MCMC for Variationally Sparse Gaussian Processes

James Hensman[1], Alexander G. de G. Matthews[2], Maurizio Filippone[3], Zoubin Ghahramani[2]

1 - Lancaster University, UK. email: james.hensman@lancaster.ac.uk
2 - University of Cambridge, UK. email: am554@cam.ac.uk, zoubin@eng.cam.ac.uk
3 - EURECOM, Sophia Antipolis, France. email: maurizio.filippone@eurecom.fr

Motivation

GP models are elegant Bayesian nonparametric models. They come with three challenges:

- $\mathcal{O}(N^3)$ complexity
- Inference of covariance function parameters
- Intractable function values (for non-Gaussian $p(\mathbf{y}\,|\,\mathbf{f})$)

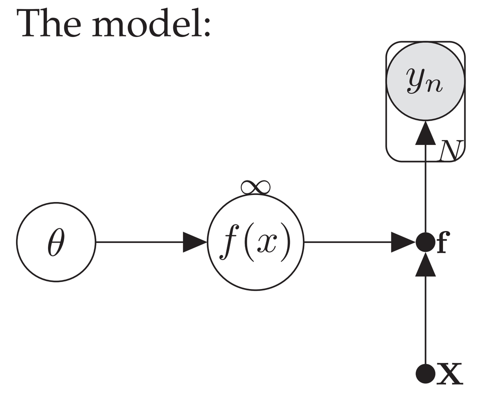
In this work, we combine a variational approximation (for $\mathcal{O}(NM^2)$ complexity) with MCMC (for function values and parameters), to give an approximation that is efficient and flexible.

Reference	$p(\mathbf{y}\,\vert\,\mathbf{f})$	Sparse	Posterior	Hyperparam.
Williams & Barber[1] [also 2, 3]	probit/logit	✗	Gaussian (assumed)	point estimate
Titsias [4]	Gaussian	✓	Gaussian (optimal)	point estimate
Chai [5]	softmax	✓	Gaussian (assumed)	point estimate
Nguyen and Bonilla [6]	any factorized	✗	Mixture of Gaussians	point estimate
Hensman et al. [7]	probit	✓	Gaussian (assumed)	point estimate
This work	any factorized	✓	free-form	free-form

Key idea

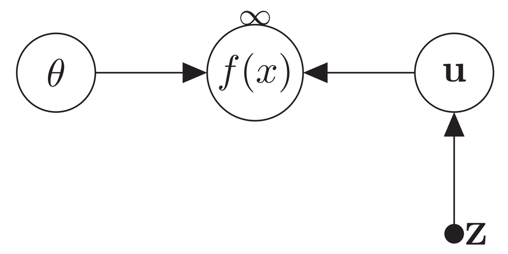
- **Inducing point representation**
  Only compute the value of the GP function at a reduced set of points $\mathbf{Z}$, not necessarily at the data points.

  The model:



$$\theta \sim p(\theta)$$
$$f(x) \sim \mathcal{GP}(0, k(x, x'; \theta))$$
$$\mathbf{f} = [f(x_1), f(x_2) \ldots f(x_n)]^\top$$
$$y_n \sim p(y_n\,|\,f(x_n))$$

  The approximation:



$$f(x) \sim \mathcal{GP}\big(k(x, \mathbf{Z})\mathbf{K}_{uu}^{-1}\mathbf{u}, k(x, x') - k(x, \mathbf{Z})\mathbf{K}_{uu}^{-1}k(\mathbf{Z}, x')\big)$$
$$\theta, \mathbf{u} \sim q(\theta, \mathbf{u})$$

  effectively,

$$\mathbf{u} = [f(z_1), f(z_2) \ldots f(z_M)]^\top$$

- **Minimize KL between Q-process and P-process [see also 8]**
  Informal argument: the the points on the function be $\mathcal{F} = \{\mathbf{f}, \mathbf{u}, \mathbf{f}^\star\}$, with $\mathbf{f} \cap \mathbf{u} = \mathbf{f} \cap \mathbf{f}^\star = \mathbf{f}^\star \cap \mathbf{u} = \varnothing$.

