Growing descent of stochastic gradient with the generalized logistic map

I. Kulikovskikh[1,2,3], S. Prokhorov[1], T. Legović[3] and T. Šmuc[3]

[1]Samara National Research University, Moskovskoe Shosse 34, Samara, Russia, 443086
[2]Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, Zagreb,Croatia, 10000
[3]Ruder Bošković Institute, Bijenička cesta 54, Zagreb, Croatia, 10000

Abstract. The paper considers the problem of accelerating the convergence of stochastic gradient descent (SGD) in an automatic way. Previous research puts forward such algorithms as Adagrad, Adadelta, RMSprop, Adam and etc. to adapt both the updates and learning rates to the slope of a loss function. However, these adaptive methods do not share the same regret bound as the gradient descent method. Adagrad provably achieves the optimal regret bound on the assumption of convexity but accumulates the squared gradients in the denominator that dramatically shrinks the learning rate. This research is aimed at introducing a generalized logistic map directly into the SGD method in order to automatically set its parameters to the slope of the logistic loss function. The optimizer based on the logistic map may be considered as a meta-learner that learns how to tune both the learning rate and gradient updates with respect to the rate of population growth. The present study yields the "growing" descent method and a series of computational experiments to point out the benefits of injecting the logistic map.

1. Introduction

The ability to rapidly adapt from small pieces of data to current tasks is essential to effective learning. However, deep learning algorithms traditionally require big datasets to learn tasks by fitting a deep neural network over them through extensive incremental updates of SGD. This approach seems time-consuming and even more challenging if fast adaptation is crucial.

Meta-learning has opened a door for learning optimizers to exploit problems structure in an automatic way [1–19]. This suggests that optimizers which are used to be hand-designed can serve as meta-learners by moving the learning level up from data to tasks.

While SGD is usually considered as a meta-learner, the algorithm itself still needs improvements. For example, it is advisable to adapt larger learning rates for smaller gradients and smaller learning rates for larger gradients to balance their respective influences. A number of adaptive methods were proposed to overcome this weakness of SGD such as Adagrad, Adadelta, RMSprop, Adam and etc. [20, 21]. However, even if these methods do speed up the convergence rate, they nevertheless result in worse generalization error compared to the SGD with a single learning rate [22].

The aim of this study is to propose a meta-learner based on the SGD method that could adapt the learning rate as well as gradient updates to the slope of the logistic loss function with better generalization error than the plain SGD. For this purpose, we suggest to "grow" descent of stochastic gradient by embedding the generalized logistic map directly in the SGD method. This allows us to:

(i) guarantee the same regret bound as the gradient descent method;
(ii) introduce deterministic chaos that may greatly assist in improving the convergence rate of gradient methods [30–33];
(iii) learn how to tune both the learning rate and gradient updates with respect to the rate of population growth automatically.

In addition, the parameters of logistic map have a clear interpretation in biological and ecological systems that may bring the potential advantage to modelling the nature-inspired framework of meta-learning.

2. Problem statement

Consider a dataset $\{\mathrm{x}_i, y_i\}_{i=1}^m$ with $\mathrm{x}_i \in \mathbb{R}^n$ and $y_i \in \{0, 1\}$. Let us state the problem [20]

$$\mathscr{L}(\boldsymbol{\theta}) \underset{\boldsymbol{\theta}}{\to} \min, \tag{1}$$

where a loss function with the weight vector $\boldsymbol{\theta} \in \mathbb{R}^n$ is defined as

$$\mathscr{L}(\boldsymbol{\theta}) = \sum_{i=1}^{m} \ell(y_i \boldsymbol{\theta}^T \mathrm{x}_i). \tag{2}$$

In equation (2) it is assumed that all the labels are positive: $\forall i : y_i = 1, \|x_i\| < 1$, the dataset is linearly separable: $\exists\, \boldsymbol{\theta}^*$ such that $\forall i : \boldsymbol{\theta}^{*T} \mathrm{x}_i > 0$ and $\forall t$: $\ell(t)$ is differentiable and monotonically decreasing to zero

$$\ell(t) > 0, \ell'(t) < 0, \lim_{t\to\infty} \ell(t) = \lim_{t\to\infty} \ell'(t) = 0,$$

and its derivative is β-Lipshitz: $\ell(t') \leq \ell(t) + \langle \nabla \ell(t), t' - t \rangle + \frac{\beta}{2}\|t' - t\|^2, \quad \beta > 0.$

