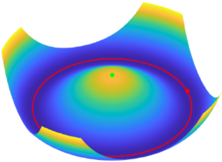
Computational Principles for High-dim Data Analysis

(Lecture Eighteen)

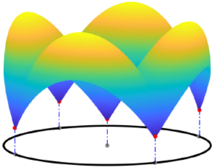
Yi Ma

EECS Department, UC Berkeley

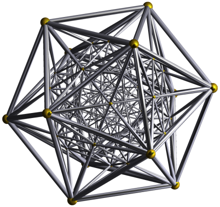
November 2, 2021



Rotational symmetry



Discrete symmetry



Nonconvex Optimization for High-Dim Problems

Power Iteration and Fixed Point

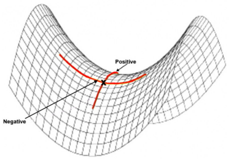
1. Power Iteration is Everywhere
2. Singular Vectors as Fixed Point
3. Complete Dictionary Learning as Fixed Point
4. Generalized Power Iteration as Fixed Point
5. Fixed Point of a Contracting Mapping

"Truth is ever to be found in the simplicity, and not in the multiplicity and confusion of things."

– Isaac Newton

Negative Curvature and Newton Descent

Consider a nonconvex program:



$$\min_{\boldsymbol{x}} f(\boldsymbol{x}).$$

Negative curvature descent: compute $\boldsymbol{e}_k$ satisfying $\boldsymbol{A}\boldsymbol{e}_k = \lambda_{\max}(\boldsymbol{A})\boldsymbol{e}_k$ with $\boldsymbol{A} \doteq \boldsymbol{I} - L_1^{-1}\nabla^2 f(\boldsymbol{x}_k) \succ \boldsymbol{0}$ by power iteration:

$$\hat{\lambda}_{i+1} = \frac{\langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{x}\rangle}{\langle \boldsymbol{x}, \boldsymbol{x}\rangle}, \quad \boldsymbol{x} = \boldsymbol{A}^i \boldsymbol{b}, \quad i = 1, 2, \dots \tag{1}$$

Newton descent: compute descent $\boldsymbol{s}_k$ from

$$\boldsymbol{s}_k = \arg\min_{\boldsymbol{s}} f(\boldsymbol{x}_k) + \langle \nabla f(\boldsymbol{x}_k), \boldsymbol{s}\rangle + \frac{1}{2}\boldsymbol{s}^* \nabla^2 f(\boldsymbol{x}_k)\boldsymbol{s} + \frac{\lambda}{2}\|\boldsymbol{s}\|_2^2 \tag{2}$$

$$= -[\nabla^2 f(\boldsymbol{x}_k) + \lambda \boldsymbol{I}]^{-1}\nabla f(\boldsymbol{x}_k). \tag{3}$$

Negative Curvature and Newton Descent

Function class: f nonconvex and $\nabla f/\nabla^2 f$ Lips. continuous with L_1/L_2.

The oracle: gradient $\nabla f(\boldsymbol{x})$ and $\nabla^2 f(\boldsymbol{x})$ (to be approximated).

Hybrid gradient and negative curvature descent:

- **if** $-\lambda_k(\nabla^2 f(\boldsymbol{x})) \geq \epsilon_H = \left(3L_2^2\epsilon\right)^{1/3}$, **then** $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \frac{2\lambda_k}{L_2}\boldsymbol{e}_k$;
- **else if** $\|\nabla f(\boldsymbol{x}_k)\|_2 \geq \epsilon_g = 3^{8/3}L_2^{1/3}\epsilon^{2/3}/2$, **then** $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \gamma_k \boldsymbol{s}_k$.

Theorem

Assume $\{\boldsymbol{x}_k\}$ are generated by the hybrid negative curvature and Newton descent. Then in at most

$$k \leq \frac{f(\boldsymbol{x}_0) - f(\boldsymbol{x}_\star)}{\epsilon} \tag{4}$$

iterations, $\boldsymbol{x}_k$ will be an an approximate second-order stationary point such that $\|\nabla f(\boldsymbol{x}_k)\|_2 \leq \epsilon_g, \lambda_{\min}(\nabla^2 f(\boldsymbol{x}_k)) \geq -\epsilon_H$.

