COMP9418 – Advanced Topics in Statistical Machine Learning

W11 – Variational Learning of GP Models

Instructor: Edwin V. Bonilla

School of Computer Science and Engineering

October 11th, 2017

(Last Update: 11/10/17 10:08 am)

Acknowledgements

- [Bonilla et al, 2016] Generic Inference in Latent Gaussian Process Models
  - https://arxiv.org/abs/1609.00577

- [Krauth et al, 2017] AutoGP: Exploring the Capabilities and Limitations of Gaussian Process Models
  - http://auai.org/uai2017/proceedings/papers/50.pdf

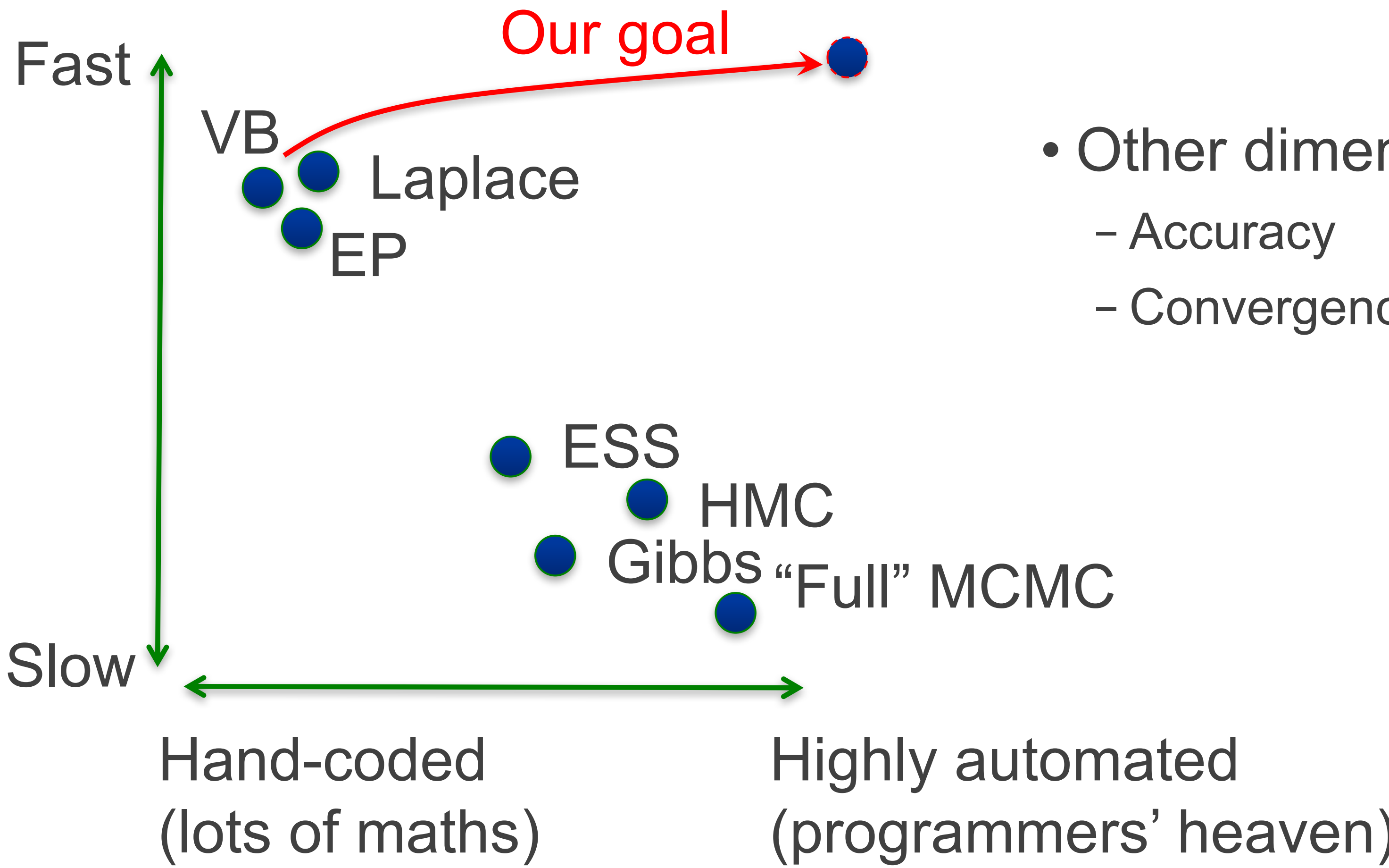
Aims

This lecture will allow you to understand general latent Gaussian process models (LGPMs) and apply variational techniques for inference in these models. Following it, you should be able to:

- Understand the main assumptions of LGPMs for modelling data with Gaussian process priors and general (possibly multivariate) likelihoods.
- Carry out variational inference via inducing variable approaches for scalable posterior inference in LGPMs.
- Understand and apply the reparameterization trick for estimating the gradients of the expected log likelihood in the variational objective of LGPMs.

Approximate Bayesian Inference

Automation vs. Efficiency



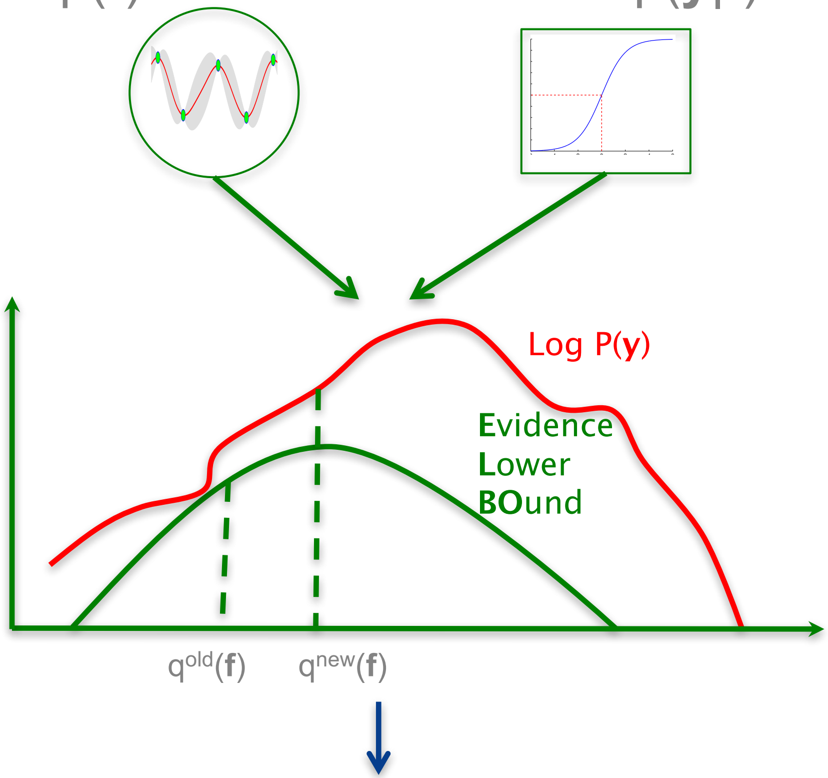
- Other dimensions
  - Accuracy
  - Convergence

