Decoupled Weight Decay Regularization

2020. 01. 30

Taehwan Kim

ghks0830@kaist.ac.kr

KAIST

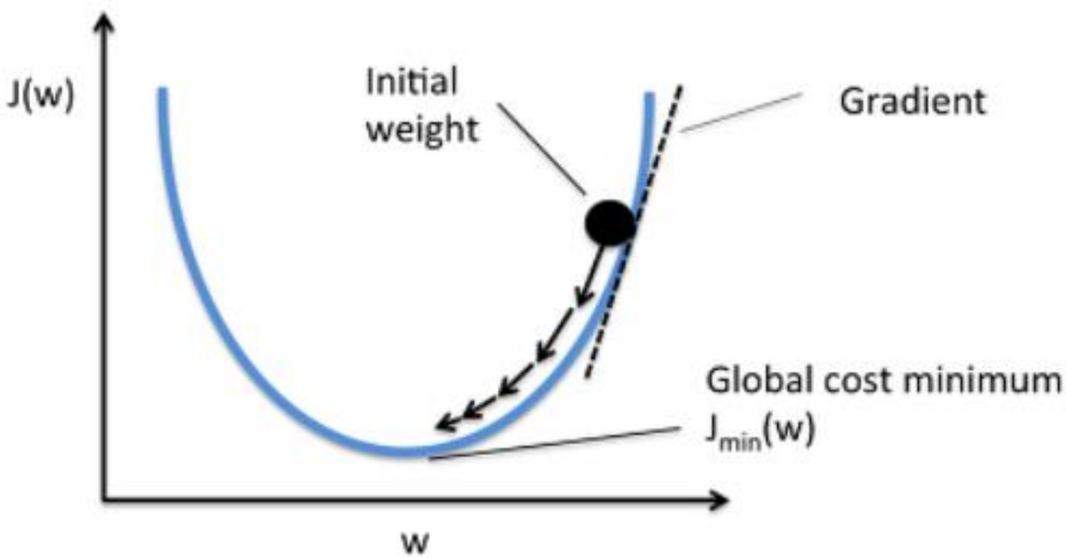
MLILAB Machine Learning & Intelligence

Content

- Background
- Adam and (decoupled) weight decay
- Experiment Results
- TODO

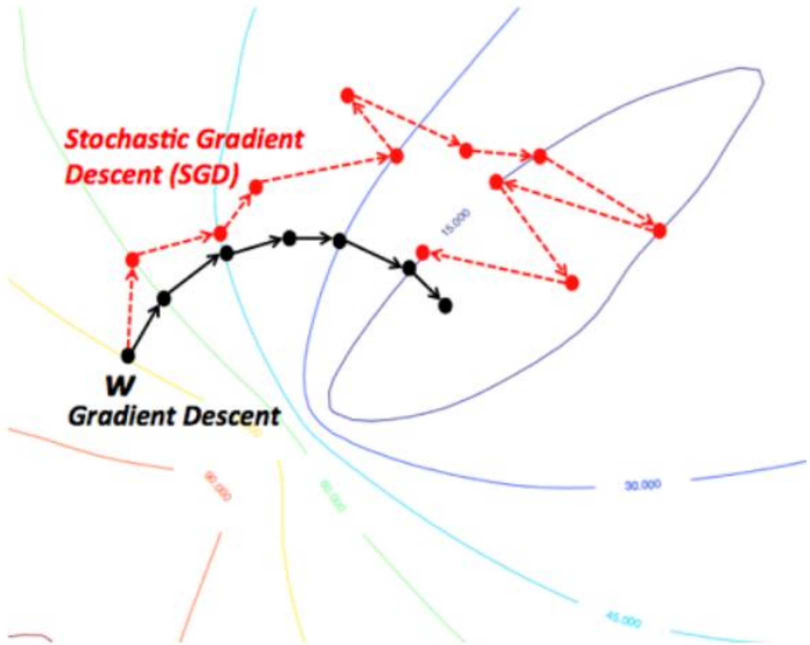
Gradient Descent (GD)

- Minimize the Loss function, $J(w)$ by using the gradient of weights.
- $while\ i < \ num\_epoch$:
  - $w_{i+1} \leftarrow w_i - \eta \nabla_w J(w_i)$ (full-batch)



Stochastic Gradient Descent (SGD)

