A Generalized Approach to Machine Learning with Deep Gaussian Processes using Heterogeneous Data

Marzieh Ajirak* **and Petar M. Djurić****

*Weill Cornell Medical College
Cornell University
**Department of Electrical and Computer Engineering,
Stony Brook University

January 23, 2024

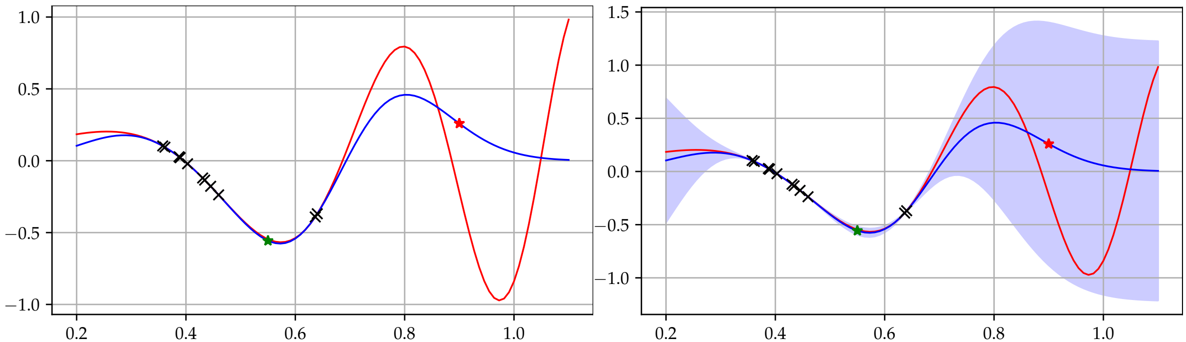
INTRODUCTION

- Machine learning has made significant strides in handling and analyzing heterogeneous data.
- Such data comprise diverse types of variables including numerical, categorical, count, and ordinal ones.
- Traditional modeling approaches face challenges in effectively handling such mixed datasets.
- For instance, electronic health records in hospitals contain various clinical measurements, diagnoses, and demographic information, combining numerical lab values with categorical variables such as race and blood type.
- The effective managing and extracting of meaningful insights from heterogeneous data holds immense importance.
- Machine learning tasks can be of different types including classification, regression, and imputation. Can we approach them in a unified way?

Deep generative models based on Gaussian processes

- Deep generative models are powerful unsupervised methods, capable of capturing latent structures in complex, high-dimensional data.
- Can deep structures and abstract learning be accomplished using smaller datasets?
- One class of such methods is known as deep Gaussian processes (DGPs).
- The building blocks of DGPs are Gaussian processes (GPs).
- GPs are Bayesian models that exploit distributions over functions, and they offer robustness against overfitting while providing a principled approach to tune hyperparameters and assess uncertainty bounds in their outputs.
- An extension of the use of GPs to unsupervised settings are GP latent variable models (GPLVMs), and they aim at learning smooth mappings from a latent space to the data space.
- These expressive unsupervised methods have demonstrated their ability to capture latent structures in complex, high-dimensional data.

An introductory example of a Gaussian process



(a) In-distribution (green star) and out-of-distribution (red star) test inputs.

(b) Area of uncertainty (shaded).

Figure 1

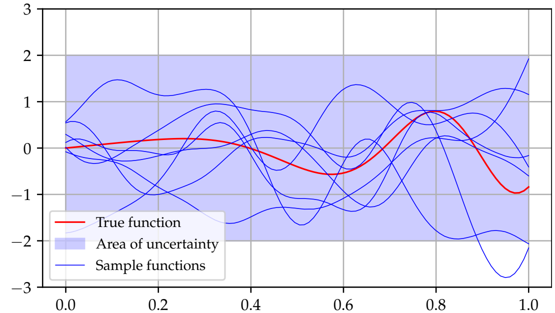
Gaussian processes

- A GP is a collection of random variables of which any finite subset has a multivariate Gaussian distribution.
- A GP is parameterized by its mean function and covariance function

