Variational Stochastic Gradient Descent for Deep Neural Networks

Haotian Chen*, Anna Kuzina*, Babak Esmaeili, Jakub M. Tomczak

VU Vrije Universiteit Amsterdam · TU/e Eindhoven University of Technology · Hybrid Intelligence

#AI #WANT #HPC @ ICML 2024

Summary

- We propose VSGD: a novel optimizer that adopts a probabilistic approach. We model the true gradient and the noisy gradient as latent and observed random variables
- We draw connections between VSGD and several established non-probabilistic optimizers.
- We carry out an empirical evaluation of VSGD by comparing its performance against the most popular optimizers

SGD

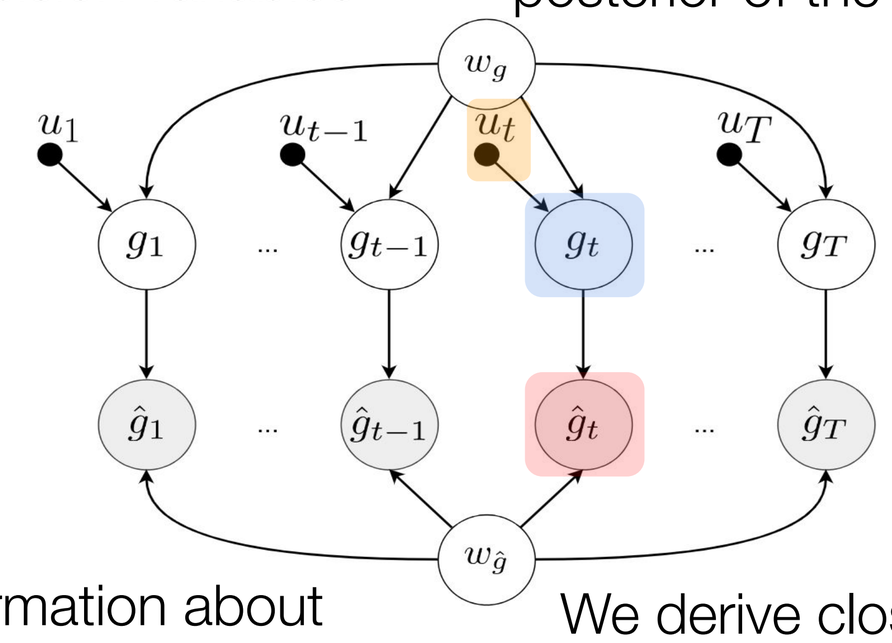
Update model parameters using noisy gradient of the loss function and step size defined by learning rate

$$\theta_t = \theta_{t-1} - \eta_t \hat{g}_t$$

Probabilistic Model

- We model **noisy gradient** (observed) and **true gradient** (latent) with gaussians
- We use Gamma prior over precision variables

$$p(g_t|w_g; u_t) = \mathcal{N}(u_t, w_g^{-1}),$$
$$p(\hat{g}_t|g_t, w_{\hat{g}}) = \mathcal{N}(g_t, w_{\hat{g}}^{-1}),$$
$$p(w_g) = \Gamma(\gamma, \gamma),$$
$$p(w_{\hat{g}}) = \Gamma(\gamma, K_g\gamma),$$



Control variate aggregates information about previously observed noisy gradients and serves as a mean for the true gradient

$$u_t = \mathbb{E}_{p(g_t|\hat{g}_{t-1}; u_{t-1})}[g_t],$$

Posterior Inference

We employ stochastic variational inference with mean field assumption to approximate posterior of the unobserved variables

$$q(w_g) = \Gamma(a_g, b_g),$$
$$q(w_{\hat{g}}) = \Gamma(a_{\hat{g}}, b_{\hat{g}}),$$
$$q(g_t) = \mathcal{N}(\mu_{t,g}, \sigma^2_{t,g}),$$

We derive closed-form updates for global and local variational parameters and scale the step size using the second moment estimate:

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\mu^2_{t,g} + \sigma^2_{t,g}}}\mu_{t,g},$$

