

VARIATIONAL BAYESIAN MONTE CARLO

LUIGI ACERBI

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



MOTIVATION

Goal: Bayesian inference with expensive black-box statistical models

Models in science and machine learning

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

Bayesian inference

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

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 $p(\mathbf{x}|\mathcal{D})$ with $q_\phi(\mathbf{x})$
 - Minimize $\text{KL}[q_\phi(\mathbf{x})||p(\mathbf{x}|\mathcal{D})] = \mathbb{E}_{q_\phi} \left[\log \frac{q_\phi(\mathbf{x})}{p(\mathbf{x}|\mathcal{D})} \right]$
- $$\Rightarrow \text{ELBO}(\phi) = \underbrace{\mathbb{E}_{q_\phi} [\log p(\mathcal{D}|\mathbf{x})p(\mathbf{x})]}_{\text{expected log joint}} + \underbrace{\mathcal{H}[q_\phi(\mathbf{x})]}_{\text{entropy}}$$

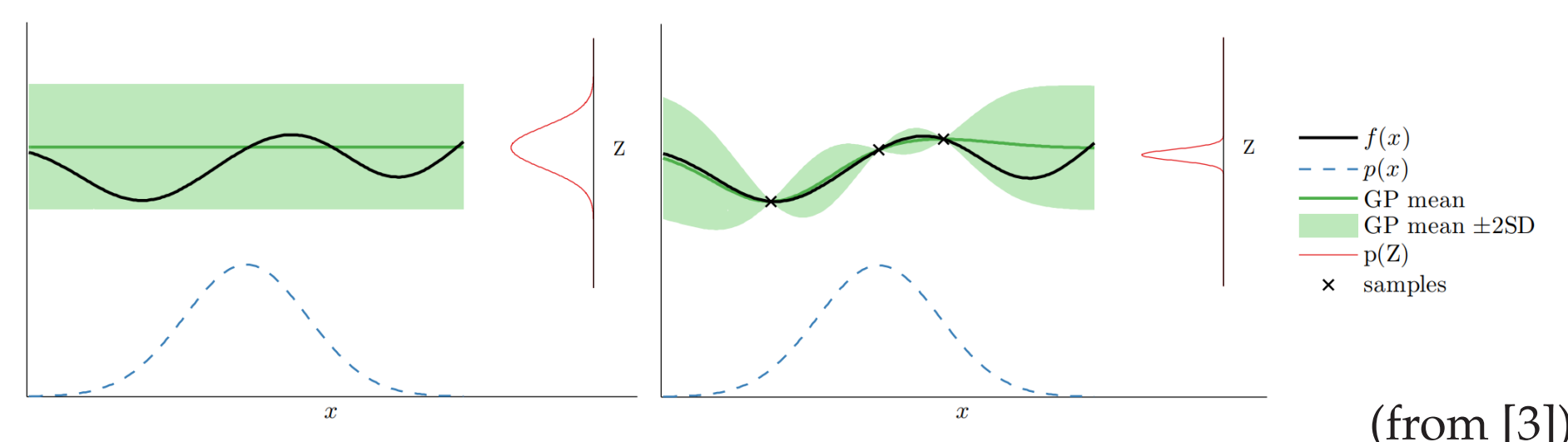
- VI casts inference into optimization + integration
- Obtains $q_\phi(\mathbf{x})$ and $\text{ELBO}(\phi) \leq \log p(\mathcal{D})$

Bayesian quadrature (BQ)

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

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

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



VI + BQ \Rightarrow VBMC

ALGORITHMIC DETAILS

Gaussian process representation

- Sample GP hyperparameters (later optimize)
- Squared exponential covariance, Gaussian noise
- Mean fcn.: $m_{\text{NQ}}(\mathbf{x}) = m_0 - \frac{1}{2} \sum_{i=1}^D \frac{(x^{(i)} - x_m^{(i)})^2}{\omega^{(i)2}}$

Variational posterior

$$q_\phi(\mathbf{x}) = \sum_{k=1}^K w_k \mathcal{N}(\mathbf{x}; \mu_k, \sigma_k^2 \Sigma) \quad \Sigma \equiv \text{diag}[\lambda^2]$$

