Mathematics of Data: From Theory to Computation

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-556 (Fall 2014)

License Information for Mathematics of Data Slides

Outline

- This lecture
  1. Unconstrained convex optimization: the basics
  2. Optimization methods
  3. What about non-smooth optimization?
- Next lecture
  1. Composite convex minimization for nonsmooth functions
  2. Proximal-gradient and proximal-Newton methods

Recommended reading

- Chapters 2, 3, 5, 6 in Nocedal, Jorge, and Wright, Stephen J., *Numerical Optimization*, Springer, 2006.
- Chapter 9 in Boyd, Stephen, and Vandenberghe, Lieven, *Convex optimization*, Cambridge university press, 2009.
- Chapter 1 in Bertsekas, Dimitris, *Nonlinear Programming*, Athena Scientific, 1999.
- Chapters 1, 2 and 4 in Nesterov, Yurii, *Introductory Lectures on Convex Optimization: A Basic Course*, Vol. 87, Springer, 2004.

Motivation

Motivation

This lecture covers the basics of numerical methods for *unconstrained* and *smooth* convex minimization.

Smooth unconstrained convex minimization

Problem (**Mathematical formulation**)

How can we find an approximately optimal solution to the following optimization problem?

$$\boxed{F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ F(\mathbf{x}) := f(\mathbf{x}) \right\}} \tag{1}$$

where f is proper, closed, convex and twice differentiable, $f \in \mathcal{F}^2$.
Note that (1) is unconstrained.

Three remarks

1. (1) covers a wide range of problems, including but not limited to
   - ML estimators, such as least squares and Poisson,
   - M-estimators,
   - Empirical risk minimization.
2. This lecture covers algorithms that have computational efficiency guarantees for (1) using additional structural assumptions on the convex functions.
3. Since F is composed of only one term f, throughout this lecture we only refer to f and $f^\star = F^\star$. We consider more general cases later.

Definition (Optimal solutions and solution set)

- (1) has solution if $f^\star$ is finite.
- $\mathbf{x}^\star \in \mathbb{R}^p$ is a solution to (1) if $\boxed{f(\mathbf{x}^\star) = f^\star}$.
- $\boxed{\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathbb{R}^p \;:\; f(\mathbf{x}^\star) = f^\star\}}$ is the solution set of (1).

Definition (Optimal solutions and solution set)

- (1) has solution if $f^\star$ is finite.
- $\mathbf{x}^\star \in \mathbb{R}^p$ is a solution to (1) if $\boxed{f(\mathbf{x}^\star) = f^\star}$.
- $\boxed{\mathcal{S}^\star := \{\mathbf{x}^\star \in \mathbb{R}^p \ : \ f(\mathbf{x}^\star) = f^\star\}}$ is the solution set of (1).

Assumption (Three key structures for (1))

Throughout, we assume f to feature one of the following structures

(a) *f is Lipschitz-gradient, i.e., $f \in \mathcal{F}_L^{2,1}(\mathbb{R}^p)$.*

(b) *f is Lipschitz-gradient* **and** *strongly convex, i.e., $f \in \mathcal{F}_{L,\mu}^{2,1}(\mathbb{R}^p)$.*

(c) *f is self-concordant, i.e., $f \in \mathcal{F}_2(\mathcal{Q})$.*

Example 1: ML estimation and M-estimators

Problem

Let $\mathbf{x}^\natural \in \mathbb{R}^p$ be an unknown vector. Let b_i be a sample of a random variable B_i with unknown probability density function $p_i(b_i; \mathbf{x}^\natural)$ in the set $\mathcal{P}_i := \{p_i(b_i; \mathbf{x}) : \mathbf{x} \in \mathbb{R}^p\}$. How do we estimate $\mathbf{x}^\natural$ given $\mathcal{P}_1, \ldots, \mathcal{P}_n$ and $b_1, \ldots, b_n$?

Optimization formulation (ML estimator)

$$\boxed{\min_{\mathbf{x}\in\mathcal{X}} \underbrace{-\frac{1}{n}\sum_{i=1}^{n} \ln\left[p_i(b_i; \mathbf{x})\right]}_{f(\mathbf{x})}.}$$

Optimization formulation (M-estimator)

In general, we can replace the negative log-likelihoods by any appropriate functions f_i

$$\boxed{\min_{x\in\mathcal{X}} \underbrace{\frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}; b_i)}_{f(\mathbf{x})}.}$$

Example 2: Least-squares estimation

Problem

Let $\mathbf{x}^\natural \in \mathbb{R}^p$. Let $\mathbf{A} \in \mathbb{R}^{n\times p}$ with full column rank. How do we estimate $\mathbf{x}^\natural$ given $\mathbf{A}$ and

$$\mathbf{b} = \mathbf{A}\mathbf{x}^\natural + \mathbf{w},$$

where $\mathbf{w}$ denotes some unknown noise (either random or deterministic)?

Optimization formulation (Least-squares estimator)

$$\boxed{\min_{\mathbf{x}\in\mathbb{R}^p} \underbrace{\frac{1}{2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2}_{f(\mathbf{x})}\ .}$$

Structural properties

- $\nabla f(\mathbf{x}) = \mathbf{A}^T(\mathbf{A}\mathbf{x} - \mathbf{b})$, and $\nabla^2 f(\mathbf{x}) = \mathbf{A}^T\mathbf{A}$.
- $\lambda_1\mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq \lambda_p\mathbf{I}$, where $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_p$ are the eigenvalues of $\mathbf{A}^T\mathbf{A}$.
- It follows that $\boxed{L = \lambda_p}$ and $\boxed{\mu = \lambda_1}$. If $\lambda_1 > 0$, then $f \in \mathcal{F}^{2,1}_{L,\mu}$, otherwise $f \in \mathcal{F}^{2,1}_{L}$. Also, if $n < p$, $\mathrm{rank}(\mathbf{A}^T\mathbf{A}) \leq n$, hence $\lambda_1 = 0$ and $f \in \mathcal{F}^{2,1}_{L}$.

Example 3: Logistic regression

Problem (**Logistic regression**)

Given a sample vector $\mathbf{a}_i \in \mathbb{R}^p$ and a binary class label $b_i \in \{-1,+1\}$ $(i = 1,\ldots,n)$, we define the conditional probability of b_i given $\mathbf{a}_i$ as:

$$\mathbb{P}(b_i|\mathbf{a}_i, \mathbf{x}^\natural, \mu) \propto 1/(1 + e^{-b_i(\langle \mathbf{x}^\natural, \mathbf{a}_i\rangle + \mu)}),$$

where $\mathbf{x}^\natural \in \mathbb{R}^p$ is some true weight vector, μ is called the intercept. How do we estimate $\mathbf{x}^\natural$ given the sample vectors, the binary labels, and μ?

Optimization formulation

$$\boxed{\min_{\mathbf{x}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n} \log(1 + \exp(-b_i(\mathbf{a}_i^T\mathbf{x} + \mu)))}_{f(\mathbf{x})}} \qquad (2)$$

Structural properties

- Let $\mathbf{A} = [\mathbf{a}_1,\ldots,\mathbf{a}_n]^T$, then $f \in \mathcal{F}_L^{2,1}$ and $\boxed{L = 0.25\lambda_p(\mathbf{A}^T\mathbf{A})}$.
- Whether we have $\mu > 0$ depends on the relative sizes of n and p. When $n > p$, the value of μ can also depend on $\mathbf{x}^\star$.

Example 4: Poisson imaging

Problem

Let $\mathbf{x}^\natural \in \mathbb{R}^p$ be an unknown vector. Let $b_1, \ldots, b_n$ be samples of independent random variables $B_1, \ldots, B_n$, and each B_i is Poisson distributed with parameter $\left\langle \mathbf{a}_i, \mathbf{x}^\natural \right\rangle$, where the vectors $\mathbf{a}_1, \ldots, \mathbf{a}_i$ are given. How do we estimate $\mathbf{x}^\natural$ given $\mathbf{a}_1, \ldots, \mathbf{a}_n$ and the measurements $b_1, \ldots, b_n$?

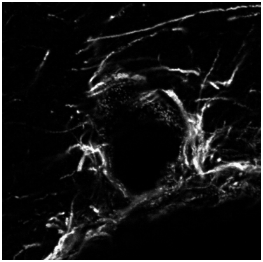
Optimization formulation

$$\hat{\mathbf{x}}_{\mathsf{ML}} \in \arg\min_{\mathbf{x}\in\mathbb{R}^p} \underbrace{\frac{1}{n}\sum_{i=1}^{n}\left[\langle \mathbf{a}_i, \mathbf{x}\rangle - b_i \ln\left(\langle \mathbf{a}_i, \mathbf{x}\rangle\right)\right]}_{f(\mathbf{x})}.$$

Structural properties

- $f \in \mathcal{F}_2$ is self-concordant with the domain $\mathcal{Q} = \{\mathbf{x} : \langle \mathbf{a}_i, \mathbf{x}\rangle \geq 0, i = 1, \ldots, n\}$ and the self-concordancy parameter

$$\boxed{M = 2\max\left\{\frac{1}{\sqrt{b_i}} \ : \ b_i > 0, \ i = 1, \ldots, n\right\}.}$$



Confocal imaging

Example 5: Graphical model learning

Problem (Graphical model selection)

Let $\mathbf{x}$ *be a random vector with zero mean and positive-definite covariance matrix* $\mathbf{\Sigma}^\natural$*. How do we estimate* $\mathbf{\Theta}^\natural := (\mathbf{\Sigma}^\natural)^{-1}$ *given independent samples* $\mathbf{x}_1, \ldots, \mathbf{x}_n$ *of the random vector* $\mathbf{x}$*?*

Optimization formulation

$$\boxed{\min_{\mathbf{\Theta} \in \mathbb{S}^p_{++}} \underbrace{\mathrm{Tr}\left(\widehat{\mathbf{\Sigma}}\mathbf{\Theta}\right) - \log\det\left(\mathbf{\Theta}\right)}_{f(\mathbf{\Theta})},}$$

where $\widehat{\mathbf{\Sigma}}$ is the empirical covariance, i.e., $\widehat{\mathbf{\Sigma}} := (1/n)\sum_{i=1}^n \mathbf{x}\mathbf{x}^T$.

Structural properties

- $f \in \mathcal{F}_2$ is standard-self concordant $\boxed{M = 2}$ with the domain $\mathcal{Q} = \mathbb{S}^p_{++}$.
- if $\alpha \mathbf{I} \preceq \Theta \preceq \beta \mathbf{I}$, then $f \in \mathcal{F}^{2,1}_{L,\mu}$ with $\boxed{L = \frac{\sqrt{p}}{\alpha^2}}$ and $\boxed{\mu = \frac{1}{\beta^2\sqrt{p}}}$.

Question: How do we design algorithms for finding a solution $\mathbf{x}^\star$?

Philosophy

- We cannot immediately design algorithms just based on the original formulation

$$F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \left\{ F(\mathbf{x}) := f(\mathbf{x}) \right\}. \tag{1}$$

- We need intermediate tools to characterize the optimal solution set $\mathcal{S}^\star$ of (1).
- One key tool is called the **optimality condition**

Optimality conditions and convexity

Lemma

Let f be a **smooth convex** *function, i.e., $f \in \mathcal{F}^1$. Then, any stationary point of f is also a global minimum.*

Proof.

Let $\mathbf{x}^\star$ be a stationary point, i.e., $\nabla f(\mathbf{x}^\star) = 0$. By convexity, we have:

$$f(\mathbf{x}) \geq f(\mathbf{x}^\star) + \langle \nabla f(\mathbf{x}^\star),\ \mathbf{x} - \mathbf{x}^\star \rangle \overset{\nabla f(\mathbf{x}^\star)=0}{=} f(\mathbf{x}^\star) \quad \text{for all } \mathbf{x} \in \mathbb{R}^p.$$

Moreover, in the special case where f is **strongly convex** ($f \in \mathcal{F}^1_\mu$), then:

$$\begin{aligned} f(\mathbf{x}) &\geq f(\mathbf{x}^\star) + \langle \nabla f(\mathbf{x}^\star),\ \mathbf{x} - \mathbf{x}^\star \rangle + \frac{\mu}{2}\|\mathbf{x} - \mathbf{x}^\star\|_2^2 \\ &\overset{\nabla f(\mathbf{x}^\star)=0}{=} f(\mathbf{x}^\star) + \frac{\mu}{2}\|\mathbf{x} - \mathbf{x}^\star\|_2^2 \end{aligned}$$

□

Remark

This result also holds if $f \in \mathcal{F}$ is not differentiable, which includes constrained problems. Simply stated, the condition $f(\mathbf{x}) \geq f(\mathbf{x}^\star) + \langle \mathbf{v},\ \mathbf{x} - \mathbf{x}^\star \rangle$ must hold for any $\mathbf{v} \in \partial f(\mathbf{x}^\star)$. In this case, the stationary point has $0 \in \partial f(\mathbf{x}^\star)$, hence $f(\mathbf{x}) \geq f(\mathbf{x}^\star)$.

Approximate vs. exact optimality

Is it possible to solve a convex optimization problem?

"In general, optimization problems are **unsolvable***"* - Y. Nesterov [4]

- Even when a closed-form solution exists, numerical accuracy may still be an issue.
- We must be content with **approximately** optimal solutions.

Definition

We say that $\mathbf{x}^\star_\epsilon$ is ϵ-optimal in **objective value** if

$$f(\mathbf{x}^\star_\epsilon) - f^\star \leq \epsilon \ .$$

We say that $\mathbf{x}^\star_\epsilon$ is ϵ-optimal in **sequence** if

$$\|\mathbf{x}^\star_\epsilon - \mathbf{x}^\star\| \leq \epsilon \ ,$$

for some norm $\|\cdot\|$.

- The latter approximation guarantee is considered stronger.

A gradient method

Lemma (First-order necessary optimality condition)

Let $\mathbf{x}^\star$ be a global minimum of a convex function $f \in \mathcal{F}^2$. Then, it holds that

$$\nabla f(\mathbf{x}^\star) = \mathbf{0}.$$

Fixed-point characterization

We can rewrite the first-order condition as

$$\mathbf{x}^\star = \mathbf{x}^\star - \alpha \nabla f(\mathbf{x}^\star) \qquad \text{for all } \alpha \in \mathbb{R}$$

Gradient method

Choose a starting point $\mathbf{x}^0$ and iterate

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k)$$

where α_k is a step-size to be chosen so that $\mathbf{x}^k$ converges to $\mathbf{x}^\star$ (ideally, as fast as possible).

Does the gradient method converge?

Lemma

Assume that

1. *There exists* $\mathbf{x}^\star \in$ *dom*(f) *such that* $\nabla f(\mathbf{x}^\star) = 0$.
2. *The mapping* $\psi(\mathbf{x}) = \mathbf{x} - \alpha \nabla f(\mathbf{x})$ *is contractive for some* α*: i.e., there exists* $\gamma \in [0, 1)$ *such that*

$$\|\psi(\mathbf{x}) - \psi(\mathbf{z})\| \leq \gamma \|\mathbf{x} - \mathbf{z}\| \quad \textit{for all } \mathbf{x}, \mathbf{z} \in \textit{dom}(f)$$

Then, for any starting point $\mathbf{x}^0 \in$ *dom*(f)*, the gradient method converges to* $\mathbf{x}^\star$.

Proof.

If we start the gradient method at $\mathbf{x}^0 \in \mathrm{dom}(f)$, then we have

$$\begin{aligned}
\|\mathbf{x}^{k+1} - \mathbf{x}^\star\| &= \|\mathbf{x}^k - \alpha \nabla f(\mathbf{x}^k) - \mathbf{x}^\star\| \\
&= \|\psi(\mathbf{x}^k) - \psi(\mathbf{x}^\star)\| && (\nabla f(\mathbf{x}^\star) = 0) \\
&\leq \gamma \|\mathbf{x}^k - \mathbf{x}^\star\| && \text{(contraction assumption)} \\
&\leq \gamma^{k+1} \|\mathbf{x}^0 - \mathbf{x}^\star\| \,.
\end{aligned}$$

We then have that the sequence $\{\mathbf{x}^k\}$ converges globally to $\mathbf{x}^\star$ at a **linear** rate. □

Short detour: Convergence rates

Definition (Convergence of a sequence)

We say that a sequence $\{\mathbf{u}^k\}$ (scalar or vector valued) converges to $\mathbf{u}^\star$ and write $\lim_{k\to\infty}\mathbf{u}^k = \mathbf{u}^\star$, if for any $\varepsilon > 0$, there is an integer K such that

$$\|\mathbf{u}^k - \mathbf{u}^\star\| \leq \varepsilon, \quad \text{for all } k \geq K.$$

Convergence rates: the **speed** at which a sequence converges

- **sublinear:** if there exists $c > 0$ such that

$$\|\mathbf{u}^k - \mathbf{u}^\star\| = O(k^{-c})$$

- **linear:** if there exists $\alpha \in (0,1)$ such that

$$\|\mathbf{u}^k - \mathbf{u}^\star\| = O(\alpha^k)$$

- **Q-linear:** if there exists a constant $r \in (0,1)$ such that

$$\lim_{k\to\infty} \frac{\|\mathbf{u}^{k+1} - \mathbf{u}^\star\|}{\|\mathbf{u}^k - \mathbf{u}^\star\|} = r$$

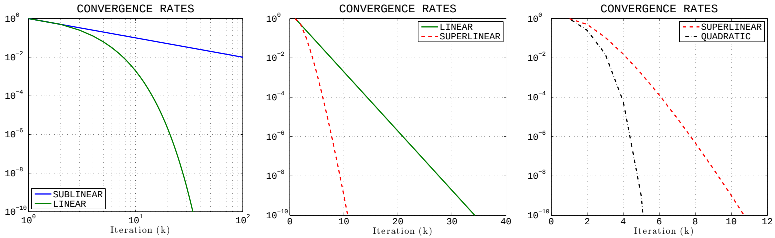
- **quadratic:** if there exists a constant $\mu > 0$ such that

$$\lim_{k\to\infty} \frac{\|\mathbf{u}^{k+1} - \mathbf{u}^\star\|}{\|\mathbf{u}^k - \mathbf{u}^\star\|^2} = \mu$$

Example: Convergence rates

Examples of sequences that all converge to $u^\star = 0$:

- Sublinear: $u^k = 1/k$
- Linear: $u^k = 0.5^k$
- Superlinear: $u^k = k^{-k}$
- Quadratic: $u^k = 0.5^{2^k}$



Remark

For **unconstrained** convex minimization as in (1), we always have $f(\mathbf{x}^k) - f^\star \geq 0$. Hence, we do not need to use the absolute value when we show convergence results based on the objective value, such as $f(\mathbf{x}^k) - f^\star \leq O(1/k^2)$, which is sublinear.

Detour: Global and local convergence

Global vs local convergence

An algorithm may show more than one type of convergence rate during execution:

1. **Global convergence rate:** overall convergence rate that applies for any starting point $\mathbf{x}^0$ (i.e., worst case scenario);
2. **Local convergence rate:** convergence rate that applies when the iterates have reached a certain region surrounding $\mathbf{x}^\star$.

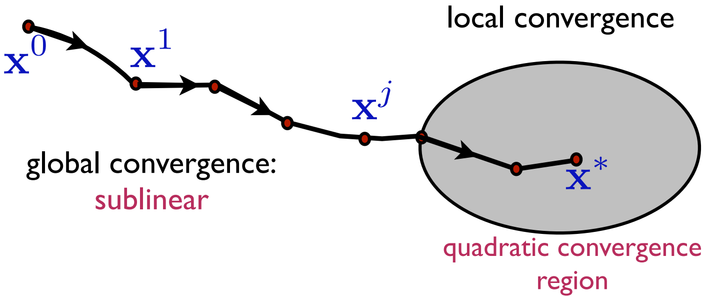
Definition (Local convergence)

We say that the algorithm **locally** converges to $\mathbf{x}^\star$, if there exists a constant $r > 0$, such that when the starting point $\mathbf{x}^0$ is chosen to satisfy $\|\mathbf{x}^0 - \mathbf{x}^\star\| \leq r$, the iterates $\mathbf{x}^k$ converge to $\mathbf{x}^\star$.