Compute Negative Curvature: the Power Iteration

Need to compute negative curvature direction $\boldsymbol{e}_k$ without Hessian: $\boldsymbol{H} \doteq \nabla^2 f(\boldsymbol{x})$:

$$\boldsymbol{H}\boldsymbol{e} = \lambda_{\min}(\boldsymbol{H})\boldsymbol{e} \quad \text{or} \quad \boldsymbol{A}\boldsymbol{e} = \lambda_{\max}(\boldsymbol{A})\boldsymbol{e}, \quad \text{with } \boldsymbol{A} \doteq \boldsymbol{I} - L_1^{-1}\boldsymbol{H} \succ \boldsymbol{0}.$$

Power iteration:

$$\hat{\lambda}_{i+1} = \frac{\langle \boldsymbol{A}\boldsymbol{x}, \boldsymbol{x} \rangle}{\langle \boldsymbol{x}, \boldsymbol{x} \rangle}, \quad \boldsymbol{x} = \boldsymbol{A}^i\boldsymbol{b}, \quad i = 1, 2, \ldots,$$

where $\boldsymbol{A}^i\boldsymbol{b}$ can be approximated for a small $t > 0$ with:

$$\boldsymbol{A}\boldsymbol{b} = \left[\boldsymbol{I} - L_1^{-1}\boldsymbol{H}\right]\boldsymbol{b} \;\approx\; \boldsymbol{b} - (tL_1)^{-1}\big(\nabla f(\boldsymbol{x} + t\boldsymbol{b}) - \nabla f(\boldsymbol{x})\big).$$

Two gradient evaluations per power iteration.

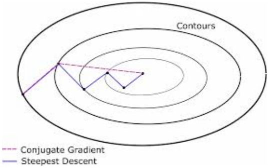
Conjugate Gradient Descent

Need to compute $\boldsymbol{s}_k$ without knowing $\boldsymbol{H} = \nabla^2 f(\boldsymbol{x})$. Notice that, similar to $\boldsymbol{e}_k$, to find $\boldsymbol{s}_k$ we need solve: $\underbrace{[\boldsymbol{H} + \lambda \boldsymbol{I}]}_{\boldsymbol{A}} \boldsymbol{s}_k = \underbrace{-\nabla f(\boldsymbol{x}_k)}_{\boldsymbol{y}}$.

A special case of the quadratic minimization problem: $\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}\|_2^2$.

Conjugate gradient descent:[1] Initialize the residual $\boldsymbol{r}_i$ and descent direction $\boldsymbol{d}_i$ as: $\boldsymbol{d}_0 = \boldsymbol{r}_0 = \boldsymbol{y} - \boldsymbol{A}\boldsymbol{x}_0$. Then for $i = 0, 1, 2, \ldots$:

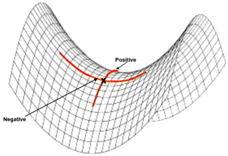
Conjugate Gradient: $$\begin{cases} \alpha_i = \dfrac{\boldsymbol{r}_i^* \boldsymbol{r}_i}{\boldsymbol{d}_i^* \boldsymbol{A} \boldsymbol{d}_i}, \\ \boldsymbol{x}_{i+1} = \boldsymbol{x}_i + \alpha_i \boldsymbol{d}_i, \\ \boldsymbol{r}_{i+1} = \boldsymbol{r}_i - \alpha_i \boldsymbol{A}\boldsymbol{d}_i, \\ \beta_{i+1} = \dfrac{\boldsymbol{r}_{i+1}^* \boldsymbol{r}_{i+1}}{\boldsymbol{r}_i^* \boldsymbol{r}_i}, \\ \boldsymbol{d}_{i+1} = \boldsymbol{r}_{i+1} + \beta_{i+1} \boldsymbol{d}_i. \end{cases}$$



[1] An introduction to the conjugate gradient method without the agonizing pain, Jonathan Shewchuk, Technical report, Carnegie Mellon University, 1994.

