# Practical sample-efficient Bayesian inference for models with and without likelihoods

Luigi Acerbi

Department of Computer Science
University of Helsinki

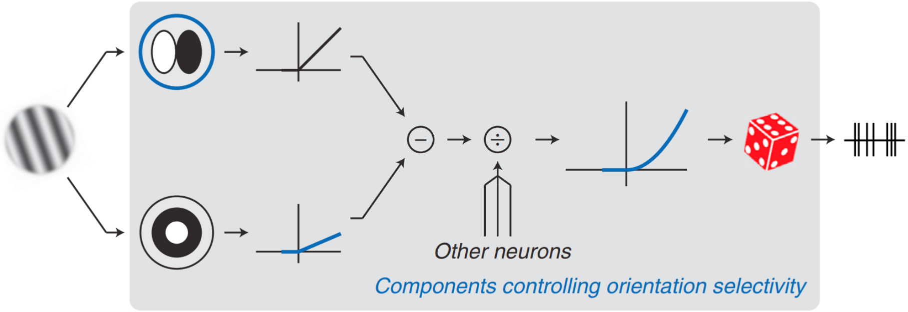
Apr 5, 2021

# The conundrum

In computational neuroscience, we would like:

- to fit complex models. . .
- to get uncertainty over model parameters. . .
- to compare models in a principled way. . .
- to do all this easy and fast, chop chop, thank you.

# Example: LN-LN neuronal model




from Goris et al., *Neuron* (2015)

# Goal

Bayesian inference with expensive black-box statistical models

- Likelihood: $p(\mathcal{D}|\boldsymbol{\theta})$ (data $\mathcal{D}$, parameters $\boldsymbol{\theta} \in \mathcal{X} \subseteq \mathbb{R}^D$)
- No detailed information (e.g., no gradient)
- $\sim$ 500–1000 likelihood evaluations

$$\overbrace{p(\boldsymbol{\theta}|\mathcal{D})}^{\text{posterior}} = \frac{\overbrace{p(\mathcal{D}|\boldsymbol{\theta})}^{\text{likelihood}}\,\overbrace{p(\boldsymbol{\theta})}^{\text{prior}}}{\underbrace{p(\mathcal{D})}_{\text{model evidence}}} \qquad p(\mathcal{D}) = \int p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}$$

# Why Bayesian inference?

$$\overbrace{p(\boldsymbol{\theta}|\mathcal{D})}^{\text{posterior}} = \frac{\overbrace{p(\mathcal{D}|\boldsymbol{\theta})}^{\text{likelihood}}\ \overbrace{p(\boldsymbol{\theta})}^{\text{prior}}}{\underbrace{p(\mathcal{D})}_{\text{evidence}}} \qquad p(\mathcal{D}) = \int p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}$$

- Uncertainty quantification
- Optimal experiment design
- Robustness
- Interpretability
- Hyperparameter tuning
- Model selection

- Better predictions

# Problem

Bayesian inference with expensive black-box statistical models?

Standard approximate Bayesian inference methods

- Markov Chain Monte Carlo (MCMC)
- Variational inference (VI)

require

- many likelihood evaluations
- knowledge of the model (e.g., gradients, detailed structure)

# Sketch solution

Bayesian inference with expensive black-box statistical models?

- Fit *surrogate model* to likelihood evaluations
- Perform *approximate inference* with surrogate model
- Use *active sampling* to smartly evaluate likelihood landscape

# What do we need?

- An *approximate inference* framework: variational inference
- A *surrogate model*: Gaussian processes
- A method to combine the two: Bayesian quadrature

# Variational inference

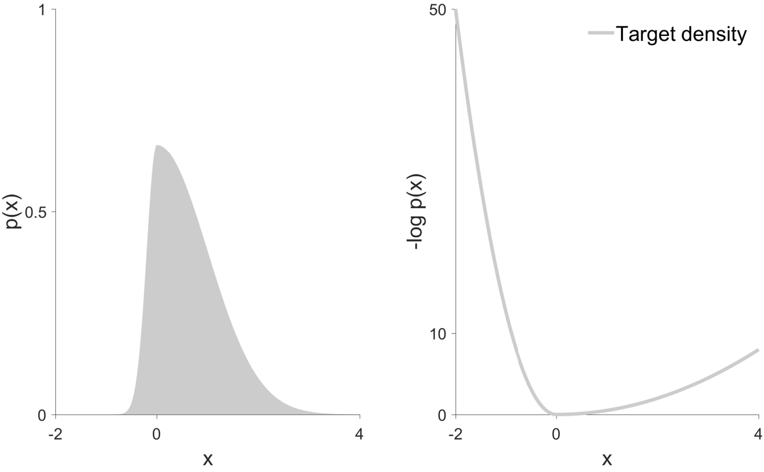
- Approximate $p(\boldsymbol{\theta}|\mathcal{D})$ with $q_\phi(\boldsymbol{\theta})$
- Minimize Kullback-Leibler divergence between $q$ and $p$

Obtains

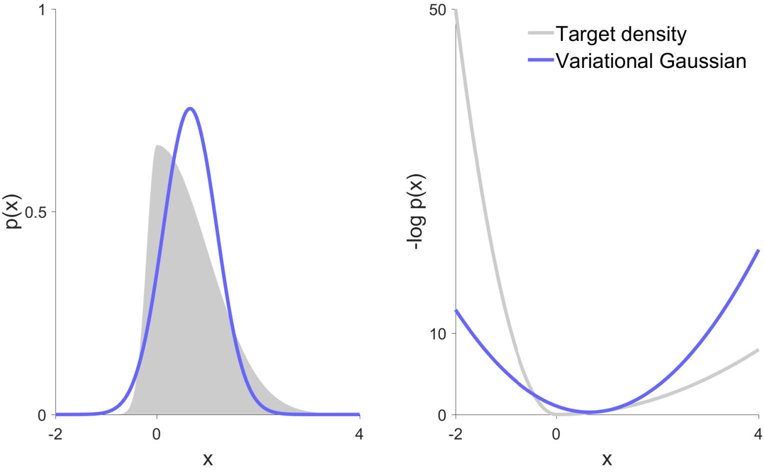
- An approximate posterior $q_\phi(\boldsymbol{\theta})$
- A lower bound to the log marginal likelihood, ELBO($\phi$)

