Variational Bayesian Monte Carlo

with and without Noisy Likelihoods

(A Review)

Gagesh Madaan
Musale Krushna Pavan
Pinaki Chakraborty
Shruti Sharma

Indian Institute of Technology, Kanpur

August 15, 2022

TABLE OF CONTENTS

Objective

Objective: Approximate Bayesian Inference .

Bayesian Inference

Posterior : $p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$

Marginal Likelihood : $p(D) = \int p(D|\theta)p(\theta)d\theta$

Objective

Objective: Approximate Bayesian Inference .

Bayesian Inference

Posterior : $p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)}$

Marginal Likelihood : $p(D) = \int p(D|\theta)p(\theta)d\theta$

Standard methods: $\begin{cases} \text{MCMC} \\ \text{VI} \end{cases}$

Issues: $\begin{cases} \text{many likelihood evaluations} \\ \text{Needs white-box models} \\ \text{No noise} \end{cases}$

Background

1. Variational Inference:

$$\max \; \mathcal{L}[q_\phi] = \mathbb{E}[\underbrace{\log p(\mathcal{D}|\mathbf{x})p(\mathbf{x})}_{f(x)}] + \mathcal{H}(q_\phi(\mathbf{x}))$$

Background

1. Variational Inference:

$$\max \; \mathcal{L}[q_\phi] = \mathbb{E}[\underbrace{\log p(\mathcal{D}|\mathbf{x})p(\mathbf{x})}_{f(x)}] + \mathcal{H}(q_\phi(\mathbf{x}))$$

2. Gaussian Process(GP): a distribution over functions

$$f \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\kappa})$$

Background

1. Variational Inference:

$$\max \; \mathcal{L}[q_\phi] = \mathbb{E}[\underbrace{\log p(\mathcal{D}|\mathbf{x})p(\mathbf{x})}_{f(x)}] + \mathcal{H}(q_\phi(\mathbf{x}))$$

2. Gaussian Process(GP): a distribution over functions

$$f \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\kappa})$$

3. Bayesian quadrature:

$$\langle f \rangle = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$$

Background

1. Variational Inference:

$$\max \; \mathcal{L}[q_\phi] = \mathbb{E}[\underbrace{\log p(\mathcal{D}|\mathbf{x})p(\mathbf{x})}_{f(x)}] + \mathcal{H}(q_\phi(\mathbf{x}))$$

2. Gaussian Process(GP): a distribution over functions

$$f \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\kappa})$$

3. Bayesian quadrature:

$$\langle f \rangle = \int f(\mathbf{x})\pi(\mathbf{x})d\mathbf{x}$$

4. Active sampling:

$$\boldsymbol{x}_{new} = \arg\max_{\boldsymbol{x}} a(\boldsymbol{x})$$

- Fast, smart, robust and efficient.

In each iteration t,

1. **Exploration-Exploitation** : Actively sample sequential $n_{active}(= 5)$ new points x^* that maximise the acquisition function $a(\theta)$ and evaluate log-joint $f = \log p(D|x^*)p(x^*)$ at each point.

- Fast, smart, robust and efficient.

In each iteration t,

1. **Exploration-Exploitation** : Actively sample sequential $n_{active} (= 5)$ new points x^* that maximise the acquisition function $a(\theta)$ and evaluate log-joint $f = \log p(D|x^*)p(x^*)$ at each point.
2. Train GP surrogate model of the log-joint f; Training set consists of the points evaluated so far.

VBMC : Overview

- Fast, smart, robust and efficient.

In each iteration t,

1. **Exploration-Exploitation** : Actively sample sequential $n_{active} (= 5)$ new points x^* that maximise the acquisition function $a(\theta)$ and evaluate log-joint $f = \log p(D|x^*)p(x^*)$ at each point.
2. Train GP surrogate model of the log-joint f; Training set consists of the points evaluated so far.
3. Update variational posterior approximation $q_{\phi_t}(x)$ by optimizing surrogate ELBO calculated via Bayesian Quadrature.

VBMC : Overview

- Fast, smart, robust and efficient.