We want to build generic yet practical inference tools for practitioners and researchers

Automated Variational Inference (Nguyen & Bonilla, NIPS 2014)

GP prior
p(**f**)

'black-box'
Likelihood p(**y**|**f**)



Approximate posterior:
Mixture of Gaussians q(**f**)

- ELBO = - KL + ELL
  - KL divergence
    - Analytical lower bound
    - Exact gradients
  - Expected log Likelihood (ELL)
    - Expectations over univariate Gaussians
    - No explicit gradients needed
- Practical framework
  - Efficient parameterization
  - As good as hand-coded solutions
  - Orders of magnitude faster than MCMC

Outline

I. Latent Gaussian Process Models (LGPMs) [Bonilla et al, 2016, Sec 1 – 3]

II. Variational Inference Revisited

III. Automated Variational Inference

IV. Examples [Bonilla et al, 2016, Sec 9]

V. Scalability through Inducing Variables and stochastic variational inference (SVI) [Bonilla et al, 2016, Sec 4 – 8]

VI. Improved SVI for LGPMs [Krauth et al, 2017, Sec 4]

I. Latent Gaussian Process Models (LGPMs)

[Bonilla et al, 2016, Sec 1 – 3]

Latent Gaussian Process Models (LGPMs)

- Supervised Learning Problems
  - Inputs: $\mathbf{x} = \{\mathbf{x}_n\}_{n=1}^{N}$ Labels: $\mathbf{y} = \{\mathbf{y}_n\}_{n=1}^{N}$

Factorization of GP prior over Q latent functions

$$f_j \sim \mathcal{GP}(0, \kappa_j(\cdot,\cdot)) \rightarrow p(\mathbf{f}|\boldsymbol{\theta}_0) = \prod_{j=1}^{Q} p(\mathbf{f}_{\bullet j}|\boldsymbol{\theta}_0) = \prod_{j=1}^{Q} \mathcal{N}(\mathbf{f}_{\bullet j}; \mathbf{0}, \mathbf{K}_j)$$

Covariance function of jth GP

All NxQ latent function values

Covariance Hyper-parameters

All N latent values for function j

Covariance matrix induced by κ_j

Factorization of conditional likelihood

$$p(\mathbf{y}|\mathbf{f}, \boldsymbol{\theta}_1) = \prod_{n=1}^{N} p(\mathbf{y}_n|\mathbf{f}_{n\bullet}, \boldsymbol{\theta}_1)$$

Cond. Likelihood parameters

Observations and latent functions for data-point n

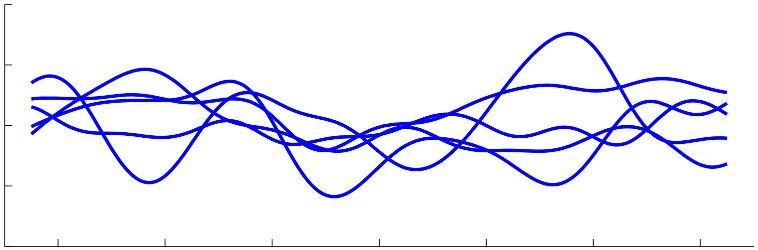
What can we model with this framework?

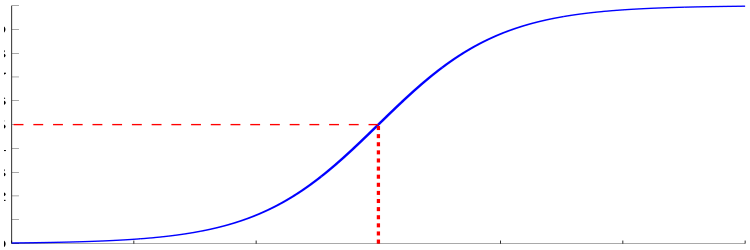
Latent Gaussian Process Models

Examples

Multi-class classification

- Q classes → Q independent GP priors $p(f_j)$, $j = 1, \dots Q$
  - Each GP can have a different covariance
- Softmax likelihood
  - $p(y=j) \propto \exp(f_j)$





Other settings

- Multi-output regression
- Warped GPs
- Log Gaussian Cox process
- Others
  - Access to 'black-box' likelihood

II. Variational Inference Revisited

The Variational Objective

$\mathbf{X}$: Observed variables

$\mathbf{Z}$: Hidden or missing variables

- Goal: given prior P(**Z**) and conditional likelihood p(**X**|**Z**) → approximate the posterior p(**Z**|**X**) with q(**Z**|**X**)
  - Omitting θ as we can include them in **Z** as random variables
- We have seen that

Approximate posterior

True posterior

$$\log p(\mathbf{X}) = \mathrm{KL}(q(\mathbf{Z}|\mathbf{X}) \| p(\mathbf{Z}|\mathbf{X})) + \mathcal{L}_{\text{lower}}(q)$$

