数理解析研究所講究録 第2112巻 2019年 201-207

共役勾配方向を適用した確率的最適化アルゴリズムによる再帰的ニューラルネットワーク上での言語モデルの生成

明治大学 理工学部 情報科学科　小林 悠
明治大学 理工学部 情報科学科　飯塚 秀明

Department of Computer Science, School of Science and Technology, Meiji University
Yu KOBAYASHI
Department of Computer Science, School of Science and Technology, Meiji University
Hideaki IIDUKA

概要

本稿では、確率的最適化問題について議論する。確率的最適化問題を解くための確率的最適化アルゴリズムは機械学習の分野で用いられており、特に画像認識や自然言語処理などのタスクで非常に高い精度を誇るディープラーニングの分野で重要な役割を担っている。確率的最適化アルゴリズムとしては、確率的勾配降下法などの確率的勾配に基づいた手法が提案されており、中でも Adam は機械学習のための最適化アルゴリズムとして優れた性能を示している。そこで本稿では、無制約非線形最適化問題にの分野で知られている共役勾配方向を Adam に適用し、通常の確率的勾配の代わりに共役勾配を用いた手法を提案する。また、自然言語処理でよく知られている PTB データセットを用いて再帰的ニューラルネットワーク上で言語モデルを生成する実験を行い、その結果に基づいて既存手法と提案手法の比較や考察を行う。

1　はじめに

本稿は、不要な情報 (ノイズ) が含まれる確率的目的関数の確率的最適化問題 [16] について扱う。この問題は、画像認識や自然言語処理などの多くのタスクで優れた性能を示し、近年注目を集めているディープラーニングにおける損失関数の最小化問題を含んでいる。

その重要性から確率的最適化問題を解くための反復アルゴリズムが多く提案されてきた。例えば、確率的勾配降下法 [1, 2, 7, 8, 16, 21, 22] が Robbins と Monro によって提案され、それを礎とした Momentum 法 [20] や Nesterov の加速法 [17]、AdaGrad [4]、RMSProp [23] などの確率的勾配に基づいた反復アルゴリズムが提案されている。Kingma と Ba [13] AdaGrad と RMSProp の利点を組み合わせた Adam (Adaptive momentum estimater) を提案し、画像認識による手書き文字認識や自然言語処理による映画の評判分析といった機械学習タスクにおける優れた性能を示している。したがって、本稿では無制約非線形最適化の分野で知られている共役勾配方向を Adam に適用したアルゴリズムを提案する。

提案手法は、無制約非線形最適化のための手法として用いられる非線形共役勾配法 [10] に基づい

た手法である。非線形共役勾配法とは、最急降下法における最急降下方向の代わりに、共役勾配パラメータによって計算された共役勾配方向を用いて点列を更新する手法である。この共役勾配方向を確率的最適化問題を解くための優れたアルゴリズムである Adam に適用したものが提案手法である。

さらに本稿では、自然言語処理のデータセットとしてよく知られている Penn Treebank (PTB) [15] データセットを用いた再帰的ニューラルネットワークによる言語モデルを生成する実験を行なった。その結果に基づいて、既存手法との比較を行うことで提案手法の機械学習における最適化アルゴリズムとしての有用性を示す。

以降の構成は次の通りである。2 章においては、以降の議論で用いる記号を定義し、確率的勾配に基づいた最適化手法や共役勾配法について紹介する。3 章においては、提案アルゴリズムについて述べる。4 章においては、PTB データセットと再帰的ニューラルネットワークを用いた数値実験結果を示す。5 章においては、本稿での議論を総括する。