  The joint distribution in the P-process can be written

$$p(\mathbf{f}^\star, \mathbf{f}, \mathbf{u}) = p(\mathbf{f}^\star\,|\,\mathbf{f}, \mathbf{u})p(\mathbf{f}\,|\,\mathbf{u})p(\mathbf{u})$$

  The joint distribution in the Q-process can be written

$$q(\mathbf{f}^\star, \mathbf{f}, \mathbf{u}) = p(\mathbf{f}^\star\,|\,\mathbf{f}, \mathbf{u})p(\mathbf{f}\,|\,\mathbf{u})q(\mathbf{u})$$

  Where the p-terms appear in the q-distribution because of the form we've chosen above.

  Since the distributions contain matching terms, they cancel inside the KL-divergence.

  Caveat: we need to deal with the infinite nature of $\mathbf{f}^\star$.

- **Optimal $q^\star(\mathbf{u}, \theta)$ available, but intractable**
  We show that the optimal variational distribution for $q(\mathbf{u}, \theta)$

$$\log q^\star(\mathbf{u}, \theta) = \mathbf{E}_{p(\mathbf{f}\,|\,\mathbf{u}, \theta)}[\log p(\mathbf{y}\,|\,\mathbf{f})] + \log p(\mathbf{u}\,|\,\theta) + \log p(\theta) + \text{const.}$$

- **Sample $q^\star(\mathbf{u}, \theta)$**
  We can evaluate $q^\star(\mathbf{u}, \theta)$ in $\mathcal{O}(NM^2)$ computations. This is easier than a 'full' GP with $\mathcal{O}(N^3)$ computations, and the dimensionality of the problem is reduced.

Tricks

- **Quadrature for the likelihood**
  Since the likelihoods factorize, compute the variational integral using 1D Gauss-Hermite quadrature.

$$\mathbb{E}_{q(\mathbf{f}\,|\,\mathbf{u})}[\log p(\mathbf{y}\,|\,\mathbf{f})] = \sum_n \mathbb{E}_{q(f_n\,|\,\mathbf{u})}[\log p(y_n|f_n)] \approx \sum_n \sum_i w_i \log p(y_n|f_n^{(i)})$$

- **Whiten/center**
  To improve the mixing, decorrelate the prior term $p(\mathbf{u}|\theta)$ as

$$\mathbf{v} \sim \mathcal{N}(0, \mathbf{I})$$
$$\mathbf{u} = \mathbf{L}\mathbf{v}, \;\text{ with } \mathbf{L}\mathbf{L}^\top = \mathbf{K}_{uu}$$

  The target density is now

$$\log q^\star(\mathbf{v}, \theta) = \mathbb{E}_{p(\mathbf{f}\,|\,(\mathbf{u}=\mathbf{L}\mathbf{v}), \theta)}[\log p(\mathbf{y}\,|\,\mathbf{f})] + \log p(\mathbf{v}) + \log p(\theta) + \text{const.}$$

  ($\mathbf{v}$ and θ are decoupled.)

- **Cholesky Backpropagation**
  In order to jointly sample the function representation $\mathbf{v}$ with the covariance function parameters θ, we use the chain rule:

$$\frac{\partial E}{\partial \theta} = \frac{\partial E}{\partial \mathbf{L}} \frac{\partial \mathbf{L}}{\partial \mathbf{K}} \frac{\partial \mathbf{K}}{\partial \theta}$$

  The middle term is tricky: need to conditional on the particular square root: see Smith [9]. Costs $\mathcal{O}(N^3)$, but worth it (see below right). Now available in GPy/Autograd, theano/TensorFlow in the works.

- **Fit a Gaussian approximation to init the sampler**
  Initialize the sampler with a draw from a Gaussian approximation [7] (and fit $\mathbf{Z}$ positions, see below)

- **Autotune the HMC using Bayesian optimization.**
  We implemented a simple autotuning scheme using BO based on Wang et al. [10].

