Computational Intelligence Laboratory

Lecture 3

Matrix Decomposition and Optimization

Thomas Hofmann

ETH Zurich – cil.inf.ethz.ch

Credit for slides and figures on optimization:
Martin Jaggi @EPFL

March 17, 2017

Section 1

Matrix Decomposition – Reloaded

Beyond Singular Value Decomposition

- Is SVD the final answer for (low-rank) matrix decomposition?
  - **Eckart-Young theorem** guarantees:

$$\mathbf{A}_k = \underset{\text{rank}(\mathbf{B})=k}{\arg\min} \|\mathbf{A} - \mathbf{B}\|_F^2$$

  - surprising: not a convex optimization problem!
  - convex combination of k-rank matrices is generally not rank k

$$\frac{1}{2}\underbrace{\begin{bmatrix}1 & 0\\ 0 & 0\end{bmatrix}}_{\text{rank } 1} + \frac{1}{2}\underbrace{\begin{bmatrix}0 & 0\\ 0 & 1\end{bmatrix}}_{\text{rank } 1} = \frac{1}{2}\underbrace{\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}}_{\text{rank } 2}$$

Beyond Singular Value Decomposition

- Problem: entries which are **unobserved** – not zero!
  - should optimize

$$\min_{\text{rank}(\mathbf{B})=k} \left[\sum_{(i,j)\in\mathcal{I}} (a_{ij} - b_{ij})^2 \right], \quad \mathcal{I} = \{(i,j) : \text{observed}\}$$

  - instead of

$$\min_{\text{rank}(\mathbf{B})=k} \left[\sum_{i,j} (a_{ij} - b_{ij})^2 \right] = \min_{\text{rank}(\mathbf{B})=k} \|\mathbf{A} - \mathbf{B}\|_F^2$$

    - usually: mean zero $a_{ij} \leftarrow a_{ij} - \frac{1}{|\mathcal{I}|} \sum_{\mathcal{I}} a_{ij}$

Matrix Factorization: Non-Convex Problem

- Singular Value Decomposition is not enough!
- **Non-convex** optimization problem
  - variant A: non-convex constraint set

$$\text{minimize over set } \{\mathbf{B} : \text{rank}(\mathbf{B}) = k\}$$

  - variant B: non-convex objective

re-parametrize $\mathbf{B} = \mathbf{U}\mathbf{V}, \quad \mathbf{U} \in \mathbb{R}^{m\times k}, \mathbf{V} \in \mathbb{R}^{k\times n}$

then $\text{rank}(\mathbf{B}) \leq k$

e.g. $f(u,v) = (a - uv)^2,\ u_1 v_1 = u_2 v_2 = a \ \wedge \ u_1 \neq u_2$

$$\implies f(u_1, v_1) = f(u_2, v_2) = 0 \wedge f\left(\frac{u_1 + u_2}{2}, \frac{v_1 + v_2}{2}\right) > 0$$

Alternating Minimization

- Is there something **convex** about the **non-convex** objective?

$$f(\mathbf{U}, \mathbf{V}) = \frac{1}{|\mathcal{I}|} \sum_{(i,j) \in \mathcal{I}} (a_{ij} - \langle \mathbf{u}_i, \mathbf{v}_j \rangle)^2$$

  - for fixed $\mathbf{U}$, f is convex in $\mathbf{V}$ – for fixed $\mathbf{V}$, f is convex in $\mathbf{U}$
  - ... which does not mean f is jointly convex in $\mathbf{U}$ and $\mathbf{V}$

- Idea: perform **alternating minimization**

$$\mathbf{U} \leftarrow \arg\min_{\mathbf{U}} f(\mathbf{U}, \mathbf{V})$$

$$\mathbf{V} \leftarrow \arg\min_{\mathbf{V}} f(\mathbf{U}, \mathbf{V}), \quad \text{repeat until convergence}$$

  - f is never increased and lower bounded by 0

Alternating Least Squares

- Alternating minimization for low-rank matrix factorization = **alternating least squares**
  - decompose f into subproblems for columns of $\mathbf{V}$

$$f(\mathbf{U}, \mathbf{V}) = \sum_i \underbrace{\left[\sum_{j:(i,j)\in\mathcal{I}} (a_{ij} - \langle \mathbf{u}_j, \mathbf{v}_i \rangle)^2 \right]}_{=: f(\mathbf{U}, \mathbf{v}_i)}$$

  - least squares problem $f(\mathbf{U}, \mathbf{v}_i)$ for column $\mathbf{v}_i$ of $\mathbf{V}$
    - each of which can be solved independently!
  - by symmetry: also holds for $\mathbf{U} \leftrightarrow \mathbf{V}$

