CS292F Convex Optimization:
Concepts, Algorithms and Analysis

Spring 2020

Instructor: Prof. Yu-Xiang Wang

Administrative information

- Instructor: Yu-Xiang Wang
  - Office hour: No official office hour. I will stay on a bit after the class or by appointment.
- Syllabus: [link]
  - Please read carefully
- Course website: https://www.cs.ucsb.edu/~yuxiangw/classes/CS292F-2020Spring/
- Questions and Discussion: Piazza
- Homework submission: Gradescope

Lectures over Zoom

- It will be the same meeting ID throughout the quarter.
  - Join Zoom Meeting: https://ucsb.zoom.us/j/199032871
  - Meeting ID: 199 032 871
  - Password: Check your email. Please do not share.

- It will be recorded (by the instructor).
  - To make the instruction available to people who are having connectivity issues, or in a different time-zone.
  - Turning on your mic and video = agreeing on being recorded.

Access to the Homeworks

- You are provided with a link in Piazza.
- You need to log in to your UCSB G Suite to access the “homework” folder.
- **The first homework is already released!**

Course evaluation

- 80% Homeworks (a total of 4 homeworks)

- 15% Reading Notes
  - Compulsory readings of the textbook chapters / notes / papers.
  - Write a summary (>1 pages).
  - **Due at the beginning of each lecture.** Starting on Thursday!

- 5% Participation
  - Ask questions in the class

Forms of the lectures

- Slides + Whiteboard

- We hope to produce a nicely typeset scribed notes that everyone can keep.

- **Bonus 5% for signing up to scribe lectures!**
  - Limited slots, sign up early from the course website.

ourse Schedule / Scribed Notes

eek	Date	Topic	Reading	Assignment	Scribe
	30-Mar	Intro + Convex Set and Convex Function	BV Ch.1, Ch.2, Ch.3	HW1 out [pdf,data]	[Scribe 1, latex]
	1-Apr	Convex Optimization Basics	BV Ch. 4.1-4.2		
	6-Apr	Canonical problem forms	BV Ch 4.3-4.7		
	8-Apr	Gradient Descent	BV Ch 9.1-9.4		[Scribe 4, latex]
	13-Apr	Subgradient and subdifferential	Boyd's subgradient notes	HW2 out / HW1 Due	[Scribe 5, latex]
	15-Apr	Subgradient method and proximal gradient descent (Part I)	Boyd's subgradient method notes		[Scribe 6, latex]
	20-Apr	Proximal Gradient Descent (Part II)	Section 1-4 of Parikh and Boyd)		[Scribe 7, latex]
	22-Apr	Stochastic (sub)gradient methods	Section 1-5 of Boyd's SGD notes		[Scribe 8, latex]
	27-Apr	Duality	Lecture 11 and 12 of CMU 10-725	HW3 out / HW2 due.	[Scribe 9]
	29-Apr	KKT conditions and its usage	Lecture 13 and 14 of of CMU 10-725		[Scribe 10, latex]
	4-May	Newton's method	BV Ch 9 and 10		
	6-May	Interior point methods	BV Ch 11, Nesterov and Nemirovski Ch 2		
	11-May	Intro to online learning: Learning from expert advice	Hazan Ch 1		[Notes on OCO intro, latex]
	13-May	Online (Projected) Gradient Descent	Hazan Ch 3	HW4 Out / HW3 Due	
	18-May	Follow the Regularized Leader	Hazan Ch 5		
	20-May	Exponential-Concavity and Online Newton Method	Hazan Ch 4		
	25-May	No class. Memorial day.			
	27-May	Modern Stochastic Gradient Methods	[Johnson and Zhang (2013), Ghadimi and Lan (2013)		[Scribe on SVRG, latex]
	1-Jun	Alternating Direction Method of Multipliers	[Ramdas and Tibshirani, Candes et al.]	HW#4 due /td>	
	3-Jun	Conditional gradient method / Frank-Wolfe	Hazan Ch. 7, Jaggi		
	3-Jun	Not covered: Bandits.	Hazan Ch. 6		[Scribed notes for bandits, latex]

Convex Optimization
First order optimization

Duality
Second-order methods

Online Convex
Optimization

Advanced topics

What will you learn?

