Random Feature Expansions for Deep Gaussian Processes

Kurt Cutajar [1], Edwin V. Bonilla [2], Pietro Michiardi [1], Maurizio Filippone [1]

[1] EURECOM, Sophia Antipolis, France
[2] University of New South Wales, Sydney, Australia

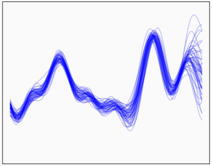
ICML 2017 - August 9th 2017

Deep Gaussian Processes

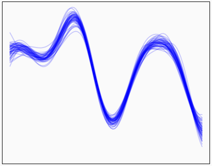
- Deep probabilistic models
- Composition of functions

$$\mathrm{f}(\mathrm{x}) = \left(\mathrm{h}^{(N_{\mathrm{h}}-1)}\left(\boldsymbol{\theta}^{(N_{\mathrm{h}}-1)}\right) \circ \ldots \circ \mathrm{h}^{(0)}\left(\boldsymbol{\theta}^{(0)}\right)\right)(\mathrm{x})$$

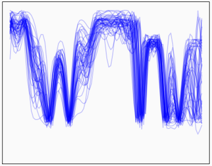
$\mathrm{h}^{(0)}(\mathrm{x})$



$\mathrm{h}^{(1)}(\mathrm{x})$



$\mathrm{h}^{(1)}\left(\mathrm{h}^{(0)}(\mathrm{x})\right)$



- Inference requires calculating the marginal likelihood:

$$
\begin{aligned}
p(Y|X,\boldsymbol{\theta}) \quad = \quad & \int p\left(Y|F^{(N_\mathrm{h})},\boldsymbol{\theta}^{(N_\mathrm{h})}\right) \times \\
& p\left(F^{(N_\mathrm{h})}|F^{(N_\mathrm{h}-1)},\boldsymbol{\theta}^{(N_\mathrm{h}-1)}\right) \times \ldots \times \\
& p\left(F^{(1)}|X,\boldsymbol{\theta}^{(0)}\right) dF^{(N_\mathrm{h})} \ldots dF^{(1)}
\end{aligned}
$$

- **Very challenging!**

Related Work

- Variational DGP (Damianou and Lawrence, 2013)
- Sequential Inference for DGPs (Wang et al., 2016)
- DGP with Expectation Propagation (Bui et al., 2016)
- Variational Auto-Encoded DGP (Dai et al., 2017)
- Dropout as a Bayesian Approximation (Gal and Gahramani, 2016)
- Structured and Efficient Variational Deep Learning with Matrix Gaussian Posteriors (Louizos and Welling, 2016)

Related Work

- Variational DGP (Damianou and Lawrence, 2013)
- Sequential Inference for DGPs (Wang et al., 2016)
- DGP with Expectation Propagation (Bui et al., 2016)
- Variational Auto-Encoded DGP (Dai et al., 2017)
- Dropout as a Bayesian Approximation (Gal and Gahramani, 2016)
- Structured and Efficient Variational Deep Learning with Matrix Gaussian Posteriors (Louizos and Welling, 2016)

Related Work

- Variational DGP (Damianou and Lawrence, 2013)
- Sequential Inference for DGPs (Wang et al., 2016)
- DGP with Expectation Propagation (Bui et al., 2016)
- Variational Auto-Encoded DGP (Dai et al., 2017)
- Dropout as a Bayesian Approximation (Gal and Gahramani, 2016)
- Structured and Efficient Variational Deep Learning with Matrix Gaussian Posteriors (Louizos and Welling, 2016)

Motivation

Can we develop better DGP models using their connection to DNNs?

1. Faster learning without compromising performance
2. Scalable to very large datasets
3. Extendable to a moderate number of layers

Motivation

Can we develop better DGP models using their connection to DNNs?

1. Faster learning without compromising performance
2. Scalable to very large datasets
3. Extendable to a moderate number of layers

Motivation

Can we develop better DGP models using their connection to DNNs?