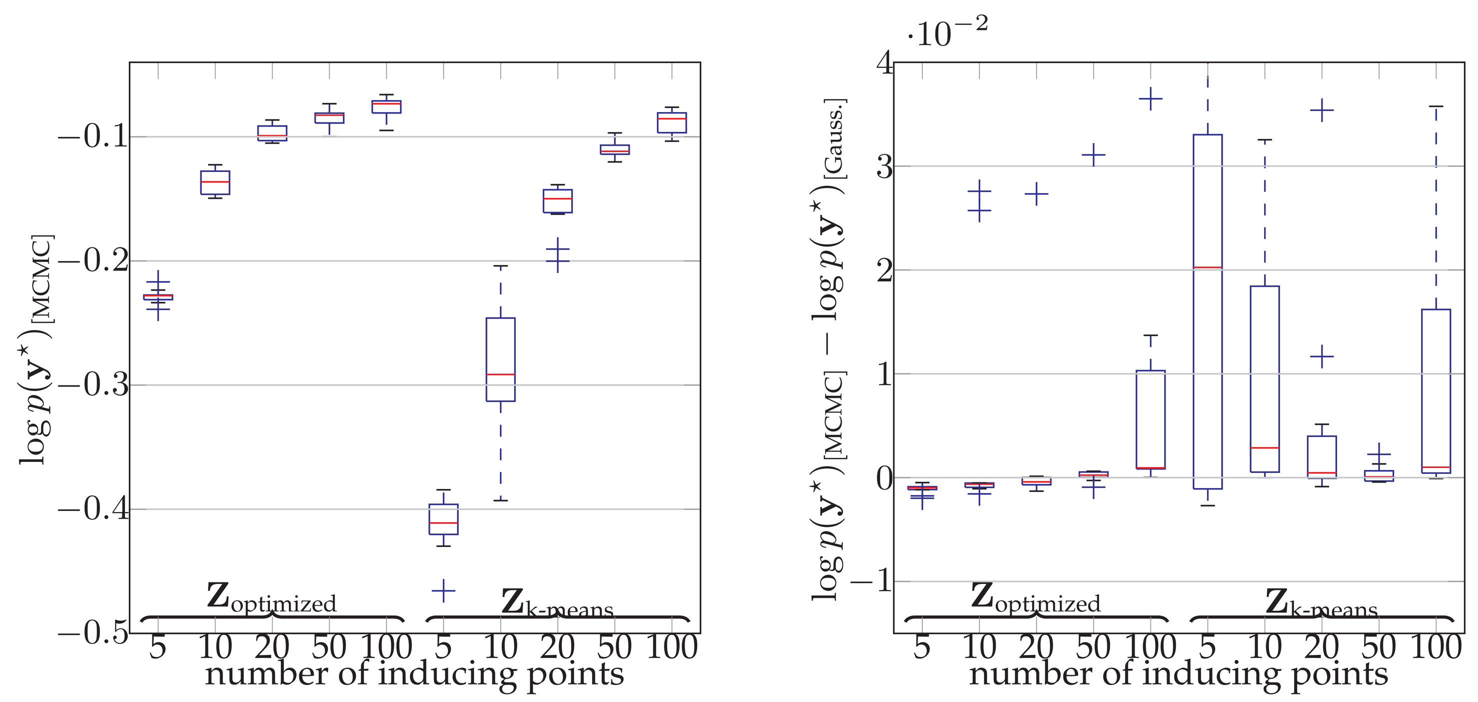
Inducing point positions

Could we be Bayesian about the inducing point positions Z?

- Short answer: no.
- Longer answer: what would the prior be? If we're free to choose any prior, the optimal one turns out to be $q(\mathbf{Z})$. In turn, this is optimal when it becomes a Dirac's delta.

We've not tried optimizing $\mathbf{Z}$ along with the sampling scheme: we have tried using Z that are optimal for a Gaussian approximation.