- Formulate problems as convex optimization problems and choose appropriate algorithms to solve these problems.
- Understand properties such as convexity, Lipschitzness, smoothness and the computational guarantees that come with these conditions.
- Learn optimality conditions and duality and use them in your research.
- Understand the connection of first order optimization and online learning.
- Know how to prove convergence bounds and analyze no-regret online learning algorithms.
- **(New to 2020 Spring)** Learn a little bit about second order algorithms and their pros and cons w.r.t. the first order.

Why focusing on First Order Methods?

- A quarter is short. The professor is lazy.
- They are arguably most useful for machine learning
  - Scalable, one pass (few passes) algorithms.
  - Information-theoretically near optimal for ML.
- Closer to the cutting edge research world
  - SGD, SDCA, SAG, SAGA, SVRG, Katyucsha, Natasha
  - Strong guarantee in machine learning with no distributional assumptions.
- Basically the only way to train deep learning models.

Cautionary notes

- The course is a PhD level course and it requires hard work!
  - Time, effort
  - A lot of math
  - Substantial homework with both math and coding

- Be ready to be out of your comfort zone

- It will be totally worth it.

Things that I expect you to know already

- Basic real analysis
- Basic multivariate calculus
- Basic linear algebra
- Basic machine learning
- Basic probability theory + tail bounds
- Familiarity with at least one of the following: Matlab, Numpy, Julia

- I will post some review materials in Piazza.

Acknowledgment

- A big part of the lectures will be based on Ryan Tibshirani’s 10-725 in Carnegie Mellon University.

- For the online learning part of it, we will mostly follow Elad Hazan’s book: Introduction to Online Convex Optimization

Optimization in Machine Learning and Statistics

Optimization problems underlie nearly everything we do in Machine Learning and Statistics. In many courses, you learn how to:

translate  into $P \; : \; \min_{x \in D} \; f(x)$

Conceptual idea *Optimization problem*

Examples of this? Examples of the contrary?

This course: how to solve P, and why this is a good skill to have

Presumably, other people have already figured out how to solve

$$P \ : \ \min_{x \in D} \ f(x)$$

So why bother? Many reasons. Here's three:

1. Different algorithms can perform better or worse for different problems P (sometimes drastically so)
2. Studying P through an optimization lens can actually give you a deeper understanding of the statistical procedure
3. Knowledge of optimization can actually help you create a new P that is even more interesting/useful

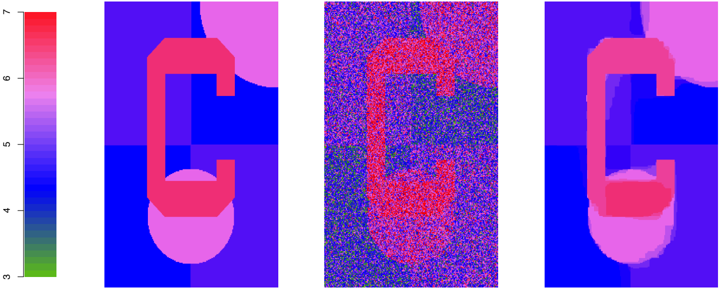
Optimization moves quickly as a field. But there is still much room for progress, especially its intersection with ML and Stats

Example: algorithms for the 2d fused lasso

The 2d fused lasso or 2d total variation denoising problem:

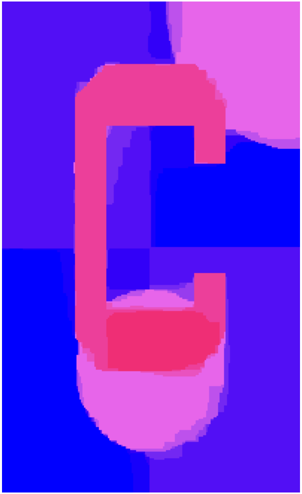
$$\min_{\theta} \ \frac{1}{2} \sum_{i=1}^{n} (y_i - \theta_i)^2 + \lambda \sum_{(i,j) \in E} |\theta_i - \theta_j|$$