1. Faster learning without compromising performance
2. Scalable to very large datasets
3. Extendable to a moderate number of layers

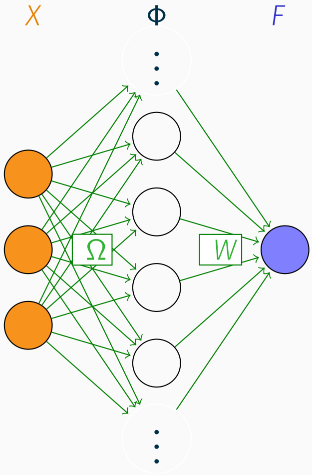
Can we develop better DGP models using their connection to DNNs?

1. Faster learning without compromising performance
2. Scalable to very large datasets
3. Extendable to a moderate number of layers

DGP Architecture

DGPs with Random Features

- GPs are single-layered Neural Nets with an infinite number of hidden units



- Weight-space view of a GP

$$F = \Phi W$$

- The priors over the weights are

$$p\left(W_{\cdot i}\right) = \mathcal{N}\left(\mathbf{0}, \mathrm{I}\right)$$

Random Feature Expansion of Kernels

- The **RBF kernel**
- The first order **Arc-Cosine kernel**

Random Feature Expansion of Kernels

- The **RBF kernel** can be approximated using **trigonometric functions**

$$\Phi_{\text{RBF}} = \sqrt{\frac{\sigma^2}{N_{\mathrm{RF}}}} \left[\cos\left(F\Omega\right), \sin\left(F\Omega\right)\right] \quad \text{with} \quad p\left(\Omega_{\cdot j} \middle| \boldsymbol{\theta}\right) = \mathcal{N}\left(\mathbf{0}, \Lambda^{-1}\right)$$

allowing for scaling factors σ^2 and $\Lambda = \mathrm{diag}(l_1^2, \ldots, l_d^2)$

Random Feature Expansion of Kernels

- The first order **Arc-Cosine kernel** can be approximated using **Rectified Linear Units (ReLU)**

$$\Phi_{\mathrm{ARC}} = \sqrt{\frac{2\sigma^2}{N_{\mathrm{RF}}}} \max\left(\mathbf{0}, F\Omega\right) \qquad \text{with} \qquad p\left(\Omega_{\cdot j} \middle| \boldsymbol{\theta}\right) = \mathcal{N}\left(\mathbf{0}, \Lambda^{-1}\right)$$

Random Feature Expansion of Kernels

- The **RBF kernel** can be approximated using **trigonometric functions**

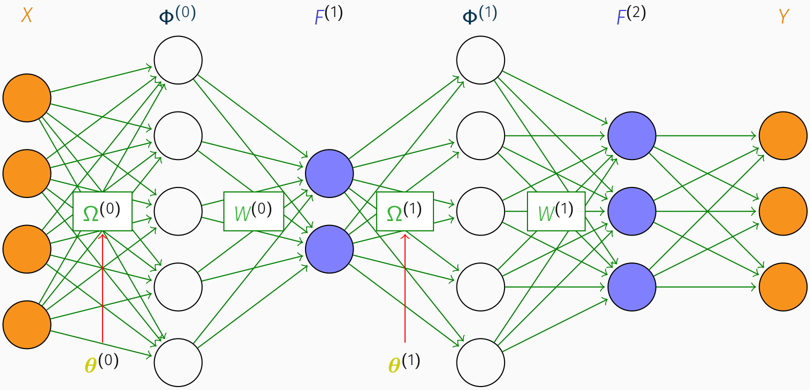
$$\Phi_{\text{RBF}} = \sqrt{\frac{\sigma^2}{N_{\mathrm{RF}}}} \left[\cos\left(F\Omega\right), \sin\left(F\Omega\right)\right] \quad \text{with} \quad p\left(\Omega_{.j} \middle| \boldsymbol{\theta}\right) = \mathcal{N}\left(\mathbf{0}, \Lambda^{-1}\right)$$

allowing for scaling factors σ^2 and $\Lambda = \mathrm{diag}(l_1^2, \ldots, l_d^2)$

