{\Sigma}}_l \mathbf{M} + \hat{\sigma}_0^2 \mathbf{I}_n)^{-1} \hat{\boldsymbol{\Sigma}}_l$ is a $1 \times n$ vector, from which we have proved that equation (22) holds.

$$\begin{aligned}
 \hat{\boldsymbol{\Sigma}}_M^*(\mathbf{x}^*) &= \mathbb{V}[\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \hat{\boldsymbol{\Theta}}] \\
 &= \mathbb{V}[\mathbb{E}[\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{B}, \mathbf{z}(\mathbf{x}^*), \hat{\boldsymbol{\Theta}}]] + \mathbb{E}[\mathbb{V}[\mathbf{Y}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{B}, \mathbf{z}(\mathbf{x}^*), \hat{\boldsymbol{\Theta}}]] \\
 &= \mathbb{V}[(\mathbf{h}(\mathbf{x}^*)\mathbf{B})^T + \hat{\mathbf{A}}\mathbf{z}(\mathbf{x}^*) \mid \mathbf{Y}] + \sigma_0^2 \mathbf{I}_k \\
 &= \mathbb{V}[\mathbb{E}[(\mathbf{h}(\mathbf{x}^*)\mathbf{B})^T + \hat{\mathbf{A}}\mathbf{z}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{Z}]] + \mathbb{E}[\mathbb{V}[(\mathbf{h}(\mathbf{x}^*)\mathbf{B})^T + \hat{\mathbf{A}}\mathbf{z}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{Z}]] + \sigma_0^2 \mathbf{I}_k \\
 &= \mathbb{V}[(\mathbf{Y} - \hat{\mathbf{A}}\mathbf{Z})\mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}^T(\mathbf{x}^*) + \hat{\mathbf{A}}\tilde{\mathbf{z}}(\mathbf{x}^*) \mid \mathbf{Y}] + \hat{\mathbf{A}} \mathbb{V}[\mathbf{z}(\mathbf{x}^*) \mid \mathbf{Y}, \mathbf{Z}] \hat{\mathbf{A}}^T \\
 &\quad + \sigma_0^2 \mathbf{I}_k \otimes (1 + \mathbf{h}^T(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}(\mathbf{x}^*)) \\
 &= \hat{\mathbf{A}} \mathbf{D}_M(\mathbf{x}^*) \hat{\mathbf{A}} + \sigma_0^2 \mathbf{I}_k \otimes (1 + \mathbf{h}^T(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}(\mathbf{x}^*))
 \end{aligned}$$

where $\mathbf{D}_M(\mathbf{x}^*)$ is a diagonal matrix where the l th diagonal term is

$$\begin{aligned}
 D_{l,M} &= (\hat{\boldsymbol{\Sigma}}_l^T(\mathbf{x}^*)\hat{\boldsymbol{\Sigma}}_l^{-1} - \mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T)(\tilde{\mathbf{M}} + \hat{\boldsymbol{\Sigma}}_l^{-1})^{-1}(\hat{\boldsymbol{\Sigma}}_l^T(\mathbf{x}^*)\hat{\boldsymbol{\Sigma}}_l^{-1} - \mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T)^T \\
 &\quad + (\sigma_l^2 \hat{K}_l(\mathbf{x}^*, \mathbf{x}^*) - \hat{\boldsymbol{\Sigma}}_l^T(\mathbf{x}^*)\hat{\boldsymbol{\Sigma}}_l^{-1}\hat{\boldsymbol{\Sigma}}_l(\mathbf{x}^*))
 \end{aligned} \tag{39}$$

