Variational Bayesian Monte Carlo

(with an Exploration of Mean and Covariance Functions)

Luigi Acerbi

Email: luigi.acerbi@gmail.com Twitter: @AcerbiLuigi

Université de Genève

Motivation

Goal: Bayesian inference with expensive black-box statistical models

Models in science and machine learning

- Likelihood: $p(\mathcal{D}|\boldsymbol{x})$ (data $\mathcal{D}$, parameters $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^D$)
- No detailed information (e.g., no gradient)
- Moderately costly evaluation ($\gtrsim 1$ s)
  - Typical budget up to 500-1000 func. evals.

Bayesian inference

- Posterior: $p(\boldsymbol{x}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{x})p(\boldsymbol{x})}{p(\mathcal{D})}$ (in usable form)
- Marginal likelihood: $p(\mathcal{D}) = \int p(\mathcal{D}|\boldsymbol{x})p(\boldsymbol{x})d\boldsymbol{x}$

Why Bayesian inference?

- Uncertainty and trade-offs between parameters
- $p(\mathcal{D})$ as principled metric of model selection
- Potential machine learning application: *AutoML*

Problem: Existing methods for (approximate) Bayesian inference (e.g., MCMC, ADVI) require many likelihood evals. or knowledge of the model

Key Ideas

Variational inference (VI)

- Approximate $p(\boldsymbol{x}|\mathcal{D})$ with $q_{\boldsymbol{\phi}}(\boldsymbol{x})$
- Minimize $\text{KL}\left[q_{\boldsymbol{\phi}}(\boldsymbol{x})||p(\boldsymbol{x}|\mathcal{D})\right] = \mathbb{E}_{q_{\boldsymbol{\phi}}}\left[\log \frac{q_{\boldsymbol{\phi}}(\boldsymbol{x})}{p(\boldsymbol{x}|\mathcal{D})}\right]$

$$\underset{\text{maximize}}{\Longrightarrow} \quad \text{ELBO}(\boldsymbol{\phi}) = \underbrace{\mathbb{E}_{q_{\boldsymbol{\phi}}}\left[\log p(\mathcal{D}|\boldsymbol{x})p(\boldsymbol{x})\right]}_{\text{expected log joint}} + \underbrace{\mathcal{H}[q_{\boldsymbol{\phi}}(\boldsymbol{x})]}_{\text{entropy}}$$

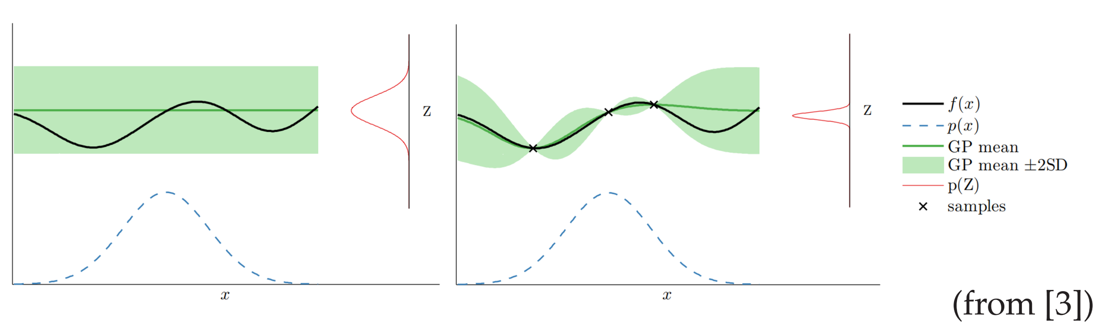
- VI casts inference into optimization + integration
- Obtains $q_{\boldsymbol{\phi}}(\boldsymbol{x})$ and $\text{ELBO}(\boldsymbol{\phi}) \leq \log p(\mathcal{D})$

Bayesian quadrature (BQ)

- Evaluate integral involving (expensive) fcn. f
- Approximate f with Gaussian process (GP)

$$Z = \int \underbrace{p(\boldsymbol{x})}_{\text{Gaussian}} \underbrace{f(\boldsymbol{x})}_{\text{GP}} d\boldsymbol{x}$$

- For some GPs, posterior $p(Z)$ is analytical
- Past work applied BQ to compute $p(\mathcal{D})$ [2,3,4]



(from [3])

VI + BQ $\Rightarrow$ VBMC

Algorithmic Details

Gaussian process representation

- Sample GP hyperparameters (later optimize)
- Squared exponential covariance, Gaussian noise
- Mean fcn.: *negative quadratic* (NQ, default)
  - Also: constant (CN), squared exponential (SE)