- The first order **Arc-Cosine kernel** can be approximated using **Rectified Linear Units (ReLU)**

$$\Phi_{\text{ARC}} = \sqrt{\frac{2\sigma^2}{N_{\mathrm{RF}}}} \max\left(\mathbf{0}, F\Omega\right) \qquad \text{with} \qquad p\left(\Omega_{.j} \middle| \boldsymbol{\theta}\right) = \mathcal{N}\left(\mathbf{0}, \Lambda^{-1}\right)$$

DGPs with RFs become DNNs



Approximating the Marginal Likelihood

DGPs with RFs - Stochastic Variational Inference

- Define $\Psi = \left(\Omega^{(0)}, \dots, \Omega^{(L)}, W^{(0)}, \dots, W^{(L)}\right)$
- Evidence lower bound

$$\log\left[p(Y|\boldsymbol{\theta})\right] \geq \mathbb{E}_{q(\Psi)}\left(\log\left[p\left(Y|\Psi\right)\right]\right) - \mathrm{D_{KL}}\left[q(\Psi)\|p\left(\Psi|\boldsymbol{\theta}\right)\right]$$

where $q(\Psi)$ approximates $p(\Psi|Y, \boldsymbol{\theta})$

- $\mathrm{D_{KL}}$ computable analytically if q and p are Gaussian!

DGPs with RFs - Stochastic variational inference

- Assume factorized likelihood:

$$p(Y|X, \Psi, \theta) = \prod_{\mathrm{k}} p(\mathbf{y}_{\mathrm{k}}|\mathbf{x}_{\mathrm{k}}, \Psi, \theta)$$

- Stochastic **unbiased** estimate of the expectation term
  - **Mini-batch**

$$\mathbb{E}_{q(\Psi)}\left(\log\left[p(Y|X, \Psi, \theta)\right]\right) \approx \frac{n}{m} \sum_{\mathrm{k} \in \mathcal{I}_m} \mathbb{E}_{q(\Psi)}\left(\log[p(y_{\mathrm{k}}|\mathbf{x}_{\mathrm{k}}, \tilde{\Psi}, \theta)]\right)$$

  - **Monte Carlo**

$$\mathbb{E}_{q(\Psi)}\left(\log\left[p(y_{\mathrm{k}}|x_{\mathrm{k}}, \Psi, \theta)\right]\right) \approx \frac{1}{N_{\mathrm{MC}}} \sum_{r=1}^{N_{\mathrm{MC}}} \log[p(y_{\mathrm{k}}|x_{\mathrm{k}}, \tilde{\Psi}_r, \theta)]$$

with $\tilde{\Psi}_r \sim q(\Psi)$

DGPs with RFs - Stochastic variational inference

- Factorized approximate posterior

$$q(\Psi) = \prod_{ijl} q\left(\Omega_{ij}^{(l)}\right) \prod_{ijl} q\left(W_{ij}^{(l)}\right)$$

where

$$q\left(\Omega_{ij}^{(l)}\right) = \mathcal{N}\left(\mu_{ij}^{(l)}, (\sigma^2)_{ij}^{(l)}\right) \quad \text{and} \quad q\left(W_{ij}^{(l)}\right) = \mathcal{N}\left(m_{ij}^{(l)}, (s^2)_{ij}^{(l)}\right)$$

DGPs with RFs - Stochastic variational inference

- Factorized approximate posterior

$$q(\Psi) = \prod_{ijl} q\left(\Omega_{ij}^{(l)}\right) \prod_{ijl} q\left(W_{ij}^{(l)}\right)$$

where

$$q\left(\Omega_{ij}^{(l)}\right) = \mathcal{N}\left(\mu_{ij}^{(l)}, (\sigma^2)_{ij}^{(l)}\right) \quad \text{and} \quad q\left(W_{ij}^{(l)}\right) = \mathcal{N}\left(m_{ij}^{(l)}, (s^2)_{ij}^{(l)}\right)$$