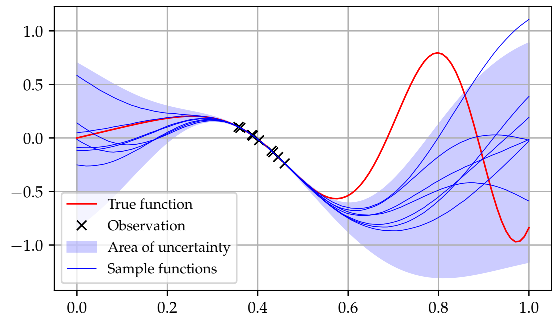
$$
\begin{aligned}
f(\mathbf{x}) &\sim \mathcal{GP}\left(m(\mathbf{x}), k\left(\mathbf{x}, \mathbf{x}'\right)\right) \\
m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\
k_\theta(\mathbf{x}, \mathbf{x}') &= \mathbb{E}\left[(f(\mathbf{x}) - m(\mathbf{x}))\left(f\left(\mathbf{x}'\right) - m\left(\mathbf{x}'\right)\right)\right]
\end{aligned}
$$

- We learn the hyperparameters by optimizing the log marginal likelihood

$$
\begin{aligned}
p(\mathbf{y}|\mathbf{X}) &= \int p(\mathbf{y}|\mathbf{f}, \mathbf{X}) p(\mathbf{f}|\mathbf{X}) d\mathbf{f} \\
\log p(\mathbf{y}|\mathbf{X}) &= -\frac{1}{2}\mathbf{y}^\top \left(\mathbf{K}_{NN} + \sigma_N^2 \mathbf{I}_N\right)^{-1} \mathbf{y} \\
&\quad - \frac{1}{2} \log \left|\mathbf{K} + \sigma_n^2 \mathbf{I}_N\right| - \frac{N}{2} \log 2\pi
\end{aligned}
$$



(a) Sample functions from the prior



(b) Sample functions from the posterior

Figure 2

Gaussian Process Latent Variable Models

- Unsupervised extension of GPs
- The outputs $\mathbf{Y} \in \mathbb{R}^{N \times D}$ are associated with inputs $\mathbf{X} \in \mathbb{R}^{N \times Q}$ through D different GPs

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

- The goal is to find the posterior of the latent input $\mathbf{X}$, $p(\mathbf{X}|\mathbf{Y})$.
- Standard variational inference

$$p(\mathbf{X}|\mathbf{Y}) \approx q(\mathbf{X}) = \prod_{n=1}^{N} \mathcal{N}(\mathbf{x}_n|\boldsymbol{\mu}_n, \mathbf{S}_n)$$
$$p(\mathbf{Y}) \geq \sum_{d=1}^{D} \int q(\mathbf{X}) \log p(\mathbf{y}_d|\mathbf{X}) d\mathbf{X} - \mathrm{KL}(q(\mathbf{X}) \| p(\mathbf{X}))$$

- Thus, instead of treating the latent variables as deterministic quantities, the Bayesian GPLVMs represent them as random variables following respective probability distributions.

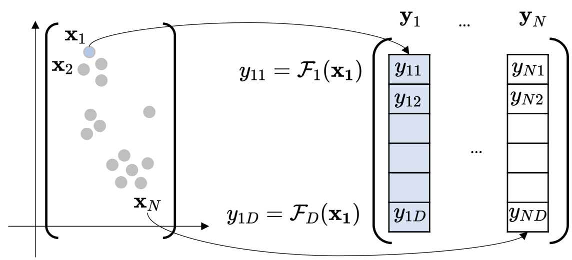
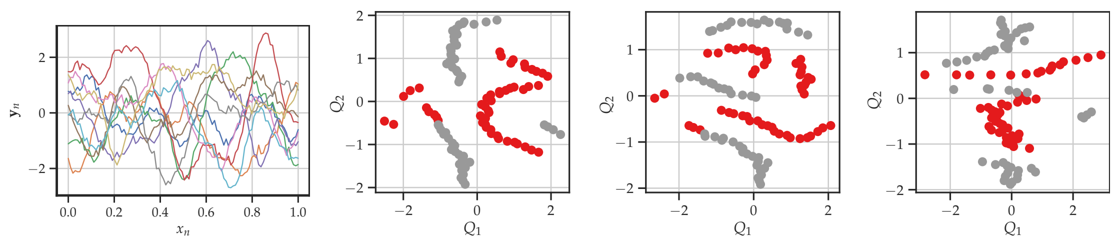


Figure 3. Mapping from a latent space to a data space

►



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

(b) Three different GPLVMs.