Variational posterior

$$q_{\boldsymbol{\phi}}(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \mathcal{N}\left(\boldsymbol{x}; \boldsymbol{\mu}_k, \sigma_k^2 \boldsymbol{\Sigma}\right) \qquad \boldsymbol{\Sigma} \equiv \text{diag}[\boldsymbol{\lambda}^2]$$

- K set adaptively each iteration (except *warm-up*)
- Expected log joint is analytical, entropy via Monte Carlo $\Longrightarrow$ Optimize with SGD (Adam)

Warm-up

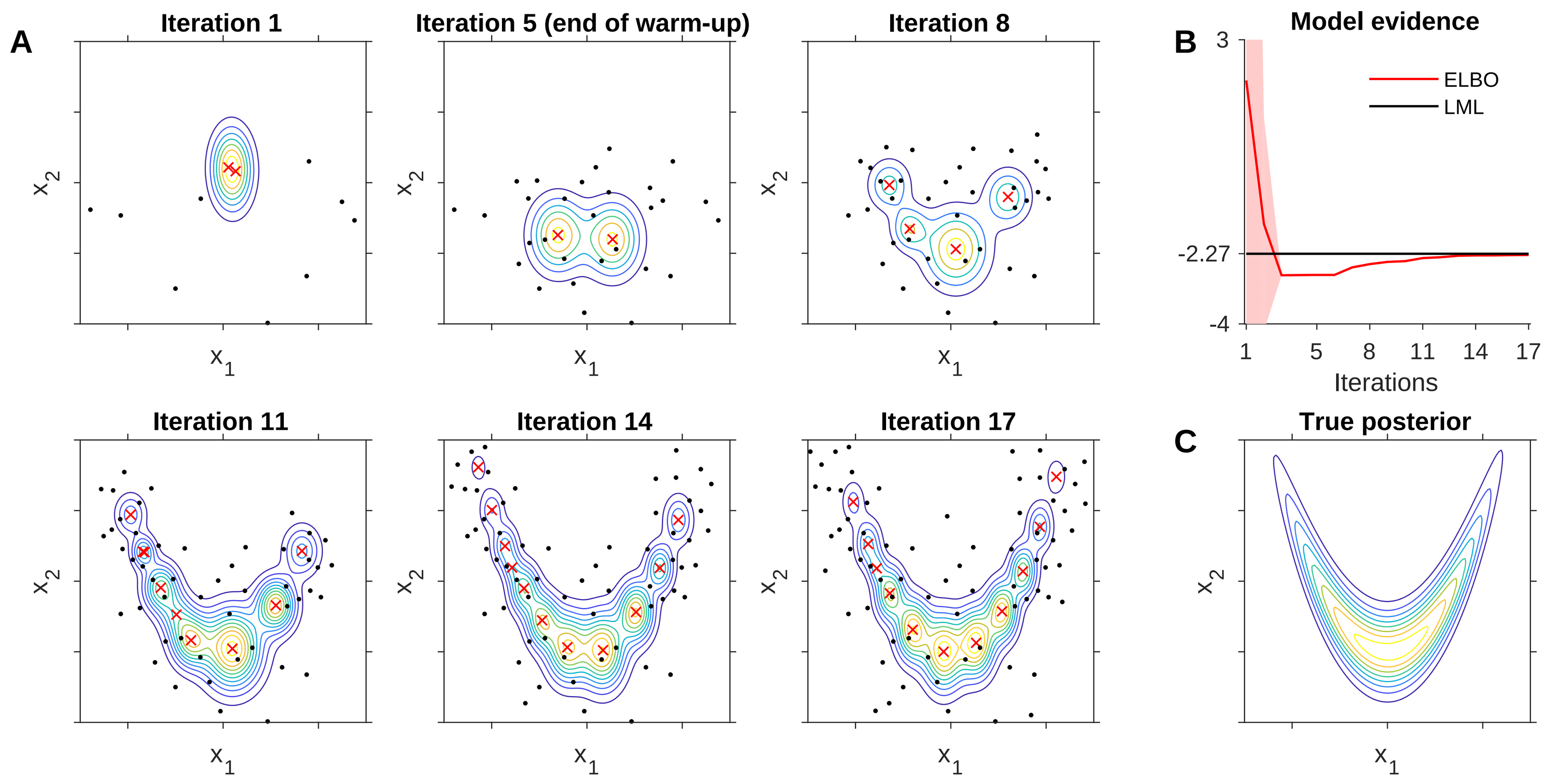
- Clamp $K = 2$, $w_1 = w_2 = 1/2$
- Ends when ELCBO improvement slows down
- $\text{ELCBO}(\boldsymbol{\phi}, f) = \text{ELBO}(\boldsymbol{\phi}, f) - \beta_{\text{LCB}} \cdot \text{SD}\left[\mathbb{E}_{\boldsymbol{\phi}} f\right]$

