Approximate Bayesian Inference Team

Mohammad Emtiyaz Khan

Goals and Challenges

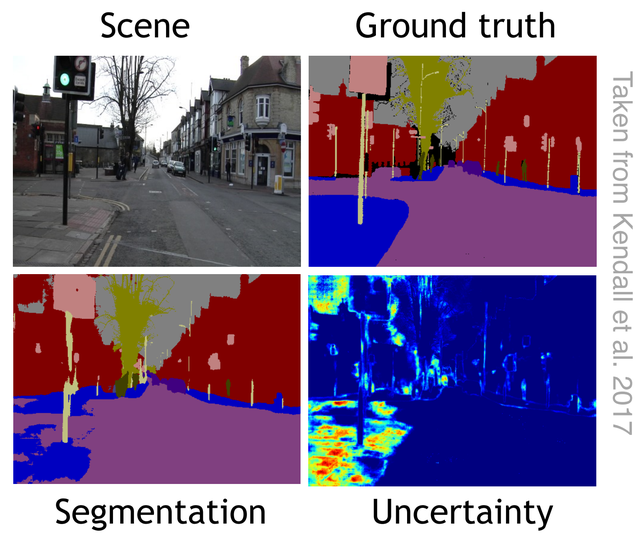
Goal: To design AI that can continually learn using Bayesian principles.

Examples: Uncertainty: Knowing how much we don't know, is useful to design

- Robots that can understand and reason about their environments.
- Methods that improve performance of deep-learning methods.

Challenge: Computation of the posterior distribution is difficult

Main Idea: Approximate integration by using optimization, and design simple algorithms that can be implemented within existing deep learning frameworks



Fast and Simple Algorithms for Variational Inference

Variational Inference

$$p(\theta|\mathcal{D}) = \frac{p(\mathcal{D}|\theta)p(\theta)}{\int p(\mathcal{D}|\theta)p(\theta)d\theta}$$

Parameters; Data; Intractable integral

$$\approx q_\lambda(\theta) = \text{ExpFamily}(\lambda)$$

Variational Approximation; Natural parameters

Maximize the Evidence Lower Bound (ELBO):

$$\max_\lambda \mathcal{L}(\lambda) := \mathbb{E}_{q_\lambda}\Big[\log p(\mathcal{D},\theta) - \log q_\lambda(\theta)\Big]$$

Gradient descent (GD) : $\lambda \leftarrow \lambda + \rho\nabla_\lambda\mathcal{L}$

VI with Natural-Gradient Descent

Sato 2001, Honkela et al. 2010, Hoffman et.al. 2013

NGD: $\lambda \leftarrow \lambda + \rho F(\lambda)^{-1}\nabla_\lambda\mathcal{L}$ Natural Gradient

Fisher Information Matrix (FIM)

$$F(\lambda) := \mathbb{E}_{q_\lambda}\Big[\nabla \log q_\lambda(\theta) \nabla \log q_\lambda(\theta)^\top\Big]$$

- Fast convergence due to optimization in Riemannian manifold (not Euclidean space).
- But requires additional computations.
- Can we simplify/reduce this computation?

Expectation Parameters

Expectation/moment/mean parameters $\mu := \mathbb{E}_{q_\lambda}[\phi(\theta)]$ Sufficient statistics

For Gaussians, it's mean and correlation matrix

$$\mathbb{E}_{q_\lambda}[\theta] = m \qquad \mathbb{E}_{q_\lambda}[\theta\theta^\top] = mm^\top + V$$

A key relationship: $F(\lambda)^{-1}\nabla_\lambda\mathcal{L} = \nabla_\mu\mathcal{L}$

Natural Gradient wrt natural parameter; Gradient wrt expectation parameter

NGD : $\lambda \leftarrow \lambda + \rho\nabla_\mu\mathcal{L}$

Example: Linear Regression

$$q_\lambda(\theta) := \mathcal{N}(m, V)$$

$$\mathbb{E}_q\Big(\underbrace{(y - X\theta)^\top(y - X\theta)}_{-\mathbb{E}_{q_\lambda}[\theta]^\top X^\top y + \text{trace}\left[X^\top X \mathbb{E}_{q_\lambda}[\theta\theta^\top]\right]} + \gamma\theta^\top\theta \quad - \log q_\lambda(\theta)\Big)$$

likelihood; prior; approx

$$\nabla_{\mathbb{E}_{q_\lambda}[\theta]} = -X^\top y + 0 - V^{-1}m$$

$$\nabla_{\mathbb{E}_{q_\lambda}[\theta\theta^\top]} = X^\top X + \gamma I - V^{-1}$$

$$m \leftarrow (1-\rho)m - \rho\big[X^\top X + \gamma I\big]^{-1}X^\top y$$

Bayesian Neural Network

$$\mathbb{E}_q\Big(\sum_{i=1}^N \log p(y_i|f_\theta(x_i)) + \gamma\theta^\top\theta \quad - \log q_\lambda(\theta)\Big)$$

likelihood; neural network; prior; approx

$$m \leftarrow m - \beta(S + \gamma I)^{-1}[g_i(\theta) + \gamma m]$$
$$S \leftarrow (1-\beta)S + \beta H_i(\theta)$$