This fits a piecewise constant function over an image, given data y_i, $i = 1, \ldots, n$ at pixels. Here $\lambda \geq 0$ is a tuning parameter



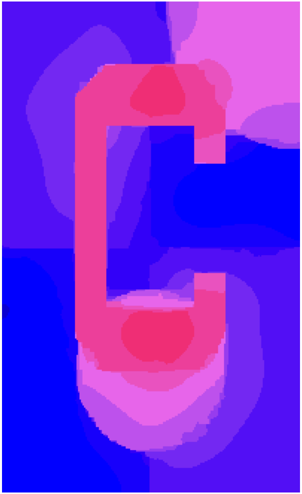
True image Data Solution

Our problem: $$\min_{\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i-\theta_i)^2+\lambda\sum_{(i,j)\in E}|\theta_i-\theta_j|$$



Specialized ADMM, 20 iterations

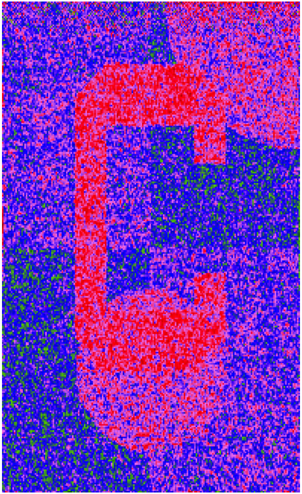
Our problem: $$\min_{\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i - \theta_i)^2 + \lambda \sum_{(i,j)\in E} |\theta_i - \theta_j|$$



Specialized ADMM, 20 iterations

Proximal gradient descent, 1000 iterations

Our problem: $$\min_{\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i - \theta_i)^2 + \lambda \sum_{(i,j)\in E} |\theta_i - \theta_j|$$

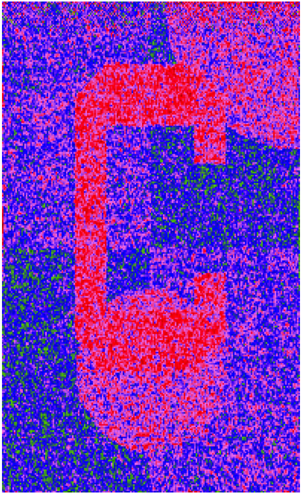


Specialized ADMM, 20 iterations

Proximal gradient descent, 1000 iterations

Coordinate descent, 10K cycles

Our problem: $$\min_{\theta} \frac{1}{2}\sum_{i=1}^{n}(y_i-\theta_i)^2+\lambda\sum_{(i,j)\in E}|\theta_i-\theta_j|$$



Specialized ADMM, 20 iterations

Proximal gradient descent, 1000 iterations

Coordinate descent, 10K cycles

(Last two from the dual)

What's the message here?

So what's the right conclusion here?

Is the alternating direction method of multipliers (ADMM) method simply a better method than proximal gradient descent, coordinate descent? ... No

In fact, different algorithms will perform better or worse in different situations. We'll learn details throughout the course

In the 2d fused lasso problem:

- Special ADMM: fast (structured subproblems)
- Proximal gradient: slow (poor conditioning)
- Coordinate descent: slow (large active set)

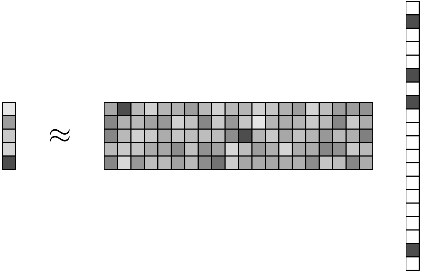
Example: sparse linear modeling

Given $y \in \mathbb{R}^n$ and a matrix $X \in \mathbb{R}^{n \times p}$, with $p \gg n$. Suppose that we know that

$$y \approx X\beta^*$$

for some unknown coefficient vector $\beta^* \in \mathbb{R}^p$. Can we generically solve for β^*? ... No!