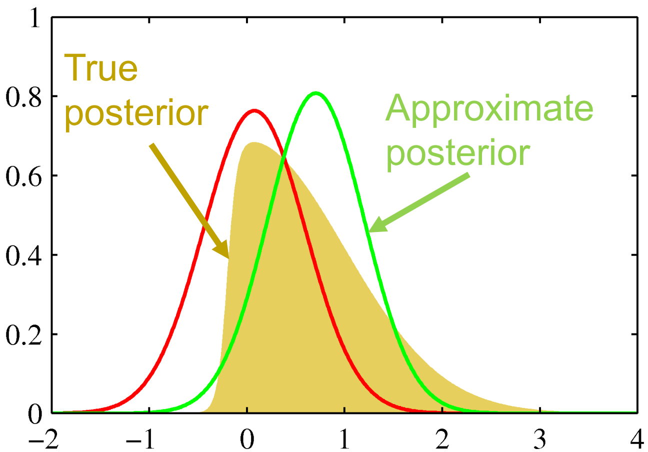
- Where: $\mathcal{L}_{\text{lower}}(q) \stackrel{\text{def}}{=} \mathbb{E}_{q(\mathbf{Z}|\mathbf{X})}\left[\log \frac{p(\mathbf{X}, \mathbf{Z})}{q(\mathbf{Z}|\mathbf{X})}\right]$

This is our variational objective (functional)

  - We will attempt to maximise $\mathcal{L}_{\text{lower}}(q)$ wrt q(**Z**|**X**)
  - This is indeed equivalent to minimizing KL(q(**Z**|**X**) || p(**Z**|**X**))

What exactly is q(Z|X)?

- Free-form q(**Z**|**X**): optimisation of the functional $\mathcal{L}_{\text{lower}}(q)$
  - Would give us the right answer as the KL vanishes at the true posterior
  - However, need to solve normalization, which was our initial problem!
- Fixed-form q(**Z**|**X**): Consider a restricted family of distributions
  - Minimize the objective wrt members of this family
  - E.g. Use factorised distribution
  - E.g. Use a parametrized distribution q(**Z**|**X**,λ)
    - » Optimisation via standard calculus



What family of distributions?

- As flexible as possible
- Tractability is the main constraint
- No risk of overfitting
  - The more flexible the better the approximation to the true posterior

Understanding the Variational Objective

The lower bound $\mathcal{L}_{\text{lower}}(q) \stackrel{\text{def}}{=} \mathbb{E}_{q(\mathbf{Z}|\mathbf{X})}\left[\log \frac{p(\mathbf{X},\mathbf{Z})}{q(\mathbf{Z}|\mathbf{X})}\right]$ can be written as:

$$\mathcal{L}_{\text{lower}}(q) = \underbrace{\mathbb{E}_{q(\mathbf{Z}|\mathbf{X})}[\log p(\mathbf{X}|\mathbf{Z})]}_{\text{Expected log likelihood (ELL)}} - \underbrace{\mathrm{KL}(q(\mathbf{Z}|\mathbf{X})\|p(\mathbf{Z}))}_{\text{KL (approx. posterior || prior)}}$$

- ELL term is a model fit: How well the (samples from the) posterior explains the observations
- KL is a penalty term: Keep posterior close to prior beliefs
- Also known as
  - Variational free energy
  - Evidence lower bound (ELBO)

III. Automated Variational Inference

AUTOMATED VARIATIONAL INFERENCE

THE GENERAL FRAMEWORK

Goal: *Approximate 'intractable' posterior p(**f** | **y**)*

- Find the closest tractable approximation q(**f**)

$$q(\mathbf{f}|\boldsymbol{\lambda}) = \sum_{k=1}^{K} \pi_k q_k(\mathbf{f}|\boldsymbol{\lambda}_k)$$

- Minimize KL[q(**f**) || p(**f**|**y**)] → Maximize ELBO:

$$\mathcal{L} = \underbrace{\mathbb{E}_q[-\log q(\mathbf{f}|\boldsymbol{\lambda})] + \mathbb{E}_q[\log p(\mathbf{f})]}_{-\mathrm{KL}[q(\mathbf{f}|\boldsymbol{\lambda})||p(\mathbf{f})]} + \underbrace{\sum_{k=1}^{K} \pi_k \mathbb{E}_{q_k}[\log p(\mathbf{y}|\mathbf{f})]}_{\mathrm{ELL}}$$

Irrespective of the likelihood models (black-box):

- KL can be lower bounded using Jensen's inequality
  - Exact gradients of the GP hyper-parameters can be obtained
- ELL and its gradients can be approximated *efficiently*

Expected Log Likelihood (ELL) Term

Theorem 1

The ELL and its gradients can be estimated using expectations over **univariate** Gaussian distributions.

$$q_{k(n)} = q_{k(n)}(\mathbf{f}_{n\bullet}|\boldsymbol{\lambda}_{k(n)})$$

$$\mathbb{E}_{q_k}[\log p(\mathbf{y}|\mathbf{f})] = \sum_{n=1}^{N} \mathbb{E}_{q_{k(n)}}[\log p(\mathbf{y}_n|\mathbf{f}_{n\bullet})]$$

$$\nabla_{\boldsymbol{\lambda}_{k(n)}} \mathbb{E}_{q_{k(n)}}[\log p(\mathbf{y}_n|\mathbf{f}_{n\bullet})] = \mathbb{E}_{q_{k(n)}} \nabla_{\boldsymbol{\lambda}_{k(n)}} \log q_{k(n)}(\mathbf{f}_{n\bullet}|\boldsymbol{\lambda}_{k(n)}) \log p(\mathbf{y}_n|\mathbf{f}_{n\bullet})$$

Practical consequences

- We can use Monte Carlo estimates
- Gradients of the likelihood are not required
  - Only likelihood evaluations are needed
- Also holds for Q > 1

PRACTICAL VARIATIONAL DISTRIBUTIONS

Two distribution classes of interest

- **FG:** Full Gaussian, i.e. K=1, full covariance matrix
- **MoDG:** Mixture of diagonal Gaussians

Theorem 2

The covariance matrices can be parameterized **linearly** in the number of observations

- Optimization is made easier (less parameters and correlations)