Global and local convergence - example

Newton method for self-concordant functions (later in this lecture)

- **Global convergence**: sublinear;
- **Local convergence**: quadratic;
- We can explicitly calculate size of the quadratic convergence region.



Contractive maps and convexity

Proposition (Contractivity implies convexity with structure)

Let $f \in \mathcal{C}^2$ and define $\psi(\mathbf{x}) = \mathbf{x} - \alpha\nabla f(\mathbf{x})$, with $\alpha > 0$.
If $\psi(\mathbf{x})$ is contractive, with a constant contraction factor $\gamma < 1$, then $f \in \mathcal{F}^{2,1}_{L,\mu}$.

Proof.

Consider $\mathbf{y} = \mathbf{x} + t\Delta\mathbf{x}$ and $\mathbf{z} = \mathbf{x}$. By the contractivity assumption it must hold that

$$\|\psi(\mathbf{x} + t\Delta\mathbf{x}) - \psi(\mathbf{x})\| \leq t\gamma\|\Delta\mathbf{x}\| \quad \forall t\ .$$

We also have that

$$\begin{aligned}
\lim_{t\to 0}\frac{1}{t}\|\psi(\mathbf{x} + t\Delta\mathbf{x}) - \psi(\mathbf{x})\| &= \lim_{t\to 0}\left\|\Delta\mathbf{x} - \frac{\alpha}{t}\left(\nabla f(\mathbf{x} + t\Delta\mathbf{x}) - \nabla f(\mathbf{x})\right)\right\| \\
&= \left\|\left(\mathbf{I} - \alpha\nabla^2 f(\mathbf{x})\right)\Delta\mathbf{x}\right\| \\
&\leq \gamma\|\Delta\mathbf{x}\| \qquad \text{(by assumption)}
\end{aligned}$$

The inequality implies (derivation on the board) that

$$\mathbf{0} \prec \frac{1-\gamma}{\alpha}\mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq \frac{1+\gamma}{\alpha}\mathbf{I},$$

which can be reinterpreted as $f \in \mathcal{F}^{2,1}_{L,\mu}$ with $L = \frac{1+\gamma}{\alpha}$ and $\mu = \frac{1-\gamma}{\alpha}$. □

Recall: convex, unconstrained, smooth minimization

Problem (**Mathematical formulation**)

$$\boxed{F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} \{F(\mathbf{x}) := f(\mathbf{x})\}} \tag{1}$$

where f is proper, closed, convex and twice differentiable.
Note that (1) *is unconstrained.*

How de we design efficient optimization algorithms with accuracy-computation tradeoffs for this class of functions?

Basic principles of descent methods

Iterative descent

1. Let $\mathbf{x}^0 \in \mathrm{dom}(f)$ be a starting point.
2. Generate sequence of vectors $\mathbf{x}^1, \mathbf{x}^2, \dots \in \mathrm{dom}(f)$ so that we have descent:

$$f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k), \quad \text{for all } k = 0, 1, \dots$$

   until $\mathbf{x}_k$ is ϵ-optimal.

Such a sequence $\left\{\mathbf{x}^k\right\}_{k\geq 0}$ can be generated as:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k$$

where $\mathbf{p}^k$ is a descent direction and $\alpha_k > 0$ a step-size.

Remark

- Iterative algorithms can implicitly use various **oracle** information from the objective, such as its value, gradient, or Hessian, in different ways to obtain α_k and $\mathbf{p}^k$, which determines their overall convergence rate and complexity. The type of oracle information they use becomes their defining characteristic.

Basic principles of descent methods

A condition for local descent directions

The iterates are given as:

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k$$

By Taylor's theorem, we have

$$f(\mathbf{x}^{k+1}) = f(\mathbf{x}^k) + \alpha_k \langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle + o(\alpha_k^2).$$

For α_k small enough, the term $\alpha_k \langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle$ dominates $o(\alpha_k^2)$ for a fixed $\mathbf{p}^k$. Therefore, in order to have $f(\mathbf{x}^{k+1}) < f(\mathbf{x}^k)$, we require:

$$\langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle < 0$$

Basic principles of descent methods

Local steepest descent direction

Since

$$\langle \nabla f(\mathbf{x}^k),\ \mathbf{p}^k \rangle = \|\nabla f(\mathbf{x}^k)\| \|\mathbf{p}^k\| \cos\theta\ ,$$

where θ is the angle between $\nabla f(\mathbf{x}^k)$ and $\mathbf{p}^k$, we have that

$$\mathbf{p}^k := -\nabla f(\mathbf{x}^k)$$

is the local *steepest descent* direction.

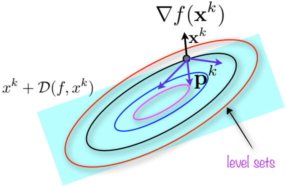


Figure: Descent directions in 2D should be an element of the cone of descent directions $\mathcal{D}(f, \cdot)$.

Gradient descent methods

Gradient descent (GD) algorithm

The gradient method we discussed before indeed use the local steepest direction:

$$\mathbf{p}^k = -\nabla f(\mathbf{x}^k)$$

so that

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k).$$

Key question: How do we choose α_k so that we have descent/contraction?

Gradient descent methods

Gradient descent (GD) algorithm

The gradient method we discussed before indeed use the local steepest direction:

$$\mathbf{p}^k = -\nabla f(\mathbf{x}^k)$$

so that

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k).$$

Key question: How do we choose α_k so that we have descent/contraction?

Answer: By exploiting the structures within the convex function

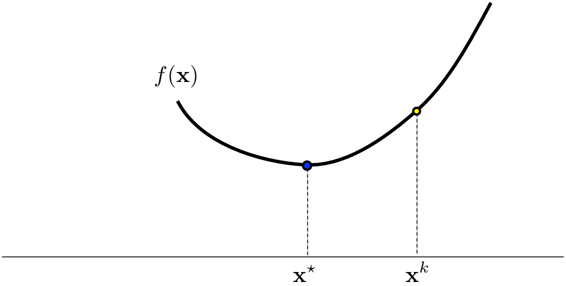
When $f \in \mathcal{F}_L^{2,1}$, we can use $\alpha_k = 1/L$ so that $\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k)$ is contractive.

- Note that the above GD method only uses the gradient information, and hence, it is called a **first-order method**.

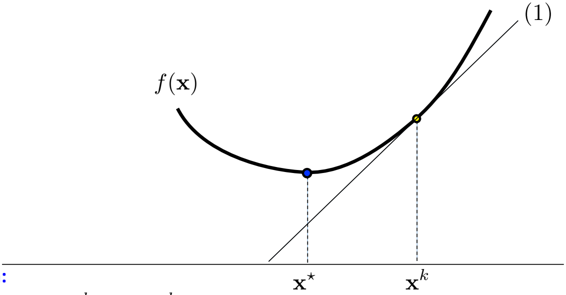
First-order methods employ only first-order oracle information about the objective, namely the value of f and ∇f at specific points.

- **Second-order methods** also use the Hessian $\nabla^2 f$.

Gradient descent methods - a geometrical intuition



Gradient descent methods - a geometrical intuition



Structure in optimization:

$(1) \qquad f(\mathbf{x}) \geq f(\mathbf{x}^k) - \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$

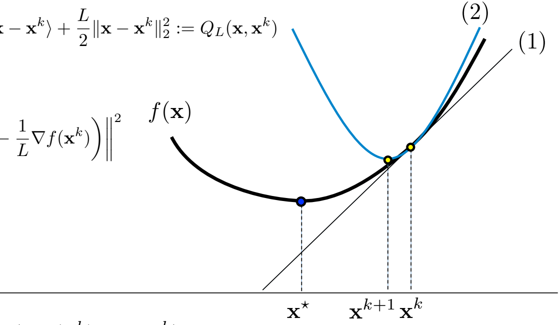
Gradient descent methods - a geometrical intuition

Majorize:

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q_L(\mathbf{x}, \mathbf{x}^k)$$

Minimize:

$$\begin{aligned} \mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_L(\mathbf{x}, \mathbf{x}^k) \\ &= \arg\min_{\mathbf{x}} \left\| \mathbf{x} - \left(\mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k) \right) \right\|^2 \\ &= \mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k) \end{aligned}$$



Structure in optimization:

(1) $\quad f(\mathbf{x}) \geq f(\mathbf{x}^k) - \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$

(2) $\quad f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

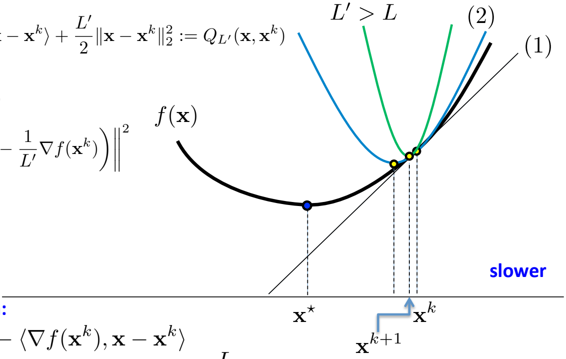
Gradient descent methods - a geometrical intuition

Majorize:

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L'}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q_{L'}(\mathbf{x}, \mathbf{x}^k)$$

Minimize:

$$\begin{aligned}\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_{L'}(\mathbf{x}, \mathbf{x}^k) \\ &= \arg\min_{\mathbf{x}} \left\| \mathbf{x} - \left(\mathbf{x}^k - \frac{1}{L'}\nabla f(\mathbf{x}^k) \right) \right\|^2 \\ &= \mathbf{x}^k - \frac{1}{L'}\nabla f(\mathbf{x}^k)\end{aligned}$$



Structure in optimization:

(1) $\quad f(\mathbf{x}) \geq f(\mathbf{x}^k) - \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$

(2) $\quad f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \dfrac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

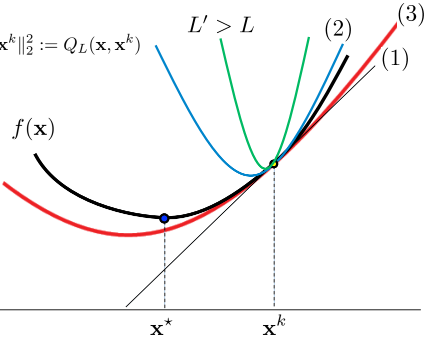
Gradient descent methods - a geometrical intuition

Majorize:

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q_L(\mathbf{x}, \mathbf{x}^k)$$

Minimize:

$$\begin{aligned} \mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_L(\mathbf{x}, \mathbf{x}^k) \\ &= \arg\min_{\mathbf{x}} \left\| \mathbf{x} - \left(\mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k) \right) \right\|^2 \\ &= \mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k) \end{aligned}$$



Structure in optimization:

(1) $\quad f(\mathbf{x}) \geq f(\mathbf{x}^k) - \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle$

(2) $\quad f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

(3) $\quad f(\mathbf{x}) \geq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{\mu}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

Convergence rate of gradient descent

Theorem

Let the starting point for GD be $\mathbf{x}^0 \in$ dom(f).

- *If $f \in \mathcal{F}_L^{2,1}$, with the choice $\alpha = \frac{1}{L}$, the iterates of GD satisfy*

$$\boxed{f(\mathbf{x}^k) - f(\mathbf{x}^\star) \leq \frac{2L}{k+4}\|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2}$$

- *If $f \in \mathcal{F}_{L,\mu}^{2,1}$, with the choice $\alpha = \frac{2}{L+\mu}$, the iterates of GD satisfy*

$$\boxed{\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2}$$

- *If $f \in \mathcal{F}_{L,\mu}^{2,1}$, with the choice $\alpha = \frac{1}{L}$, the iterates of GD satisfy*

$$\boxed{\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^{\frac{k}{2}} \|\mathbf{x}^0 - \mathbf{x}^\star\|_2}$$

Proof of convergence rates of gradient descent

- We first need to prove a basic result about functions in $\mathcal{F}_L^{1,1}$

Lemma

Let $f \in \mathcal{F}_L^{1,1}$. Then it holds

$$\frac{1}{L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \leq \langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \tag{3}$$

Proof.

First, recall the following result about Lipschitz gradient functions $h \in \mathcal{F}_L^{1,1}$

$$h(\mathbf{y}) \leq h(\mathbf{x}) + \langle \nabla h(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|_2^2. \tag{4}$$

To prove the result, let $\phi(\mathbf{y}) := f(\mathbf{y}) - \langle \nabla f(\mathbf{x}), \mathbf{y} \rangle$, with $\nabla \phi(\mathbf{y}) = \nabla f(\mathbf{y}) - \nabla f(\mathbf{x})$. Clearly, $\phi(\mathbf{y})$ attains its minimum value at $\mathbf{y}^\star = \mathbf{x}$. Now, let us apply (4) as follows

$$\phi(\mathbf{x}) \leq \phi\left(\mathbf{y} - \frac{1}{L}\nabla \phi(\mathbf{y})\right) \leq \phi(\mathbf{y}) - \frac{1}{2L}\|\nabla \phi(\mathbf{y})\|_2^2$$

which yields

$$f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{1}{2L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|_2^2 \leq f(\mathbf{y}) \tag{5}$$

By adding two copies of (5) with each other $\mathbf{x}$ and $\mathbf{y}$ swapped, we obtain (3).

□

The proof of convergence rates - part I

Theorem

If $f \in \mathcal{F}_L^{2,1}$, with the choice $\alpha = \frac{1}{L}$, the iterates of GD satisfy

$$\boxed{f(\mathbf{x}^k) - f(\mathbf{x}^\star) \leq \frac{2L}{k+4}\|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2} \tag{6}$$

Proof - part I

- Consider the constant step-size iteration $\mathbf{x}^{k+1} = \mathbf{x}^k - \alpha\nabla f(\mathbf{x}^k)$.
- Let $r_k := \|\mathbf{x}^k - \mathbf{x}^\star\|$. Show $\boxed{r_k \leq r_0}$.

$$\begin{aligned} r_{k+1}^2 &:= \|\mathbf{x}^{k+1} - \mathbf{x}^\star\|^2 = \|\mathbf{x}^k - \mathbf{x}^\star - \alpha\nabla f(\mathbf{x}^k)\|^2 \\ &= \|\mathbf{x}^k - \mathbf{x}^\star\|^2 - 2\alpha\langle\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^\star), \mathbf{x}^k - \mathbf{x}^\star\rangle + \alpha^2\|\nabla f(\mathbf{x}^k)\|^2 \\ &\leq r_k^2 - \alpha(2/L - \alpha)\|\nabla f(\mathbf{x}^k)\|^2 \quad \text{(by (3))} \\ &< r_k^2, \quad \forall \alpha < 2/L. \end{aligned}$$

Hence, the gradient iterations are contractive when $\alpha < 2/L$ for all $k \geq 0$.

- **An auxiliary result:** Let $\Delta_k := f(\mathbf{x}^k) - f^\star$. Show $\boxed{\Delta_k \leq r_0\|\nabla f(\mathbf{x}^k)\|}$.

$$\Delta_k \leq \langle\nabla f(\mathbf{x}^k), \mathbf{x}^k - \mathbf{x}^\star\rangle \leq \|\nabla f(\mathbf{x}^k)\|\|\mathbf{x}^k - \mathbf{x}^\star\| = r_k\|\nabla f(\mathbf{x}^k)\| \leq r_0\|\nabla f(\mathbf{x}^k)\|.$$

The proof of convergence rates - part II

Proof - part II

- We can establish **convergence** along with the auxiliary result above:

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \rangle + \frac{L_f}{2}\|\mathbf{x}^{k+1} - \mathbf{x}^k\|^2$$
$$\leq f(\mathbf{x}^k) - \omega_k \|\nabla f(\mathbf{x}^k)\|^2, \quad \omega_k := \alpha(1 - L\alpha/2).$$

Combine three inequalities to get $\boxed{\Delta_{k+1} \leq \Delta_k - (\omega_k/r_0^2)\Delta_k^2}$. Thus, dividing by $\Delta_{k+1}\Delta_k$

$$\Delta_{k+1}^{-1} \geq \Delta_k^{-1} + (\omega_k/r_0^2)\Delta_k/\Delta_{k+1} \geq \Delta_k^{-1} + (\omega_k/r_0^2).$$

By induction, we have $\Delta_{k+1}^{-1} \geq \Delta_0^{-1} + (\omega_k/r_0^2)(k+1)$, which implies

$$f(\mathbf{x}^k) - f(\mathbf{x}^\star) \leq \frac{2(f(\mathbf{x}_0) - f(\mathbf{x}^\star))\|\mathbf{x}_0 - \mathbf{x}^\star\|_2^2}{2\|\mathbf{x}_0 - \mathbf{x}^\star\|_2^2 + k\alpha(2 - \alpha L)(f(\mathbf{x}_0) - f^\star)},$$

- In order to choose the **optimal** step-size, we maximize the function $\phi(\alpha) = \alpha(2 - \alpha L)$. Hence, the optimal step size for the gradient method for $f \in \mathcal{F}_L^{1,1}$ is given by $\alpha = \frac{1}{L}$.
- Finally, since $f(\mathbf{x}_0) \leq f^* + \nabla f(\mathbf{x}^\star)^T(\mathbf{x}_0 - \mathbf{x}^\star) + (L/2)\|\mathbf{x}_0 - \mathbf{x}^\star\|_2^2 = f^* + (L/2)r_0^2$, we obtain (6).

□

The proof of convergence rates - part III

Theorem

- *If $f \in \mathcal{F}^{2,1}_{L,\mu}$, with the choice $\alpha = \frac{2}{L+\mu}$, the iterates of GD satisfy*

$$\boxed{\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2} \tag{7}$$

- *If $f \in \mathcal{F}^{2,1}_{L,\mu}$, with the choice $\alpha = \frac{1}{L}$, the iterates of GD satisfy*

$$\boxed{\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^{\frac{k}{2}} \|\mathbf{x}^0 - \mathbf{x}^\star\|_2} \tag{8}$$

Before proving the convergence rate, we first need a result about functions in $\mathcal{F}^{1,1}_{L,\mu}$

Theorem

If $f \in \mathcal{F}^{1,1}_{L,\mu}$, then for any $\mathbf{x}$ and $\mathbf{y}$, we have

$$\langle \nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle \geq \frac{\mu L}{\mu + L}\|\mathbf{x} - \mathbf{y}\|^2 + \frac{1}{\mu + L}\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2. \tag{9}$$

The proof of convergence rates - part III

Proof of (7) and (8)

- Let $r_k = \|\mathbf{x}^k - \mathbf{x}^\star\|$. Then, using (9), we have

$$\begin{aligned} r_{k+1}^2 &= \|\mathbf{x}_{k+1} - \mathbf{x}^\star - \alpha \nabla f(\mathbf{x}^k)\|^2 \\ &= r_k^2 - 2\alpha \langle f(\mathbf{x}^k), \mathbf{x}^k - \mathbf{x}^\star \rangle + \alpha^2 \|\nabla f(\mathbf{x}^k)\|^2 \\ &\leq \left(1 - \frac{2\alpha\mu L}{\mu + L}\right) r_k^2 + \alpha \left(\alpha - \frac{2}{\mu + L}\right) \|\nabla f(\mathbf{x}^k)\|^2 \end{aligned}$$

- Since $\alpha \leq \frac{2}{\mu+L}$, the last term in the previous inequality is less than 0, therefore

$$r_{k+1}^2 \leq \left(1 - \frac{2\alpha\mu L}{\mu + L}\right)^k r_0^2$$

- Plugging $\alpha = \frac{1}{L}$ and $\alpha = \frac{2}{\mu+L}$, we obtain the rates as advertised.
- For $f \in \mathcal{F}_{L,\mu}^{1,1}$, the **optimal** step-size is given by $\alpha = \frac{2}{\mu+L}$ (i.e., it optimizes the worst case bound).

