Vprop: Variational Inference using RMSprop

Mohammad Emtiyaz Khan[1], Zuozhu Liu[2], Voot Tangkaratt[1] and Yarin Gal[3]

[1]Center for Advanced Intelligence Project, RIKEN, Tokyo, Japan
[2]Singapore University of Technology and Design, Singapore
[3]The University of Oxford, UK

Introduction

Issues:

- Existing variational inference (VI) methods, e.g., black-box VI (BBVI) [2], require significant changes to existing deep-learning implementation and demand large memory.

Contributions:

- For Gaussian variational approximations, we propose a method called Vprop which can be implemented with a slight modification to RMSprop code.
- Vprop simplifies the implementation of BBVI and also reduces its memory requirement by half.
- Vprop is a natural-gradient method for VI and also related to Newton's method.

Algorithm 1 Vprop-1 for VI

1: $\boldsymbol{\theta} \leftarrow \boldsymbol{\mu} + \boldsymbol{\epsilon}./\sqrt{\mathbf{s}+\lambda}$, $\boldsymbol{\epsilon} \sim \mathcal{N}(\boldsymbol{\epsilon}|0, \mathbf{I})$
2: $\mathbf{g} \leftarrow \nabla_\theta f(\boldsymbol{\theta})$
3: $\mathbf{s} \leftarrow (1-\beta)\mathbf{s} + \beta(\mathbf{g}.*\mathbf{g})$
4: $\boldsymbol{\mu} \leftarrow \boldsymbol{\mu} - \beta((\mathbf{g}+\lambda\boldsymbol{\mu})./(\mathbf{s}+\lambda))$

(a) Vprop-1 minimizes $\mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\sigma}) = \mathbb{E}_q\left[f(\boldsymbol{\theta}) + \log p(\boldsymbol{\theta})/q(\boldsymbol{\theta})\right]$.

Algorithm 2 RMSprop for MLE

1: $\boldsymbol{\theta} \leftarrow \boldsymbol{\mu}$
2: $\mathbf{g} \leftarrow \nabla_\theta f(\boldsymbol{\theta})$
3: $\mathbf{s} \leftarrow (1-\beta)\mathbf{s} + \beta(\mathbf{g}.*\mathbf{g})$
4: $\boldsymbol{\mu} \leftarrow \boldsymbol{\mu} - \alpha(\mathbf{g}./\sqrt{\mathbf{s}+\delta})$

(b) RMSprop minimizes $f(\boldsymbol{\theta}) := -\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$.

Optimization Algorithms & VI

Maximum Likelihod Estimation (MLE):
We can perform MLE by minimizing $f(\boldsymbol{\theta}) = -\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta})$ using RMSprop [1], as shown in Algorithm 2:

$$\text{RMSProp:}\quad \begin{aligned} \boldsymbol{\theta}_{t+1} &= \boldsymbol{\theta}_t - \alpha_t \left(\mathbf{s}_{t+1} + \delta\mathbf{1}\right)^{-\frac{1}{2}} \circ \nabla_\theta f(\boldsymbol{\theta}_t), \\ \mathbf{s}_{t+1} &= (1-\beta_t)\mathbf{s}_t + \beta_t\left(\nabla_\theta f(\boldsymbol{\theta}_t)\right)^2, \end{aligned}$$

where $\mathbf{s}$ is the adaptive step-size scaling vector.

Gaussian Variational Inference:
In contrast, in Gaussian VI, we approximate the posterior distribution with a Gaussian distribution $q(\boldsymbol{\theta}) := \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}, \text{diag}(\boldsymbol{\sigma}^2))$ by maximizing a variational lower-bound:

$$\log p(\mathbf{y}|\mathbf{X}) = \max_{\boldsymbol{\mu},\boldsymbol{\sigma}} \mathbb{E}_q\left[\log p(\mathbf{y}|\mathbf{X}, \boldsymbol{\theta}) + \log p(\boldsymbol{\theta}) - \log q(\boldsymbol{\theta})\right] := \mathcal{L}(\boldsymbol{\mu}, \boldsymbol{\sigma})$$

where $p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}|0, \mathbf{I}/\lambda)$ is the Gaussian prior.