In each iteration t,

1. **Exploration-Exploitation** : Actively sample sequential $n_{active} (= 5)$ new points x^* that maximise the acquisition function $a(\theta)$ and evaluate log-joint $f = \log p(D|x^*)p(x^*)$ at each point.
2. Train GP surrogate model of the log-joint f; Training set consists of the points evaluated so far.
3. Update variational posterior approximation $q_{\phi_t}(x)$ by optimizing surrogate ELBO calculated via Bayesian Quadrature.

Loop until termination criterion (eg. when reliability index $\rho(t) \leq 1$ for $n_{stable} = 8$ iterations or when n_{max} function evaluations) is met.

VBMC : Overview

- Fast, smart, robust and efficient.

In each iteration t,

1. **Exploration-Exploitation** : Actively sample sequential $n_{active}(= 5)$ new points x^* that maximise the acquisition function $a(\theta)$ and evaluate log-joint $f = \log p(D|x^*)p(x^*)$ at each point.
2. Train GP surrogate model of the log-joint f; Training set consists of the points evaluated so far.
3. Update variational posterior approximation $q_{\phi_t}(x)$ by optimizing surrogate ELBO calculated via Bayesian Quadrature.

Loop until termination criterion (eg. when reliability index $\rho(t) \leq 1$ for $n_{stable} = 8$ iterations or when n_{max} function evaluations) is met.

Return : Estimate of mean and standard deviation of ELBO, and variational posterior.

VBMC : Algorithmic Details

Gaussian Process Representation

- Sample GP hyperparameters and optimize them later.
- GP surrogate with squared exponential kernel, Gaussian likelihood with observation noise $\sigma_{obs} > 0$
- Negative quadratic mean,

$$m_{NQ}(\boldsymbol{x}) = m_0 - \frac{1}{2} \sum_{i=1}^{D} \frac{(x^{(i)} - x_m^{(i)})^2}{(\omega^{(i)})^2}$$

VBMC : Algorithmic Details

Gaussian Process Representation

- Sample GP hyperparameters and optimize them later.
- GP surrogate with squared exponential kernel, Gaussian likelihood with observation noise $\sigma_{obs} > 0$
- Negative quadratic mean,

$$m_{NQ}(\boldsymbol{x}) = m_0 - \frac{1}{2}\sum_{i=1}^{D}\frac{(x^{(i)} - x_m^{(i)})^2}{(\omega^{(i)})^2}$$

Variational Posterior

$$q_\phi(\boldsymbol{x}) = \sum_{k=1}^{K} w_k \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_k, \sigma_k^2 \Sigma)$$

- K is set adaptively in each iteration (except warm-up). Initially, K=2.
- Expected log-joint f is analytical. Entropy of $q_\phi(x)$ is estimated via Monte Carlo sampling, and its gradients via reparameterization trick. Optimize ELBO via SGD.

VBMC : Acquisition Functions

To perform active sampling, solve this optimization problem:

$$x^* = \underset{x}{argmax}\, a(x)$$

- **'Vanilla' Uncertainty Sampling :** Maximize variance under current variational parameters; Lacks exploration.
- **Prospective Uncertainty Sampling :** Reduces uncertainty of variational objective both for current posterior and at prospective locations where it might go. It selects points from regions of high probability density.

VBMC : Acquisition Functions

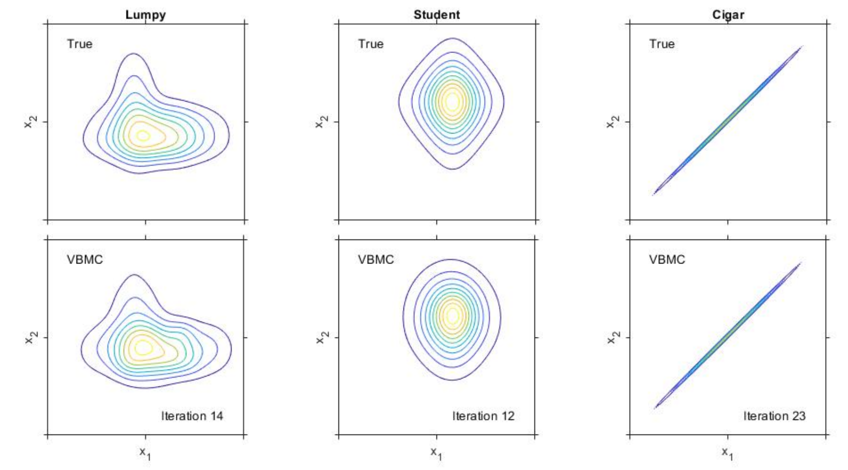
- **Noisy Prospective Uncertainty Sampling :** Account for potential noise at the chosen point location for maximizing.
- **Expected Information Gain :** Sample points that maximize the EIG of integral $\mathcal{G}$ present in ELBO's equation and choose the next location θ^* that maximizes mutual information $I[\mathcal{G}; y_*]$
- **Variational Interquantile Range :** Replace the surrogate posterior inside the integrated median interquantile range function integral with variational posterior (up to a normalization constant). It can be approximated via simple Monte Carlo methods.