Frobenius Norm Regularization

- Typically: regularize matrix factors $\mathbf{U}, \mathbf{V}$
- Frobenius norm regularizer

$$\Omega(\mathbf{U}, \mathbf{V}) = \|\mathbf{U}\|_F + \|\mathbf{V}\|_F$$

  - then

$$\text{minimize} \;\rightarrow\; f(\mathbf{U}, \mathbf{V}) + \mu\,\Omega(\mathbf{U}, \mathbf{V}), \quad \mu > 0$$

  - does not change separability structure of problem

ALS for Collaborative Filtering

- given low-dimensional representations for items
- compute for each user independently the best representation

- given low-dimensional representations for users
- compute for each item independently the best representation

- all optimization problems are small least-square problems

Matrix Decomposition as Optimization

- Is this the best we can do?
- Does this strategy always work (more factorizations to come ...)?
- We need to better understand the power of **convex optimization**!

Section 2

Unconstrained Optimization

Optimization

- General optimization problem (unconstrained minimization)

$$\begin{aligned} \text{minimize} \qquad & f(\mathbf{x}) \\ \text{with} \qquad & \mathbf{x} \in \mathbb{R}^m \end{aligned}$$

  - solutions $\mathbf{x} \in \mathbb{R}^m$ (e.g. parameters in learning)
  - objective $f : \mathbb{R}^m \to \mathbb{R}$
  - technical assumption: f is continuous and differentiable

Why? And How?

Optimization is everywhere: *machine learning, big data, statistics, data analysis of all kinds, finance, logistics, planning, control theory, mathematics, search engines, simulations, and many other applications ...*

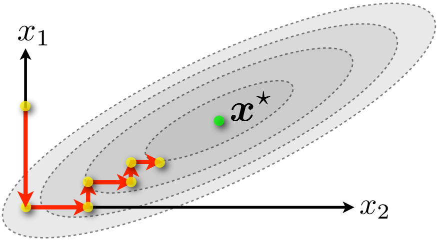
- Mathematical Modeling:
  - defining & modeling learning as optimization problems
- Computational Optimization:
  - designing & running an (appropriate) optimization algorithm

Optimization Algorithms

- Optimization at large scale: **simplicity** rules!
- Main approaches:
  - Coordinate Descent
  - Gradient Descent
  - Stochastic Gradient Descent (SGD)
- History:
  - 1950s: Linear Programming
  - 1980s: General optimization, convex optimization theory
  - 2005-today: Large scale optimization

Coordinate Descent

Goal: Find $\mathbf{x}^* = \arg\min_{\mathbf{x}} f(\mathbf{x})$.



Idea: update one coordinate at a time, keeping all others fixed.

Coordinate Descent

initialize $\mathbf{x}^{(0)} \in \mathbb{R}^m$

for t = 0,. . . ,T-1 **do**

sample coordinate $d \overset{\text{uni}}{\sim} \{1 \ldots m\}$

optimize (analytically or via line search)

$$u^* \leftarrow \arg\min_{u \in \mathbb{R}} f\left(x_1^{(t)}, \ldots, x_{d-1}^{(t)}, u, x_{d+1}^{(t)}, \ldots, x_m^{(t)}\right)$$

update

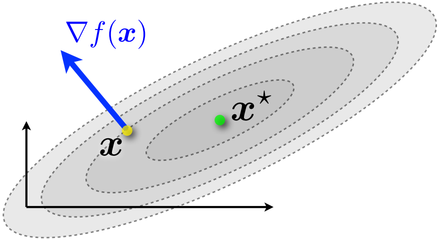
$$x_i^{(t+1)} \leftarrow \begin{cases} u^* & \text{if } i = d \\ x_i^{(t)} & \text{otherwise} \end{cases}$$

end for

Gradient

Gradient of a function $f : \Omega \subseteq \mathbb{R}^m \to \mathbb{R}$:

$$\nabla f := \begin{pmatrix} \frac{\partial f}{\partial x_1} \\ \dots \\ \frac{\partial f}{\partial x_m} \end{pmatrix}, \quad \nabla f : \Omega \to \mathbb{R}^m$$



Steepest Descent

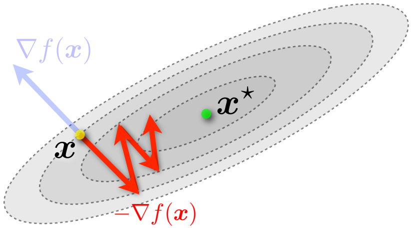
- first suggested by Cauchy in 1847
- simple to implement, scalable and robust

initialize $\mathbf{x}^{(0)} \in \mathbb{R}^m$

for t = 0:T-1 **do**

update $\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} - \eta \nabla f(\mathbf{x}^{(t)}) \quad \{\eta > 0 : \text{step size}\}$

end for



Stochastic Gradient Descent

- Empirical risk minimization: additive objective

$$\text{minimize} \qquad f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x})$$