Effect of Noisy Gradient around a Saddle Point

Consider a standard quadratic function:
$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^*\boldsymbol{H}\boldsymbol{x}$ for a constant $\boldsymbol{H} \in \mathbb{R}^{n\times n}$,
with the smallest eigenvalue $\lambda_{\min} < 0$,
and the Lipschitz constant $L_1 = \max_i |\lambda_i(\boldsymbol{H})|$.



The Langevin dynamics is:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{x}_k - \frac{1}{L_1}\nabla f(\boldsymbol{x}_k) + \sqrt{2\lambda/L_1}\boldsymbol{n}_k \\ &= \underbrace{(\boldsymbol{I} - L_1^{-1}\boldsymbol{H})}_{\boldsymbol{A}}\boldsymbol{x}_k + \underbrace{\sqrt{2\lambda/L_1}}_{b}\boldsymbol{n}_k. \end{aligned} \tag{5}$$

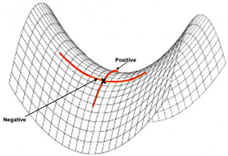
Since $\lambda_{\max}(\boldsymbol{A}) = 1 - \lambda_{\min}(\boldsymbol{H})/L_1 > 1$, this is **an unstable linear dynamic system** with random noise as the input:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + b\,\boldsymbol{n}_k. \tag{6}$$

Escaping Saddle Point

Therefore, the accumulated dynamics:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}^{k+1}\boldsymbol{x}_0 + b\sum_{i=0}^{k}\boldsymbol{A}^{k-i}\boldsymbol{n}_i. \tag{7}$$



$\boldsymbol{A}^{k+1}\boldsymbol{x}_0$ and $\boldsymbol{A}^{k-i}\boldsymbol{n}_i$ are **powers** of the matrix $\boldsymbol{A}$ applied to random vectors (assuming $\boldsymbol{x}_0$ random too).

Question: which direction survives in power iteration?

Proposition (Escaping Saddle Point via Noisy Gradient Descent)

Consider the noisy gradient descent via the Langevin dynamics (5) *for the function* $f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^*\boldsymbol{H}\boldsymbol{x}$*, starting from* $\boldsymbol{x}_0 \sim \mathcal{N}(\boldsymbol{0}, \sigma^2\boldsymbol{I})$*. Then after* $k \geq \frac{\log n - \log(|\lambda_{\min}|/L_1)}{2\log(1+|\lambda_{\min}|/L_1)}$ *steps, we have*

$$\mathbb{E}[f(\boldsymbol{x}_{k+1}) - f(\boldsymbol{x}_0)] \leq -\lambda. \tag{8}$$