Theorem 3

Gradients estimates of MoDG have **lower variance** than FG's

- Optimization with MoDG converges faster

IV. Examples

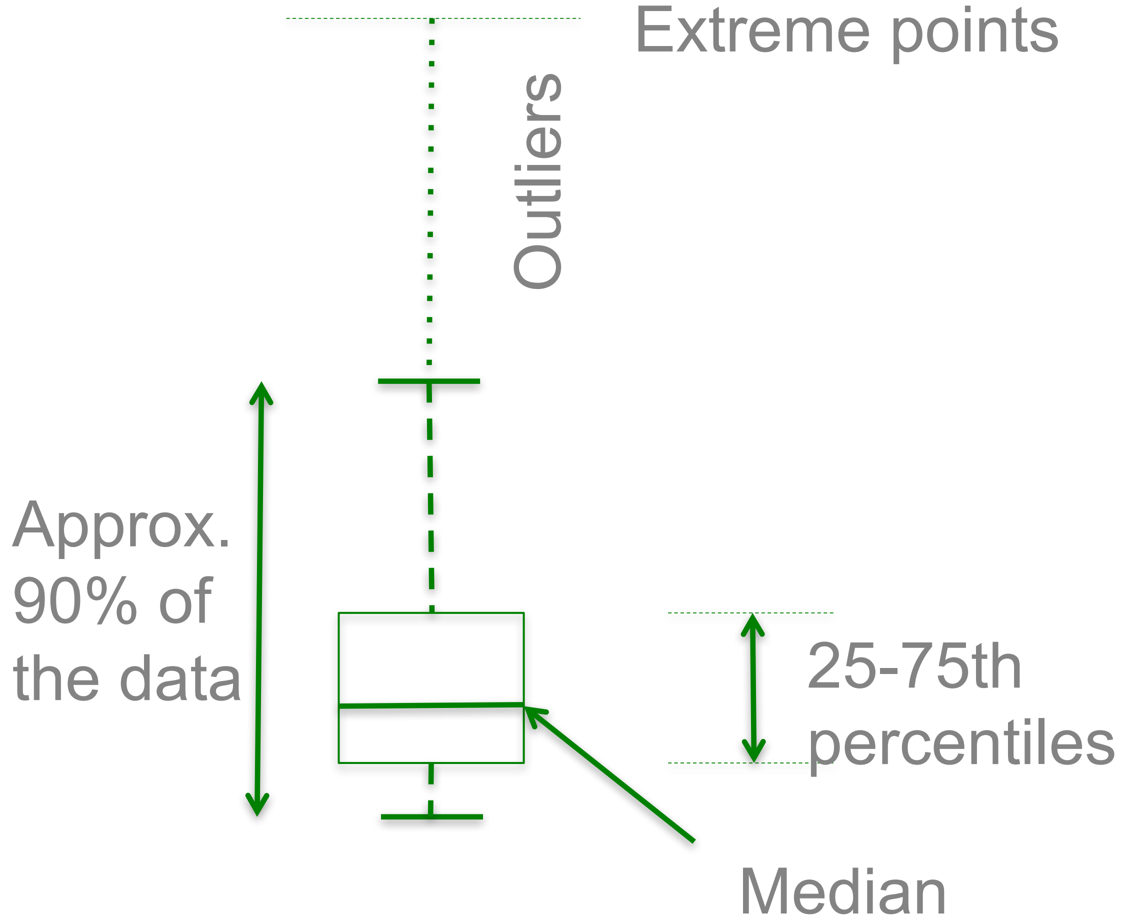
[Bonilla et al, 2016, Sec 9]

PREAMBLE

Performance measures

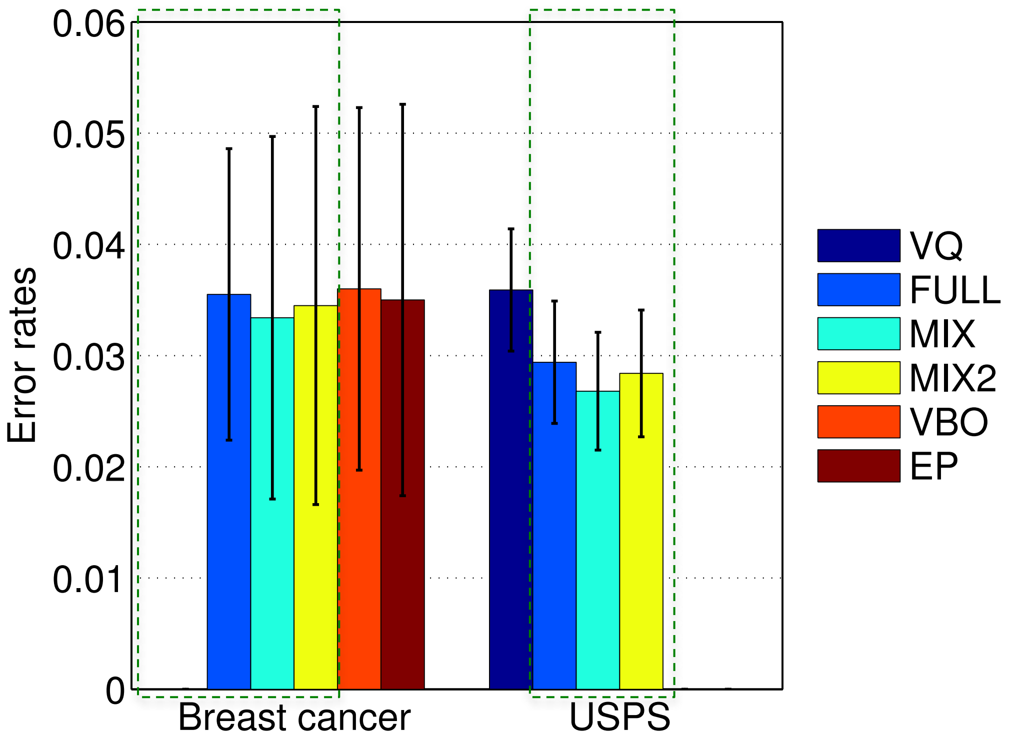
- SSE
  - Standardised square error
- NLPD
  - Negative log predictive density