2 数学的準備

2.1 記法と定義

本稿において、$\mathbb{R}$ を実数全体の集合、$\mathbb{R}^+$ を正の実数全体の集合、$\mathbb{N}$ を (0 を含まない) 自然数全体の集合とする。$\mathbb{R}^N$ を N 次元ユークリッド空間とし、$\|\cdot\| : \mathbb{R}^N \to [0, \infty)$ をノルムとする。$\mathbb{E}[X]$ を確率変数 X から計算される期待値とし、$f : \mathbb{R}^N \to \mathbb{R}$ を不要な情報 (ノイズ) を有する $\mathbb{R}^N$ 上で微分可能な確率的目的関数 (以下、確率的ノイズあり目的関数と呼ぶ) [16] とする. ξ を集合 $\Xi \subset \mathbb{R}$ にサポートされた確率分布 P に従う乱数し、乱数 ξ の独立同分布 (iid) に従う実現値を生成することができると仮定する。$f_1(\boldsymbol{\theta}), \ldots, f_T(\boldsymbol{\theta})$ を部分列 $\{1, \ldots, T\}$ によって示される時刻における確率的ノイズあり目的関数の実現値とする。

2.2 確率的最適化問題と確率的勾配に基づいた最適化アルゴリズム

次式で与えらえる最適化問題を本稿で考える確率的最適化問題とする。

問題 1 (確率的最適化問題)

$$\text{Minimize} \sum_{t=1}^{T} f_t(\boldsymbol{\theta}) \text{ subject to } \boldsymbol{\theta} \in \mathbb{R}^N. \tag{1}$$

問題 1 を解くための手法として、以下のような確率的勾配を用いたアルゴリズムが提案されている。ただし、時刻 $t \in \{1, \ldots, T\}$ に対して $\mathbf{g}_t = [\mathrm{g}_{t,1}, \ldots, \mathrm{g}_{t,N}]^\top := \nabla f_{\xi_t}(\boldsymbol{\theta}_t)$ を確率的勾配とし、$\tilde{\mathbf{g}}_t = [\mathrm{g}_{t,1}^2, \ldots, \mathrm{g}_{t,N}^2]^\top$ とし、$\alpha_t \in (0, 1)$ をステップ幅とする。また、以降では $\nabla f_t(\boldsymbol{\theta}_t) := \nabla f_{\xi_t}(\boldsymbol{\theta}_t)$ とする。

確率的勾配降下法 [1, 2, 7, 8, 16, 21, 22]:

$$\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t - \alpha_t \mathbf{g}_t \tag{2}$$

AdaGrad [4]:

$$\begin{aligned}
&\boldsymbol{h}_{t+1} := \boldsymbol{h}_t + \tilde{\mathbf{g}}_t, \\
&\boldsymbol{\alpha}_{t+1} := \left[\frac{\alpha_0}{\sqrt{h_{t+1,1}}}, \ldots, \frac{\alpha_0}{\sqrt{h_{t+1,N}}}\right]^\top, \\
&\boldsymbol{d}_{t+1} := [\alpha_{t+1,1}\mathrm{g}_{t,1}, \ldots, \alpha_{t+1,N}\mathrm{g}_{t,N}]^\top, \\
&\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t - \boldsymbol{d}_{t+1},
\end{aligned} \tag{3}$$

ただし、 $\epsilon := 10^{-8}$, $h_0 := \epsilon$, $\alpha_0 \in (0,1)$ とする。

RMSProp [23]:

$$\begin{aligned}
&\boldsymbol{h}_{t+1} := \beta\boldsymbol{h}_t + (1-\beta)\tilde{\mathbf{g}}_t, \\
&\alpha_{t+1} := \left[\frac{\alpha_0}{\sqrt{h_{t+1,1}+\epsilon}}, \ldots, \frac{\alpha_0}{\sqrt{h_{t+1,N}+\epsilon}}\right]^\top, \\
&\boldsymbol{d}_{t+1} := [\alpha_{t+1,1}\mathrm{g}_{t,1}, \ldots, \alpha_{t+1,N}\mathrm{g}_{t,N}]^\top, \\
&\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t - \boldsymbol{d}_{t+1},
\end{aligned} \tag{4}$$

ただし、 $h_0 := 0$, $\beta \in [0,1)$, $\alpha_0 \in (0,1)$, $\epsilon := 10^{-2}$ とする。

