Outline

- Convex analysis, optimality
- First-order methods
- Proximal methods, operator splitting
- Stochastic optimization, incremental methods
- Nonconvex models, algorithms
- Geometric optimization
Nonconvex problems

- SVD, PCA
- Other eigenvalue problems
- Matrix & tensor factorization, clustering
- Deep neural networks
- Topic models, Bayesian nonparametrics
- Probabilistic mixture models
- Combinatorial optimization
- Linear, nonlinear mixed integer programming
- Optimization on manifolds
- Optimization in metric spaces
- ...
Introduction

Nonlinear program

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \quad i = 1, \ldots, m.
\end{align*}
\]
Introduction

Nonlinear program

\[ \begin{align*}
\text{min} & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \quad i = 1, \ldots, m.
\end{align*} \]

Claim: If \( f \) and \( g_i \) are convex, then under some “constraint qualifications” (e.g., there exists an \( x \) for which \( g_i(x) < 0 \) holds), \textit{necessary and sufficient} conditions characterizing global optimality are known (e.g., \textit{Karush-Kuhn-Tucker}).
Introduction

Nonlinear program

\[
\begin{align*}
\text{min} & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \quad i = 1, \ldots, m.
\end{align*}
\]

\[0 \in \partial f(x^*)\] necessary and sufficient \((m = 0, \text{cvx})\)
Introduction

Nonlinear program

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \quad i = 1, \ldots, m.
\end{align*}
\]

**Nonconvex:** Under some constraint qualification, *necessary* conditions known. But *no known* simple conditions that are both necessary and sufficient.
Introduction

Nonlinear program

\[
\begin{align*}
\min & \quad f(x) \\
\text{s.t.} & \quad g_i(x) \leq 0, \quad i = 1, \ldots, m.
\end{align*}
\]

♠ Is alleged solution a local min? – often skipped question

♠ **Myth:** Algorithms converge to global minima for convex local minima for nonconvex
NP-Hardness of nonconvex opt.

Recall **subset-sum** – well-known NP-Complete problem

Given a set of integers \( \{a_1, \ldots, a_n\} \), is there a non-empty subset whose sum is zero?

Optimization version

\[
\begin{align*}
\min & \quad \sum_i a_i z_i \\
\text{s.t.} & \quad 0 \leq z_i \leq 1, \quad i = 1, \ldots, n
\end{align*}
\]

Subset-sum has feasible solution, iff global min objval is zero. But subset-sum is NP-Complete; so above problem also NPC.

Suvrit Sra (MIT) Convex, nonconvex, and geometric optim.
NP-Hardness of nonconvex opt.

Recall **subset-sum** – well-known NP-Complete problem

Given a set of integers \( \{a_1, \ldots, a_n\} \), is there a non-empty subset whose sum is zero?

In other words, is there a solution \( z \) to

\[
\sum_i a_i z_i = 0 \quad z_i \in \{0, 1\} \quad \text{for } i = 1, \ldots, n.
\]
NP-Hardness of nonconvex opt.

Recall **subset-sum** – well-known NP-Complete problem

Given a set of integers \( \{ a_1, \ldots, a_n \} \), is there a non-empty subset whose sum is zero?

In other words, is there a solution \( z \) to

\[
\sum_i a_i z_i = 0 \quad z_i \in \{0, 1\} \quad \text{for } i = 1, \ldots, n.
\]

**Optimization version**

\[
\min \left( \sum_i a_i z_i \right)^2 + \sum_i z_i(1 - z_i)
\]

s.t. \( 0 \leq z_i \leq 1 \), \( i = 1, \ldots, n \).

Subset-sum has feasible solution, **iff** global min objval is zero. But subset-sum is NP-Complete; so above problem also NPC.
Nonconvex quadratic optimization

Let $A$ be a symmetric matrix (not necessarily positive definite).

$$\min \quad x^T A x \quad \text{s.t.} \quad x \geq 0.$$ 

Is $x = 0$ not a local minimum?
Nonconvex quadratic optimization

Let $A$ be a symmetric matrix (not necessarily positive definite).

$$\min \quad x^T Ax \quad \text{s.t.} \quad x \geq 0.$$  

Is $x = 0$ not a local minimum?

This is NP-Hard!

Generally, even for unconstrained nonconvex problems testing local minimality or objective boundedness (below) are NP-Hard.
In “convex” words

**Copositive cone**

**Def.** Let $CP_n := \{ A \in S^{n \times n} \mid x^T A x \geq 0, \ \forall x \geq 0 \}$.

**Exercise:** Verify that $CP_n$ is a convex cone.

- Testing membership in $CP_n$ is co-NP complete.
  (Deciding whether given matrix is not copositive is NP-complete.)
- Copositive cone programming: NP-Hard

**Exercise:** Verify that the following matrix is copositive:

$$A := \begin{bmatrix}
1 & -1 & 1 & 1 & -1 \\
-1 & 1 & -1 & 1 & 1 \\
1 & -1 & 1 & -1 & 1 \\
1 & 1 & -1 & 1 & -1 \\
-1 & 1 & 1 & -1 & 1
\end{bmatrix}.$$
Let $A$ be a symmetric matrix; $b$ some vector.

$$\min \quad x^T Ax + 2b^T x, \quad \text{s.t. } x^T x \leq 1.$$
Let $A$ be a symmetric matrix; $b$ some vector.

$$\min x^T Ax + 2b^T x, \quad \text{s.t. } x^T x \leq 1.$$ 

When $A \not\succeq 0$, above problem is nonconvex. Also known as, trust-region subproblem (TRS).
Solvable nonconvex QP

Let $A$ be a symmetric matrix; $b$ some vector.

$$\min \ x^T Ax + 2b^T x, \ \text{s.t.} \ x^T x \leq 1.$$  

When $A \not\succeq 0$, above problem is nonconvex.

Also known as, *trust-region subproblem* (TRS).

Lagrangian

$$L(x, \theta) = x^T Ax + 2b^T x + \theta(x^T x - 1)$$

$$L(x, \theta) = x^T (A + \theta I)x + 2b^T x - \theta.$$
Let $A$ be a symmetric matrix; $b$ some vector.

$$
\min x^T Ax + 2b^T x, \quad \text{s.t. } x^T x \leq 1.
$$

When $A \not\succeq 0$, above problem is nonconvex.

Also known as, trust-region subproblem (TRS).

Lagrangian

$$
L(x, \theta) = x^T Ax + 2b^T x + \theta(x^T x - 1)
$$

$$
L(x, \theta) = x^T (A + \theta I)x + 2b^T x - \theta.
$$

If $b \notin \mathcal{R}(A + \theta I)$, then we can choose an $x \in \mathcal{N}(A + \theta I)$ that drives $\inf_x L(x, \theta)$ to $-\infty$. 
Let $A$ be a symmetric matrix; $b$ some vector.

$$
\min \quad x^T Ax + 2b^T x, \quad \text{s.t. } x^T x \leq 1.
$$

When $A \not\succeq 0$, above problem is nonconvex.