But if β^* is known to be sparse (i.e., have many zero entries), then it's a whole different story



There are many different approaches for estimating β^*. A popular approach is to solve the lasso problem:

$$\min_{\beta \in \mathbb{R}^p} \; \frac{1}{2}\|y - X\beta\|_2^2 + \lambda\|\beta\|_1$$

Here $\lambda \geq 0$ is a tuning parameter, and $\|\beta\|_1 = \sum_{i=1}^p |\beta_i|$ denotes the ℓ_1 norm of β

There are numerous algorithms for computing a lasso solution (in fact, it can be cast as a quadratic program)

Furthermore, some key statistical insights can be derived from the Karush-Kuhn-Tucker (KKT) optimality conditions for the lasso

Lasso support recovery

The KKT conditions for the lasso problem are

$$X^T(y - X\beta) = \lambda s$$
$$s_j \in \begin{cases} \{+1\} & \beta_j > 0 \\ \{-1\} & \beta_j < 0 \\ [-1, 1] & \beta_j = 0 \end{cases}, \quad \text{for } j = 1, \ldots, p$$

We call s a subgradient of the ℓ_1 norm at β, denoted $s \in \partial\|\beta\|_1$

Under favorable conditions (low correlations in X, large nonzeros in β^*), can show that lasso solution has same support as β^*

Proof idea: plug in (shrunken version of) β^* into KKT conditions, and show that they are satisfied with high probability (primal-dual witness method of Wainwright 2009)

Widsom from Friedman (1985)

From Jerry Friedman's discussion of Peter Huber's 1985 projection pursuit paper, in Annals of Statistics:

> **A good idea poorly implemented will not work well and will likely be judged not good. It is likely that the idea of projection pursuit would have been delayed even further if working implementations of the exploratory (Friedman and Tukey, 1974) and regression (Friedman and Stuetzle, 1981) procedures had not been produced. As data analytic algorithms become more complex, this problem becomes more acute. The best way to guard against this is to become as literate as possible in algorithms, numerical methods and other aspects of software implementation. I suspect that more than a few important ideas have been discarded because a poor implementation performed badly.**

Arguably, less true today due to the advent of disciplined convex programming? Maybe, but it still rings true in large part ...

Central concept: convexity

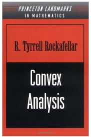
Historically, linear programs were the focus in optimization

Initially, it was thought that the important distinction was between linear and nonlinear optimization problems. But some nonlinear problems turned out to be much harder than others ...

Now it is widely recognized that the right distinction is between convex and nonconvex problems

Your supplementary textbooks for the course:

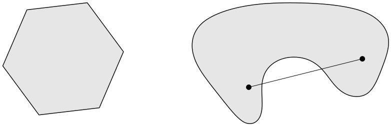
Boyd and Vandenberghe (2004) 

, Rockafellar (1970) 

Convex sets and functions

Convex set: $C \subseteq \mathbb{R}^n$ such that

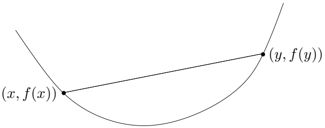
$$x, y \in C \implies tx + (1-t)y \in C \text{ for all } 0 \leq t \leq 1$$



Convex function: $f : \mathbb{R}^n \to \mathbb{R}$ such that $\mathrm{dom}(f) \subseteq \mathbb{R}^n$ convex, and

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y) \text{ for all } 0 \leq t \leq 1$$

and all $x, y \in \mathrm{dom}(f)$



Convex optimization problems

Optimization problem:

$$\begin{aligned} \min_{x \in D} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0,\ i = 1, \ldots m \\ & h_j(x) = 0,\ j = 1, \ldots r \end{aligned}$$

Here $D = \mathrm{dom}(f) \cap \bigcap_{i=1}^m \mathrm{dom}(g_i) \cap \bigcap_{j=1}^p \mathrm{dom}(h_j)$, common domain of all the functions