We write $D_{l,M} + \sigma_0^2 \mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}^T(\mathbf{x}^*)$ as the following three terms. First, one has

$$\begin{aligned}
 &\mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T(\tilde{\mathbf{M}} + \hat{\boldsymbol{\Sigma}}_l^{-1})^{-1}\mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}^T(\mathbf{x}^*) + \sigma_0^2 \mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{h}^T(\mathbf{x}^*) \\
 &= \sigma_0^2 \mathbf{h}(\mathbf{x}^*) \left\{ (\mathbf{H}^T\mathbf{H})^{-1} - (\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T \left(\mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1}\mathbf{H}^T - \mathbf{I}_n - \hat{\sigma}_0^2 \hat{\boldsymbol{\Sigma}}_l^{-1} \right)^{-1} \mathbf{H}(\mathbf{H}^T\mathbf{H})^{-1} \right\} \mathbf{h}^T(\mathbf{x}^*) \\
 &= \sigma_0^2 \mathbf{h}(\mathbf{x}^*) \left\{ \mathbf{H}^T\mathbf{H} - \mathbf{H}^T \left(\mathbf{I}_n + \hat{\sigma}_0^2 \hat{\boldsymbol{\Sigma}}_l^{-1} \right)^{-1} \mathbf{H} \right\}^{-1} \mathbf{h}^T(\mathbf{x}^*) \\
 &= \mathbf{h}(\mathbf{x}^*) \left\{ \mathbf{H}^T \left(\hat{\boldsymbol{\Sigma}}_l + \hat{\sigma}_0^2 \mathbf{I}_n \right)^{-1} \mathbf{H} \right\}^{-1} \mathbf{h}^T(\mathbf{x}^*),
 \end{aligned} \tag{40}$$

where the third and fourth equality is based on the Woodbury matrix identity.

Note

$$\begin{aligned}
 (\tilde{\mathbf{M}} + \hat{\Sigma}_l^{-1})^{-1} &= \left(\frac{\mathbf{I}_n}{\hat{\sigma}_0^2} + \hat{\Sigma}_l^{-1} - \frac{\mathbf{H}(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T}{\hat{\sigma}_0^2} \right)^{-1} \\
 &= \left(\frac{\mathbf{I}_n}{\hat{\sigma}_0^2} + \hat{\Sigma}_l^{-1} \right)^{-1} - \left(\frac{\mathbf{I}_n}{\hat{\sigma}_0^2} + \hat{\Sigma}_l^{-1} \right)^{-1} \mathbf{H} \left\{ \hat{\sigma}_0^2 \mathbf{H}^T \mathbf{H} - \mathbf{H}^T \left(\frac{\mathbf{I}_n}{\hat{\sigma}_0^2} + \hat{\Sigma}_l^{-1} \right)^{-1} \mathbf{H} \right\}^{-1} \mathbf{H}^T \left(\frac{\mathbf{I}_n}{\hat{\sigma}_0^2} + \hat{\Sigma}_l^{-1} \right)^{-1} \\
 &= \left(\frac{\mathbf{I}_n}{\hat{\sigma}_0^2} + \hat{\Sigma}_l^{-1} \right)^{-1} - \left(\mathbf{I}_n + \hat{\sigma}_0^2 \hat{\Sigma}_l^{-1} \right)^{-1} \mathbf{H} \left\{ \mathbf{H}^T \left(\hat{\Sigma}_l + \hat{\sigma}_0^2 \mathbf{I}_n \right)^{-1} \mathbf{H} \right\}^{-1} \mathbf{H}^T \left(\mathbf{I}_n + \hat{\sigma}_0^2 \hat{\Sigma}_l^{-1} \right)^{-1},
 \end{aligned}$$