- Stochastic Gradient Descent (SGD)

initialize $\mathbf{x}^{(0)} \in \mathbb{R}^m$

for t = 0:T-1 **do**

sample $i \overset{\text{uni}}{\sim} \{1 \dots n\}$

update $\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} - \eta \nabla f_i(\mathbf{x}^{(t)})$

end for

Stochastic Gradient Descent

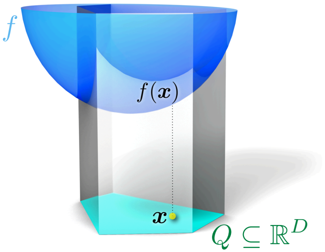
SGD update $\mathbf{x}^{(t+1)} \leftarrow \mathbf{x}^{(t)} - \eta \nabla f_i(\mathbf{x}^{(t)})$

- Idea: Cheap but unbiased estimate of the gradient
  - $\mathbf{E} \nabla f_i(\mathbf{x}) = \nabla f(\mathbf{x})$ over random choice of i
  - downside: variance = randomness in descent directions
- computing $\nabla f_i(\mathbf{x})$ is a factor n cheaper than computing $\nabla f(\mathbf{x})$
- convergence: decreasing stepsize $\eta \propto \frac{1}{t}$

Section 3

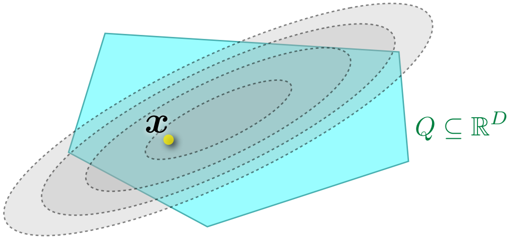
Constrained Optimization

Constrained Optimization



Constrained Optimization Problem

$$\begin{array}{ll} \text{minimize} & f(\mathbf{x}) \\ \text{subject to} & \mathbf{x} \in Q \end{array}$$



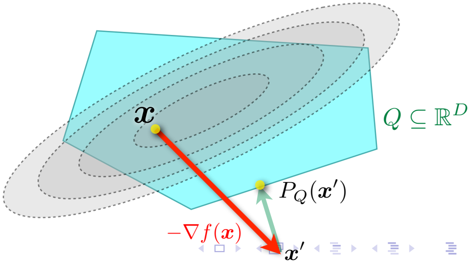
Projected Gradient Descent

Idea: project onto Q after each step:

$$P_Q(\mathbf{x}) := \arg\min_{\hat{\mathbf{x}} \in Q} \|\hat{\mathbf{x}} - \mathbf{x}\|$$

Projected gradient update

$$\mathbf{x}^{(t+1)} \leftarrow P_Q\left[\mathbf{x}^{(t)} - \eta \nabla f(\mathbf{x}^{(t)})\right]$$



Lagrangian Function

Primal optimization problem

$$
\begin{aligned}
\text{minimize} \quad & f(\mathbf{x}) \\
\text{subject to} \quad & g_i(\mathbf{x}) \leq 0, \; i = 1, \ldots, p \\
& h_j(\mathbf{x}) = 0, \; j = 1, \ldots, q
\end{aligned}
$$

Lagrangian

$$
L(\mathbf{x}, \lambda, \nu) := f(\mathbf{x}) + \sum_{i=1}^{p} \lambda_i g_i(\mathbf{x}) + \sum_{j=1}^{q} \nu_j h_j(\mathbf{x})
$$

- $\lambda_i \geq 0$, $\nu_k \in \mathbb{R}$: Lagrange multipliers

Lagrangian Dual

- Lagrange **dual function**:

$$\mathcal{D}(\lambda, \nu) := \inf_{\mathbf{x}} L(\mathbf{x}, \lambda, \nu) \quad \in \mathbb{R}$$

  - for any feasible $\mathbf{x}$: $\nu_i h_i(\mathbf{x}) = 0$ and $\lambda_j g_j(\mathbf{x}) \leq 0$
  - hence: $\mathcal{D}(\lambda, \nu) \leq f(\mathbf{x})$, $\mathbf{x}$: feasible