- Evaluate a gradient of loss only on random subset of samples
- $while\ i < \ num\_epoch$:
    - $for\ batch_j\ in\ mini\_batch\_list$:
        - $w_{i,j+1} \leftarrow w_{i,j} - \eta \nabla_w J(w_{i,j})$



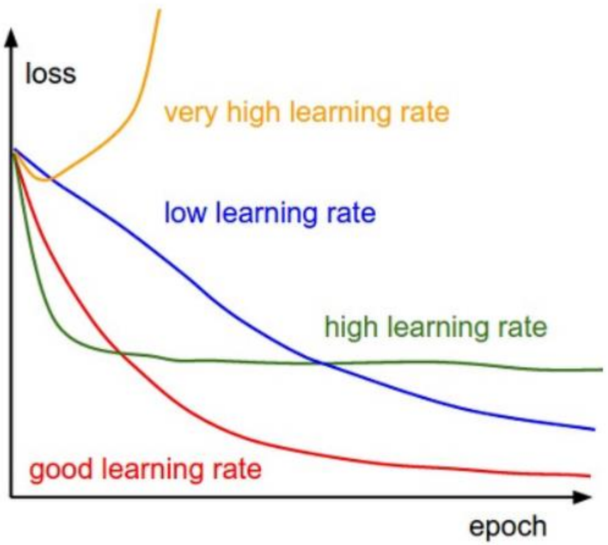
Issues on SGD (Step Direction)

- Oscillation Problem on SGD

- Momentum
  - $v_t = \gamma v_{t-1} + \eta \nabla_w J(w)$
  - $w = w - v_t$

Issues on SGD (Over / Under -fitting)

- Regularization (= Weight Decay)
  - Penalizes large weights to avoid overfitting
  - L_2 Regularization
    - $\widetilde{J(w)} = J(w) + \lambda * l_2(w) = J(w) + \lambda \frac{1}{2} \left|\left|w\right|\right|_2^2$
    - $w = w - \eta \nabla_w J(w) - \lambda \left|\left|w\right|\right|_2$



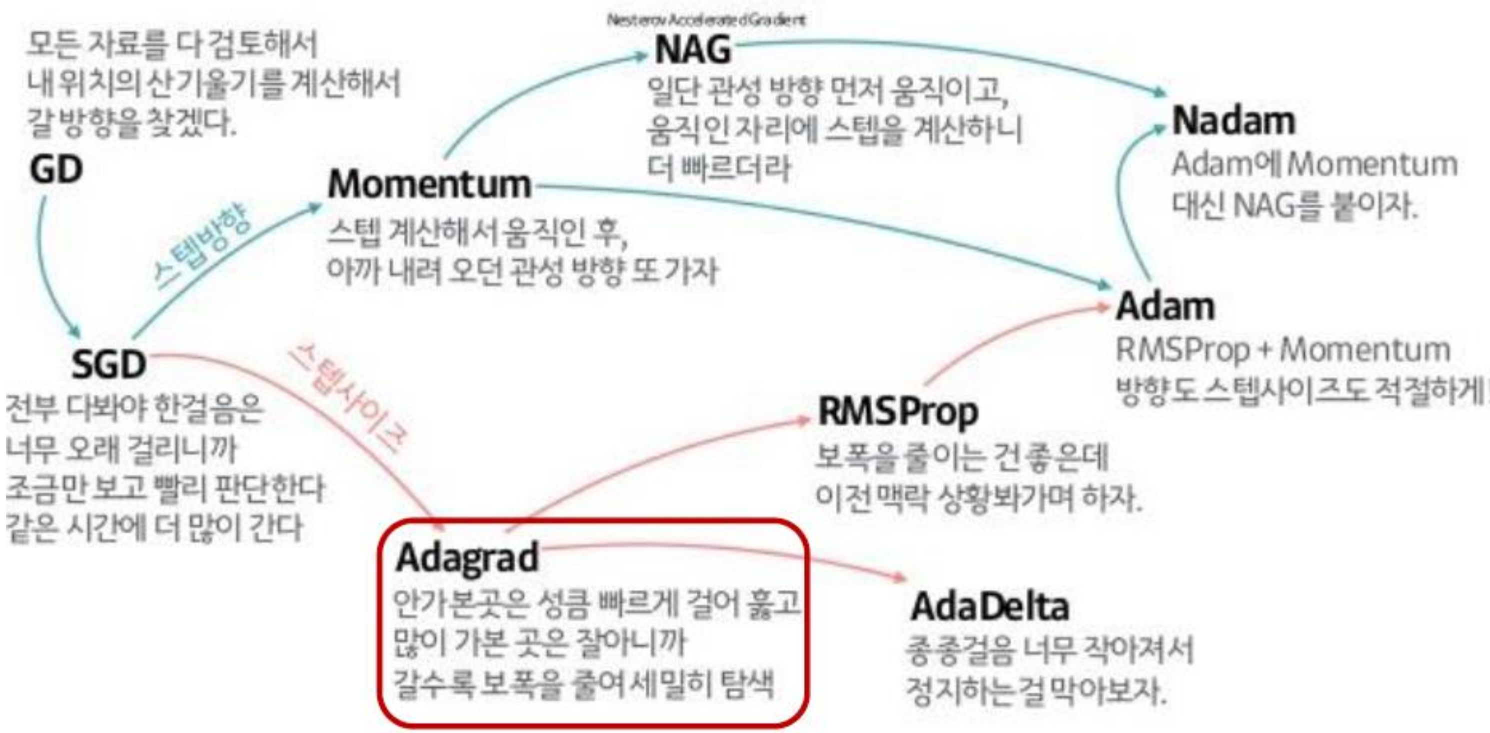
- Step Size (= Learning Rate) vs Batch Size
- RMSProp
  - Use exponential moving averages and Effective Learning rate
  - $G = \gamma G + (1 - \gamma)\left(\nabla_w J(w_t)\right)^2$
  - $w = w - \frac{\eta}{\sqrt{G + \varepsilon}} \cdot \nabla_w J(w_t)$