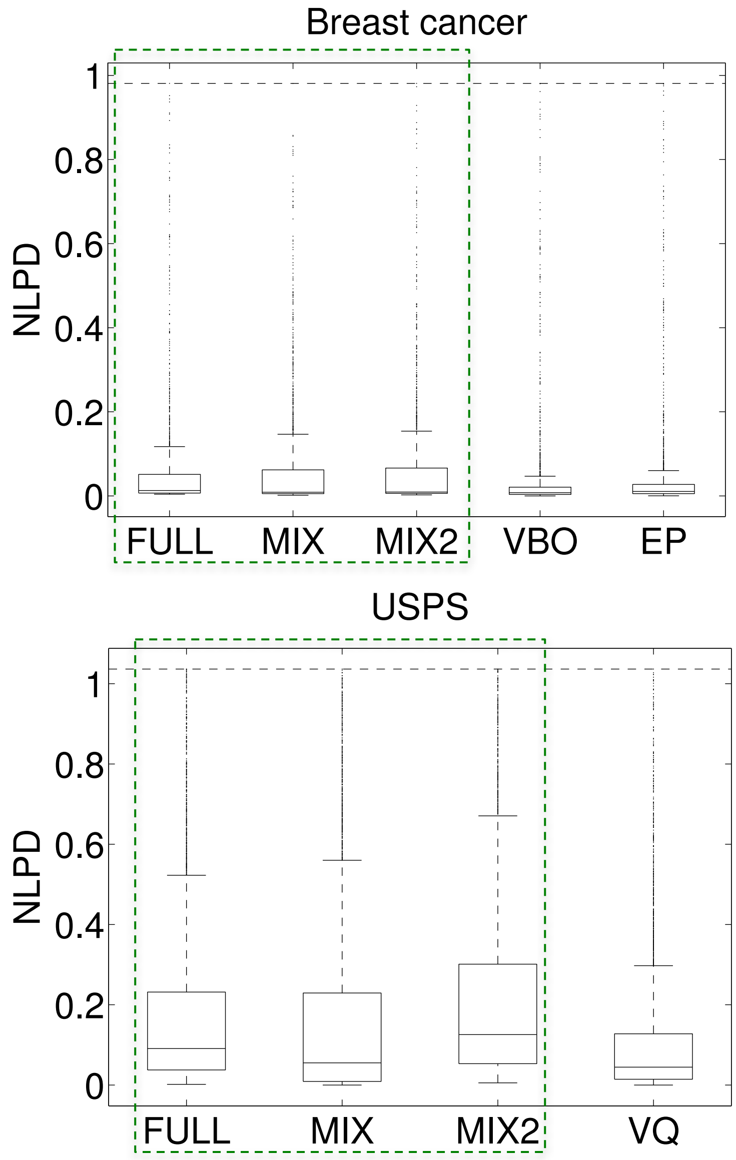
Box-and-whisker plots



Binary and Multi-class Classification

Sigmoid and softmax likelihoods



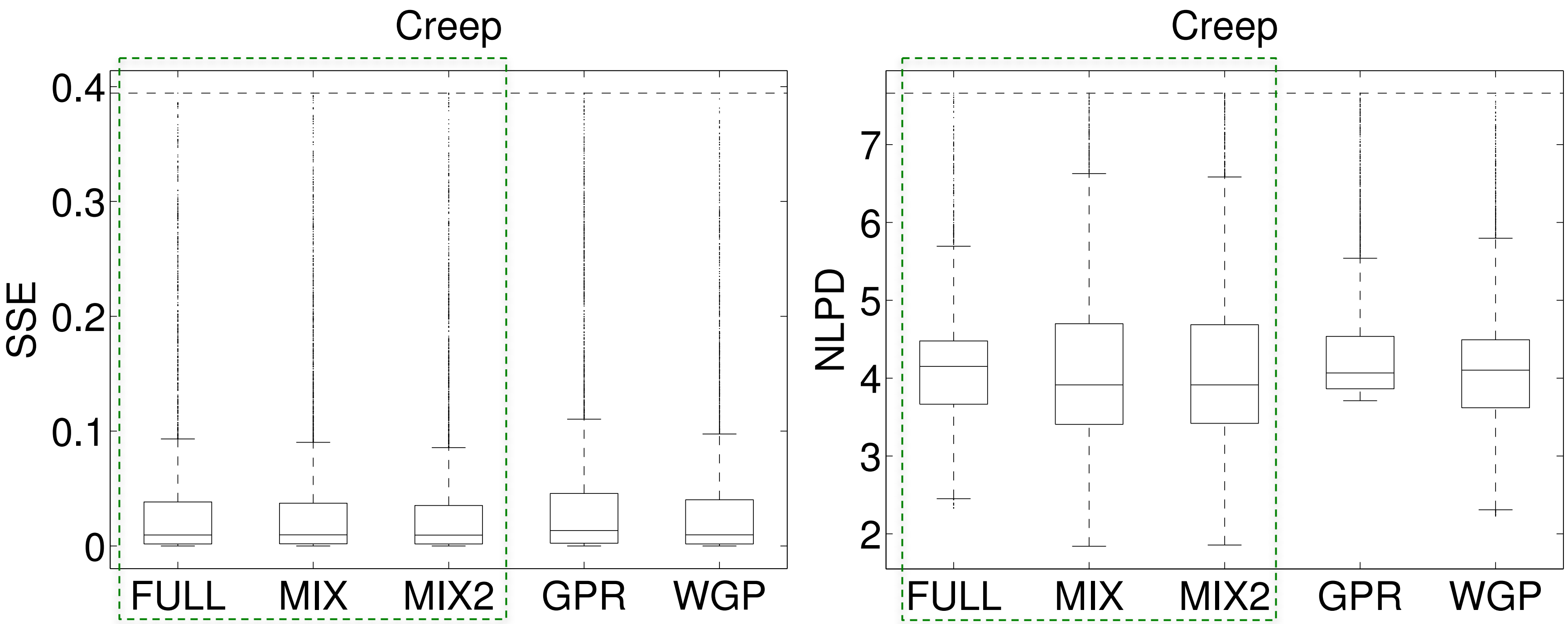


Comparable performance to hard-coded methods

Warped Gaussian Processes

Likelihood: $p(y|f) = \nabla_y t(y) \mathcal{N}(t(y); f, \sigma^2)$

t(y): Non-linear monotonic transformation



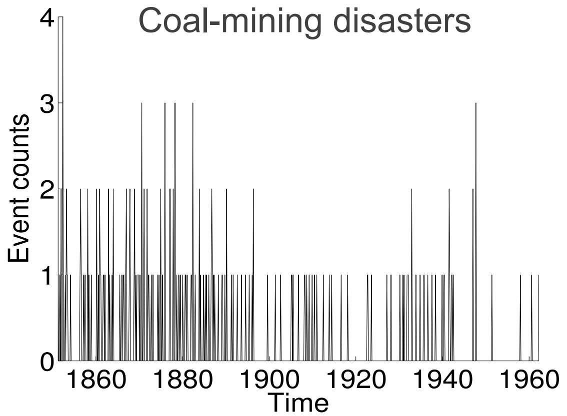
Comparable performance to exact method WGP

