METHODS WITH PRECONDITIONING WITH WEIGHT DECAY REGULARIZATION

A PREPRINT

Kreinin M.
Department of Intelligent Systems
MIPT
Moscow, Russia
kreinin.mv@phystech.edu

Beznosikov A.
Department of Data Science
MIPT
Dolgoprudny, Russia
beznosikov.an@phystech.edu

ABSTRACT

This paper examines the convergence rate of adaptive gradient methods when a regularization function is added to the target function. This is an area of research, since many machine learning problems use the heuristic, heuristic regularization, and we investigate the theoretical and practical convergence of adaptive gradient methods.

Keywords Adam · OASIS · Regularization · ADAHessian · Weight Decay

1 Introduction

In machine learning we consider unconstrained optimization problem

$$\min_{w \in \mathbb{R}^d} f(w) \tag{1}$$

Problems of the form (1) cover a plethora of applications, including empirical risk minimization, deep learning [LeCun et al., 2015], and supervised learning [Cunningham et al., 2008] tasks such as regularized least squares [Rifkin and Lippert, 2007] or logistic regression [Shalev-Shwartz and Ben-David, 2014].

The classic base method for solving the optimization is gradient descent, but this minimization problem can be difficult to solve, particularly when the number of training samples n, or problem dimension d, is large. In modern machine learning especially large problems represent the greatest interest. For such cases stochastic gradient descent [Bottou, 2010] became popular solution. Despite its simplicity, it proved itself to be an efficient and effective optimization method. For a long time first-ordered methods were most popular approach of solving optimization problems.

Other way of solving the problem are methods with adaptive gradient [Wilson et al., 2017]. These methods posses several superiority over first-ordered methods. Firstly, they have bigger potential of distributed solving, because first ordered methods spend majority of time on "communication". Secondly, they are less sensitive to the choice of hyperparameters up to the point that hyperparameters can be set equal to one. Lastly, this methods often simply show faster convergence on modern large optimization problems, especially this methods became applicable in neural networks solving. Nowadays it is known that preconditioning methods often outperform other methods on modern large optimization problems [Zhang et al., 2018, Yao et al., 2021, Kingma and Ba, 2014, Goldberg et al., 2011].

Preconditioning methods refer to techniques that involve scaling the gradient of a problem by a specific matrix D_t, which enables the gradient to take into account the geometry of the problem. In the classical case $D_t = (\nabla^2 f)^{-1}$, which corresponds newton's method, however hessian is difficult to calculate and even more difficult to reverse, because of that some heuristics are used to replace the reversed hessian [Dennis and Moré, 1977]. In OASIS [Goldberg et al., 2011] or AdaHessian [Yao et al., 2021] hessian is assumed to have diagonal dominance. In Adam [Kingma and Ba, 2014] gradient is simply normalized, etc. This heuristics were proved to be effective and efficient. General scheme of methods with preconditioning can be framed in the following algorithm

Algorithm 1 General scheme for preconditions methods

```
Require: $\eta, f$
  while $w$ not converged do
    $t = t + 1$
    $g_t \leftarrow \nabla f(w_{t-1})$
    $D_t \leftarrow$ pseudo-hessian, precondtioning, scaling
    $w_t = w_{t-1} - \eta \cdot g_t D_t^{-1}$
  end while
```

Regularization is a powerful technique in machine learning that aims to prevent overfitting by adding additional constraints to the model. It has been widely applied to various machine learning problems, including image classification, speech recognition, and natural language processing, and has shown its effectiveness in improving the generalization capability of neural networks [Poggio et al., 1987].

In methods with preconditioning appears to be several ways to include regularization. We can include regularizer r in g_t calculation so it will be taken into consideration while calculating D_t. This method is equal to considering optimization problem

$$\min_{w \in \mathbb{R}^d} f(w) + r(w). \tag{2}$$

Or we can include regularizer only on last step, decreasing norm of w [Loshchilov and Hutter, 2017]. This way of regularization is called weight decay and surprisingly turns out to be more efficient in practical problems. There are few other ways of considering regularizer which will be discussed further in the paper.

