

NN Optimization Problem I

- Recall that the NN optimization problem is

$$\min_{\theta} f(\theta)$$

where

$$f(\theta) = \frac{1}{2C} \theta^T \theta + \frac{1}{I} \sum_{i=1}^I \xi(z^{L+1,i}(\theta); \mathbf{y}^i, Z^{1,i})$$

- Let's simplify the loss part

$$f(\theta) = \frac{1}{2C} \theta^T \theta + \frac{1}{I} \sum_{i=1}^I \xi(\theta; \mathbf{y}^i, Z^{1,i})$$

- The issue now is how to do the minimization

Gradient Descent I

- This is one of the most used optimization method
- First-order approximation

$$f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) \approx f(\boldsymbol{\theta}) + \nabla f(\boldsymbol{\theta})^T \Delta\boldsymbol{\theta},$$

where

$$\nabla f(\boldsymbol{\theta}) = \begin{bmatrix} \frac{\partial f(\boldsymbol{\theta})}{\partial \theta_1} \\ \vdots \\ \frac{\partial f(\boldsymbol{\theta})}{\partial \theta_n} \end{bmatrix}$$

is the gradient of $f(\boldsymbol{\theta})$

Gradient Descent II

- Solve

$$\begin{aligned} \min_{\Delta\theta} \quad & \nabla f(\theta)^T \Delta\theta \\ \text{subject to} \quad & \|\Delta\theta\| = 1 \end{aligned} \quad (1)$$

to find a direction $\Delta\theta$

- The constraint $\|\Delta\theta\| = 1$ is needed. Otherwise, the above sub-problem goes to $-\infty$
- The solution of (1) is

$$\Delta\theta = -\frac{\nabla f(\theta)}{\|\nabla f(\theta)\|} \quad (2)$$

Gradient Descent III

- This is called the **steepest descent direction**
- However, because we only consider an approximation

$$f(\theta + \Delta\theta) \approx f(\theta) + \nabla f(\theta)^T \Delta\theta$$

we may not have the strict decrease of the function value

- That is,

$$f(\theta) < f(\theta + \Delta\theta)$$

may occur

Gradient Descent IV

- But in general we need the descent property to get the convergence
- We have

$$f(\boldsymbol{\theta} + \alpha \Delta \boldsymbol{\theta}) = f(\boldsymbol{\theta}) + \alpha \nabla f(\boldsymbol{\theta})^T \Delta \boldsymbol{\theta} + \frac{1}{2} \alpha^2 \Delta \boldsymbol{\theta}^T \nabla^2 f(\boldsymbol{\theta}) \Delta \boldsymbol{\theta} + \dots,$$

where

$$\nabla^2 f(\boldsymbol{\theta}) = \begin{bmatrix} \frac{\partial^2 f}{\partial \theta_1 \partial \theta_1} & \cdots & \frac{\partial^2 f}{\partial \theta_1 \partial \theta_n} \\ \vdots & & \vdots \\ \frac{\partial^2 f}{\partial \theta_n \partial \theta_1} & \cdots & \frac{\partial^2 f}{\partial \theta_n \partial \theta_n} \end{bmatrix}$$

Gradient Descent V

is the Hessian of $f(\theta)$

- If

$$\nabla f(\theta)^T \Delta \theta < 0,$$

then a small enough α can ensure

$$f(\theta + \alpha \Delta \theta) < f(\theta)$$

- Thus in optimization for any direction (not necessarily the steepest descent direction), it is called a **descent direction** if

$$\nabla f(\theta)^T \Delta \theta < 0$$

Gradient Descent VI

- The direction chosen in (2) is a descent direction:

$$-\nabla f(\boldsymbol{\theta})^T \frac{\nabla f(\boldsymbol{\theta})}{\|\nabla f(\boldsymbol{\theta})\|} < 0.$$

Line Search I

- We have seen that we need a step size α such that

$$f(\boldsymbol{\theta} + \alpha\Delta\boldsymbol{\theta}) < f(\boldsymbol{\theta})$$

- In optimization this is called a **line search** procedure
- Exact line search

$$\min_{\alpha} f(\boldsymbol{\theta} + \alpha\Delta\boldsymbol{\theta})$$

This is a one-dimensional optimization problem

- In practice, people use **backtracking line search**

Line Search II

- We check

$$\alpha = 1, \beta, \beta^2, \dots$$

with $\beta \in (0, 1)$ until

$$f(\boldsymbol{\theta} + \alpha \Delta \boldsymbol{\theta}) < f(\boldsymbol{\theta}) + \nu \nabla f(\boldsymbol{\theta})^T (\alpha \Delta \boldsymbol{\theta})$$

- Here

$$\nu \in (0, \frac{1}{2})$$

- The convergence is well established.

Line Search III

- For example, if the steepest descent direction is used with the backtracking line search, Corollary 1.1.2 at <https://sites.math.washington.edu/~burke/crs/408/notes/nlp/line.pdf> shows that for every limit point $\bar{\theta}$ of a convergent subsequence of $\{\theta^k\}$, where k is the iteration index, we have

$$\nabla f(\bar{\theta}) = \mathbf{0}$$

- This means we can reach a **stationary point** of a non-convex problem

Line Search IV

- The back-tracking line search procedure is simple and useful in practice

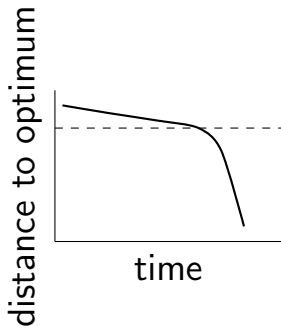
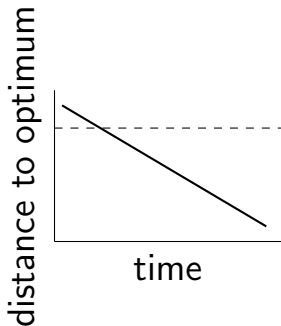
Practical Use of Gradient Descent I

- It is known that the convergence is slow for difficult problems
- Thus in many optimization applications, methods of using second-order information (e.g., quasi Newton or Newton) are preferred

$$f(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) \approx f(\boldsymbol{\theta}) + \nabla f(\boldsymbol{\theta})^T \Delta\boldsymbol{\theta} + \frac{1}{2} \Delta\boldsymbol{\theta}^T \nabla^2 f(\boldsymbol{\theta}) \Delta\boldsymbol{\theta}$$

- These methods have fast final convergence
- An illustration (modified from Tsai et al. (2014))

Practical Use of Gradient Descent II



Slow final convergence Fast final convergence

- But fast final convergence may not be needed in machine learning

Practical Use of Gradient Descent III

- The reason is that an optimal solution θ^* may not lead to the best model
- We will discuss such issues again later

References I

C.-H. Tsai, C.-Y. Lin, and C.-J. Lin. Incremental and decremental training for linear classification. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014. URL <http://www.csie.ntu.edu.tw/~cjlin/papers/ws/inc-dec.pdf>.