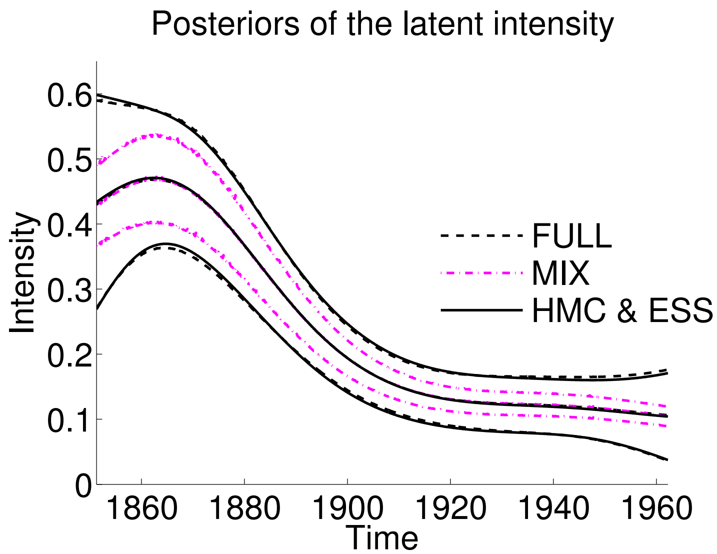
Log Gaussian Cox Process

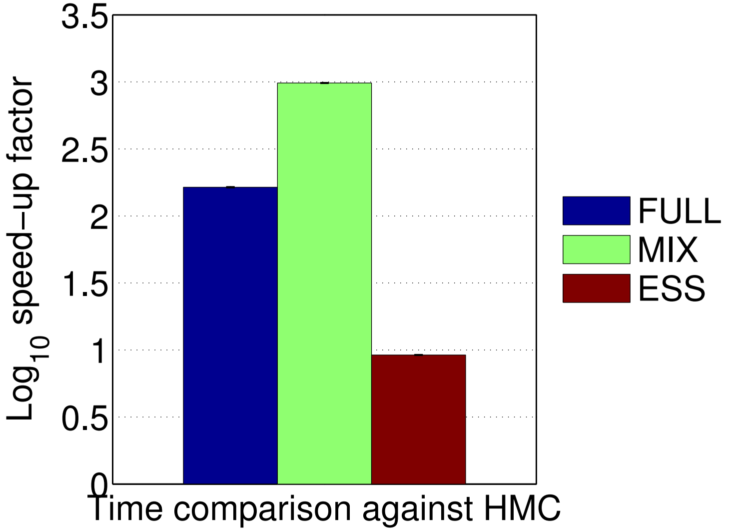
Likelihood:

$$p(y_n|f_n) = \frac{\lambda_n^{y_n} \exp(-\lambda_n)}{y_n!}$$

where $\lambda_n = \exp(f_n + m)$





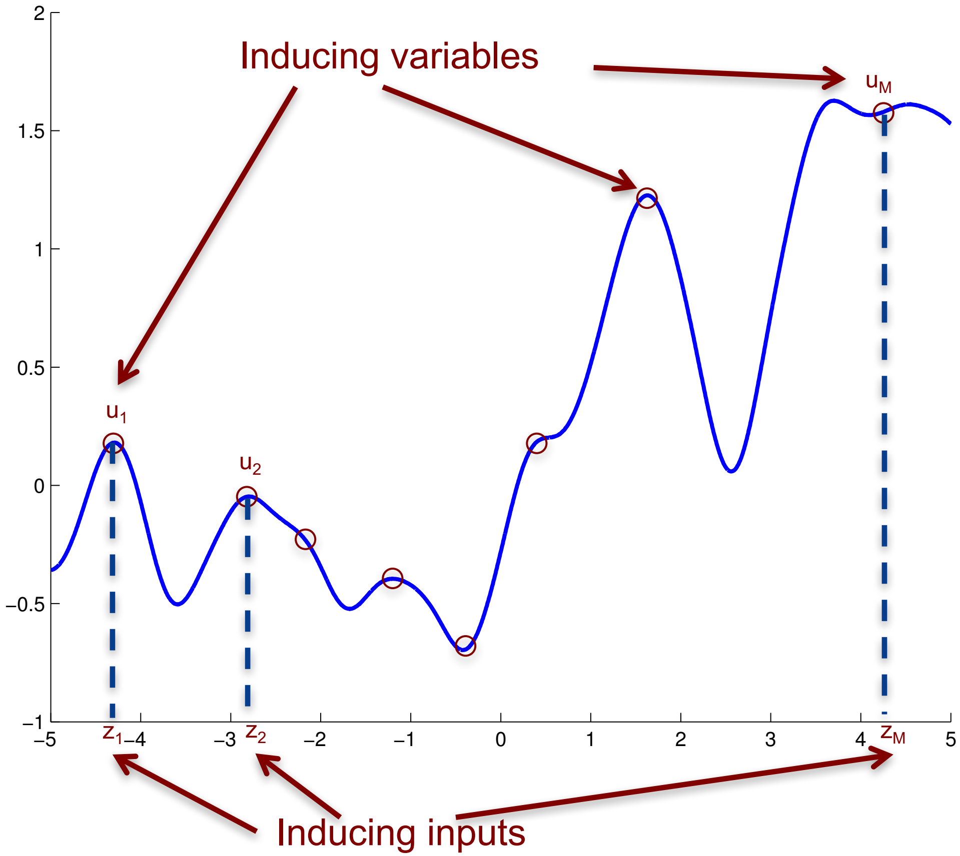


Same performance as sampling, orders of magnitude faster

V. Scalability through Inducing Variables and Stochastic Variational Inference (SVI)

[Bonilla et al, 2016, Sec 4 – 8]

What Are the Inducing Points?