Experiments

- The MATLAB code for the VBMC framework is maintained in the github repository `https://github.com/lacerbi/vbmc`.
- We used the existing github repository `https://github.com/lacerbi/infbench` actively maintained by the original author (Luigi Acrebi) to run some of the benchmarks.
- However, due to logistical challenges and the experimental nature of the code repository, some of the comparisons could not be performed.
- We were able to reproduce the custom target densities and corresponding example solutions as described in `https://arxiv.org/pdf/1810.05558.pdf`

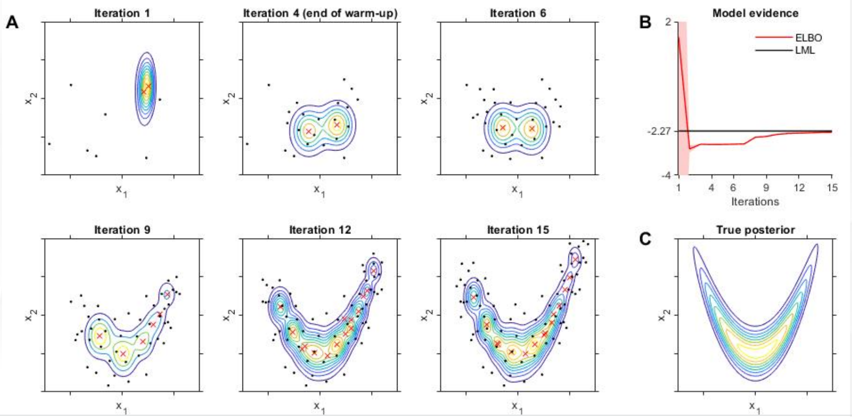
Experiments

Figure: Top: Contour plots of 2D custom target densities Bottom: Contour plots of variational posteriors returned by VBMC



Experiments

Figure: Example run of VBMC on 2-D Banana Distribution, (http://www.roboticsproceedings.org/rss08/p34.pdf) **A** Contour plots of the variational posterior at different iterations of the algorithm. **Red crosses indicate the centers of the variational mixture components, black dots are the training samples**. **B** ELBO as a function of iteration. The black line is the true log marginal likelihood (LML). **C** True target pdf



Experiments

In the following slides, we show the vbmc framework being run on a real-world data set taken from Goris, R. L., Simoncelli, E. P., Movshon, J. A. (2015).

Origin and function of tuning diversity in macaque visual cortex. *Neuron*, 88(4), 819-831

`https://www.sciencedirect.com/science/article/pii/S0896627315008752`

The dimensionality value for this problem is 7. The problem parameters are shown in next slide.