Variational Bayesian Monte Carlo (VBMC) [1]

In each iteration t:

1. Actively sample new points $\boldsymbol{x}^*$, evaluate $f = \log p(\mathcal{D}|\boldsymbol{x}^*)p(\boldsymbol{x}^*)$
2. train GP model of the log joint f
3. update variational posterior $q_{\boldsymbol{\phi}_t}$ by optimizing the ELBO

Loop until reaching termination criterion

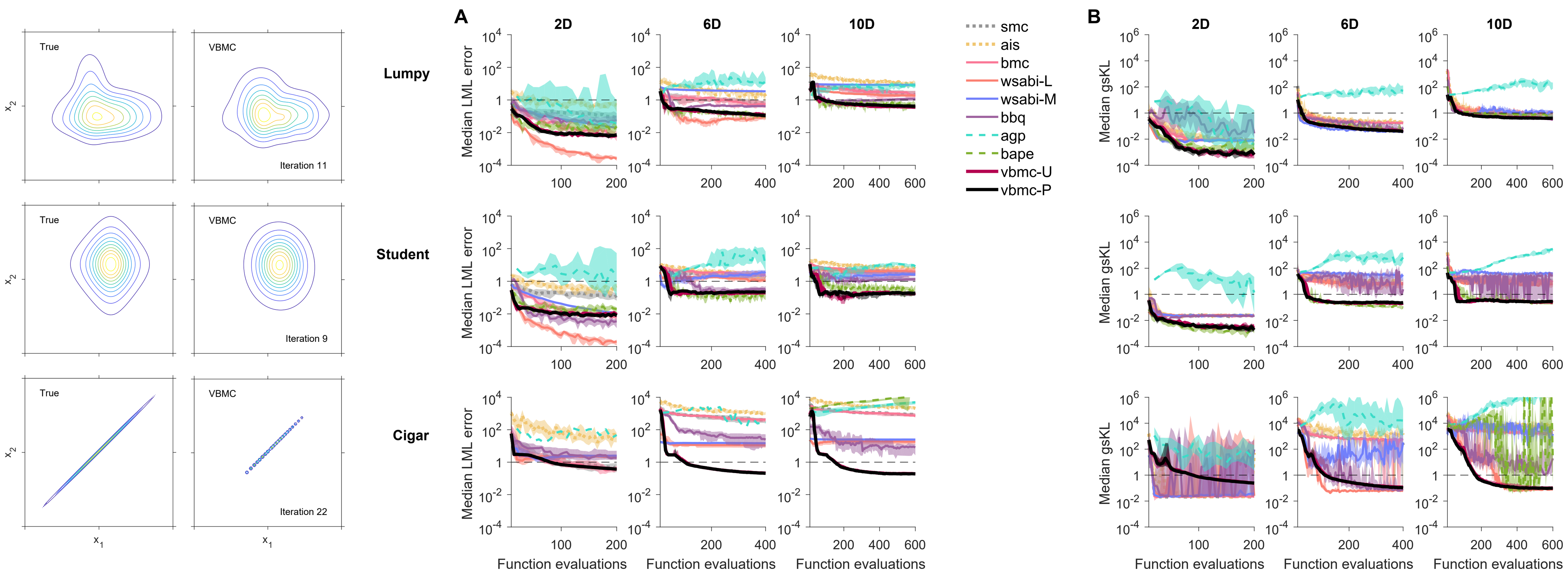


Ready-to-use MATLAB package at: https://github.com/lacerbi/vbmc

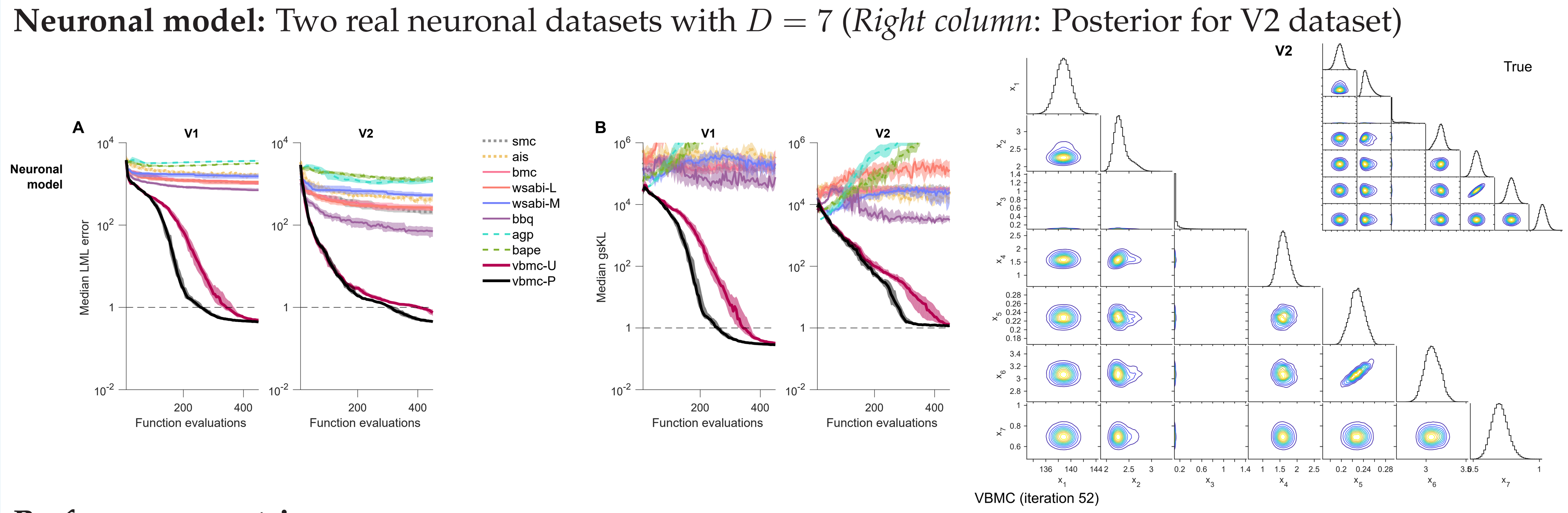
Results

Methods: Simple Monte Carlo (SMC), Annealed importance sampling (AIS), Bayesian Monte Carlo (BMC) [2], Doubly-Bayesian quadrature (BBQ) [3], WSABI [4], Posterior estimation via GPs (AGP, BAPE), VBMC-U (a_{us}), VBMC-P (a_{pro})

Synthetic densities: *Lumpy*, *Student*, *Cigar* families $\times D \in \{2, 4, 6, 8, 10\}$ (*Left column*: $D = 2$ examples)



Neuronal model: Two real neuronal datasets with $D = 7$ (*Right column*: Posterior for V2 dataset)



Performance metrics

A: Median absolute error of the log marginal likelihood (LML) wrt. ground truth

B: Median "Gaussianized" symmetrized KL divergence (gsKL) bw. algorithm's posterior and ground truth

Active Sampling

Optimize *acquisition function*

$$\boldsymbol{x}^* = \arg\max_{\boldsymbol{x}} a(\boldsymbol{x})$$

Generalized uncertainty sampling

$$a_{\text{gus}}(\boldsymbol{x}) = V^{\alpha}(\boldsymbol{x}) q_{\boldsymbol{\phi}}^{\beta}(\boldsymbol{x}) \exp\left(\gamma \overline{f}(\boldsymbol{x})\right), \quad \alpha, \beta, \gamma \geq 0$$