- Reparameterization trick

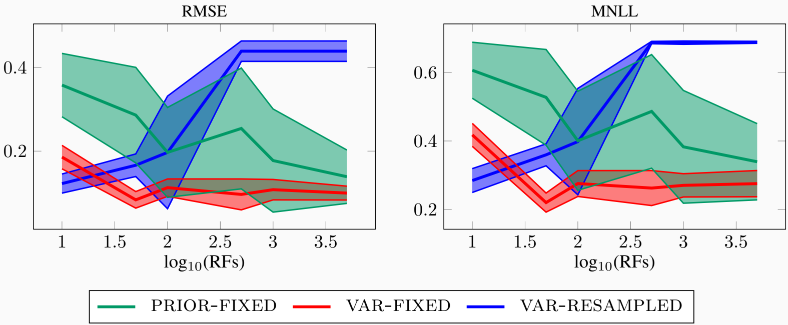
$$\left(\tilde{W}_r^{(l)}\right)_{ij} = \left(s^2\right)_{ij}^{(l)} \epsilon_{rij}^{(l)} + m_{ij}^{(l)}$$

with $\epsilon_{rij}^{(l)} \sim \mathcal{N}(0, 1)$

Optimization Strategies for Ω

- Random features can be **fixed**
  - PRIOR-FIXED

  or treated **variationally**
  - VAR-FIXED (with fixed randomness)
  - VAR-RESAMPLED (resampled at each iteration)

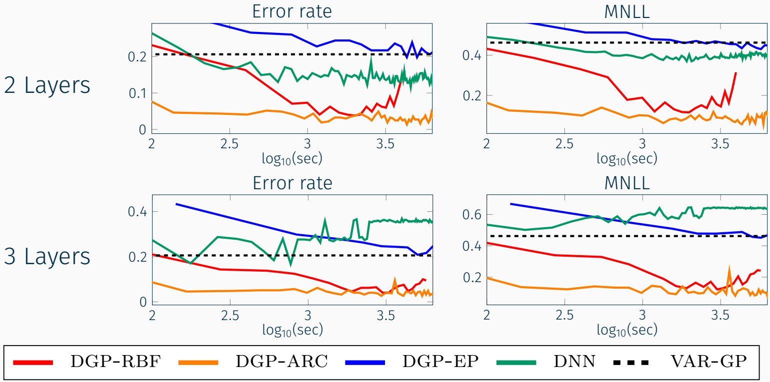


Evaluation

Model Comparison - Binary Classification

EEG Dataset

(n = 14979, d = 14)

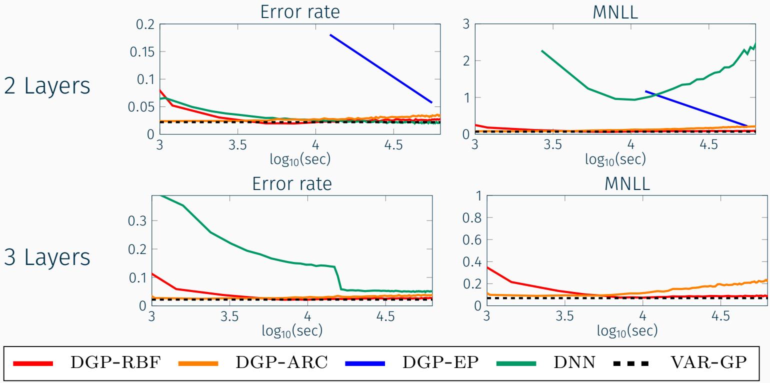


- Consistently outperforms competing techniques both in terms of **speed** and **predictive performance**