□

Convergence rate of gradient descent

Convergence rate of gradient descent

$$f \in \mathcal{F}_L^{2,1}, \quad \alpha = \frac{1}{L} \qquad f(\mathbf{x}^k) - f(\mathbf{x}^\star) \leq \frac{2L}{k+4}\|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2$$

$$f \in \mathcal{F}_{L,\mu}^{2,1}, \quad \alpha = \frac{2}{L+\mu} \qquad \|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2$$

$$f \in \mathcal{F}_{L,\mu}^{2,1}, \quad \alpha = \frac{1}{L} \qquad \|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{L-\mu}{L+\mu}\right)^{\frac{k}{2}} \|\mathbf{x}^0 - \mathbf{x}^\star\|_2$$

Remarks

- Assumption: Lipschitz gradient. Result: convergence rate in **objective values**.
- Assumption: Strong convexity. Result: convergence rate in **sequence** of the iterates and in **objective values**.
- Note that the suboptimal step-size choice $\alpha = \frac{1}{L}$ adapts to the strongly convex case (i.e., it features a linear rate vs. the standard sublinear rate).

Example: Ridge regression

Optimization formulation

- Let $\mathbf{A} \in \mathbb{R}^{n \times p}$ and $\mathbf{b} \in \mathbb{R}^n$ given by the model $\mathbf{b} = \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$, where $\mathbf{w} \in \mathbb{R}^n$ is some noise.
- We can try to estimate $\mathbf{x}^\natural$ by solving the Tikhonov regularized least squares

$$\min_{\mathbf{x} \in \mathbb{R}^p} f(\mathbf{x}) := \frac{1}{2}\|\mathbf{b} - \mathbf{A}\mathbf{x}\|_2^2 + \frac{\rho}{2}\|\mathbf{x}\|_2^2.$$

where $\rho \geq 0$ is a regularization parameter.

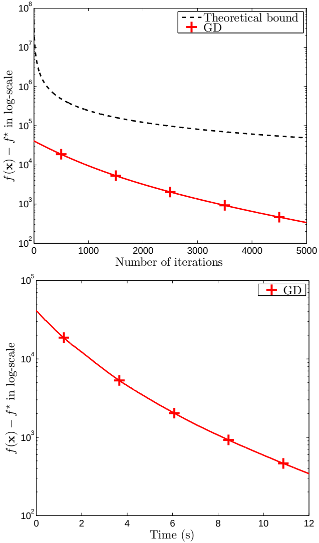
Remarks

- $f \in \mathcal{F}_{L,\mu}^{2,1}$ with:
  - $L = \lambda_p(\mathbf{A}^T\mathbf{A}) + \rho$;
  - $\mu = \lambda_1(\mathbf{A}^T\mathbf{A}) + \rho$;
  - where $\lambda_1 \leq \ldots \leq \lambda_p$ are the eigenvalues of $\mathbf{A}^T\mathbf{A}$.
- The ratio $\frac{L}{\mu}$ decreases as ρ increases, leading to faster linear convergence.
- Note that if $n < p$ and $\rho = 0$, we have $\mu = 0$, hence $f \in \mathcal{F}_L^{2,1}$ and we can expect only $O(1/k)$ convergence from the gradient descent method.

Example: Ridge regression

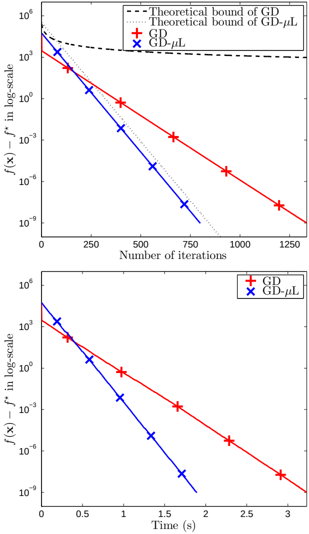
Case 1:

$n = 500, p = 2000, \rho = 0$



Case 2:

$n = 500, p = 2000, \rho = 0.01\lambda_p(\mathbf{A}^T\mathbf{A})$



Information theoretic lower bounds [4]

What is the **best** achievable rate for a **first-order** method?

$f \in \mathcal{F}_L^{\infty,1}$: Smooth and Lipschitz-gradient

It is possible to construct a function in $\mathcal{F}_L^{\infty,1}$, for which **any** first order method must satisfy

$$f(\mathbf{x}^k) - f(\mathbf{x}^\star) \geq \frac{3L}{32(k+1)^2}\|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2 \quad \text{for all } k \leq (p-1)/2$$

$f \in \mathcal{F}_{L,\mu}^{\infty,1}$: Smooth and strongly convex

It is possible to construct a function in $\mathcal{F}_{L,\mu}^{\infty,1}$, for which **any** first order method must satisfy

$$\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \geq \left(\frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2$$

Gradient descent is $O(1/k)$ for $\mathcal{F}_L^{\infty,1}$ and it is slower for $\mathcal{F}_{L,\mu}^{\infty,1}$, hence it is not optimal!

Accelerated Gradient Descent algorithm

Problem

Can we design an optimal first-order method, whose convergence rates match the theoretical lower bounds?

Solution

Accelerated Gradient Descent (AGD) methods achieve optimal convergence rates at a negligible increase in the computational cost.

Accelerated Gradient Descent for $\mathcal{F}_L^{1,1}$ (AGD-L)

1. Choose $\mathbf{x}^0 \in \text{dom}(f)$. Set $\mathbf{y}^0 := \mathbf{x}^0$ and $t_0 := 1$.

2. For $k = 0, 1, \ldots,$ iterate

$$\begin{cases} \mathbf{x}^{k+1} & = \mathbf{y}^k - \frac{1}{L}\nabla f(\mathbf{y}^k) \\ t_{k+1} & = 0.5(1 + \sqrt{4t_k^2 + 1}), \\ \gamma_{k+1} & = (t_k - 1)/t_{k+1}, \\ \mathbf{y}^{k+1} & = \mathbf{x}^{k+1} + \gamma_{k+1}(\mathbf{x}^{k+1} - \mathbf{x}^k) \end{cases}$$

Accelerated Gradient Descent for $\mathcal{F}_{L,\mu}^{1,1}$ (AGD-μL)

1. Choose $\mathbf{x}^0 \in \text{dom}(f)$ and set $\mathbf{y}^0 := \mathbf{x}^0$.

2. For $k = 0, 1, \ldots,$ iterate

$$\begin{cases} \mathbf{x}^{k+1} & = \mathbf{y}^k - \frac{1}{L}\nabla f(\mathbf{y}^k) \\ \mathbf{y}^{k+1} & = \mathbf{x}^{k+1} + \gamma(\mathbf{x}^{k+1} - \mathbf{x}^k) \end{cases}$$

where $\gamma = \frac{\sqrt{L} - \sqrt{\mu}}{\sqrt{L} + \sqrt{\mu}}$.

Global convergence of AGD [4]

Theorem (f is convex with Lipschitz gradient)

If $f \in \mathcal{F}_L^{1,1}$ or $\mathcal{F}_{L,\mu}^{1,1}$, the sequence $\{\mathbf{x}^k\}_{k\geq 0}$ generated by **AGD-L** *satisfies*

$$f(\mathbf{x}^k) - f^\star \leq \frac{4L}{(k+2)^2}\|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2, \ \forall k \geq 0. \tag{10}$$

Global convergence of AGD [4]

Theorem (f is convex with Lipschitz gradient)

If $f \in \mathcal{F}_L^{1,1}$ or $\mathcal{F}_{L,\mu}^{1,1}$, the sequence $\{\mathbf{x}^k\}_{k\geq 0}$ generated by **AGD-L** *satisfies*

$$f(\mathbf{x}^k) - f^\star \leq \frac{4L}{(k+2)^2}\|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2, \ \forall k \geq 0. \tag{10}$$

AGD-L is **optimal** *for $\mathcal{F}_L^{1,1}$ but NOT for $\mathcal{F}_{L,\mu}^{1,1}$!*

Theorem (f is strongly convex with Lipschitz gradient)

If $f \in \mathcal{F}_{L,\mu}^{1,1}$, the sequence $\{\mathbf{x}^k\}_{k\geq 0}$ generated by **AGD-μL** *satisfies*

$$f(\mathbf{x}^k) - f^\star \leq L\left(1 - \sqrt{\frac{\mu}{L}}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2^2, \ \forall k \geq 0 \tag{11}$$

$$\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \sqrt{\frac{2L}{\mu}}\left(1 - \sqrt{\frac{\mu}{L}}\right)^{\frac{k}{2}} \|\mathbf{x}^0 - \mathbf{x}^\star\|_2, \ \forall k \geq 0. \tag{12}$$

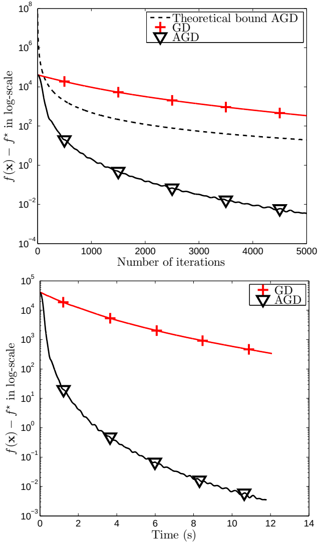
- AGD-L's iterates are not guarantee to converge.
- AGD-L does not have a **linear** convergence rate for $\mathcal{F}_{L,\mu}^{1,1}$.
- AGD-μL does, but needs to know μ.

AGD achieves the iteration lowerbound within a constant!

Example: Ridge regression

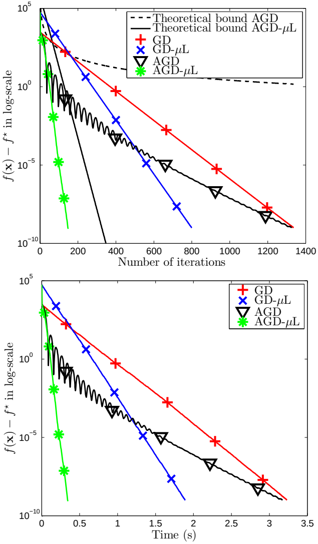
Case 1:

$n = 500, p = 2000, \rho = 0$

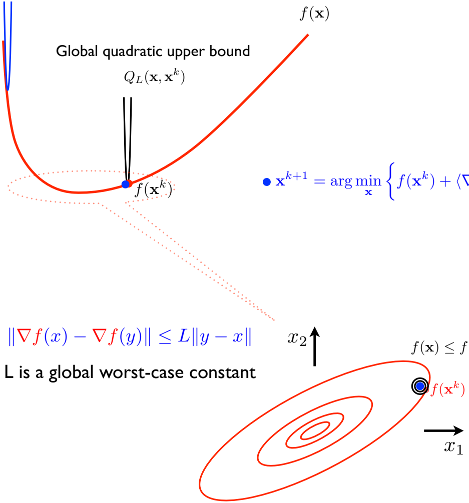


Case 2:

$n = 500, p = 2000, \rho = 0.01\lambda_p(\mathbf{A}^T\mathbf{A})$



How can we better adapt to the local geometry?



Global quadratic upper bound

$Q_L(\mathbf{x}, \mathbf{x}^k)$

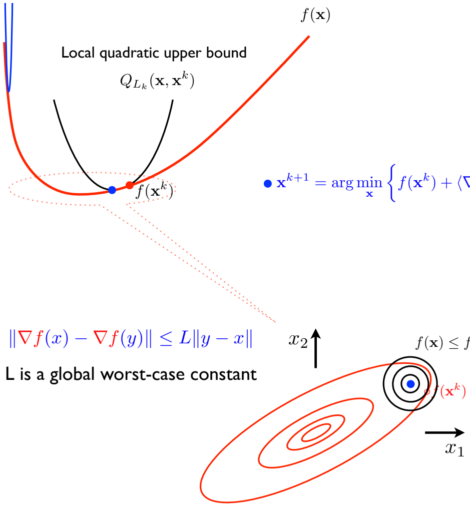
- $\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{x}^k\|_2^2 \right\}$

$\|\nabla f(x) - \nabla f(y)\| \leq L\|y - x\|$

L is a global worst-case constant

$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T(\mathbf{x} - \mathbf{x}^k) + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

How can we better adapt to the local geometry?


$f(\mathbf{x})$
Local quadratic upper bound
$Q_{L_k}(\mathbf{x}, \mathbf{x}^k)$
$f(\mathbf{x}^k)$
x_2
$f(\mathbf{x}^k)$
x_1

- $\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L_k}{2} \|\mathbf{x} - \mathbf{x}^k\|_2^2 \right\}$

$\|\nabla f(x) - \nabla f(y)\| \leq L\|y - x\|$

L is a global worst-case constant

$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T(\mathbf{x} - \mathbf{x}^k) + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

applies only locally

Enhancements

Two enhancements

1. Line-search for evaluating L for both GD and AGD.
2. Restart strategies for AGD.

When do we need a line-search procedure?

We can use a line-search procedure for both GD and AGD when

- L is **known** but it is expensive to evaluate;
- The global constant L usually does not capture the local behavior of f or it is **unknown**;

Line-search for Gradient Descent

Line-search

At each iteration, we try to find a constant L_k that satisfies

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L_k}{2} \|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q_{L_k}(\mathbf{x}, \mathbf{x}^k)$$

Define also, $\mathcal{S}_L(\mathbf{x}) := \mathbf{x} - \frac{1}{L} \nabla f(\mathbf{x})$.

Line-search gradient descent scheme (LSGD)

1. Choose $\mathbf{x}^0 \in \text{dom}(f)$ arbitrarily as a starting point.

2. For $k = 0, 1, \cdots$, generate a sequence $\{\mathbf{x}^k\}_{k \geq 0}$ as:

2.a. Find the smallest $j \geq 0$ such that $L_k^j := 2^j L_0$ satisfying

$$f\left(\mathcal{S}_{L_k^j}\left(\mathbf{x}^k\right)\right) \leq Q_{L_k^j}\left(\mathcal{S}_{L_k^j}\left(\mathbf{x}^k\right), \mathbf{x}^k\right),$$

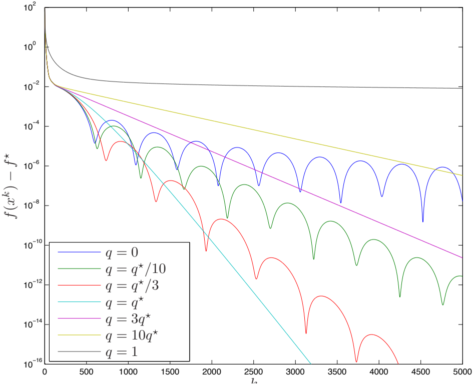
where $L_0 > 0$ is given (e.g., $\frac{\|\nabla f(\mathbf{x}^1) - \nabla f(\mathbf{x}^0)\|_2}{\|\mathbf{x}^1 - \mathbf{x}^0\|_2}$).

2.b. Update

$$\mathbf{x}^{k+1} = \mathcal{S}_{L_k^j}(\mathbf{x}^k) = \mathbf{x}^k - \frac{1}{L_k^j} \nabla f(\mathbf{x}^k)$$

Oscillatory behavior of AGD

- Minimizing a quadratic function $f(\mathbf{x}) = \mathbf{x}^T\mathbf{\Phi}\mathbf{x}$, with $p = 200$ and $\kappa(\mathbf{\Phi}) = L/\mu = 2.4 \times 10^4$
- Use stepsize $\alpha = 1/L$ and update $\mathbf{x}^{k+1} + \gamma_{k+1}(\mathbf{x}^{k+1} - \mathbf{x}^k)$ where
  - $\gamma_{k+1} = \theta_k(1-\theta_k)/(\theta_k^2 + \theta_{k+1})$
  - θ_{k+1} solves $\theta_{k+1}^2 = (1-\theta_{k+1})\theta_k^2 + q\theta_{k+1}$.
- The parameter q should be equal to the reciprocal of condition number $q = \mu/L$.
- A different choice of q might lead to oscillatory behaviour.



Enhancements

Two enhancements

1. Line-search for evaluating L for both GD and AGD.
2. Restart strategies for AGD.

When do we need a line-search procedure?

We can use a line-search procedure for both GD and AGD when

- L is **known** but it is expensive to evaluate;
- The global constant L usually does not capture the local behavior of f or it is **unknown**;

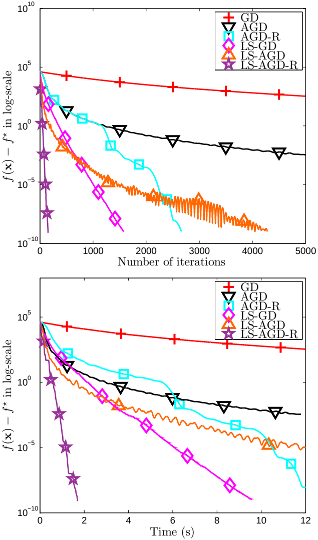
Why do we need a restart strategy?

- AGD-μL requires knowledge of μ and AGD-L does not have optimal convergence for strongly convex f.
- AGD is non-monotonic (i.e., $f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k)$ is not always satisfied).
- AGD has a periodic behavior, where the momentum depends on the local condition number $c := L/\mu$ (μ is the local strong convexity parameter).
- A **restart strategy** tries to reset this momentum whenever we observe high periodic behavior. We often use function values but other strategies are possible.

Example: Ridge regression

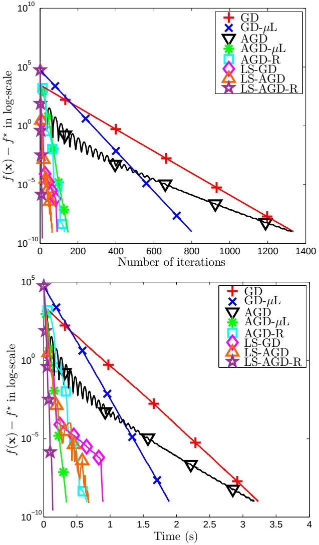
Case 1:

$n = 500, p = 2000, \rho = 0$



Case 2:

$n = 500, p = 2000, \rho = 0.01\lambda_p(\mathbf{A}^T\mathbf{A})$



Performance of optimization algorithms

Time-to-reach ϵ

`time-to-reach` ϵ `= number of iterations to reach` ϵ $\times$ `per iteration time`

The **speed** of numerical solutions depends on two factors:

- **Convergence rate** determines the number of iterations needed to obtain an ϵ-optimal solution.
- **Per-iteration time** depends on the information oracles, implementation, and the computational platform.

