Appendices

Basic Algorithms for Nonlinear Programming

.1 Gradient Algorithms

.1.1 Preliminaries: convergence analysis

Consider an iterative algorithm for solving the optimization problem $\min f(x)$, producing iterates $\{x^0, x^1, \dots\}$.

1. The possible error measurements are as follows. The stopping criteria depends on these error measurements.
   - $e(x^k) := \|x^k - x^*\|$;
   - $e(x^k) = f(x^k) - f(x^*)$;

   where x^* denotes the underlying optimal solution.

2. We say the algorithm converges if $\lim_{k\to\infty} e(x^k) = 0$

3. There are different types of convergence rate:
   (a) R-linear convergence: there exists $a \in (0,1)$ such that $e(x^k) \leq Ca^k$;
   (b) Q-linear convergence: there exists $a \in (0,1)$ such that $\frac{e(x^{k+1})}{e(x^k)} \leq a$;

(c) Sub-linear convergence: $e(x^k) \le C/k^p$ for some $p > 0$.

question: when say about convergence rate, do we need to specify which error measurements we use?

.1.2 The (Sub)gradient algorithm for Unconstrained Optimization

Consider an unconstrained optimization problem $\min f(x)$, where f may not necessarily be smooth. Let $\{t_k > 0 \mid k = 0, 1, \dots\}$ be a sequence of step-sizes. Let's study the simpleest first order optimization algorithm.

Algorithm 1 The (Sub)gradient Algorithm

Input: Initial guess $x^0 \in \mathcal{X}$
Output: Optimal solution $\hat{x}$
For $k = 0, 1, \dots,$ **do**

- Take $d^k \in \partial f(x^k)$;
- $x^{k+1} \leftarrow x^k - t_k d^k$

end for.

Worst Case Bounds Consier a *convex optimization model* where f is a completely unknown function. The first order type algorithm esentially produces a sequence of iterates $\{x^k \mid k = 0, 1, 2, \dots\}$ in such a way that x^k is in the *affine space* spanned by

$$x^0, g(x^0), \dots, g(x^{k-1}), \text{ where } g(\cdot) = \partial f(\cdot).$$

- Suppose f is *Lipschitz continuous* and no other information is known, we can construct an example such that

$$\min_{x \in \text{Span}\{x^0, g(x^0), \dots, g(x^{k-1})\}} f(x) - f(x^*) \ge \mathcal{O}(\frac{1}{\sqrt{k}}), \ \forall k = 1, 2, \dots, \lfloor \frac{n}{2} \rfloor$$

  Therefore, the first order type algorithm can never reach the convergence rate faster than $\mathcal{O}(\frac{1}{\sqrt{k}})$.

- Additionally, if we know f is *differentiable* and ∇f is *Lipschitz continuous*, then we can construct an example such that

$$\min_{x\in\mathrm{Span}\{x^0,g(x^0),\ldots,g(x^{k-1})\}} f(x)-f(x^*) \geq \mathcal{O}(\frac{1}{k^2}),\ \forall k=1,2,\ldots,\lfloor\frac{n}{2}\rfloor$$

  Therefore, the first order type algorithm can never reach the convergence rate faster than $\mathcal{O}(\frac{1}{k^2})$ for optimizing this class of function.

.1.3 Gradient Algorithm with Exact Line-Search

First we discuss the optimization with a uniform convex function. This assumption is by default unless specifically mentioned. A nice Q-linear convergence result is obtained:

Theorem .1. Suppose there exists $0 < m \leq M$ such that $0 \succ mI \succeq \nabla^2 f(x) \succeq MI$ (i.e., f is *uniformly convex*), and an exact line search is performed per iteration:

$$t_k := \arg\min_t f(x^k - t\nabla f(x^k)),$$

then

$$f(x^{k+1}) - f(x^*) \leq \left(1-\frac{m}{M}\right)[f(x^k)-f(x^*)] \tag{1}$$

Proof.