Adam (Adaptive Moment Estimation)

- RMSProp + exponential moving averages of 1_{st} momentum
  - $m_0 = 0, v_0 = 0$
  - $m_t = \beta_1 m_{t-1} + (1-\beta_1)\nabla_w J(w_{t-1})$
  - $v_t = \beta_2 v_{t-1} + (1-\beta_2)(\nabla_w J(w_{t-1}))^2$
  - $\widehat{m_t} = \frac{m_t}{1-\beta_1^t}$
  - $\widehat{v_t} = \frac{v_t}{1-\beta_2^t}$
  - $w_{t+1} = w_t - \frac{\eta}{\sqrt{\widehat{v_t}+\varepsilon}} \cdot \widehat{m_t}$

Overview

- Optimization methods in deep learning



Adam with L_2 regularization

- RMSProp + exponential moving averages of 1_{st} momentum + L_2 regularization
  - $m_0 = 0, v_0 = 0$
  - $m_t = \beta_1 m_{t-1} + (1 - \beta_1)(\nabla_w J(w_{t-1}) + \lambda \left|\left|w_t\right|\right|_2)$
  - $v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\nabla_w J(w_{t-1}) + \ \lambda \left|\left|w_t\right|\right|_2)^2$
  - $\widehat{m_t} = \frac{m_t}{1-\beta_1^t}$
  - $\widehat{v_t} = \frac{v_t}{1-\beta_2^t}$
  - $w_{t+1} = w_t - \frac{\eta}{\sqrt{\widehat{v_t}+\varepsilon}} \cdot \widehat{m_t} = \mathrm{w_t} - \frac{\eta \cdot (\beta_1 m_{t-1} + (1-\beta_1)(\nabla_w J(w_{t-1}) + \lambda \left|\left|w_t\right|\right|_2))}{\sqrt{\widehat{v_t}+\varepsilon}}$

- We can see that L_2 regularization is normalized by v_t.
- Therefore, if the gradient of a certain weight is large (or is changing a lot)
- $\rightarrow v_t$ is too large → the weight is regularized less than weights with small → slowly changing gradients!