Power Iteration and Fixed-Point Style Algorithms

- **PCA**
  - Optimization: $$\max_{w \in \mathbb{S}^{n-1}} \varphi(w) \doteq \frac{1}{2} \|\boldsymbol{w}^* \boldsymbol{Y}\|_2^2$$
  - Algorithm: $$\boldsymbol{w}_{t+1} = \mathcal{P}_{\mathbb{S}^{n-1}}[\nabla_{\boldsymbol{w}} \varphi(\boldsymbol{w}_t)] = \frac{\boldsymbol{Y}\boldsymbol{Y}^* \boldsymbol{w}_t}{\|\boldsymbol{Y}\boldsymbol{Y}^* \boldsymbol{w}_t\|_2}$$
- **ICA**
  - Optimization: $$\max_{\boldsymbol{w} \in \mathbb{S}^{n-1}} \psi(\boldsymbol{w}) \doteq \frac{1}{4}\mathrm{kurt}[\boldsymbol{w}^* \boldsymbol{y}] = \frac{1}{4}\mathbb{E}\left[\boldsymbol{w}^* \boldsymbol{y}\right]^4 - \frac{3}{4}\|\boldsymbol{w}\|_2^4$$
  - Algorithm: $$\boldsymbol{w}_{t+1} = \mathcal{P}_{\mathbb{S}^{n-1}}\left[\nabla_{\boldsymbol{w}} \psi(\boldsymbol{w}_t)\right] = \frac{\mathbb{E}\left[\boldsymbol{y}\left(\boldsymbol{y}^* \boldsymbol{w}_t\right)^3\right] - 3\|\boldsymbol{w}_t\|_2^2 \boldsymbol{w}_t}{\left\|\mathbb{E}\left[\boldsymbol{y}\left(\boldsymbol{y}^* \boldsymbol{w}_t\right)^3\right] - 3\|\boldsymbol{w}_t\|_2^2 \boldsymbol{w}_t\right\|_2}$$
- **DL**
  - Optimization: $$\max_{\boldsymbol{W} \in \mathsf{St}(k,n;\mathbb{R})} \phi(\boldsymbol{W}) \doteq \frac{1}{4}\|\boldsymbol{W}^* \boldsymbol{Y}\|_4^4$$
  - Algorithm: $$\boldsymbol{W}_{t+1} = \mathcal{P}_{\mathsf{St}(k,n;\mathbb{R})}\left[\nabla_{\boldsymbol{W}} \phi(\boldsymbol{W}_t)\right] = \boldsymbol{U}_t \boldsymbol{V}_t^*,$$

  where $\boldsymbol{U}_t \boldsymbol{\Sigma}_t \boldsymbol{V}_t^* = \mathsf{SVD}[\boldsymbol{Y}(\boldsymbol{Y}^* \boldsymbol{W})^{\circ 3}]$.

Singular Vectors via Nonconvex Optimization

To compute a singular vector of $\boldsymbol{Y}$, say $\boldsymbol{u}_1$, solve the eigenvector of $\boldsymbol{\Gamma} \doteq \boldsymbol{Y}\boldsymbol{Y}^*$:

$$\min \varphi(\boldsymbol{q}) \;\equiv\; -\tfrac{1}{2}\boldsymbol{q}^*\boldsymbol{\Gamma}\boldsymbol{q} \quad \text{s.t.} \quad \|\boldsymbol{q}\|_2^2 = 1 \tag{9}$$

Consider the Lagrangian formulation:

$$\mathcal{L}(\boldsymbol{q}, \lambda) = \varphi(\boldsymbol{q}) + \lambda(\|\boldsymbol{q}\|_2^2 - 1). \tag{10}$$

From the optimality condition $\nabla_{\boldsymbol{q}}\mathcal{L}(\boldsymbol{q}, \lambda) = 0$:

$$\nabla\varphi(\boldsymbol{q}) = \boldsymbol{\Gamma}\boldsymbol{q} = 2\lambda\boldsymbol{q} \quad \text{for some } \lambda. \tag{11}$$

The critical points are precisely the eigenvectors $\pm\boldsymbol{u}_i$ of $\boldsymbol{\Gamma}$:

All $\pm u_i$ are unstable critical points of φ over $\mathbb{S}^{n-1}$ except $\pm u_1$!

Fixed Point Interpretation and Power Iteration

Any critical point, including the optimal solution, is a "fixed point" to the following equation:

$$\boldsymbol{q} = \mathcal{P}_{\mathbb{S}^{n-1}}(\boldsymbol{\Gamma}\boldsymbol{q}) = \frac{\boldsymbol{\Gamma}\boldsymbol{q}}{\|\boldsymbol{\Gamma}\boldsymbol{q}\|_2}, \tag{12}$$

where $\mathcal{P}_{\mathbb{S}^{n-1}}$ means projection onto the sphere $\mathbb{S}^{n-1}$. The map:

$$g(\cdot) \doteq \mathcal{P}_{\mathbb{S}^{n-1}}[\boldsymbol{\Gamma}(\cdot)] : \mathbb{S}^{n-1} \to \mathbb{S}^{n-1}$$

is actually a **contracting map** from $\mathbb{S}^{n-1}$ to $\mathbb{S}^{n-1}$:

$$d(g(\boldsymbol{q}), g(\boldsymbol{p})) \leq \rho \cdot d(\boldsymbol{q}, \boldsymbol{p})$$

for some $0 < \rho \leq \lambda_2/\lambda_1 < 1$ and $d(\cdot, \cdot)$ a natural distance on the sphere. Hence the power iteration:

$$\boldsymbol{q}_{k+1} = g(\boldsymbol{q}_k) = \frac{\boldsymbol{\Gamma}\boldsymbol{q}_k}{\|\boldsymbol{\Gamma}\boldsymbol{q}_k\|_2} \in \mathbb{S}^{n-1}. \tag{13}$$