- **(Uniform Convexity implies Strongly Convexity)** For $\forall \boldsymbol{x}_1,\boldsymbol{x}_2 \in \mathrm{dom}(f)$, by mean-value theorem,

$$f(\boldsymbol{x}_2) = f(\boldsymbol{x}_1)+\langle\nabla f(\boldsymbol{x}_1),\boldsymbol{x}_2-\boldsymbol{x}_1\rangle+\frac{1}{2}(\boldsymbol{x}_2-\boldsymbol{x}_1)^{\mathrm{T}}\nabla^2 f(\boldsymbol{\xi})(\boldsymbol{x}_2-\boldsymbol{x}_1),$$

  where $\boldsymbol{\xi}$ is some number between $\boldsymbol{x}_2$ and $\boldsymbol{x}_1$. Applying the uniform convexity of f, we derive the strongly convexity property:

$$\frac{m}{2}\|\boldsymbol{x}_1-\boldsymbol{x}_2\|_2^2 \leq f(\boldsymbol{x}_2)-f(\boldsymbol{x}_1)-\langle\nabla f(\boldsymbol{x}_1),\boldsymbol{x}_2-\boldsymbol{x}_1\rangle \leq \frac{M}{2}\|\boldsymbol{x}_1-\boldsymbol{x}_2\|_2^2 \tag{2}$$

- **(Applying Strongly Convexity Property)** On the one hand, by setting $\boldsymbol{x}_1=\boldsymbol{x}^*$ and $\boldsymbol{x}_2=\boldsymbol{x}$ in (2), we obtain:

$$\frac{m}{2}\|\boldsymbol{x}-\boldsymbol{x}^*\|_2^2 \leq f(\boldsymbol{x})-f(\boldsymbol{x}^*) \leq \frac{M}{2}\|\boldsymbol{x}-\boldsymbol{x}^*\|_2^2 \tag{3}$$

On the other hand, by setting $\boldsymbol{x}_1 = \boldsymbol{x}$ and $\boldsymbol{x}_2 = \boldsymbol{x}^*$ in (2), we obtain

$$\begin{aligned}\frac{m}{2}\|\boldsymbol{x}-\boldsymbol{x}^*\|_2^2 &\leq f(\boldsymbol{x}^*) - f(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \boldsymbol{x}^* - \boldsymbol{x}\rangle \\ &\leq f(\boldsymbol{x}^*) - f(\boldsymbol{x}) + \|\nabla f(\boldsymbol{x})\| \cdot \|\boldsymbol{x}^* - \boldsymbol{x}\| \\ &\leq -\frac{m}{2}\|\boldsymbol{x}-\boldsymbol{x}^*\|_2^2 + \|\nabla f(\boldsymbol{x})\| \cdot \|\boldsymbol{x}^* - \boldsymbol{x}\|\end{aligned}$$

which implies $m\|\boldsymbol{x}-\boldsymbol{x}^*\| \leq \|\nabla f(\boldsymbol{x})\|$. Similarly, we get

$$m\|\boldsymbol{x}-\boldsymbol{x}^*\| \leq \|\nabla f(\boldsymbol{x})\| \leq M\|\boldsymbol{x}-\boldsymbol{x}^*\| \tag{4}$$

- **(Upper Bounding left and right side of (1))** Moreover, we upper bounding the left side of (1) by setting $\boldsymbol{x}_2 = \boldsymbol{x}^{k+1}$ and $\boldsymbol{x}_1 = \boldsymbol{x}^k$ in (2):

$$\begin{aligned}f(\boldsymbol{x}^{k+1}) - f(\boldsymbol{x}^k) &\leq \langle \nabla f(\boldsymbol{x}^k), \boldsymbol{x}^{k+1} - \boldsymbol{x}^k\rangle + \frac{M}{2}\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^k\|^2 \\ &\leq -\frac{1}{2M}\|\nabla f(x^k)\|^2\end{aligned} \tag{5}$$

where the second inequality is active when $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - \frac{1}{M}\nabla f(\boldsymbol{x}^k)$. On the other hand, by setting $\boldsymbol{x}_2 = \boldsymbol{x}^*$ and $\boldsymbol{x}_1 = \boldsymbol{x}^k$ in (2), we obtain

$$\begin{aligned}f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*) &\leq \langle \nabla f(x^k), \boldsymbol{x}^k - \boldsymbol{x}^*\rangle - \frac{m}{2}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|_2^2 \\ &\leq \|\nabla f(x^k)\|\|\boldsymbol{x}^k - \boldsymbol{x}^*\| - \frac{m}{2}\|\boldsymbol{x}^k-\boldsymbol{x}^*\|_2^2 \\ &\leq \frac{1}{2m}\|\nabla f(\boldsymbol{x}^k)\|^2\end{aligned} \tag{6}$$