by Woodbury matrix identity, one has

$$\begin{aligned}
 &(\hat{\Sigma}_l^T(\mathbf{x}^*) \hat{\Sigma}_l^{-1})(\tilde{\mathbf{M}} + \hat{\Sigma}_l^{-1})^{-1}(\hat{\Sigma}_l^T(\mathbf{x}^*) \hat{\Sigma}_l^{-1})^T - \hat{\Sigma}_l^T(\mathbf{x}^*) \hat{\Sigma}_l^{-1} \hat{\Sigma}_l(\mathbf{x}^*) \\
 &= -\hat{\Sigma}_l^T(\mathbf{x}^*) \hat{\Sigma}_l^{-1} \hat{\Sigma}_l(\mathbf{x}^*) - \hat{\Sigma}_l^T(\mathbf{x}^*) \hat{\Sigma}_l^{-1} \mathbf{H}(\mathbf{H}^T \hat{\Sigma}_l^{-1} \mathbf{H})^{-1} \mathbf{H}^T \hat{\Sigma}_l^{-1} \hat{\Sigma}_l(\mathbf{x}^*). \tag{41}
 \end{aligned}$$

Third, one has

$$\begin{aligned}
 &\mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T (\tilde{\mathbf{M}} + \hat{\Sigma}_l^{-1})^{-1} \hat{\Sigma}_l^{-1} \hat{\Sigma}_l(\mathbf{x}^*) \\
 &= \mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T \mathbf{H})^{-1} \mathbf{H}^T \left\{ \mathbf{I}_n - \hat{\Sigma}_l (\tilde{\mathbf{M}} \hat{\Sigma}_l + \mathbf{I}_n)^{-1} \tilde{\mathbf{M}} \right\} \hat{\Sigma}_l(\mathbf{x}^*) \\
 &= \mathbf{h}(\mathbf{x}^*)(\mathbf{H}^T \hat{\Sigma}_l^{-1} \mathbf{H})^{-1} \mathbf{H}^T \hat{\Sigma}_l^{-1} \hat{\Sigma}_l(\mathbf{x}^*). \tag{42}
 \end{aligned}$$

where the first equation is from the Woodbury matrix identity and the second equation is from Lemma 9.

From equation (40), (41) and (42), we have shown that equation (23) holds. ■

Appendix C: Simulated examples when models are misspecified

We discuss two numerical examples where the latent factor model is misspecified. First, we let the Assumption 1 be violated. In both examples, we assume that each entry of the factor loading matrix is sampled independently from a uniform distribution, hence not constrained in the Stiefel manifold. The second misspecification comes from the misuse of the kernel function in the factor processes. In reality, the smoothness of the true process may be unknown, therefore the use of any particular type of kernels may lead to an under-smoothing or over-smoothing scenario. Moreover, the factor may be an unknown deterministic function, rather than a sample from a Gaussian process. All these possible misspecifications will be discussed using the following Examples 4 and 5.

Example 4 (Unconstrained factor loadings and misspecified kernel functions) *The data are sampled from model (1) with $\Sigma_1 = \dots = \Sigma_d = \Sigma$ and $x_i = i$ for $1 \leq i \leq n$. Each entry of the factor loading matrix is assumed to be uniformly sampled from $[0, 1]$ independently (without the orthogonal constraints in (3)). The exponential kernel and the Gaussian kernel are assumed in generating the data with different combinations of σ_0^2 and n , while in the GPPCA, we still use the Matérn kernel function in (10) for the estimation. We assume*