Black-box Variational Inference:
Black-box VI (BBVI) [2] optimizes $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ by the following stochastic gradient-descent algorithm (or adaptive-gradient variants of it):

$$\text{BBVI:}\quad \begin{aligned} \boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t + \eta_t\left[\widehat{\nabla}_\mu \mathcal{L}_t\right], \\ \boldsymbol{\sigma}_{t+1} &= \boldsymbol{\sigma}_t + \rho_t\left[\widehat{\nabla}_\sigma \mathcal{L}_t\right]. \end{aligned}$$

However, BBVI update has two issues.

- **Memory:** The number of parameters is doubled for adaptive-gradient since it needs to store scaling vectors for $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$.
- **Implementation:** Computing the gradients w.r.t. $\boldsymbol{\sigma}$ requires a different implementation from that of existing deep learning.

Conjugate-computation Variational Inference:
We propose Vprop to solve these issues by using a natural gradient method called conjugate-computation VI (CVI) [3]:

$$\text{CVI:}\quad \begin{aligned} \boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t + \beta_t\, \boldsymbol{\sigma}^2_{t+1} \circ \left[\nabla_\mu \mathcal{L}_t\right], \\ \boldsymbol{\sigma}^{-2}_{t+1} &= \boldsymbol{\sigma}^{-2}_t - \ 2\beta_t\left[\nabla_{\sigma^2}\mathcal{L}_t\right]. \end{aligned}$$

Vprop

Derivation of Vprop:
Vprop is derived from the CVI update in two steps.

- Apply Bonnet's and Price's theorem:

$$\begin{aligned} \nabla_\mu \mathcal{L} &= -\mathbb{E}_q\left[\nabla_\theta f(\boldsymbol{\theta})\right] - \lambda\boldsymbol{\mu}, \\ \nabla_\sigma \mathcal{L} &= -\frac{1}{2}\mathbb{E}_q\big[\underbrace{\text{diag}\left(\nabla^2_\theta f(\boldsymbol{\theta})\right)}_{(\nabla_\theta f(\boldsymbol{\theta}))^2}\big] - \frac{1}{2}\lambda\mathbf{1} + \frac{1}{2}\boldsymbol{\sigma}^{-2}. \end{aligned}$$

The Hessian is approximated by a Gauss-Newton approximation.

By defining $\mathbf{s}_t := \boldsymbol{\sigma}_t^{-2} - \lambda\mathbf{1}$ and using the above two steps in the CVI update, we obtain Vprop:

$$\text{Vprop:}\quad \begin{aligned} \boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t - \beta_t\left(\mathbf{s}_{t+1} + \lambda\mathbf{1}\right)^{-1} \circ \left\{\mathbb{E}_{q_t}\left[\nabla_\theta f(\boldsymbol{\theta})\right] + \lambda\boldsymbol{\mu}_t\right\}, \\ \mathbf{s}_{t+1} &= (1-\beta_t)\mathbf{s}_t + \beta_t\, \mathbb{E}_{q_t}\left[\left(\nabla_\theta f(\boldsymbol{\theta})\right)^2\right]. \end{aligned}$$

The expectations can be approximated by using one Monte-Carlo (MC) sample giving us Vprop-1:

$$\text{Vprop-1:}\quad \begin{aligned} \boldsymbol{\theta}_t &\sim \mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}_t, \text{diag}(\mathbf{s}_t^{-2})), \\ \boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t - \beta_t\left(\mathbf{s}_{t+1} + \lambda\mathbf{1}\right)^{-1} \circ \left\{\nabla_\theta f(\boldsymbol{\theta}_t) + \lambda\boldsymbol{\mu}_t\right\}, \\ \mathbf{s}_{t+1} &= (1-\beta_t)\mathbf{s}_t + \beta_t\left(\nabla_\theta f(\boldsymbol{\theta}_t)\right)^2. \end{aligned}$$