Experiments

Figure: **MATLAB command window**

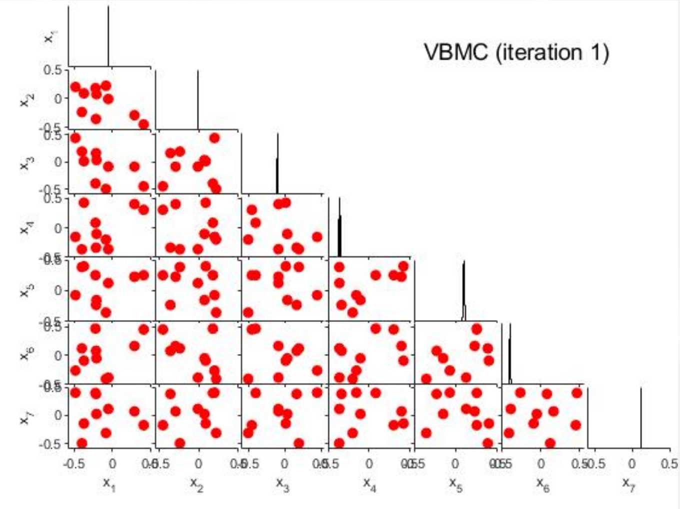
```
struct with fields:

                 ProbSet: 'vbmc18'
                  Number: 6
                    Prob: 'goris2015'
                 SubProb: 'S7'
                      Id: 1
                ProbInfo: [1×1 struct]
                   Title: 'goris2015'
                    func: '@(x_,probstruct_) infbench_goris2015(x_(:)',probstruct_.ProbInfo)'
                   Noise: []
           NoiseEstimate: 0
                       D: 7
                      LB: [-Inf -Inf -Inf -Inf -Inf -Inf -Inf]
                      UB: [Inf Inf Inf Inf Inf Inf Inf]
                     PLB: [-0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000 -0.5000]
                     PUB: [0.5000 0.5000 0.5000 0.5000 0.5000 0.5000 0.5000]
                    Mean: [5.0528e-17 0.3346 0.0428 0.0428 -4.4837e-17 0.1053 0.9566]
                     Cov: [7×7 double]
                    Mode: [-0.2119 -0.0738 -0.2809 -0.1039 0.3805 -0.0890 0.4155]
                     lnZ: 0
             MaxFunEvals: 450
                  TolFun: 1.0000e-06
               SaveTicks: [1×90 double]
              NoiseSigma: 0
          NoiseIncrement: 0
           LocalDataFile: []
 VariableComputationTime: 0
  NonAdmissibleFuncValue: -708.3964
             AddLogPrior: 0
                   Debug: 0
      NoiseEstimateJitter: 0
        TotalMaxFunEvals: 450
                 Verbose: 1
                  trinfo: [1×1 struct]
```

Figure: **MATLAB command window**

57	285	-2620.03	0.12	0.06	27	1.61	
58	290	-2620.00	0.12	0.12	28	1.99	
59	295	-2620.05	0.06	0.11	29	1.72	
60	300	-2619.96	0.07	0.05	29	1.23	
61	305	-2619.95	0.05	0.01	29	0.277	
62	310	-2619.93	0.05	0.02	30	0.456	
63	315	-2619.88	0.05	0.01	32	0.454	
64	320	-2619.83	0.17	0.12	32	2.21	
65	325	-2619.81	0.05	0.11	32	1.61	
66	330	-2619.79	0.05	0.00	33	0.267	
67	335	-2619.78	0.04	0.00	35	0.213	
68	340	-2619.78	0.04	0.00	35	0.201	
69	345	-2619.77	0.04	0.00	36	0.181	
70	350	-2619.74	0.04	0.01	39	0.308	
71	355	-2619.68	0.05	0.03	41	0.748	
72	360	-2619.66	0.04	0.01	41	0.294	stable
73	365	-2619.65	0.04	0.00	41	0.193	stable
74	370	-2619.65	0.04	0.00	41	0.162	stable
75	375	-2619.63	0.04	0.00	41	0.241	stable
76	380	-2619.62	0.03	0.00	41	0.186	stable
77	385	-2619.61	0.03	0.00	42	0.158	stable
78	390	-2619.60	0.03	0.00	45	0.177	stable
79	395	-2619.60	0.03	0.00	44	0.173	stable
80	400	-2619.59	0.03	0.00	43	0.184	stable
81	405	-2619.59	0.03	0.00	42	0.156	stable
82	410	-2619.59	0.03	0.00	42	0.148	stable
83	415	-2619.59	0.03	0.00	42	0.103	stable
84	420	-2619.58	0.03	0.00	42	0.148	stable
85	425	-2619.57	0.02	0.00	42	0.129	stable
86	430	-2619.57	0.02	0.00	42	0.099	stable
87	435	-2619.58	0.02	0.00	42	0.139	stable
88	440	-2619.58	0.02	0.00	42	0.101	stable
89	445	-2619.57	0.02	0.00	42	0.106	stable
90	450	-2619.58	0.02	0.00	42	[illegible]	stable