Therefore, substituting (6) into (5), we obtain

$$f(\boldsymbol{x}^{k+1}) - f(\boldsymbol{x}^k) \leq -\frac{m}{M}[f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*)]$$

Or equivalently,

$$f(\boldsymbol{x}^{k+1}) - f(\boldsymbol{x}^*) \leq \left(1 - \frac{m}{M}\right)[f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*)]$$

□

question: this proof also holds for $t_k = \frac{1}{M}$. Thus what is the intuition behind the line search.

question: is uniformly convex and strongly convex talking about the same thing?

.1.4 Gradient Algorithm with Diminishing Step Sizes

Consider a pre-scribed diminishing step size $\{\alpha_k\} \to 0$ but satisfies the infinite travel condition $\sum_{k=1}^{\infty} \alpha_k = \infty$.

In this case, for sufficiently large k, we have $\alpha_k \leq \frac{1}{M}$ and simiar to the idea in (5),

$$f(\boldsymbol{x}^{k+1}) \leq f(\boldsymbol{x}^k) - \frac{\alpha_k}{2}\|\nabla f(\boldsymbol{x}^k)\|^2$$

which implies that $\nabla f(\boldsymbol{x}^k)$ cannot be bounded away from 0 whenever $f(\boldsymbol{x}^k)$ is finitely lower bounded. In other words, if a finite minimum exists for $f(\boldsymbol{x}^k)$, then the iterates satisfy $\lim_{k\to\infty} \inf \|\nabla f(\boldsymbol{x}^k)\| = 0$.

We can further show the whole sequence $f(\boldsymbol{x}^k)$ converges:

Proof. w.l.o.g., assume the inequality below holds for $k = 1, 2, \ldots$, i.e., $\alpha_k \leq \frac{1}{M}$:

$$f(\boldsymbol{x}^{k+1}) \leq f(\boldsymbol{x}^k) - \frac{\alpha_k}{2}\|\nabla f(\boldsymbol{x}^k)\|^2$$

Therefore, for any $k = 1, 2, \ldots,$

$$\begin{aligned} f(\boldsymbol{x}^{k+1}) - f(\boldsymbol{x}^*) &\leq f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*) - \frac{\alpha_k}{2}\|\nabla f(\boldsymbol{x}^k)\|^2 \\ &\leq (1 - m\alpha_k)[f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*)] \end{aligned}$$

where the second inequality is by applying (6). It follows that

$$f(\boldsymbol{x}^n) - f(\boldsymbol{x}^*) \leq [f(\boldsymbol{x}^1) - f(\boldsymbol{x}^*)] \prod_{k=1}^{n} (1 - m\alpha_k) \to 0,$$

i.e., $\lim_{n\to\infty} f(\boldsymbol{x}^n) = f(\boldsymbol{x}^*)$.

There is another way to show the convergence of $\{\nabla f(\boldsymbol{x}^k)\}$:

$$\|\nabla f(\boldsymbol{x}^n)\|^2 \leq 2M[f(\boldsymbol{x}^n) - f(\boldsymbol{x}^*)] \implies \lim_{n\to\infty} \nabla f(\boldsymbol{x}^k) = \boldsymbol{0}.$$

$\square$

We summarize the results above as a theorem for the convergence of the gradient algorithm with diminishing step sizes:

Theorem .2. Suppose there exists $0 < m \leq M$ such that $0 \succ mI \succeq \nabla^2 f(x) \succeq MI$ (i.e., f is *uniformly convex*), and the dimishing step size is performed per iteration:

$$\alpha_k \to 0, \text{ but } \sum_{k=1}^{\infty} \alpha_k = \infty,$$

then either $f(\boldsymbol{x}^k) \to -\infty$ or else $\{f(\boldsymbol{x}^k)\}$ converges to a finite value and $\nabla f(\boldsymbol{x}^k) \to \mathbf{0}$.