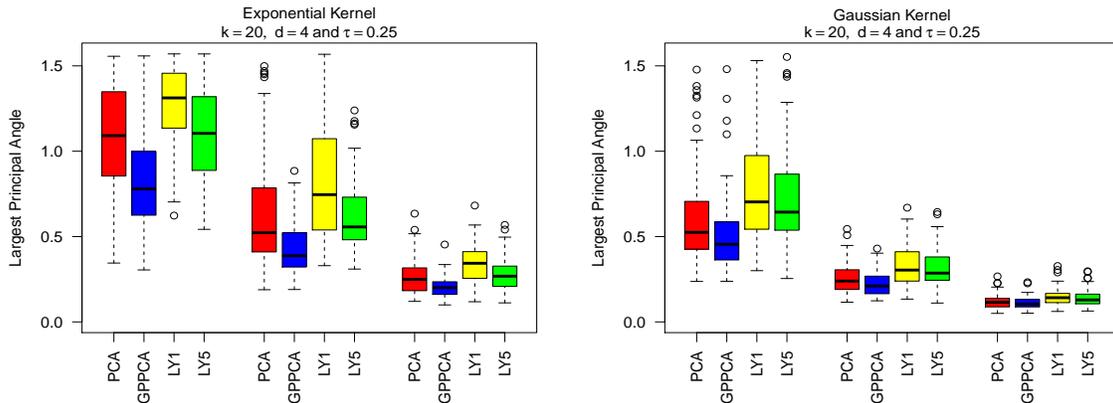


Figure 9: The largest principal angle between the estimated subspace of four approaches and the true subspace for Example 4. The number of observations are assumed to be $n = 100$, $n = 200$ and $n = 400$ for left 4 boxplots, middle 4 boxplots and right 4 boxplots in both panels, respectively. The kernel in simulating the data is assumed to be the exponential kernel in the left panel, whereas the kernel is assumed to be the Gaussian kernel in the right panel.

$k = 20$, $d = 4$, $\gamma = 100$ and $\sigma^2 = 1$ in sampling the data. We repeat $N = 100$ times for each scenario. All the kernel parameters and the noise variance are treated as unknown and estimated from the data.

The largest principal angles between $\mathcal{M}(\mathbf{A})$ and $\mathcal{M}(\hat{\mathbf{A}})$ of the four approaches for Example 4 are plotted in Figure 9. Even though the factor loading matrix is not constrained on the Stiefel manifold and the kernels are misspecified in GPPCA, GPPCA still has a better performance than other approaches in all scenarios. The PCA is an extreme case of the GPPCA where the range parameter of the kernel is estimated to be zero, meaning that the covariance of the factor process is an identity matrix.

Another interesting finding is that all methods seem to perform better when the Gaussian kernel is used in simulating the data, even if the SNR of the simulation using a Gaussian kernel is smaller. This is because the variation of the factors is much larger when the Gaussian kernel is used, which makes the effect of the noise relatively small. In both cases, the GPPCA seems to be efficient in estimating the subspace of the factor loading matrix.

Furthermore, since only the linear subspace of the factor loading matrix is identifiable, rather than the factor loading matrix, the estimation of the factor loadings without the orthogonal constraints is also accurate by the GPPCA. Note the interpretation of the estimated variance parameter in the kernel by the GPPCA changes, because each column of \mathbf{A} is not orthonormal in generating the data.

The AvgMSE of the four approaches for Example 4 is shown in Table 5. The estimation of the GPPCA is more accurate than the other approaches. Because of the larger variation in the factor processes with the Gaussian kernel, the corresponding variation in the mean

	exponential kernel and $\tau = 4$			Gaussian kernel and $\tau = 1/4$		
	$n = 100$	$n = 200$	$n = 400$	$n = 100$	$n = 200$	$n = 400$
PCA	7.4×10^{-2}	6.1×10^{-2}	5.4×10^{-2}	1.1×10^0	8.9×10^{-1}	8.4×10^{-1}
GPPCA	3.1×10^{-2}	2.6×10^{-2}	2.4×10^{-2}	7.2×10^{-1}	6.6×10^{-1}	6.2×10^{-1}
LY1	1.5×10^{-1}	8.2×10^{-1}	5.7×10^{-2}	1.3×10^0	1.0×10^0	8.6×10^{-1}
LY5	1.3×10^{-1}	7.3×10^{-1}	5.6×10^{-2}	1.3×10^0	1.0×10^0	8.6×10^{-1}

Table 5: AvgMSE for Example 4.