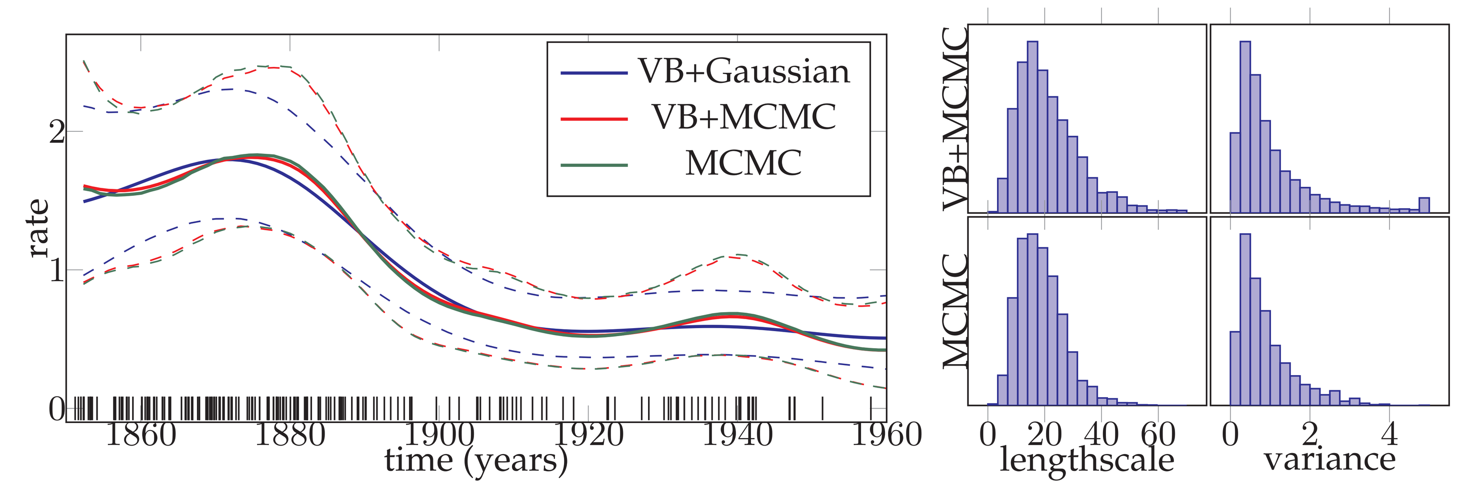
Illustration: Binary classification



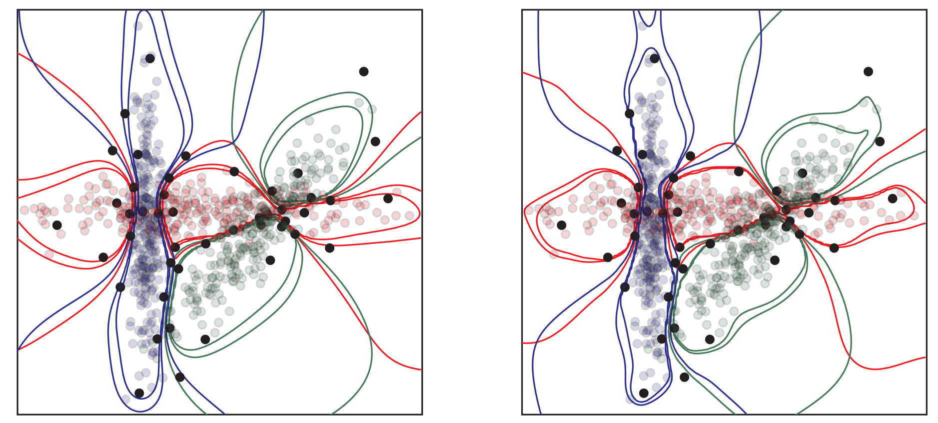
Using the image dataset, left: investigating the effect of increasing the number of inducing points (and optimizing them). Right: the benefits of the method over a Gaussian approximation.