- Lagrange **dual problem**: best lower bound on $f(\mathbf{x}^*)$

$$(\lambda^*, \nu^*) = \operatorname*{arg\,max}_{\lambda \geq \mathbf{0},\, \nu} \mathcal{D}(\lambda, \nu)$$

$$\mathcal{D}(\lambda, \nu) \leq \mathcal{D}(\lambda^*, \nu^*) \leq f(\mathbf{x}^*) \leq f(\mathbf{x})$$

Karush-Kuhn-Tucker Conditions

- assume $\mathbf{x}^*$ is a local minimum, f, g_i, h_j continuously diff at $\mathbf{x}^*$
- ... under some regularity conditions (e.g. affine g_i and h_j)
- **KKT** conditions hold for λ^* and ν^*

$$
\begin{aligned}
&(1) \quad \nabla_{\mathbf{x}} L(\mathbf{x}^*; \lambda^*, \nu^*) = \mathbf{0} \quad \text{1st order optimality} \\
&(2) \quad g_i(\mathbf{x}^*) \leq 0,\ h_j(\mathbf{x}^*) = 0,\ \lambda_i^* \geq 0, \quad \text{feasability} \\
&(3) \quad \lambda_i^* g_i(\mathbf{x}^*) = 0 \quad \text{complementary slackness}
\end{aligned}
$$

- it is implied that $f(\mathbf{x}^*) = L(\mathbf{x}^*; \lambda^*, \nu^*)$

Strong Duality

- Duality gap:

$$\triangle = \min_{\mathbf{x}} f(\mathbf{x}) - \max_{\lambda \geq \mathbf{0},\, \nu} \mathcal{D}(\lambda, \nu) \geq 0$$

- $\triangle = 0$, if primal problem is **convex** and under additional conditions

- if $\triangle = 0$, then

$$\mathcal{D}(\lambda^*, \nu^*) = \min_{\mathbf{x}} L(\mathbf{x}, \lambda^*, \nu^*) = L(\mathbf{x}^*, \lambda^*, \nu^*) = f(\mathbf{x}^*)$$

  - can recover $\mathbf{x}^*$ by unconstrained minimization of $L(\mathbf{x}, \lambda^*, \nu^*)$
  - dual problem has very simple constraints!

Section 4

Convex Optimization

Convex Set

A set Q is *convex* if the line segment between any two points of Q lies in Q, i.e., if for any $\mathbf{x}, \mathbf{y} \in Q$ and any θ with $0 \leq \theta \leq 1$, we have

$$\theta \mathbf{x} + (1 - \theta)\mathbf{y} \in Q.$$

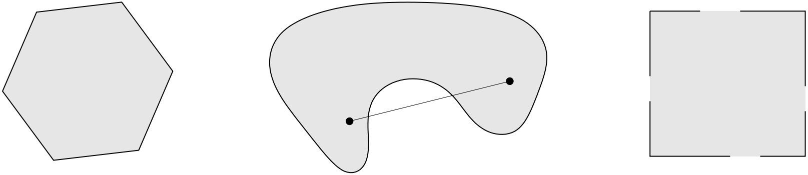


*Figure 2.2 from S. Boyd, L. Vandenberghe

Left Convex.

Middle Not convex, since line segment not in set.

Right Not convex, since some, but not all boundary points are contained in the set.

Properties of Convex Sets

- Intersections of convex sets are convex
- Projections onto convex sets are *unique*.
  (and often efficient to compute)

recall $P_Q(\mathbf{x}') := \arg\min_{\mathbf{y}\in Q} \|\mathbf{y} - \mathbf{x}'\|$

Convex Function

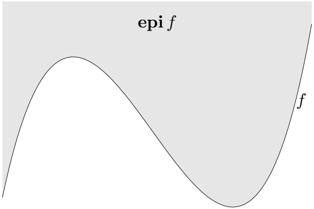
Epigraph: the *graph* of a function $f : \mathbb{R}^m \rightarrow \mathbb{R}$ is defined as

$$\{(\mathbf{x}, f(\mathbf{x})) \,|\, \mathbf{x} \in \mathrm{dom} f\},$$

The *epigraph* of a function $f : \mathbb{R}^m \rightarrow \mathbb{R}$ is defined as

$$\{(\mathbf{x}, t) \,|\, \mathbf{x} \in \mathrm{dom} f, f(\mathbf{x}) \leq t\},$$

A function is convex *iff* its epigraph is a convex set.



Convex?

*Figure 3.5 from S. Boyd, L. Vandenberghe

Convex Function

Examples of convex functions

- Linear functions: $f(\mathbf{x}) = \mathbf{a}^\top \mathbf{x}$
- Affine functions: $f(\mathbf{x}) = \mathbf{a}^\top \mathbf{x} + b$
- Exponential: $f(\mathbf{x}) = e^{\alpha \mathbf{x}}$
- Norms. Every norm on $\mathbb{R}^m$ is convex.