Experiments

Figure: **Red indicate the centers of the variational mixture components, black are chosen training samples**



Experiments

Figure: **Red indicate the centers of the variational mixture components, black are chosen training samples**

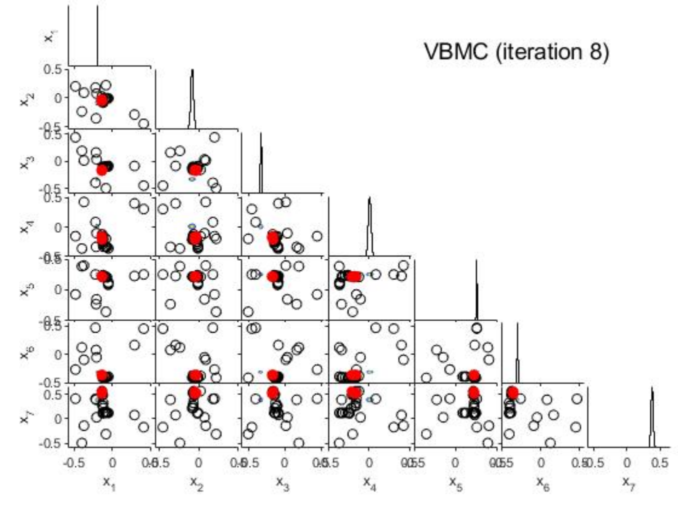


Figure: **Red indicate the centers of the variational mixture components, black are chosen training samples**

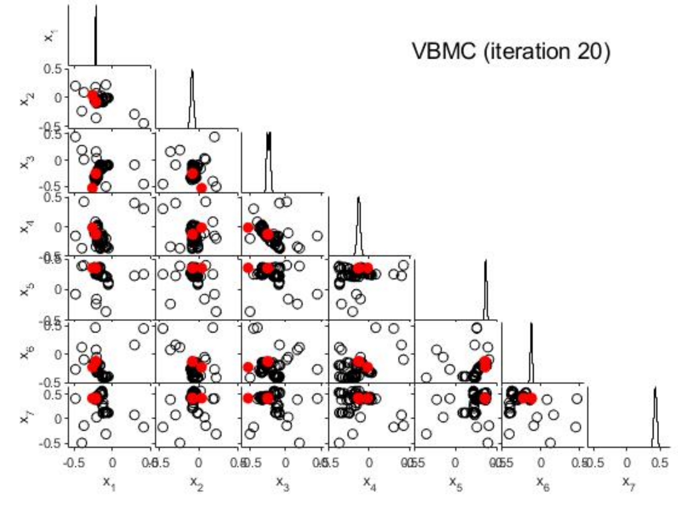
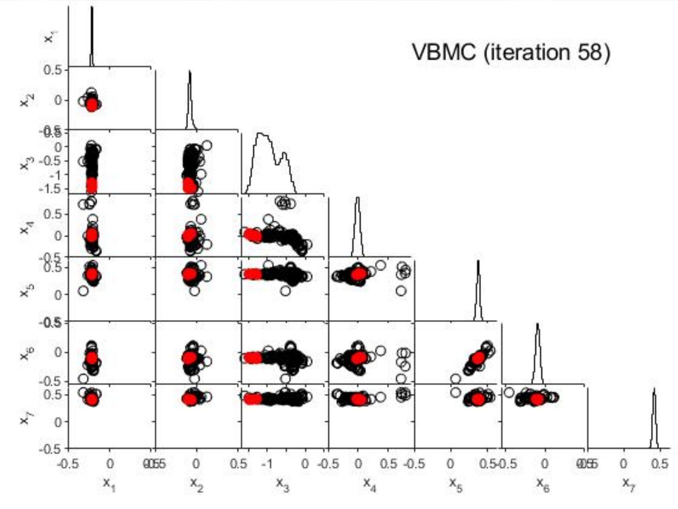
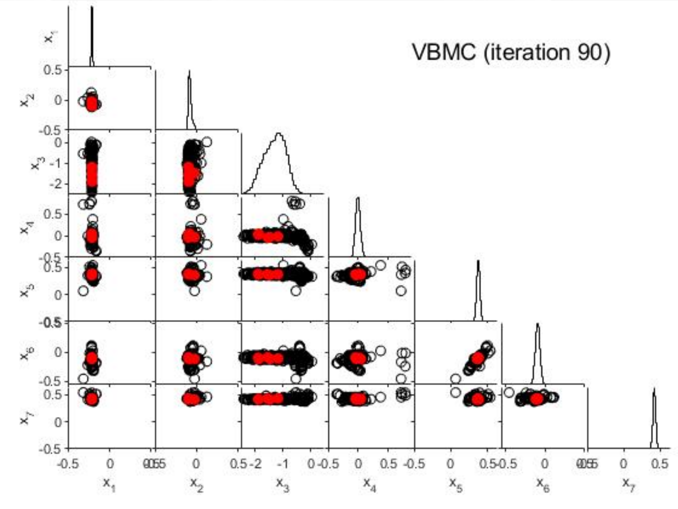