In general, convergence rate and per-iteration time are inversely proportional.
Finding the **fastest** algorithm is tricky! A non-exhaustive illustration:

Assumptions on f	Algorithm	Convergence rate	Iteration complexity
Lipschitz-gradient $f \in \mathcal{F}_L^{2,1}(\mathbb{R}^p)$	Gradient descent	Sublinear $(1/k)$	One gradient
	Accelerated GD	Sublinear $(1/k^2)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system
Strongly convex, smooth $f \in \mathcal{F}_{L,\mu}^{2,1}(\mathbb{R}^p)$	Gradient descent	Linear (e^{-k})	One gradient
	Accelerated GD	Linear (e^{-k})	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Linear (e^{-k}), Quadratic	One gradient, one linear system
Self-concordant, smooth	Gradient descent	Sublinear $(1/k)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system

Performance of optimization algorithms

A non-exhaustive comparison:

Assumptions on f	Algorithm	Convergence rate	Iteration complexity
Lipschitz-gradient $f \in \mathcal{F}_L^{2,1}(\mathbb{R}^p)$	Gradient descent	Sublinear $(1/k)$	One gradient
	Accelerated GD	Sublinear $(1/k^2)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system
Strongly convex, smooth $f \in \mathcal{F}_{L,\mu}^{2,1}(\mathbb{R}^p)$	Gradient descent	Linear (e^{-k})	One gradient
	Accelerated GD	Linear (e^{-k})	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Linear (e^{-k}), Quadratic	One gradient, one linear system
Self-concordant, smooth	Gradient descent	Sublinear $(1/k)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system

Accelerated gradient descent:

$$\mathbf{x}^{k+1} = \mathbf{y}^k - \alpha \nabla f(\mathbf{y}^k)$$
$$\mathbf{y}^{k+1} = \mathbf{x}^{k+1} + \gamma_{k+1}(\mathbf{x}^{k+1} - \mathbf{x}^k).$$

for some proper choice of α and γ_{k+1}.

Performance of optimization algorithms

A non-exhaustive comparison:

Assumptions on f	Algorithm	Convergence rate	Iteration complexity
Lipschitz-gradient $f \in \mathcal{F}_L^{2,1}(\mathbb{R}^p)$	Gradient descent	Sublinear $(1/k)$	One gradient
	Accelerated GD	Sublinear $(1/k^2)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system
Strongly convex, smooth $f \in \mathcal{F}_{L,\mu}^{2,1}(\mathbb{R}^p)$	Gradient descent	Linear (e^{-k})	One gradient
	Accelerated GD	Linear (e^{-k})	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Linear (e^{-k}), Quadratic	One gradient, one linear system
Self-concordant, smooth	Gradient descent	Sublinear $(1/k)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system

Main computations of the Quasi-Newton method, which we will discuss in the sequel

$$\mathbf{p}^k = -\mathbf{B}_k^{-1}\nabla f(\mathbf{x}^k) \ ,$$

where $\mathbf{B}_k^{-1}$ is updated at each iteration by adding a rank-2 matrix.

Performance of optimization algorithms

A non-exhaustive comparison:

Assumptions on f	Algorithm	Convergence rate	Iteration complexity
Lipschitz-gradient $f \in \mathcal{F}_L^{2,1}(\mathbb{R}^p)$	Gradient descent	Sublinear $(1/k)$	One gradient
	Accelerated GD	Sublinear $(1/k^2)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system
Strongly convex, smooth $f \in \mathcal{F}_{L,\mu}^{2,1}(\mathbb{R}^p)$	Gradient descent	Linear (e^{-k})	One gradient
	Accelerated GD	Linear (e^{-k})	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Linear (e^{-k}), Quadratic	One gradient, one linear system
Self-concordant, smooth	Gradient descent	Sublinear $(1/k)$	One gradient
	Quasi-Newton	Superlinear	One gradient, rank-2 update
	Newton method	Sublinear $(1/k)$, Quadratic	One gradient, one linear system

The main computation of the Newton method we discuss in the sequel requires the solution of the linear system

$$\nabla^2 f(\mathbf{x}^k)\mathbf{p}^k = -\nabla f(\mathbf{x}^k) \ .$$

A detour: Linear systems of equations

Problem (Solving a linear system)

Which is the best method for solving the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b}\ ?$$

Solving a linear system via optimization

To find a solution to the linear system, we can also solve the optimization problem

$$\min_{\mathbf{x}} f_{\mathbf{A},\mathbf{b}}(\mathbf{x}) := \frac{1}{2}\langle \mathbf{A}\mathbf{x}, \mathbf{x}\rangle - \langle \mathbf{b}, \mathbf{x}\rangle.$$

- $f_{\mathbf{A},\mathbf{b}}$ is a quadratic function with **Lipschitz-gradient** $(L = \|\mathbf{A}\|)$.
- If $\mathbf{A}$ is a $p \times p$ symmetric positive definite matrix, (i.e., $\mathbf{A} = \mathbf{A}^T \succ 0$), $f_{\mathbf{A}}$ is also **strongly convex** $(\mu = \lambda_1(\mathbf{A}))$[1].
- if $\mathbf{A}$ is not symmetric, but full column rank, we can consider the system

  $$\mathbf{A}^T\mathbf{A}\mathbf{x} = \mathbf{A}^T\mathbf{b}$$

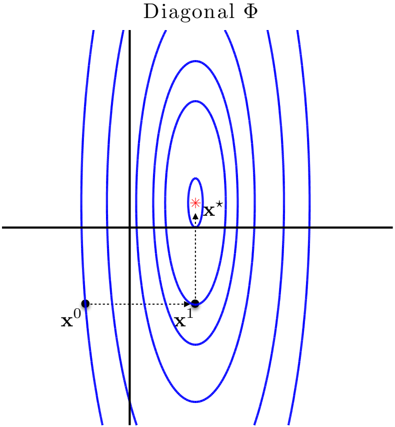
  which can be seen as: $\mathbf{\Phi}\mathbf{x} = \mathbf{y}$ where $\mathbf{\Phi}$ is symmetric and positive definite.

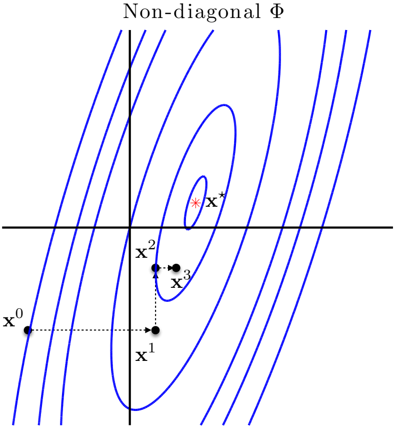
[1] $\lambda_1(\mathbf{A})$ is the smallest eigenvalue of $\mathbf{A}$

A detour: Linear systems of equations

Remark

If $\mathbf{\Phi}$ is diagonal and positive definite, given a starting point $\mathbf{x}^0 \in \mathrm{dom}(f)$, successive minimization of $f_{\mathbf{\Phi},\mathbf{y}}(\mathbf{x})$ along the coordinate axes yield $\mathbf{x}^\star$ is at most p steps.





How can we adapt to the geometry of Φ?

Conjugate gradients method- $\mathbf{\Phi}$ symmetric and positive definite

Generate a set of *conjugate* directions $\{\mathbf{p}^0, \mathbf{p}^1, \ldots, \mathbf{p}^{p-1}\}$ such that

$$\langle \mathbf{p}^i, \mathbf{\Phi}\mathbf{p}^j \rangle = 0 \qquad \text{for all } i \neq j \qquad \text{(which also implies linear independence).}$$

Successively minimize $f_{\mathbf{\Phi},\mathbf{y}}$ along the individual conjugate directions. Let

$$\mathbf{r}^k = \mathbf{\Phi}\mathbf{x}^k - \mathbf{y} \quad \text{and} \quad \mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k \ ,$$

where α_k is the minimizer of $f_{\mathbf{\Phi},\mathbf{y}}(\mathbf{x})$ along $\mathbf{x}_k + \alpha \mathbf{p}_k$, i.e.,

$$\alpha_k = -\frac{\langle \mathbf{r}^k, \mathbf{p}^k \rangle}{\langle \mathbf{p}^k, \mathbf{\Phi}\mathbf{p}^k \rangle}$$

Theorem

For any $\mathbf{x}^0 \in \mathbb{R}^p$ the sequence $\{\mathbf{x}^k\}$ generated by the conjugate directions algorithm converges to the solution $\mathbf{x}^\star$ of the linear system in **at most** *p steps.*

Intuition

The conjugate directions adapt to the geometry of the problem, taking the role of the canonical directions when $\mathbf{\Phi}$ is a generic symmetric positive definite matrix.

Conjugate gradients method

Intuition

The conjugate directions adapt to the geometry of the problem, taking the role of the canonical directions when $\mathbf{\Phi}$ is a generic symmetric positive definite matrix.

Back to diagonal

For a generic symmetric positive definite $\mathbf{\Phi}$, let us consider the variable $\bar{\mathbf{x}} := \mathbf{S}^{-1}\mathbf{x}$, with

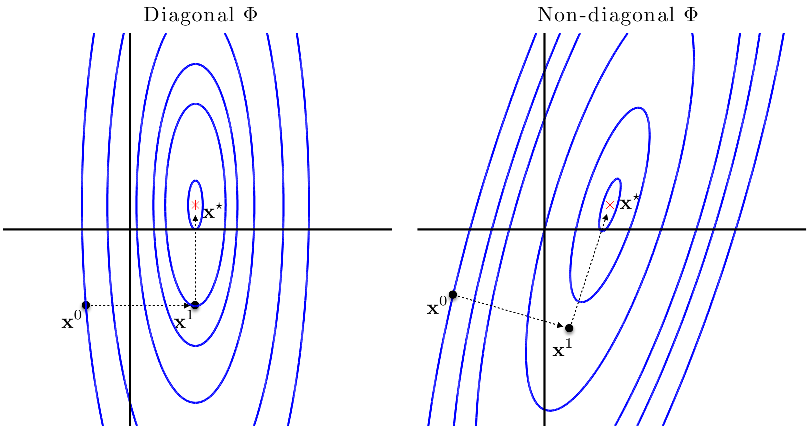
$$\mathbf{S} = \left[\mathbf{p}^0, \ldots, \mathbf{p}^{p-1}\right]$$

where $\{\mathbf{p}^k\}$ are the conjugate directions with respect to $\mathbf{\Phi}$. $f_{\mathbf{\Phi},\mathbf{y}}(\mathbf{x})$ now becomes

$$\bar{f}_{\mathbf{\Phi},\mathbf{y}}(\bar{\mathbf{x}}) := f_{\mathbf{\Phi},\mathbf{y}}(\mathbf{S}\bar{\mathbf{x}}) = \frac{1}{2}\langle \bar{\mathbf{x}}, (\mathbf{S}^T\mathbf{\Phi}\mathbf{S})\bar{\mathbf{x}}\rangle - \langle \mathbf{S}^T\mathbf{y}, \bar{\mathbf{x}}\rangle.$$

By the conjugacy property, $\langle \mathbf{p}^i, \mathbf{\Phi}\mathbf{p}^j\rangle = 0, \ \forall\ i \neq j$, the matrix $\mathbf{S}^T\mathbf{\Phi}\mathbf{S}$ is diagonal. Therefore, we can find the minimum of $\bar{f}(\bar{\mathbf{x}})$ in at most p steps along the canonical directions in $\bar{\mathbf{x}}$ space, which are the $\{\mathbf{p}^k\}$ directions in $\mathbf{x}$ space.

Conjugate directions naturally adapt to the linear operator



Conjugate gradients method

Theorem

For any $\mathbf{x}^0 \in \mathbb{R}^p$ *the sequence* $\{\mathbf{x}^k\}$ *generated by the conjugate directions algorithm converges to the solution* $\mathbf{x}^\natural$ *of the linear system in* **at most** p *steps.*

Proof.

Since $\{\mathbf{p}^k\}$ are linearly independent, they span $\mathbb{R}^p$. Therefore, we can write

$$\mathbf{x}^\star - \mathbf{x}^0 = a_0\mathbf{p}^0 + a_1\mathbf{p}^1 + \cdots + a_{p-1}\mathbf{p}^{p-1}$$

for some values of the coefficients a_k. By multiplying on the left by $(\mathbf{p}^k)^T\mathbf{\Phi}$ and using the conjugacy property, we obtain

$$a_k = \frac{\langle \mathbf{p}^k, \mathbf{\Phi}(\mathbf{x}^\star - \mathbf{x}^0)\rangle}{\langle \mathbf{p}^k, \mathbf{\Phi}\mathbf{p}^k\rangle}.$$

Since $\mathbf{x}^k = \mathbf{x}^{k-1} + \alpha_{k-1}\mathbf{p}^{k-1}$, we have $\mathbf{x}^k = \mathbf{x}^0 + \alpha_0\mathbf{p}^0 + \alpha_1\mathbf{p}^1 + \cdots + \alpha_{k-1}\mathbf{p}^{k-1}$. By premultiplying by $(\mathbf{p}^k)^T\mathbf{\Phi}$ and using the conjugacy property, we obtain $\langle \mathbf{p}^k, \mathbf{\Phi}(\mathbf{x}^k - \mathbf{x}^0)\rangle = 0$ which implies

$$\langle \mathbf{p}^k, \mathbf{\Phi}(\mathbf{x}^\star - \mathbf{x}^0)\rangle = \langle \mathbf{p}^k, \mathbf{\Phi}(\mathbf{x}^\star - \mathbf{x}^k)\rangle = \langle \mathbf{p}^k, \mathbf{y} - \mathbf{\Phi}\mathbf{x}^0)\rangle = -\langle \mathbf{p}^k, \mathbf{r}^k\rangle$$

so that $a_k = -\frac{\langle \mathbf{p}^k, \mathbf{r}^k\rangle}{\langle \mathbf{p}^k, \mathbf{\Phi}\mathbf{p}^k\rangle} = \alpha_k$. □

Conjugate gradients method

How can we efficiently generate a set of conjugate directions?

Iteratively generate the new descent direction $\mathbf{p}^k$ from the previous one:

$$\mathbf{p}^k = -\mathbf{r}^k + \beta_k \mathbf{p}^{k-1}$$

For ensuring conjugacy $\langle \mathbf{p}^k, \mathbf{\Phi}\mathbf{p}^{k-1}\rangle = 0$, we need to choose β_k as

$$\beta_k = \frac{\langle \mathbf{r}^k, \mathbf{\Phi}\mathbf{p}^{k-1}\rangle}{\langle \mathbf{p}^{k-1}, \mathbf{\Phi}\mathbf{p}^{k-1}\rangle}\ .$$

Lemma

The directions $\{\mathbf{p}^0, \mathbf{p}^1, \ldots, \mathbf{p}^p\}$ *form a* conjugate directions *set.*

Conjugate gradients method

Conjugate gradients (CG) method

1 Initialization:

- **1.a** Choose $\mathbf{x}^0 \in \mathrm{dom}(f)$ arbitrarily as a starting point.
- **1.b** Set $\mathbf{r}^0 = \mathbf{\Phi}\mathbf{x}^0 - \mathbf{y}$, $\mathbf{p}^0 = -\mathbf{r}^0$, $k = 0$.

2. While $\mathbf{r}^k \neq \mathbf{0}$, generate a sequence $\{\mathbf{x}^k\}_{k \geq 0}$ as:

$$\begin{aligned}
\alpha_k &= -\frac{\langle \mathbf{r}^k, \mathbf{p}^k \rangle}{\langle \mathbf{p}^k, \mathbf{\Phi}\mathbf{p}^k \rangle} \\
\mathbf{x}^{k+1} &= \mathbf{x}^k + \alpha_k \mathbf{p}^k \\
\mathbf{r}^{k+1} &= \mathbf{\Phi}\mathbf{x}^{k+1} - \mathbf{y} \\
\beta_{k+1} &= \frac{\langle \mathbf{r}^{k+1}, \mathbf{\Phi}\mathbf{p}^k \rangle}{\langle \mathbf{p}^k, \mathbf{\Phi}\mathbf{p}^k \rangle} \\
\mathbf{p}^{k+1} &= -\mathbf{r}^{k+1} + \beta_{k+1}\mathbf{p}^k \\
k &= k + 1
\end{aligned}$$

Theorem

Since the directions $\{\mathbf{p}^0, \mathbf{p}^1, \ldots, \mathbf{p}^k\}$ are conjugate, CG converges in at most p steps.

Other properties of the conjugate gradient method

Theorem

if $\mathbf{\Phi}$ has only r distinct eigenvalues, then the CG iterations will terminate at the solution in at most r iterations.

Theorem

If $\mathbf{\Phi}$ has eigenvalues $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_p$, we have that

$$\|\mathbf{x}^{k+1} - \mathbf{x}^\star\|_{\mathbf{\Phi}} \leq \left(\frac{\lambda_{p-k} - \lambda_1}{\lambda_{p-k} + \lambda_1}\right) \|\mathbf{x}^0 - \mathbf{x}^\star\|_{\mathbf{\Phi}},$$

where the local norm is given by $\|\mathbf{x}\|_{\mathbf{\Phi}} = \sqrt{\mathbf{x}^T \mathbf{\Phi} \mathbf{x}}$.

Theorem

Conjugate gradients algorithm satisfy the following iteration invariant for the solution of $\mathbf{\Phi}\mathbf{x} = \mathbf{y}$

$$\|\mathbf{x}^{k+1} - \mathbf{x}^\star\|_{\mathbf{\Phi}} \leq 2\left(\frac{\sqrt{\kappa(\mathbf{\Phi})} - 1}{\sqrt{\kappa(\mathbf{\Phi})} + 1}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_{\mathbf{\Phi}},$$

where the condition number of $\mathbf{\Phi}$ is defined as $\kappa(\mathbf{\Phi}) := \|\mathbf{\Phi}\|\|\mathbf{\Phi}^{-1}\| = \frac{\lambda_p}{\lambda_1}$.

GD and AGD for the quadratic case: choice of the step size

Gradient Descent

$$\alpha_k = \frac{2}{L+\mu} \quad \text{with } L = \lambda_p(\mathbf{\Phi}) \text{ and } \mu = \lambda_1(\mathbf{\Phi})$$

Steepest descent

Choose α_k so as to minimize $f(\mathbf{x}^{k+1})$.

$$\alpha_k = \frac{\|\nabla f(\mathbf{x}^k)\|^2}{\langle \nabla f(\mathbf{x}^k), \mathbf{\Phi}\nabla f(\mathbf{x}^k)\rangle} \tag{13}$$

Barzilai-Borwein

$$\alpha_k = \frac{\|\nabla f(\mathbf{x}^{k-1})\|^2}{\langle \nabla f(\mathbf{x}^{k-1}), \mathbf{\Phi}\nabla f(\mathbf{x}^{k-1})\rangle} \tag{14}$$

The quadratic case - convergence rates summary

Convergence rates

Gradient descent $\left(\alpha_k = \frac{2}{L+\mu}\right)$: $\quad \|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{\lambda_p - \lambda_1}{\lambda_p}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2$

Steepest descent: $\quad \|\mathbf{x}^{k+1} - \mathbf{x}^\star\|_{\mathbf{\Phi}} \leq \left(\frac{\lambda_p - \lambda_1}{\lambda_p + \lambda_1}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_{\mathbf{\Phi}}$

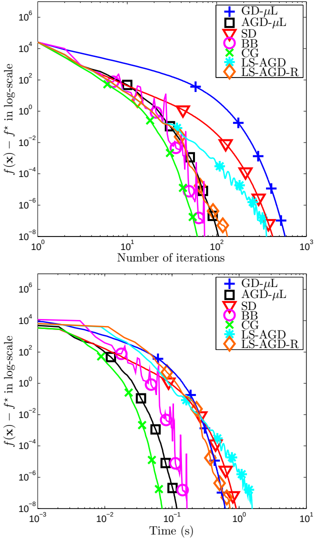
Barzilai-Borwein $(\lambda_p < 2\lambda_1)$: $\quad \|\mathbf{x}^{k+1} - \mathbf{x}^\star\|_2 \leq \left(\frac{\lambda_p - \lambda_1}{\lambda_1}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_2$

AGD-μL: $\quad \|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \left(\frac{\sqrt{\lambda_p} - \sqrt{\lambda_1}}{\sqrt{\lambda_p}}\right)^{\frac{k}{2}} \|\mathbf{x}^0 - \mathbf{x}^\star\|_2$

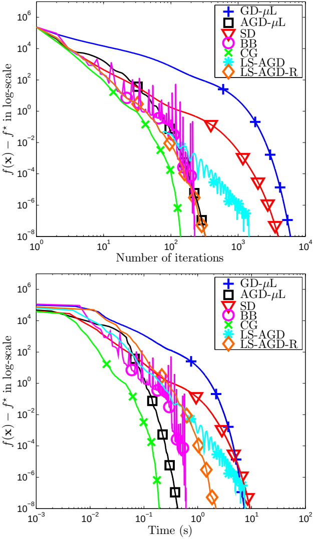
Conjugate gradient method: $\quad \|\mathbf{x}^{k+1} - \mathbf{x}^\star\|_{\mathbf{\Phi}} \leq \left(\frac{\sqrt{\lambda_p} - \sqrt{\lambda_1}}{\sqrt{\lambda_p} + \sqrt{\lambda_1}}\right)^k \|\mathbf{x}^0 - \mathbf{x}^\star\|_{\mathbf{\Phi}}$

Example: Quadratic function

Case 1: $n = p = 1000, \kappa(\mathbf{A}) = 100$



Case 2: $n = p = 1000, \kappa(\mathbf{A}) = 1000$



Time-data tradeoff for solving linear systems

Can we trade time with data?