Decoupled Weight Decay Regularization

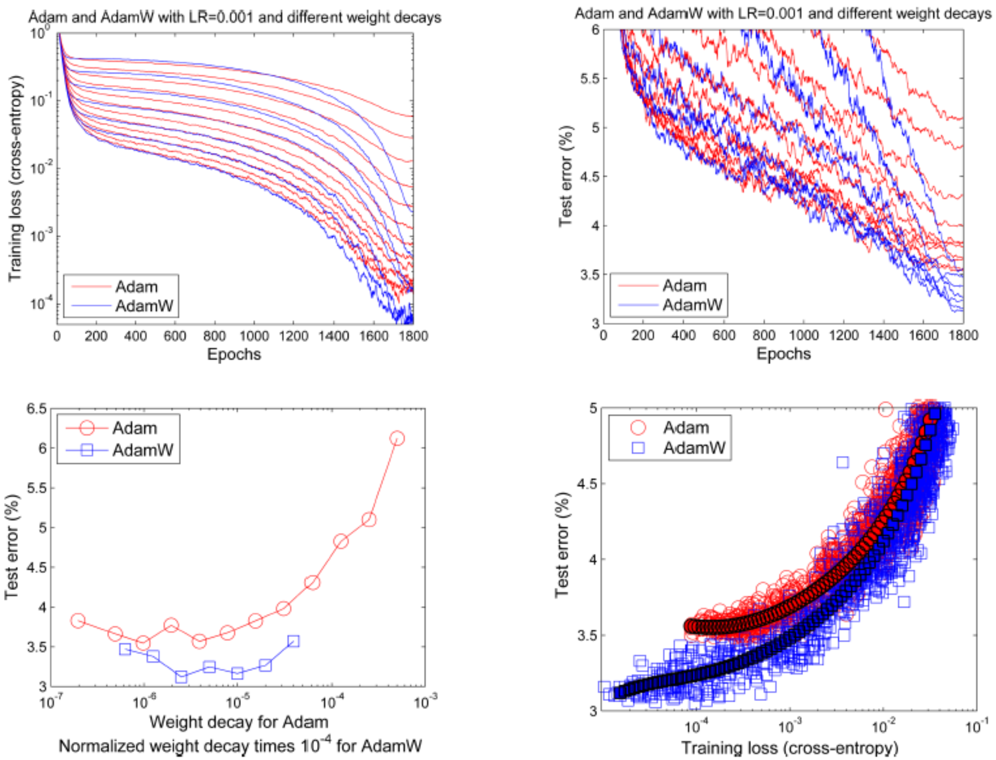
- **Algorithm 2** Adam with L_2 regularization and Adam with decoupled weight decay (AdamW)

  given $\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}, \lambda \in \mathbb{R}$
  initialize time step $t \leftarrow 0$, parameter vector $\boldsymbol{\theta}_{t=0} \in \mathbb{R}^n$, first moment vector $\boldsymbol{m}_{t=0} \leftarrow \boldsymbol{0}$, second moment vector $\boldsymbol{v}_{t=0} \leftarrow \boldsymbol{0}$, schedule multiplier $\eta_{t=0} \in \mathbb{R}$
  repeat
      $t \leftarrow t + 1$
      $\nabla f_t(\boldsymbol{\theta}_{t-1}) \leftarrow \text{SelectBatch}(\boldsymbol{\theta}_{t-1})$ ▷ select batch and return the corresponding gradient
      $\boldsymbol{g}_t \leftarrow \nabla f_t(\boldsymbol{\theta}_{t-1}) \; +\lambda\boldsymbol{\theta}_{t-1}$
      $\boldsymbol{m}_t \leftarrow \beta_1 \boldsymbol{m}_{t-1} + (1 - \beta_1)\boldsymbol{g}_t$ ▷ here and below all operations are element-wise
      $\boldsymbol{v}_t \leftarrow \beta_2 \boldsymbol{v}_{t-1} + (1 - \beta_2)\boldsymbol{g}_t^2$
      $\hat{\boldsymbol{m}}_t \leftarrow \boldsymbol{m}_t / (1 - \beta_1^t)$ ▷ β_1 is taken to the power of t
      $\hat{\boldsymbol{v}}_t \leftarrow \boldsymbol{v}_t / (1 - \beta_2^t)$ ▷ β_2 is taken to the power of t
      $\eta_t \leftarrow \text{SetScheduleMultiplier}(t)$ ▷ can be fixed, decay, or also be used for warm restarts
      $\boldsymbol{\theta}_t \leftarrow \boldsymbol{\theta}_{t-1} - \eta_t \left(\alpha \hat{\boldsymbol{m}}_t / (\sqrt{\hat{\boldsymbol{v}}_t} + \epsilon) \; +\lambda\boldsymbol{\theta}_{t-1} \right)$
  until *stopping criterion is met*
  return optimized parameters $\boldsymbol{\theta}_t$

- Decoupled weight decay can fix this!