Figure 4. An example of 10-dimensional feature vectors projected on a two-dimensional space.

Inference with inducing points

- The model is defined as follows:

$$p(\mathbf{X}) = \prod_{n=1}^{N} p(\mathbf{x}_n)$$

$$p(\mathbf{F}|\mathbf{U}, \mathbf{X}, \boldsymbol{\theta}) = \prod_{d=1}^{D} \mathcal{N}\left(\mathbf{f}_d; \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{u}_d, \mathbf{R}_{NN}\right)$$

$$p(\mathbf{Y}|\mathbf{F}, \mathbf{X}, \sigma_y^2) = \prod_{n=1}^{N}\prod_{d=1}^{D} \mathcal{N}\left(y_{n,d}; \mathbf{f}_d\left(\mathbf{x}_n\right), \sigma_y^2\right)$$

  where $\mathbf{F} \in \mathbb{R}^{N\times D}$ and $\mathbf{U} \in \mathbb{R}^{M\times D}$; $\mathbf{K}_{NN}$ corresponds to a covariance matrix generated by evaluating a user-specified positive-definite kernel function $k_\theta\left(\mathbf{x}, \mathbf{x}'\right)$ on the latent points $\{\mathbf{x}_n\}_{n=1}^{N}$, with hyperparameters $\boldsymbol{\theta}$, which are shared across all dimensions D. Similarly, $\mathbf{K}_{MM}$ is a covariance matrix evaluated on the latent points $\{\mathbf{z}_m\}_{m=1}^{M}$. Finally,

$$\mathbf{R}_{NN} = \mathbf{K}_{NN} - \mathbf{K}_{NM}\mathbf{K}_{MM}^{-1}\mathbf{K}_{MN}$$

  where $\mathbf{K}_{NM} \in \mathbb{R}^{N\times M}$ and $\mathbf{K}_{MN} \in \mathbb{R}^{M\times N}$ are cross-covariance matrices evaluated at the latent points $\{\mathbf{x}_n\}_{n=1}^{N}$ and $\{\mathbf{z}_m\}_{m=1}^{M}$.
- The unknowns of the model are $\mathbf{F}, \mathbf{U}, \mathbf{X}$, $\boldsymbol{\theta}$, and σ_y^2.
- The joint posterior of interest is $p(\mathbf{F}, \mathbf{X}, \mathbf{U}, \boldsymbol{\theta}, \sigma_y^2|\mathbf{Y})$.
- Learning the unknowns is a highly nonlinear problem.

Evidence Lower Bound

$$
\begin{aligned}
\log p(\mathbf{Y}) &= \log \int p(\mathbf{X})p(\mathbf{U})p(\mathbf{F}|\mathbf{X},\mathbf{U})p(\mathbf{Y}|\mathbf{F})\mathrm{d}\mathbf{X}\mathrm{d}\mathbf{F}\mathrm{d}\mathbf{U} \\
&\geq -\mathrm{KL}(q(\mathbf{X})\|p(\mathbf{X})) - \mathrm{KL}(q(\mathbf{U})\|p(\mathbf{U})) \\
&\quad + \sum_{n=1}^{N}\sum_{d=1}^{D}\int q\left(\mathbf{x}_n\right) q\left(\mathbf{U}_d\right) p\left(\mathbf{f}_{nd}|\mathbf{x}_n,\mathbf{U}_d\right) \\
&\qquad \times \log p\left(\mathbf{y}_{nd}|\mathbf{f}_{nd}\right) \mathrm{d}\mathbf{x}_n \mathrm{d}\mathbf{f}_{nd}\mathbf{U}_d := \mathcal{L}
\end{aligned}
$$