In general, our paper provides insight into comparison of different consideration ways of regularization is methods with preconditioning. Here, we provide a brief summary of our main contributions:

- **Proof of preconditioned methods' with weight decay convergence**. We derive convergence guarantees for preconditioned methods considering assumptions of smoothness, strongly convex and PL-condition.
- **Research of the loss function** Comparison of accuracy and loss function for AdamW and AdamL2. As a result we saw that AdamW asymptotically converges to a non-zero value
- **Competitive Numerical Results** We investigate the empirical performance of Adam's variation including new one on a variety of standard machine learning tasks, including logistic regression.

2 Problem statement

We want to investigate the convergence speed of the AdamW method and the newly proposed MyAdamW method in machine learning problems, and we also plan to prove the convergence of these methods and investigate the obtained solution.

We consider the unconstrained optimization problem

$$\min_{w \in \mathbb{R}^d} f(w)$$

But we can add regulirazation $r(w)$ – regularization, and solve the unconstrained optimization problem.

$$\min_{w \in \mathbb{R}^d} F(w) = f(w) + r(w)$$

In the convergence algorithm that we study, we will investigate algorithms of the following two kinds. In the first one, the regularization function is taken out separately in the recalculation of model weights, and in the second one, the function is dominated by the inverse "hessian".

Algorithm 2 General scheme for preconditions methods

```
Require: $\eta, \epsilon, f, r$
  while $w$ not converged do
    $t = t + 1$
    $g_t = \nabla f_t(w_{t-1})$
    $D_t = \text{diag}(\sqrt{g_t \odot g_t} + \varepsilon)$                    AdamW
    $D_t$ = $\mathbb{E}[z^T \nabla^2 f(w_{t-1})z]$                    OASIS, where $z$ in Rademacher distribution
    $w_t = w_{t-1} - \eta \cdot g_t D_t^{-1}$
  end while
```

The OASIS algorithm differs from Adam in that it updates the learning rate and calculates the pseudo-Hessian matrix at each step using the expectation and a random variable distributed according to the Rademacher distribution.

This is OASIS, the main difference from Adam is how the Hessian is considered, but when proving convergence, this will not play any role.

Algorithm 3 OASIS

```
Require: $w_0, \eta_0, D_0, \theta_0 = +\infty$
  $w_1 = w_0 - \eta \hat{D_0}^{-1} \nabla F(w_0)$
  for $k = 1, 2, ...$ do
    $D_k = \beta D_{k-1} + (1 - \beta_2) \cdot diag\left(z_k \odot \nabla^2 F(w_k) z_k\right)$
    $(\hat{D}_k)_{ii} = max\{|D_k|_{i,i}; \alpha\}, \forall i = \overline{1, d}$
    $\eta_k = min\{\sqrt{1 + \theta_{k-1}} \cdot \eta_{k-1}; \frac{||w_k - w_{k-1}||_{\hat{D_k}}}{2||\nabla F(w_k) - \nabla F(w_{k-1})||^*_{\hat{D_k}}}\}$
    $\theta_k = \frac{\eta_k}{\eta_{k-1}}$
  end for
```

Now we will show the main differences between the methods of AdamL2, AdamW and MyAdamW – the algorithm that we propose in this article

Algorithm 4 Adam(λ)

```
Require: $\eta, \beta_1, \beta_2, \epsilon, f, r$
  while $\theta$ not converged do
    $t = t + 1$
    $g_t = \nabla f(w_{t-1})$ + $\nabla r(w_{t-1})$                    AdamL2
    $m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$
    $v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$
    $\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$ + $\nabla r(w_{t-1})$                    MyAdamW
    $\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$
    $w_t = w_{t-1} - \eta \cdot \frac{\hat{m}_t}{\sqrt{v_t} + \epsilon}$ - $\eta \nabla r(w_{t-1})$                    AdamW
  end while
```

3 Experiment

We will conduct an experiment in which we compare Adam with added L2 regularization with AdamW, at different learning rate and decoupled weight, also we will compare it with the modified AdamW algorithm where the regularizer is put under the hessian, also we will compare it under the condition that now the minimal eigenvalue of hessian is not epsilon, but decoupled weight hyperparameter.We will conduct the same experiments with SGD and L2 regularization and SGD with decoupled weitght at the same parameters. All experiments will be performed on batches of size 128, on a ResNet18 grid of 200 epochs. All this will be implemented using the PyTorch library on dataset CIFAR10.