Adam [13]:

$$\begin{aligned}
&\boldsymbol{m}_{t+1} := \beta_1\boldsymbol{m}_t + (1-\beta_1)\mathbf{g}_t,\ \boldsymbol{v}_{t+1} := \beta_1\boldsymbol{v}_t + (1-\beta_1)\tilde{\mathbf{g}}_t, \\
&\widehat{\boldsymbol{m}}_{t+1} := \frac{1}{1-\beta_1^t}\boldsymbol{m}_{t+1},\ \widehat{\boldsymbol{v}}_{t+1} := \frac{1}{1-\beta_2^t}\boldsymbol{v}_{t+1}, \\
&\boldsymbol{d}_{t+1} := \left[\frac{\widehat{m}_{t+1,1}}{\sqrt{\widehat{v}_{t+1,N}}+\epsilon}, \ldots, \frac{\widehat{m}_{t+1,1}}{\sqrt{\widehat{v}_{t+1,1}}+\epsilon}\right]^\top, \\
&\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t - \alpha_t\boldsymbol{d}_{t+1},
\end{aligned} \tag{5}$$

ただし、 $\alpha_t := \alpha \in (0,1)$ $(t = 1, \ldots, T)$, $\beta_1 \in [0,1)$, $\beta_2 \in [0,1)$, $\epsilon := 10^{-8}$, $\boldsymbol{m}_0 := \mathbf{0}$, $\boldsymbol{v}_0 := \mathbf{0}$ とする。

2.3 無制約非線形最適化問題と非線形共役勾配法

問題 2 (無制約非線形最適化問題) $f : \mathbb{R}^N \to \mathbb{R}$ を連続的微分可能な下に有界な関数とするとき

$$\text{Minimize } f(\boldsymbol{\theta}) \text{ subject to } \boldsymbol{\theta} \in \mathbb{R}^N. \tag{6}$$

問題 2 を解くための手法として、任意の初期点 $\boldsymbol{\theta}_0$ から開始して、次の式 (7) で点列を更新する非線形共役勾配法 [10] がよく知られている。ただし、$\alpha_t > 0$ はステップ幅、$\boldsymbol{g}_t := \nabla f(\boldsymbol{\theta}_t)$ は勾配、γ_t は共役勾配パラメータである。

$$\begin{aligned}
&\boldsymbol{d}_0 := -\boldsymbol{g}_0,\ \boldsymbol{d}_{t+1} := -\boldsymbol{g}_{t+1} + \gamma_t\boldsymbol{d}_t \\
&\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t + \alpha_t\boldsymbol{d}_t,
\end{aligned} \tag{7}$$

共役勾配パラメータとして、Hestenes-Stiefel(HS) [11], Fletcher-Reeves(FR) [6], Polak-Ribière-Polyak (PRP) [18, 19], Conjugate-Descent(CD) [5], Liu-Storey(LS) [14], Dai-Yuan (DY) [3] などがよく知られており、以下に示す。

$$\begin{aligned}\gamma_t^{\mathrm{HS}} &:= \frac{\boldsymbol{g}_{t+1}^\top \boldsymbol{y}_t}{\boldsymbol{d}_t^\top \boldsymbol{y}_t},\ \gamma_t^{\mathrm{FR}} := \frac{\|\boldsymbol{g}_{t+1}\|^2}{\|\boldsymbol{g}_t\|^2},\ \gamma_t^{\mathrm{PRP}} := \frac{\boldsymbol{g}_{t+1}^\top \boldsymbol{y}_t}{\|\boldsymbol{g}_t\|^2},\\ \gamma_t^{\mathrm{CD}} &:= \frac{\|\boldsymbol{g}_{t+1}\|^2}{-\boldsymbol{d}_t^\top \boldsymbol{g}_t},\ \gamma_t^{\mathrm{LS}} := \frac{\boldsymbol{g}_{t+1}^\top \boldsymbol{y}_t}{-\boldsymbol{d}_t^\top \boldsymbol{g}_t},\ \gamma_t^{\mathrm{DY}} := \frac{\|\boldsymbol{g}_{t+1}\|^2}{\boldsymbol{d}_t^\top \boldsymbol{y}_t},\end{aligned} \tag{8}$$