It seems counter-intuitive that we could do so, but...

- The condition number of $\mathbf{\Phi} = \mathbf{A}^T\mathbf{A}$ **decreases** as n increases;
- The convergence rate is **faster** as the condition number decreases;
- The computational cost of each CG iteration **increases** as n increases.

Can we find a trade-off between these two trends?

Time-data tradeoff for solving linear systems

Can we trade time with data?

It seems counter-intuitive that we could do so, but...

- The condition number of $\mathbf{\Phi} = \mathbf{A}^T\mathbf{A}$ **decreases** as n increases;
- The convergence rate is **faster** as the condition number decreases;
- The computational cost of each CG iteration **increases** as n increases.

Can we find a trade-off between these two trends?

Example (Inverse problem with Gaussian coefficient matrix)

Consider the inverse problem with coefficient matrix $\mathbf{A} \in \mathbb{R}^{n\times p}$ whose entries are independent identically distributed Gaussian random variables, $A_{ij} \sim \mathcal{N}(0,1)$. We want to recover $\mathbf{x}^\natural \in \mathbb{R}^p$ from the noisy oversampled observation $\mathbf{b} \in \mathbb{R}^n$,

$$\mathbf{b} = \mathbf{A}\mathbf{x}^\natural + \mathbf{w}$$

where $w_i \sim \mathcal{N}(0,\sigma^2)$. Using conjugate gradient method, $\|\mathbf{x}^k - \mathbf{x}^\natural\|_2 \leq \varepsilon$ with very high probability after k iterations where

$$k \geq 2\log_{\frac{n}{p}} \frac{2\Big((\sqrt{n}+\sqrt{p})\|\mathbf{x}^0\|_2 + \|\mathbf{b}\|_2 + \sigma\frac{n+p}{\sqrt{n}-\sqrt{p}}\Big)}{\varepsilon(\sqrt{n}-\sqrt{p}) - \sigma\sqrt{p}} + 1. \qquad (15)$$

Time-data tradeoff

Lemma 1 (Tail bounds for eigenvalues of Wishart matrices) [3]

Maximum and minimum eigenvalues of $p \times p$ random matrix $\mathbf{\Phi} = \mathbf{A}^T\mathbf{A}$ where the entries of $n \times p$ matrix $\mathbf{A}$ are independent identically distributed Gaussian random variables satisfy for any $t > 0$

$$\mathbb{P}\left(\left| \sqrt{\lambda_p} - (\sqrt{n} + \sqrt{p})\right| \geq t\right) \leq 2e^{-t^2/2}$$

$$\mathbb{P}\left(\left| \sqrt{\lambda_1} - (\sqrt{n} - \sqrt{p})\right| \geq t\right) \leq 2e^{-t^2/2}$$

Lemma 2 (Exponential estimates for chi-square distributions)

Let $\mathbf{w}$ be an n dimensional vector with independent identically distributed Gaussian random entries $w_i \sim \mathcal{N}(0, \sigma^2)$, then for any $t > 0$

$$\mathbb{P}\left(\left|\|\mathbf{w}\|_2 \geq \sigma\sqrt{n}\right| \geq t\right) \leq 2e^{-n(t^2 - t^3)/4}$$

Time-data tradeoff

Proof.

Considering the convergence rate of the conjugate gradient method we can show

$$\|\mathbf{x}^k - \mathbf{x}^\star\|_2 \leq \frac{1}{\sqrt{\mu}}\|\mathbf{x}^k - \mathbf{x}^\star\|_{\mathbf{\Phi}} \leq \frac{2}{\sqrt{\mu}}\left(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^{k-1}\|\mathbf{x}^0 - \mathbf{x}^\star\|_{\mathbf{\Phi}}$$

which implies

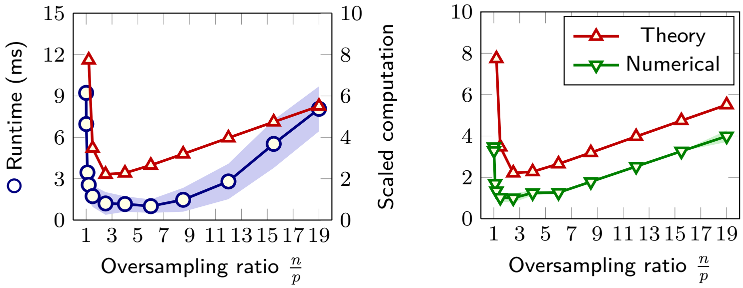
$$\begin{aligned}\|x^k - \mathbf{x}^\natural\|_2 &\leq \|\mathbf{x}^\natural - \mathbf{x}^\star\|_2 + \|x^\star - \mathbf{x}^k\|_2 \\ &\leq \|\mathbf{x}^\natural - \mathbf{x}^\star\|_2 + \frac{2}{\sqrt{\mu}}\left(\frac{\sqrt{L}-\sqrt{\mu}}{\sqrt{L}+\sqrt{\mu}}\right)^{k-1}\|\mathbf{x}^0 - \mathbf{x}^\star\|_{\mathbf{\Phi}}.\end{aligned}$$

Proof follows by equating the right hand side to ε and solving for k, after considering

$$\begin{aligned}\|\mathbf{x}^0 - \mathbf{x}^\star\|_{\mathbf{\Phi}} &\leq \|\mathbf{x}^0 - \mathbf{x}^\natural\|_{\mathbf{\Phi}} + \|\mathbf{x}^\natural - \mathbf{x}^\star\|_{\mathbf{\Phi}} \\ &\leq \sqrt{L}\|\mathbf{x}^0\|_2 + \|\mathbf{b}\|_2 + \|\mathbf{w}\|_2 + \sqrt{L}\|\mathbf{x}^\natural - \mathbf{x}^\star\|_2.\end{aligned}$$

and the concentration inequalities given in Lemma 1 and 2. □

Time-data tradeoff - Numerical results



Specifications

Results above are averaged over 100 simulations. p, σ, ε and $\mathbf{x}^0$ are chosen as 200, 10^{-2}, 0.2 and all zero vector respectively. Stopping conditions is defined as:

- (Numerical): Stop when $\|\mathbf{x}^k - \mathbf{x}^\natural\|_2 \leq \varepsilon$

For conjugate gradient method, computational complexity per iteration is proportional to n. Scaled computations considered above are obtained by taking n times number of iterations and scaling it by the minimum value obtained for computation with stopping condition explained above (numerical). We compare the theoretical computation with the runtime and the computation for numerical case.

Krylov subspaces

Let $\mathbf{y} \in \mathbb{R}^p$ and $\mathbf{\Phi}$ be an **invertible** $p \times p$ matrix.

Definition (Krylov subspaces)

Krylov subspaces are a sequence of nested subspaces $\{\mathcal{K}_0 \subseteq \mathcal{K}_1 \subseteq \cdots\}$ such that

$$\mathcal{K}_0 = \{\emptyset\}, \qquad \mathcal{K}_k = \operatorname{span}\{\mathbf{y}, \mathbf{\Phi}\mathbf{y}, \mathbf{\Phi}^2\mathbf{y}, \ldots, \mathbf{\Phi}^{k-1}\mathbf{y}\}.$$

Theorem (Key property)

Inverse of a matrix can be found in terms of linear combinations of its powers:

$$\mathbf{\Phi}^{-1}\mathbf{y} \in \mathcal{K}_p$$

Proof.

Let $p(\lambda) = \lambda^p + a_1\lambda^{p-1} + \ldots + a_{p-1}\lambda + a_p$ be a p-th order polynomial. We call $p(\lambda)$ the characteristic polynomial of $\mathbf{\Phi}$ if it evaluates to zero for all the eigenvalues of $\mathbf{\Phi}$. The Cayley-Hamilton theorem states that every matrix should satisfy its characteristic polynomial. Hence, we have $p(\mathbf{\Phi}) = \mathbf{\Phi}^p + a_1\mathbf{\Phi}^{p-1} + \cdots + a_p\mathbf{I} = 0$. By multiplying the previous equation by $\mathbf{\Phi}^{-1}\mathbf{y}$ and dividing by a_p, we obtain

$$\mathbf{\Phi}^{-1}\mathbf{y} = -\frac{1}{a_p}\left(\mathbf{\Phi}^{p-1}\mathbf{y} + a_1\mathbf{\Phi}^{p-2}\mathbf{y} + \cdots + a_{p-1}\mathbf{y}\right) \in \mathcal{K}_p.$$

Krylov subspaces methods and conjugate gradients

Krylov subspace methods

Given $f_{\mathbf{\Phi},\mathbf{y}}(\mathbf{x}) := \frac{1}{2}\langle \mathbf{\Phi}\mathbf{x}, \mathbf{x}\rangle - \langle \mathbf{y}, \mathbf{x}\rangle$, the *Krylov sequence* is defined as

$$\mathbf{x}^k = \arg\min_{\mathbf{x}\in\mathcal{K}_k} f_{\mathbf{\Phi},\mathbf{y}}(\mathbf{x})$$

- By the key property of Krylov subspaces, we have $\mathbf{x}^p = \mathbf{\Phi}^{-1}\mathbf{y}$.
- *Krylov methods* are ways of computing the *Krylov* sequence iteratively.

Remark

Conjugate gradient is an efficient iterative way for computing the *Krylov sequence* when $\mathbf{\Phi}$ is **symmetric and positive definite.**
Depending on the linear system, we can also use other Krylov subspace methods, such as the Arnoldi, Lanczos, GMRES (generalized minimum residual), BiCGSTAB (biconjugate gradient stabilized), QMR (quasi minimal residual), TFQMR (transpose-free QMR), and MINRES (minimal residual) methods.

*Nonlinear Conjugate Gradient method

Two changes:

1. Line-search instead of closed-form expression for the step-size;
2. Replace the residual $\mathbf{r}$ by the gradient of the non-linear function $f(\mathbf{x})$.

Non-Linear Conjugate Gradient (NLCG) method

```
1 Initialization:
    1.a Choose $\mathbf{x}^0 \in \mathrm{dom}(f)$ arbitrarily as a starting point.
    1.b Let $f_0 = f(\mathbf{x}^0)$ and $\nabla f_0 = \nabla f(\mathbf{x}^0)$.
    1.c Set $\mathbf{p}^0 = -\nabla f_0$, $k = 0$.
2. While $\nabla f_k \neq \mathbf{0}$
    2.a Compute the step-size $\alpha_k$ by line search;
    2.b Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k$;
    2.c Evaluate $\nabla f_{k+1}$;
    2.d Compute the updates:
```

$$
\begin{aligned}
\beta_{k+1} &= \frac{\langle \nabla f_{k+1}, \nabla f_{k+1} \rangle}{\langle \nabla f_k, \nabla f_k \rangle} \\
\mathbf{p}^{k+1} &= -\nabla f_{k+1} + \beta_{k+1} \mathbf{p}^k \\
k &= k + 1
\end{aligned}
$$

*Non-linear Conjugate Gradient method - convergence

Theorem

Suppose that

- *the level set* $\mathcal{L} := \{\mathbf{x} | f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$ *is bounded;*
- f *is Lipschitz differentiable in an open neighborhood* $\mathcal{N}$ *of* $\mathcal{L}$*;*
- *NLCG is implemented with a line-search that satisfies the strong Wolfe conditions with* $0 < c_1 < c_2 < \frac{1}{2}$.

Then

$$\liminf_{k\to\infty} \|\nabla f_k\| = 0$$

*Nonlinear Conjugate Gradient method - step-size choice

Line-search

We want $\mathbf{p}_k$ to be a **descent** direction. It must hold $\langle \nabla f_k, \mathbf{p}^k \rangle < 0$.

$$\langle \nabla f_k, \mathbf{p}^k \rangle = -\|\nabla f_k\|_2^2 + \beta_k \langle \nabla f_k, \mathbf{p}^{k-1} \rangle \tag{16}$$

- **Exact line search**: α_{k-1} is the local minimizer of f along the direction $\mathbf{p}^{k-1}$, then $\langle \nabla f_k, \mathbf{p}^{k-1} \rangle = 0$.
  In this case $\langle \nabla f_k, \mathbf{p}^k \rangle < 0$, so $\mathbf{p}^k$ is indeed a descent direction.
- **Exact line search**:
  - the second term in (16) may dominate the first, implying that $\langle \nabla f_k, \mathbf{p}^k \rangle > 0$.
  - To avoid this situation, α_k must satisfy the *strong* Wolfe conditions:

$$f(\mathbf{x}^k + \alpha_k \mathbf{p}^k) \leq f(\mathbf{x}^k) + c_1 \alpha_k \langle \nabla f_k, \mathbf{p}^k \rangle \tag{17}$$

$$|\langle \nabla f(\mathbf{x}^k + \alpha_k \mathbf{p}^k), \mathbf{p}^k \rangle| \leq -c_2 \langle \nabla f_k, \mathbf{p}^k \rangle, \tag{18}$$

where $0 < c_1 < c_2 < \frac{1}{2}$.

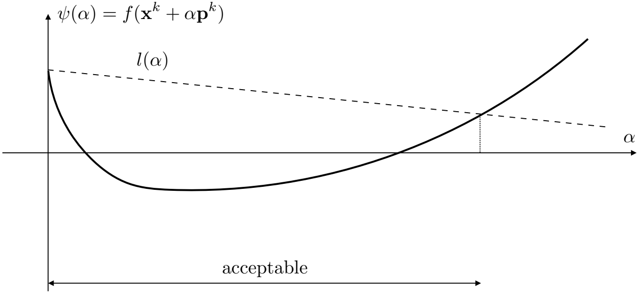
*The Wolfe conditions for line search

Sufficient decrease

The first condition stipulates that the step-size α_k should give **sufficient decrease** in the objective value

$$\psi(\alpha) := \boxed{f(\mathbf{x}^k + \alpha\mathbf{p}^k) \leq f(\mathbf{x}^k) + c_1\alpha\langle\nabla f(\mathbf{x}^k), \mathbf{p}^k\rangle} := l(\alpha)$$

for some constant $c_1 \in (0, 1)$ (usually a small value like 10^{-4} is used).



*The Wolfe conditions for line search

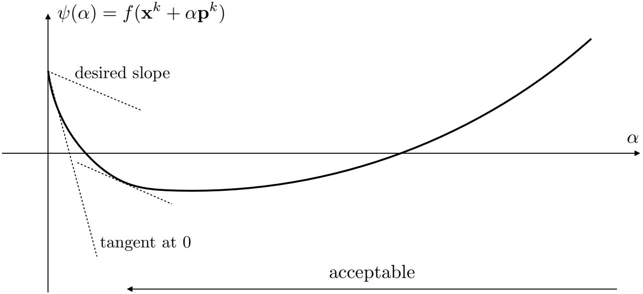
Curvature condition

The sufficient decrease condition is satisfied for all small step-sizes.
To avoid making small progress, the **curvature condition** requires

$$\psi'(\alpha) := \boxed{\langle \nabla f(\mathbf{x}^k + \alpha \mathbf{p}^k), \mathbf{p}^k \rangle \geq c_2 \langle \nabla f(\mathbf{x}^k), \mathbf{p}^k \rangle}$$

for some constant $c_2 \in (c_1, 1)$ (usually between 0.1 and 0.9).

- The slope of ψ at α_k must be greater than c_2 times the slope at 0.
- **Strong version:** $|\psi'(\alpha)| \leq -c_2 \langle \nabla f(\mathbf{x}^k), \mathbf{p}^k \rangle$.



*The Wolfe conditions for line search

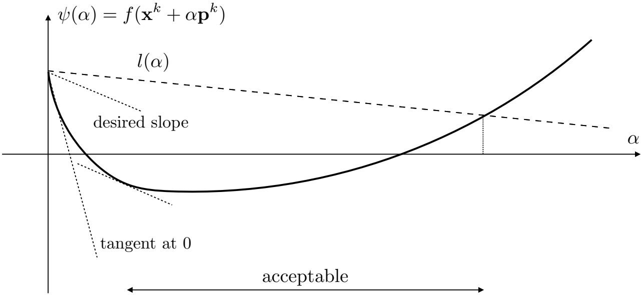
Wolfe conditions

- **Sufficient decrease**

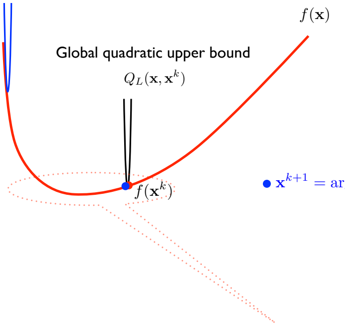
$$f(\mathbf{x}^k + \alpha \mathbf{p}^k) \leq f(\mathbf{x}^k) + c_1 \alpha \langle \nabla f(\mathbf{x}^k), \mathbf{p}^k \rangle := l(\alpha) \quad \text{for } c_1 \in (0,1)$$

- **Curvature condition**

$$\langle \nabla f(\mathbf{x}^k + \alpha \mathbf{p}^k), \mathbf{p}^k \rangle \geq c_2 \langle \nabla f(\mathbf{x}^k), \mathbf{p}^k \rangle \quad \text{for } c_2 \in (c_1, 1)$$



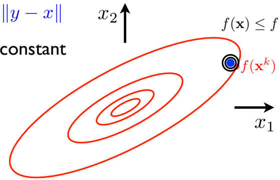
How can we better adapt to the local geometry?



- $\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2} \|\mathbf{x} - \mathbf{x}^k\|_2^2 \right\}$

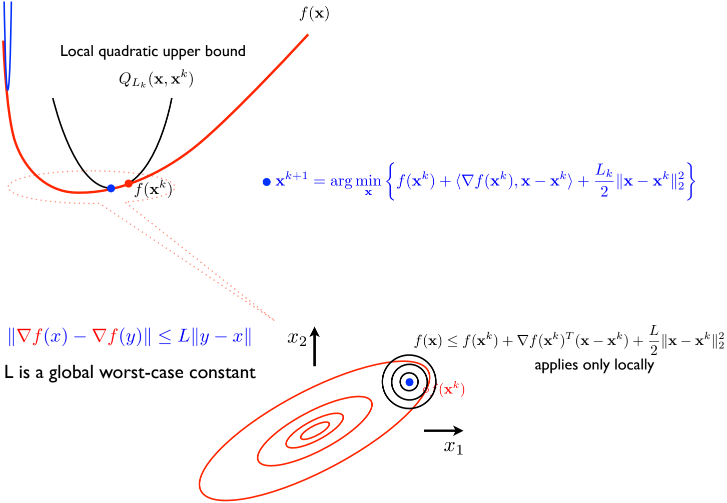
$$\|\nabla f(x) - \nabla f(y)\| \leq L\|y - x\|$$

L is a global worst-case constant



$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T (\mathbf{x} - \mathbf{x}^k) + \frac{L}{2} \|\mathbf{x} - \mathbf{x}^k\|_2^2$$

How can we better adapt to the local geometry?