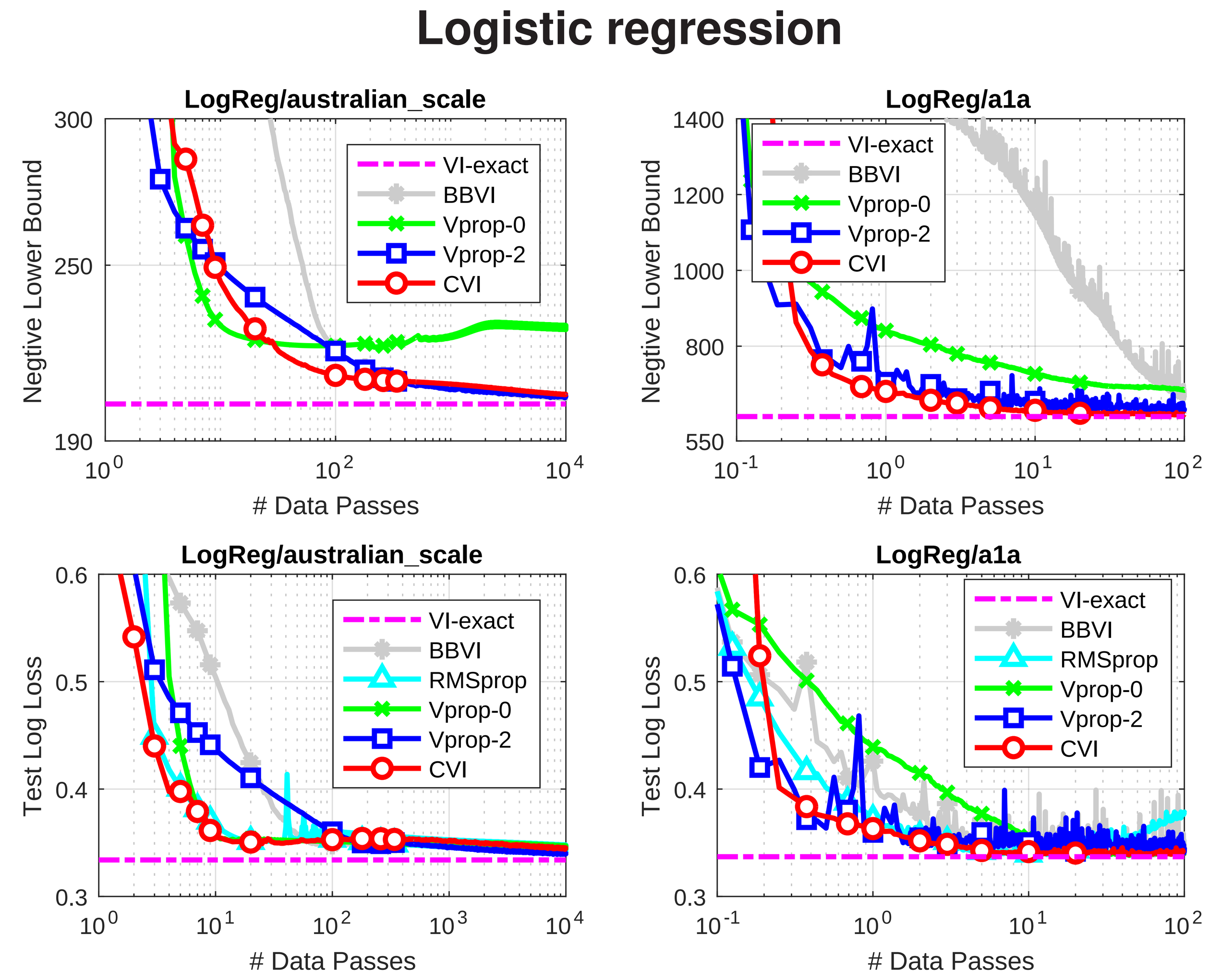
Comparison between Vprop-1 and RMSprop:
Vprop-1 and RMSprop are similar but have 3 significant differences:

- Vprop-1 computes the gradients at $\boldsymbol{\theta}$ sampled from $\mathcal{N}(\boldsymbol{\theta}|\boldsymbol{\mu}_t, \text{diag}(\mathbf{s}_t^{-2}))$ but RMSprop computes gradients at $\boldsymbol{\theta} = \boldsymbol{\mu}_t$.
- RMSprop raise the scaling vector to a power of $\frac{1}{2}$, while Vprop-1 does not.
- Vprop-1 adds $\lambda\boldsymbol{\mu}$ to the gradient of $\boldsymbol{\mu}$ update.