ただし、$\boldsymbol{y}_t := \boldsymbol{g}_{t+1} - \boldsymbol{g}_t$ とする。Hager と Zhang らは [9] で次の共役勾配パラメータの公式を提案した。

$$\gamma_t^{\mathrm{HZ}} := \frac{\boldsymbol{g}_{t+1}^\top \boldsymbol{y}_t}{\boldsymbol{d}_t^\top \boldsymbol{y}_t} - \lambda \frac{\boldsymbol{g}_{t+1}^\top \boldsymbol{y}_t}{\|\boldsymbol{y}_t\|^2} \boldsymbol{g}_{t+1}^\top \boldsymbol{d}_t, \tag{9}$$

ただし、$\lambda > 1/4$ かつ $\boldsymbol{d}_t^\top \boldsymbol{y}_t \neq 0$ とする。これを

$$\gamma_t^{HZ+} := \max\{\eta_t, \gamma_t\},\ \eta_t := \frac{-1}{\|\boldsymbol{d}_t\| \min\{\eta, \|\boldsymbol{g}_t\|\}}\ (\eta > 0) \tag{10}$$

と修正することで、Wolfe 条件の下で大域的収束性が保証されている。ここで、Wolfe 条件は次のように定義される。

定義 1 (Wolfe 条件) ξ_1, ξ_2 が $0 < \xi_1 < \xi_2 < 1$ を満たす定数であるとき

$$f(\boldsymbol{\theta}_t + \alpha_t \boldsymbol{d}_t) \leq f(\boldsymbol{\theta}_t) + \xi_1 \alpha_t \boldsymbol{g}_t^\top \boldsymbol{d}_t, \tag{11}$$

$$\xi_2 \boldsymbol{g}_t^\top \boldsymbol{d}_t \leq \boldsymbol{g}(\boldsymbol{\theta}_t + \alpha_t \boldsymbol{d}_t)^\top \boldsymbol{d}_t. \tag{12}$$

3 提案アルゴリズム

問題 1 に対するアルゴリズムとして、Algorithm 1 を与える。既存手法である Adam のステップ 10 から 12 において確率的勾配 $\mathbf{g}_t := \nabla_{\boldsymbol{\theta}_t} f_{t+1}(\boldsymbol{\theta}_t)$ を用いて計算していた部分をステップ 4 から 8 で計算した共役勾配方向 $\boldsymbol{d}_t$ を用いたものに置き換えたものである。

4 数値実験

本稿で示す数値実験は、[12] を参考にして、Penn Treebank (PTB) [15] データセットを用いた再帰的ニューラルネットワークによる言語モデル生成を既存手法と提案手法それぞれでパラメータを更新することで行った。実験に用いた計算機は Intel Core i5 (1.8 GHz) CPU、8 GB 1600 MHz DDR3 メモリ、 MacOS Mojave 10.14.1 を有し、プログラムのライブラリは Python 3.6.6 、NumPy 1.15.0. を用いた。また、評価指標としては言語モデルの性能を示すパープレキシティを用いた。パープレキシティは損失 (目的) 関数値 L に対して $\exp\ L$ で表され、パープレキシティが小さいほど適切な単語の予測ができる優れたモデルということを表している。図 1 は学習時の各エ

Algorithm 1 CoBA: Conjugate-gradient-Based Adam

Require: $\alpha_1, \dots \alpha_T \in \mathbb{R}^+, \beta_1, \beta_2 \in [0, 1), \epsilon := 10^{-8}, f_1, \dots, f_T : \mathbb{R}^N \to \mathbb{R}, \boldsymbol{\theta}_0 \in \mathbb{R}^N$