VI casts Bayesian inference into optimization + integration

# Variational inference: example

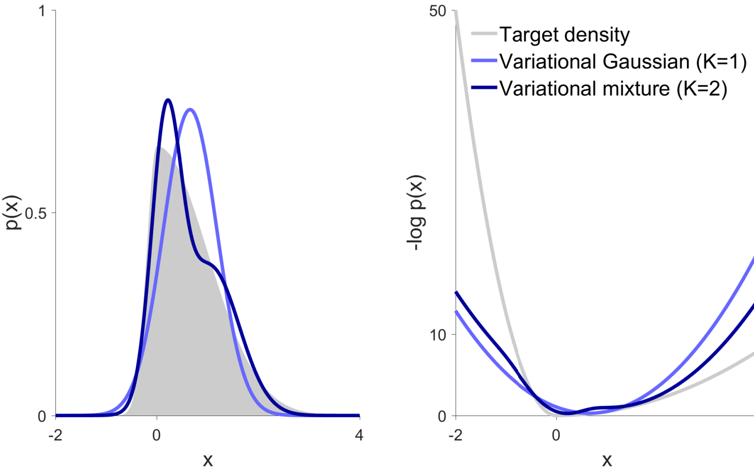


# Variational inference: example



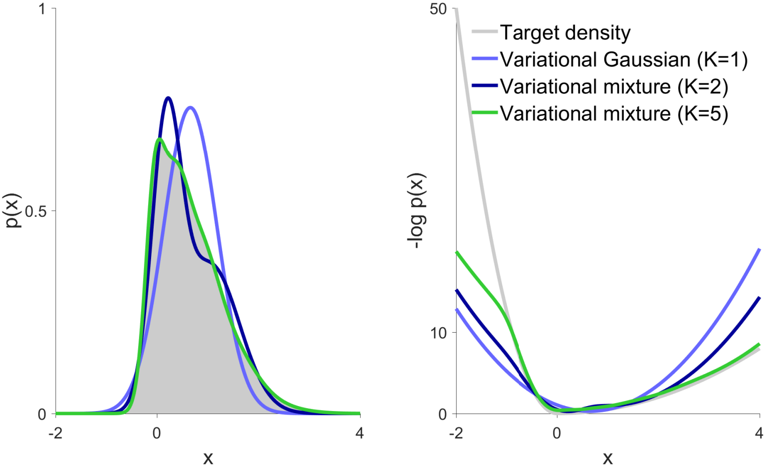

$$q_\phi(x) = \mathcal{N}\left(x, \mu, \sigma^2\right) \qquad \phi = \left(\mu, \sigma^2\right)$$

# Variational inference: example



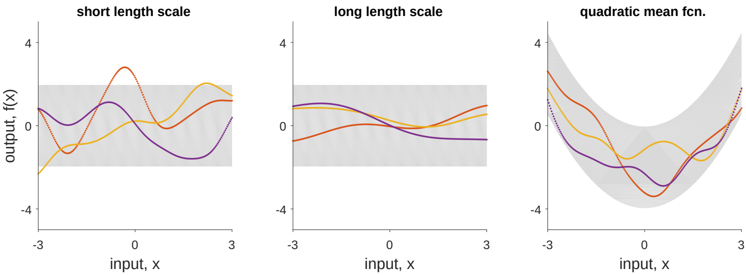

$$q_\phi(x) = \sum_{k=1}^{K} w_k \mathcal{N}\left(x, \mu_k, \sigma_k^2\right) \qquad \phi = \left(w_k, \mu_k, \sigma_k^2\right)_{k=1}^{K}$$

# Variational inference: example

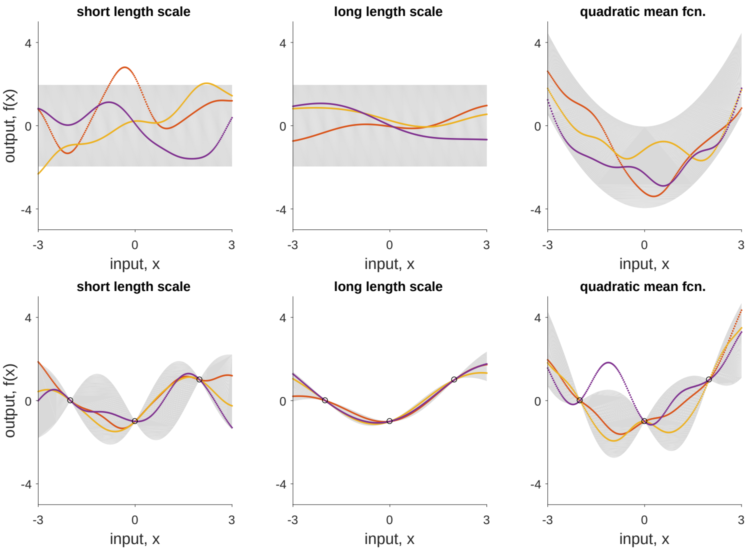



$$q_\phi(x) = \sum_{k=1}^{K} w_k \mathcal{N}\left(x, \mu_k, \sigma_k^2\right) \qquad \phi = \left(w_k, \mu_k, \sigma_k^2\right)_{k=1}^{K}$$

# Gaussian Processes (GPs)



# Gaussian Processes (GPs)



# Bayesian Quadrature (BQ)

**Evaluate integral of (expensive) black-box functions**

$$Z = \int p(\boldsymbol{\theta}) f(\boldsymbol{\theta}) d\boldsymbol{\theta}$$

- $p(\boldsymbol{\theta})$ is Gaussian
- $f(\boldsymbol{\theta})$ approximated via a GP