Also known as, trust-region subproblem (TRS).

Lagrangian

$$
L(x, \theta) = x^T Ax + 2b^T x + \theta(x^T x - 1)
$$

$$
L(x, \theta) = x^T (A + \theta I)x + 2b^T x - \theta.
$$

If $b \notin \mathcal{R}(A + \theta I)$, then we can choose an $x \in \mathcal{N}(A + \theta I)$ that drives $\inf_x L(x, \theta)$ to $-\infty$.

$$
g(\theta) := \begin{cases} 
-b^T (A + \theta I)^{\dagger} b - \theta & A + \theta I \succeq 0, \ b \in \mathcal{R}(A + \theta I) \\
-\infty & \text{otherwise}
\end{cases}
$$
A nice nonconvex problem

Dual optimization problem

\[
\begin{align*}
\max & \quad -b^T (A + \theta I)^\dagger b - \theta \\
\text{s.t.} & \quad A + \theta I \succeq 0, \ b \in \mathcal{R}(A + \theta I).
\end{align*}
\]
A nice nonconvex problem

Dual optimization problem

\[
\begin{align*}
\text{max} & \quad - b^T (A + \theta I)^\dagger b - \theta \\
\text{s.t.} & \quad A + \theta I \succeq 0, \ b \in \mathcal{R}(A + \theta I).
\end{align*}
\]

Consider eigendecomposition of \( A = U \Lambda U^T \). Then,

\[
(A + \theta I)^\dagger = U \text{Diag}(1 + \lambda_i)^{-1} U^T.
\]
A nice nonconvex problem

Dual optimization problem

\[
\begin{align*}
\max & \quad -b^T(A + \theta I)^\dagger b - \theta \\
\text{s.t.} & \quad A + \theta I \succeq 0, \ b \in \mathcal{R}(A + \theta I).
\end{align*}
\]

Consider eigendecomposition of \( A = U\Lambda U^T \). Then,

\[
(A + \theta I)^\dagger = U\text{Diag}(1 + \lambda_i)^{-1}U^T.
\]

Thus, above problem can be written as

\[
\begin{align*}
\max & \quad -\sum_{i=1}^n \frac{(u_i^T b)^2}{\lambda_i + \theta} - \theta \\
\text{s.t.} & \quad \theta \geq -\lambda_{\min}(A).
\end{align*}
\]

😊 Convex optimization problem!
Matrix Factorization
The SVD

Singular Value Decomposition

**Theorem** SVD (Thm. 2.5.2 [GoLo96]). Let $A \in \mathbb{R}^{m \times n}$. There exist orthogonal matrices $U$ and $V$

$$U^T AV = \text{Diag}(\sigma_1, \ldots, \sigma_p), \quad p = \min(m, n),$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$. 

Suvrit Sra (MIT)  Convex, nonconvex, and geometric optim.
**Theorem** Let $A$ have the SVD $U\Sigma V^T$. If $k < \text{rank}(A)$ and

$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T,$$

then,

$$\|A - A_k\|_2 \leq \|A - B\|_2, \text{ s.t. } \text{rank}(B) \leq k$$

$$\|A - A_k\|_F \leq \|A - B\|_F, \text{ s.t. } \text{rank}(B) \leq k.$$
**Theorem** Let $A$ have the SVD $U \Sigma V^T$. If $k < \text{rank}(A)$ and

$$A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^T,$$

then,

$$\|A - A_k\|_2 \leq \|A - B\|_2, \text{ s.t. } \text{rank}(B) \leq k$$

$$\|A - A_k\|_F \leq \|A - B\|_F, \text{ s.t. } \text{rank}(B) \leq k.$$ 

SVD gives **globally optimal** solution to the nonconvex problem

$$\min \|X - A\|_F, \quad \text{s.t. } \text{rank}(X) \leq k.$$
Truncated SVD – proof

Prove: TSVD yields “best” rank-$k$ approximation to matrix $A$

**Proof.**

1. First verify that $\|A - A_k\|_2 = \sigma_{k+1}$
2. Let $B$ be any rank-$k$ matrix
3. Prove that $\|A - B\|_2 \geq \sigma_{k+1}$

Since $\text{rank}(B) = k$, there are $n - k$ vectors that span the null-space $\mathcal{N}(B)$. But $\mathcal{N}(B) \cap V_{k+1} \neq \{0\}$ (??), so we can pick a unit-norm vector $z \in \mathcal{N}(B) \cap V_{k+1}$. Now $Bz = 0$, so

$$\|A - B\|_2^2 \geq \|(A - B)z\|_2^2 = \|Az\|_2^2 = \sum_{i=1}^{k+1} \sigma_i^2 (v_i^T z)^2 \geq \sigma_{k+1}^2$$

We used: $\|Az\|_2 \leq \|A\|_2 \|z\|_2$
Nonnegative matrix factorization

Say we want a \textit{low-rank approximation} \( A \approx BC \)
Nonnegative matrix factorization

Say we want a *low-rank approximation* $A \approx BC$

- SVD yields dense $B$ and $C$
- $B$ and $C$ contain negative entries, even if $A \geq 0$
- SVD factors may not be that easy to interpret
Nonnegative matrix factorization

Say we want a *low-rank approximation* \( A \approx BC \)

- SVD yields dense \( B \) and \( C \)
- \( B \) and \( C \) contain negative entries, even if \( A \geq 0 \)
- SVD factors may not be that easy to interpret

\[
\text{NMF imposes } B \geq 0, \ C \geq 0
\]
Algorithms

\[ A \approx BC \quad \text{s.t.} \quad B, C \geq 0 \]

**Least-squares NMF**

\[ \min \quad \frac{1}{2} \| A - BC \|_F^2 \quad \text{s.t.} \quad B, C \geq 0. \]

**KL-Divergence NMF**

\[ \min \quad \sum_{ij} a_{ij} \log \frac{(BC)_{ij}}{a_{ij}} - a_{ij} + (BC)_{ij} \quad \text{s.t.} \quad B, C \geq 0. \]
Algorithms

\[ A \approx BC \quad \text{s.t. } B, C \geq 0 \]

**Least-squares NMF**

\[
\begin{align*}
\min & \quad \frac{1}{2} \| A - BC \|_F^2 \\
\text{s.t.} & \quad B, C \geq 0.
\end{align*}
\]

**KL-Divergence NMF**

\[
\begin{align*}
\min & \quad \sum_{ij} a_{ij} \log \frac{(BC)_{ij}}{a_{ij}} - a_{ij} + (BC)_{ij} \\
\text{s.t.} & \quad B, C \geq 0.
\end{align*}
\]

♠ NP-Hard (Vavasis 2007) – no surprise

♠ Recently, Arora et al. showed that if the matrix \( A \) has a special “separable” structure, then actually globally optimal NMF is approximately solvable. More recent progress too!