Inducing variables u

- Latent values of the GP (as **f** and $\mathbf{f}_*$)
- Usually marginalized

Inducing inputs **z**

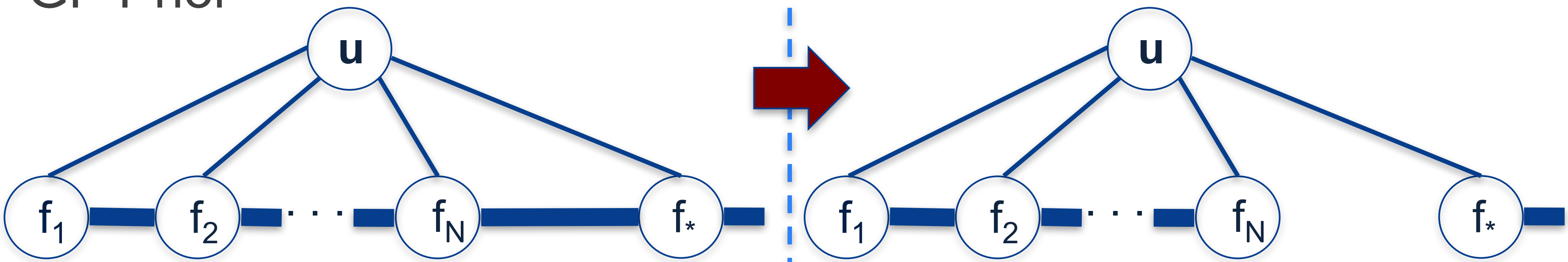
- Corresponding input locations (as **x**)
- Imprint on final solution

- Generalization of “support points”, “active set”, “pseudo-inputs”
  - ‘Good’ summary statistics → *induce* statistical dependencies
  - Can be a subset of the training set
  - Can be arbitrary inducing variables

A Unifying Framework for GP Approximations

(Quiñonero-Candela & Rassmussen, 2005)

GP Prior



Full GP (no approximations). All latent functions are fully connected

Training and test values are **conditionally independent** given **u**

- The joint prior is modified through the inducing variables:

$$p(\mathbf{f}_*, \mathbf{f}) \approx q(\mathbf{f}_*, \mathbf{f}) \stackrel{\text{def}}{=} \int q(\mathbf{f}_* | \mathbf{u}) q(\mathbf{f} | \mathbf{u}) p(\mathbf{u}) \, \mathrm{d}\mathbf{u}$$

Test conditional

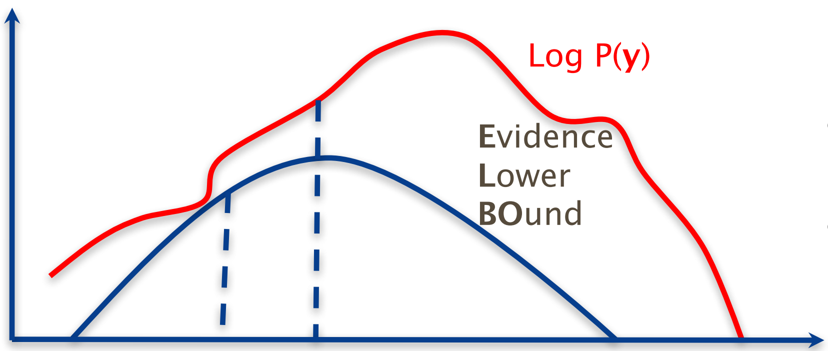
Training conditional

Exact from GP prior with $\mathbf{K}_{\mathbf{uu}}$

- Most (previously proposed) approx. methods:
  - Different specifications of these conditionals
  - Different **Z**: Subset of training/test inputs, new **z** inputs

VFE: Variational Free Energy Optimization

(Titsias, 2009)



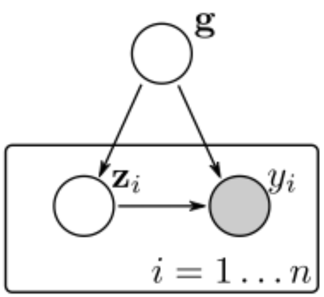
Inducing-point model

- Do not modify the (prior) model
- Approximate posterior over inducing variables

- ELBO: Single consistent objective function
  - Inducing variables are 'marginalized' variationally
  - *Inducing inputs are additional variational parameters*
  - Joint learning of posterior and variational parameters
  - Additional regularization term appears naturally
- Predictive distribution in regression case
  - Equivalent to PP
  - $O(M^2N)$ → Good enough?

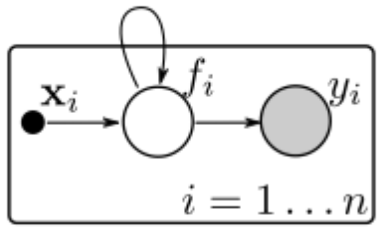
Stochastic Variational Inference (SVI)

SVI for 'big data'



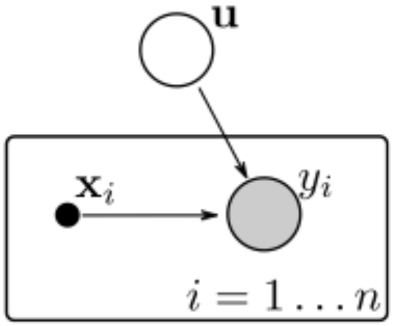
Decomposition across data-points through global variables

GPs



Fully coupled by definition

Large scale GPs



Inducing variables can be such global variables

Maintain an explicit representation of inducing variables in lower bound (cf. Titsias)

- Lower bound decomposes across inputs
- Use stochastic optimization
- Cost O(M^3) in time → Can scale to very large datasets!

