AIMS African Institute for Mathematical Sciences RWANDA

Adam Optimization Algorithm

Keziah Naggita

African Master's In Machine Intelligence

knaggita@aimsammi.org

October 19, 2018

Outline

1. Introduction
2. Adam Algorithm
3. Pros and Cons of the Adam
4. Adam versus Adagrad and RMSprop
5. Future Work
6. Supplementary Literature
7. Recap
8. References

Optimisation

Optimisation is concerned with finding the minimizer of a function subject to constraints.

An optimizer minimises/maximises the objective function

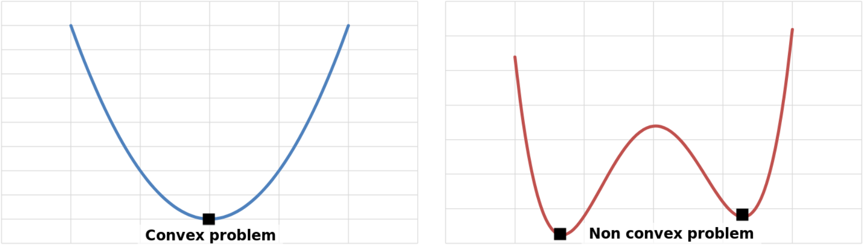
In most cases, the optimizer minimises error in the model(difference between predicted data and expected data)

Optimisation problems

Classification of Optimization problems based on Objective Function

- Convex problems: Local minima = Global minima
- Non convex problems: Several local minimas
  Examples include: DNNs, PCA

Convex and Non Convex Problems



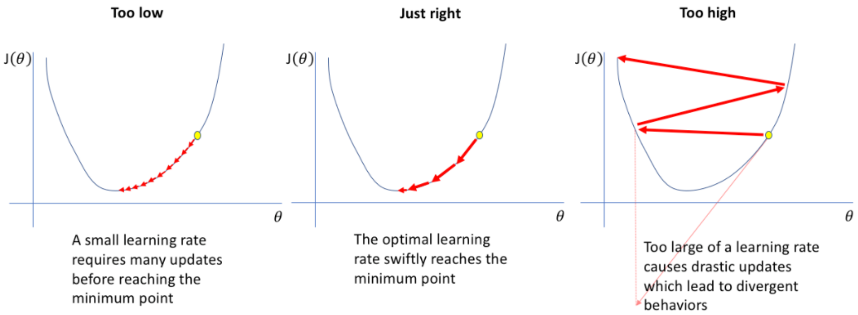
Cornell optimization lecture [1]

Learning Rate

Learning Rate also called a step size is a used for adjusting the weights with respect to the objective function.

Learning rates determine how quickly we reach the local minima however choosing a good one is rather hard and mostly random.

Learning Rate



Too low

A small learning rate requires many updates before reaching the minimum point

Just right

The optimal learning rate swiftly reaches the minimum point

Too high

Too large of a learning rate causes drastic updates which lead to divergent behaviors

Setting the learning rate of your NN [2]

Play with the learning rate to see it's effect on convergence

`https://developers.google.com/machine-learning/crash-course/reducing-loss/learning-rate`

Learning Rate

Ways of setting a learning rate include the following;

- Adaptive learning rate
- Annealing learning rate [3]
- Learning rate schedules [4]
- Differential learning rate: ”It is a method where you set different learning rates to different layers in the network during training.” [4]

Adam

Adam is not an acronym. It is derived from adaptive moment estimation.

Properties of Adam

- First order gradient based optimisation algorithm
- Has 2 moments of the Gradient
- Has a bias correction term
- Updates are directly estimated using a running average of first and second moment of the gradient.

Adam Parameters

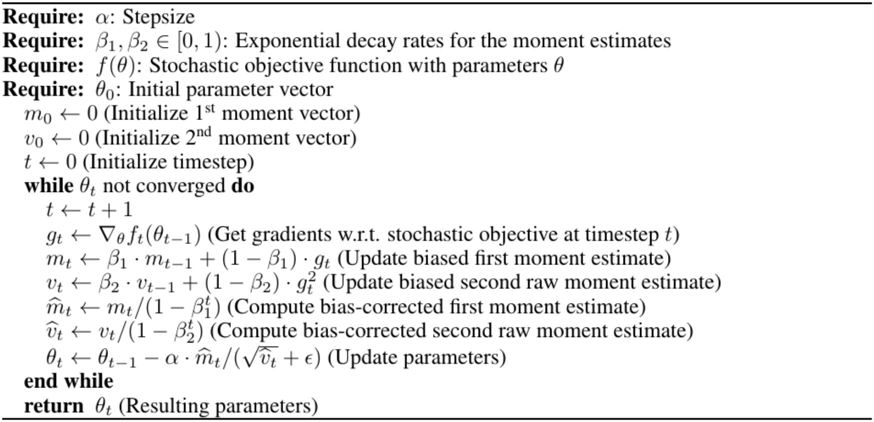
$\boldsymbol{\beta}$**1**: It is a hyper-parameter that decays the running average of the gradient