♣ We look at only basic methods in this lecture
NMF Algorithms

- Hack: Compute TSVD; “zero-out” negative entries
- Alternating minimization (AM)
- Majorize-Minimize (MM)
- Global optimization (not covered)
- Incremental gradient algorithms
Alternating Descent

\[
\min \quad F(B, C)
\]

Alternating Descent

1. Initialize \( B^0, k \leftarrow 0 \)
2. Compute \( C^{k+1} \) s.t. \( F(A, B^k C^{k+1}) \leq F(A, B^k C^k) \)
3. Compute \( B^{k+1} \) s.t. \( F(A, B^{k+1} C^{k+1}) \leq F(A, B^k C^{k+1}) \)
4. \( k \leftarrow k + 1 \), and repeat until stopping criteria met.
Alternating Minimization

Alternating Least Squares

\[ C = \arg\min_C \| A - B^k C \|^2_F; \]
Alternating Minimization

Alternating Least Squares

\[ C = \arg\min_C \| A - B^k C \|^2_F; \quad C^{k+1} \leftarrow \max(0, C) \]
Alternating Minimization

Alternating Least Squares

\[ C = \arg\min_C \| A - B^k C \|_F^2; \quad C^{k+1} \leftarrow \max(0, C) \]

\[ B = \arg\min_B \| A - BC^{k+1} \|_F^2; \quad B^{k+1} \leftarrow \max(0, B) \]
Alternating Minimization

Alternating Least Squares

\[ C = \arg\min_C \| A - B^k C \|_F^2; \quad C^{k+1} \leftarrow \max(0, C) \]

\[ B = \arg\min_B \| A - BC^{k+1} \|_F^2; \quad B^{k+1} \leftarrow \max(0, B) \]

ALS is fast, simple, often effective, but ...
Alternating Minimization

Alternating Least Squares

\[ C = \arg\min_C \| A - B^k C \|_F^2; \quad C^{k+1} \leftarrow \max(0, C) \]

\[ B = \arg\min_B \| A - B C^{k+1} \|_F^2; \quad B^{k+1} \leftarrow \max(0, B) \]

ALS is fast, simple, often effective, but ...

\[ \| A - B^{k+1} C^{k+1} \|_F^2 \leq \| A - B^k C^{k+1} \|_F^2 \leq \| A - B^k C^k \|_F^2 \]

descent need not hold

Suvrit Sra (MIT)  Convex, nonconvex, and geometric optim.
Alternating Minimization: correctly

Use alternating **nonnegative least-squares**

\[
C^{k+1} = \arg\min_C \|A - B^k C\|_F^2 \quad \text{s.t.} \quad C \geq 0
\]

\[
B^{k+1} = \arg\min_B \|A - BC^{k+1}\|_F^2 \quad \text{s.t.} \quad B \geq 0
\]

**Advantages:** Guaranteed descent. Theory of block-coordinate descent guarantees convergence to *stationary point*.

**Disadvantages:** more complex; slower than ALS
Convergence

AM / two block CD

\[ \min F(x) = F(x_1, x_2) \quad x \in \mathcal{X}_1 \times \mathcal{X}_2. \]

**Theorem** (Grippo & Sciandrone (2000)). Let \( F \) be continuously differentiable, and the sets \( \mathcal{X}_1, \mathcal{X}_2 \) be closed and convex. Assume that the both BCD subproblems have solutions, and that the sequence \( \{x^k\} \) has limit points. Then, every limit point of \( \{x^k\} \) is stationary.

- No need of *unique solutions* to subproblems
- BCD for 2 blocks aka **Alternating Minimization**
Alternating Proximal Method

\[ \min L(x, y) := F(x, y) + G(x) + H(y). \]

Assume: \( \nabla F \) Lipschitz cont. on bounded subsets of \( \mathbb{R}^m \times \mathbb{R}^n \)

\( G \): lower semicontinuous on \( \mathbb{R}^m \)

\( H \): lower semicontinuous on \( \mathbb{R}^n \).

Example: \( F(x, y) = \frac{1}{2} \| x - y \|^2 \)
Alternating Proximal Method

\[
\min L(x, y) := F(x, y) + G(x) + H(y).
\]

Assume: \( \nabla F \) Lipschitz cont. on bounded subsets of \( \mathbb{R}^{\text{m}} \times \mathbb{R}^{\text{n}} \)

\( G \): lower semicontinuous on \( \mathbb{R}^{\text{m}} \)

\( H \): lower semicontinuous on \( \mathbb{R}^{\text{n}} \).

**Example:** \( F(x, y) = \frac{1}{2} \| x - y \|^2 \)

**Alternating Proximal Method**

\[
x_{k+1} \in \text{argmin} \left\{ L(x, y_k) + \frac{1}{2} c_k \| x - x_k \|^2 \right\}
\]

\[
y_{k+1} \in \text{argmin} \left\{ L(x_{k+1}, y) + \frac{1}{2} c'_k \| y - y_k \|^2 \right\},
\]

here \( c_k, c'_k \) are suitable sequences of positive scalars.

Descent Techniques
Consider $F(B, C) = \frac{1}{2} \| A - BC \|_F^2$: convex separately in $B$ and $C$.

We use $F(C)$ to denote function restricted to $C$.

Since $F(C)$ is separable, suffices to illustrate for

$$\min_{c \geq 0} f(c) = \frac{1}{2} \| a - Bc \|_2^2$$

Recall, our aim is: find $C_{k+1}$ such that $F(B_k, C_{k+1}) \leq F(B_k, C_k)$.
Descent technique

\[
\min_{c \geq 0} \quad f(c) = \frac{1}{2} \| a - Bc \|_2^2
\]

1. Find a function \( g(c, \tilde{c}) \) that satisfies:

\[
\begin{align*}
  g(c, c) & = f(c), \quad \text{for all } c, \\
  g(c, \tilde{c}) & \geq f(c), \quad \text{for all } c, \tilde{c}.
\end{align*}
\]
Descent technique

\[
\min_{c \geq 0} \quad f(c) = \frac{1}{2} \| a - Bc \|_2^2
\]

1. Find a function \( g(c, \tilde{c}) \) that satisfies:

\[
\begin{align*}
  g(c, c) &= f(c), \quad \text{for all } c, \\
  g(c, \tilde{c}) &\geq f(c), \quad \text{for all } c, \tilde{c}.
\end{align*}
\]

2. Compute \( c^{t+1} = \arg\min_{c \geq 0} g(c, c^t) \).
Descent technique

\[
\min_{c \geq 0} \quad f(c) = \frac{1}{2} \| a - Bc \|_2^2
\]