.1.5 Gradient Algorithm with Armijo's Rule

Consider a general iterative descent algorithm $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k$. The Armiijo's rule for choosing step sizes is as follows:

> Let $\gamma \in (0,1)$ (question: 1/2?). Start with $s > 0$ and continue with $\beta s, \beta^2 s, \dots,$ until $\beta^\ell s$ falls within the set of α with the condition
>
> $$f(\boldsymbol{x}^k) - f(\boldsymbol{x}^k + \alpha \boldsymbol{d}^k) \geq -\gamma\alpha \cdot \nabla^{\mathrm{T}} f(\boldsymbol{x}^k)\boldsymbol{d}^k$$

In this case we have $\alpha_k = s\beta^\ell$ and

$$f(\boldsymbol{x}^k) \geq f(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k) - \gamma\alpha_k \nabla^{\mathrm{T}} f(\boldsymbol{x}^k)\boldsymbol{d}^k \tag{7a}$$

$$f(\boldsymbol{x}^k) < f(\boldsymbol{x}^k + \alpha_k/\beta \boldsymbol{d}^k) - \gamma\alpha_k/\beta \cdot \nabla^{\mathrm{T}} f(\boldsymbol{x}^k)\boldsymbol{d}^k \tag{7b}$$

We can analysis the convergence result for gradient algorithm, i.e., $\boldsymbol{d}^k = -\nabla f(\boldsymbol{x}^k)$:

- From the (7b) and the Taylor expansion on $f(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k)$ we obtain:

  $$f(\boldsymbol{x}^k)+\gamma\alpha_k/\beta\cdot\nabla^{\mathrm{T}} f(\boldsymbol{x}^k)\boldsymbol{d}^k < f(\boldsymbol{x}^k)+\alpha_k/\beta\nabla^{\mathrm{T}} f(\boldsymbol{x}^k)\boldsymbol{d}^k+\frac{M}{2}(\alpha_k/\beta)^2\|\boldsymbol{d}^k\|^2$$

  Or equivalently, $\alpha_k > \frac{2\beta(1-\gamma)}{M}$

- Combining the (7a), (6) and the bound on α_k, we obtain

  $$f(\boldsymbol{x}^k + \alpha_k \boldsymbol{d}^k) \leq f(\boldsymbol{x}^k) - 4\beta\gamma(1-\gamma)\frac{m}{M}[f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*)]$$

Therefore, we get the Q-linear convergence for Armijo's rule:

$$f(\boldsymbol{x}^{k+1}) - f(\boldsymbol{x}^*) \leq \left(1 - 4\beta\gamma(1-\gamma)\frac{m}{M}\right)[f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*)]$$

.1.6 The Gradient Algorithm for non-strongly convex case

The estimations of convergence so far are based on the assumption that $m > 0$. Now we discuss the case where $m = 0$. The function is convex but not necessarily strongly convex.

Assume that the set of optimal solutions is a bounded set, and that there is a bounded *level set*. If still apply the exact line search, the iterates will be bounded. Note that the inequalities below still hold:

$$\begin{aligned} f(\boldsymbol{x} + \alpha\boldsymbol{d}) &\leq f(\boldsymbol{x}) - \frac{1}{2M}\|\nabla f(\boldsymbol{x})\|^2 \\ f(\boldsymbol{x}) - f(\boldsymbol{x}^*) &\leq \|\nabla f(\boldsymbol{x})\| \cdot \|\boldsymbol{x} - \boldsymbol{x}^*\| \end{aligned}$$

Assume that $\|\boldsymbol{x}^k - \boldsymbol{x}^*\| \leq C$, and let $e(\boldsymbol{x}^k) = f(\boldsymbol{x}^k) - f(\boldsymbol{x}^*)$, Using the inequalities above, it's easy to show that

$$e(\boldsymbol{x}^{k+1}) \leq e(\boldsymbol{x}^k) - c[e(\boldsymbol{x}^k)]^2, \text{ where } c = \frac{1}{2MC^2}.$$

which follows that

$$\begin{aligned} \frac{1}{e(\boldsymbol{x}^{k+1})} &\geq \frac{1}{e(\boldsymbol{x}^k)} + \frac{c}{1 - c \cdot e(\boldsymbol{x}^k)} \\ &\geq \frac{1}{e(\boldsymbol{x}^k)} + c \\ &\geq \cdots \\ &\geq \frac{1}{e(\boldsymbol{x}^1)} + k \cdot c \end{aligned}$$

