# Scaling Up Deep Gaussian Processes

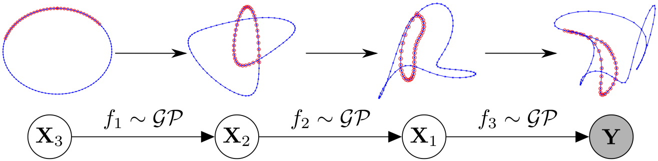
Zhenwen Dai

University of Sheffield

*z.dai@sheffield.ac.uk*

June 8, 2016

# Deep GPs




$$\mathbf{Y} = f_1(\mathbf{X}_1) + \epsilon_1, \quad \epsilon_1 \sim \mathcal{N}(0, \sigma_1^2 \mathbf{I})$$

$$\mathbf{X}_{l-1} = f_l(\mathbf{X}_l) + \epsilon_l, \quad \epsilon_l \sim \mathcal{N}(0, \sigma_l^2 \mathbf{I}), \quad l = 2 \dots L$$

$$f_l(x) \sim \mathcal{GP}(0, k_l(x, x'))$$

# Motivations

- Represent a complex family of functions with one type of kernels such as RBF.
- Cubic scaling with width, while linear scaling with depth

# Limitations

- Damianou and Lawrence [2013] show the results with a small number of data points.
- The number of variational parameters scales linearly with the size of data.

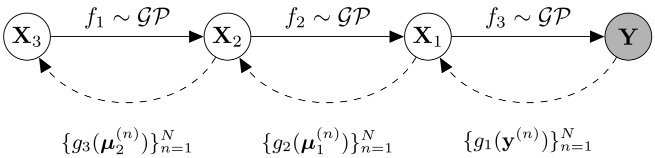
# Parallelization / GPU

- Parallelization of Bayesian GPLVM and Sparse GP ([Gal et al., 2014, Dai et al., 2014])
- Go towards millions of data points
- Deep GPs

# Variational Auto-Encoder / Back-constrained GP

- [Lawrence and Quiñonero-Candela, 2006]
- [Kingma and Welling, 2013]

# Variational Auto-encoder in DGP

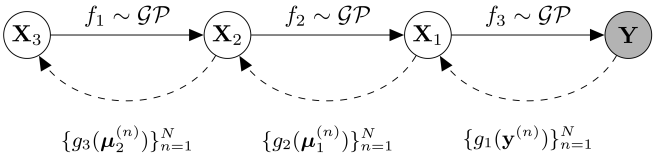



- Mean-field approximation for $\mathbf{X}$: $q(\mathbf{X}) = q(\mathbf{X}_1)q(\mathbf{X}_2)q(\mathbf{X}_3)$
- Assume Gaussian distribution:

$$q(\mathbf{X}_1) = \prod_{n=1}^{N} \mathcal{N}(\mathbf{x}_1^{(n)} | \boldsymbol{\mu}_1^{(n)}, \Sigma_1)$$

- Reparameterization of variational posteriors: $\boldsymbol{\mu}_1^{(n)} = g_1(\mathbf{y}^{(n)})$, $\boldsymbol{\mu}_2^{(n)} = g_2(\boldsymbol{\mu}_1^{(n)})$, $\boldsymbol{\mu}_3^{(n)} = g_3(\boldsymbol{\mu}_2^{(n)})$
- Deterministic transformations: $\boldsymbol{\mu}_l^{(n)} = g_l(\ldots g_1(\mathbf{y}^{(n)}))$.

# Variational Auto-encoder in DGP




The choice of $\Sigma_1$?