Model Comparison - Multiclass Classification

MNIST Dataset

$(n = 60000, d = 784)$



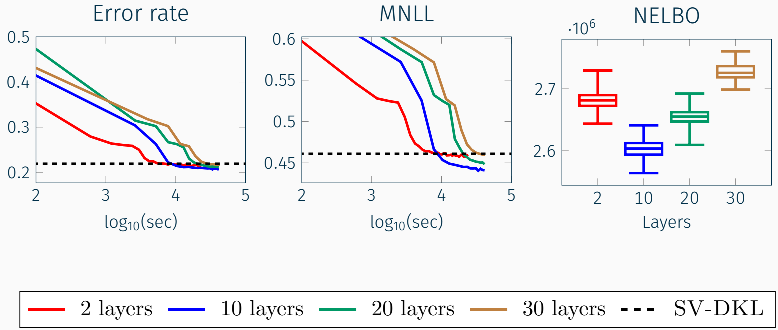
- Model performance remains resilient compared to other techniques

Scalability to Large Datasets

- Over 99% accuracy on variant of MNIST with 8.1 million images!

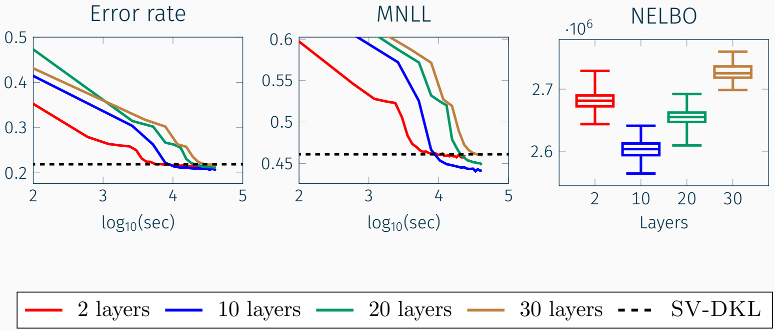
Dataset	Accuracy		MNLL	
	RBF	ARC	RBF	ARC
MNIST8M	99.14%	99.04%	0.0454	0.0465
AIRLINE	78.55%	72.76%	0.4583	0.5335

Performance of Deeper Models - Airline Dataset



- Model converges to an optimal state for every configuration
- NELBO confirmed to be a suitable criteria for model selection
- Includes feed-forward of inputs to each layer

Performance of Deeper Models - Airline Dataset



- Model converges to an optimal state for every configuration
- NELBO confirmed to be a suitable criteria for model selection
- Includes feed-forward of inputs to each layer

Conclusions

Conclusions

- Contributions
  - A complete evaluation of DGPs inspired by DNNs
  - Scalable and practical DGP inference without requiring Cholesky
  - A study of various options for optimizing random features
  - Distributed implementation using Parameter-Server framework

Conclusions

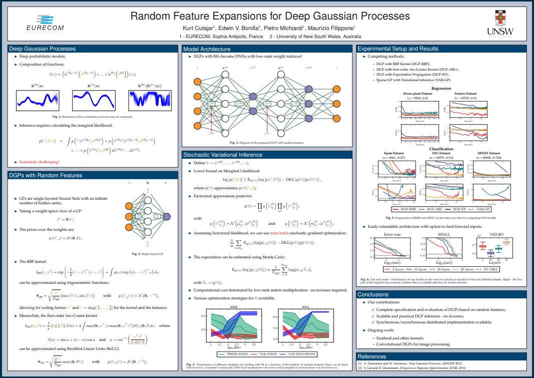
- Contributions
  - A complete evaluation of DGPs inspired by DNNs
  - Scalable and practical DGP inference without requiring Cholesky
  - A study of various options for optimizing random features
  - Distributed implementation using Parameter-Server framework
- Ongoing work
  - Fastfood kernel and orthogonal random features
  - Convolutional GP layers for complex image datasets
  - Hybrid synchronous/asynchronous distributed approach

Code in TensorFlow:

`github.com/mauriziofilippone/deep_gp_random_features`

Code in TensorFlow:

`github.com/mauriziofilippone/deep_gp_random_features`



Poster #126

Thank you!