$f(\mathbf{x})$
Local quadratic upper bound
$Q_{L_k}(\mathbf{x}, \mathbf{x}^k)$
$f(\mathbf{x}^k)$
x_2
x_1
$f(\mathbf{x}^k)$

- $\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L_k}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 \right\}$

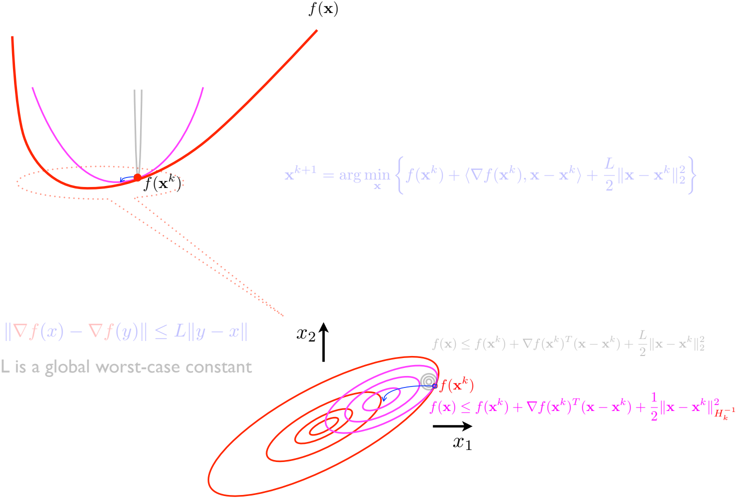
$\|\nabla f(x) - \nabla f(y)\| \leq L\|y - x\|$

L is a global worst-case constant

$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T(\mathbf{x} - \mathbf{x}^k) + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$

applies only locally

How can we better adapt to the local geometry?



$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} \left\{ f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 \right\}$$

$$\|\nabla f(x) - \nabla f(y)\| \leq L\|y - x\|$$

L is a global worst-case constant

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T(\mathbf{x} - \mathbf{x}^k) + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$$

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)^T(\mathbf{x} - \mathbf{x}^k) + \frac{1}{2}\|\mathbf{x} - \mathbf{x}^k\|_{H_k^{-1}}^2$$

Variable metric gradient descent algorithm

Variable metric gradient descent algorithm

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$ as a starting point and $\mathbf{H}_0 \succ 0$.
2. For $k = 0, 1, \cdots$, perform:

$$\begin{cases} \mathbf{d}^k & = \mathbf{H}_k \nabla f(\mathbf{x}^k), \\ \mathbf{x}^{k+1} & = \mathbf{x}^k - \alpha_k \mathbf{d}^k, \end{cases} \tag{19}$$

where $\alpha_k \in (0, 1]$ is a given step size. Update $\mathbf{H}_{k+1} \succ 0$ if necessary.

Variable metric gradient descent algorithm

Variable metric gradient descent algorithm

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$ as a starting point and $\mathbf{H}_0 \succ 0$.
2. For $k = 0, 1, \cdots$, perform:

$$\begin{cases} \mathbf{d}^k & = \mathbf{H}_k \nabla f(\mathbf{x}^k), \\ \mathbf{x}^{k+1} & = \mathbf{x}^k - \alpha_k \mathbf{d}^k, \end{cases} \tag{19}$$

where $\alpha_k \in (0, 1]$ is a given step size. Update $\mathbf{H}_{k+1} \succ 0$ if necessary.

Common choices of $\mathbf{H}_k$

- $\mathbf{H}_k = \lambda_k \mathbf{I}$, we obtain a gradient descent method.
- $\mathbf{H}_k = \mathbf{D}$ a diagonal matrix, we obtain a gradient descent method.
- $\mathbf{H}_k = \nabla^2 f(\mathbf{x}^k)^{-1}$, we obtain a Newton method, which requires solving the linear system $\nabla^2 f(\mathbf{x}^k)\mathbf{d}^k = \nabla f(\mathbf{x}^k)$ at each iteration.
- $\mathbf{H}_k \approx \nabla^2 f(\mathbf{x}^k)^{-1}$, subject to certain conditions, yields a quasi-Newton method, which needs an efficient update of the metric $\mathbf{H}_k$ at each iteration.

Quasi-Newton methods

In many problems, estimating the Hessian is expensive. Quasi-Newton methods use an approximate Hessian oracle.

Quadratic model and step-size

- Iteratively build a quadratic model of the objective function

$$f(\mathbf{x}^k + \mathbf{p}) \approx f(\mathbf{x}^k) + \langle \mathbf{p}, \nabla f(\mathbf{x}^k) \rangle + \frac{1}{2}\langle \mathbf{p}, \mathbf{B}_k \mathbf{p} \rangle := m_k(\mathbf{p}) \ ,$$

  where the symmetric positive definite matrix $\mathbf{B}_k$ is an approximation of $\nabla^2 f(\mathbf{x}^k)$.
- The search direction is the minimizer of $m_k(\mathbf{p})$, namely

$$\mathbf{p}^k = -\mathbf{B}_k^{-1} \nabla f(\mathbf{x}^k) \ .$$

- The iterates are then given by $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k$.
- The step-size α_k is chosen to satisfy the **Wolfe conditions**:

$$f(\mathbf{x}^k + \alpha_k \mathbf{p}^k) \leq f(\mathbf{x}^k) + c_1 \alpha_k \langle \nabla f(\mathbf{x}^k), \mathbf{p}^k \rangle \qquad \text{(sufficient decrease)}$$
$$\langle \nabla f(\mathbf{x}^k + \alpha_k \mathbf{p}^k), \mathbf{p}^k \rangle \geq c_2 \langle \nabla f(\mathbf{x}^k), \mathbf{p}^k \rangle \qquad \text{(curvature condition)}$$

  with $0 < c_1 < c_2 < 1$. For quasi-Newton methods, we usually use $c_1 = 0.1$.

Quasi-Newton method - convergence

Dennis & Moré condition [2]

In order for the quasi-newton methods to converge, it is **necessary and sufficient** that the matrices $\mathbf{B}_k$ satisfy the following condition on the quasi-Newton directions $\mathbf{p}^k = -\mathbf{B}_k^{-1}\nabla f(\mathbf{x}^k)$:

$$\lim_{k\to\infty} \frac{\|(\mathbf{B}_k - \nabla^2 f(\mathbf{x}^\star))\mathbf{p}^k\|}{\|\mathbf{p}^k\|} = 0 \qquad (20)$$

- It is not necessary that $\mathbf{B}_k$ converges to $\nabla^2 f(\mathbf{x}^\star)$.
- $\mathbf{B}_k$ needs to converge only along the search directions $\mathbf{p}^k$ (not conjugate this time!).

Theorem (Convergence of quasi-Newton methods)

- *Suppose that $f \in \mathcal{C}^2$.*
- *Consider the iteration $\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{B}_k^{-1}\nabla f(\mathbf{x}^k)$, (i.e., $\alpha_k = 1$ for all k).*
- *Assume also that $\{\mathbf{x}^k\}$ converges to a point $\mathbf{x}^\star$ such that $\nabla f(\mathbf{x}^\star) = 0$ and $\nabla^2 f(\mathbf{x}^\star)$ is positive definite.*

Then $\{\mathbf{x}^k\}$ converges **superlinearly** *to $\mathbf{x}^\star$ if and only if* (20) *holds.*

*Quasi-Newton methods

How do we update $\mathbf{B}_{k+1}$?

Suppose we have (note the coordinate change from $\mathbf{p}$ to $\bar{\mathbf{p}}$)

$$m_{k+1}(\bar{\mathbf{p}}) := f(\mathbf{x}^{k+1}) + \langle \nabla f(\mathbf{x}^{k+1}), \bar{\mathbf{p}} - \mathbf{x}^{k+1} \rangle + \frac{1}{2} \left\langle \mathbf{B}_{k+1}(\bar{\mathbf{p}} - \mathbf{x}^{k+1}), (\bar{\mathbf{p}} - \mathbf{x}^{k+1})) \right\rangle .$$

We require the gradient of m_{k+1} to match the gradient of f at $\mathbf{x}^k$ and $\mathbf{x}^{k+1}$.

- $\nabla m_{k+1}(\mathbf{x}^{k+1}) = \nabla f(\mathbf{x}^{k+1})$ as desired;
- For $\mathbf{x}^k$, we have

  $$\nabla m_{k+1}(\mathbf{x}^k) = \nabla f(\mathbf{x}^{k+1}) + \mathbf{B}_{k+1}(\mathbf{x}^k - \mathbf{x}^{k+1})$$

  which must be equal to $\nabla f(\mathbf{x}^k)$.
- Rearranging, we have that $\mathbf{B}_{k+1}$ must satisfy the **secant equation**

  $$\mathbf{B}_{k+1}\mathbf{s}^k = \mathbf{y}^k$$

  where $\mathbf{s}^k = \mathbf{x}^{k+1} - \mathbf{x}^k$ and $\mathbf{y}^k = \nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^k)$.
- The secant equation can be satisfied only if $\langle \mathbf{s}^k, \mathbf{y}^k \rangle > 0$, which is guaranteed to hold if the step-size α_k satisfies the Wolfe conditions.

*Quasi-Newton methods

How do we update $\mathbf{B}_k$?

- There might be an infinite number of matrices satisfying the secant equation.
- To determine $\mathbf{B}_{k+1}$ uniquely, we find the symmetric matrix that is closest to $\mathbf{B}_k$:

$$\mathbf{B}_{k+1} = \arg\min_{\mathbf{B}} \|\mathbf{B} - \mathbf{B}_k\|_{\mathbf{W}} \quad \text{subject to } \mathbf{B} = \mathbf{B}^T \text{ and } \mathbf{B}\mathbf{s}^k = \mathbf{y}^k \qquad (21)$$

- The choice $\|\mathbf{A}\|_{\mathbf{W}} = \|\mathbf{W}^{\frac{1}{2}}\mathbf{A}\mathbf{W}^{\frac{1}{2}}\|_{\mathrm{F}}$ where $\mathbf{W}$ can be *any* matrix satisfying $\mathbf{W}\mathbf{y}^k = \mathbf{s}^k$, leads to an easy solution of (21) and a *scale-invariant* method.

DFP method (from Davidon, Fletcher & Powell)

- The DFP method arises from $\mathbf{W} = \overline{\mathbf{G}}^{-1}$, where $\overline{\mathbf{G}}$ is the average Hessian

$$\overline{\mathbf{G}} = \int_0^1 \nabla^2 f(\mathbf{x}^k + \tau\alpha_k\mathbf{p}^k)d\tau.$$

- It yields the following update for $\mathbf{B}_k$

$$\mathbf{B}_{k+1} = \mathbf{V}_k^T\mathbf{B}_k\mathbf{V}_k + \eta_k\mathbf{y}^k(\mathbf{y}^k)^T \ ,$$

where $\eta_k = \frac{1}{\langle \mathbf{y}^k, \mathbf{s}^k \rangle}$ and $\mathbf{V}_k = \mathbf{I} - \eta_k\mathbf{y}^k(\mathbf{s}^k)^T$.

*Quasi-Newton methods

BFGS method [5] (from Broyden, Fletcher, Goldfarb & Shanno)

The BFGS method arises from directly updating $\mathbf{H}_k = \mathbf{B}_k^{-1}$. It interchanges the roles of $\mathbf{s}^k$ and $\mathbf{y}^k$ in the DFP method. The update on the inverse $\mathbf{B}$ is found by solving

$$\min_{\mathbf{H}} \|\mathbf{H} - \mathbf{H}_k\|_{\mathbf{W}} \quad \text{subject to } \mathbf{H} = \mathbf{H}^T \text{ and } \mathbf{H}\mathbf{y}^k = \mathbf{s}^k \tag{22}$$

Also in this case, $\mathbf{W} = \overline{\mathbf{G}}^{-1}$. The solution is a rank-2 update of the matrix $\mathbf{H}_k$:

$$\mathbf{H}_{k+1} = \mathbf{V}_k^T \mathbf{H}_k \mathbf{V}_k + \eta_k \mathbf{s}^k (\mathbf{s}^k)^T ,$$

where $\mathbf{V}_k = \mathbf{I} - \eta_k \mathbf{s}^k (\mathbf{y}^k)^T$.

- Initialization of $\mathbf{H}_0$ is an art. We can choose to set it to be an approximation of $\nabla^2 f(\mathbf{x}^0)$ obtained by finite differences or just a multiple of the identity matrix.

*Quasi-Newton methods

BFGS method [5] (from Broyden, Fletcher, Goldfarb & Shanno)

The BFGS method arises from directly updating $\mathbf{H}_k = \mathbf{B}_k^{-1}$. It interchanges the roles of $\mathbf{s}^k$ and $\mathbf{y}^k$ in the DFP method. The update on the inverse $\mathbf{B}$ is found by solving

$$\min_{\mathbf{H}} \|\mathbf{H} - \mathbf{H}_k\|_{\mathbf{W}} \quad \text{subject to } \mathbf{H} = \mathbf{H}^T \text{ and } \mathbf{H}\mathbf{y}^k = \mathbf{s}^k \tag{22}$$

Also in this case, $\mathbf{W} = \overline{\mathbf{G}}^{-1}$. The solution is a rank-2 update of the matrix $\mathbf{H}_k$:

$$\mathbf{H}_{k+1} = \mathbf{V}_k^T \mathbf{H}_k \mathbf{V}_k + \eta_k \mathbf{s}^k (\mathbf{s}^k)^T \ ,$$

where $\mathbf{V}_k = \mathbf{I} - \eta_k \mathbf{s}^k (\mathbf{y}^k)^T$.

Theorem (Convergence of BFGS)

Let $f \in \mathcal{C}^2$. Assume that the BFGS sequence $\{\mathbf{x}^k\}$ converges to a point $\mathbf{x}^\star$ and $\sum_{k=1}^{\infty} \|\mathbf{x}^k - \mathbf{x}^\star\| \leq \infty$. Assume also that $\nabla^2 f(\mathbf{x})$ is Lipschitz continuous at $\mathbf{x}^\star$. Then $\mathbf{x}^k$ converges to $\mathbf{x}^\star$ at a **superlinear** *rate.*

Remarks

The proof shows that given the assumptions, the BFGS updates for $\mathbf{B}_k$ satisfy the Dennis & Moré condition, which in turn imply superlinear convergence.

*Quasi-Newton methods

BFGS method [5] (from Broyden, Fletcher, Goldfarb & Shanno)

The BFGS method arises from directly updating $\mathbf{H}_k = \mathbf{B}_k^{-1}$. It interchanges the roles of $\mathbf{s}^k$ and $\mathbf{y}^k$ in the DFP method. The update on the inverse $\mathbf{B}$ is found by solving

$$\min_{\mathbf{H}} \|\mathbf{H} - \mathbf{H}_k\|_{\mathbf{W}} \quad \text{subject to } \mathbf{H} = \mathbf{H}^T \text{ and } \mathbf{H}\mathbf{y}^k = \mathbf{s}^k \tag{22}$$

Also in this case, $\mathbf{W} = \overline{\mathbf{G}}^{-1}$. The solution is a rank-2 update of the matrix $\mathbf{H}_k$:

$$\mathbf{H}_{k+1} = \mathbf{V}_k^T \mathbf{H}_k \mathbf{V}_k + \eta_k \mathbf{s}^k (\mathbf{s}^k)^T \ ,$$

where $\mathbf{V}_k = \mathbf{I} - \eta_k \mathbf{s}^k (\mathbf{y}^k)^T$.

SR1 (Symmetric-rank-1)

The SR method performs rank-1 updates of the matrix $\mathbf{H}_k$:

$$\mathbf{H}_{k+1} = \mathbf{H}_k + \frac{(\mathbf{s}^k - \mathbf{H}_k\mathbf{y}^k)(\mathbf{s}^k - \mathbf{H}_k\mathbf{y}^k)^T}{\langle \mathbf{s}^k - \mathbf{H}_k\mathbf{y}^k, \mathbf{y}^k \rangle}.$$

$\mathbf{H}_{k+1}$ is not guaranteed to be positive definite, but SR1 performs very well in practice.

L-BFGS

Challenges for BFGS

- BFGS approach stores and applies a dense $p \times p$ matrix $\mathbf{H}_k$.
- When p is very large, $\mathbf{H}_k$ can prohibitively expensive to store and apply.

L(imited memory)-BFGS

- Do not of store $\mathbf{H}_k$, but keep only the m most recent pairs $\{(\mathbf{s}^i, \mathbf{y}^i)\}$.
- Compute $\mathbf{H}_k \nabla f(\mathbf{x}_k)$ by performing a sequence of operations with $\mathbf{s}^i$ and $\mathbf{y}^i$:
    - Choose a temporary initial approximation $\mathbf{H}_k^0$.
    - Recursively apply $\mathbf{H}_{k+1} = \mathbf{V}_k^T \mathbf{H}_k \mathbf{V}_k + \eta_k \mathbf{s}^k (\mathbf{s}^k)^T$, m times starting from $\mathbf{H}_k^0$:

$$\begin{aligned}\mathbf{H}_k = & \left(\mathbf{V}_{k-1}^T \cdots \mathbf{V}_{k-m}^T\right) \mathbf{H}_k^0 \left(\mathbf{V}_{k-m} \cdots \mathbf{V}_{k-1}\right) \\ & + \eta_{k-m} \left(\mathbf{V}_{k-1}^T \cdots \mathbf{V}_{k-m+1}^T\right) \mathbf{s}^{k-m} (\mathbf{s}^{k-m})^T \left(\mathbf{V}_{k-m+1} \cdots \mathbf{V}_{k-1}\right) \\ & + \cdots \\ & + \eta_{k-1} \mathbf{s}^{k-1} (\mathbf{s}^{k-1})^T\end{aligned}$$

    - From the previous expression, we can compute $\mathbf{H}_k \nabla f(\mathbf{x}^k)$ recursively.
- Replace the oldest element in $\{\mathbf{s}^i, \mathbf{y}^i\}$ with $(\mathbf{s}^k, \mathbf{y}^k)$.
- From practical experience, $m \in (3, 50)$ does the trick.