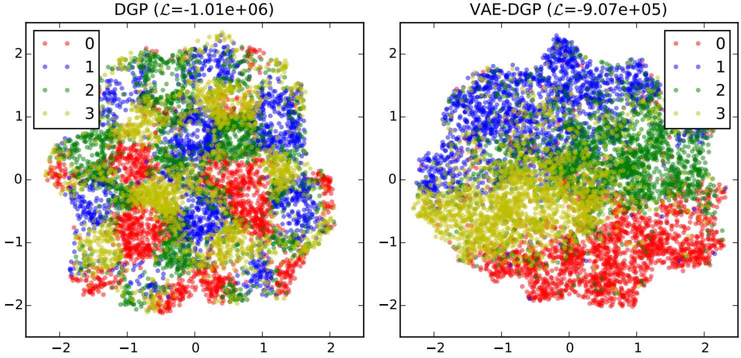
- constant for all the data points (our choice)
- $(\Sigma_1^{(n)})_{ii} = \exp\left(h_{1i}(\mathbf{y}^{(n)})\right)$
- Normalizing flow [Rezende and Mohamed, 2015]
- Variational GP [Tran et al., 2016]

# Benefits: Avoid Local Optima

Unsupervised Learning with the *Same Initialization*

one layer with 2D latent space (Bayesian GPLVM)

10,000 noisy MNIST digits (0,1,2,3)



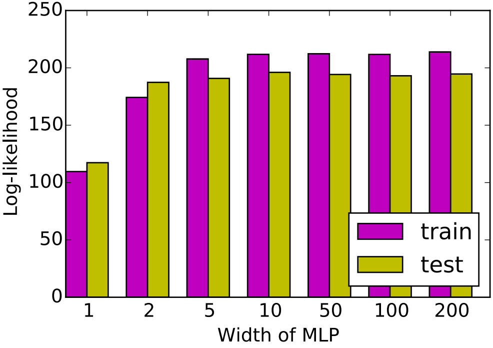

https://youtu.be/4IryFhMYvn4

# Benefits: Recognition Model does not Overfit

Unsupervised Learning with varying the size of recognition model

one layer with 2D latent space (Bayesian GPLVM)

2,000 MNIST digit ‘0’



# Parallelization of Deep GPs

The marginal likelihood for two hidden layer model:

$$p(\mathbf{Y}) = \int p(\mathbf{Y}|\mathbf{X}_1)p(\mathbf{X}_1|\mathbf{X}_2)p(\mathbf{X}_2)\mathsf{d}\mathbf{X}_1\mathsf{d}\mathbf{X}_2$$

The variational lower bound:

$$\begin{aligned}\log p(\mathbf{Y}) \geq & \int q(\mathbf{X}_1)q(\mathbf{X}_2)\log\frac{p(\mathbf{Y}|\mathbf{X}_1)p(\mathbf{X}_1|\mathbf{X}_2)p(\mathbf{X}_2)}{q(\mathbf{X}_1)q(\mathbf{X}_2)}\mathsf{d}\mathbf{X}_1\mathsf{d}\mathbf{X}_2 \\ = & \left\langle p(\mathbf{Y}|\mathbf{X}_1)\right\rangle_{q(\mathbf{X}_1)} + \left\langle p(\mathbf{X}_1|\mathbf{X}_2)\right\rangle_{q(\mathbf{X}_2)} + H(q(\mathbf{X}_1)) \\ & - \mathsf{KL}\left(q(\mathbf{X}_2)\,\|\,p(\mathbf{X}_2)\right)\end{aligned}$$

# Parallelization of Deep GPs

$p(\mathbf{Y}|\mathbf{X}_1)$ and $p(\mathbf{X}_1|\mathbf{X}_2)$ are both GPs.
With the sparse GP formulation and Gaussian likelihood distribution:

$$p(\mathbf{Y}|\mathbf{X}_1) = \int p(\mathbf{Y}|\mathbf{F}_1)p(\mathbf{F}_1|\mathbf{U}_1, \mathbf{X}_1)p(\mathbf{U}_1)\mathrm{d}\mathbf{F}_1\mathrm{d}\mathbf{U}_1,$$