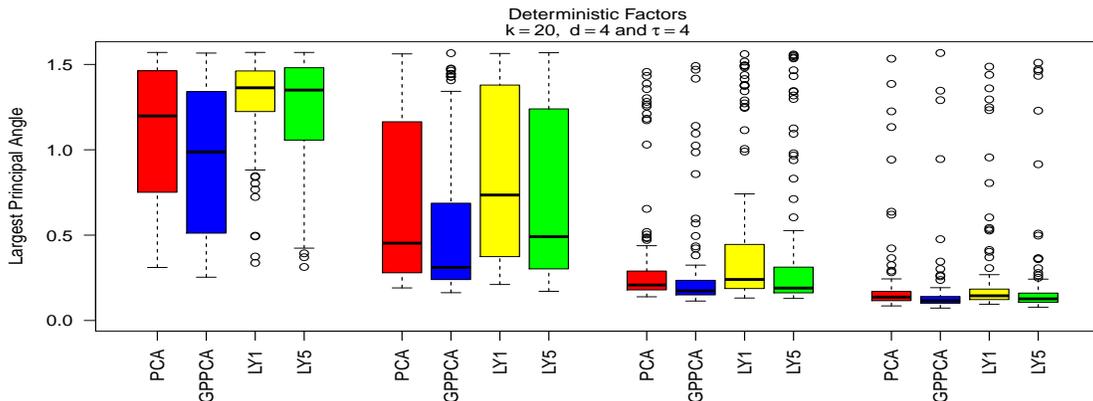


Figure 10: The largest principal angle between the estimated subspace of the loading matrix and the true subspace for Example 5. From the left to the right, the number of observations is assumed to be $n = 100$, $n = 200$, $n = 400$ and $n = 800$ for each 4 boxplots, respectively.

of the output is also larger than the one when the exponential kernel is used. Consequently, all approaches have larger estimated errors for the case with the Gaussian kernel.

We show an example when the factor is generated from a deterministic function.

Example 5 (Unconstrained factor loadings and deterministic factors) *The data are sampled from model (1) with each latent factor being a deterministic function*

$$Z_l(x_i) = \cos(0.05\pi\theta_l x_i)$$

where $\theta_l \stackrel{i.i.d.}{\sim} \text{unif}(0, 1)$ for $l = 1, \dots, d$, with $x_i = i$ for $1 \leq i \leq n$, $\sigma_0^2 = 0.25$, $k = 20$ and $d = 4$. Four cases are tested with the sample size $n = 100$, $n = 200$, $n = 400$ and $n = 800$.

For the GPPCA, we assume the covariance is shared for each factor and the Matérn kernel in (10) is used for Example 5. The largest principal angle between $\mathcal{M}(\mathbf{A})$ and $\mathcal{M}(\hat{\mathbf{A}})$ of the four approaches is given in Figure 10. When the number of observations increases, all four methods estimate $\mathcal{M}(\mathbf{A})$ more accurately, even though the factors are no longer

	$n = 100$	$n = 200$	$n = 400$	$n = 800$
PCA	7.0×10^{-2}	6.0×10^{-2}	5.4×10^{-2}	5.2×10^{-2}
GPPCA	1.4×10^{-2}	9.2×10^{-3}	6.7×10^{-3}	5.5×10^{-3}
LY1	9.8×10^{-1}	7.6×10^{-1}	6.3×10^{-2}	5.7×10^{-2}
LY5	9.3×10^{-2}	7.3×10^{-2}	6.2×10^{-2}	5.6×10^{-2}
Ind GP	2.0×10^{-2}	1.9×10^{-2}	1.7×10^{-2}	1.7×10^{-2}
PP GP	2.0×10^{-2}	1.9×10^{-2}	1.8×10^{-2}	1.8×10^{-2}

Table 6: AvgMSE for Example 5.

sampled from Gaussian processes. Note the reproducing kernel Hilbert space attached to the Gaussian process with the Matérn kernel contains those functions in the Sobolev space that are squared integrable up to the order 2, while the deterministic function to generate the factors in Example 5 is infinitely integrable. The GPPCA is the most precise method in estimating $\mathcal{M}(\mathbf{A})$ among the four approaches under this model misspecification.