2.1. Stochastic gradient descent

The solution to the problem (1) can be found as the iterates of gradient descent with a full batch [20, 21]:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \nabla \mathscr{L}(\boldsymbol{\theta}_t) = \boldsymbol{\theta}_t - \eta \sum_{i=1}^{m} \ell'(\boldsymbol{\theta}_t^{\mathrm{T}} \mathrm{x}_i)\mathrm{x}_i,$$

where η is the learning rate. In the stochastic setting, gradient descent updates the weight vector for each i^{th} mini-batch dataset such as

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \nabla \mathscr{L}(\boldsymbol{\theta}_t; \mathrm{x}_i) = \boldsymbol{\theta}_t - \eta \sum_{l=1}^{i} \ell'(\boldsymbol{\theta}_t^{\mathrm{T}} \mathrm{x}_l)\mathrm{x}_l. \tag{3}$$

The equation (3) implies that only the partial information of the loss function for i^{th} batch data guides the gradient direction for each iteration.

The regret bound of gradient descent algorithm with a full batch is bounded by a constant that entails the loss function at a certain step that is bounded by the inverse of the number of iterations. The SGD method shares the same regret bound as it modifies the basic structure of the gradient descent.

In the following section we define the logistic map for insertion into the SGD method.

2.2. The generalized logistic map

The model that describes the population growth $P(t)$ in an infinite environment can be presented as [23]:

$$\frac{\mathrm{d}P(t)}{\mathrm{d}t} = rP(t),$$

where $r \equiv b - d$ is the *per capita* rate of population growth, b and d are respective *per capita* rates of birth and death.

Let us introduce the generalized logistic equation and its solution [23]:

$$\frac{dP(t)}{dt} = -r\left(P(t) - A\right)\ln^{[q]}\left(\frac{P(t) - A}{K - A}\right), \tag{4}$$

where

$$\ln^{[q]}(x) = \int_1^x \frac{dt}{t^{1-q}} = \begin{cases} \frac{x^q - 1}{q}, & q \neq 0; \\ \ln(x), & q \to 0, \end{cases} \tag{5}$$

$$P(t) = A + \frac{K - A}{\left(1 - \left(1 - \left(\frac{K-A}{P_0+A}\right)^q\right)\exp(-rt)\right)^{\frac{1}{q}}}, \tag{6}$$

where q is the generalization parameter, $0 \leq A < K$, K is the upper asymptote or the carrying capacity of population; A is the lower asymptote that indicates critical population thresholds below which a population crashes to extinction. The asymptote A serves as a substitute for the Allee effects [24] which are broadly defined as a decline in individual fitness at low population size or density. Even if modelling these effects is beyond the scope of this paper, it draws promising directions for further research. Given that [23] :

(i) $q = 1$ and $K \to \infty$: the generalized model reduces to the Malthus model that describes the exponential growth of a population [25];

(ii) $q = 1$: the generalized model reduces to the Verhulst model that describes the logistic growth of a population [26];

(iii) $q \to 0$: the generalized model reduces to the Gompertz model that also describes the logistic growth of a population, but it is more flexible in the way of approaching these asymptotes [27, 28].

A discrete-time population model (the logistic map) at the time k is given by [29]:

$$\Delta P_k = rP_k(1 - P_k),$$

where $P_k \in [0, 1]$, $k \in \mathbb{N}$ and $r \in (0, 4]$. Let $\Delta t = 1$, $\ell_r'(P) = rP(1 - P)$. Then $P_{k+1} = \ell_r'(P_k)$ and the composition of k functions $\ell_r(P)$ can be represented by:

$$\mathcal{L}_r^{k'}(P) = \begin{cases} \ell_r'(P), & k = 1; \\ \left(\ell_r' \circ \mathcal{L}_r^{[k-1]'}\right)(P), & k > 1. \end{cases} \tag{7}$$

The equation (7) describes the dynamics of a population. If the growth rate $0 < r \leq 1$, the population dies out and goes extinct. Increasing the rate of growth allows the population to settle at the stable value or fluctuate across booms and busts. Finally, at a relatively high values of growth rate, the logistic equation produces chaos [29].

3. Growing descent of stochastic gradient

Let us extend the definition $\ell(t)$ taking into account the generalized logistic equation (4) and its solution (6):

$$\ell'_r(t;a,b,q) = r(\ell_r(t;a,b,c,q) - a)\ln^q\left(\frac{\ell_r(t;a,b,c(a,b),q) - a}{b-a}\right); \tag{8}$$

$$\ell_r(t;a,b,q) = a + \frac{b-a}{(1-(1-(\exp(-c(a,b)))^q)\exp(-rt))^{\frac{1}{q}}}, \tag{9}$$

where $a \equiv A$, $b \equiv K$, $0 \le a < b \le 1$; $c(a,b) \equiv \ln\left(\frac{P_0+A}{K-A}\right)$, $P_0 < K - 2A$, $c(a,b) < 0$. Here $u \equiv t$, $k = 1$, $r \in (0,4]$.