With the variational approximation:
$q(\mathbf{F}_1, \mathbf{U}_1|\mathbf{X}_1) = p(\mathbf{F}_1|\mathbf{U}_1, \mathbf{X}_1)q(\mathbf{U}_1)$

$$\begin{aligned}\langle p(\mathbf{Y}|\mathbf{X}_1)\rangle_{q(\mathbf{X}_1)} \geq \langle \log p(\mathbf{Y}|\mathbf{F}_1)\rangle_{p(\mathbf{F}_1|\mathbf{U}_1,\mathbf{X}_1)q(\mathbf{U}_1)q(\mathbf{X}_1)} \\ - \mathsf{KL}\left(q(\mathbf{U}_1) \,\|\, p(\mathbf{U}_1)\right)\end{aligned}$$

# Parallelization of Deep GPs

Gal et al. [2014], Dai et al. [2014] show the computation can be parallelized following *data parallelism*.

$$\mathrm{Tr}(\mathbf{Y}^\top \mathbf{Y}) = \sum_{n=1}^{N} (\mathbf{y}^{(n)})^\top \mathbf{y}^{(n)},$$

$$\mathrm{Tr}(\mathbf{\Lambda}_1^{-1} \mathbf{\Psi}_1^\top \mathbf{Y}\mathbf{Y}^\top \mathbf{\Psi}_1) = \mathrm{Tr}\left( \mathbf{\Lambda}_1^{-1} \left( \sum_{n=1}^{N} \mathbf{\Psi}_1^{(n)} (\mathbf{y}^{(n)})^\top \right) \left( \sum_{n=1}^{N} \mathbf{\Psi}_1^{(n)} (\mathbf{y}^{(n)})^\top \right)^\top \right),$$

$$\mathbf{\Psi}_1 = \left\langle \mathbf{K}_{\mathbf{F}_1\mathbf{U}_1} \right\rangle_{q(\mathbf{X}_1)}, \ \mathbf{\Phi}_1 = \left\langle \mathbf{K}_{\mathbf{F}_1\mathbf{U}_1}^\top \mathbf{K}_{\mathbf{F}_1\mathbf{U}_1} \right\rangle_{q(\mathbf{X}_1)},$$

$$\mathbf{\Lambda}_1 = \mathbf{K}_{\mathbf{U}_1\mathbf{U}_1} + \mathbf{\Phi}_1$$

# Parallelization of Deep GPs

In deep GPs, $p(\mathbf{X}_1|\mathbf{X}_2)$ leads to the computation of $\mathrm{Tr}(\left\langle \mathbf{X}_{l-1}^\top \mathbf{X}_{l-1} \right\rangle_{q(\mathbf{X}_{l-1})})$ and $\mathrm{Tr}(\mathbf{\Lambda}_l^{-1} \mathbf{\Psi}_l^\top \left\langle \mathbf{X}_{l-1} \mathbf{X}_{l-1}^\top \right\rangle_{q(\mathbf{X}_{l-1})} \mathbf{\Psi}_l)$.

$$\mathrm{Tr}(\left\langle \mathbf{X}_{l-1}^\top \mathbf{X}_{l-1} \right\rangle_{q(\mathbf{X}_{l-1})}) = \sum_{n=1}^{N} (\boldsymbol{\mu}_{l-1}^{(n)})^\top \boldsymbol{\mu}_{l-1}^{(n)} + \mathrm{Tr}(\mathbf{\Sigma}_{l-1}^{(n)})$$

