4U-05

自然勾配法に基づく変分深層学習

中田 光[1,†1,a)]　大沢 和樹[1,†1,b)]　横田 理央[1,†1,c)]

概要：深層学習は与えられた膨大なデータに対し柔軟な学習を可能にする一方，学習を汎化させ未知のデータに対しても精度を保つことが一つの大きな課題となる．近年では，ベイズ推定を深層学習に適用し，学習によって得られたニューラルネットワークの重みの不確かさを推定することにより学習を汎化させる試みが注目されつつある．Zhang らによって提案された Noisy K-FAC は，自然勾配法に基づく一種の変分推論を行うことによりベイズ推定を行う手法であり，学習が汎化することが示されている．本研究では Noisy K-FAC に着目し，重みの更新時に複数のサンプルを用いた場合の学習の変化ついて比較検証を行った．

1. はじめに

近年では，ベイズ推定を深層学習に適用し，ニューラルネットワークの重みの不確かさに相当する事後分布を推定することにより，学習を汎化させる試みが注目されつつある．Hoffman らは事後分布が指数型分布族に含まれる場合の確率的な変分推論と自然勾配法 (NGD;Natural Gradient Descent)[2] の関係性を示した [1]．Zhang らは自然勾配法の近似手法 K-FAC(Kronecker-Factored Approximate Curvature)[4], [5] を変分推論に適用させ，既存手法と比較してより複雑な事後分布の推定を可能とする手法 Noisy K-FAC[3] を提案した．本研究では Noisy K-FAC に着目し，事後分布の更新の際にモンテカルロ法 (MC;Monte Carlo method) に基づき，複数の重みのサンプルを用いた場合の学習の検証を行なった．検証は深層ニューラルネットワーク (DNN;Deep Neural Network) による画像分類を行い，K-FAC と比較して学習が汎化し，重みのサンプル数を増やすことにより分類精度が向上することを示した．

2. 背景

2.1 変分推論

データセットを $\mathcal{D}=\{(\mathbf{x}_i, y_i)_{i=1}^n\}$，ニューラルネットワークの持つ重みをパラメータ $\boldsymbol{\theta}$ として，ある入力 $\mathbf{x}$ に対する予測値 y の従う予測分布 $p(y|\mathbf{x},\mathcal{D})$ は，パラメータ $\boldsymbol{\theta}$ の事後分布 $p(\boldsymbol{\theta}\,|\,\mathcal{D})$ から，

$$p(y|\mathbf{x},\mathcal{D}) \;=\; \int p(y|\mathbf{x},\boldsymbol{\theta})p(\boldsymbol{\theta}\,|\,\mathcal{D})\,d\boldsymbol{\theta} \tag{1}$$

によって得られる．そのため，ベイズ推定では事後分布 $p(\boldsymbol{\theta}\,|\,\mathcal{D})$ の推定が目的となる．変分推論では事後分布 $p(\boldsymbol{\theta}\,|\,\mathcal{D})$ の近似分布 $q(\boldsymbol{\theta})$ を，パラメータ $\boldsymbol{\theta}$ の事前分布と尤度をそれぞれ $p(\boldsymbol{\theta})$，$p(\mathcal{D}\,|\,\boldsymbol{\theta})$ として，変分下限 $\mathcal{L}(q)$(ELBO;Evidence Lower Bound)，

$$\mathcal{L}(q) \;=\; \mathbb{E}_{\boldsymbol{\theta}\sim q}\left[\log p(\mathcal{D}\,|\,\boldsymbol{\theta})\right] - D_{KL}(q(\boldsymbol{\theta})||p(\boldsymbol{\theta}))] \tag{2}$$

の最大化問題により推定し，ベイズ推定を行う．ただし，D_{KL} は KL ダイバージェンス (Kullback-Leibler divergence) を表すものとする．

2.2 Noisy K-FAC

Noisy K-FAC では近似分布 $q(\boldsymbol{\theta})$ を平均 $\boldsymbol{\mu}$，共分散行列 $\boldsymbol{\Sigma}$ を持つ多変量正規分布 $\mathcal{N}(\boldsymbol{\mu},\boldsymbol{\Sigma})$ と仮定し，変分パラメータ $\boldsymbol{\mu}$，$\boldsymbol{\Sigma}$ に関する変分下限 $\mathcal{L}(q)$ の最大化問題に対し，自然勾配法の近似手法 K-FAC による二次最適化を行う．変分下限 $\mathcal{L}(q)$ に関する平均 $\boldsymbol{\mu}$，精度行列 $\boldsymbol{\Lambda}=\boldsymbol{\Sigma}^{-1}$ の自然勾配 $\tilde{\nabla}_{\boldsymbol{\mu}}\mathcal{L}$，$\tilde{\nabla}_{\boldsymbol{\Lambda}}\mathcal{L}$ は，