Categorical variables

- The output $\mathbf{Y} \in \mathbb{R}^{N \times D}$ is categorical
- Motivation: clinical patient records

$$\boxed{\begin{array}{ccccc} \text{Exam } 1: & c_{11} & c_{12} & \cdots & c_{1K} \\ \text{Exam } 2: & c_{21} & c_{22} & \cdots & c_{2K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \text{Exam } D: & c_{D1} & c_{D2} & \cdots & c_{DK} \end{array}} \qquad \boldsymbol{y}_n = \begin{bmatrix} y_{n1} \\ y_{n2} \\ \vdots \\ y_{nD} \end{bmatrix}$$

- Form of a real-world database

$$\boldsymbol{Y} = \begin{bmatrix} y_{11} & y_{21} & \cdots & y_{N1} \\ y_{21} & y_{22} & \cdots & y_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ y_{D1} & y_{2D} & \cdots & y_{ND} \end{bmatrix}_{D \times N}$$

The involved distributions

- Evidence

$$p(\mathbf{Y}) = \int p(\mathbf{X})p(\mathbf{U})p(\mathbf{F}|\mathbf{X},\mathbf{U})p(\mathbf{Y}|\mathbf{F})\mathrm{d}\mathbf{X}\mathrm{d}\mathbf{F}\mathrm{d}\mathbf{U}$$

- The prior

$$p\left(\mathbf{f}_{nd}|\mathbf{x}_n,\mathbf{U}_d\right) = \prod_{k=1}^{K} \mathcal{N}(f_{ndk}; \mathbf{k}_{d,nM}^\top \mathbf{K}_{d,MM}^{-1}\mathbf{u}_{dk}, k_{d,nn} - \mathbf{k}_{d,nM}^\top \mathbf{K}_{d,MM}^{-1}\mathbf{k}_{d,Mn})$$

- The posterior distribution of $\mathbf{X}, \mathbf{F}$ and $\mathbf{U}$

$$q(\mathbf{X},\mathbf{F},\mathbf{U}) = q(\mathbf{X})q(\mathbf{U})p(\mathbf{F}|\mathbf{X},\mathbf{U}).$$

- We put variational distributions $q(\mathbf{X})$ and $q(\mathbf{U})$ on $\mathbf{X}$ and $\mathbf{U}$, respectively.

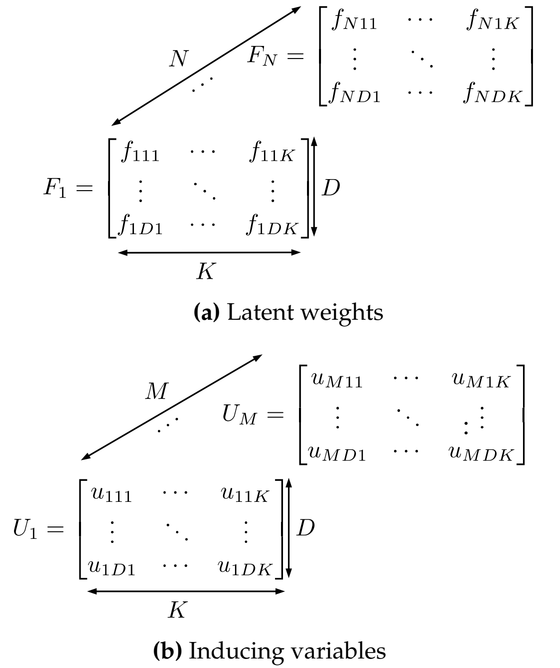


Figure 5. Latent weights and inducing variables.