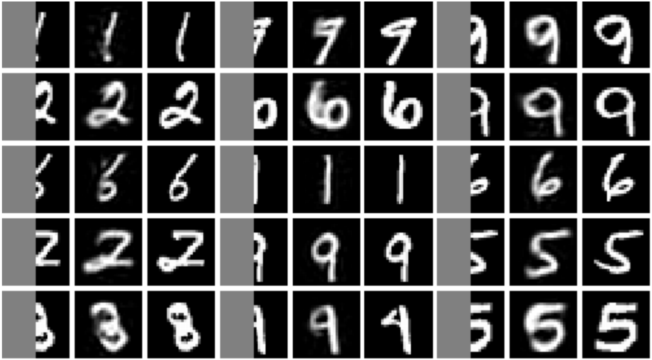
# Parallelization of Deep GPs

For the second term, we can rewrite $\left\langle \mathbf{X}_{l-1}\mathbf{X}_{l-1}^\top \right\rangle_{q(\mathbf{X}_{l-1})} = \mathbf{R}_{l-1}^\top \mathbf{R}_{l-1} + \mathbf{A}_{l-1}\mathbf{A}_{l-1}$, where $\mathbf{R}_{l-1} = [(\boldsymbol{\mu}_{(l-1)}^{(1)})^\top, \ldots, (\boldsymbol{\mu}_{(l-1)}^{(N)})^\top]$, $\mathbf{A}_{l-1} = \mathrm{diag}(\alpha_{l-1}^{(1)}, \ldots, \alpha_{l-1}^{(N)})$ and $\alpha_{l-1}^{(n)} = \mathrm{Tr}(\boldsymbol{\Sigma}_{l-1}^{(n)})^{\frac{1}{2}}$. This enables us to formulate it into a distributable form:

$$\mathrm{Tr}(\boldsymbol{\Lambda}_l^{-1}\boldsymbol{\Psi}_l^\top \left\langle \mathbf{X}_{l-1}\mathbf{X}_{l-1} \right\rangle_{q(\mathbf{X}_{l-1})}^\top \boldsymbol{\Psi}_l) = \mathrm{Tr}\left(\boldsymbol{\Lambda}_l^{-1}\left(\boldsymbol{\Psi}_l^\top \mathbf{R}_{l-1}^\top\right)\left(\mathbf{R}_{l-1}\boldsymbol{\Psi}_l\right)\right)$$
$$+ \mathrm{Tr}\left(\boldsymbol{\Lambda}_l^{-1}\left(\sum_{n=1}^N \boldsymbol{\Psi}_l^{(n)}\alpha_{l-1}^{(n)}\right)\left(\sum_{n=1}^N \boldsymbol{\Psi}_l^{(n)}\alpha_{l-1}^{(n)}\right)^\top\right).$$

# Yale + Frey Faces

# MNIST Imputation



# MNIST Test log-likelihood

| Model | MNIST |
| --- | --- |
| DBN | 138$\pm$2 |
| Stacked CAE | 121 $\pm$ 1.6 |
| Deep GSN | 214 $\pm$ 1.1 |
| Adversarial nets | 225 $\pm$ 2 |
| GMMN+AE | 282 $\pm$ 2 |
| VAE-DGP (5) | 301.67 |
| VAE-DGP (10-50) | 674.86 |
| VAE-DGP (5-20-50) | 723.65 |

# SVHN Imputation

# Future Directions

- Improve the recognition model with a better handling of variance.
- Conditional Variational Posterior $q(\mathbf{X}_2|\mathbf{X}_1)q(\mathbf{X}_1)$ instead of mean-field $q(\mathbf{X}_2)q(\mathbf{X}_1)$
- Stochastic Variational Inference (like [Bui and Turner, 2015])

# Collaborators

- Andreas Damianou
- Javier González
- Neil Lawrence

# References


Thang D Bui and Richard E Turner. Stochastic Variational Inference for Gaussian Process Latent Variable Models using Back Constraints. In *Black box learning & inference workshop (NIPS)*, 2015.

Zhenwen Dai, Andreas Damianou, James Hensman, and Neil Lawrence. Gaussian process models with parallelization and GPU acceleration, 2014.

Andreas Damianou and Neil D. Lawrence. Deep Gaussian processes. volume 31, pages 207–215, 2013.

Yarin Gal, Mark van der Wilk, and Carl E Rasmussen. Distributed Variational Inference in Sparse Gaussian Process Regression and Latent Variable Models. In *Advances in Neural Information Processing System*, 2014.

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. In *ICLR*, 2013.

Neil Lawrence and Joaquin Quiñonero-Candela. Local distance preservation in the GP-LVM through back constraints. In *International Conference on Machine Learning*, pages 513–520, 2006.

Danilo Jimenez Rezende and Shakir Mohamed. Variational Inference with Normalizing Flows. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1530–1538, 2015.