1. Find a function \( g(c, \tilde{c}) \) that satisfies:
   \[
   g(c, c) = f(c), \quad \text{for all} \quad c,
   \]
   \[
   g(c, \tilde{c}) \geq f(c), \quad \text{for all} \quad c, \tilde{c}.
   \]

2. Compute \( c^{t+1} = \arg\min_{c \geq 0} g(c, c^t) \)

3. Then we have descent
Descent technique

\[
\min_{c \geq 0} \quad f(c) = \frac{1}{2} \|a - Bc\|_2^2
\]

1. Find a function \( g(c, \tilde{c}) \) that satisfies:
   \[
   g(c, c) = f(c), \quad \text{for all} \quad c,
   \]
   \[
   g(c, \tilde{c}) \geq f(c), \quad \text{for all} \quad c, \tilde{c}.
   \]

2. Compute \( c^{t+1} = \arg\min_{c \geq 0} g(c, c^t) \)

3. Then we have descent
   \[
   f(c^{t+1})
   \]
1. Find a function $g(c, \tilde{c})$ that satisfies:

$$
g(c, c) = f(c), \quad \text{for all } c,$$

$$
g(c, \tilde{c}) \geq f(c), \quad \text{for all } c, \tilde{c}.
$$

2. Compute $c^{t+1} = \arg\min_{c \geq 0} g(c, c^t)$

3. Then we have descent

$$
f(c^{t+1}) \overset{\text{def}}{=} g(c^{t+1}, c^t)
$$
Descent technique

\[ \min_{c \geq 0} f(c) = \frac{1}{2} \| a - Bc \|_2^2 \]

1. Find a function \( g(c, \tilde{c}) \) that satisfies:
   \[
   g(c, c) = f(c), \quad \text{for all } c, \\
   g(c, \tilde{c}) \geq f(c), \quad \text{for all } c, \tilde{c}.
   \]

2. Compute \( c^{t+1} = \arg\min_{c \geq 0} g(c, c^t) \)

3. Then we have descent

\[
 f(c^{t+1}) \overset{\text{def}}{=} \arg\min_{c \geq 0} g(c^{t+1}, c^t) \leq g(c^{t+1}, c^t) \leq g(c^t, c^t)
\]
Descent technique

$$\min_{c \geq 0} f(c) = \frac{1}{2}\|a - Bc\|^2_2$$

1. Find a function $g(c, \tilde{c})$ that satisfies:
   $$g(c, c) = f(c), \quad \text{for all } c,$$
   $$g(c, \tilde{c}) \geq f(c), \quad \text{for all } c, \tilde{c}.$$

2. Compute $c^{t+1} = \text{argmin}_{c \geq 0} g(c, c^t)$

3. Then we have descent
   $$f(c^{t+1}) \overset{\text{def}}{=} g(c^{t+1}, c^t) \overset{\text{argmin}}{\leq} g(c^t, c^t) \overset{\text{def}}{=} f(c^t).$$
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2} x^2$ is a convex function

$$h(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i h(x_i), \text{ where } \lambda_i \geq 0, \sum_i \lambda_i = 1$$
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2} x^2$ is a convex function

$$h(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i h(x_i), \text{ where } \lambda_i \geq 0, \sum_i \lambda_i = 1$$

$$f(c) = \frac{1}{2} \sum_i (a_i - b_i^T c)^2 =$$
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2}x^2$ is a **convex function**

\[
h(\sum \lambda_i x_i) \leq \sum \lambda_i h(x_i), \text{ where } \lambda_i \geq 0, \sum \lambda_i = 1
\]

\[
f(c) = \frac{1}{2} \sum_i (a_i - b_i^T c)^2 = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + (b_i^T c)^2
\]
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2} x^2$ is a convex function

$$h(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i h(x_i), \text{ where } \lambda_i \geq 0, \sum_i \lambda_i = 1$$

$$f(c) = \frac{1}{2} \sum_i (a_i - b_i^T c)^2 = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + (b_i^T c)^2$$

$$= \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i (\sum_j b_{ij} c_j)^2$$
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2} x^2$ is a convex function

\[
h\left(\sum_i \lambda_i x_i\right) \leq \sum_i \lambda_i h(x_i), \text{ where } \lambda_i \geq 0, \sum_i \lambda_i = 1
\]

\[
f(c) = \frac{1}{2} \sum_i (a_i - b_i^T c)^2 = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + (b_i^T c)^2
\]

\[
= \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i \left(\sum_j b_{ij} c_j\right)^2
\]

\[
= \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c
\]

Suvrit Sra (MIT)  
Convex, nonconvex, and geometric optim.
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2} x^2$ is a convex function

\[
  h(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i h(x_i) \text{, where } \lambda_i \geq 0, \sum_i \lambda_i = 1
\]

\[
  f(c) = \frac{1}{2} \sum_i (a_i - b_i^T c)^2 = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + (b_i^T c)^2
\]

\[
  = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i \left( \sum_j b_{ij} c_j \right)^2
\]

\[
  = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i \left( \sum_j \lambda_{ij} b_{ij} c_j / \lambda_{ij} \right)^2
\]
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2} x^2$ is a convex function

$h(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i h(x_i)$, where $\lambda_i \geq 0$, $\sum_i \lambda_i = 1$

$$f(c) = \frac{1}{2} \sum_i (a_i - b_i^T c)^2 = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + (b_i^T c)^2$$

$$= \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i (\sum_j b_{ij} c_j)^2$$

$$= \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i (\sum_j \lambda_{ij} b_{ij} c_j / \lambda_{ij})^2$$

$$\leq \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_{ij} \lambda_{ij} (b_{ij} c_j / \lambda_{ij})^2$$
Descent technique – constructing $g$

We exploit that $h(x) = \frac{1}{2} x^2$ is a convex function

$$h(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i h(x_i), \text{ where } \lambda_i \geq 0, \sum_i \lambda_i = 1$$

$$f(c) = \frac{1}{2} \sum_i (a_i - b_i^T c)^2 = \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + (b_i^T c)^2$$

$$= \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i (\sum_j b_{ij} c_j)^2$$

$$= \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_i (\sum_j \lambda_{ij} b_{ij} c_j / \lambda_{ij})^2$$

$$\leq \frac{1}{2} \sum_i a_i^2 - 2a_i b_i^T c + \frac{1}{2} \sum_{ij} \lambda_{ij} (b_{ij} c_j / \lambda_{ij})^2$$

$$=: g(c, \tilde{c}), \text{ where } \lambda_{ij} \text{ are convex coeffts}$$
Descent technique – constructing $g$

\[
\begin{align*}
f(c) &= \frac{1}{2} \| a - Bc \|_2^2 \\
g(c, \tilde{c}) &= \frac{1}{2} \| a \|_2^2 - \sum_i a_i b_i^T c + \frac{1}{2} \sum_{ij} \lambda_{ij} (b_{ij} c_j / \lambda_{ij})^2.
\end{align*}
\]