The logistic loss and its derivative with regard to (9) and (8) can be given as follows:

$$\ln \mathscr{L}(\boldsymbol{\theta}; \mathrm{x}_i, \mathcal{L}'^k_r(a,b,q)) = -\sum_{l=1}^{i} \ln \ell(\boldsymbol{\theta}^T \mathrm{x}_l; a, b, q), \tag{10}$$

$$-\nabla \ln \mathscr{L}(\boldsymbol{\theta}; \mathrm{x}_i, \mathcal{L}'^k_r(a,b,q)) = -\sum_{l=1}^{i}\left(\left(1 - \ell(\boldsymbol{\theta}^T \mathrm{x}_l; a, b, q)\right)\frac{\mathcal{L}'^k_r(\boldsymbol{\theta}^{\mathrm{T}} \mathrm{x}_l; a, b, q)}{\ell'_r(\boldsymbol{\theta}^{\mathrm{T}} \mathrm{x}_l)} \mathrm{x}_l\right), \tag{11}$$

where $\mathcal{L}'^k_r(\boldsymbol{\theta}^{\mathrm{T}} \mathrm{x}_l; a, b, q)$ defines the generalized logistic map according to (7), $\ell'_r(\boldsymbol{\theta}^{\mathrm{T}} \mathrm{x}_l) = \ell(\boldsymbol{\theta}^T \mathrm{x}_l)\big(1 - \ell(\boldsymbol{\theta}^T \mathrm{x}_l)\big)$ gives the derivative of the sigmoid function $\ell(\boldsymbol{\theta}^T \mathrm{x}_l)$ with $a = 0$ and $b = 1$. Then, the growing descent method based on the generalized logistic map with reference to (10) and (11) can be defined as

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_r \nabla \ln \mathscr{L}(\boldsymbol{\theta}_t; \mathrm{x}_i, \mathcal{L}'^k_r(a,b,q)),$$

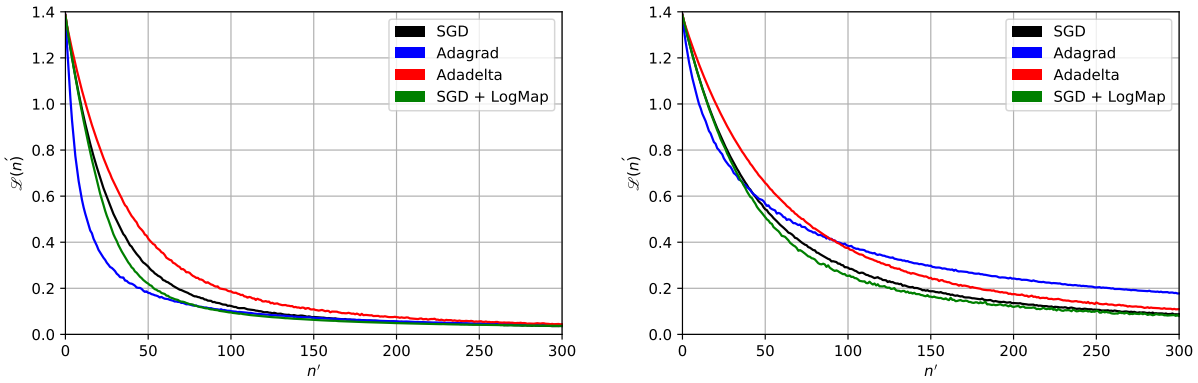
where the adaptive step size $\eta_r = (b-a)c(a,b)r_k\eta$.

In the following section we present the experimental evidence on the improved convergence rate of proposed growing descent method.

4. Results

We conducted a series of computational experiments on a linear separable dataset modelled with $\mathcal{N}(0,1)$ subject to $m = 5000$, $n = 10$, $n_{\text{epochs}} = 100$, the mini-batch size $m_{\text{batch}} = 100$ and the growth parameters $a = 0$, $b = 1$, $q = 1$. A simple model of neural network consists of one hidden layer with different number of neurons $n_{\text{neuron}} = 4$ (see Fig. 1 a)) and $n_{\text{neuron}} = 1$ (see Fig. 1 b)).