$\Longrightarrow$ posterior $Z$ can be computed analytically

# Putting things together

- Variational inference:

$$
\begin{aligned}
q_{\phi}(\boldsymbol{\theta}) &= \text{argmax}_{\phi} \text{ELBO}(\phi) \\
&= \text{argmax}_{\phi} \left\{ \int q_{\phi}(\boldsymbol{\theta}) \log \left[ p(\mathcal{D}|\boldsymbol{\theta}) p(\boldsymbol{\theta}) \right] d\boldsymbol{\theta} + \mathcal{H}[q_{\phi}(\boldsymbol{\theta})] \right\}
\end{aligned}
$$

- Bayesian quadrature:

$$
Z = \int q(\boldsymbol{\theta}) f(\boldsymbol{\theta}) d\boldsymbol{\theta}
$$

VI + BQ $\Rightarrow$ VBMC

# Variational Bayesian Monte Carlo (VBMC)

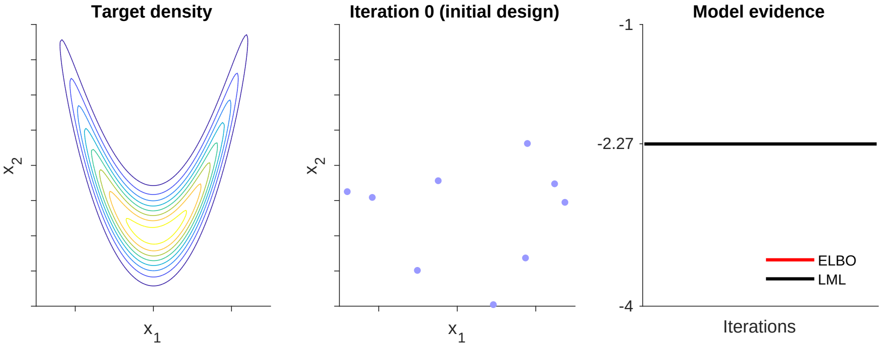
In each iteration $t$:

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

Loop until reaching termination criterion

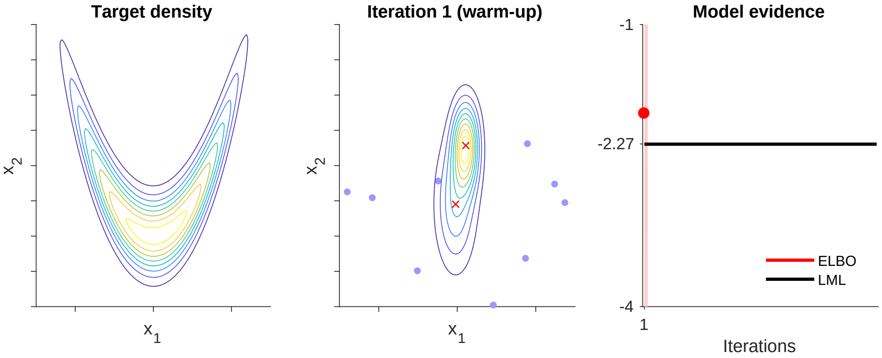
Acerbi, *NeurIPS* (2018), Acerbi, *PMLR* (2019)

# VBMC demo



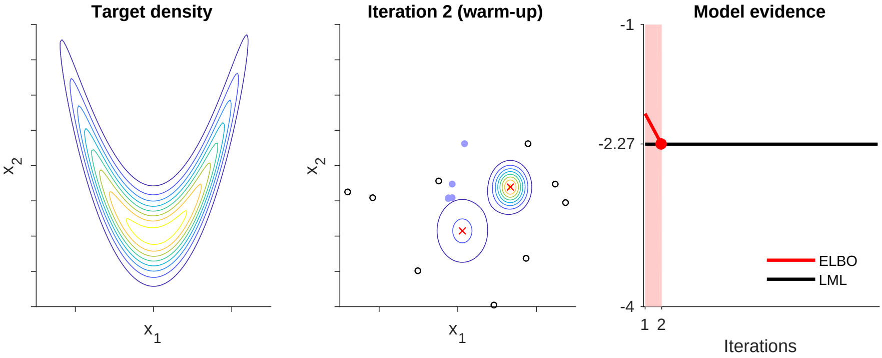

Acerbi, *NeurIPS* (2018)

# VBMC demo



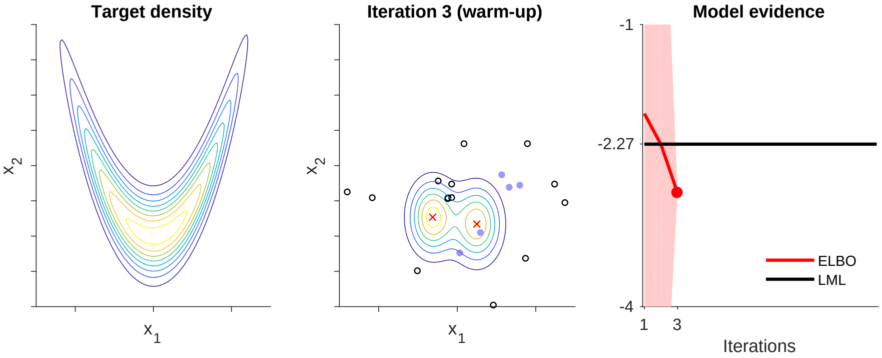

Acerbi, *NeurIPS* (2018)