Ensure: $\boldsymbol{\theta}_t$

$t \leftarrow 0, \boldsymbol{m}_0 := \mathbf{0}, \boldsymbol{v}_0 := \mathbf{0}$

while $\boldsymbol{\theta}_t$ not converged **do**

  $\mathbf{g}_{t+1} := \nabla_{\boldsymbol{\theta}_t} f_{t+1}(\boldsymbol{\theta}_t)$

  if t = 0 **then**

    $\boldsymbol{d}_{t+1} := \mathbf{g}_{t+1}$

  else

    γ_{t+1} : computed by the conjugate gradient parameter's rules.

    $\boldsymbol{d}_{t+1} := \mathbf{g}_{t+1} - \gamma_{t+1} \boldsymbol{d}_t$

  end if

  $\tilde{\boldsymbol{d}}_{t+1} := [d_{t+1,1}^2, d_{t+1,2}^2, \dots, d_{t+1,N}^2]^\top$

  $\boldsymbol{m}_{t+1} := \beta_1 \boldsymbol{m}_t + (1 - \beta_1) \boldsymbol{d}_{t+1}$

  $\boldsymbol{v}_{t+1} := \beta_2 \boldsymbol{v}_t + (1 - \beta_2) \tilde{\boldsymbol{d}}_{t+1}$

  $\widehat{\boldsymbol{m}}_{t+1} := (1 - \beta_1^t)^{-1} \boldsymbol{m}_{t+1}$

  $\widehat{\boldsymbol{v}}_{t+1} := (1 - \beta_2^t)^{-1} \boldsymbol{v}_{t+1}$

  $\widehat{\boldsymbol{d}}_{t+1} := [\widehat{m}_{t+1,1}/(\widehat{v}_{t+1,1} + \epsilon), \widehat{m}_{t+1,2}/(\widehat{v}_{t+1,2} + \epsilon), \dots, \widehat{m}_{t+1,N}/(\widehat{v}_{t+1,N} + \epsilon)]^\top$

  $\boldsymbol{\theta}_{t+1} := \boldsymbol{\theta}_t - \alpha_{t+1} \widehat{\boldsymbol{d}}_{t+1}$

  $t \leftarrow t + 1$

end while

ポックに対するパープレキシティの値を示している。ただし、CoBA(HZ), CoBA(HS), CoBA(FR), CoBA(PRP), CoBA(CD),CoBA(LS), CoBA(DY) はそれぞれ共役勾配パラメータ γ^{HZ}, γ^{HS}, γ^{FR}, γ^{PRP}, γ^{CD}, γ^{LS}, γ^{DY} を用いた提案手法 CoBA (Conjugate-gradient-based Adam) 法とする。提案手法すべてが既存手法よりもパープレキシティが小さい値、つまり学習によって優れた言語モデルが得られたことを示している。

5 まとめ

本稿では、確率的最適化アルゴリズムについて議論した。確率的最適化アルゴリズムとしては、確率的勾配に基づいた手法が多く存在し、その中でも機械学習のための最適化アルゴリズムとして特に優れた性能を有する Adam に対して、無制約非線形最適化アルゴリズムである共役勾配方向を適用した確率的最適化アルゴリズムを提案した。最後に、再帰的ニューラルネットワークを用いた自然言語処理において既存手法との数値実験による比較を行うことで、提案手法の有用性を示した。

