Optimization Theory and Algorithm	**Lecture 6 - 05/14/2021**

Lecture 6

Lecturer:Xiangyu Chang *Scribe: Xiangyu Chang*

Edited by: Xiangyu Chang

1 Gradient Descent with Line Search

Let us consider a unconstrained optimization problem

$$\min_{\mathbf{x}} f(\mathbf{x}) \tag{1}$$

where $\mathbf{x} \in dom(f) \subseteq \mathbb{R}^n$, f is a continuous and F-differential function, i.e. $f \in C^1$.

Basic Idea: The algorithm we need is

$$\mathbf{x}^{t+1} = \mathbf{x}^t + s\mathbf{d}, \text{ such that } f(\mathbf{x}^{t+1}) \leq f(\mathbf{x}^t),$$

where $\mathbf{d} \in \mathbb{R}^n$ is a *descent direction* and $s \in \mathbb{R}$ is referred as to the *step size* of the descent algorithm. **Note that s is also called learning rate in the machine learning or deep learning community.**

Given the descent algorithm, we need to determinate that

- How to choose the descent direction?
- How to choose the step size?

Insights: According to the Taylor expansion, we have that

$$f(\mathbf{x}^{t+1}) = f(\mathbf{x}^t) + \langle \nabla f(\mathbf{x}^t), \mathbf{x}^{t+1} - \mathbf{x}^t \rangle + o(\|\mathbf{x}^{t+1} - \mathbf{x}^t\|), \tag{2}$$

where $\lim\limits_{\mathbf{x}^{t+1} \to \mathbf{x}^t} \frac{o(\|\mathbf{x}^{t+1}-\mathbf{x}^t\|)}{\|\mathbf{x}^{t+1}-\mathbf{x}^t\|} = 0$. You can review the little "o" notation by yourself. Furthermore,

$$f(\mathbf{x}^{t+1}) = f(\mathbf{x}^t) + s\langle \nabla f(\mathbf{x}^t), \mathbf{d} \rangle + o(s\|\mathbf{d}\|). \tag{3}$$

Q: Could you please guess a descent direction?

Let $\mathbf{d} = -\nabla f(\mathbf{x}^t)$, then

$$f(\mathbf{x}^{t+1}) = f(\mathbf{x}^t) - s\|\nabla f(\mathbf{x}^t)\|^2 + o(s\|\nabla f(\mathbf{x}^t)\|) \approx f(\mathbf{x}^t) - s\|\nabla f(\mathbf{x}^t)\|^2 \leq f(\mathbf{x}^t), \tag{4}$$

when s is "small enough".

The iterative algorithm choosing the descent direction $\mathbf{d} = -\nabla f(\mathbf{x}^t)$ is referred to as Gradient Descent Method. The remaining question is to find the proper step size s.

The first method is the *Exact Line Search*:

$$s_t = \arg\min_{s \in \mathbb{R}} f(\mathbf{x}^t - s \cdot \nabla f(\mathbf{x}^t)). \tag{5}$$

The second method is the *Backtracking Line Search:*

Algorithm 1 Backtracking Line Search

Input: given a initial step size s_0, two constant $0 < \alpha, \beta < 1$ and index $k = 0$
while $f(\mathbf{x}^t - s_k \cdot \nabla f(\mathbf{x}^t)) > f(\mathbf{x}^t) - \alpha s_k \|\nabla f(\mathbf{x}^t)\|^2$ **do**
    $s_{k+1} := \beta s_k$, and $k := k + 1$.
end while
Output: s_t that breaks the stop condition.

Algorithm 2 Gradient Descent with Line Search

Input: Given a initial starting point $\mathbf{x}^0 \in dom(f)$ and $t = 0$.
while stop condition is false **do**
    Using the exact line search or backtracking line search to find a proper s_t;
    $\mathbf{x}^{t+1} = \mathbf{x}^t - s_t \nabla f(\mathbf{x}^t)$ and $t := t + 1$.
end while
Output: $\tilde{\mathbf{x}}^T$ where T is the last step index.

Combining the descent direction and the step size, the gradient descent algorithm with line search can be formalized as in Algorithm 2.

To complete Algorithm 2, we need to concrete the stop condition and output.

How to stop?

- Give a $T_{\max}$.
- Give a tolerance ϵ and $\|\mathbf{x}^{t+1} - \mathbf{x}^t\| \leq \epsilon$.
- $|f(\mathbf{x}^{t+1}) - f(\mathbf{x}^t)| \leq \epsilon$.
- $\|\nabla f(\mathbf{x}^t)\| \leq \epsilon$.

What is the output?

- $\tilde{\mathbf{x}}^T = \mathbf{x}^T$.
- $\tilde{\mathbf{x}}^T = \frac{1}{T} \sum_{t=0}^{T} \mathbf{x}^t$.
- $\tilde{\mathbf{x}}^T = \frac{1}{T - T_0} \sum_{t=T_0}^{T} \mathbf{x}^t$.

References