# VBMC demo



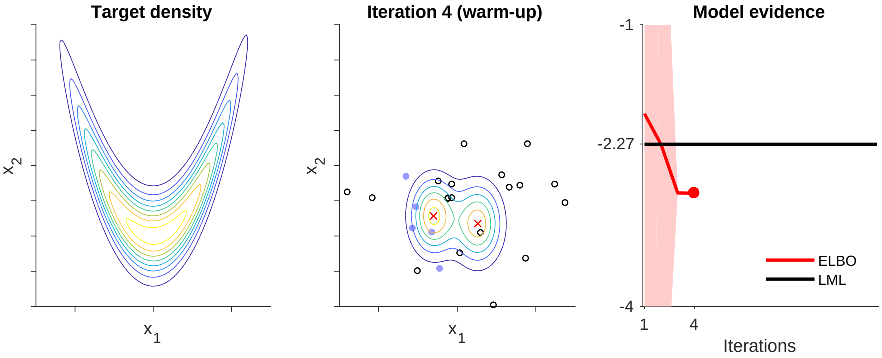

Acerbi, *NeurIPS* (2018)

# VBMC demo



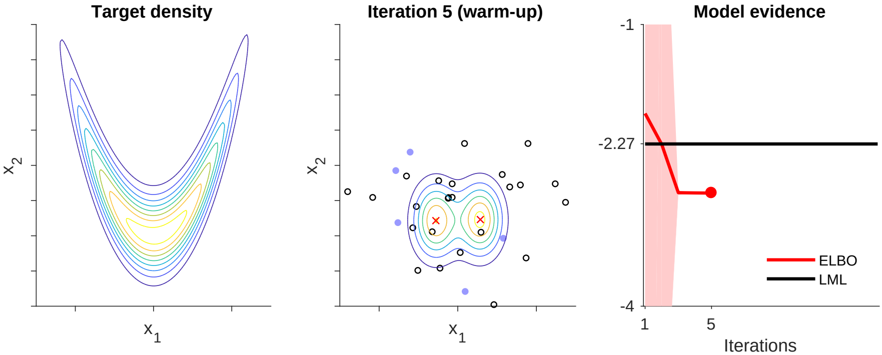

Acerbi, *NeurIPS* (2018)

# VBMC demo



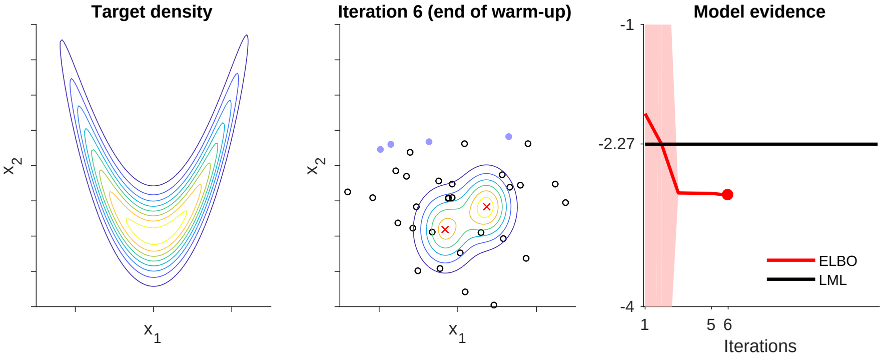

Acerbi, *NeurIPS* (2018)

# VBMC demo



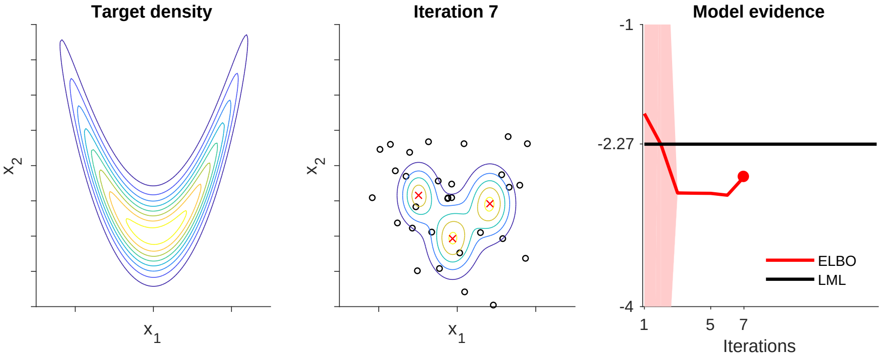

Acerbi, *NeurIPS* (2018)

# VBMC demo



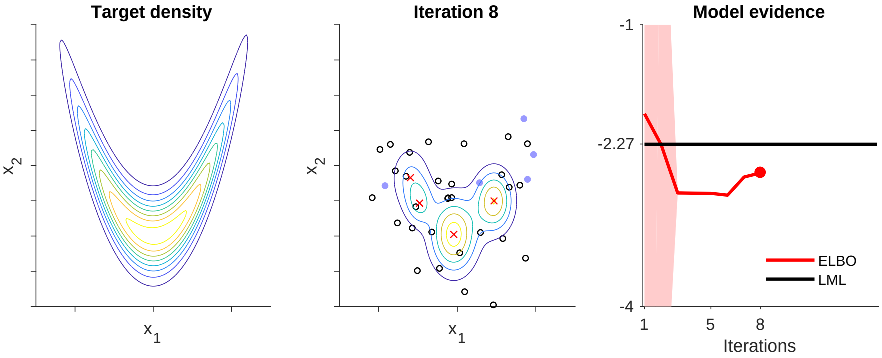

Acerbi, *NeurIPS* (2018)

# VBMC demo



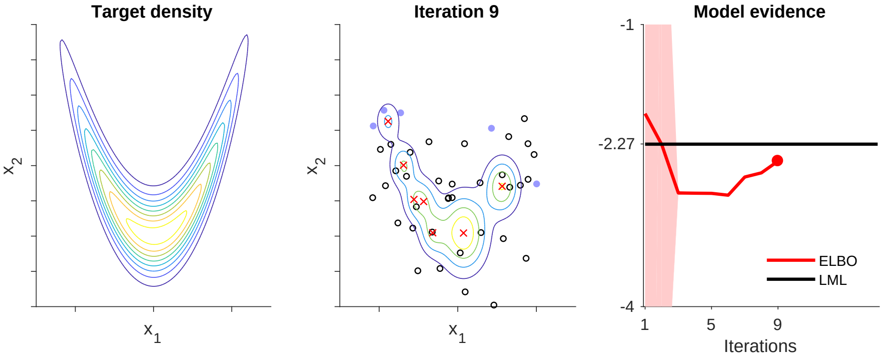

Acerbi, *NeurIPS* (2018)