Results

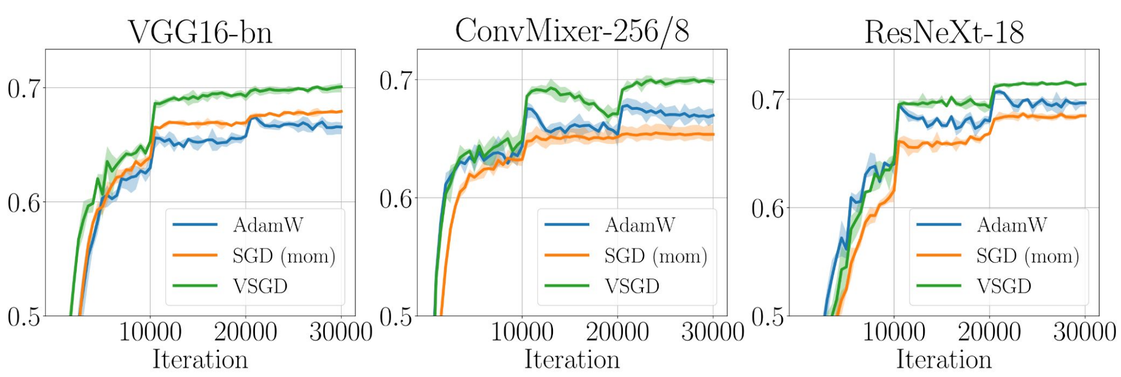


Table 1. Final Average test accuracy, over three random seeds.

	VSGD (w/ L2)	VSGD (w/o L2)	ADAM (w/o L2)	ADAMW (w/ L2)	SGD (w/ mom)
			CIFAR100		
VGG16	**70.1**	70.0	66.8	66.6	67.9
CONVMIXER	**69.8**	69.1	66.5	67.0	65.4
RESNEXT-18	**71.4**	71.2	68.2	69.7	68.5
			TINYIMAGENET-200		
VGG19	51.2	**52.0**	47.6	49.0	50.9
CONVMIXER	**53.1**	52.6	51.9	52.4	52.4
RESNEXT-18	48.7	47.2	48.8	**48.9**	47.0

VSGD almost always converges to a better solution compared to ADAM and SGD, outperforming ADAM by an average of 2.6% for CIFAR100 and 0.9% for TINY IMAGENET-200.

Constant VSGD

A simplified model assumes constant variance ratio between **noisy** and **true** gradient

$$p(g_t|\omega; u_t) = \mathcal{N}(u_t, K_g^{-1}\omega^{-1}),$$
$$p(\hat{g}_t|g_t, \omega) = \mathcal{N}(g_t, \omega^{-1}),$$
$$p(\omega) = \Gamma(\gamma, \gamma).$$

The first and second gradient momentum allows us to draw connection to Adam, SGD with momentum, and AmsGrad

$$\mu_{t,g} = \mu_{t-1,g}\frac{K_g}{K_g+1} + \hat{g}_t\frac{1}{K_g+1},$$

$$\mathbb{E}[g_t^2] = \mu^2_{t-1,g}\frac{K_g^2}{(K_g+1)^2} + \hat{g}_t^2\frac{1}{(K_g+1)^2} + \frac{2K_g}{(K_g+1)^2}\mu_{t-1,g}\hat{g}_t + \frac{1}{K_g+1}\frac{b_{t-1,\hat{g}}}{a_{t-1}}.$$

Variational Stochastic Gradient Descent for Deep Neural Networks

Haotian Chen*, Anna Kuzina*, Babak Esmaeili, Jakub M. Tomczak

#AI #WANT #HPC @ ICML 2024

Summary

- We propose VSGD: a novel optimizer that adopts a probabilistic approach.
- In VSGD, we model the true gradient and the noisy gradient as latent and observed random variables, respectively, within a probabilistic model.
- We draw connections between VSGD and several established non-probabilistic optimizers.
- We carry out an empirical evaluation of VSGD by comparing its performance against the