Back-propagated gradient & Hessian

$$\theta \sim q_\lambda(\theta), \qquad g_i(\theta) := -\nabla_\theta \log p(y_i|f_\theta(x_i)),$$
$$V^{-1} \leftarrow S + \gamma I, \qquad H_i(\theta) := -\nabla^2_\theta \log p(y_i|f_\theta(x_i))$$

MLE vs NGD-VI

RMSprop for MLE

$\theta \leftarrow \mu$
$g \leftarrow \frac{1}{M}\sum_i \nabla_\theta \log p(\mathcal{D}_i|\theta)$
$s \leftarrow (1-\beta)s + \beta g^2$
$\mu \leftarrow \mu + \alpha\,\frac{g}{\sqrt{s}+\delta}$

NGD for mean-field VI

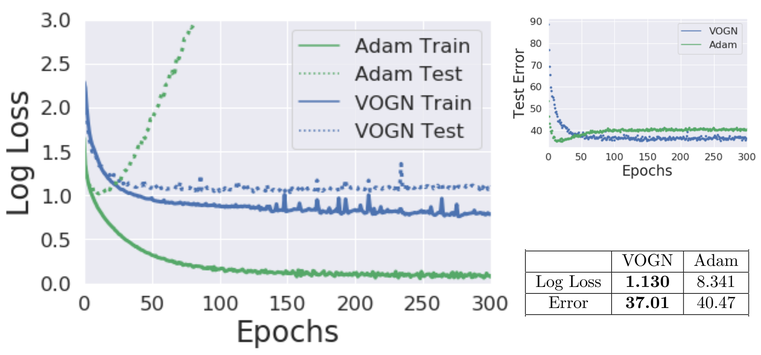
$\theta \leftarrow \mu + \epsilon$, where $\epsilon \sim \mathcal{N}(0, Ns + \lambda)$
$g \leftarrow \frac{1}{M}\sum_i \nabla_\theta \log p(\mathcal{D}_i|\theta)$
$s \leftarrow (1-\beta)s + \beta\frac{1}{M}\sum_i \Big[\nabla_\theta \log p(\mathcal{D}_i|\theta)\Big]^2$
$\mu \leftarrow \mu + \alpha\,\frac{g + \lambda\mu/N}{s + \lambda/N}$

Variational Online Gauss-Newton (VOGN)

$s \leftarrow (1-\beta)s + \beta\frac{1}{M}\sum_i \nabla^2_{\theta\theta} \log p(\mathcal{D}_i|\theta)$

Variational RMSprop (Vprop) $s \leftarrow (1-\beta)s + \beta g^2$

LeNet-5 on CIFAR10



	VOGN	Adam
Log Loss	**1.130**	8.341
Error	**37.01**	40.47

(By Anirudh Jain)

Stochastic, Low-Rank, Approximate, Natural-Gradient (SLANG)

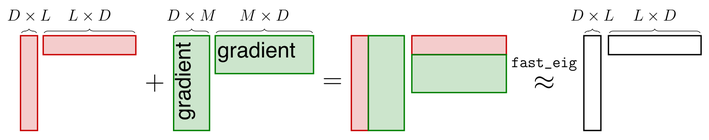
NeurIPS 2018

- Low-rank + diagonal covariance matrix.
- SLANG is linear in D!

Low-Rank + diagonal

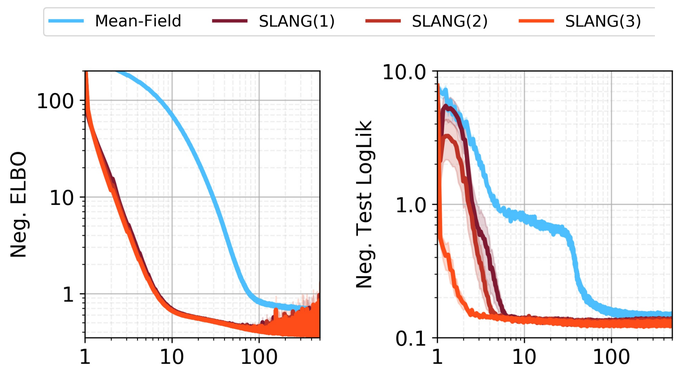
$$m \leftarrow m - \rho\big[UU^\top + D\big]^{-1}[g_i + \gamma m]$$

$$(1-\beta)S + \beta H_i(\theta)$$



SLANG is Faster than GD

Classification on USPS with BNNs



References

1. Fast and Scalable Bayesian Deep Learning by Weight-Perturbation in Adam, (ICML 2018), Khan, Nielsen, Tangkaratt, Lin, Gal, and Srivastava.
2. SLANG: Fast Structured Covariance Approximations for Bayesian Deep Learning with Natural Gradient, (NeurIPS 2018), Mishkin, Kunstner, Nielsen, Schmidt, Khan.
3. Fast and Simple Natural-Gradient Descent for Variational Inference in Complex Models (ISITA 2018), Khan and Nielsen.