# VBMC demo



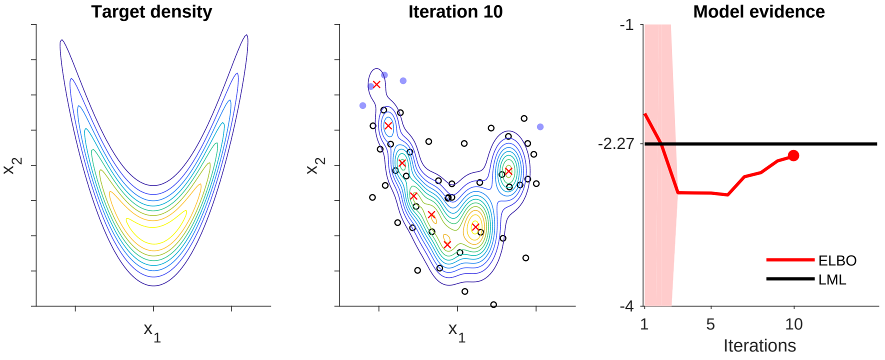

Acerbi, *NeurIPS* (2018)

# VBMC demo



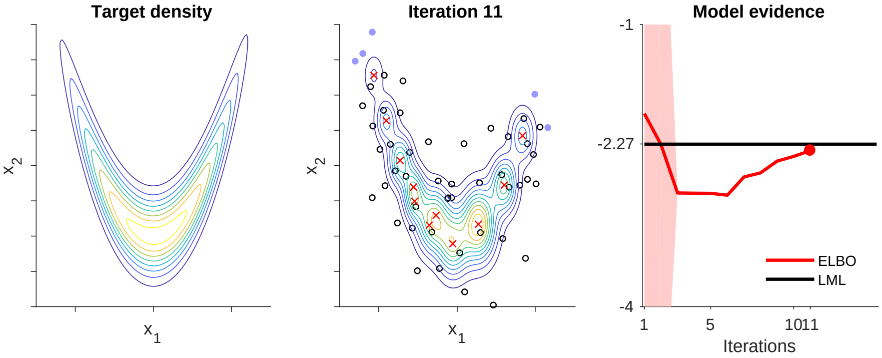

Acerbi, *NeurIPS* (2018)

# VBMC demo



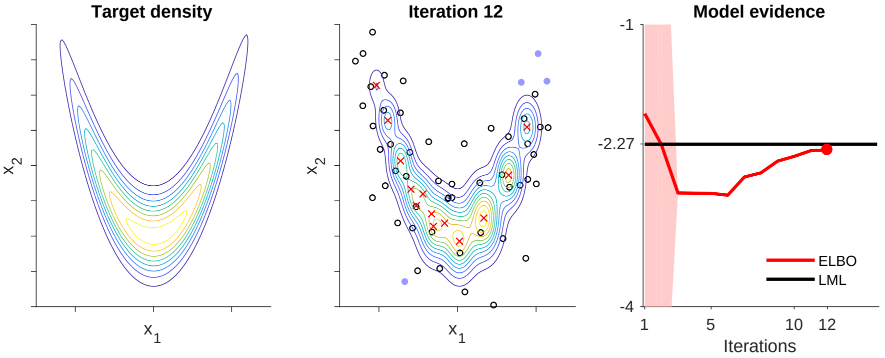

Acerbi, *NeurIPS* (2018)

# VBMC demo



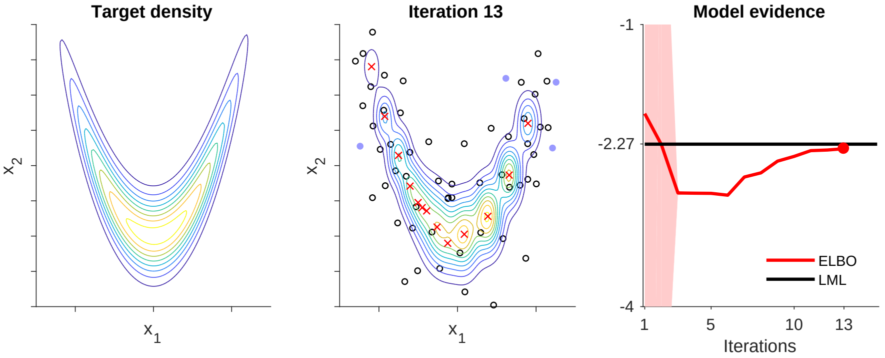

Acerbi, *NeurIPS* (2018)

# VBMC demo



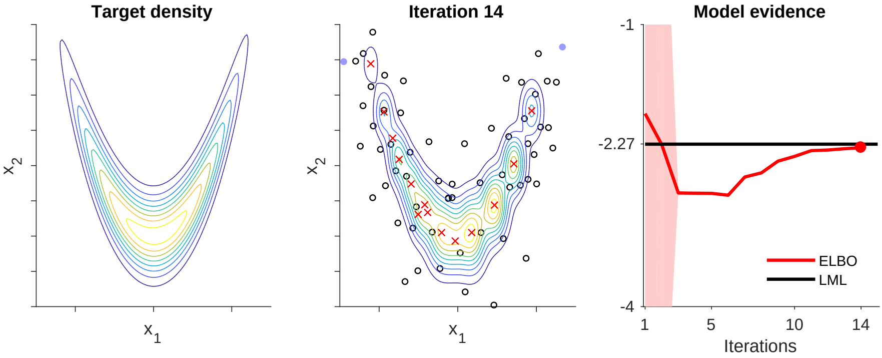

Acerbi, *NeurIPS* (2018)