Contracting Map

Proposition

Let $\mathbf{\Gamma} \in \mathbb{R}^{n\times n}$ be a matrix with left eigenvalue-eigenvector pairs $(\lambda_1, \boldsymbol{u}_1), \ldots, (\lambda_n, \boldsymbol{u}_n)$ such that $\lambda_1 > \lambda_2 \geq \cdots \geq \lambda_n$. Then the power iteration is contracting under the metric: $d(\boldsymbol{x}, \boldsymbol{y}) \doteq \left\| \frac{\boldsymbol{x}}{\langle \boldsymbol{x}, \boldsymbol{u}_1\rangle} - \frac{\boldsymbol{y}}{\langle \boldsymbol{y}, \boldsymbol{u}_1\rangle} \right\|_2$ with contraction constant λ_2/λ_1 for all $\boldsymbol{x}, \boldsymbol{y} \perp \boldsymbol{u}_1$: $d(g(\boldsymbol{x}), g(\boldsymbol{y})) \leq \frac{\lambda_2}{\lambda_2} d(\boldsymbol{x}, \boldsymbol{y})$.

Proof. $\forall \boldsymbol{x}$, we have $\langle \mathbf{\Gamma}\boldsymbol{x}, \boldsymbol{u}_1\rangle = \langle \boldsymbol{x}, \mathbf{\Gamma}^*\boldsymbol{u}_1\rangle = \lambda_1\langle \boldsymbol{x}, \boldsymbol{u}_1\rangle$. So we have:

$$
\begin{aligned}
d(g(\boldsymbol{x}), g(\boldsymbol{y})) &= \left\| \frac{\mathbf{\Gamma}\boldsymbol{x}}{\langle \mathbf{\Gamma}\boldsymbol{x}, \boldsymbol{u}_1\rangle} - \frac{\mathbf{\Gamma}\boldsymbol{y}}{\langle \mathbf{\Gamma}\boldsymbol{y}, \boldsymbol{u}_1\rangle} \right\|_2 \\
&= \frac{1}{\lambda_1} \left\| \mathbf{\Gamma}\left(\frac{\boldsymbol{x}}{\langle \boldsymbol{x}, \boldsymbol{u}_1\rangle} - \frac{\boldsymbol{y}}{\langle \boldsymbol{y}, \boldsymbol{u}_1\rangle} \right) \right\|_2 \\
&\leq \frac{\lambda_2}{\lambda_1} \left\| \frac{\boldsymbol{x}}{\langle \boldsymbol{x}, \boldsymbol{u}_1\rangle} - \frac{\boldsymbol{y}}{\langle \boldsymbol{y}, \boldsymbol{u}_1\rangle} \right\|_2 = \frac{\lambda_2}{\lambda_1} d(\boldsymbol{x}, \boldsymbol{y}).
\end{aligned}
$$

□

The sequence q_k converges linearly to a unique fixed point $q_\star = \boldsymbol{u}_1$.