Only remains to pick $\lambda_{ij}$ as functions of $\tilde{c}$
Descent technique – constructing $g$

\[ f(c) = \frac{1}{2} \| a - Bc \|_2^2 \]
\[ g(c, \tilde{c}) = \frac{1}{2} \| a \|_2^2 - \sum_i a_i b_i^T c + \frac{1}{2} \sum_{ij} \lambda_{ij} (b_{ij} c_j / \lambda_{ij})^2. \]

Only remains to pick $\lambda_{ij}$ as functions of $\tilde{c}$

\[ \lambda_{ij} = \frac{b_{ij} \tilde{c}_j}{\sum_k b_{ik} \tilde{c}_k} = \frac{b_{ij} \tilde{c}_j}{b_i^T \tilde{c}} \]
Descent technique – constructing $g$

\[ f(c) = \frac{1}{2} \| a - Bc \|_2^2 \]
\[ g(c, \tilde{c}) = \frac{1}{2} \| a \|_2^2 - \sum_i a_i b_i^T c + \frac{1}{2} \sum_{ij} \lambda_{ij} (b_{ij} c_j / \lambda_{ij})^2. \]

Only remains to pick $\lambda_{ij}$ as functions of $\tilde{c}$

\[ \lambda_{ij} = \frac{b_{ij} \tilde{c}_j}{\sum_k b_{ik} \tilde{c}_k} = \frac{b_{ij} \tilde{c}_j}{b_i^T \tilde{c}} \]

Exercise: Verify that $g(c, c) = f(c)$;

Exercise: Let $f(c) = \sum_i a_i \log(\frac{a_i}{(Bc)_i}) - a_i + (Bc)_i$. Derive an auxiliary function $g(c, \tilde{c})$ for this $f(c)$. 

Suvrit Sra (MIT) Convex, nonconvex, and geometric optim.
Descent technique – Exercise

Key step

\[ c^{t+1} = \arg\min_{c \geq 0} g(c, c^t). \]

Exercise: Solve \( \frac{\partial g(c, c^t)}{\partial c_p} = 0 \) to obtain

\[ c_p = c^t_p \frac{[B^T a]_p}{[B^T B c^t]_p} \]

This yields the “multiplicative update” algorithm of Lee/Seung (1999).
MM algorithms

- We exploited convexity of $x^2$
- Expectation Maximization (EM) algorithm exploits convexity of $-\log x$
- Other choices possible, e.g., by varying $\lambda_{ij}$
- Our technique one variant of repertoire of Majorization-Minimization (MM) algorithms
- gradient-descent also an MM algorithm
- Related to \textit{d.c. programming}
- MM algorithms subject of a separate lecture!
Generic descent method

Nonsmooth, nonconvex min

\[
\min \quad f(x)
\]

Methods that generate \((x_k, w_k)\) such that

\[
f(x_{k+1}) + a\|x_{k+1} - x_k\|^2 \leq f(x_k)
\]

there exists \(w_{k+1} \in \partial f(x_{k+1})\) s.t. \(\|x_{k+1} - x_k\| \geq b\|w_{k+1}\|\).
**Generic descent method**

**Nonsmooth, nonconvex min**

\[
\min f(x)
\]

Methods that generate \((x_k, w_k)\) such that

\[
f(x_{k+1}) + a\|x_{k+1} - x_k\|^2 \leq f(x_k)
\]

there exists \(w_{k+1} \in \partial f(x_{k+1})\) s.t. \(\|x_{k+1} - x_k\| \geq b\|w_{k+1}\|\).

**Condition 1:** Sufficient descent from \(x_k\) to \(x_{k+1}\)

**Condition 2:** Captures inexactness (approx. optimality)

**Example:** captures nonconvex proximal gradient method.

Other Alternating methods

- Nonconvex ADMM (e.g., arXiv:1410.1390)
- Nonconvex Douglas-Rachford (e.g., Borwein’s webpage!)
- Alternating minimization for global optimization
  e.g., [Jain, Netrapalli, Sanghavi (2013). *Low-rank matrix completion using alternating minimization*. STOC 2013.]
- BCD with more than 2 blocks
- Several others...
Large-scale methods
**Stochastic optimization**

**Assumption 1:** Possible to generate iid samples $\xi_1, \xi_2, \ldots$

**Assumption 2:** Oracle yields **stochastic gradient** $g(x, \xi)$, i.e.,

$$G(x) := \mathbb{E}[g(x, \xi)] \quad \text{s.t.} \quad G(x) \in \partial F(x).$$
Stochastic optimization

**Assumption 1:** Possible to generate iid samples $\xi_1, \xi_2, \ldots$

**Assumption 2:** Oracle yields stochastic gradient $g(x, \xi)$, i.e.,

$$G(x) := \mathbb{E}[g(x, \xi)] \text{ s.t. } G(x) \in \partial F(x).$$

**Theorem** Let $\xi \in \Omega$; If $f(\cdot, \xi)$ is convex, and $F(\cdot)$ is finite valued in a neighborhood of $x$, then

$$\partial F(x) = \mathbb{E}[\partial_x f(x, \xi)].$$
Stochastic optimization

**Assumption 1:** Possible to generate iid samples $\xi_1, \xi_2, \ldots$

**Assumption 2:** Oracle yields stochastic gradient $g(x, \xi)$, i.e.,

$$G(x) := \mathbb{E}[g(x, \xi)] \quad \text{s.t.} \quad G(x) \in \partial F(x).$$

**Theorem** Let $\xi \in \Omega$; If $f(\cdot, \xi)$ is convex, and $F(\cdot)$ is finite valued in a neighborhood of $x$, then

$$\partial F(x) = \mathbb{E}[\partial_x f(x, \xi)].$$

- So $g(x, \omega) \in \partial_x f(x, \omega)$ is a stochastic subgradient.
Stochastic gradient

- Let $x_0 \in \mathcal{X}$
- For $k \geq 0$
  - Sample $\xi_k$; compute $g(x_k, \xi_k)$ using oracle
  - Update $x_{k+1} = P_{\mathcal{X}}(x_k - \alpha_k g(x_k, \xi_k))$, where $\alpha_k > 0$

Simply write

$$x_{k+1} = P_{\mathcal{X}}(x_k - \alpha_k g_k)$$
The incremental gradient method (IGM)

- Let $x_0 \in \mathbb{R}^n$
- For $k \geq 0$
  1. Pick $i(k) \in \{1, 2, \ldots, n\}$ uniformly at random
  2. $x_{k+1} = x_k - \eta_k \nabla f_{i(k)}(x_k)$
Incremental Gradient Methods

\[ \min F(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x) \]