4 Theory

Assumption 1 (Convex). *The function f is convex, i.e.* $\forall w, w' \in \mathbb{R}^d$

$$f(w) \geq f(w') + \langle \nabla F(w'), w - w' \rangle \tag{3}$$

Assumption 2 (L-smoothness). *The gradients of F are L-Lipschitz continuous* $\forall w \in \mathbb{R}^d$*, i.e. there exists a constant* $L > 0$ *such that* $\forall w, w' \in \mathbb{R}^d$*,*

$$f(w) \leq f(w') + \langle \nabla f(w'), w - w' \rangle + \frac{L}{2}||w - w'||^2$$

Assumption 3 (Twice differentiable). *The function f is twice continuously differentiable.*

Assumption 4 (μ - strongly convex). *The function f is* μ*-strongly convex, i.e., there exists a constant* $\mu > 0$ *such that* $\forall w, w' \in \mathbb{R}^d$

$$f(w) \geq f(w') + \langle f(w'), w - w' \rangle + \frac{\mu}{2}||w - w'||^2$$

To proof this algorigthm we use simplified form of AdamW algorithm

Algorithm 5 AdamW

```
Require: $r, \varepsilon, f$
  while $w$ not converged do
    $t = t + 1$
    $D_t = \text{diag}(|\nabla f(w_t)|_i)$
    $w_t = w_{t-1} - \eta \cdot \nabla f(w_t) D_t^{-1} - \lambda \nabla r(w_t)$
  end while
```

Theorem 1. *Suppose the Assumption 1, 2, 5 and let* $\varepsilon > 0$ *and let the step-size satisfy*

$$\eta < \frac{2\alpha}{L + l \cdot \alpha}$$

Then, the number of iterations performed by AdamW algorithm, starting from an initial point $w_0 \in \mathbb{R}^d$ *with* $\Delta_0 = \tilde{F}(w_0) - \tilde{F}^*$*, required to obtain and* ε*-approximate solution of the convex problem (link here to problem 1) can be bounded by*

$$T = \mathcal{O}\left(\frac{2\Delta_0 \Gamma \alpha}{(2\alpha - \tilde{L}\eta)\eta\varepsilon}\right)$$

Theorem 2. *Suppose the Assumption 1, 2, 4, 5 and let* $\varepsilon > 0$ *and let the step-size satisfy*

$$\eta \leq \frac{2\alpha}{\tilde{L}}$$

Then, the number of iterations performed by AdamW algorithm, starting from an initial point $w_0 \in \mathbb{R}^d$ *with* $\Delta_0 = \tilde{F}(w_0) - \tilde{F}^*$*, required to obtain and* ε*-approximate solution of the convex problem (link here to problem 1) can be bounded by*

$$T = \mathcal{O}\left(\frac{\ln \frac{\Delta_0}{\epsilon}}{2\mu\eta^2(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})}\right)$$

References

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Pádraig Cunningham, Matthieu Cord, and Sarah Jane Delany. Supervised learning. *Machine learning techniques for multimedia: case studies on organization and retrieval*, pages 21–49, 2008.

Ryan M Rifkin and Ross A Lippert. Notes on regularized least squares. 2007.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010: 19th International Conference on Computational StatisticsParis France, August 22-27, 2010 Keynote, Invited and Contributed Papers*, pages 177–186. Springer, 2010.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. *Advances in neural information processing systems*, 30, 2017.

Guodong Zhang, Chaoqi Wang, Bowen Xu, and Roger Grosse. Three mechanisms of weight decay regularization. *arXiv preprint arXiv:1810.12281*, 2018.

Zhewei Yao, Amir Gholami, Sheng Shen, Mustafa Mustafa, Kurt Keutzer, and Michael Mahoney. Adahessian: An adaptive second order optimizer for machine learning. In *proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 10665–10673, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Andrew Goldberg, Xiaojin Zhu, Alex Furger, and Jun-Ming Xu. Oasis: Online active semi-supervised learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 25, pages 362–367, 2011.

