Imperial College London

Doubly Stochastic Inference for Deep Gaussian Processes

Hugh Salimbeni

Department of Computing
Imperial College London

29/5/2017

Motivation

- DGPs promise much, but are difficult to train

Motivation

- DGPs promise much, but are difficult to train
- Fully factorized VI doesn't work well

Motivation

- DGPs promise much, but are difficult to train
- Fully factorized VI doesn't work well
- We seek a variational approach that works and scales

Motivation

- DGPs promise much, but are difficult to train
- Fully factorized VI doesn't work well
- We seek a variational approach that works and scales

Motivation

- DGPs promise much, but are difficult to train
- Fully factorized VI doesn't work well
- We seek a variational approach that works and scales

Other recently proposed schemes [1, 2, 5] make additional approximations and require more machinery than VI

Talk outline

1. **Summary**: Model ⏩ Inference ⏩ Results
2. **Details:** Model ⏩ Inference ⏩ Results
3. **Questions**

Model

We use the standard DGP model, with one addition:

Model

We use the standard DGP model, with one addition:

- We include a linear (identity) mean function for all the internal layers

Model

We use the standard DGP model, with one addition:

- We include a linear (identity) mean function for all the internal layers

(1D example in [4])

Inference

- We use the model conditioned on the inducing points as a conditional variational posterior

Inference

- We use the model conditioned on the inducing points as a conditional variational posterior
- We impose Gaussians on the inducing points, (independent between layers but full rank within layers)

Inference

- We use the model conditioned on the inducing points as a conditional variational posterior
- We impose Gaussians on the inducing points, (independent between layers but full rank within layers)
- We use sampling to deal with the intractable expectation

Inference

- We use the model conditioned on the inducing points as a conditional variational posterior
- We impose Gaussians on the inducing points, (independent between layers but full rank within layers)
- We use sampling to deal with the intractable expectation

Inference

- We use the model conditioned on the inducing points as a conditional variational posterior
- We impose Gaussians on the inducing points, (independent between layers but full rank within layers)
- We use sampling to deal with the intractable expectation

We never compute $N \times N$ matrices (we make no additional simplifications to variational posterior)

Results

- We show significant improvement over single layer models on large ($\sim 10^6$) and massive ($\sim 10^9$) data

Results

- We show significant improvement over single layer models on large ($\sim 10^6$) and massive ($\sim 10^9$) data
- Big jump in improvement over single layer GP with $5\times$ number of inducing points

Results

- We show significant improvement over single layer models on large ($\sim 10^6$) and massive ($\sim 10^9$) data
- Big jump in improvement over single layer GP with $5\times$ number of inducing points
- On small data we never do worse than the single layer model, and often better

Results

- We show significant improvement over single layer models on large ($\sim 10^6$) and massive ($\sim 10^9$) data
- Big jump in improvement over single layer GP with $5\times$ number of inducing points
- On small data we never do worse than the single layer model, and often better
- We can get 98.1% on mnist with only 100 inducing points

Results

- We show significant improvement over single layer models on large ($\sim 10^6$) and massive ($\sim 10^9$) data
- Big jump in improvement over single layer GP with $5\times$ number of inducing points
- On small data we never do worse than the single layer model, and often better
- We can get 98.1% on mnist with only 100 inducing points
- We surpass all permutation invariant methods on rectangles-images (designed to test deep vs shallow architectures)

Results

- We show significant improvement over single layer models on large ($\sim 10^6$) and massive ($\sim 10^9$) data
- Big jump in improvement over single layer GP with $5\times$ number of inducing points
- On small data we never do worse than the single layer model, and often better
- We can get 98.1% on mnist with only 100 inducing points
- We surpass all permutation invariant methods on rectangles-images (designed to test deep vs shallow architectures)
- Identical model/inference hyperparameters for all our models

Details: The Model

We use the standard DGP model, with a linear mean function for all the internal layers:

- If dimensions agree use the identity, otherwise PCA

Details: The Model

We use the standard DGP model, with a linear mean function for all the internal layers:

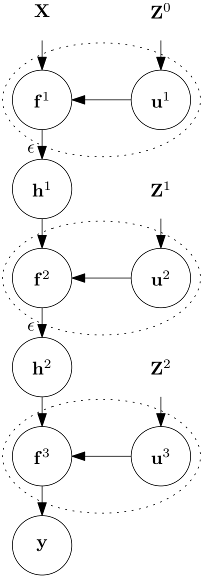
- If dimensions agree use the identity, otherwise PCA
- Sensible alternative: initialize latents to identity (but linear mean function works better)

Details: The Model

We use the standard DGP model, with a linear mean function for all the internal layers:

- If dimensions agree use the identity, otherwise PCA
- Sensible alternative: initialize latents to identity (but linear mean function works better)
- Not so sensible alternative: random. Doesn't work well (posterior is (very) multimodal)

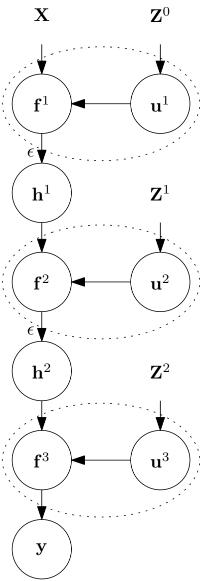
The DGP: Graphical Model

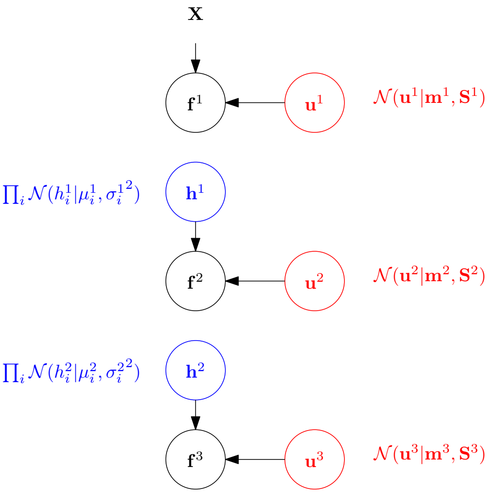


The DGP: Density

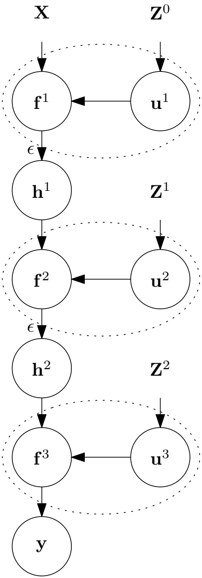
$$p(\mathbf{y}, \{\mathbf{h}^l, \mathbf{f}^l, \mathbf{u}^l\}_{l=1}^L) = \overbrace{\prod_{i=1}^N p(y_i | f_i^L)}^{\text{likelihood}} \times \underbrace{\prod_{l=1}^L p(\mathbf{h}^l | \mathbf{f}^l) p(\mathbf{f}^l | \mathbf{u}^l; \mathbf{h}^{l-1}, \mathbf{Z}^{l-1}) p(\mathbf{u}^l; \mathbf{Z}^{l-1})}_{\text{DGP prior}}$$

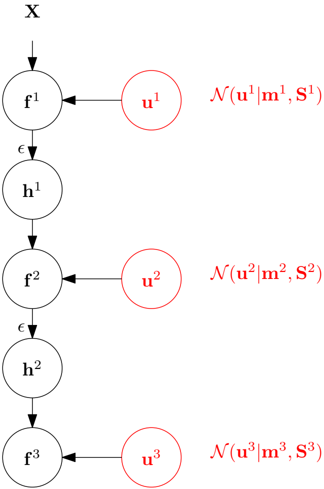
Factorised Variational Posterior





Our Variational Posterior





Recap: ‘GPs for Big Data’ [3]

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})$$

Recap: ‘GPs for Big Data’ [3]

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})$$

Marginalise **u** from the variational posterior:

$$\int p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})d\mathbf{u} = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) =: q(\mathbf{f}|\mathbf{m}, \mathbf{S}; \mathbf{X}, \mathbf{Z}) \tag{1}$$