Convexity of a norm $f(\mathbf{x})$

- triangle inequality $f(\mathbf{x} + \mathbf{y}) \leq f(\mathbf{x}) + f(\mathbf{y})$
- homogeneity of a norm $f(a\mathbf{x}) = |a| f(\mathbf{x})$

$$f(\theta \mathbf{x} + (1-\theta)\mathbf{y}) \leq f(\theta \mathbf{x}) + f((1-\theta)\mathbf{y}) = \theta f(\mathbf{x}) + (1-\theta) f(\mathbf{y}).$$

Convex Optimization

- **Convex Optimization** problems

$$\min \ f(\mathbf{x}) \qquad \text{s.t.} \qquad \mathbf{x} \in Q$$

  where both
  - f is a convex function
  - Q is a convex set (note: $\mathbb{R}^m$ is convex)

- Properties of Convex Optimization Problems
  - **every local minimum is a global minimum**

Solving Convex Optimization Problems (provably)

- For convex optimization problems, all algorithms
  - Coordinate Descent
  - Gradient Descent
  - Stochastic Gradient Descent
  - Projected Gradient Descent (projections onto convex sets do work!)

  do converge to the global optimum! (assuming f differentiable)

- **Theorem:** For convex problems, the **convergence rate** of the above four algorithms is (at least) proportional to $\frac{1}{t}$, i.e.

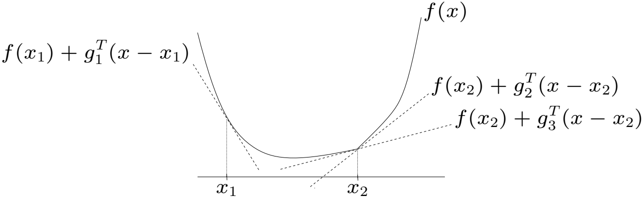
$$f(\mathbf{x}^{(t)}) - f(\mathbf{x}^*) \leq \frac{c}{t}$$

caveat: SGD rate can be $1/\sqrt{t}$ if f is not strongly convex

Sub-Gradient Descent

- What if f is not differentiable?
- Work with Sub-gradient: $\mathbf{g} \in \mathbb{R}^m$ is a **subgradient** of f at $\mathbf{x}$ if

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \mathbf{g}^\top(\mathbf{y} - \mathbf{x}) \qquad \text{for all } \mathbf{y}$$



Sub-Gradient Descent

- **Subgradient Descent**: in algorithms, replace the gradient with a subgradient.
- **Theorem:** For convex problems, the convergence rate of [plain or projected] subgradient descent is (at least) proportional to $\frac{1}{\sqrt{t}}$, i.e.

$$f(\mathbf{x}^{(t)}) - f(\mathbf{x}^*) \leq \frac{c}{\sqrt{t}}$$

Section 5

Convex Relaxation

Convex Relaxation

- For cases where Q is not convex ...
  - ... we can aim to find $Q' \supset Q$
  - ... such that Q' is convex
  - ... and then solve the **convex relaxation**

$$\min f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{x} \in Q'$$

  - ... finally we can try to find a close(st) point in Q

Convex Relaxation for Matrix Completion

- Replace non-convex rank contraint by convex norm constraint.
- Nuclear norm

$$\|\mathbf{A}\|_* = \sum_i \sigma_i, \quad \sigma_i : \text{singular values}$$

- Exact reconstruction:

$$\min_{\mathbf{B}} \|\mathbf{B}\|_*, \quad \text{subject to} \quad A_{ij} = B_{ij} \;\; \forall (i,j) \in \mathcal{I}$$

- Approximate reconstruction (unconstrained problem)

$$\min_{\mathbf{B}} \|\mathbf{B}\|_* + \lambda \sum_{(i,j)\in\mathcal{I}} (a_{ij} - b_{ij})^2$$

Compressed Sensing for Matrix Completion

- **Theorem**: exact reconstruction of a $\mathbb{R}^{m \times n}$ rank k matrix $(m \leq n)$ is possible with probability $1 - n^{-3}$, if it is strongly incoherent with parameter μ (spread of singular values) and as long as

$$|\mathcal{I}| \geq C\mu^4 k^2 n (\log n)^2, \quad \text{typically} \quad \mu = \mathbf{O}(\sqrt{\log n})$$

  - Candes, Tao: The power of convex relaxation: Near-optimal matrix completion, 2010
  - explains, why $\|\cdot\|_*$ minimization works well in practice!