John E Dennis, Jr and Jorge J Moré. Quasi-newton methods, motivation and theory. *SIAM review*, 19(1):46–89, 1977.

Tomaso Poggio, Vincent Torre, and Christof Koch. Computational vision and regularization theory. *Readings in computer vision*, pages 638–643, 1987.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

5 Appendix

Theorem 3. *Suppose the Assumption 1, 2, 5 and let $\varepsilon > 0$ and let the step-size satisfy*

$$\eta < \frac{2\alpha}{L + l \cdot \alpha}$$

Then, the number of iterations performed by AdamW algorithm, starting from an initial point $w_0 \in \mathbb{R}^d$ with $\Delta_0 = \tilde{F}(w_0) - \tilde{F}^*$, required to obtain and ε-approximate solution of the convex problem (link here to problem 1) can be bounded by

$$T = \mathcal{O}\left(\frac{2\Delta_0 \Gamma \alpha}{(2\alpha - \tilde{L}\eta)\eta\varepsilon}\right)$$

Proof. Let's write first assumption for step t and $t+1$:

$$f(w_{t+1}) \leq f(w_t) + \langle \nabla f(w_t), w_{t+1} - w_t \rangle + \frac{L}{2}||w_{t+1} - w_t||^2$$

Okay, by definition for our algorithm we have:

$$w_{t+1} - w_t = -\eta D_t^{-1} \nabla f(w_t) - \eta \nabla r(w_t)$$

and

$$\nabla f(w_t) = \frac{1}{\eta} D^t (w_t - w_{t+1}) - D^t \nabla r(w_t)$$

Okay, now let's replace $\nabla f(w_t)$ and $I \leq \frac{D_t}{\alpha}$

$$f(w_{t+1}) \leq f(w_t) + \langle \frac{1}{\eta} D_t (w_t - w_{t+1}) - D_t \nabla r(w_t), w_{t+1} - w_t \rangle + \frac{L}{2\alpha}||w_{t+1} - w_t||^2_{D_t}$$

$$f(w_{t+1}) \leq f(w_t) + \left(\frac{L}{2\alpha} - \frac{1}{\eta}\right)||w_{t+1} - w_t||^2_{D_t} - \langle D_t \nabla r(w_t), w_{t+1} - w_t \rangle$$

Lets define new variable $\tilde{r} : \nabla \tilde{r} = D_t \nabla r(w_t)$. Then rewrite step using the variable and 5-th assumption.

$$\tilde{r}(w_{t+1}) \leq \tilde{r}(w_t) + \langle \tilde{r}(w_t), w_{t+1} - w_t \rangle + \frac{l}{2}(w_{t+1} - w_t)^T D_t (w_{t+1} - w_t)$$

$$f(w_{t+1}) \leq f(w_t) + \left(\frac{L}{2\alpha} - \frac{1}{\eta}\right)||w_{t+1} - w_t||^2_{D_t} + \tilde{r}(w_t) - \tilde{r}(w_{t+1}) + \frac{l}{2}||w_{t+1} - w_t||^2_{D_t}$$

$\tilde{F}(w) = f(w) + \tilde{r}(w)$, $F(w) = f(w) + r(w)$, $(\tilde{L} = L + l\alpha)$, we get:

$$\tilde{F}(w_{t+1}) \leq \tilde{F}(w_t) + \left(\frac{\tilde{L}}{2\alpha} - \frac{1}{\eta}\right) ||w_{t+1} - w_t||^2_{D_t}$$

$$\left(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha}\right) ||w_{t+1} - w_t||^2_{D_t} \leq \tilde{F}(w_t) - \tilde{F}(w_{t+1})$$

$$\frac{\eta^2(T+1)}{\Gamma}\left(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha}\right) \cdot \min_{k=0,T} ||\nabla f(w_t) + \nabla \tilde{r}(w_t)||^2 \leq \frac{\eta^2}{\Gamma}\left(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha}\right) \cdot \sum_{t=0}^{T} ||\nabla f(w_t) + \nabla \tilde{r}(w_t)||^2 \leq \tilde{F}(w_0) - \tilde{F}(w_*)$$

$$\min_{t=0,T} ||\nabla f(w_t) + \nabla \tilde{r}(w_t)||^2 \leq \frac{(\tilde{F}(w_0) - \tilde{F}(w_*))\Gamma}{(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})\eta^2(T+1)} = \varepsilon$$

$$T + 1 \geq \frac{\Delta_0 \Gamma}{(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})\eta^2 \varepsilon}$$