The AvgMSE of the different approaches in estimating the mean of the output of the Example 5 is given in Table 6. We also include two more approaches, namely the independent Gaussian processes (Ind GP) and the parallel partial Gaussian processes (PP GP). The Ind GP approach treats each output variable independently and the mean of the output is estimated by the predictive mean in the Gaussian process regression (Rasmussen, 2006). The PP GP approach also models each output variable independently by a Gaussian process, whereas the covariance function is shared for k independent Gaussian processes and estimated based on all data (Gu and Berger, 2016).

As shown in Table 6, the estimation by the GPPCA is the most accurate among six approaches. The estimation by the Ind GP and PP GP perform similarly and they seem to perform better than the estimation by the PCA, LY1 and LY5. One interesting finding in Table 4 is that the AvgMSE by the GPPCA seems to decrease faster than those of the Ind GP and PP GP, when the sample size increases. This numerical result may shed some lights on the convergence rate of the GPPCA in the nonparametric regression problem.

Appendix D: Model fitting for the gridded temperature data

For the GPPCA, we consider the model (16), where the input x is an integer time point ranging from 1 to 240. The mean function is assumed as $\mathbf{h}(x) = (1, x)$ to model the trend of the temperature anomalies over time. For the case with estimated variance, the parameters are estimated by maximizing the marginal likelihood in (19) using the matrix of temperature anomalies \mathbf{Y} with $k = 1639$ rows and $n = 220$ columns. The marginal likelihood with known variance is derived by plugging the variance value, instead of integrating it out with a prior. We use Equation (24) to compute the predictive distribution by the GPPCA, where $\mathbf{Y}_1(x^*)$ is a 439×20 matrix of the temperature anomalies at the 439 spatial locations (which has the entire observations over the whole 240 months). The 1200×20 matrix $\mathbf{Y}_2(x^*)$ is the held-out temperature anomalies for testing.

For the PPCA, we first subtract the mean of each location of the 1639×220 output matrix of normalized temperature anomalies. We then estimate the factor loading matrix

by Equation (7) in Tipping and Bishop (1999). The predictive distribution of the test output by the PPCA was obtained in a similar fashion as the Equation (24) in the GPPCA and the empirical mean for the test output was added back for comparison. The PPCA does not incorporate the temporal correlation and linear trend in the model.

The temporal model is constructed by a GaSP separately for each test location. The Matérn kernel in (10) and the linear trend $\mathbf{h}(x) = (1, x)$ are assumed for the temporal model. The spatial model uses a GaSP with a constant mean separately for each test month. The product kernel in (8) is assumed for the two-dimensional input (latitude and longitude) and the Matérn kernel in (10) is used for each subkernel. The range and the nugget parameters in the temporal model and spatial model are estimated using the `RobustGaSP` R package (Gu et al. (2019)), and the predictions are also obtained by this package.

The temporal regression by the random forest are trained separately for each location. For each test month, the 439 observations of that month are used as the responses and the 439×220 output on the other months for the same locations are used as the covariates. The regression parameters of this temporal regression capture the temporal dependence of the output between the test month and the training months. The 1200×200 matrix of the temperature anomalies at the test locations and observed time points are used as the test input. The spatial regression by the random forest uses 220 observations of a test location as responses and the 220×439 matrix of the temperature anomalies of the observed locations are used as the input. The 20×439 matrix of the temperature anomalies at the observed locations and test time points are used as the test input. The `randomForest` R package (Liaw and Wiener, 2002) is used for training models and compute predictions.