Therefore, we obtain the *sublinear rate of convergence*:

$$e(\boldsymbol{x}^{k+1}) \leq \frac{e(\boldsymbol{x}^1)}{1 + k(c \cdot e(\boldsymbol{x}^1))}$$

.1.7 Linear Convergence without Second Order Differentiability

Acutally, the assumptions on the existence of $\nabla^2 f$ is unnecessaryin Theorem (.1). We can weaken the condition by the inequality below to obtain the same linear convergence result:

$$\sigma\|\boldsymbol{x}-\boldsymbol{y}\|^2 \leq \langle \nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y}), \boldsymbol{x}-\boldsymbol{y}\rangle \leq L\|\boldsymbol{x}-\boldsymbol{y}\|^2,\ \forall \boldsymbol{x}, \boldsymbol{y}, \tag{8}$$

where $0 < \sigma \leq L < \infty$.

Remark .4.
- The condition (8) can be implied by uniform convexity.
- The interpretation of (8) is that, restricting f to any line segment between $\boldsymbol{x}$ and $\boldsymbol{y}$, the function $h(t) := f(x + t(y - x))$ satisfies

  $$0 \leq \frac{h'(t) - h'(s)}{t - s} \leq L, \quad \forall 0 \leq s < t \leq 1,$$

  i.e., the slope of ∇f is bounded.
- The condition (8) implies the strong convexity, which can be shown by appying the directional derivative and (8):

  $$\frac{\sigma}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2 \leq f(\boldsymbol{y}) - f(\boldsymbol{x}) - \langle \nabla f(\boldsymbol{x}), \boldsymbol{y}-\boldsymbol{x}\rangle \leq \frac{L}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2$$

Therefore, we can use the same logic to show the following inequalities:

$$\begin{aligned} f(\boldsymbol{x} - \alpha\nabla f(\boldsymbol{x})) - f(\boldsymbol{x}) &\leq -\frac{1}{2L}\|\nabla f(\boldsymbol{x})\|^2 \\ \sigma\|\boldsymbol{x}-\boldsymbol{x}^*\|^2 &\leq \|\nabla f(\boldsymbol{x})\|\|\boldsymbol{x}-\boldsymbol{x}^*\| \\ f(\boldsymbol{x}^*) &\geq f(\boldsymbol{x}) - \frac{1}{2\sigma}\|\nabla f(\boldsymbol{x})\|^2 \end{aligned}$$

nad therefore,

$$f(\boldsymbol{x} - \alpha\nabla f(\boldsymbol{x})) - f(\boldsymbol{x}^*) \leq \left(1 - \frac{\sigma}{L}\right)[f(\boldsymbol{x}) - f(\boldsymbol{x}^*)].$$

.2 The Pure Newton's Method

Now we discuss a particularly important method in optimization: Newton's method.

Motivation This method is a *linearlization scheme* for solving a nonlinear equation.

- For scalar form of nonlinear equation $g(x) = 0$, we apply Taylor's expansion on the root $\hat{x}$:

  $$g(\hat{x}) = g(x) + g'(x)(\hat{x} - x) + o(|\hat{x} - x|)$$

  Ignoring the high order part we get an approximation, i.e., iterative formula

  $$\bar{x} = x - \frac{g(x)}{g'(x)}.$$

- Consider a n-dimensional equation $g_{1:n}(x_1, \dots, x_n) = 0$, we have a similar solution

  $$g(\hat{\boldsymbol{x}}) = g(\boldsymbol{x}) + J(g(\boldsymbol{x})) \cdot (\hat{\boldsymbol{x}} - \boldsymbol{x}) + o(\|\hat{\boldsymbol{x}} - \boldsymbol{x}\|)$$

  where $J(g(\boldsymbol{x}))$ denotes the Jacobian matrix of g:

  $$\mathbb{R}^{n \times n} \ni J(g(\boldsymbol{x})) := \left[\frac{\partial g_i(\boldsymbol{x})}{\partial x_j}\right]$$

Therefore, the unconstrained optimization problem suffices to solve a nonlinear equation $\nabla f(\boldsymbol{x}) = 0$, and the iterative formula is

$$\bar{\boldsymbol{x}} = \boldsymbol{x} - [\nabla^2 f(\boldsymbol{x})]^{-1} \nabla f(\boldsymbol{x}), \quad \text{(Newton's Method)}$$

Remark .5.