Then:

$$T = \mathcal{O}\left(\frac{2\Delta_0 \Gamma \alpha}{(2\alpha - \tilde{L}\eta)\eta\varepsilon}\right)$$

□

Theorem 4. *Suppose the Assumption 1, 2, 4, 5 and let $\varepsilon > 0$ and let the step-size satisfy*

$$\eta \leq \frac{2\alpha}{\tilde{L}}$$

Then, the number of iterations performed by AdamW algorithm, starting from an initial point $w_0 \in \mathbb{R}^d$ with $\Delta_0 = \tilde{F}(w_0) - \tilde{F}^*$, required to obtain and ε-approximate solution of the convex problem (link here to problem 1) can be bounded by

$$T = \mathcal{O}\left(\frac{\ln \frac{\Delta_0}{\epsilon}}{2\mu\eta^2(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})}\right)$$

Proof. Assume

$$\nabla \tilde{F} = \nabla f + \nabla \tilde{r} \tag{4}$$

$$L + ||D_t|| l = \tilde{L} \tag{5}$$

$$w_{t+1} - w_t = -\eta D_t^{-1} \nabla r(w_t) - \eta \nabla r(w_t) = -\eta D_t^{-1}(\nabla f + \nabla \tilde{r})(w_t) = -\eta D_t^{-1} \nabla \tilde{F}(w_t)$$

Then we write $\tilde{L}$-smoothness for $\tilde{F}$

$$\tilde{F}(w_{t+1}) - \tilde{F}(w_t) \leq \langle \nabla \tilde{F}(w_t), w_{t+1} - w_t \rangle + \frac{\tilde{L}}{2} ||w_{t+1} - w_t||^2$$

$$\tilde{F}(w_{t+1}) - \tilde{F}(w_t) \leq -\langle \frac{1}{\eta} D_t(w_{t+1} - w_t), w_{t+1} - w_t \rangle + \frac{\tilde{L}}{2} ||w_{t+1} - w_t||^2 = (\frac{\tilde{L}}{2\alpha} - \frac{1}{\eta}) ||w_{t+1} - w_t||^2_{D_t}$$

$$= (\frac{\tilde{L}}{2\alpha} - \frac{1}{\eta}) ||w_{t+1} - w_t||^2_{D_t} = (\frac{\tilde{L}}{2\alpha} - \frac{1}{\eta}) || - \eta D_t^{-1} \nabla \tilde{F}(w_t)||^2_{D_t} \leq (\frac{\tilde{L}}{2\alpha} - \frac{1}{\eta}) \eta^2 ||\nabla \tilde{F}(w_t)||^2_{D_t^{-1}}$$

Then we use PL-condition for the function $\tilde{F}$:

$$||\nabla \tilde{F}(w_t)||^2_{D_t^{-1}} \geq 2\mu(\tilde{F}(w_t) - \tilde{F}^*)$$

$$\tilde{F}(w_t) - F^* \geq \tilde{F}(w_{t+1}) - \tilde{F}^* + (\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})\eta^2 2\mu(\tilde{F}(w_t) - \tilde{F}^*) = \left(1 + 2\mu\eta^2(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})\right)(\tilde{F}(w_{t+1}) - \tilde{F}^*)$$

$$\epsilon \geq \Delta_0 \left(1 + 2\mu\eta^2(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})\right)^{-T} \geq (\tilde{F}(w_T) - \tilde{F}^*)$$

$$T = \frac{\ln\frac{\Delta_0}{\epsilon}}{\ln(1 + 2\mu\eta^2(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha}))} \approx \frac{\ln\frac{\Delta_0}{\epsilon}}{2\mu\eta^2(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})}$$

Then:

$$T = \mathcal{O}\left(\frac{\ln\frac{\Delta_0}{\epsilon}}{2\mu\eta^2(\frac{1}{\eta} - \frac{\tilde{L}}{2\alpha})}\right)$$

□