Inference of the model

$$
\begin{aligned}
x_{nq} &\overset{\text{iid}}{\sim} \mathcal{N}\left(0, \sigma_x^2\right) \\
\mathcal{F}_{dk} &\overset{\text{iid}}{\sim} \mathcal{GP}\left(0, k_d\right) \\
f_{ndk} &= \mathcal{F}_{dk}\left(\mathbf{x}_n\right) \\
u_{mdk} &= \mathcal{F}_{dk}\left(\mathbf{z}_m\right) \\
p(y_{nd} = k) &= \frac{\exp\left(f_{ndk}\right)}{\sum_{k'=1}^{K} \exp\left(f_{ndk'}\right)}
\end{aligned}
$$

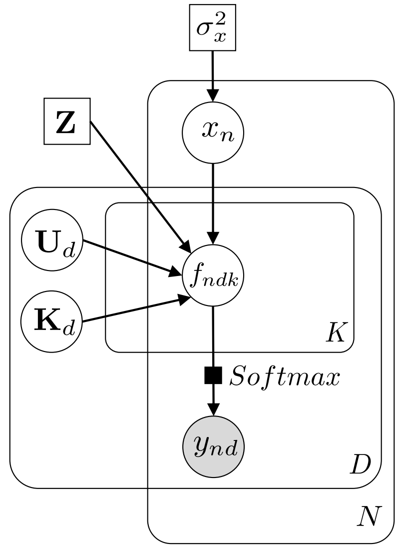


Figure 6. A graphical representation of a model for categorical variables.

Mixed categorical and numerical data

The generative model

$$\mathbf{x}_n \overset{\text{iid}}{\sim} \mathcal{N}(0, \boldsymbol{\Sigma})$$

$$\mathcal{F}_{d,k} \overset{\text{iid}}{\sim} \mathcal{GP}(0, \mathbf{K}_d), \qquad d = 1 : D_c$$

$$f_{n,d,k} = \mathcal{F}_{d,k}(\mathbf{x}_n), \qquad d = 1 : D_c$$

$$\mathcal{F}_d \overset{\text{iid}}{\sim} \mathcal{GP}(0, \mathbf{K}_d), \qquad d = D_c{+}1 : D_c{+}D_q$$

$$f_{n,d} = \mathcal{F}_d(\mathbf{x}_n), \qquad d = D_c{+}1 : D_c{+}D_q$$

$$p(y_{n,d} = k) = \frac{\exp(f_{n,d,k})}{\sum_{k'=1}^{K} \exp(f_{n,d,k'})}, \qquad d = 1 : D_c$$

$$p(y_{n,d}) = \mathcal{N}(f_{n,d}, \sigma_q^2), \qquad d = D_c{+}1 : D_c{+}D_q$$

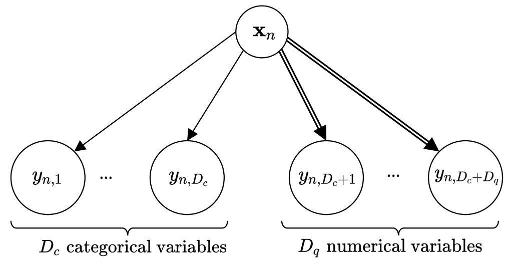


Figure 7. A generative model, where every dimension in the observation vector $\mathbf{y}_n = [y_{n1}, \ldots, y_{nD}]$ corresponds to either numerical or categorical variable.

Deep Gaussian processes latent variable models

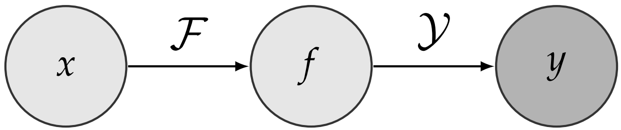


Figure 8. A two-layer DGPLVM. The functions $\mathcal{F}$ and $\mathcal{Y}$ are determined by the GPs.

- DGPs are organized as sequences of hidden layers of latent variables.
- The nodes in this architecture serve as inputs for the layer to the right, while the observed outputs reside in the leaves of the hierarchical structure.
- GPs play a crucial role in modeling the relationships between these layers.
- Each layer in the DGP is essentially a GPLVM, where latent variables can be approximately marginalized, allowing for the computation of a variational lower bound on the likelihood.
- The appropriate size of the latent spaces can be determined using automatic relevance determination (ARD) priors.