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

Warm-up

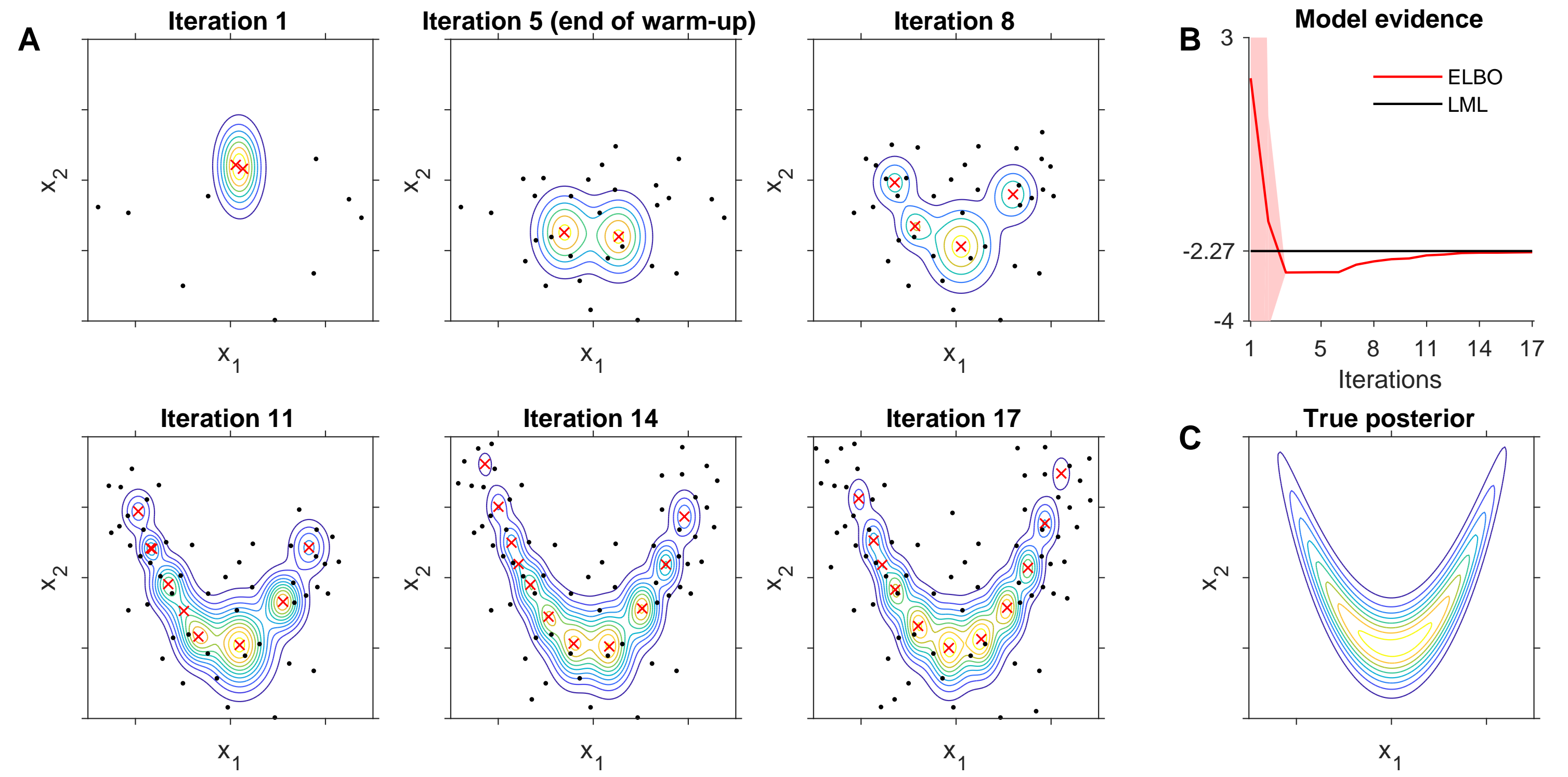
- Clamp $K = 2$, $w_1 = w_2 = 1/2$
- Ends when ELCBO improvement slows down
- $\text{ELCBO}(\phi, f) = \text{ELBO}(\phi, f) - \beta_{\text{LCB}} \cdot \text{SD}[\mathbb{E}_\phi f]$

VARIATIONAL BAYESIAN MONTE CARLO (VBMC) [1]

In each iteration t :