L-BFGS: A quasi-Newton method

Procedure for computing $\mathbf{H}_k \nabla f(\mathbf{x}^k)$

```
0. Recall $\eta_k = 1/\langle \mathbf{y}^k, \mathbf{s}^k \rangle$.
1. $\mathbf{q} = \nabla f(\mathbf{x}^k)$.
2. For $i = k-1, \ldots, k-m$
       $\alpha_i = \eta_i \langle \mathbf{s}^i, \mathbf{q} \rangle$
       $\mathbf{q} = \mathbf{q} - \alpha_i \mathbf{y}^i$.
3. $\mathbf{r} = \mathbf{H}_k^0 \mathbf{q}$.
4. For $i = k-m, \ldots, k-1$
       $\beta = \eta_i \langle \mathbf{y}^i, \mathbf{r} \rangle$
       $\mathbf{r} = \mathbf{r} + (\alpha_i - \beta)\mathbf{s}^i$.
5. $\mathbf{H}_k \nabla f(\mathbf{x}^k) = \mathbf{r}$.
```

Remarks

- Apart from the step $\mathbf{r} = \mathbf{H}_k^0 \mathbf{q}$, the algorithm requires only $4mp$ multiplications.
- If $\mathbf{H}_k^0$ is chosen to be diagonal, another p multiplications are needed.
- An effective initial choice is $\mathbf{H}_k^0 = \gamma_k \mathbf{I}$, where

$$\gamma_k = \frac{\langle \mathbf{s}^{k-1}, \mathbf{y}^{k-1} \rangle}{\langle \mathbf{y}^{k-1}, \mathbf{y}^{k-1} \rangle}$$

L-BFGS: A quasi-Newton method for big data

L-BFGS

```
1. Choose starting point $\mathbf{x}^0$ and $m > 0$.
2. For $k = 0, 1, \ldots$
   2.a Choose $\mathbf{H}_k^0$.
   2.b Compute $\mathbf{p}^k = -\mathbf{H}_k \nabla f(\mathbf{x}^k)$ using the previous algorithm.
   2.c Set $\mathbf{x}^{k+1} = \mathbf{x}^k + \alpha_k \mathbf{p}^k$, where $\alpha_k$ satisfies the Wolfe conditions.
       if $k > m$, discard the pair $\{\mathbf{s}^{k-m}, \mathbf{p}^{k-m}\}$ from storage.
   2.d Compute and store $\mathbf{s}^k = \mathbf{x}^{k+1} - \mathbf{x}^k$, $\mathbf{y}^k = \nabla f(\mathbf{x}^{k+1}) - \nabla f(\mathbf{x}^k)$.
```

Warning

L-BFGS updates does not guarantee positive semidefiniteness of the variable metric $\mathbf{H}_k$ in contrast to BFGS.

Newton method

Local quadratic approximation using the Hessian

- Obtain a local quadratic approximation using the second-order Taylor series approximation to $f(\mathbf{x}^k + \mathbf{p})$:

$$f(\mathbf{x}^k + \mathbf{p}) \approx f(\mathbf{x}^k) + \langle \mathbf{p}, \nabla f(\mathbf{x}^k) \rangle + \frac{1}{2}\langle \mathbf{p}, \nabla^2 f(\mathbf{x}^k)\mathbf{p} \rangle := m_k(\mathbf{p})$$

- Assuming that the Hessian $\nabla^2 f_k$ is **positive definite**, the Newton direction is the vector $\mathbf{p}^k$ that minimizes $m_k(\mathbf{p})$:

$$\mathbf{p}^k = -\left(\nabla^2 f(\mathbf{x}^k)\right)^{-1} \nabla f(\mathbf{x}^k) \ .$$

- A unit step-size $\alpha_k = 1$ can be chosen near convergence. Then, the iterates become

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\nabla^2 f(\mathbf{x}^k)\right)^{-1} \nabla f(\mathbf{x}^k) \ .$$

Remarks

- For $f \in \mathcal{F}_L^{2,1}$ but $f \notin \mathcal{F}_{L,\mu}^{2,1}$, the Hessian may not always be positive definite.

(Local) Convergence of Newton method

Lemma

Assume f is a twice differentiable convex function with minimum at $\mathbf{x}^\star$ such that:

- $\nabla^2 f(\mathbf{x}^\star) \succeq \mu\mathbf{I}$ *for some* $\mu > 0$,
- $\|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{y})\|_{2\to 2} \leq M\|\mathbf{x} - \mathbf{y}\|_2$ *for some constant* $M > 0$ *and all* $\mathbf{x}, \mathbf{y} \in$ *dom*(f).

Moreover, assume the starting point $\mathbf{x}^0 \in$ *dom*(f) *is such that* $\|\mathbf{x}^0 - \mathbf{x}^\star\|_2 < \frac{2\mu}{3M}$. *Then, the Newton method iterates converge* **quadratically***:*

$$\|\mathbf{x}^{k+1} - \mathbf{x}^\star\| \leq \frac{M\|\mathbf{x}^k - \mathbf{x}^\star\|_2^2}{2\left(\mu - M\|\mathbf{x}^k - \mathbf{x}^\star\|_2\right)}.$$

Remark

This is the fastest convergence rate we have seen so far, but it requires to solve a $p \times p$ linear system at each iteration, $\nabla^2 f(\mathbf{x}^k)\mathbf{p}^k = -\nabla f(\mathbf{x}^k)$!

Newton's method local quadratic convergence - Proof I/II [5]

Since $\nabla f(\mathbf{x}^\star) = 0$ we have

$$\begin{aligned}\mathbf{x}^{k+1} - \mathbf{x}^\star &= \mathbf{x}^k - \mathbf{x}^\star - (\nabla^2 f(\mathbf{x}^k))^{-1} \nabla f(\mathbf{x}^k) \\ &= (\nabla^2 f(\mathbf{x}^k))^{-1} \left(\nabla^2 f(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^\star) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^\star))\right)\end{aligned}$$

By Taylor's theorem, we also have

$$\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^\star) = \int_0^1 \nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^\star - \mathbf{x}^k))(\mathbf{x}^k - \mathbf{x}^\star)\, dt$$

Combining the two above, we obtain

$$\begin{aligned}&\|\nabla^2 f(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^\star) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^\star))\| \\ &= \left\| \int_0^1 \left(\nabla^2 f(\mathbf{x}^k) - \nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^\star - \mathbf{x}^k))\right)(\mathbf{x}^k - \mathbf{x}^\star)\, dt \right\| \\ &\leq \int_0^1 \left\|\nabla^2 f(\mathbf{x}^k) - \nabla^2 f(\mathbf{x}^k + t(\mathbf{x}^\star - \mathbf{x}^k))\right\| \|\mathbf{x}^k - \mathbf{x}^\star\|\, dt \\ &\leq M\|\mathbf{x}^k - \mathbf{x}^\star\|^2 \int_0^1 t dt = \frac{1}{2} M \|\mathbf{x}^k - \mathbf{x}^\star\|^2\end{aligned}$$

Newton's method local quadratic convergence - Proof II/II [5].

- Recall

$$\mathbf{x}^{k+1} - \mathbf{x}^\star = (\nabla^2 f(\mathbf{x}^k))^{-1} \left(\nabla^2 f(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^\star) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^\star))\right)$$

$$\|\nabla^2 f(\mathbf{x}^k)(\mathbf{x}^k - \mathbf{x}^\star) - (\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^\star))\| \leq \frac{1}{2} M \|\mathbf{x}^k - \mathbf{x}^\star\|^2$$

- Since $\nabla^2 f(\mathbf{x}^\star)$ is nonsingular, there must exist a radius r such that $\|(\nabla^2 f(\mathbf{x}^k))^{-1}\| \leq 2\|(\nabla^2 f(\mathbf{x}^\star))^{-1}\|$ for all $\mathbf{x}^k$ with $\|\mathbf{x}^k - \mathbf{x}^*\| \leq r$.
- Substituting, we obtain

$$\|\mathbf{x}^{k+1} - \mathbf{x}^\star\| \leq M \|(\nabla^2 f(\mathbf{x}^\star))^{-1}\| \|\mathbf{x}^k - \mathbf{x}^\star\|^2 = \widetilde{M} \|\mathbf{x}^k - \mathbf{x}^\star\|^2,$$

  where $\widetilde{M} = M\|(\nabla^2 f(\mathbf{x}^\star))^{-1}\|$.
- If we choose $\|\mathbf{x}^0 - \mathbf{x}^\star\| \leq \min(r, 1/(2\widetilde{M}))$, we obtain by induction that the iterates $\mathbf{x}^k$ converge quadratically to $\mathbf{x}^\star$.

□

Example: Logistic regression

Problem (Logistic regression)

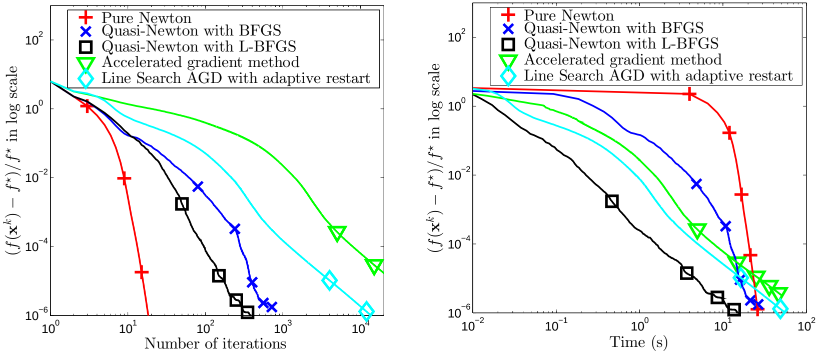
Given $\mathbf{A} \in \{0,1\}^{n \times p}$ *and* $\mathbf{b} \in \{-1,+1\}^n$*, solve:*

$$f^\star := \min_{\mathbf{x},\beta} \left\{ f(\mathbf{x}) := \frac{1}{n} \sum_{j=1}^{n} \log\left(1 + \exp\left(-\mathbf{b}_j(\mathbf{a}_j^T \mathbf{x} + \beta)\right)\right) \right\}.$$

Real data

- Real data: `w5a` with $n = 9888$ data points, $p = 300$ features
- Available at `http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/binary.html`.

Example: Logistic regression - numerical results



Parameters

- For BFGS, L-BFGS and Newton's method: maximum number of iterations 200, tolerance 10^{-6}. L-BFGS memory $m = 50$.
- For accelerated gradient method: maximum number of iterations 20000, tolerance 10^{-6}.
- Ground truth: Get a high accuracy approximation of $\mathbf{x}^\star$ and $f^\star$ by applying Newton's method for 200 iterations.

Affine invariance of Newton method

Lemma

The convergence characterization above changes when we apply an affine transform to the space. However, ***each Newton step is affine invariant****. That is, the Newton's step in the transformed space results into estimates that easily lead to corresponding estimates of the original space through the inverse affine transformation.*

Proof.

Let $\mathbf{T} \in \mathbb{R}^{p \times p}$ be a *nonsingular* affine transformation of $\mathbb{R}^{p \times p}$. We define $\bar{f}(\mathbf{y}) = f(\mathbf{T}\mathbf{y})$ where $\mathbf{x} = \mathbf{T}\mathbf{y}$. We compute the following quantities:

$$\nabla \bar{f}(\mathbf{y}) = \mathbf{T}^T \nabla f(\mathbf{x}) \quad \text{and} \quad \nabla^2 \bar{f}(\mathbf{y}) = \mathbf{T}^T \nabla^2 f(\mathbf{x}) \mathbf{T}.$$

Then, in the iterates $\mathbf{y}^{k+1} = \mathbf{y}^k - \left(\nabla^2 \bar{f}(\mathbf{y})\right)^{-1} \nabla \bar{f}(\mathbf{y})$, we observe:

$$-\left(\nabla^2 \bar{f}(\mathbf{y})\right)^{-1} \nabla \bar{f}(\mathbf{y}) = -\mathbf{T}^{-1} \left(\nabla^2 f(\mathbf{x})\right)^{-1} \nabla f(\mathbf{x}),$$

and thus,

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \left(\nabla^2 f(\mathbf{x})\right)^{-1} \nabla f(\mathbf{x}) = \mathbf{T}\left(\mathbf{y}^k - \left(\nabla^2 \bar{f}(\mathbf{y})\right)^{-1} \nabla \bar{f}(\mathbf{y})\right) = \mathbf{T}\mathbf{y}^{k+1}$$

□

Affine invariance in optimization

On the positive side...

- We have shown that Newton method is *affine invariant* in **practice**: it is insensitive to the choice of the coordinate system.
- Moreover, to apply Newton in practice, we often do not require the knowledge of global constants such as strongly convexity parameter μ or Lipschitz constants M and L.

Affine invariance in optimization

On the positive side...

- We have shown that Newton method is *affine invariant* in **practice**: it is insensitive to the choice of the coordinate system.
- Moreover, to apply Newton in practice, we often do not require the knowledge of global constants such as strongly convexity parameter μ or Lipschitz constants M and L.

On the negative side...

- The analysis of classic Newton method includes global constants μ, M and/or L, which are usually **unknown a priori and/or are very hard to compute...**
- As a by-product, we might not know in reality the actual number of Newton steps required for a given accuracy.
- While Newton method is affine invariant in practice, the above analysis is *not*!

Self-concordant minimization

Self-concordant minimization (SCM) problem

$$F^\star := \min_{\mathbf{x} \in \mathrm{dom}(f)} f(\mathbf{x})$$

- $f \in \mathcal{F}_2(\mathrm{dom}(f))$ - self-concordant on $\mathrm{dom}(f) := \{\mathbf{x} \in \mathbb{R}^p \ : \ f(\mathbf{x}) < +\infty\}$

Self-concordant minimization

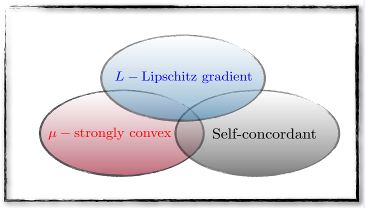
Self-concordant minimization (SCM) problem

$$F^{\star} := \min_{\mathbf{x} \in \text{dom}(f)} f(\mathbf{x})$$

- $f \in \mathcal{F}_2(\text{dom}(f))$ - self-concordant on $\text{dom}(f) := \{\mathbf{x} \in \mathbb{R}^p \ : \ f(\mathbf{x}) < +\infty\}$

"I'm not convinced... Why to use self-concordance in optimization?"

- A self-concordant function might not not necessarily be strongly convex or have a continuous Lipschitz gradient.



Fundamental self-concordant inequality

Fundamental inequality

$$\frac{\|\mathbf{y}-\mathbf{x}\|_{\mathbf{x}}^2}{1+\|\mathbf{y}-\mathbf{x}\|_{\mathbf{x}}} \leq \langle \nabla f(\mathbf{y}) - \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x} \rangle \leq \frac{\|\mathbf{y}-\mathbf{x}\|_{\mathbf{x}}^2}{1-\|\mathbf{y}-\mathbf{x}\|_{\mathbf{x}}}$$

- The left-hand side inequality holds for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom}(f)$.
- The right-hand side inequality holds for $\|\mathbf{y}-\mathbf{x}\|_{\mathbf{x}} < 1$.

Recall: Local norm $\|\mathbf{b}\|_{\mathbf{x}} := \sqrt{\langle \mathbf{b}, \nabla^2 f(\mathbf{x})\mathbf{b} \rangle}$.

Previously seen...

$$\mu\|\mathbf{x}-\mathbf{y}\|_2^2 \leq \langle \nabla f(\mathbf{y}) - \nabla f(\mathbf{x}), \mathbf{y}-\mathbf{x} \rangle \leq L\|\mathbf{x}-\mathbf{y}\|_2^2$$

- The left-hand side inequality holds for $f \in \mathcal{F}_{\mu}^{2,1}$ and for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom}(f)$.
- The right-hand side inequality holds for $f \in \mathcal{F}_{L}^{2,1}$ and for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom}(f)$.
- Both inequalities hold for $f \in \mathcal{F}_{L,\mu}^{2,1}$ and for all $\mathbf{x}, \mathbf{y} \in \mathrm{dom}(f)$.

Newton method for SCM

Damped Newton algorithm

1. Choose $\mathbf{x}^0 \in \mathrm{dom}(f)$ as a starting point.
2. For $k = 0, 1, \cdots$, perform:

$$\begin{cases} \mathbf{d}^k & = -\left(\nabla^2 f(\mathbf{x}^k)\right)^{-1} \nabla f(\mathbf{x}^k) & \text{(Newton direction)} \\ \lambda_k & = \|\mathbf{d}^k\|_{\mathbf{x}^k} & \text{(Newton decrement)} \\ \alpha_k & = (1 + \lambda_k)^{-1} & \text{(step-size)} \\ \mathbf{x}^{k+1} & = \mathbf{x}^k + \alpha_k \mathbf{d}^k & \end{cases}$$

Newton method for SCM

Damped Newton algorithm

1. Choose $\mathbf{x}^0 \in \mathrm{dom}(f)$ as a starting point.
2. For $k = 0, 1, \cdots,$ perform:

$$\begin{cases} \mathbf{d}^k & = -\left(\nabla^2 f(\mathbf{x}^k)\right)^{-1} \nabla f(\mathbf{x}^k) & \text{(Newton direction)} \\ \lambda_k & = \|\mathbf{d}^k\|_{\mathbf{x}^k} & \text{(Newton decrement)} \\ \alpha_k & = (1+\lambda_k)^{-1} & \text{(step-size)} \\ \mathbf{x}^{k+1} & = \mathbf{x}^k + \alpha_k \mathbf{d}^k & \end{cases}$$

Complexity per iteration

- Evaluation of $\nabla^2 f(\mathbf{x}^k)$ and $\nabla f(\mathbf{x}^k)$ (closed form expressions).
- Computing the Newton direction requires solving the linear system $\nabla^2 f(\mathbf{x}^k)\mathbf{d}^k = -\nabla f(\mathbf{x}^k)$.
- Computing the Newton decrement λ_k requires $\langle \mathbf{d}^k, \nabla^2 f(\mathbf{x})\mathbf{d}^k \rangle$.

Global convergence

Lemma (Descent lemma)

Let $\{\mathbf{x}^k\}_{k\geq 0}$ *be the sequence generated by Damped Newton algorithm. Then*

$$\boxed{f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) - \omega(\lambda_k)}$$

where $\omega(\tau) := \tau - \ln(1+\tau) > 0$ *for* $\tau > 0$*.*

Global convergence

Lemma (Descent lemma)

Let $\{\mathbf{x}^k\}_{k\geq 0}$ *be the sequence generated by Damped Newton algorithm. Then*

$$\boxed{f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) - \omega(\lambda_k)}$$

where $\omega(\tau) := \tau - \ln(1+\tau) > 0$ *for* $\tau > 0$.

We observe that:

- $[f(\mathbf{x}^{k+1}) - f^\star] \leq [f(\mathbf{x}^k) - f^\star] - \omega(\lambda_k)$ for all $k \geq 0$.
- $[f(\mathbf{x}^k) - f^\star] \leq [f(\mathbf{x}^0) - f^\star] - \sum_{j=0}^{k-1} \omega(\lambda_j)$.
- If $\lambda_k \geq \lambda > 0$ for $k = 0, \ldots, K$, then

  $$[f(\mathbf{x}^K) - f^\star] \leq [f(\mathbf{x}^0) - f^\star] - K\omega(\lambda).$$

  The **number of iterations** to reach $f(\mathbf{x}^K) - f^\star \leq \varepsilon$ is

  $$K := \left\lfloor \frac{[f(\mathbf{x}^0) - f^\star] - \varepsilon}{\omega(\lambda)} \right\rfloor + 1.$$

- Global convergence rate is just sublinear, i.e., $O(1/k)$.

Proof of descent lemma

Sketch of proof.