Recap: 'GPs for Big Data' [3]

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})$$

Marginalise **u** from the variational posterior:

$$\int p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})d\mathbf{u} = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) =: q(\mathbf{f}|\mathbf{m}, \mathbf{S}; \mathbf{X}, \mathbf{Z}) \tag{1}$$

Define the following mean and covariance functions:

$$\mu_{\mathbf{m},\mathbf{Z}}(\mathbf{x}_i) = m(\mathbf{x}_i) + \boldsymbol{\alpha}(\mathbf{x}_i)^T(\mathbf{m} - m(\mathbf{Z}))\,,$$
$$\Sigma_{\mathbf{S},\mathbf{Z}}(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j) - \boldsymbol{\alpha}(\mathbf{x}_i)^T(k(\mathbf{Z}, \mathbf{Z}) - \mathbf{S})\boldsymbol{\alpha}(\mathbf{x}_j)\,.$$

where $\boldsymbol{\alpha}(\mathbf{x}_i) = k(\mathbf{x}_i, \mathbf{Z})k(\mathbf{Z}, \mathbf{Z})^{-1}$

Recap: 'GPs for Big Data' [3]

$$q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})$$

Marginalise $\mathbf{u}$ from the variational posterior:

$$\int p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})d\mathbf{u} = \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \boldsymbol{\Sigma}) =: q(\mathbf{f}|\mathbf{m}, \mathbf{S}; \mathbf{X}, \mathbf{Z}) \quad (1)$$

Define the following mean and covariance functions:

$$\mu_{\mathbf{m},\mathbf{Z}}(\mathbf{x}_i) = m(\mathbf{x}_i) + \boldsymbol{\alpha}(\mathbf{x}_i)^T(\mathbf{m} - m(\mathbf{Z})),$$
$$\Sigma_{\mathbf{S},\mathbf{Z}}(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j) - \boldsymbol{\alpha}(\mathbf{x}_i)^T(k(\mathbf{Z}, \mathbf{Z}) - \mathbf{S})\boldsymbol{\alpha}(\mathbf{x}_j).$$

where $\boldsymbol{\alpha}(\mathbf{x}_i) = k(\mathbf{x}_i, \mathbf{Z})k(\mathbf{Z}, \mathbf{Z})^{-1}$
With these functions $[\boldsymbol{\mu}]_i = \mu_{\mathbf{m},\mathbf{Z}}(\mathbf{x}_i)$ and $[\boldsymbol{\Sigma}]_{ij} = \Sigma_{\mathbf{S},\mathbf{Z}}(\mathbf{x}_i, \mathbf{x}_j)$.

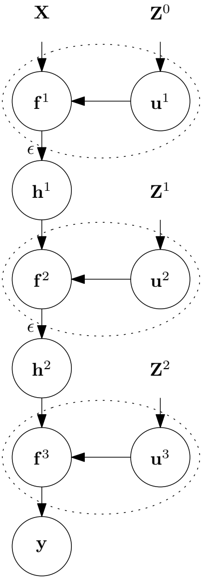
Recap: 'GPs for Big Data' [3] cont.

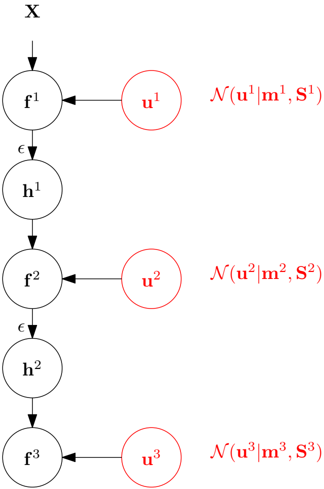
Key idea:

The f_i marginals of $q(\mathbf{f}, \mathbf{u}) = p(\mathbf{f}|\mathbf{u}; \mathbf{X}, \mathbf{Z})\mathcal{N}(\mathbf{u}|\mathbf{m}, \mathbf{S})$ depend only on the inputs $\mathbf{x}_i$

(and the variational parameters)

Our Variational Posterior





Our approach

We can marginalise all the $\mathbf{u}^l$ from our posterior

Our approach

We can marginalise all the $\mathbf{u}^l$ from our posterior
Result for lth layer is $q(\mathbf{f}^l|\mathbf{m}^l, \mathbf{S}^l; \mathbf{h}^{l-1}, \mathbf{Z}^{l-1})$.