Fig. 1 a) shows the advantage of Adagrad with the adaptive learning rate over other methods. Fig. 1 b), in turn, reveals the Adagrad's main weakness: the result of the accumulation of the squared gradients in the denominator. This practically leads to a dramatic decrease in the learning rate for which the algorithm is no longer able to gain additional knowledge. Adadelta is an extension of Adagrad that is aimed at reducing its aggressive, steadily decreasing learning rate. However, it gives worse results than the proposed SGD+LogMap (see Fig. 1). We can also see that the growing descent algorithm SGD+LogMap improves the SGD convergence rate: it requires smaller number of iterations to achieve the same regret bound for both $n_{\text{neuron}} = 4$ if $n' > 10$ (see Fig. 1 a)) and $n_{\text{neuron}} = 1$ if $n' > 25$ (see Fig. 1 b)).



a) $n_{\text{neuron}} = 4$, $r = 2.8$ b) $n_{\text{neuron}} = 1$, $r = 3.8$

Figure 1. The results of computational experiments.

5. Conclusions

The proposed meta-learner is based on the plain SGD method and the generalized logistic map. It shares the same regret bound as the SGD and learns how to adapt both the learning rate and the gradient updates with the rate of population growth in an automatic way. In addition, the proposed approach to growing descent of stochastic gradient allows one to introduce deterministic chaos that improves the convergence rate. In a series of computational experiments the validity of the proposed approach has been confirmed.

6. References

[1] Wu, X. WNGrad: Learn the Learning Rate in Gradient Descent / X. Wu, R. Ward, L. Bottou [Electronic resource]. – Access mode: arXiv:1803.02865 (15.11.2018).
[2] Andrychowicz, M. Learning to Learn by Gradient Descent by Gradient Descent / M. Andrychowicz, M. Denil, S. Gomez Colmenarejo, M.W. Hoffman, D. Pfau, T. Schaul , B. Shillingford, N. de Freitas [Electronic resource]. – Access mode: arXiv:1606.04474 (15.11.2018).
[3] Ren, M. Learning to Reweight Examples for Robust Deep Learning / M. Ren, W. Zeng, B. Yang, R. Urtasun [Electronic resource]. – Access mode: arXiv:1803.09050 (15.11.2018).
[4] Ren, M. Stochastic Modified Equations and Adaptive Stochastic Gradient Algorithms / Q. Li, C. Tai, W. E [Electronic resource]. – Access mode: arXiv:1511.06251 (15.11.2018).
[5] Wichrowska, O. Learned Optimizers that Scale and Generalize / O. Wichrowska, N. Maheswaranathan, M.W. Hoffman, S. Gomez Colmenarejo, M. Denil, N. de Freitas, J. Sohl-Dickstein [Electronic resource]. – Access mode: arXiv:1703.04813 (15.11.2018).
[6] Li, Z. Meta-SGD: Learning to Learn Quickly for Few-Shot Learning / Z. Li, F. Zhou, F. Chen, H. Li [Electronic resource]. – Access mode: arXiv:1707.09835 (15.11.2018).
[7] Li, K. Learning to Optimize / K. Li, J. Malik [Electronic resource]. – Access mode: arXiv:1606.01885 (15.11.2018).
[8] Li, K. Learning to Optimize Neural Nets / K. Li, J. Malik [Electronic resource]. – Access mode: arXiv:1703.00441 (15.11.2018).
[9] Al-Shedivat, M. Continuous Adaptation via Meta-learning in Nonstationary and Competitve Environments / M. Al-Shedivat, T. Bansal, Y. Burda, I. Sutskever, I. Mordatch, P. Abbeel K. Li, J. Malik [Electronic resource]. – Access mode: arXiv:1710.03641 (15.11.2018).
[10] Aljundi, R. Memory Aware Synapses: Learning What (not) to Forget / R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, T. Tuytelaars [Electronic resource]. – Access mode: arXiv:1711.09601 (15.11.2018).
[11] Li, D. Learning to Generalize: Meta-Learning for Domain Generalization / D. Li, Y. Yang, Y.-Z. Song, T.M. Hospedales [Electronic resource]. – Access mode: arXiv:1710.03463 (15.11.2018).