$$\tilde{\nabla}_{\boldsymbol{\mu}}\mathcal{L} \;=\; \boldsymbol{\Lambda}^{-1}\,\mathbb{E}_{\boldsymbol{\theta}\sim q}\left[\nabla_{\boldsymbol{\theta}}\log p(\mathcal{D}\,|\,\boldsymbol{\theta}) + \nabla_{\boldsymbol{\theta}}\log p(\boldsymbol{\theta})\right] \tag{3}$$

$$\tilde{\nabla}_{\boldsymbol{\Lambda}}\mathcal{L} \;=\; -\,\mathbb{E}_{\boldsymbol{\theta}\sim q}\left[\nabla^2_{\boldsymbol{\theta}}\log p(\mathcal{D}\,|\,\boldsymbol{\theta}) + \nabla^2_{\boldsymbol{\theta}}\log p(\boldsymbol{\theta})\right] - \boldsymbol{\Lambda} \tag{4}$$

となる．ここでヘッセ行列 $\nabla^2_{\boldsymbol{\theta}} - \log p(\mathcal{D}\,|\,\boldsymbol{\theta})$ を層ごとにブロック対角なフィッシャー情報行列 (FIM;Fisher Information Matrix) に近似し，K-FAC により各層のフィッシャー情報行列の近似計算を行う．Noisy K-FAC では事前分布 $p(\boldsymbol{\theta})$ を等方な正規分布 $\mathcal{N}(0,\eta\,\mathbf{I})$ と仮定し，パラメータ $\boldsymbol{\theta}\sim q_{\boldsymbol{\mu},\boldsymbol{\Sigma}}(\boldsymbol{\theta})$ をイテレーションごとにサンプルすることにより自然勾配 (式 3,4) の計算を行い，変分パラメータ $\boldsymbol{\mu}$，$\boldsymbol{\Lambda}$ を自然勾配に従って更新する．

3. 複数の重みのサンプルを用いた学習 (KFAC-VI)

本論文では複数の重みのサンプルを用いた学習を KFAC-VI と呼ぶ．KFAC-VI ではモンテカルロ法に基づき，Noisy K-FAC における変分パラメータの更新時に近似分布 $q_{\boldsymbol{\mu},\boldsymbol{\Sigma}}(\boldsymbol{\theta})$ に従ったパラメータ $\boldsymbol{\theta}$ のサンプルを複数サンプルし，(式 3,4) の近似分布 $q_{\boldsymbol{\mu},\boldsymbol{\Sigma}}(\boldsymbol{\theta})$ 上の期待値をそのサンプル内の平均によって近似する．本検証では，平均 $\boldsymbol{\mu}$ の更新にモメンタム (momentum) と重み減衰 (weight decay) を適用させた．l 層目の変分パラメータ $\boldsymbol{\mu}_l$，$\boldsymbol{\Sigma}_l$ は自然勾配 (式 3,4) に従い更新則，

$$\mathcal{D}_{\boldsymbol{\theta}} \;:=\; \nabla_{\boldsymbol{\theta}}\log p(y|\,\mathbf{x},\boldsymbol{\theta}) \tag{5}$$

$$m^{(t)} \;=\; \frac{\alpha^{(t)}}{\alpha^{(base)}}m^{(base)} \tag{6}$$

$$\bar{\mathbf{g}}_l \;\leftarrow\; -\frac{1}{|\,\mathcal{M}\,||\,\mathcal{B}\,|}\sum_{\boldsymbol{\theta}_l\in\mathcal{M}}\sum_{(\mathbf{x},y)\in\mathcal{B}}\mathcal{D}_{\boldsymbol{\theta}_l} \tag{7}$$

$$\bar{\mathbf{F}}_l \;\leftarrow\; \frac{1}{|\,\mathcal{M}\,||\,\mathcal{B}\,|}\sum_{\boldsymbol{\theta}_l\in\mathcal{M}}\sum_{(\mathbf{x},y)\in\mathcal{B}}\mathcal{D}_{\boldsymbol{\theta}_l}\,\mathcal{D}^{\mathrm{T}}_{\boldsymbol{\theta}_l} \tag{8}$$

$$\hat{\mathbf{F}}_l \;\leftarrow\; (1-\beta)\hat{\mathbf{F}}_l + \beta\bar{\mathbf{F}}_l \tag{9}$$

$$\mathbf{v}_l \;\leftarrow\; (\hat{\mathbf{F}}_l + \gamma\,\mathbf{I})^{-1}\bar{\mathbf{g}}_l + \frac{\lambda}{\alpha^{(t)}}\,\boldsymbol{\mu}_l + m^{(t)}\,\mathbf{v}_l \tag{10}$$

$$\boldsymbol{\mu}_l \;\leftarrow\; \boldsymbol{\mu}_l - \alpha^{(t)}\,\mathbf{v}_l \tag{11}$$

$$\boldsymbol{\Sigma}_l \;\leftarrow\; \rho^2(\hat{\mathbf{F}}_l + \gamma\,\mathbf{I})^{-1} \tag{12}$$