Complete Dictionary Learning

Given a data matrix $\boldsymbol{Y} = \boldsymbol{D}_o \boldsymbol{X}_o$ where $\boldsymbol{D}_o$ is orthogonal and $\boldsymbol{X}_o$ is sparse, try to solve the following optimization problem:

$$\min_{\boldsymbol{A}} \psi(\boldsymbol{A}) \;\equiv\; -\frac{1}{4}\|\boldsymbol{A}\boldsymbol{Y}\|_4^4, \quad \text{subject to} \quad \boldsymbol{A}^*\boldsymbol{A} = \boldsymbol{I}. \tag{14}$$

Consider the Lagrangian:

$$\mathcal{L}(\boldsymbol{A}, \boldsymbol{\Lambda}) \doteq -\frac{1}{4}\|\boldsymbol{A}\boldsymbol{Y}\|_4^4 + \langle \boldsymbol{\Lambda}, \boldsymbol{A}^*\boldsymbol{A} - \boldsymbol{I} \rangle. \tag{15}$$

This gives the necessary condition $\nabla_{\boldsymbol{A}} \mathcal{L}(\boldsymbol{A}, \boldsymbol{\Lambda}) = \boldsymbol{0}$:

$$-\nabla_{\boldsymbol{A}} \psi(\boldsymbol{A}) = (\boldsymbol{A}\boldsymbol{Y})^{\circ 3}\boldsymbol{Y}^* = \boldsymbol{A}\boldsymbol{S}, \tag{16}$$

for a symmetric matrix $\boldsymbol{S} = (\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^*)$ (of Lagrange multipliers).

Fixed Point Interpretation

For an orthogonal $\boldsymbol{A}$ and symmetric $\boldsymbol{S}$, we have:$\mathcal{P}_{\mathsf{O}(n)}[\boldsymbol{A}\boldsymbol{S}] = \boldsymbol{A}$. (Why?)

By projecting both sides of (16) onto the orthogonal group $\mathsf{O}(n)$:

$$\boldsymbol{A} = \mathcal{P}_{\mathsf{O}(n)}[(\boldsymbol{A}\boldsymbol{Y})^{\circ 3}\boldsymbol{Y}^*]. \tag{17}$$

Consider the map from $\mathsf{O}(n)$ to $\mathsf{O}(n)$:

$$g(\cdot) \doteq \mathcal{P}_{\mathsf{O}(n)}[((\cdot)\boldsymbol{Y})^{\circ 3}\boldsymbol{Y}^*] : \mathsf{O}(n) \to \mathsf{O}(n)$$

The optimal solutions $\boldsymbol{A}_\star$ is a "fixed point" of the map $g(\cdot)$. This gives the *matching, stretching, and projection* algorithm for dictionary learning:

$$\boldsymbol{A}_{k+1} = \mathcal{P}_{\mathsf{O}(n)}[(\boldsymbol{A}_k\boldsymbol{Y})^{\circ 3}\boldsymbol{Y}^*]. \tag{18}$$

The sequence $\boldsymbol{A}_k$ converges locally to $\boldsymbol{A}_\star$ with a cubic rate.

Minimizing a Concave Function on a Stiefel Manifold

Consider a concave function $f(\boldsymbol{X})$ over the Stiefel Manifold:

$$\mathsf{V}_m(\mathbb{R}^n) \doteq \{\boldsymbol{X} \in \mathbb{R}^{n\times m} \mid \boldsymbol{X}^*\boldsymbol{X} = \boldsymbol{I}_{m\times m}\}.$$

Then for the program:

$$\min_{\boldsymbol{X}} f(\boldsymbol{X}) \quad \text{subject to} \quad \boldsymbol{X}^*\boldsymbol{X} = \boldsymbol{I}, \tag{19}$$

we consider the Lagrangian:

$$\mathcal{L}(\boldsymbol{X}, \boldsymbol{\Lambda}) \doteq f(\boldsymbol{X}) + \langle \boldsymbol{\Lambda}, \boldsymbol{X}^*\boldsymbol{X} - \boldsymbol{I}\rangle. \tag{20}$$

The necessary condition for optimality $\nabla_{\boldsymbol{X}}\mathcal{L}(\boldsymbol{X}, \boldsymbol{\Lambda}) = \mathbf{0}$ gives

$$-\nabla f(\boldsymbol{X}) = \boldsymbol{X}\boldsymbol{S}, \tag{21}$$

for a symmetric matrix $\boldsymbol{S} = (\boldsymbol{\Lambda} + \boldsymbol{\Lambda}^*)$.