[12] Wang, Y.-X. Learning to Model the Tail / Y.-X. Wang, D. Ramanan, M. Hebert [Electronic resource]. – Access mode: http://papers.nips.cc/paper/7278-learning-to-model-the-tail.pdf (15.11.2018).
[13] Lv, K. Learning Gradient Descent: Better Generalization and Longer Horizons / K. Lv, S. Jiang, J. Li [Electronic resource]. – Access mode: arXiv:1703.03633 (15.11.2018).
[14] Ren, M. Learning to Reweight Examples for Robust Deep Learning / M. Ren, W. Zeng, B. Yang, R. Urtasun [Electronic resource]. – Access mode: arXiv:1803.09050 (15.11.2018).
[15] Munkhdalai, T. Meta Networks / T. Munkhdalai, H. Yu [Electronic resource]. — Access mode: arXiv:1703.00837 (15.11.2018).
[16] Mishra, N. A Simple Neural Attentive Meta-learner / N. Mishra, M. Rohaninejad, X. Chen, P. Abbeel [Electronic resource]. – Access mode: arXiv:1707.03141 (15.11.2018).
[17] Finn, C. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks / C. Finn, P. Abbeel, S. Levine [Electronic resource]. – Access mode: arXiv:1703.03400 (15.11.2018).
[18] Duan, Y. One-Shot Imitation Learning / Y. Duan, M. Andrychowicz, B. Stadie, J. Ho, J. Schneider, I. Sutskever, P. Abbeel, W. Zaremba [Electronic resource]. – Access mode: http://papers.nips.cc/paper/6709-one-shot-imitation-learning.pdf (15.11.2018).
[19] Ha, D. Hypernetworks / D. Ha, A. Dai, Q.V. L [Electronic resource]. – Access mode: arXiv:1609.09106 (15.11.2018).
[20]Kim, H.S. Convergence Analysis of Optimization Algorithms / H.S. Kim, J.H. Kang, W.M. Park, S.H. Ko, Y.H. Cho, D.S. Yu, Y.S. Song, J.W. Choi [Electronic resource]. – Access mode: arXiv:1707.01647 (15.11.2018).
[21] Ruder, S. An Overview of Gradient Descent Optimization Algorithms [Electronic resource]. – Access mode: arXiv:1609.04747 (15.11.2018).
[22] Wilson, A. The Marginal Value of Adaptive Gradient Methods in Machine Learning / A. Wilson, R. Roelofs, M. Stern, N. Srebro, B. Recht // NIPS Proceedings,2017. – P. 4151-4161.
[23] Ribeiro, F.L. An Attempt to Unify Some Population Growth Models From First Principle // Revista Brasileira de Ensino de Fisica. – 2017. – Vol. 39(1). – P. e1311.
[24] Allee, W.C. Animal Aggregations // The Quarterly Review of Biology. – 1927. – Vol. 2(3). – P. 367-398.
[25] Malthus, T.R. An Essay on the Principle of Population, as it Affects the Future Improvement of Society. – London: J. Johnson, 1798. – 432 p.
[26] Verhulst, P.F. Notice Sur la loi que la Population Poursuit dans son Accroissement // Correspondance mathamatique et physique. – 1838. – Vol. 10. – P. 113-121.
[27] Gompertz, B. On the Nature of the Function Expressive of the law of Human Mortality, and on a new Mode of Determining the Value of Life Contingencies // Philosophical Transactions of the Royal Society of London B: Biological Sciences. – 1825. – Vol. 182. – P. 513-585.
[28] Winsor, C.P. The Gompertz Curve as a Growth Curve // Proc. Nat. Acad. Sci. – 1932. – Vol. 18(1). – P. 1-8.
[29] May, R.M. Simple Mathematical Models With Very Complicated Dynamics // Nature. – 1976. – Vol. 261(5560). – P. 459-467.
[30] Doel, K. The Chaotic Nature of Faster Gradient Descent / K. Doel, U. Ascher [Electronic resource]. – Access mode: www.cs.ubc.ca/ascher/papers/doas1.pdf (15.11.2018).
[31] Mpitsos, G.J. Convergence and Divergence in Neural Networks: Processing of chaos and biological analogy / G.J. Mpitsos, R.M. Jr. Burton // Neural Networks. – 1992. – Vol. 5. – P. 605-625.
[32] Verschure, P.F.M.J. Chaos-based Learning // Complex Systems. – 1991. – Vol. 5. – P. 359-370.
[33] Zhang, H. Deterministic Convergence of Chaos Injection-based Gradient Method for Training Feedforward Neural Networks / H. Zhang, Y. Zhang, D. Xu, X. Liu // Cognitive Neurodynamics. – 2015. – Vol. 9(3). – P. 331-340.

Acknowledgements
This work was supported by the Russian Federation President grant No. MK-6218.2018.9 and the Ministry of Education and Science of the Russian Federation grant No. 074-U01. The research in Section 3 was partly supported by RFBR (project No. 18-37-00219). The authors acknowledge the support by the Centre of Excellence project "DATACROSS", co-financed by the Croatian Government and the European Union through the European Regional Development Fund – the Competitiveness and Cohesion Operational Programme (KK.01.1.1.01.0009).