によって更新される．ただし，t は更新時のイテレーション，$\mathcal{B}$ は訓練データのミニバッチ，$\mathcal{M}$ は $\boldsymbol{\theta}_l\sim q_{\boldsymbol{\mu}_l,\boldsymbol{\Sigma}_l}(\boldsymbol{\theta})$ に従ってサンプルされた $\boldsymbol{\theta}_l$ の集合，$|\,\mathcal{B}\,|$，$|\,\mathcal{M}\,|$ はそれぞれ要素数，$\bar{\mathbf{g}}$ は $\mathcal{M}$ 内の負の対数尤度に対する勾配の平均，$\bar{\mathbf{F}}$ は $\mathcal{M}$ 内のフィッシャー情報行列の平均，$\hat{\mathbf{F}}$ は $\bar{\mathbf{F}}$ の移動平均，α は学習率，β は $\bar{\mathbf{F}}$ の移動平均の比率を定める係数，m は momentum，γ は学習を安定化するダンピング値，$\mathbf{I}$ は単位行列，λ は weight decay，ρ は共分散行列 $\boldsymbol{\Sigma}_l$ の成分の大きさを調整する係数を表す．

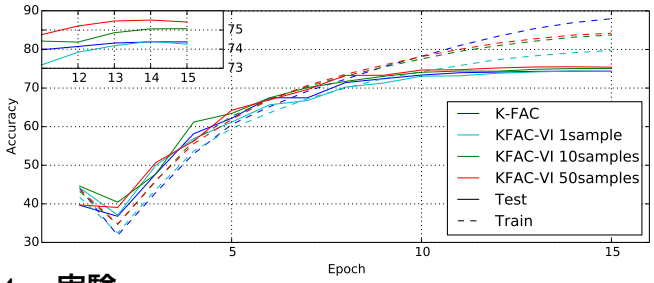
1 情報処理学会
IPSJ, Chiyoda, Tokyo 101–0062, Japan
†1 現在，東京工業大学
Presently with Tokyo Institute of Technology
a) nakata.h.ac@rio.gsic.titech.ac.jp
b) oosawak@rio.gsic.titech.ac.jp
c) rioyokota@gsic.titech.ac.jp

*1 https://www.cs.toronto.edu/~kriz/cifar.html
*2 https://github.com/pytorch/pytorch

表 1 各種ハイパーパラメータ

model	optimizer	$\alpha^{(0)}$	$\alpha^{(base)}$	$m^{(base)}$	λ	β	γ	ρ	p_{decay}
LeNet5	K-FAC	9E-7	9E-6	0.9	2E-3	0.333	1E-7	-	2
LeNet5	KFAC-VI	9E-7	9E-6	0.9	2E-3	0.333	1E-7	6E-5	2
ResNet18	K-FAC	6e-7	6E-6	0.3	9E-3	0.333	2E-7	-	2.3
ResNet18	KFAC-VI	6e-7	6E-6	0.3	9E-3	0.333	2E-7	8E-6	2.3

図 1 LeNet5 各学習の比較



4. 実験

4.1 実験設定

実装は機械学習フレームワーク PyTorch[*1]上で行い，学習は 1GPU(NVIDIA Tesla P100) 上で行なった．画像分類タスクのデータセットとして CIFAR-10[*2]を使用し，訓練データに対するデータの水増し (Data Augmentation) は行なっていない．ネットワークモデルは LeNet5[6]，ResNet18[7][*3]を使用した．学習率のスケジューリングは学習初期 1000 イテレーションに Gradual warmup[8] を行い，それ以降は polynomial decay に従いイテレーションごとの学習率の減衰を行なった．polynomial decay は，

$$\mathrm{T}'_{fin} = \mathrm{T}_{fin} + \mathrm{T}_{epoch} \tag{13}$$

$$\alpha^{(t)} = \alpha^{(base)}(1 - \frac{t - \mathrm{T}_{start}}{\mathrm{T}'_{fin} - \mathrm{T}_{start}})^{p_{decay}} \tag{14}$$

