Adaptive Gradient Descent (AdaGrad)

- Key idea: make the step size adaptive to each coordinate of the gradient.
- Update rules:

$$z_k = z_{k-1} + G_k \odot G_k$$
$$\tilde{z}_k(i) = 1/\sqrt{z_k(i)}, \quad i = 1, \cdots, n$$
$$x_{k+1} = x_k - \eta\,(\tilde{z}_k \odot G_k)$$

  where $z_0 = \epsilon > 0$ is very small, e.g., 10^{-6} and *eta* is set to a small value, e.g., 0.05 or 0.01.
- Issue: the "effective" step size vector may become too small over iterations and hence may slow down convergence.

Adaptive Movement Estimation (ADAM)

- Key idea: combine momentum and adaptive step sizes to improve convergence of gradient descent.
- Update rules:

$$
\begin{aligned}
m_k &= \beta_1 m_{k-1} + (1-\beta_1)\, G_k && \text{(1st moment)}\\
z_k &= \beta_2 z_{t-1} + (1-\beta_2)\, (G_k \odot G_k) && \text{(2nd moment)}\\
\hat{m}_k &= \frac{m_k}{1-\beta_1^k}, \quad \hat{z}_k = \frac{z_k}{1-\beta_2^k} && \text{(bias correction)}\\
\tilde{z}_k(i) &= 1/\left(\sqrt{\hat{z}_k(i)} + \epsilon\right)\\
x_{k+1} &= x_k - \eta\,(\tilde{z}_k \odot \hat{m}_k)\,, \eta > 0
\end{aligned}
$$

  where $m_0 = 0$ and $z_0 = \epsilon$ is very small.
- Typical values for the hyper-parameters: $\beta_1 = 0.9, \beta_2 = 0.999$.
- ADAM can fail to converge in some convex settings due to the adaptive learning rate increasing.

AMSGrad Modification of ADAM

- Key idea: The AMSGrad modification uses the maximum of all historical z_k terms to prevent adaptive step sizes from increasing.
- Update rules:

$$
\begin{aligned}
m_k &= \beta_1 m_{k-1} + (1-\beta_1)\, G_k \\
z_k &= \beta_2 z_{k-1} + (1-\beta_2)\, (G_k \odot G_k) \\
\hat{m}_k &= \frac{m_k}{1-\beta_1^k}, \\
\hat{z}_k &= \max\{\hat{z}_{k-1}, z_k\} \\
\tilde{z}_k(i) &= 1/\sqrt{\hat{z}_k(i)+\epsilon} \\
x_{k+1} &= x_k - \eta\, (\tilde{z}_k \odot \hat{m}_k)\, , \eta > 0
\end{aligned}
$$

ADAM-W Modification

- Key idea: introduce a weight decay in the update.
- Update rule: $x_{k+1} = (1 - \eta\lambda)x_k - \eta\,(\tilde{z}_k \odot \hat{m}_k)$.

Stochastic Gradient Descent

- Key idea: Calculate the gradient on mini-batches of the full dataset.
- Update rule: $x_{k+1} = x_k - \eta_k \left[\nabla f_{I_k}(x_k)\right]$ where $\nabla f_{I_k}(x_k) = \frac{1}{s}\sum_{i \in I_k} \nabla f_i(x_k)$.
- Benefit of SGD
  - Reduce computational requirements, both in terms of memory and calculations.
  - Its built-in randomness can "escape" local minima and saddle points.
- We can not guarantee that the direction selected by SGD is necessarily a decent one. But it is a descent direction in expectation.