Our approach

We can marginalise all the $\mathbf{u}^l$ from our posterior
Result for lth layer is $q(\mathbf{f}^l|\mathbf{m}^l, \mathbf{S}^l; \mathbf{h}^{l-1}, \mathbf{Z}^{l-1})$.
The fully coupled (both *between* and *within* layers) variational posterior is

$$\prod_{l=1}^{L} p(\mathbf{h}^l|\mathbf{f}^l) q(\mathbf{f}^l|\mathbf{m}^l, \mathbf{S}^l; \mathbf{h}^{l-1}, \mathbf{Z}^{l-1})$$

But what about the ith marginals?

Since at each layer the ith marginal depends only on the ith component of the layer below, we have

$$q(\{f_i^l, h_i^l\}_{l=1}^L) = \prod_{l=1}^{L} p(h_i^l | f_i^l) q(f_i^l | \mathbf{m}^l, \mathbf{S}^l; h_i^{l-1}, \mathbf{Z}^{l-1}) \tag{2}$$

The lower bound

Since our variational posterior matches the model everywhere except the inducing points, the bound is:

$$\mathcal{L} = \mathbb{E}_{q(\{f_i^l, h_i^l\}_{l=1}^L)} \log p(y_i | f_i^L) - \sum_{l=1}^{L} KL(q(\mathbf{u}^l) || p(\mathbf{u}^l))$$

The analytic marginalisation of the all the inner layers $q(f_i^L)$ is intractable, but we can draw samples ancestrally

Sampling from the variational posterior

- Each layer is Gaussian, given the layer below

Whole sampling process is differentiable wrt variational parameters

Sampling from the variational posterior

- Each layer is Gaussian, given the layer below
- We draw samples using unit Gaussians $\epsilon \sim \mathcal{N}(0, 1)$ at each layer

Whole sampling process is differentiable wrt variational parameters

Sampling from the variational posterior

- Each layer is Gaussian, given the layer below
- We draw samples using unit Gaussians $\epsilon \sim \mathcal{N}(0, 1)$ at each layer
- For $\hat{f}_u^l$, mean and var from $q(f_i^l|\mathbf{m}^l, \mathbf{S}^l; \hat{h}_i^{l-1}, \mathbf{Z}^{l-1})$

$$\hat{f}_i^l = \mu_{\mathbf{m}^l,\mathbf{Z}^{l-1}}(\hat{h}_i^{l-1}) + \epsilon\sqrt{\Sigma_{\mathbf{S}^l,\mathbf{Z}^{l-1}}(\hat{h}_i^{l-1}, \hat{h}_i^{l-1})}$$

where for the first layer $\hat{h}_i^0 = \mathbf{x}_i$

Whole sampling process is differentiable wrt variational parameters

Sampling from the variational posterior

- Each layer is Gaussian, given the layer below
- We draw samples using unit Gaussians $\epsilon \sim \mathcal{N}(0, 1)$ at each layer
- For $\hat{f}_u^l$, mean and var from $q(f_i^l | \mathbf{m}^l, \mathbf{S}^l; \hat{h}_i^{l-1}, \mathbf{Z}^{l-1})$

$$\hat{f}_i^l = \mu_{\mathbf{m}^l, \mathbf{Z}^{l-1}}(\hat{h}_i^{l-1}) + \epsilon \sqrt{\Sigma_{\mathbf{S}^l, \mathbf{Z}^{l-1}}(\hat{h}_i^{l-1}, \hat{h}_i^{l-1})}$$

  where for the first layer $\hat{h}_i^0 = \mathbf{x}_i$
- Just add noise for the $\hat{h}_i^l$