- $\alpha = 1, \beta = 2, \gamma = 0$: 'vanilla' uncertainty sampling
- $\alpha = 1, \beta = 1, \gamma = 1$: prospective uncertainty sampling
- $\alpha = 1, \beta = 0, \gamma = 2$: GP-uncertainty sampling
- $\alpha(n) = \log n, \beta, \gamma = 1$: 'square-root', iter-dependent
- $\alpha(n) = \log n, \beta, \gamma = 1$: 'logarithmic', iter-dependent

Discussion

VBMC produces good approximations on realistic problems, outperforming other methods – why?

BMC,BBQ,WSABI: $Z = \int p(\boldsymbol{x})p(\mathcal{D}|\boldsymbol{x})d\boldsymbol{x}$

VBMC: $\mathcal{I}_k = \int q_k(\boldsymbol{x}) \log\left[p(\boldsymbol{x})p(\mathcal{D}|\boldsymbol{x})\right] d\boldsymbol{x}$

Future directions

- Port VBMC to other languages (Python!)
- Nonstationarity, model mismatch and robustness
- Alternative GP representations
- More principled algorithmic solutions
- Killer application in machine learning?

References

[1] Acerbi, L. (2018). Variational Bayesian Monte Carlo. In *NeurIPS 2018*. arXiv:1810.05558

[2] Ghahramani, Z. & Rasmussen, C. E. (2002) Bayesian Monte Carlo. In *NIPS 2002*.

[3] Osborne, M., Duvenaud, D. K., Garnett, R., Rasmussen, C. E., Roberts, S. J., & Ghahramani, Z. (2012) Active learning of model evidence using Bayesian quadrature. In *NIPS 2012*.

[4] Gunter, T., Osborne, M. A., Garnett, R., Hennig, P., & Roberts, S. J. (2014) Sampling for inference in probabilistic models with fast Bayesian quadrature. In *NIPS 2014*.