Figure: **Red indicate the centers of the variational mixture components, black are chosen training samples**



Experiments

Figure: **Red indicate the centers of the variational mixture components, black are chosen training samples**



Correlation b/w VBMC and PG-based RL

1. **VBMC Objective**

$$max_{\phi} F(q(\phi)) = \mathbf{E}_{q_{\phi}(\mathbf{x})}[log(P(\mathbf{D}/\mathbf{x}) P(\mathbf{x}) - log(q_{\phi}(\mathbf{x})))]$$

2. **VBMC Rough Algorithm Sketch**

   (a) Initialise ϕ, the parameters of $q_{\phi}(x)$.

   (b) Exploration-Exploitation using some intuitive acquisition function so as to maximise the above discussed objective: actively sample from training examples.

   (c) Given these new actively sampled points, build the posterior of objective using the Bayesian Quadrature framework.

   (d) Use gradient methods to maximise this new posterior objective with respect to the parameters set ϕ

   (e) if not converged, go to step (b)

1. **Policy Gradient based RL objective**

$$max_{\theta}\, U(\theta) \approx \mathbf{E}_{\tau \sim P(\tau;\theta)}[\sum_{t=1}^{T} log(\pi_{\theta}(a_t|s_t) R(\tau))]$$

2. **Vanila Policy Gradient Algorithm Rough Sketch**

   (a) Initialise θ, the parameters of π_{θ}

   (b) Exploitation-Exploration : Sample trajectories $\{\tau_n = \{s_t^n, a_t^n\}_{t=1}^{T}\}_{n=1}^{N}$ using the current policy $\pi_{\theta}(a^t|s^t)$

   (c) Given these trajectories, build the objective U(θ) using Monte Carlo Averaging.

   (d) Use gradient methods to maximise this new objective with respect to the parameter set θ, $(\theta = \theta + \alpha * \nabla_{\theta} U(\theta))$

   (e) if not converged, go to step (b)

Natural Gradient instead of Gradient

Gradient Ascent.

- Objective :

$$\phi_{new} = \phi_{old} + d^*$$

$$d^* = \underset{d \text{ s.t. } ||d|| \leq \epsilon}{ArgMax} \; F(\phi + d)$$

Natural Gradient Ascent.

- : Objective :

$$\phi_{new} = \phi_{old} + d^*$$

$$d^* = \underset{d \text{ s.t. } KL(q_\phi(\mathbf{x})||q_{\phi+d}(\mathbf{x})) \leq \epsilon}{ArgMax} \; F(\phi+d)$$

Natural Gradient instead of Gradient

Gradient Ascent.

- Objective :

$$\phi_{new} = \phi_{old} + d^*$$

$$d^* = \underset{d \text{ s.t. } ||d|| \leq \epsilon}{ArgMax} \quad F(\phi + d)$$

- Update Equation:

$$\phi_{new} = \phi_{old} + \alpha * \mathbf{g}$$

$$\mathbf{g} = \nabla_\phi F(\phi)|_{\phi_{old}}$$

$$\alpha = \textit{manually set}$$

Natural Gradient Ascent.