# VBMC demo



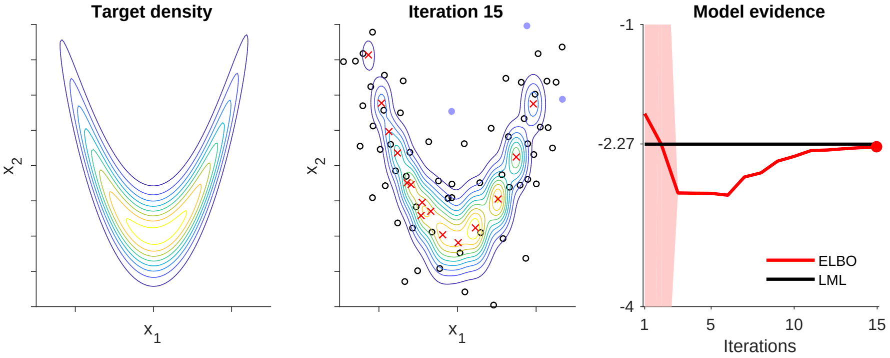

Acerbi, *NeurIPS* (2018)

# VBMC demo




Acerbi, *NeurIPS* (2018)

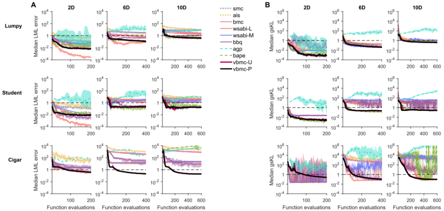
# VBMC demo

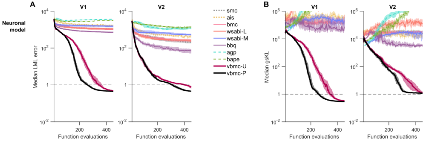



Acerbi, *NeurIPS* (2018)

# VBMC Results

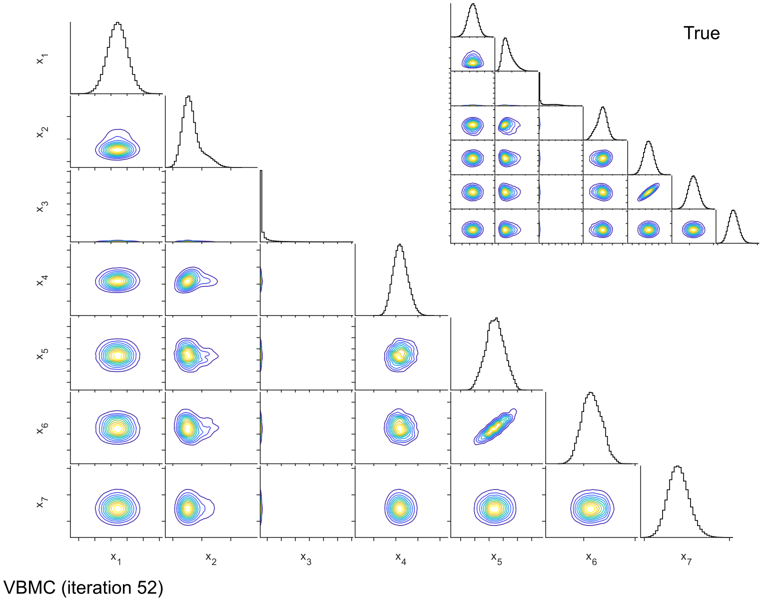
- VBMC is efficient and robust on several problems




Acerbi, *NeurIPS* (2018)

# Example posterior for neuronal model



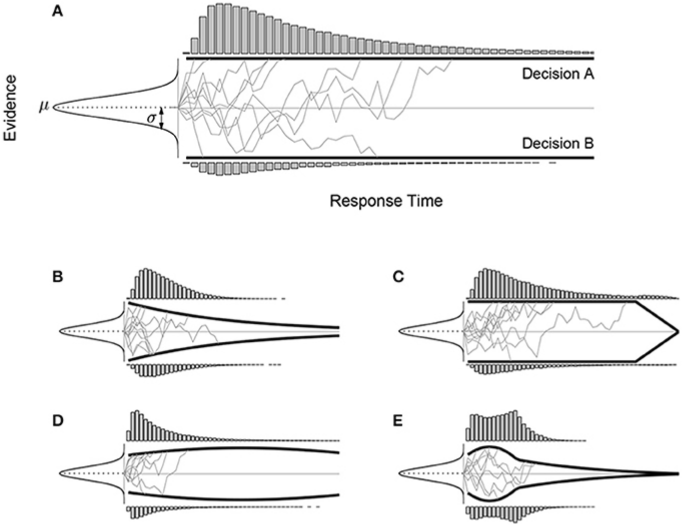
# Limitations

VBMC (2018):

- It requires exact/numerical likelihood
- Relatively smooth and not-too-complex posteriors
- Low-dimensional problems (up to $D \approx 10$)
- Limited to $\approx 10^3$ likelihood evaluations

What if we do not have a likelihood?

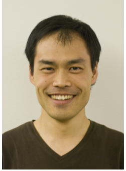
# Example: Drift-diffusion models




from Zhang et al., *Front Psychol* (2014)

# Simulator-based inference

- Approximate Bayesian Computation (ABC)
  - Bypass likelihood, do approximate inference with summary statistics
- Synthetic likelihood
  - Gaussian approximation of likelihood over summary statistics
- Inverse binomial sampling (IBS)
  - Estimate log-likelihood of full data set



Haldane, *Biometrika* (1945)
de Groot, *Annals Math. Stats.* (1959)
van Opheusden*, Acerbi* & Ma, *PLoS Comp Biology* (2020)

# Problem setup

- $N$ data points (e.g., trials)
- $\boldsymbol{X} = \{x_1, \ldots, x_N\}$ independent variables (e.g., experimental conditions)
- $\boldsymbol{Y} = \{y_1, \ldots, y_N\}$ dependent variables (e.g., observations)