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

Loop until reaching termination criterion

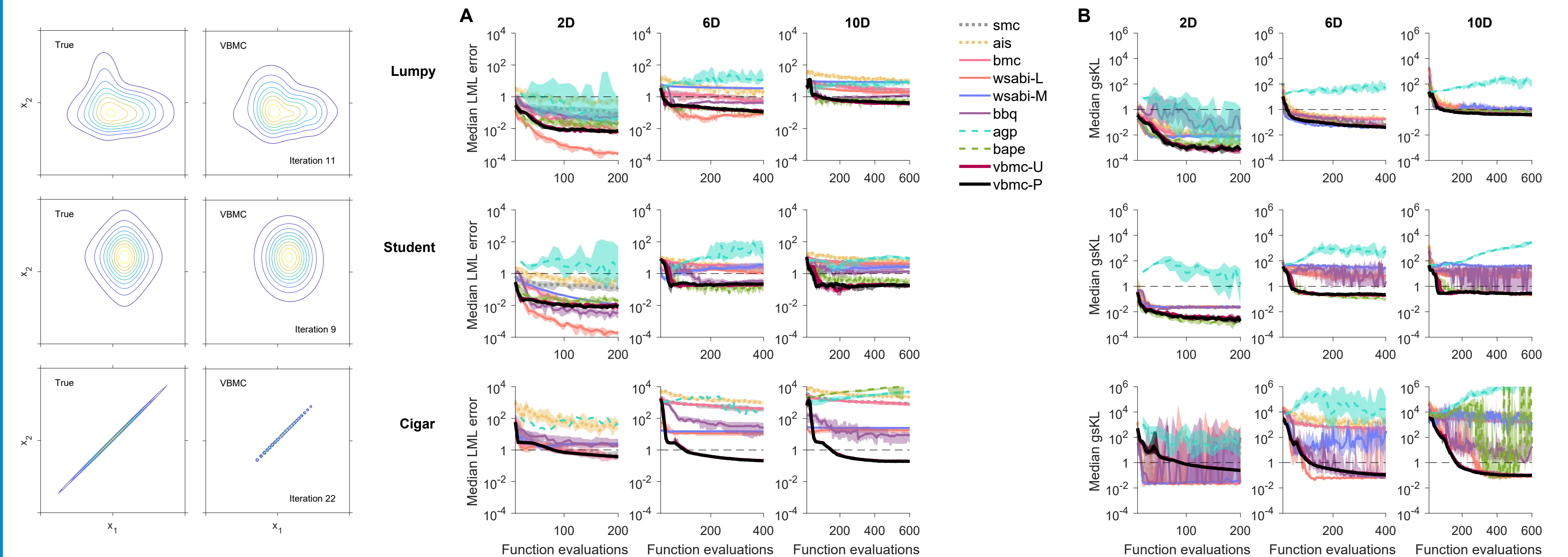


Ready-to-use MATLAB package at: <https://github.com/lacerbi/vbmc>

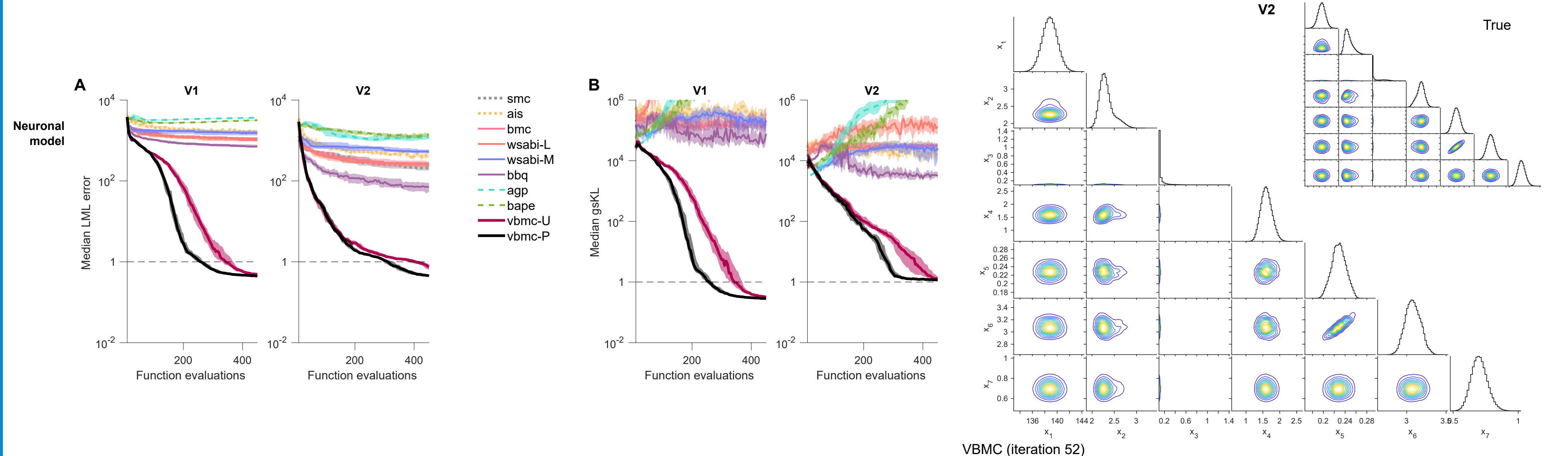
RESULTS

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*

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

Goal: Evaluate $\mathbb{E}_\phi[f] = \int q_\phi(\mathbf{x})f(\mathbf{x})d\mathbf{x}$
 \Rightarrow ‘Vanilla’ uncertainty sampling:

$$a_{\text{us}}(\mathbf{x}) = V(\mathbf{x})q_\phi(\mathbf{x})^2$$

Goal: Evaluate $\mathbb{E}_{\phi_1}[f], \mathbb{E}_{\phi_2}[f], \dots, \mathbb{E}_{\phi_T}[f]$
 \Rightarrow Prospective uncertainty sampling:

$$a_{\text{pro}}(\mathbf{x}) = V(\mathbf{x})q_\phi(\mathbf{x}) \exp(\bar{f}(\mathbf{x}))$$

DISCUSSION

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

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

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

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*.