This is a convex optimization problem provided the functions f and $g_i, i = 1, \ldots m$ are convex, and $h_j, j = 1, \ldots p$ are affine:

$$h_j(x) = a_j^T x + b_j, \quad j = 1, \ldots p$$

Local minima are global minima

For convex optimization problems, local minima are global minima

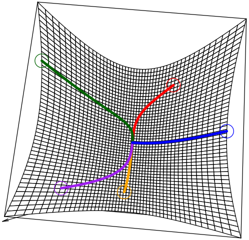
Formally, if x is feasible—$x \in D$, and satisfies all constraints—and minimizes f in a local neighborhood,

$$f(x) \le f(y) \;\text{ for all feasible } y, \; \|x - y\|_2 \le \rho,$$

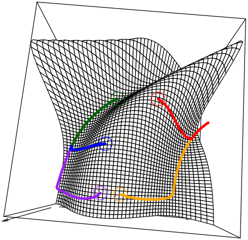
then

$$f(x) \le f(y) \;\text{ for all feasible } y$$

This is a very useful fact and will save us a lot of trouble!



Convex



Nonconvex

In summary: why convexity?

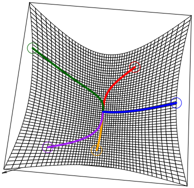
Why convexity? Simply put: because we can broadly understand and solve convex optimization problems

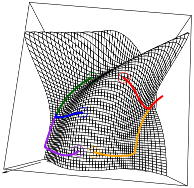
Nonconvex problems are mostly treated on a case by case basis

Reminder: a convex optimization problem is of the form

$$\begin{aligned} \min_{x \in D} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0, \ i = 1, \ldots m \\ & h_j(x) = 0, \ j = 1, \ldots r \end{aligned}$$

where f and g_i, $i = 1, \ldots m$ are all convex, and h_j, $j = 1, \ldots r$ are affine. Special property: any local minimizer is a global minimizer





Remainder of today's lecture

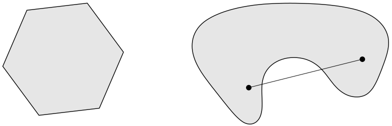
- Convex sets
- Examples
- Key properties
- Operations preserving convexity
- Same, for convex functions

Convex sets

Convex set: $C \subseteq \mathbb{R}^n$ such that

$$x, y \in C \implies tx + (1-t)y \in C \text{ for all } 0 \le t \le 1$$

In words, line segment joining any two elements lies entirely in set



Convex combination of $x_1, \ldots x_k \in \mathbb{R}^n$: any linear combination

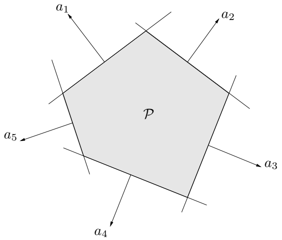
$$\theta_1 x_1 + \ldots + \theta_k x_k$$

with $\theta_i \ge 0$, $i = 1, \ldots k$, and $\sum_{i=1}^k \theta_i = 1$. Convex hull of a set C, $\mathrm{conv}(C)$, is all convex combinations of elements. Always convex

Examples of convex sets

- Trivial ones: empty set, point, line
- Norm ball: $\{x : \|x\| \leq r\}$, for given norm $\|\cdot\|$, radius r
- Hyperplane: $\{x : a^T x = b\}$, for given a, b
- Halfspace: $\{x : a^T x \leq b\}$
- Affine space: $\{x : Ax = b\}$, for given A, b

- Polyhedron: $\{x : Ax \leq b\}$, where inequality $\leq$ is interpreted componentwise. Note: the set $\{x : Ax \leq b, Cx = d\}$ is also a polyhedron (why?)



- Simplex: special case of polyhedra, given by $\mathrm{conv}\{x_0, \ldots x_k\}$, where these points are affinely independent. The canonical example is the probability simplex,

$$\mathrm{conv}\{e_1, \ldots e_n\} = \{w : w \geq 0,\ 1^T w = 1\}$$

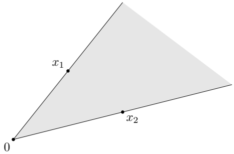
Cones

Cone: $C \subseteq \mathbb{R}^n$ such that

$$x \in C \implies tx \in C \text{ for all } t \geq 0$$

Convex cone: cone that is also convex, i.e.,

$$x_1, x_2 \in C \implies t_1 x_1 + t_2 x_2 \in C \text{ for all } t_1, t_2 \geq 0$$