The spatio-temporal model assumes a 3 dimensional product kernel in (8) for both time points and locations, and the Matérn kernel in (10) is used as the subkernel for each input variables. Note that if we use the whole training output, the computational order of inverting the covariance matrix is $O(N^3)$, where $N = 369360$ is the total number of inputs, which is computationally challenging. When the output can be written as an $n_1 \times n_2$ matrix, the likelihood corresponds to a matrix normal distribution, where two kernel functions model the correlation between rows and between columns of the output. The computational order of the matrix normal distribution is the maximum of $O(n_1^3)$ and $O(n_2^3)$. We choose the 439×240 temperature anomalies at the locations with the whole observations to estimate the parameters. The constant mean is assumed for each location. The MLE is used for estimating the range parameters in kernel, nugget, mean and variance parameters. After plugging in the parameters, the predictive distribution of the test data is used for predictions. Though only 439×240 observations are used for estimating the parameters due to the computational convenience, all 369360 training observations are used for computing the predictive distribution of the test output.

References

- P.-A. Absil, Alan Edelman, and Plamen Koev. On the largest principal angle between random subspaces. *Linear Algebra and its applications*, 414(1):288–294, 2006.
- Mauricio A Alvarez, Lorenzo Rosasco, and Neil D. Lawrence. Kernels for vector-valued functions: A review. *Foundations and Trends® in Machine Learning*, 4(3):195–266, 2012.

- Jushan Bai. Inferential theory for factor models of large dimensions. *Econometrica*, 71(1):135–171, 2003.
- Jushan Bai and Serena Ng. Determining the number of factors in approximate factor models. *Econometrica*, 70(1):191–221, 2002.
- M. J. Bayarri, James O. Berger, Eliza S. Calder, Keith Dalbey, Simon Lunagomez, Abani K. Patra, E. Bruce Pitman, Elaine T. Spiller, and Robert L. Wolpert. Using statistical and computer models to quantify volcanic hazards. *Technometrics*, 51:402–413, 2009.
- James O Berger, Victor De Oliveira, and Bruno Sansó. Objective bayesian analysis of spatially correlated data. *Journal of the American Statistical Association*, 96(456):1361–1374, 2001.
- James O Berger, José M Bernardo, and Dongchu Sun. The formal definition of reference priors. *The Annals of Statistics*, 37(2):905–938, 2009.
- A. Björck and Gene H Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of computation*, 27(123):579–594, 1973.
- Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- Stefano Conti and Anthony O’Hagan. Bayesian emulation of complex multi-output and dynamic computer models. *Journal of statistical planning and inference*, 140(3):640–651, 2010.
- Marian Farah, Paul Birrell, Stefano Conti, and Daniela De Angelis. Bayesian emulation and calibration of a dynamic epidemic model for A/H1N1 influenza. *Journal of the American Statistical Association*, 109(508):1398–1411, 2014.
- Thomas E Fricker, Jeremy E Oakley, and Nathan M Urban. Multivariate Gaussian process emulators with nonseparable covariance structures. *Technometrics*, 55(1):47–56, 2013.
- Alan E Gelfand, Alexandra M Schmidt, Sudipto Banerjee, and C. F. Sirmans. Nonstationary multivariate process modeling through spatially varying coregionalization. *Test*, 13(2):263–312, 2004.
- Alan E Gelfand, Peter Diggle, Peter Guttorp, and Montserrat Fuentes. *Handbook of spatial statistics*. CRC Press, 2010.
- Mengyang Gu. Jointly robust prior for Gaussian stochastic process in emulation, calibration and variable selection. *Bayesian Analysis*, 14(3):857–885, 2019.
- Mengyang Gu and James O Berger. Parallel partial Gaussian process emulation for computer models with massive output. *Annals of Applied Statistics*, 10(3):1317–1347, 2016.
- Mengyang Gu and Yanxun Xu. Fast nonseparable Gaussian stochastic process with application to methylation level interpolation. *Journal of Computational and Graphical Statistics, In Press*, 2019.