**Goal:** Compute $\log p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta})$

$$\log p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}) = \sum_{n=1}^{N} \log p(y_n|y_1, \ldots, y_{n-1}, x_1, \ldots, x_N, \boldsymbol{\theta})$$

# Problem setup

- $N$ data points (e.g., trials)
- $\boldsymbol{X} = \{x_1, \ldots, x_N\}$ independent variables (e.g., experimental conditions)
- $\boldsymbol{Y} = \{y_1, \ldots, y_N\}$ dependent variables (e.g., observations)

**Goal:** Compute $\log p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta})$

$$\log p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}) = \sum_{n=1}^{N} \log p(y_n|x_n, \boldsymbol{\theta}) \qquad \text{(simple case)}$$

# Problem setup

- $N$ data points (e.g., trials)
- $\boldsymbol{X} = \{x_1, \ldots, x_N\}$ independent variables (e.g., experimental conditions)
- $\boldsymbol{Y} = \{y_1, \ldots, y_N\}$ dependent variables (e.g., observations)

**Goal:** Compute $\log p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta})$

$$\log p(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{\theta}) = \sum_{n=1}^{N} \log p(y_n|y_1, \ldots, y_{n-1}, x_1, \ldots, x_N, \boldsymbol{\theta})$$
$$\equiv \sum_{n=1}^{N} \mathcal{L}_n$$

# Inverse binomial sampling (IBS)

**Requirements:**

- We can sample $y_n \sim p(y_n | y_1, \ldots, y_{n-1}, x_1, \ldots, x_N, \boldsymbol{\theta})$
- $y_n$ is discrete
- $\forall n$, $p(y_n | y_1, \ldots, y_{n-1}, x_1, \ldots, x_N, \boldsymbol{\theta})$ is non-negligible

## Basic algorithm:

For each data point $n$:

- Sample $\tilde{y} \sim p(y_n | y_1, \ldots, y_{n-1}, x_1, \ldots, x_N, \boldsymbol{\theta})$ until $\tilde{y} \equiv y_n$
- Let $K$ be the number of samples
- $\widehat{\mathcal{L}}_{n,\text{IBS}}(\boldsymbol{\theta}) = -\sum_{k=1}^{K-1} \frac{1}{k}$

$$\widehat{\mathcal{L}}_{IBS}(\boldsymbol{\theta}) = \sum_{n=1}^{N} \widehat{\mathcal{L}}_{n,\text{IBS}(\boldsymbol{\theta})}$$

van Opheusden*, Acerbi* & Ma, *PLoS Comp Biology* (2020)

# Properties of the IBS estimator

The IBS estimator $\widehat{\mathcal{L}}_{IBS}(\boldsymbol{\theta})$:

- is unbiased
- is $\approx$ normally distributed
- has uniformly bounded variance
- comes with a calibrated estimator of the variance
- is perfectly suited for the GP regression model

$$z_i = f(\boldsymbol{\theta}_i) + \sigma_{\text{obs}}(\boldsymbol{\theta}_i)\varepsilon_i \qquad \varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$$

van Opheusden*, Acerbi* & Ma, *PLoS Comp Biology* (2020)

# VBMC with noisy likelihoods

**Goal:**

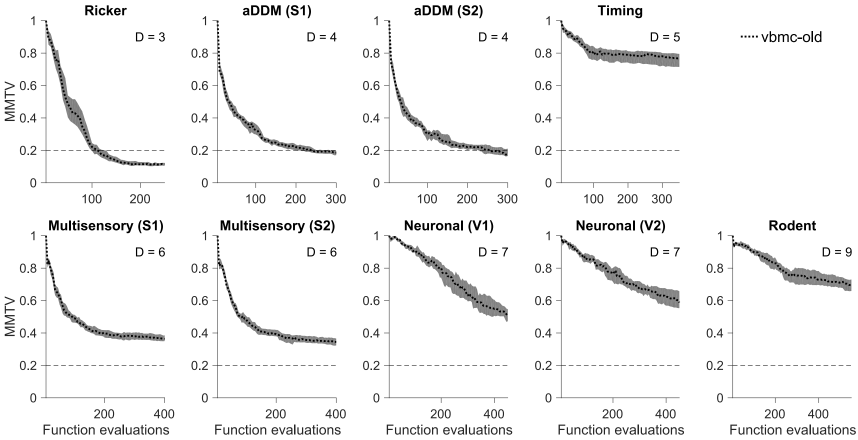
Extend VBMC (2018) to noisy log-likelihood evaluations

$$z_i = f(\boldsymbol{\theta}_i) + \sigma_{\text{obs}}(\boldsymbol{\theta}_i)\varepsilon_i \qquad \varepsilon_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$$

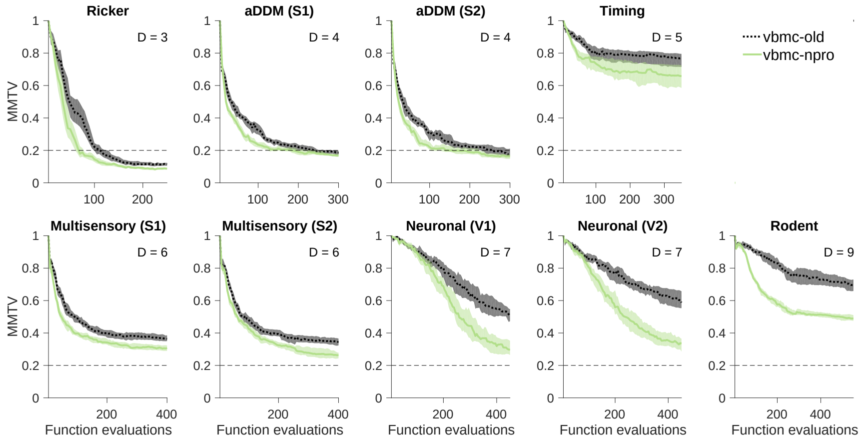
Acerbi, *NeurIPS* (2020)

Is an extension needed?