1. Newton's direction may not necessarily exist;
2. It is a descent direction for strongly convex functions;
3. However, the function may not necessarily decrease even for strongly convex function.
4. It minimizes a strongly convex *quadratic* function in just *one* step.
5. The pure form of Newton's method can be modified by taking another step length.

.2.1 Local Convergence Analysis

We analysis the convergence rate for Newton's method under the convexity and continuity conditions first:

> **Assumption**: The function f is *convex*, *twice continuously differentiable*, and that $\nabla^2 f(\boldsymbol{x}^*)$ is non-singular for local minimum $\boldsymbol{x}^*$.

A key inequality for the analysis is

$$\nabla f(\boldsymbol{y}) = \nabla f(\boldsymbol{x}) + \int_0^1 \nabla^2 f(\boldsymbol{x} + t(\boldsymbol{y} - \boldsymbol{x})) \cdot (\boldsymbol{y} - \boldsymbol{x})\,\mathrm{d}t$$

Suppose that $\boldsymbol{x}^k$ is close to $\boldsymbol{x}^*$ enough, then $\nabla^2 f(\boldsymbol{x}^k)$ is non-singular as well due to the continuity of determinant function. It follows that

$$\begin{aligned}
&\boldsymbol{x}^{k+1} - \boldsymbol{x}^* = \boldsymbol{x}^k - \boldsymbol{x}^* - [\nabla^2 f(\boldsymbol{x}^k)]^{-1}\nabla f(\boldsymbol{x}^k)\\
&= [\nabla^2 f(\boldsymbol{x}^k)]^{-1}[\nabla^2 f(\boldsymbol{x}^k)(\boldsymbol{x}^k - \boldsymbol{x}^*) - \nabla f(\boldsymbol{x}^k)]\\
&= [\nabla^2 f(\boldsymbol{x}^k)]^{-1}\left[\nabla^2 f(\boldsymbol{x}^k)(\boldsymbol{x}^k - \boldsymbol{x}^*) - \int_0^1 \nabla^2 f(\boldsymbol{x}^* + t(\boldsymbol{x}^k - \boldsymbol{x}^*))(\boldsymbol{x}^k - \boldsymbol{x}^*)\,\mathrm{d}t\right]\\
&= [\nabla^2 f(\boldsymbol{x}^k)]^{-1}\left\{\int_0^1 [\nabla^2 f(\boldsymbol{x}^k) - \nabla^2 f(\boldsymbol{x}^* + t(\boldsymbol{x}^k - \boldsymbol{x}^*))](\boldsymbol{x}^k - \boldsymbol{x}^*)\,\mathrm{d}t\right\}
\end{aligned}$$

Thererfore,

$$\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\| \le \|\boldsymbol{x}^k - \boldsymbol{x}^*\| \cdot \|[\nabla^2 f(\boldsymbol{x}^k)]^{-1}\| \cdot \int_0^1 \|\nabla^2 f(\boldsymbol{x}^k) - \nabla^2 f(\boldsymbol{x}^* + t(\boldsymbol{x}^k - \boldsymbol{x}^*))]\|\,\mathrm{d}t$$

Since $\boldsymbol{x}^k$ is close to $\boldsymbol{x}^*$, $\|[\nabla^2 f(\boldsymbol{x}^k)]^{-1}\|$ is bounded. Since $\nabla^2 f(\boldsymbol{x})$ is continuous, the integration term goes to zero as $\|\boldsymbol{x}^k - \boldsymbol{x}^*\| \to 0$. Thus we imply $\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\| = o(\|\boldsymbol{x}^k - \boldsymbol{x}^*\|)$, ensuring a superlinear convergence.