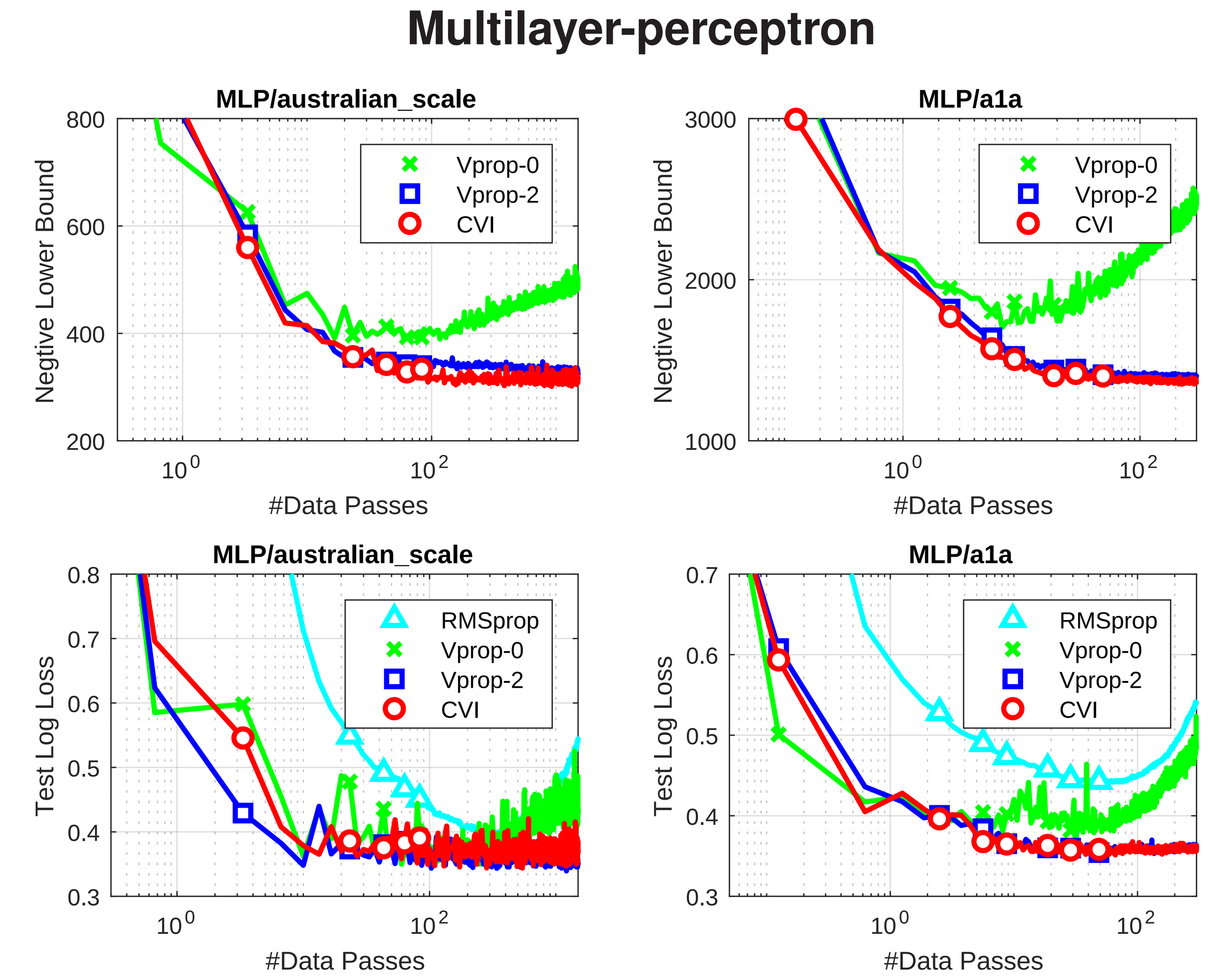
Connections to natural gradient and Newton's method:
For full Gaussian $q(\boldsymbol{\theta})$, using similar derivations result in a variant of Newton's method. The resulting update also resembles an online natural gradient descent proposed by Ollivier 2017 [4].

Experiments

Logistic regression



Multilayer-perceptron



References

[1] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural Networks for Machine Learning: Lecture 6a.
[2] Rajesh Ranganath, Sean Gerrish, and David M Blei. Black box variational inference. *AIStats*, 2014.
[3] Mohammad Emtiyaz Khan and Wu Lin. Conjugate-computation variational inference. *AIStats*, 2017.
[4] Yann Ollivier. Online Natural Gradient as a Kalman Filter. *arXiv*, 2017.