- Let $\mathbf{s}^k := \mathbf{x}^k + \mathbf{d}^k$. We have $\mathbf{x}^{k+1} - \mathbf{x}^k = \alpha_k \mathbf{d}^k$ and $\mathbf{x}^{k+1} = (1-\alpha_k)\mathbf{x}^k + \alpha_k \mathbf{s}^k$.
- By self-concordance of f (upper bound inequality):

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) + \nabla f(\mathbf{x}^k)(\mathbf{x}^{k+1} - \mathbf{x}^k) + \omega_*(\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{x}^k}),$$

  under condition $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|_{\mathbf{x}^k} < 1$, where $\omega_*(\tau) = -\tau - \ln(1-\tau)$.
- Substituting $\mathbf{x}^{k+1} - \mathbf{x}^k = \alpha_k \mathbf{d}^k$, we get

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) + \alpha_k \nabla f(\mathbf{x}^k)\mathbf{d}^k + \omega_*(\alpha_k \|\mathbf{d}^k\|_{\mathbf{x}^k}).$$

- Substituting $\nabla f(\mathbf{x}^k) = -\nabla^2 f(\mathbf{x}^k)\mathbf{d}^k$ and using $\lambda_k := \|\mathbf{d}^k\|_{\mathbf{x}^k}$, we get

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) - \alpha_k \lambda_k^2 + \omega_*(\alpha_k \lambda_k).$$

- Let $\psi(\alpha) := \alpha\lambda_k^2 - \omega_*(\alpha\lambda_k) = \alpha\lambda_k^2 + \alpha\lambda_k + \ln(1-\alpha\lambda_k)$. This function attains the maximum at $\alpha_k = (1+\lambda_k)^{-1}$ and $\psi(\alpha_k) = \lambda_k - \ln(1+\lambda_k)$. Hence, we have

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) - \omega(\lambda_k).$$

□

Local convergence

Newton algorithm

1. Choose $\mathbf{x}^0 \in \mathrm{dom}(f)$ as a starting point.
2. For $k = 0, 1, \cdots,$ perform:

$$\begin{cases} \mathbf{d}^k & = -\left(\nabla^2 f(\mathbf{x}^k)\right)^{-1} \nabla f(\mathbf{x}^k) \quad \text{(Newton direction)} \\ \mathbf{x}^{k+1} & = \mathbf{x}^k + \mathbf{d}^k \quad \text{(Unit step-size)} \end{cases}$$

Theorem (Local quadratic convergence)

*Let $\lambda_k = \|\mathbf{d}^k\|_{\mathbf{x}^k}$ and $\{\mathbf{x}^k\}$ be the sequence generated by **Newton algorithm**. If $\lambda_0 < 0.3819 := \bar{\lambda}$ then*

$$\boxed{\lambda_{k+1} \leq \left(\frac{\lambda_k}{1-\lambda_k}\right)^2 < \lambda_k}$$

Consequently, $\{\mathbf{x}^k\}_{k\geq 0}$ converges to $\mathbf{x}^\star$ at a quadratic rate.

Quadratic convergence region

Let $\sigma := \omega_*'(\bar{\lambda}) = 0.6180.$ Then the **quadratic convergence region** $\mathcal{Q}_\sigma$ is defined as:

$$\boxed{\mathcal{Q}_\sigma := \{\mathbf{x} \in \mathrm{dom}(f) \ : \ \|\mathbf{x} - \mathbf{x}^\star\|_{\mathbf{x}^\star} \leq \sigma\}.}$$

For any $\mathbf{x}^0 \in \mathcal{Q}_\sigma$, $\{\mathbf{x}^k\}$ converges to $\mathbf{x}^\star$ at a quadratic rate.

A two-step approach for SCM

- Recall: $\lambda_k = \|\mathbf{d}^k\|_{\mathbf{x}^k} := \left\| \left(\nabla^2 f(\mathbf{x}^k)\right)^{-1} \nabla f(\mathbf{x}^k) \right\|_{\mathbf{x}^k}$.
- Let $\bar{\lambda} = 0.3819$, which is the solution of $\lambda/(1-\lambda)^2 = 1$.
- Choose a constant $\hat{\lambda} \in (0, \bar{\lambda})$ and a starting point $\mathbf{x}^0$.

First stage: $\lambda_k \geq \hat{\lambda}$

- We apply the Damped Newton method, with guarantee:

$$f(\mathbf{x}^{k+1}) \leq f(\mathbf{x}^k) - \omega(\hat{\lambda})$$

- The number of iterations at this stage is bounded:

$$K \leq \frac{1}{\omega(\hat{\lambda})} \left(f(\mathbf{x}^0) - f(\mathbf{x}^\star) \right)$$

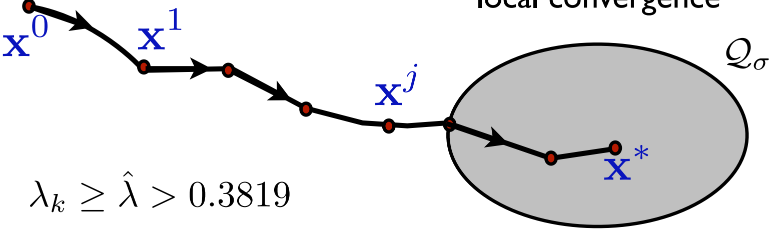
Second stage: $\lambda_k \leq \hat{\lambda}$

- We apply the standard Newton method, with quadratic convergence:

$$\lambda_{k+1} \leq \left(\frac{\lambda_k}{1-\lambda_k} \right)^2 \leq \frac{\hat{\lambda}\lambda_k}{(1-\hat{\lambda})^2} < \lambda_k$$

Overall analytical worst-case complexity

local convergence



$\lambda_k \geq \hat{\lambda} > 0.3819$

global convergence:
sublinear

$\lambda_k < \hat{\lambda} \qquad \sigma = \omega_*'(\hat{\lambda})$

quadratic convergence
region

*From gradient descent to mirror descent

Gradient descent as a majorization-minimization scheme

- **Majorize** f at $\mathbf{x}^k$ by using L-Lipschitz gradient continuity

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{L}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2 := Q(\mathbf{x}, \mathbf{x}^k)$$

- **Minimize** $Q(\mathbf{x}, \mathbf{x}^k)$ to obtain the next iterate $\mathbf{x}^{k+1}$

$$\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}} Q(\mathbf{x}, \mathbf{x}^k) \Rightarrow \nabla f(\mathbf{x}^k) + L(\mathbf{x}^{k+1} - \mathbf{x}^k) = 0$$

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \frac{1}{L}\nabla f(\mathbf{x}^k)$$

Other majorizers

We can re-write the majorization step as

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \alpha d(\mathbf{x}, \mathbf{x}^k)$$

where $d(\mathbf{x}, \mathbf{x}^k) = \frac{1}{2}\|\mathbf{x} - \mathbf{x}^k\|_2^2$ is the Euclidean distance and $\alpha = L$.

- Can we use a different function $d(\mathbf{x}, \mathbf{x}^k)$ that is better suited to minimizing f?

*Bregman divergences

Definition (Bregman divergence)

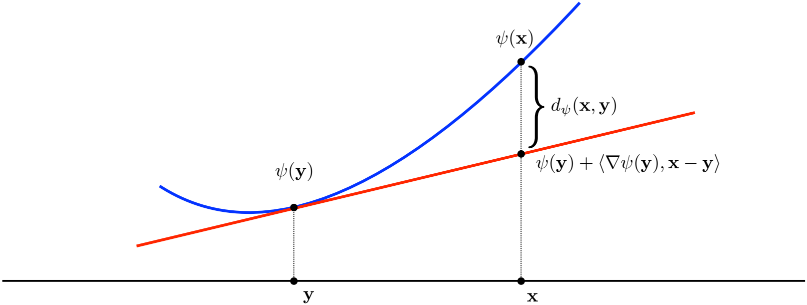
Let $\psi : \mathcal{S} \to \mathbb{R}$ be a continuously-differentiable and strictly convex function defined on a closed convex set $\mathcal{S}$. The **Bregman divergence** (d_ψ) associated with ψ for points $\mathbf{x}$ and $\mathbf{y}$ is:

$$d_\psi(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{x}) - \psi(\mathbf{y}) - \langle \nabla\psi(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle$$

- $\psi(\cdot)$ is referred to as the Bregman or proximity function.
- The Bregman divergence satisfies the following properties:
  - (a) $d_\psi(\mathbf{x}, \mathbf{y}) \geq 0$ for all $\mathbf{x}$ and $\mathbf{y}$ with equality if and only if $\mathbf{x} = \mathbf{y}$
  - (b) Define $q(\mathbf{x}) := d_\psi(\mathbf{x}, \mathbf{y})$ for a fixed $\mathbf{y}$, then $\nabla q(\mathbf{x}) = \nabla\psi(\mathbf{x}) - \nabla\psi(\mathbf{y})$
  - (c) For all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{S}$, $d_\psi(\mathbf{x}, \mathbf{y}) = d_\psi(\mathbf{x}, \mathbf{z}) + d_\psi(\mathbf{z}, \mathbf{y}) + \langle (\mathbf{x} - \mathbf{z}), \nabla\psi(\mathbf{y}) - \nabla\psi(\mathbf{z}) \rangle$
  - (d) For all $\mathbf{x}, \mathbf{y} \in \mathcal{S}$, $d_\psi(\mathbf{x}, \mathbf{y}) + d_\psi(\mathbf{y}, \mathbf{x}) = \langle (\mathbf{x} - \mathbf{y}), \nabla\psi(\mathbf{x}) - \nabla\psi(\mathbf{y}) \rangle$
- The Bregman divergence becomes a Bregman distance when it is *symmetric* (i.e. $d_\psi(\mathbf{x}, \mathbf{y}) = d_\psi(\mathbf{y}, \mathbf{x})$) and satisfies the *triangle inequality*.
- "*All Bregman distances are Bregman divergences but the reverse is not true!*"

*Bregman divergences

- The Bregman divergence is the vertical distance at $\mathbf{x}$ between ψ and the tangent of ψ at $\mathbf{y}$, see figure below


$\psi(\mathbf{x})$
$d_\psi(\mathbf{x},\mathbf{y})$
$\psi(\mathbf{y}) + \langle \nabla\psi(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle$
$\psi(\mathbf{y})$
$\mathbf{y}$
$\mathbf{x}$

- The Bregman divergence measures the strictness of convexity of $\psi(\cdot)$.

*Bregman divergences

Table: **Bregman functions** $\psi(\mathbf{x})$ & corresponding Bregman divergences/distances $d_\psi(\mathbf{x},\mathbf{y})^a$.

Name (or Loss)	**Domain**^{b}	$\psi(\mathbf{x})$	$d_\psi(\mathbf{x},\mathbf{y})$
Squared loss	$\mathbb{R}$	x^2	$(x-y)^2$
Itakura-Saito divergence	$\mathbb{R}_{++}$	$-\log x$	$\frac{x}{y} - \log\left(\frac{x}{y}\right) - 1$
Squared Euclidean distance	$\mathbb{R}^p$	$\lVert\mathbf{x}\rVert_2^2$	$\lVert\mathbf{x}-\mathbf{y}\rVert_2^2$
Squared Mahalanobis distance	$\mathbb{R}^p$	$\langle\mathbf{x},\mathbf{A}\mathbf{x}\rangle$	$\langle(\mathbf{x}-\mathbf{y}),\mathbf{A}(\mathbf{x}-\mathbf{y})\rangle^c$
Entropy distance	p-simplex^d	$\sum_i x_i \log x_i$	$\sum_i x_i \log\left(\frac{x_i}{y_i}\right)$
Generalized I-divergence	$\mathbb{R}^p_+$	$\sum_i x_i \log x_i$	$\sum_i \left(\log\left(\frac{x_i}{y_i}\right) - (x_i - y_i)\right)$
von Neumann divergence	$\mathbb{S}^{p\times p}_+$	$\mathbf{X}\log\mathbf{X} - \mathbf{X}$	$\mathrm{tr}\,(\mathbf{X}(\log\mathbf{X} - \log\mathbf{Y}) - \mathbf{X} + \mathbf{Y})^e$
logdet divergence	$\mathbb{S}^{p\times p}_+$	$-\log\det\mathbf{X}$	$\mathrm{tr}\left(\mathbf{X}\mathbf{Y}^{-1}\right) - \log\det\left(\mathbf{X}\mathbf{Y}^{-1}\right) - p$

[a] $x, y \in \mathbb{R}$, $\mathbf{x},\mathbf{y} \in \mathbb{R}^p$ and $\mathbf{X},\mathbf{Y} \in \mathbb{R}^{p\times p}$.

[b] $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote non-negative and positive real numbers respectively.

[c] $\mathbf{A} \in \mathbb{S}^{p\times p}_+$, the set of symmetric positive semidefinite matrix.

[d] p-simplex:= $\{\mathbf{x} \in \mathbb{R}^p : \sum_{i=1}^p x_i = 1, x_i \geq 0, i = 1,\ldots,p\}$

[e] $\mathrm{tr}(\mathbf{A})$ is the trace of $\mathbf{A}$.

*Mirror descent [1]

What happens if we use a Bregman distance d_ψ in gradient descent?

Let $\psi : \mathbb{R}^p \to \mathbb{R}$ be a μ-strongly convex and continuously differentiable function and let the associated Bregman distance be $d_\psi(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{x}) - \psi(\mathbf{y}) - \langle \mathbf{x} - \mathbf{y}, \nabla\psi(\mathbf{y})\rangle$. Assume that the inverse mapping $\psi^\star$ of ψ is easily computable (i.e., its convex conjugate).

- **Majorize**: Find α_k such that

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k\rangle + \frac{1}{\alpha_k} d_\psi(\mathbf{x}, \mathbf{x}^k) := Q_\psi^k(\mathbf{x}, \mathbf{x}^k)$$

- **Minimize**

$$\begin{aligned}
\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_\psi^k(\mathbf{x}, \mathbf{x}^k) \Rightarrow \nabla f(\mathbf{x}^k) + \frac{1}{\alpha_k}\Big(\nabla\psi(\mathbf{x}^{k+1}) - \nabla\psi(\mathbf{x}^k)\Big) = 0 \\
\nabla\psi(\mathbf{x}^{k+1}) &= \nabla\psi(\mathbf{x}^k) - \alpha_k \nabla f(\mathbf{x}^k) \\
\mathbf{x}^{k+1} &= \nabla\psi^*(\nabla\psi(\mathbf{x}^k) - \alpha_k \nabla f(\mathbf{x}^k)) \qquad (\nabla\psi(\cdot))^{-1} = \nabla\psi^*(\cdot)[6].
\end{aligned}$$

- Mirror descent is a **generalization** of gradient descent for functions that are Lipschitz-gradient in norms other than the Euclidean.
- MD allows to deal with some **constraints** via a proper choice of ψ.

*Convergence analysis of mirror descent

Problem

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \tag{23}$$

where

- *$\mathcal{X}$ is a closed convex subset of $\mathbb{R}^p$;*
- *f is convex L_f-Lipschitz continuous with respect to some norm $\|\cdot\|$.*

Theorem ([1])

Let $\{\mathbf{x}^k\}$ be the sequence generated by mirror descent with $\mathbf{x}^0 \in \mathrm{int}\mathcal{X}$.
If the step-sizes are chosen as

$$\alpha_k = \frac{\sqrt{2\mu d_\psi(\mathbf{x}^\star, \mathbf{x}^0)}}{L_f}\frac{1}{\sqrt{k}}$$

the following convergence rate holds

$$\min_{0\leq s\leq k} f(\mathbf{x}^k) - f^\star \leq L_f\sqrt{\frac{2d_\psi(\mathbf{x}^\star, \mathbf{x}^0)}{\mu}}\frac{1}{\sqrt{k}}$$

- *This convergence rate is* **optimal** *for solving* (23) *with a first-order method.*

*Mirror descent example

How can we minimize a convex function over the unit simplex?

$$\min_{\mathbf{x}\in\Delta} f(\mathbf{x}),$$

where

- $\Delta := \{\mathbf{x}\in\mathbb{R}^p \ : \ \sum_{j=1}^p x_j = 1, \mathbf{x}\geq 0\}$ is the **unit simplex**;
- f is convex L_f-Lipschitz continuous with respect to some norm $\|\cdot\|$.

Entropy function

- Define the entropy function

$$\psi_e(\mathbf{x}) = \sum_{j=1}^p x_j \ln x_j \quad \text{if } \mathbf{x}\in\Delta, \quad +\infty \text{ otherwise.}$$

- ψ_e is 1-strongly convex over $\mathrm{int}\Delta$ with respect to $\|\cdot\|_1$.
- $\psi_e^\star(\mathbf{z}) = \ln\sum_{j=1}^p e^{z_j}$ and $\|\nabla\psi_e(\mathbf{x})\|\to\infty$ as $\mathbf{x}\to\tilde{\mathbf{x}}\in\Delta$.
- Let $\mathbf{x}^0 = p^{-1}\mathbf{1}$, then $d_\psi(\mathbf{x},\mathbf{x}^0)\leq \ln p$ for all $\mathbf{x}\in\Delta$.

*Entropic descent algorithm [1]

Entropic descent algorithm (EDA)

Let $\mathbf{x}^0 = p^{-1}\mathbf{1}$ and generate the following sequence

$$x_j^{k+1} = \frac{x_j^k e^{-t_k f_j'(\mathbf{x}^k)}}{\sum_{j=1}^p x_j^k e^{-t_k f_j'(\mathbf{x}^k)}}, \quad t_k = \frac{\sqrt{2\ln p}}{L_f}\frac{1}{\sqrt{k}},$$

where $f'(\mathbf{x}) = (f_1(\mathbf{x})', \ldots, f_p(\mathbf{x})')^T \in \partial f(\mathbf{x})$, which is the **subdifferential** of f at $\mathbf{x}$.

- This is an example of **non-smooth** and **constrained** optimization;
- The updates are multiplicative.

Convergence analysis

$$\min_{0\leq s\leq k} f(\mathbf{x}^s) - f^\star \leq \sqrt{2\ln p}\frac{\max_{0\leq s\leq k}\|f'(\mathbf{x}^s)\|_\infty}{\sqrt{k}}$$

Non-smooth unconstrained convex minimization

Problem (**Mathematical formulation**)

How can we find an optimal solution to the following optimization problem?

$$\boxed{F^\star := \min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x})} \tag{24}$$

where f is proper, closed, convex, but not everywhere differentiable, $f \in \mathcal{F}$.
Note that (24) is unconstrained.

Subgradient method

The subgradient method relies on the fact that even though f is non-smooth, we can still compute its **subgradients**, informing of the local descent directions.

Subgradient method

1. Choose $\mathbf{x}^0 \in \mathbb{R}^p$ as a starting point.
2. For $k = 0, 1, \cdots$, perform:

$$\left\{ \begin{array}{ll} \mathbf{x}^{k+1} & = \mathbf{x}^k - \alpha_k \mathbf{d}^k, \end{array} \right. \tag{25}$$

where $\mathbf{d}^k \in \partial f(\mathbf{x}^k)$ and $\alpha_k \in (0, 1]$ is a given step size.

Convergence of the subgradient method

Theorem

Assume that the following conditions are satisfied:

1. $\|\mathbf{g}\|_2 \leq G$ *for all* $\mathbf{g} \in \partial f(\mathbf{x})$ *for any* $\mathbf{x} \in \mathbb{R}^p$.
2. $\|\mathbf{x}^0 - \mathbf{x}^\star\|_2 \leq R$

Let the stepsize be chosen as

$$\alpha_k = \frac{R}{G\sqrt{k}}$$

then the iterates generated by the subgradient method satisfy

$$\min_{0 \leq i \leq k} f(\mathbf{x}^i) - f^\star \leq \frac{RG}{\sqrt{k}}.$$

Remarks

- Condition (1) holds, for example, when f is G-Lipschitz.
- The convergence rate of $O(\frac{1}{\sqrt{k}})$ is the slowest we have seen so far!

References

[1] Amir Beck and Marc Teboulle.
Mirror descent and nonlinear projected subgradient methods for convex optimization.
Operations Research Letters, 31(3):167–175, 2003.

[2] JE Dennis and Jorge J Moré.
A characterization of superlinear convergence and its application to quasi-newton methods.
Mathematics of Computation, 28(126):549–560, 1974.

[3] Simon Foucart and Holger Rauhut.
A Mathematical Introduction to Compressive Sensing.
Birkhäuser, Basel, 2013.

[4] Y. Nesterov.
Introductory lectures on convex optimization: A basic course, volume 87.
Springer, 2004.

[5] J. Nocedal and S.J. Wright.
Numerical Optimization.
Springer, 2006.

References

[6] R.T. Rockafellar.
Convex analysis.
Princeton University Press (Princeton, NJ), 1970.