IEOR 160: Nonlinear and Discrete Optimization
Optimization Algorithms

Yatong Bai

February 6, 2024

This is a summarization of Professor Javad Lavaei's IEOR 160 lecture notes.

1 Golden Section

Golden section is a zeroth-order algorithm for uni-variate optimization problems. Consider the problem $\max_{x\in\mathbb{R}} f(x)$.

- **Unimodal function:** a function $f : [a, b] \to \mathbb{R}$ is unimodal if there exists $x_\star$ such that $f(x)$ strictly increases on $[a, x_\star]$ and strictly decreases on $[x_\star, b]$.
- **Implementation of Golden Section:**

  Initialize $x^{(1)} \leftarrow a + (1-r)(b-a)$, $x^{(2)} \leftarrow a + r(b-a)$
  while $b - a > \epsilon$ **do**
    if $f(x^{(1)}) \leq f(x^{(2)})$ **then**
      $a \leftarrow x^{(1)}$, $x^{(1)} \leftarrow x^{(2)}$, $x^{(2)} \leftarrow a + r(b-a)$
    else
      $b \leftarrow x^{(2)}$, $x^{(2)} \leftarrow x^{(1)}$, $x^{(1)} \leftarrow a + (1-r)(b-a)$
    end if
  end while
  return $x_\star \in [a, b]$ (often $\frac{a+b}{2}$)

- **Computation:** There are two evaluations and one comparison in each iteration.
- **The value of the golden ratio** r**:** $\frac{1+\sqrt{5}}{2} \approx 0.618$. Other numbers between 0 and 1 can be used, but will make the section algorithm less efficient.
- **Required number of steps:** $k = \text{ceil}\left(\frac{\log\left(\epsilon/(b-a)\right)}{\log r}\right) = \text{ceil}\left(\frac{\log\epsilon - \log(b-a)}{\log r}\right)$. I.e., find the smallest integer k such that $r^k(b-a) \leq \epsilon$.

2 Gradient and Newton's Methods

The gradient method is a first-order method, whereas Newton's method is second-order. They apply to uni-variate and multi-variate optimization problems. Consider the problem $\min_{x\in\mathbb{R}^n} f(x)$.

- **Descent direction:** At a point $\bar{x} \in \mathbb{R}^n$, Δx is a descent direction if $\nabla f(\bar{x})^\top \Delta x < 0$.

- A family of iterative optimization algorithms can be designed based on descent directions: the $(k+1)^{\text{th}}$ iteration is $x^{(k+1)} \leftarrow x^{(k)} + t^{(k)} \Delta x^{(k)}$, where $\Delta x^{(k)}$ is a descent direction w.r.t. $x^{(k)}$, and $t^{(k)}$ is the step size at the $(k+1)^{\text{th}}$ iteration.

- **Gradient method:** $x^{(k+1)} \leftarrow x^{(k)} - t^{(k)} \nabla f(x^{(k)})$.
  Here, we use $-\nabla f(x^{(k)})$, which is a descent direction, as $\Delta x^{(k)}$.

- **Newton's method:** $x^{(k+1)} \leftarrow x^{(k)} - t^{(k)} \left(\nabla^2 f(x^{(k)})\right)^{-1} \nabla f(x^{(k)})$.
  Here, we use $-\left(\nabla^2 f(x^{(k)})\right)^{-1} \nabla f(x^{(k)})$, which is another descent direction, as $\Delta x^{(k)}$.

- **Why gradient/Newton?** The gradient direction minimizes a local first-order Taylor approximation of the objective function. Similarly, the Newton direction minimizes a second-order approximation, and therefore Newton's method can solve certain quadratic problems in one iteration with $t^{(k)} = 1$.

- Newton's method converges faster than the gradient method, but each iteration takes longer.

2.1 Determining the Step Size $t^{(k)}$

- We can use fixed step size (e.g. $t^{(k)} = t$ for all k). However, too small $t^{(k)}$ leads to slow convergence, whereas too large $t^{(k)}$ leads to divergence.

- **Exact line search:** At each iteration, solve the problem $t^{(k)} = \arg\min_{t>0} f\left(x^{(k)} + t\Delta x^{(k)}\right)$. Exact line search breaks one multi-variate problem into a series of uni-variate problems, which can then be solved using golden section.

- **Backtracking line search:** A practical approximation of exact line search:

  Initialize $\alpha > 0$, $0 < \beta < 1$, and $m = 0$
  while $f\left(x^{(k)} + \alpha\beta^m \Delta x^{(k)}\right) \geq f\left(x^{(k)}\right)$ **do**
  $\quad m \leftarrow m + 1$
  end while
  return $t^{(k)} = \alpha\beta^m$