$\boldsymbol{\beta}$**2**: It is the hyper-parameter that decays the running average of the square of gradient

$\boldsymbol{\alpha}$: It is the step size or learning rate

$\boldsymbol{\epsilon}$: It is prevents division by zero error

Adam Algorithm



```
Require: $\alpha$: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters $\theta$
Require: $\theta_0$: Initial parameter vector
  $m_0 \leftarrow 0$ (Initialize 1st moment vector)
  $v_0 \leftarrow 0$ (Initialize 2nd moment vector)
  $t \leftarrow 0$ (Initialize timestep)
  while $\theta_t$ not converged do
    $t \leftarrow t + 1$
    $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep $t$)
    $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
    $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
    $\widehat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
    $\widehat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
    $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \widehat{m}_t / (\sqrt{\widehat{v}_t} + \epsilon)$ (Update parameters)
  end while
  return $\theta_t$ (Resulting parameters)
```

Algorithm from the paper [5]

Adam Demo and Update rule

Demo and Explanation of the algorithm

Comparison with Adagrad and RMSprop

RMSprop and Adam

- While RMSProp first rescales gradient and then applies momentum, Adam first applies momentum and then rescale the gradient.
- While Adam has a bias correction term, RMSProp doesn't.

Adagrad and Adam

- The biggest difference is still the bias correction term. If the bias one hyper-parameter is set to 0, and bias 2 becomes infinitesimally small, then Adam will be Adagrad

Pros of Adam

- Straightforward to implement
- Computationally efficient
- Little memory requirements
- Invariant to diagonal rescaling of the gradients
- Suitable for large data/parameter problems
- Appropriate for non-stationary objectives
- Appropriate for problems with very noisy/or sparse gradients.
- The hyper-parameters have intuitive interpretation and typically require little tuning

cited from Adam Paper[5]

Limitations of Adam

Below are some of the limitations of Adam

1. Poor generalization performance especially for training DNNs
2. Weight decay regularization issues

Hyper-parameter tuning is still a challenge.

Future Works

Research focused on improving Adam include the following;

1. Accelerated Adam (AAdam) [7]
2. Nestrov Accelerated Adam (NAdam) [8]
3. AdamW [9]
4. SWATS, a simple strategy which switches from Adam to SGD [10]
5. Adam with warm restarts (AdamWR) [11]
6. Normalised Direction preserving Adam (ND-Adam) [12]

Conclusion

Adam performs relatively well in comparison to other adaptive optimization algorithms

It is now used as a benchmark for doing research in optimization of non-convex problems.

A lot of work is being done in finding the most optimal algorithm, you can read more about hyper-parameter tuning, regularisation and optimisation and **hopefully come up with an algorithm better than all the current ones.**

Supplementary Resources

In addition to the references, below are clickable links to some resources I found helpful.

- Blog: Optimising gradient descent
- Webpage: Types of optimization algorithms used in NNs
- Webpage: Estimating optimal learning rate
- Blog: Adam optimisation algorithm
- Course: Third NNs course at Stanford
- Blog: Comparison of SGD and adaptive learning algorithms

Supplementary Resources ...

- Cousera: Hyper-parameter tuning, regularization and optimization
- Youtube: Stanford NNs class
- Coursera: Adaptive learning rates
- GitHub: Visualisation of different Optimisation algorithms
- GitHub: Overview of all Optimisation Algorithms
- GitHub: Deep Learning AI summary
- GitHub: Deep Learning papers reading roadmap

Recap

1. Concave and Convex Problems
2. Learning Rate
3. Adam Algorithm
4. Pros and Cons of the Algorithm
5. Applicability of the Algorithm
6. Adam versus Adagrad and RMSprop
7. Future Work
8. Supplementary Resources
9. References

[1] www.cs.cornell.edu/, “Non-Convex Optimization,” 2017.

[2] Jeremy Jordan, “Setting the learning rate of your neural network.,” 2018.

[3] Stanford CS231n, “CS231n Convolutional Neural Networks for Visual Recognition.”

[4] Hafidz Zulkifli, “Understanding Learning Rates and How It Improves Performance in Deep Learning,” 2018.

[5] D. P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization,” *http://arxiv.org/abs/1412.6980*, 2014.

[6] S. J. Reddi, S. Kale, and S. Kumar, “On the convergence of adam and beyond,” *Iclr*, 2018.

[7] A. Tato and R. Nkambou, “Improving Adam Optimizer,” pp. 1–4, 2018.

[8] T. Dozat, “Incorporating Nesterov Momentum into Adam,” *ICLR Workshop*, 2016.

[9] Sylvain Gugger and Jeremy Howard, “AdamW and Super-convergence is now the fastest way to train neural nets · fast.ai,” 2018.

[10] N. S. Keskar and R. Socher, "Improving Generalization Performance by Switching from Adam to SGD," no. 1, 2017.

[11] I. Loshchilov and F. Hutter, "Fixing Weight Decay Regularization in Adam," vol. 100, 2017.

[12] Z. Zhang, *Improved Adam Optimizer for Deep Neural Networks*. PhD thesis.