- Mengyang Gu, Xiaojing Wang, and James O Berger. Robust Gaussian stochastic process emulation. *Annals of Statistics*, 46(6A):3038–3066, 2018.
- Mengyang Gu, Jesús Palomo, and James O Berger. RobustGaSP: Robust Gaussian stochastic process emulation in R. *The R Journal*, 11(1):112–136, June 2019.
- Jouni Hartikainen and Simo Sarkka. Kalman filtering and smoothing solutions to temporal gaussian process regression models. In *Machine Learning for Signal Processing (MLSP), 2010 IEEE International Workshop for Signal Processing*, pages 379–384. IEEE, 2010.
- Dave Higdon, James Gattiker, Brian Williams, and Maria Rightley. Computer model calibration using high-dimensional output. *Journal of the American Statistical Association*, 103(482):570–583, 2008.
- Peter D Hoff. Bayesian analysis of matrix data with rstiefel. *arXiv preprint arXiv:1304.3673*, 2013.
- Heiko Hoffmann. Kernel PCA for novelty detection. *Pattern recognition*, 40(3):863–874, 2007.
- Ian Jolliffe. *Principal component analysis*. Springer, 2011.
- Effrosini Kokiopoulou, Jie Chen, and Yousef Saad. Trace optimization and eigenproblems in dimension reduction methods. *Numerical Linear Algebra with Applications*, 18(3):565–602, 2011.
- Clifford Lam and Qiwei Yao. Factor modeling for high-dimensional time series: inference for the number of factors. *The Annals of Statistics*, 40(2):694–726, 2012.
- Clifford Lam, Qiwei Yao, and Neil Bathia. Estimation of latent factors for high-dimensional time series. *Biometrika*, 98(4):901–918, 2011.
- Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R news*, 2(3):18–22, 2002.
- Fei Liu and Mike West. A dynamic modelling strategy for Bayesian computer model emulation. *Bayesian Analysis*, 4(2):393–411, 2009.
- Sebastian Mika, Bernhard Schölkopf, Alex J Smola, Klaus-Robert Müller, Matthias Scholz, and Gunnar Rätsch. Kernel PCA and de-noising in feature spaces. In *Advances in neural information processing systems*, pages 536–542, 1999.
- Jorge Nocedal. Updating quasi-newton matrices with limited storage. *Mathematics of computation*, 35(151):773–782, 1980.
- Jeremy Oakley. *Bayesian uncertainty analysis for complex computer codes*. PhD thesis, University of Sheffield, 1999.
- Antony M Overstall and David C Woods. Multivariate emulation of computer simulators: model selection and diagnostics with application to a humanitarian relief model. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 65(4):483–505, 2016.

- Rui Paulo, Gonzalo García-Donato, and Jesús Palomo. Calibration of computer models with multivariate output. *Computational Statistics and Data Analysis*, 56(12):3959–3974, 2012.
- Carl Edward Rasmussen. *Gaussian Processes for Machine Learning*. MIT Press, 2006.
- Yousef Saad. *Numerical Methods for Large Eigenvalue Problems*. Manchester University Press, 1992.
- Jerome Sacks, William J Welch, Toby J Mitchell, and Henry P Wynn. Design and analysis of computer experiments. *Statistical Science*, 4(4):409–423, 1989.
- Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.
- Matthias Seeger, Yee-Whye Teh, and Michael Jordan. Semiparametric latent factor models. Technical report, 2005.
- Samuel S.P. Shen. *R programming for climate data analysis and visualization: computing and plotting for NOAA data applications*. San Diego State University, San Diego, USA., 2017.
- B Taylor and A Lane. Development of a novel family of military campaign simulation models. *Journal of the Operational Research Society*, 55(4):333–339, 2004.
- Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.
- Zaiwen Wen and Wotao Yin. A feasible method for optimization with orthogonality constraints. *Mathematical Programming*, 142(1-2):397–434, 2013.
- M. West. Bayesian factor regression models in the “large p, small n” paradigm. In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. David, D. Heckerman, A. F. M. Smith, and M. West, editors, *Bayesian Statistics 7*, pages 723–732. Oxford University Press, 2003. URL <http://ftp.isds.duke.edu/WorkingPapers/02-12.html>.