VI. Improved SVI for LGPMs

[Krauth et al, 2017, Sec 4]

Stochastic Variational Inference for Latent Gaussian Process Models

Augmented model: Augment prior with M inducing variables $\{\mathbf{u}_{\bullet j}\}$ to approximate 'intractable' posterior $p(\mathbf{f}|\mathbf{y})$

- Find the closest approximation $p(\mathbf{u}|\mathbf{y}) \approx q(\mathbf{u}|\boldsymbol{\lambda})$ such that:

$$q(\mathbf{u}|\boldsymbol{\lambda}) = \sum_{k=1}^{K} \pi_k q_k(\mathbf{u}|\mathbf{m}_k, \mathbf{S}_k) = \sum_{k=1}^{K} \pi_k \prod_{j=1}^{Q} \mathcal{N}(\mathbf{u}_{\bullet j}; \mathbf{m}_{kj}, \mathbf{S}_{kj}),$$

- Minimize $\mathrm{KL}[q(\mathbf{u}|\boldsymbol{\lambda})||p(\mathbf{u}|\mathbf{y})] \rightarrow$ maximize ELBO:

$$\mathcal{L}_{\text{elbo}} = -\mathrm{KL}[q(\mathbf{u}|\boldsymbol{\lambda})||p(\mathbf{u})] + \underbrace{\sum_{n=1}^{N}\sum_{k=1}^{K} \pi_k \mathbb{E}q_{k(n)}(\mathbf{f}_{n\bullet}|\boldsymbol{\lambda}_k) \log p(\mathbf{y}_n|\mathbf{f}_{n\bullet})}_{\text{ELL}}$$

Efficient estimates via samples from univariate Gaussians and reparametrization trick.

The Reparameterization Trick in a Nutshell

- Need to have low-variance gradient estimates
- Before we said we can use:

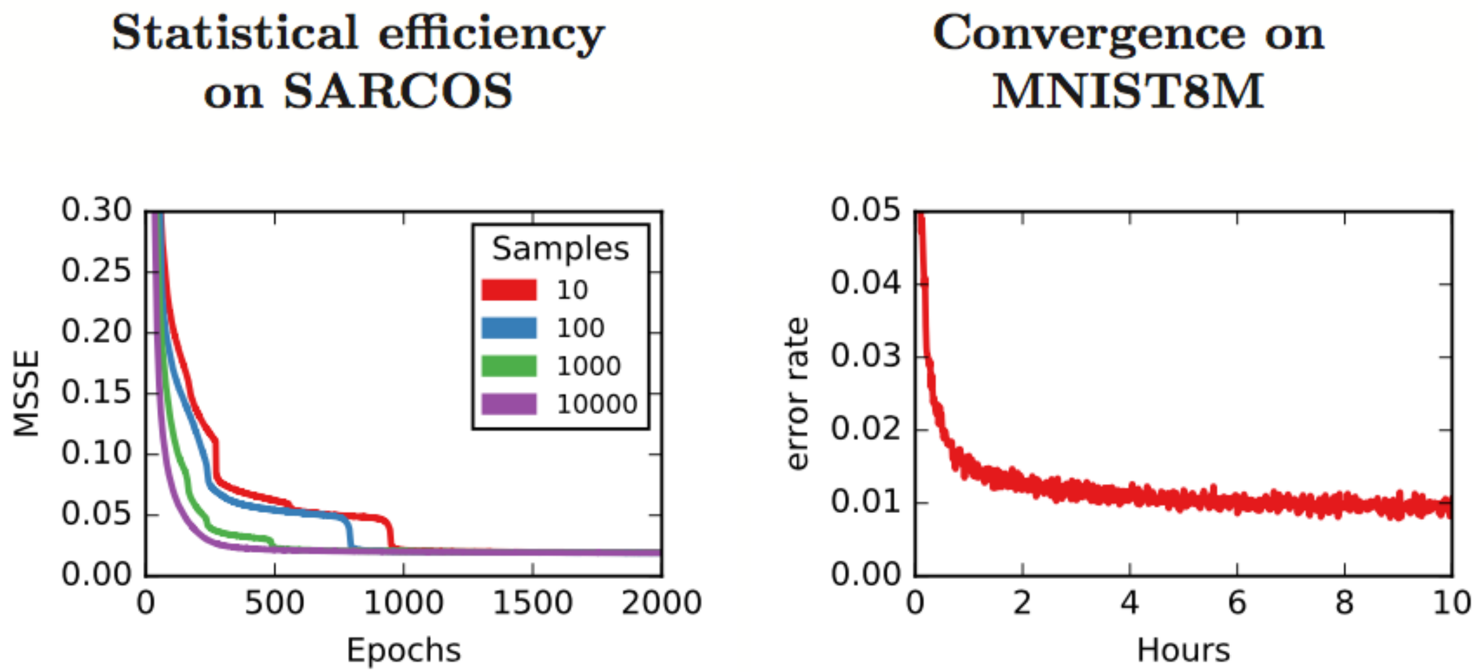
$$\nabla_{\boldsymbol{\lambda}} \mathbb{E}_{q(\mathbf{f}|\boldsymbol{\lambda})}[\log p(\mathbf{y}|\mathbf{f})] = \mathbb{E}_{q(\boldsymbol{\lambda})}[\nabla_{\boldsymbol{\lambda}} \log q(\mathbf{f}|\boldsymbol{\lambda}) \log p(\mathbf{y}|\mathbf{f})]$$

- If we have access to the likelihood implementation, we can use an MC estimate instead:

$$\epsilon_{knj} \sim \mathcal{N}(0,1),$$
$$f_{nj}^{(k,i)} = b_{knj} + \sigma_{knj}\epsilon_{knj}, \quad j = 1, \ldots, Q,$$
$$\widehat{\mathcal{L}}_{\text{ell}}^{(n)} = \frac{1}{S}\sum_{k=1}^{K}\pi_k \sum_{i=1}^{S} \log p(\mathbf{y}_n|\mathbf{f}_{n\cdot}^{(k,i)}),$$

  – And exploit automatic differentiation frameworks

The Reparametrization Trick in Action



Stability enhanced thanks to the reparameterization trick

Conclusions

- Main assumption of LGPMs:
  - GPs over distinct latent functions uncorrelated in the prior
  - Observations conditionally independent given the latent functions
- Applications in multi-class classification, multi-output regression, modelling count data and more
- Generic inference via optimisation of the variational objective (ELBO)
  - KL term bounded using Jensen's inequality
  - ELL term estimated using MC
- Scalability via inducing-variable approaches
- Low-variance gradient estimates using the reparameterization trick

Reading

- [Bonilla et al, 2016] Generic Inference in Latent Gaussian Process Models
  - https://arxiv.org/abs/1609.00577
- [Krauth et al, 2017] AutoGP: Exploring the Capabilities and Limitations of Gaussian Process Models
  - http://auai.org/uai2017/proceedings/papers/50.pdf
- (Recommended but not required) [Kingma and Welling, 2014] Auto-encoding variational Bayes
  - https://arxiv.org/pdf/1312.6114