Whole sampling process is differentiable wrt variational parameters

The second source of stochasticity

- We sample the bound in minibatches
- Linear scaling in N
- Can be used when only steaming is possible ($>$ 50GB datasets)

Inference recap

- We use the full model as a variational posterior, conditioned on the inducing points

Inference recap

- We use the full model as a variational posterior, conditioned on the inducing points
- We use Gaussians for the inducing points

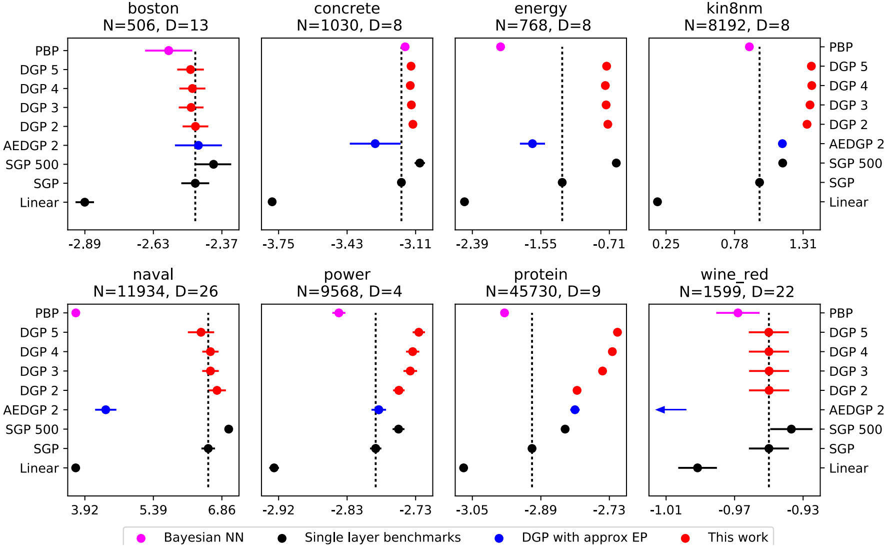
Inference recap

- We use the full model as a variational posterior, conditioned on the inducing points
- We use Gaussians for the inducing points
- The lower bound requires only the posterior marginals

Inference recap

- We use the full model as a variational posterior, conditioned on the inducing points
- We use Gaussians for the inducing points
- The lower bound requires only the posterior marginals
- We can take samples from the posterior marginals using a Monte Carlo estimate

Results (1): UCI



Code Demo

`https://github.com/ICL-SML/Doubly-Stochastic-DGP/blob/master/demos/demo_regression_UCI.ipynb`

Results (2): Large and Massive Data

Test RMSE

	N	D	SGP	SGP 500	DGP 2	DGP 3	DGP 4	DGP 5
year	463810	90	10.67	9.89	9.58	8.98	8.93	**8.87**
airline	700K	8	25.6	25.1	24.6	24.3	24.2	**24.1**
taxi	1B	9	337.5	330.7	281.4	270.4	268.0	**266.4**

Thanks for listening

Questions?

References

[1] T. D. Bui, D. Hernández-Lobato, Y. Li, J. M. Hernández-Lobato, and R. E. Turner. Deep Gaussian Processes for Regression using Approximate Expectation Propagation. *Icml*, 2016.

[2] K. Cutajar, E. V. Bonilla, P. Michiardi, and M. Filippone. Practical Learning of Deep Gaussian Processes via Random Fourier Features. *arXiv preprint arXiv:1610.04386*, 2016.

[3] J. Hensman, N. Fusi, and N. D. Lawrence. Gaussian Processes for Big Data. *Uncertainty in Artificial Intelligence*, pages 282–290, 2013.

[4] M. Lázaro-Gredilla. Bayesian Warped Gaussian Processes. *Advances in Neural Information Processing Systems*, pages 1619–1627, 2012.

[5] Y. Wang, M. Brubaker, B. Chaib-Draa, and R. Urtasun. Sequential Inference for Deep Gaussian Process. *Artificial Intelligence and Statistics*, 2016.