Experiment Results

- Adam with L_2 regularization vs AdamW
  - Resnet with CIFAR-100



- Better Generalization of AdamW

Experiment Results

- AdamWR with warm restarts for better anytime performance

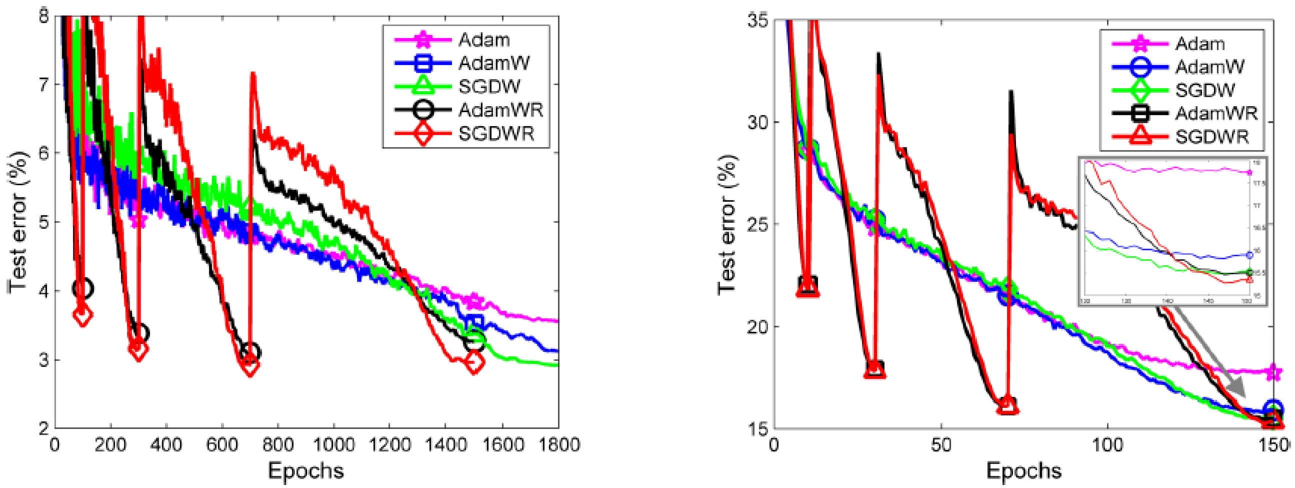


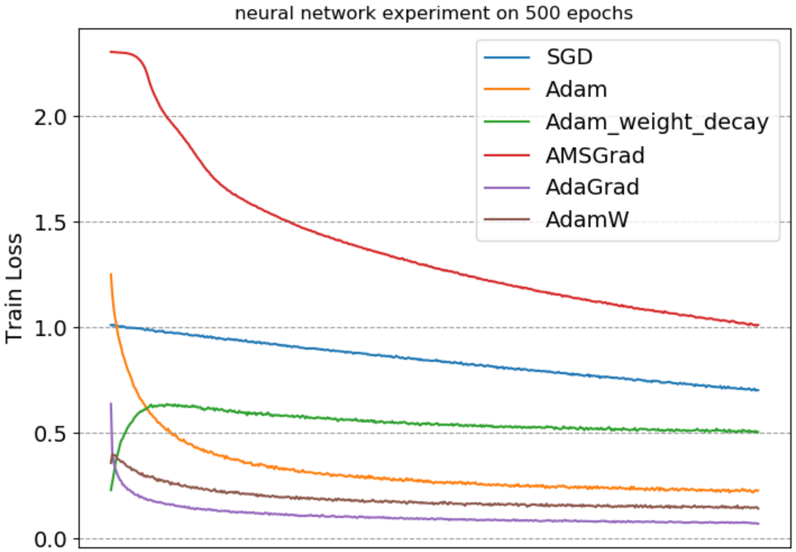
Figure 4: Top-1 test error on CIFAR-10 (left) and Top-5 test error on ImageNet32x32 (right). For a better resolution and with training loss curves, see SuppFigure 5 and SuppFigure 6 in the supplementary material.

MLILAB Machine Learning & Intelligence

TODO

- **Implement AdamW**

- ~ Experiment 4.1
  - Model : pretrained VGG11
  - Dataset : CIFAR100
  - Compare three Optimizers
    - SGD
    - Adam with L2 regularization
    - AdamW (Adam with decoupled weight decay)
  - Plot the learning curve and generalization result



Reference

- Decoupled Weight Decay Regularization, ICLR 2019