Multi-input – multi-output generalization

- Multi-layer generalization of GPs and GPLVMs
- Input layer $\mathbf{X} = \mathbf{F}_0 \in \mathbb{R}^{N \times Q}$
- Intermediate latent layers $\mathbf{F}^l \in \mathbb{R}^{N \times D^l}$ for $l = 1, \ldots, L$
- Observation layer, denoted as $\mathbf{Y} \in \mathbb{R}^{N \times D}$
- The layers are characterized by inducing inputs $\mathbf{Z}^l$ and inducing outputs $\mathbf{U}^l$
- The input $\mathbf{X}$ can be unobserved with our choice of prior

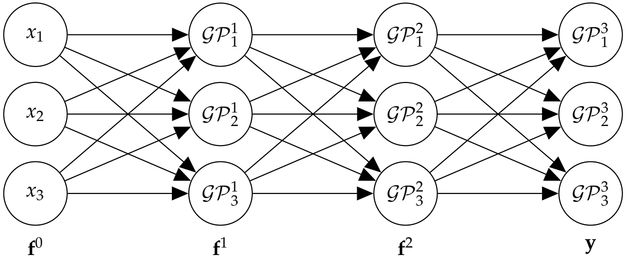


Figure 9. A network of GPs.

A TWO-STAGE FRAMEWORK FOR DEEP GPLVM

The generative model:

$$\mathbf{x}_n \sim p(\mathbf{x})$$
$$\mathcal{F}_d \sim \mathcal{GP}\left(0, k_d^f(.,.)|\boldsymbol{\theta}_d^f\right)$$
$$f_{nd} = \mathcal{F}_d(\mathbf{x}_n)$$
$$\mathcal{Y}_d \sim \mathcal{GP}\left(0, k_d^y(.,.|\boldsymbol{\theta}_d^y)\right)$$
$$y_{nd} = \mathcal{Y}_d(f_{nd})$$

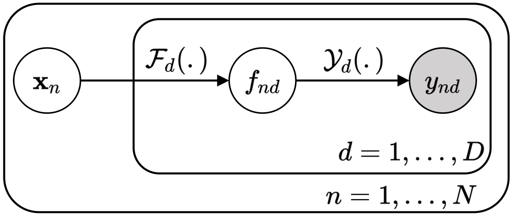


Figure 10. A graphical representation of the generative model.

Inference

- The marginal distribution of each variable is defined by

$$p_{\theta_d}(y_{nd}) = \mathbb{E}_{p(f_{nd})} p_{\theta_d}(y_{nd}|f_{nd}) \tag{1}$$

- The optimization objective is to maximize

$$\sum_{n=1}^{N} \mathbb{E}_{q_{\phi_d}(f_{nd}|y_{nd})} \log \frac{p_{\theta_d}(y_{nd}|f_{nd})p(f_{nd})}{q_{\phi_d}(f_{nd}|y_{nd})} \tag{2}$$

- The model of f_{nd} is given by

$$f_{nd} \sim q_{\phi_d}\left(f_{nd}|y_{nd}\right), \quad \forall d \in \{1, \ldots, D\} \tag{3}$$

- The optimization objective is to maximize

$$\sum_{n=1}^{N} \mathbb{E}_{q_\lambda(\mathbf{x}_n|\mathbf{f}_n, \mathbf{y}_n)} \log \frac{p_\psi\left(\mathbf{f}_n, \mathbf{x}_n\right)}{q_\lambda\left(\mathbf{x}_n|\mathbf{f}_n, \mathbf{y}_n\right)} \tag{4}$$