The incremental gradient method (IGM)

- Let \( x_0 \in \mathbb{R}^n \)
- For \( k \geq 0 \)
  1. Pick \( i(k) \in \{1, 2, \ldots, n\} \) uniformly at random
  2. \( x_{k+1} = x_k - \eta_k \nabla f_{i(k)}(x_k) \)

\( g \equiv \nabla f_{i(k)} \) may be viewed as a \textbf{stochastic gradient}

\[ g := g^\text{true} + e, \text{ where } e \text{ is mean-zero noise: } \mathbb{E}[e] = 0 \]
Example application

Multiframe blind deconvolution

(video)
Problem setup

time $t$

\[
\begin{align*}
y_t &= a_t \ast x + n_t \\
0 &= \ast + n_0 \\
1 &= \ast + n_1 \\
2 &= \ast + n_2 \\
k &= \ast + n_k
\end{align*}
\]
Formulation as matrix factorization

\[
\begin{bmatrix}
\vdots \\
y_1 & y_n \\
\vdots
\end{bmatrix} \approx \begin{bmatrix}
\vdots \\
a_1 & a_t \\
\vdots
\end{bmatrix} \times x
\]

Rewrite: \( a \times x = Ax = Xa \)

\[
\begin{bmatrix}
y_1 & y_2 & \cdots & y_t
\end{bmatrix} \approx X \begin{bmatrix}
a_1 & a_2 & \cdots & a_t
\end{bmatrix}
\]

\( Y \approx XA \)
Large-scale problem

Example, 5000 frames of size $512 \times 512$

$$Y_{262144 \times 5000} \approx X_{262144 \times 262144} A_{262144 \times 5000}$$

Without structure $\approx$ 70 billion parameters!
With structure, $\approx$ 4.8 million parameters!
Example, 5000 frames of size $512 \times 512$

$Y_{262144 \times 5000} \approx X_{262144 \times 262144} A_{262144 \times 5000}$

Without structure $\approx 70$ billion parameters!

With structure, $\approx 4.8$ million parameters!

Despite structure, alternating minimization impractical

Fix $X$, solve for $A$, requires updating $\approx 4.5$ million params
Solving the problem

\[ \min_{A_t, x} \sum_{t=1}^{T} \frac{1}{2} \| y_t - A_t x \|^2 + \Omega(x) + \Gamma(A_t) \]
Solving the problem

\[
\min_{A_t, x} \sum_{t=1}^{T} \frac{1}{2} \| y_t - A_t x \|^2 + \Omega(x) + \Gamma(A_t)
\]

Initialize guess \( x_0 \)
For \( t = 1, 2, \ldots \)
1. Observe image \( y_t \);
Solving the problem

\[
\min_{A_t, x} \sum_{t=1}^{T} \frac{1}{2} \| y_t - A_t x \|^2 + \Omega(x) + \Gamma(A_t)
\]

Initialize guess \( x_0 \)
For \( t = 1, 2, \ldots \)
1. Observe image \( y_t \);
2. Use \( x_{t-1} \) to estimate \( A_t \)
Solving the problem

\[
\min_{A_t, x} \sum_{t=1}^{T} \frac{1}{2} \|y_t - A_t x\|^2 + \Omega(x) + \Gamma(A_t)
\]

Initialize guess \(x_0\)
For \(t = 1, 2, \ldots\)
1. Observe image \(y_t\);
2. Use \(x_{t-1}\) to **estimate** \(A_t\)
3. Solve **optimization subproblem** to obtain \(x_t\)

[Harmeling, Hirsch, Sra, Schölkopf (ICCP'09); Hirsch, Sra, Schölkopf, Harmeling (CVPR'10); Hirsch, Harmeling, Sra, Schölkopf (Astron. & Astrophy. (AA) 2011); Harmeling, Hirsch, Sra, Schölkopf, Schuler (Patent 2012); Sra (NIPS'12)]
Solving the problem

\[
\min_{A_t, x} \sum_{t=1}^{T} \frac{1}{2} \| y_t - A_t x \|^2 + \Omega(x) + \Gamma(A_t)
\]

Initialize guess \(x_0\)

For \(t = 1, 2, \ldots\)

1. Observe image \(y_t\);
2. Use \(x_{t-1}\) to **estimate** \(A_t\)
3. Solve **optimization subproblem** to obtain \(x_t\)

---

**Step 2.** Model, estimate blur \(A_t\) — separate talk

**Step 3.** convex subproblem — reuse convex building blocks

Do Steps 2, 3 **inexactlly** \(\Rightarrow\) realtime processing!

[Harmeling, Hirsch, Sra, Schölkopf (ICCP’09); Hirsch, Sra, Schölkopf, Harmeling (CVPR’10); Hirsch, Harmeling, Sra, Schölkopf (Astron. & Astrophy. (AA) 2011); Harmeling, Hirsch, Sra, Schölkopf, Schuler (Patent 2012); Sra (NIPS’12)]

Suvrit Sra (MIT) Convex, nonconvex, and geometric optim.
Solving the problem: rewriting

Key idea

\[
\min_{X,A} \Phi(X,A) \equiv \min_X \left( \min_A \Phi(X,A) \right) =
\]

but now \( F \) is nonconvex

Suvrit Sra (MIT) Convex, nonconvex, and geometric optim.
Solving the problem: rewriting

Key idea

\[
\min_{X,A} \Phi(X, A) \equiv \min_X \left( \min_A \Phi(X, A) \right) = \min_X F(X)
\]

\[
F(X) \ := \ \min_A \Phi(X, A)
\]
Solving the problem: rewriting

Key idea

\[
\min_{X,A} \Phi(X, A) \equiv \min_X \left( \min_A \Phi(X, A) \right) = \min_X F(X)
\]

\[
F(X) := \min_A \Phi(X, A)
\]

\[
\Phi(X, A) = \| Y - XA \|^2 + \Omega(X) + \Gamma(A)
\]

\[
\min_X F(X) + \Omega(X)
\]

but now \( F \) is nonconvex
Key to scalability

\[ X^{\text{new}} \leftarrow \text{prox}_{\alpha \Omega}(X - \alpha \nabla F(X)) \]
Key to scalability

\[ X^{\text{new}} \leftarrow \text{prox}_{\alpha \Omega}(X - \alpha \nabla F(X) + e) + p \]

If gradient is \textit{inexactly} computed

If \text{prox}_\Omega \textit{inexactly} computed
Key to scalability

\[ X^{\text{new}} \leftarrow \text{prox}_{\alpha \Omega}(X - \alpha \nabla F(X) + e) + p \]

If gradient is **inexactly** computed
If \( \text{prox}_\Omega \) **inexactly** computed