> **Extra Assumption**: The term $\nabla^2 f(\boldsymbol{x})$ is *Lipschitz continuous*: there exists $L_2 > 0$ such that
>
> $$\|\nabla^2 f(\boldsymbol{x}) - \nabla^2 f(\boldsymbol{y})\| \le L_2\|\boldsymbol{x} - \boldsymbol{y}\|, \quad \forall \boldsymbol{x}, \boldsymbol{y}.$$

This extra assumption will ensure a *quadratic convergence rate*:

$$\begin{aligned}
&\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|\\
&\le \|[\nabla^2 f(\boldsymbol{x}^k)]^{-1}\| \cdot \|\boldsymbol{x}^k - \boldsymbol{x}^*\| \cdot \int_0^1 \|\nabla^2 f(\boldsymbol{x}^k) - \nabla^2 f(\boldsymbol{x}^* + t(\boldsymbol{x}^k - \boldsymbol{x}^*))]\|\,\mathrm{d}t\\
&\le \frac{L_2}{2}\|[\nabla^2 f(\boldsymbol{x}^k)]^{-1}\| \cdot \|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2.
\end{aligned}$$

Further Assumption: Based on the previous two assumptions, we assume that f is *strongly convex*.

In this case, it is easy to show that

$$\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\| \leq \frac{L_2}{2m}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2.$$

This inequality introduces a *region of attraction*, i.e., as soon as $\boldsymbol{x}^k$ falls into the neighborhood of $\boldsymbol{x}^*$ with radius $2m/L_2$, the iterates will be trapped in the neighborhood and converge to $\boldsymbol{x}^*$ *quadratically*.

Remark .6. The pure form of Newton's method, however, has several drawbacks:

1. It in general does not guarantees global convergence if no additional assumption is given. Fortunartely, if f is strongly convex, then the Newton's method with *line-search* (e.g., with Armijo's step-length rule) will be globally convergent with a globally linear convergence rate.

2. If f is not *strictly* convex, then $\nabla^2 f$ may be singular. Even worse, if f is not convex, then the Newton's direction may not be a *descent direction*. In next section we will discuss how to handle such a situation.

.3 Practical Implementation of Newton's method

.3.1 Cholesky Factorization

First let's introduce a technique in optimization algorithms that can reduce computational complexity: the *Cholesky factorization*.

Consider the case where $\nabla^2 f(\boldsymbol{x}^k) \succ 0$, and the Newton's direction can be found by solving the linear system

$$\nabla^2 f(\boldsymbol{x}^k)\boldsymbol{d} = -\nabla f(\boldsymbol{x}^k).$$

Directly computing the inverse of $\nabla^2 f(\boldsymbol{x}^k)$ is computationally expansive, which motivates us to apply the *Cholesky factorization* as follows:

1. First apply the Cholesky factorization to get $\nabla^2 f(\boldsymbol{x}^k) = \boldsymbol{L}_k \boldsymbol{L}_k^{\mathrm{T}}$, where $\boldsymbol{L}_k$ is a lower triangular matrix, resulting in the following Newton's equation
$$\boldsymbol{L}_k \boldsymbol{L}_k^{\mathrm{T}} \boldsymbol{d} = -\nabla f(\boldsymbol{x}^k).$$

2. Firstly solve the lower triangular system below by forward substitution:
$$\boldsymbol{L}_k \boldsymbol{y} = -\nabla f(\boldsymbol{x}^k)$$
The complexity for this process is $\mathcal{O}(n^2)$.

3. Then solve the triangular system below by backforward substitution:
$$\boldsymbol{L}_k^{\mathrm{T}} \boldsymbol{d} = \boldsymbol{y}_k$$
Again, this step takes complexity $\mathcal{O}(n^2)$.

The basic Cholesky factorization algorithm is as follows:

Algorithm 2 Basic Cholesky factorization Algorithm

Input: A positive definite $n \times n$matrix $\boldsymbol{A}$
Output: Lower triangular matrix $\boldsymbol{L}$ such thtat $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^{\mathrm{T}}$
For $j = 1 : n$**, do**

- **For** $i = j + 1 : n$**, do**
  - $l_{ij} = \left(a_{ij} - \sum_{k=1}^{j-1} l_{jk} l_{ik}\right) / l_{jj}$

  end for.
  $l_{jj} = \left(a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2\right)^{1/2}.$

end for.