# Old VBMC with noisy likelihoods



# Old VBMC with noisy likelihoods



# Why is old VBMC failing with noise?

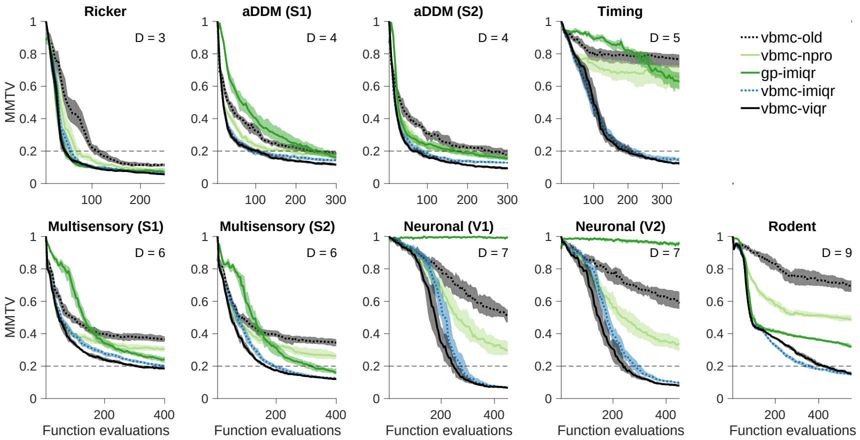
Original VBMC uses *local* acquisition function

- Maximizes *pointwise* reduction in uncertainty of the ELBO
- Uses *variance* as metric for uncertainty

We need a *global* and *robust* acquisition function

- Järvenpää, Gutmann, Vehtari, & Marttinen (2019) propose *integrated median interquantile range* (IMIQR)
  - Minimizes *integrated uncertainty* of the posterior over the space
  - Uses *robust statistics* for uncertainty (IQR vs. variance)
- We also implement *variational interquantile range* (VIQR)
  - Fast variational approximation of IMIQR

# New VBMC with noisy likelihoods




Acerbi, *NeurIPS* (2020)

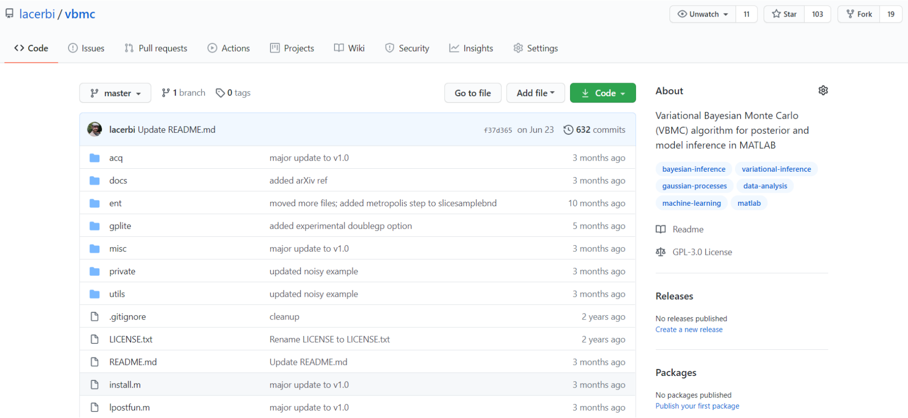
# Speed comparison

Table 1: Average algorithmic overhead per likelihood evaluation (in seconds) over a full run, assessed on a single-core reference machine (mean $\pm$ 1 SD across 100 runs).

| | Model | | | | | |
|---|---|---|---|---|---|---|
| Algorithm | Ricker | aDDM | Timing | Multisensory | Neuronal | Rodent |
| VBMC-VIQR | $\mathbf{1.5 \pm 0.1}$ | $\mathbf{1.5 \pm 0.1}$ | $\mathbf{1.8 \pm 0.2}$ | $\mathbf{2.0 \pm 0.2}$ | $\mathbf{2.8 \pm 0.8}$ | $\mathbf{2.6 \pm 0.2}$ |
| VBMC-IMIQR | $5.5 \pm 0.5$ | $5.1 \pm 0.3$ | $5.8 \pm 0.6$ | $5.6 \pm 0.3$ | $6.5 \pm 1.3$ | $5.6 \pm 0.4$ |
| GP-IMIQR | $15.6 \pm 0.9$ | $16.0 \pm 1.7$ | $17.1 \pm 1.2$ | $26.3 \pm 1.8$ | $29.6 \pm 2.8$ | $40.1 \pm 2.1$ |

Acerbi, *NeurIPS* (2020)

# Toolbox




https://github.com/lacerbi/vbmc

# Discussion

**Limitations of** VBMC (2020):

- ~~It requires exact/numerical likelihood~~
- Relatively smooth and not-too-complex posteriors
- Low-dimensional problems (up to $D \approx 10$)
- Limited to $\approx 10^3$ likelihood evaluations

**Future directions:**

- Model mismatch and robustness (e.g., nonstationarity)
- Alternative representations
- Extend the usage cases for ‘smart’ Bayesian inference
- More applications in computational neuroscience!

# Final slide

- Variational Bayesian Monte Carlo (VBMC)
  - MATLAB toolbox: `https://github.com/lacerbi/vbmc`
  - 2018 paper: `https://arxiv.org/abs/1810.05558`
  - 2020 paper: `https://arxiv.org/abs/2006.08655`
- Inverse binomial sampling (IBS)
  - MATLAB toolbox: `https://github.com/lacerbi/ibs`
  - Paper: `https://doi.org/10.1371/journal.pcbi.1008483`

Get in touch! `@AcerbiLuigi`, `luigi.acerbi@helsinki.fi`

**Acknowledgments**

- Bas van Opheusden and Weiji Ma
- Alexandre Pouget and the Pouget lab
- Finnish Center for Artificial Intelligence (FCAI)

Thanks!