- : Objective :

$$\phi_{new} = \phi_{old} + d^*$$

$$d^* = \underset{d \text{ s.t. } KL(q_\phi(\mathbf{x})||q_{\phi+d}(\mathbf{x})) \leq \epsilon}{ArgMax} \quad F(\phi+d)$$

- Update Equation :

$$\phi_{new} = \phi_{old} + \alpha_N * \mathbf{g}_N$$

$$\mathbf{F}_I(\phi) = \mathbf{E}_{x \sim q_{\phi_{old}}}[\nabla_\phi log q_\phi(\mathbf{x})|_{\phi_{old}} (\nabla_\phi log q_\phi(\mathbf{x})|_{\phi_{old}})^T]$$

$$\mathbf{g}_N = \mathbf{F}_I^{-1}(\phi_{old}) \nabla_\phi F(\phi)|_{\phi_{old}}$$

$$\alpha_N = \sqrt{\frac{2 * \epsilon}{\mathbf{g}_N^T \mathbf{F}_I^{-1} \mathbf{g}_N}}$$

Natural Gradient instead of Gradient

Gradient Ascent.

- Objective :

$$\phi_{new} = \phi_{old} + d^*$$

$$d^* = \underset{d \text{ s.t. } ||d|| \leq \epsilon}{ArgMax} \; F(\phi + d)$$

- Update Equation:

$$\phi_{new} = \phi_{old} + \alpha * \mathbf{g}$$

$$\mathbf{g} = \nabla_\phi F(\phi)|_{\phi_{old}}$$

$$\alpha = \textit{manually set}$$

- Does not take into account the resulting distance between the old posterior and the newly built one.

Natural Gradient Ascent.

- : Objective :

$$\phi_{new} = \phi_{old} + d^*$$

$$d^* = \underset{d \text{ s.t. } KL(q_\phi(\mathbf{x})||q_{\phi+d}(\mathbf{x})) \leq \epsilon}{ArgMax} \; F(\phi + d)$$

- Update Equation :

$$\phi_{new} = \phi_{old} + \alpha_N * \mathbf{g}_N$$

$$\mathbf{F}_I(\phi) = \mathbf{E}_{x \sim q_{\phi_{old}}}[\nabla_\phi log q_\phi(\mathbf{x})|_{\phi_{old}} (\nabla_\phi log q_\phi(\mathbf{x})|_{\phi_{old}})^T]$$

$$\mathbf{g}_N = \mathbf{F}_I^{-1}(\phi_{old}) \nabla_\phi F(\phi)|_{\phi_{old}}$$

$$\alpha_N = \sqrt{\frac{2 * \epsilon}{\mathbf{g}_N^T \mathbf{F}_I^{-1} \mathbf{g}_N}}$$

- Takes into account this distance between the old posterior and the newly built one using ϵ.

Discussions

- Approximate bayesian inference framework that works even with noisy observations.
- VBMC has state-of-the-art inference performance.
- Sample-efficient so reduction in carbon footprint of environment.
- **Application Areas :** Computational Biology, Cognitive Neuroscience, Environmental Science

Discussions

- Approximate bayesian inference framework that works even with noisy observations.
- VBMC has state-of-the-art inference performance.
- Sample-efficient so reduction in carbon footprint of environment.
- **Application Areas :** Computational Biology, Cognitive Neuroscience, Environmental Science

Future Directions

- Account for non-stationarity and model mismatch
- Alternate GP representations
- Theoretical aspects of VBMC

Bibiliography

(1) Blog : https://wiseodd.github.io/techblog/2018/03/14/natural-gradient.
(2) Luigi Acerbi. Variational bayesian monte carlo, 2018.
(3) Luigi Acerbi. An exploration of acquisition and mean functions in variational bayesian montecarlo. volume 96 ofProceedings of Machine Learning Research. PMLR, 2019.
(4) Luigi Acerbi. Variational bayesian monte carlo with noisy likelihoods, 2020.