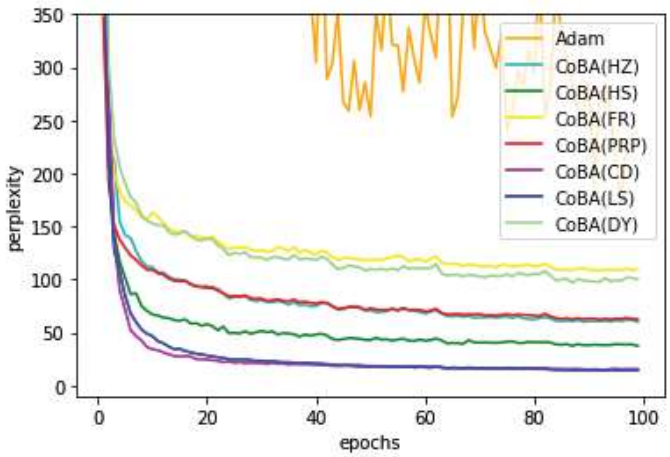

図 1: 既存手法と提案手法それぞれの各エポックにおけるパープレキシティ

謝辞

本研究の遂行におきましては、研究発表および議論の機会を与えて下さりました千葉大学法政経部経済学コースの青山耕治先生に、心より感謝申し上げます。

参考文献

[1] V. S. Borkar. *Stochastic approximation: a dynamical systems viewpoint*, volume 48. Springer, 2009.

[2] L. Bottou. Online algorithms and stochastic approximations. In David Saad, editor, *Online Learning and Neural Networks*. Cambridge University Press, Cambridge, UK, 1998. revised, oct 2012.

[3] Y.H. Dai and Y. Yuan. A nonlinear conjugate gradient method with a strong global convergence property. *SIAM Journal on optimization*, 10(1):177–182, 1999.

[4] J. Duchi, E. Hazan, and Y. Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[5] R. Fletcher. *Practical methods of optimization*, volume 1. John Wiley & Sons, New York, 1987.

[6] R. Fletcher and C. M. Reeves. Function minimization by conjugate gradients. *The Computer Journal*, 7(2):149–154, 1964.

[7] S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization i: A generic algorithmic framework. *SIAM Journal on*

Optimization, 22(4):1469–1492, 2012.

[8] S. Ghadimi and G. Lan. Optimal stochastic approximation algorithms for strongly convex stochastic composite optimization, ii: shrinking procedures and optimal algorithms. *SIAM Journal on Optimization*, 23(4):2061–2089, 2013.

[9] W.W. Hager and H. Zhang. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM Journal on optimization*, 16(1):170–192, 2005.

[10] W.W. Hager and H. Zhang. A survey of nonlinear conjugate gradient methods. *Pacific journal of Optimization*, 2(1):35–58, 2006.

[11] M. R. Hestenes and E. Stiefel. *Methods of conjugate gradients for solving linear systems*, volume 49. NBS Washington, DC, 1952.

[12] O'Reilly Japan. Github repository "deep-learning-from-scratch-2". `https://github.com/oreilly-japan/deep-learning-from-scratch-2`.

[13] D.P. Kingma and J.L. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[14] Y. Liu and C. Storey. Efficient generalized conjugate gradient algorithms, part 1: theory. *Journal of optimization theory and applications*, 69(1):129–137, 1991.

[15] T. Mikolov. Penn Treebank dataset. `http://www.fit.vutbr.cz/~imikolov/rnnlm/`, 2010-2012.

[16] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

[17] Y. Nesterov. A method for solving the convex programming problem with convergence rate $o(1/k^2)$. *Doklady Akademii Nauk SSSR*, 269(3):543–547, 1983.

[18] E. Polak and G. Ribière. Note on convergence of conjugate direction methods. *Revue Francaise D Informatique De Recherche Operationnelle*, 3(16):35–43, 1969.

[19] B. T. Polyak. The conjugate gradient method in extremal problems. *USSR Computational Mathematics and Mathematical Physics*, 9(4):94–112, 1969.

[20] N. Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 12(1):145–151, 1999.

[21] H. Robbins and S. Monro. A stochastic approximation method. In *Herbert Robbins Selected Papers*, pages 102–109. Springer, 1985.

[22] S. S. Shwartz, Y. Singer, N. Srebro, and A. Cotter. Pegasos: Primal estimated subgradient solver for svm. *Mathematical Programming*, 27(1):807–814, 2011.

[23] T. Tieleman and G. Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.