Example: Say \( F(X) = \sum_{i=1}^{m} f_i(X) \)
Instead of \( \nabla F(X) \), use \( \nabla f_k(x) \)—**incremental**
\( m \) times cheaper (\( m \) can be in the millions or more)

**Inexactness**: key to scalability
incremental prox-method for **large-scale nonconvex**

[Sra (NIPS 12)]; (also arXiv: [math.OC-1109.0258])

**Theorem** Limits points are approximately stationary.
Non-asymptotic convergence

\[ \min \frac{1}{n} \sum_i f_i(x) \]

**SGD**

1. For \( t = 0 \) to \( T - 1 \):
   1. Pick \( i_t \) from \( \{1, \ldots, n\} \)
   2. Update \( x_{t+1} \leftarrow x_t - \eta_t \nabla f_i(x_t) \)

Theorem (Ghadimi, Lan). Suppose \( \|\nabla f_i(x)\| \leq G \) for all \( i \), \( \eta_t = c / \sqrt{T} \), and \( f \in C_1^L \). Then,

\[ \mathbb{E} \left[ \|\nabla f\|^2 \right] \leq \frac{1}{c} \sqrt{T} \left( f(x_0) - f(x^*) + \frac{1}{2} L c^2 G^2 \right) \]

Non-asymptotic convergence

\[ \min \frac{1}{n} \sum_i f_i(x) \]

**SGD**

1. For \( t = 0 \) to \( T - 1 \):
   1. Pick \( i_t \) from \( \{1, \ldots, n\} \)
   2. Update \( x_{t+1} \leftarrow x_t - \eta_t \nabla f_i(x_t) \)

**Theorem** (Ghadimi, Lan). Suppose \( \|\nabla f_i(x)\| \leq G \) for all \( i \), \( \eta_t = c/\sqrt{T} \), and \( f \in C^1_L \). Then,

\[
\mathbb{E}[\|\nabla f\|^2] \leq \frac{1}{c\sqrt{T}} \left( f(x_0) - f(x^*) + \frac{1}{2}Lc^2G^2 \right)
\]

Other ncvx incremental methods


First two do not prove rates; third one builds on Ghadimi & Lan’s analysis to provide rate on $\mathbb{E}[\|\nabla f\|^2]$
Geometric Optimization
Geometry Everywhere

- The usual vector space
- Manifolds (hypersphere, orthogonal matrices, complicated surfaces)
- Convex sets (probability simplex, semidefinite cone, polyhedra)
- Metric spaces (tree space, Wasserstein metric, negatively curved spaces)

Machine Learning
Graphics
Robotics
Vision
BCI
NLP
Statistics
Geometric Data

Rotations

Covariances as data / features / params: $X_1, X_2, \ldots, X_n \succeq 0$

Radar  DTI  CV  BCI  DeepLrn

[Cherian, Sra, Papanikolopoulos (2012); Cherian, Sra (2015)]
Averaging Matrices

\[
\min_{M \succ 0} \sum_i \delta^2_R(M, A_i)
\]

\[
\min_{M \succ 0} \sum_i \delta^2_S(M, A_i)
\]

\[
\delta^2_R(X, Y) := \| \log \text{Eig}(X^{-1}Y) \|^2
\]

\[
\delta^2_S(X, Y) := \log \det \left( \frac{X + Y}{2} \right) - \frac{1}{2} \log \det(XY)
\]

nonconvex but globally solvable!

[Sra (2012, 2014)]
Non-Gaussian Models

Natural Image Statistics
- Extract 200,000 training patches from 4167 images
- 10 sets of 100,000 test patches
- Log-transform intensities; add small amount of white noise

\[ p(x) \propto \frac{(x^T \Sigma^{-1} x)^{a-d/2} e^{-\frac{a}{d} x^T \Sigma^{-1} x}}{\det(\Sigma)^{1/2}} \]

Elliptically Contoured Distributions (ECD)

[Hosseini, Sra (2015a)]
Likelihood maximization

Given observations $x_1, x_2, \ldots, x_n$ find m.l.e. by solving

\[
\frac{n}{2} \log \det(\Sigma) - \left(a - \frac{d}{2}\right) \sum_{i=1}^{n} \log(x_i^T \Sigma^{-1} x_i) + \frac{a}{d} \text{trace}(\Sigma^{-1} \sum_{i} x_i x_i^T)
\]

convex or nonconvex: often globally solvable!
Geometric Convexity

Convexity

Geodesic convexity

Convex, nonconvex, and geometric opt.

Metric spaces & curvature: [Alexandrov; Busemann; Cartan; Bridson, Häflinger; Gromov; Perelman]
Geometric Optimization

Recognizing, constructing, and optimizing geodesically convex functions

[Hauser, Sra (2013)]

Corollaries

\[ X \mapsto \log \det (B + \sum_i A_i^* X A_i) \]

\[ X \mapsto \log \text{per}(B + \sum_i A_i^* X A_i) \]

\[ \delta^2_R(X, Y), \quad \delta^2_S(X, Y) \]

(jointly g-convex)

Many more theorems and corollaries

One-D version known as: Geometric Programming

www.stanford.edu/~boyd/papers/gp_tutorial.html


[Hauser, Sra (2015)]

\[ X \#_t Y := X^{1/2} (X^{-1/2} Y X^{-1/2})^t X^{1/2} \]

\[ f(X \#_t Y) \leq (1 - t)f(X) + tf(Y) \]
Averaging Matrices

$$\min_{M \geq 0} \Phi(M) = \sum_i \delta^2_{S}(M, A_i)$$

$$\nabla \Phi(M) = 0$$

$$M^{-1} = \frac{1}{n} \sum_{i=1}^{n} \left( \frac{M + A_i}{2} \right)^{-1}$$
Averaging Matrices

\[
\min_{M > 0} \Phi(M) = \sum_i \delta^2_S(M, A_i)
\]

\[\nabla \Phi(M) = 0\]

\[
M_{k+1}^{-1} = \frac{1}{n} \sum_{i=1}^{n} \left( \frac{M_k + A_i}{2} \right)^{-1}
\]

Plug-and-play!