Experiments

- **Log Gaussian Cox processes**

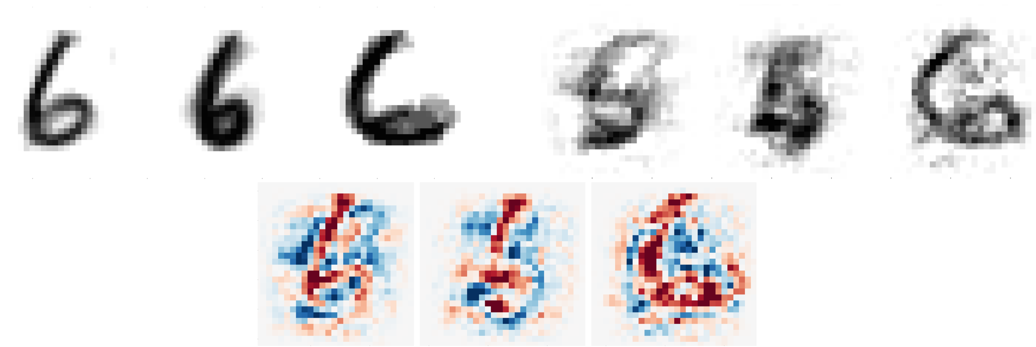


- **Multiclass classification**



Left: Gaussian VB. Right: VBMCMC

- **MNIST** Accuracy: 98.04 %



Top: initial/final inducing point positions. Below: difference.

Sampling Efficiency

Our method: jointly sample $\mathbf{v}, \theta$ with HMC. Extra $\mathcal{O}(M^3)$ operation to backprop the Cholesky. Alternative (Gibbs) method: sample alternately $\mathbf{v}, \theta$: using HMC for $\mathbf{v}$, MH for θ

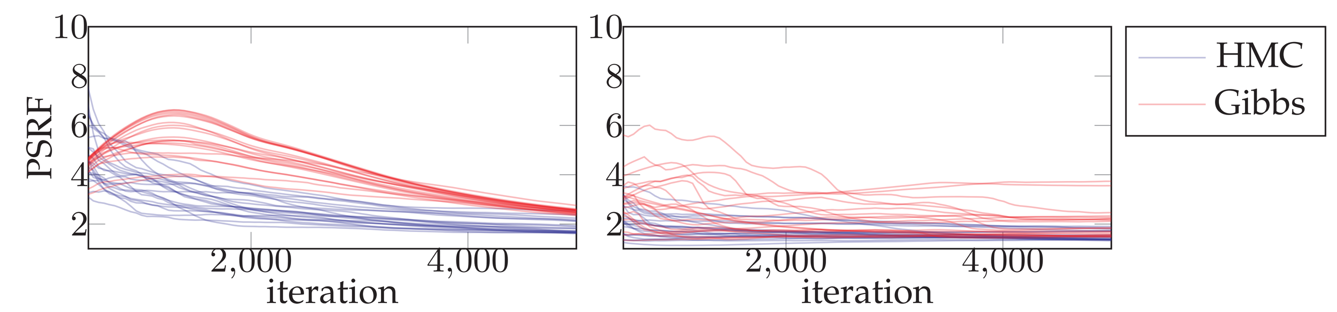


Image dataset - Evolution of the PSRF of the twenty least efficient parameter traces for our method (blue) and Gibbs (red). Left: RBF; right: RBF with ARD.

References

[1] C. K. I. Williams and D. Barber. Bayesian classification with Gaussian processes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(12): 1342–1351, 1998.
[2] M. Opper and C. Archambeau. The variational Gaussian approximation revisited. *Neural comp.*, 21(3):786–792, 2009.
[3] E. Khan, S. Mohamed, and K. P. Murphy. Fast Bayesian inference for non-conjugate Gaussian process regression. In *NIPS*, pages 3140–3148, 2012.
[4] M. K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *AISTATS*, pages 567–574, 2009.
[5] K. M. A. Chai. Variational multinomial logit Gaussian process. *JMLR*, 13(1):1745–1808, June 2012.
[6] T. V. Nguyen and E. V. Bonilla. Automated variational inference for Gaussian process models. In *NIPS*, pages 1404–1412, 2014.
[7] J. Hensman, A. Matthews, and Z. Ghahramani. Scalable variational Gaussian process classification. In *AISTATS*, pages 351–360, 2014.
[8] A. G. D. G. Matthews, J. Hensman, R. E. Turner, and Z. Ghahramani. On sparse variational methods and the KL divergence between stochastic processes. *arXiv preprint 1504.07027*, 2015.
[9] S. P. Smith. Differentiation of the cholesky algorithm. *J. Comp. Graph. Stat.*, 4(2):134–147, 1995.
[10] Z. Wang, S. Mohamed, and N. De Freitas. Adaptive Hamiltonian and Riemann manifold Monte Carlo. In *ICML*, volume 28, pages 1462–1470, 2013.