Variational Stochastic Gradient Descent

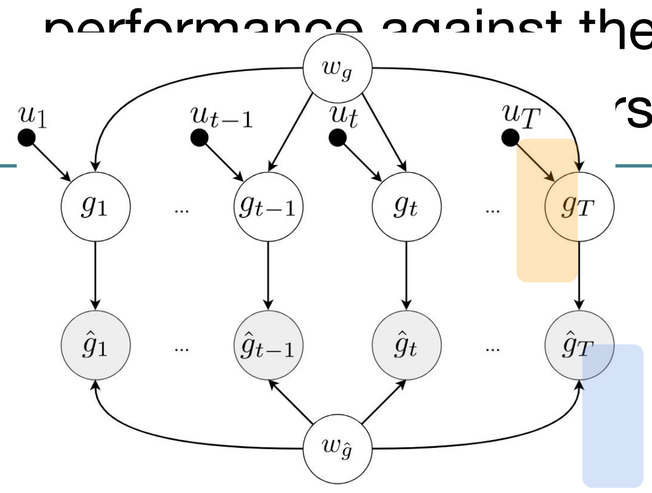
Model

- We model **noisy gradient** (observed) and **true gradient** (latent with gaussians)
- We use Gamma prior over precision variables
- **Control variate** aggregates information about previously observed noisy gradients and serves as a mean for the true gradie

$$u_t = \mathbb{E}_{p(g_t|\hat{g}_{t-1};u_{t-1})}[g_t],$$

Stochastic Variational Inference

$$q(w_g) = \Gamma(a_g, b_g),$$
$$q(w_{\hat{g}}) = \Gamma(a_{\hat{g}}, b_{\hat{g}}),$$
$$q(g_t) = \mathcal{N}(\mu_{t,g}, \sigma^2_{t,g}),$$



$$p(g_t|w_g;u_t) = \mathcal{N}(u_t, w_g^{-1}),$$
$$p(\hat{g}_t|g_t, w_{\hat{g}}) = \mathcal{N}(g_t, w_{\hat{g}}^{-1}),$$
$$p(w_g) = \Gamma(\gamma, \gamma),$$
$$p(w_{\hat{g}}) = \Gamma(\gamma, K_g\gamma),$$

Constant VSGD and Adam

Results

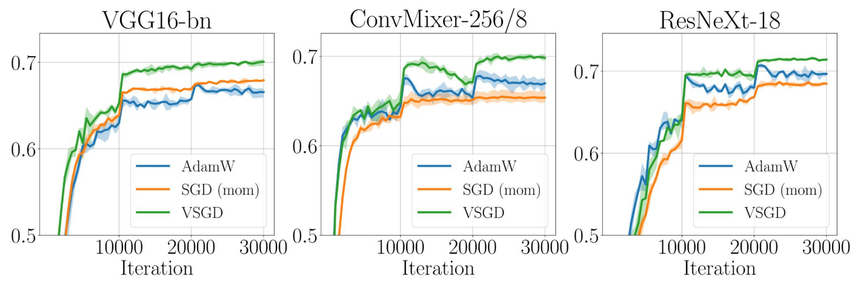


Table 1. Final Average test accuracy, over three random seeds.

	VSGD (w/ L2)	VSGD (w/o L2)	ADAM (w/o L2)	ADAMW (w/ L2)	SGD (w/ mom)
	CIFAR100				
VGG16	**70.1**	70.0	66.8	66.6	67.9
CONVMIXER	**69.8**	69.1	66.5	67.0	65.4
RESNEXT-18	**71.4**	71.2	68.2	69.7	68.5
	TINYIMAGENET-200				
VGG19	51.2	**52.0**	47.6	49.0	50.9
CONVMIXER	**53.1**	52.6	51.9	52.4	52.4
RESNEXT-18	48.7	47.2	48.8	**48.9**	47.0

We observe that VSGD almost always converges to a better solution compared to ADAM and SGD, outperforming ADAM by an average of 2.6% for CIFAR100 and 0.9% for TINYIMAGENET-200.

Ablation Studies

γ	Accuracy
1e-9	67.75
5e-9	68.59
1e-8	69.03
5e-8	69.77
1e-7	69.71
1e-3	60.97

Test accuracy on Clfar100