Nonlinear Perron-Frobenius fixed-point theory

[Sra (2012)]
Theorem: Iteration is a contraction in a suitable metric space

\[ \delta_T(X, Y) := \| \log(X^{-1/2} Y X^{-1/2}) \|_\infty \]

Key properties of this metric (see [Sra, Hosseini SIOPT'15] for details)

\[
\begin{align*}
\delta_T(X^{-1}, Y^{-1}) &= \delta_T(X, Y) \\
\delta_T(B^*X B, B^*Y B) &= \delta_T(X, Y), \quad B \in \text{GL}_n(\mathbb{C}) \\
\delta_T(X^t, Y^t) &\leq |t| \delta_T(X, Y), \quad \text{for } t \in [-1, 1] \\
\delta_T \left( \sum_i w_i X_i, \sum_i w_i Y_i \right) &\leq \max_{1 \leq i \leq m} \delta_T(X_i, Y_i), \quad w_i \geq 0, w \neq 0 \\
\delta_T(X + A, Y + A) &\leq \frac{\alpha}{\alpha + \beta} \delta_T(X, Y), \quad A \succeq 0,
\end{align*}
\]

Note: Contraction does not depend on geodesic convexity
Matrix Square Root

Broadly applicable

Key to ‘expm’, ‘logm’

Convex, nonconvex, and geometric opt.
Nonconvex optimization through the Euclidean lens

\[
\min_{X \in \mathbb{R}^{n \times n}} \| M - X^2 \|_F^2
\]

**Gradient descent**

\[
X_{t+1} \leftarrow X_t - \eta (X_t^2 - M)X_t - \eta X_t (X_t^2 - M)
\]

Simple(ish) algo; linear convergence; **nontrivial** analysis

[Jain, Jin, Kakade, Netrapalli; Jul. 2015]
Matrix Square Root

Nonconvex optimization thorough non-Euclidean lens

\[
\min_{X \succ 0} \delta_S^2(X, A) + \delta_S^2(X, I)
\]

**Fixed-point iteration**

\[
X_{k+1} \leftarrow [(X_k + A)^{-1} + (X_k + I)^{-1}]^{-1}
\]

Simple method; linear convergence; 1/2 page analysis!

**Global optimality thanks to geodesic convexity**

[Sra; Jul. 2015] \( \delta_S^2(X, Y) := \frac{1}{2} \log \det \left( \frac{X+Y}{2} \right) - \frac{1}{2} \log \det(XY) \)
Matrix Square Root

50 × 50 matrix $I + \beta UU^T$

$\kappa \approx 64$
Gaussian Mixture Models

\[ p_{\text{mix}}(x) := \sum_{k=1}^{K} \pi_k p_{\mathcal{N}}(x; \Sigma_k, \mu_k) \]

\[ \max \prod_i p_{\text{mix}}(x_i) \]

Expectation maximization (EM): default choice

\[ p_{\mathcal{N}}(x; \Sigma, \mu) \propto \frac{1}{\sqrt{\det(\Sigma)}} \exp \left( -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right) \]

[Hosseini, Sra (2015b)]
Gaussian Mixture Models

- **Nonconvex**: both via Euclidean and manifold view

- Recent surge of theoretical results in TCS

- Numerically: EM as default choice
  
  *(Newton, quasi-Newton, other optim. often inferior to EM for GMMs — Xu, Jordan ’96)*

**Difficulty**: Positive definiteness constraint on $\Sigma$
# Gaussian Mixture Models

<table>
<thead>
<tr>
<th>K</th>
<th>EM</th>
<th>Manifold-CG</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>17s / 29.28</td>
<td>947s / 29.28</td>
</tr>
<tr>
<td>5</td>
<td>202s / 32.07</td>
<td>5262s / 32.07</td>
</tr>
<tr>
<td>10</td>
<td>2159s / 33.05</td>
<td>17712 / 33.03</td>
</tr>
</tbody>
</table>

GMM for $d=35$

Off-the-shelf manifold optim. fails!

[www.manopt.org](http://www.manopt.org)
How To Fix: Intuition

**log-likelihood for 1 component**

\[
\max_{\mu, \Sigma \succ 0} \mathcal{L}(\mu, \Sigma) := \sum_{i=1}^{n} \log p_{\mathcal{N}}(x_i; \mu, \Sigma).
\]

Euclidean convex \textbf{not} geodesically convex
Geodesic Convexity

\[ y_i = [x_i; 1] \quad S = \begin{bmatrix} \Sigma & \mu \mu^T \\ \mu^T & \mu \end{bmatrix} \]

\[
\max_{S > 0} \hat{\mathcal{L}}(S) := \sum_{i=1}^{n} \log q_N(y_i; S),
\]

**Theorem.** The modified log-likelihood is g-convex. Local max of modified LL is local max of original.

\[ f(X\#_t Y) \leq (1 - t)f(X) + tf(Y) \]
\[ X\#_t Y := X^{\frac{1}{2}}(X^{-\frac{1}{2}}YX^{-\frac{1}{2}})^t X^{\frac{1}{2}} \]

[Hosseini, Sra (2015b)] [Sra, Hosseini (2015)]
### Numerical Results

<table>
<thead>
<tr>
<th>$K$</th>
<th>EM</th>
<th>Manifold-CG</th>
<th>Reparam-LBFGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>17s / 29.28</td>
<td>947s / 29.28</td>
<td><strong>14s</strong> / 29.28</td>
</tr>
<tr>
<td>5</td>
<td>202s / 32.07</td>
<td>5262s / 32.07</td>
<td><strong>117s</strong> / 32.07</td>
</tr>
<tr>
<td>10</td>
<td>2159s / 33.05</td>
<td>17712 / 33.03</td>
<td><strong>658s</strong> / 33.06</td>
</tr>
</tbody>
</table>

GMM, $d=35$; convergence tol = 1E-5

Many more results in: [Hosseini, Sra (2015b); arXiv: 1506.07677]
Gaussian Mixture Models

Key ingredients

1. L-BFGS on the manifold
2. Careful line-search procedure

Toolboxes at:
suvrit.de/work/soft/gopt.html
github.com/utvisionlab/mixest

[Sra, Hosseini (2015); Hosseini, Sra (2015b)]
Many More Connections!

- Fundamental theory, duality, etc.
- Machine learning
- Deep learning
- Signal processing
- Engineering (EE, Aero, etc.)
- Brain-Computer interfaces
- Quantum Information Theory
- Geometry of tree-space
- Hyperbolic cones, graphs, spaces
- Nonlinear Perron-Frobenius Theory
- Matrix analysis, algebra

http://suvrit.de/gopt.html
Convex, nonconvex, and geometric opt.
Suvrit Sra (MIT)

See Springer Encyclopedia on Optimization (over 4500 pages!)
Convex relaxations of nonconvex problems (SDP relaxations, SOS, etc.)
Algorithms (trust-region methods, cutting plane techniques, bundle methods, active-set methods, and 100s of others)
Applications
Software, Systems
Parallel and distributed algorithms
Theory: convex analysis, geometry, probability
Polynomials, sums-of-squares, noncommutative polynomials
Infinite dimensional optimization
Discrete optimization, including submodular minimization and maximization
Multi-stage stochastic programming,
Optimizing with probabilistic (chance) constraints
Robust optimization
Algorithms and theory details for optimization on manifolds
Optimization in geodesic metric spaces
And 100s of other things!
Thank you