Generalized Power Iteration

Since $\boldsymbol{X}^*\boldsymbol{X} = \boldsymbol{I}$, this gives $\nabla f(\boldsymbol{X})^* \nabla f(\boldsymbol{X}) = \boldsymbol{S}^*\boldsymbol{X}^*\boldsymbol{X}\boldsymbol{S} = \boldsymbol{S}^2$ hence $\boldsymbol{S} = [\nabla f(\boldsymbol{X})^* \nabla f(\boldsymbol{X})]^{1/2}$. When $\boldsymbol{S}$ is invertible, the necessary condition (21) for optimality becomes:

$$\boldsymbol{X} = -\nabla f(\boldsymbol{X})[\nabla f(\boldsymbol{X})^* \nabla f(\boldsymbol{X})]^{-1/2}. \tag{22}$$

This gives a mapping from $\mathsf{V}_m(\mathbb{R}^n)$ to itself:

$$g(\boldsymbol{X}) \doteq -\nabla f(\boldsymbol{X})[\nabla f(\boldsymbol{X})^* \nabla f(\boldsymbol{X})]^{-1/2} : \mathsf{V}_m(\mathbb{R}^n) \to \mathsf{V}_m(\mathbb{R}^n). \tag{23}$$

The optimal fixed point solution can be computed with the iteration:

$$\boldsymbol{X}_{k+1} = g(\boldsymbol{X}_k) = -\nabla f(\boldsymbol{X}_k)[\nabla f(\boldsymbol{X}_k)^* \nabla f(\boldsymbol{X}_k)]^{-1/2}. \tag{24}$$

$\boldsymbol{X}_k$ converges to a critical point with a rate $O(1/k)$.[2]

[2] Generalized power method for sparse principal component analysis, M. Journee, Y. Nesterov, P. Richtarik, and R. Sepulchre, Journal of Machine Learning Research, 2010.

Fixed Point of a Contracting Mapping

let $\mathcal{M}$ be a compact smooth manifold with a distance metric $d(\cdot, \cdot)$.

Definition (Contraction Mapping)

A map $g : \mathcal{M} \to \mathcal{M}$ is called a contraction mapping on $\mathcal{M}$ if there exists $\rho \in (0, 1)$ such that $d(g(\boldsymbol{x}), g(\boldsymbol{y})) \leq \rho \cdot d(\boldsymbol{x}, \boldsymbol{y})$ for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{M}$.

Theorem (Banach-Caccioppoli Fixed Point)

Let $(\mathcal{M}, d)$ be a complete metric space with a contraction mapping: $g : \mathcal{M} \to \mathcal{M}$. Then g has a unique fixed point $\boldsymbol{x}_\star \in \mathcal{M}$: $g(\boldsymbol{x}_\star) = \boldsymbol{x}_\star$.

The unique fixed point $\boldsymbol{x}_\star$ can be found through iteration:

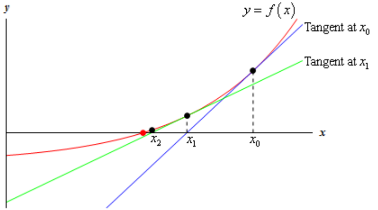
$$\boldsymbol{x}_{k+1} \leftarrow g(\boldsymbol{x}_k), \quad k = 0, 1, \ldots$$

with $\boldsymbol{x}_k \to \boldsymbol{x}_\star$ at least geometrically.

Back to the Origin

Newton's Method: finding the zero $x_\star$ of a function $f(x)$ such that $f(x_\star) = 0$ as a fixed point to the mapping:

$$g(x) \doteq x - \frac{f(x)}{f'(x)}. \qquad (25)$$



The Newton iteration is just:

$$x_{k+1} = g(x_k) = x_k - \frac{f(x_k)}{f'(x_k)}. \qquad (26)$$

Applying to $\min f(x)$ or equivalently solving $f'(x) = 0$ leads to Newton descent!

Assignments

- Reading: Section 9.6 of Chapter 9.
- Written Homework #4.