に従って計算される．ここで $T_{start}(= 1000)$ は polynomial decay の開始するイテレーション，T_{fin} は学習が完了するイテレーション，T_{epoch} は 1 エポックあたりのイテレーション数，p_{decay} は減衰率を調整するパラメータである．学習率および momentum は Gradual warmup 終了時に基準値 $\alpha^{(base)}$，$m^{(base)}$ に到達するものとする．ミニバッチサイズはいずれも 64 とし，学習エポックは LeNet5 は 15，ResNet18 は 30 とした．KFAC-VI の推論は学習時と同様に重みのサンプルを複数サンプルし，そのサンプル内での出力の平均をもとに test accuracy を算出している．推論時の重みのサンプル数は学習時のサンプル数によらず LeNet5 は 50，ResNet18 は 10 で固定とした．その他実験に使用したハイパーパラメータは表 1 に記した．ただし，”KFAC-VI”に続く”sample”の表記は学習に使用した重みのサンプル数を表す．

4.2 実験

本実験では K-FAC および KFAC-VI 1 sample[*4]による学習と複数の重みのサンプルを用いた KFAC-VI による学習の比較を行うため，LeNet5 は K-FAC，KFAC-VI 1-10-50 samples，ResNet18 は K-FAC，KFAC-VI 1-10 samples による検証を行なった．各検証における test accuracy を表 2 に示す．表 2 より，LeNet5，ResNet18 いずれのモデルにおいても，K-FAC，KFAC-VI 1 sample と比較して複数の重みのサンプルによる KFAC-VI の方が高い精度が得られた．また，LeNet5 では KFAC-VI 10 samples の 75.07%に比べ KFAC-VI 50 samples では 75.47%となり， 学習に使用するサンプル数を増やすことにより高い精度が得られている．図 1 は LeNet5 の各学習における train accuracy および test accuracy の学習曲線である．KFAC-VI 50 samples は K-FAC に比べ高い test accuracy が得られ，train accuracy と test accuracy の差を小さくしており，K-FAC に比べ学習が汎化していることが言える．

5. おわりに

本研究では画像認識による K-FAC および KFAC-VI 1 sample による学習と複数の重みのサンプルを用いた KFAC-VI による学習の比較を行なった．モンテカルロ法に基づいた複数の重みのサンプルによる変分推論を行うことにより，1 サンプルによる学習よりもより高い分類精度が得られ，K-FAC と比較して学習を汎化させることが可能であることを示した．

表 2 LeNet5，ResNet18 の各 test accuracy

	test accuracy			
	K-FAC	KFAC-VI		
model	-	1 sample	10 samples	50 samples
LeNet5	74.38	74.26	75.07	**75.42**
ResNet18	91.15	91.0	**91.9**	-

5.1 今後の課題

5.1.1 分散学習への適用

KFAC-VI では内部で複数のモデルによるアンサンブル学習を行なっているとみなせるため，複数の GPU 上での分散学習との相性が良いと考えられ，分散学習へ適用させることで学習時間が大幅に短縮されると期待できる．また，それに伴いより大きなミニバッチによる学習が可能となるため，その検証も今後の課題となる．

5.1.2 学習に使用する重みのサンプル数の調査

KFAC-VI において学習に使用する適切な重みのサンプル数はネットワークモデルやデータセットなど問題設定によって異なると考えられるため，その調査および理論的な解明が必要である．

謝辞 本研究は，JST CREST JPMJCR1687 の支援を受けたものである．

参考文献

[1] Hoffman M. D., Blei D. M., Wang C., and Paisley J.. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1) 1303–1347, 2013.

[2] Amari Shun-Ichi. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

[3] Guodong Zhang, Shengyang Sun, David Duvenaud, and Roger Grosse. Noisy natural gradient as variational inference. `https://arxiv.org/abs/1712.02390`, 2017.

[4] J. Martens and R. Grosse. Optimizing Neural Networks with Kronecker-factored Approximate Curvature. In *International Conference on Machine Learning*, pp. 2408–2417, 2015.

[5] R. Grosse and J. Martens. A Kronecker-factored approximate Fisher matrix for convolution layers. In *International Conference on Machine Learning*, pp. 573–582, 2016.

[6] Y. Lecun, L. Bottou, Y. Bengio, and P Haffner. Gradient-Based Learning Applied to Document Recognition. In *Proceedings of the IEEE*, 86(11):2278-2324 1998a.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Proceedings of CVPR*, pp 770–778, 2016.

[8] P. Goyal, P. Dolláer, R. Girshick, P. Noordhuis et al. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. `https://arxiv.org/abs/1706.02677`, 2017.

[*3] CIFAR-10 で使用するための修正を行なっている

[*4] Noisy K-FAC に近い学習となるが，momentum および weight decay の適用により厳密には同等の学習を行なっていない．同条件下の精度比較を行うため，本実験では比較対象として KFAC-VI 1 sample を使用した．