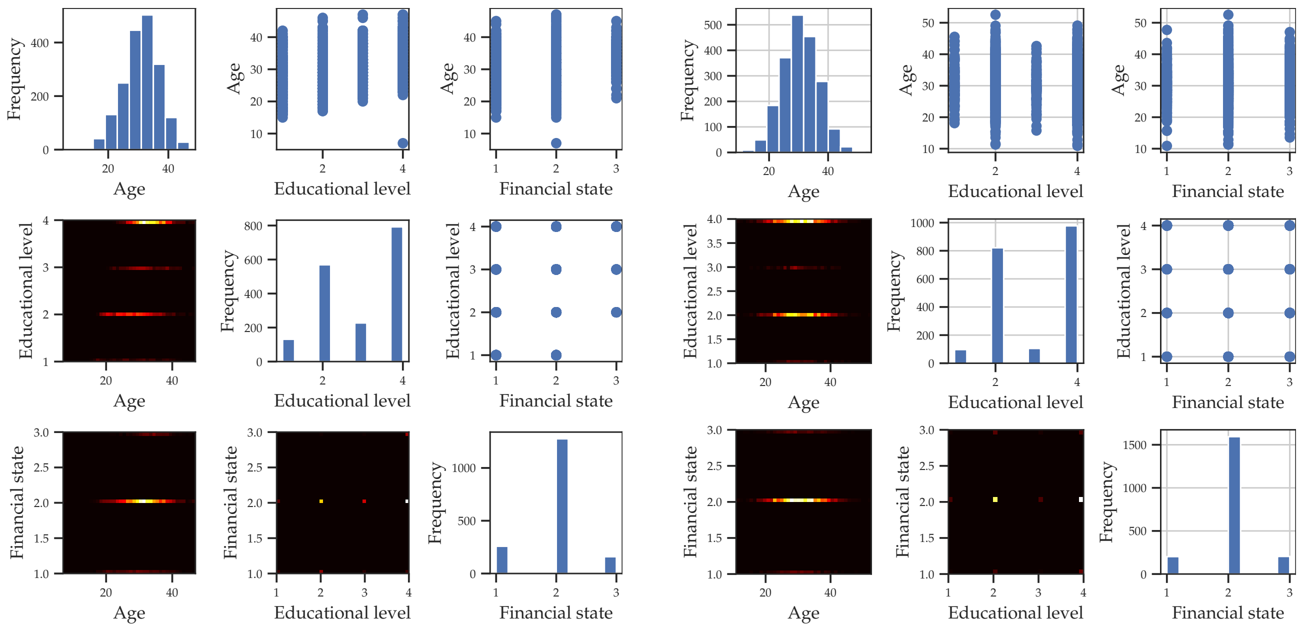
Experiments and Results on PROMOTE data

- $$AvgErr = \frac{1}{D} \sum_{d} \text{err}(d) \tag{5}$$
- $$\text{err}(d) = \frac{1}{n} \sum_{n=1}^{N} I\left(y_{nd} \neq \hat{y}_{nd}\right) \tag{6}$$
- $$\text{err}(d) = \frac{\sqrt{\frac{1}{n} \sum_{n=1}^{N} \left(y_{nd} - \hat{y}_{nd}\right)^2}}{\max\left(\mathbf{y}_d\right) - \min\left(\mathbf{y}_d\right)} \tag{7}$$

Table 1. Average imputation error for different variable types with 20% of missing data of each variable.

	Depression (Continuous)	Financial (Categorical)	Emotional (Binary)
Mean Imputation	0.277	0.237	0.362
One-hot/Iterative	0.240	0.231	0.359
HI-GP	0.246	0.215	0.347
Two-stage-GP	0.230	0.214	0.338

Experiments and Results (contd.)



(a) Original data **(b)** Data generated by the two-stage GP

Figure 11. Distributions of the original and generated data. Within each subfigure, histograms for each dimension are displayed on the diagonal, while off-diagonal plots illustrate joint distributions among the dimensions.

Conclusions

- We discussed deep Gaussian process latent variable models for processing heterogeneous data.
- The main idea is that the generative model of all the heterogeneous data uses the same latent input to produce all the data.
- The latent input data undergo two transformations, both represented by sets of Gaussian processes.
- We optimize our model by using variational inference and exploiting the concept of inducing points.
- The model was tested on a dataset called PROMOTE, which is used for studying unwanted perinatal outcomes and maternal mental health morbidities.
- The results suggest that the deep Gaussian process latent variable model has an excellent capacity to learn from heterogeneous data.
- If we have missing output data, the machine learning task is to predict them, which may amount to regression, classification, or imputation.