Remark .7. If $\boldsymbol{A}$ is not positive semidefinite, then at a certain stage we will encounter a j such that
$$a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2 < 0.$$
In that case, the Cholesky decomposition cannot proceed. Note that the Cholesky decomposition takes about $\mathcal{O}(n^3)$ operations.

.3.2 Modified Newton's method

In case the Hessian matrix is not positive definite, the following remedies can be applied:

> If there occurs $a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2 < 0$ for a certain j, then we simply increase a_{jj} so that the quantity becomes positive again. (question: increase how much?)

This remedy has the same effect of changing $\nabla^2 f(\boldsymbol{x}^k)$ into $\nabla^2 f(\boldsymbol{x}^k) + \Delta^k \succ 0$, where Δ^k is non-negative (question: positive or non-negative?) diagonal, which suffices to solve the regularized equation

$$(\nabla^2 f(\boldsymbol{x}^k) + \Delta^k)\boldsymbol{d} = -\nabla f(\boldsymbol{x}^k).$$

Moreover, we may use the direction with Armijo's line search technique to guarantee the global convergence. (how to show?)

.3.3 The Trust Region Approach

Another way to handle the case that $\nabla^2 f(\boldsymbol{x}^k)$ is indefinite is to use the *trust region approach*. It is the complement of the line search approach.

The direction $\boldsymbol{d}^k$ for each iteration suffices to consider the trust region subproblem

$$\begin{array}{ll} \min & \langle \nabla f(\boldsymbol{x}^k), \boldsymbol{d} \rangle + \frac{1}{2}\boldsymbol{d}^{\mathrm{T}} \nabla^2 f(\boldsymbol{x}^k)\boldsymbol{d} \\ \text{such that} & \|\boldsymbol{d}\| \leq \delta \end{array}$$

where $\delta > 0$ is called the trust region radius.

Remark .8. It can be shown that when δ is sufficiently small, $f(\boldsymbol{x}^k + \boldsymbol{d}^k) < f(\boldsymbol{x}^k)$, i.e., $\boldsymbol{d}^k$ is the descent direction. This trust region subproblem can be efficiently solved (question: which method? curious about it)

.3.4 Implementation of Least Squares Problem

Consider solving the *nonlinear least square problem* (NLSP)

$$\min \quad f(x) = \frac{1}{2}\sum_{i=1}^m f_i^2(x)$$

Firstly note that

$$\nabla f(x) = \sum_{i=1}^{m} f_i(x) \nabla f_i(x)$$
$$\nabla^2 f(x) = \sum_{i=1}^{m} [\nabla f_i(x) \nabla^{\mathrm{T}} f_i(x) + f_i(x) \nabla^2 f_i(x)]$$

The so-called Gauss-Newton method is a *quasi-Newton's method*, specialized to this NLSP:

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - \alpha_k \left(\sum_{i=1}^{m} \nabla f_i(\boldsymbol{x}^k) \nabla^{\mathrm{T}} f_i(\boldsymbol{x}^k) \right)^{-1} \left(\sum_{i=1}^{m} f_i(\boldsymbol{x}^k) \nabla f_i(\boldsymbol{x}^k) \right)$$

Remark .9. It works well when f_i's are not *too linear*, or when at the optimality, f_i's are close to zero.

A variantion of the Gauss-Newton's method operates as follows:

$$\boldsymbol{x}^{k+1} = \boldsymbol{x}^k - \alpha_k \left(\sum_{i=1}^{m} \nabla f_i(\boldsymbol{x}^k) \nabla^{\mathrm{T}} f_i(\boldsymbol{x}^k) + \lambda_k \boldsymbol{I} \right)^{-1} \left(\sum_{i=1}^{m} f_i(\boldsymbol{x}^k) \nabla f_i(\boldsymbol{x}^k) \right)$$

which is called the *Levenberg-Marquardt method.*

Note that if consider solving the equation $f_{1:n}(\boldsymbol{x}_{1:n}) = \mathbf{0}$, the Gauss-Newton direction is just the Newton direction itself.