Conic combination of $x_1, \ldots x_k \in \mathbb{R}^n$: any linear combination

$$\theta_1 x_1 + \ldots + \theta_k x_k$$

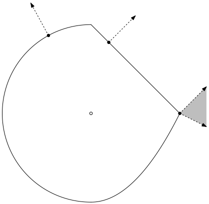
with $\theta_i \geq 0$, $i = 1, \ldots k$. Conic hull collects all conic combinations

Examples of convex cones

- Norm cone: $\{(x,t) : \|x\| \le t\}$, for a norm $\|\cdot\|$. Under the ℓ_2 norm $\|\cdot\|_2$, called second-order cone

- Normal cone: given any set C and point $x \in C$, we can define

$$\mathcal{N}_C(x) = \{g : g^T x \ge g^T y, \ \text{for all } y \in C\}$$

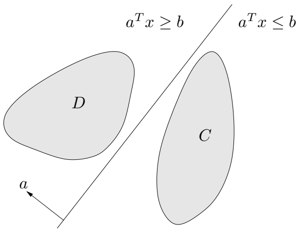


This is always a convex cone, regardless of C

- Positive semidefinite cone: $\S^n_+ = \{X \in \S^n : X \succeq 0\}$, where $X \succeq 0$ means that X is positive semidefinite (and $\S^n$ is the set of $n \times n$ symmetric matrices)

Key properties of convex sets

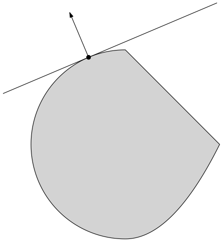
- Separating hyperplane theorem: two disjoint convex sets have a separating between hyperplane them



Formally: if C, D are nonempty convex sets with $C \cap D = \emptyset$, then there exists a, b such that

$$C \subseteq \{x : a^T x \leq b\}$$
$$D \subseteq \{x : a^T x \geq b\}$$

- Supporting hyperplane theorem: a boundary point of a convex set has a supporting hyperplane passing through it



Formally: if C is a nonempty convex set, and $x_0 \in \mathrm{bd}(C)$, then there exists a such that

$$C \subseteq \{x : a^T x \leq a^T x_0\}$$

Both of the above theorems (separating and supporting hyperplane theorems) have partial converses; see Section 2.5 of BV

Operations preserving convexity

- Intersection: the intersection of convex sets is convex
- Scaling and translation: if C is convex, then

$$aC + b = \{ax + b : x \in C\}$$

  is convex for any a, b

- Affine images and preimages: if $f(x) = Ax + b$ and C is convex then

$$f(C) = \{f(x) : x \in C\}$$

  is convex, and if D is convex then

$$f^{-1}(D) = \{x : f(x) \in D\}$$

  is convex

Example: linear matrix inequality solution set

Given $A_1, \ldots A_k, B \in \S^n$, a linear matrix inequality is of the form

$$x_1 A_1 + x_2 A_2 + \ldots + x_k A_k \preceq B$$

for a variable $x \in \mathbb{R}^k$. Let's prove the set C of points x that satisfy the above inequality is convex

Approach 1: directly verify that $x, y \in C \Rightarrow tx + (1-t)y \in C$. This follows by checking that, for any v,

$$v^T \Big(B - \sum_{i=1}^{k} (tx_i + (1-t)y_i) A_i \Big) v \geq 0$$

Approach 2: let $f : \mathbb{R}^k \to \S^n$, $f(x) = B - \sum_{i=1}^{k} x_i A_i$. Note that $C = f^{-1}(\S^n_+)$, affine preimage of convex set

More operations preserving convexity

- Perspective images and preimages: the perspective function is $P : \mathbb{R}^n \times \mathbb{R}_{++} \to \mathbb{R}^n$ (where $\mathbb{R}_{++}$ denotes positive reals),

$$P(x, z) = x/z$$

  for $z > 0$. If $C \subseteq \mathrm{dom}(P)$ is convex then so is $P(C)$, and if D is convex then so is $P^{-1}(D)$

- Linear-fractional images and preimages: the perspective map composed with an affine function,

$$f(x) = \frac{Ax + b}{c^T x + d}$$

  is called a linear-fractional function, defined on $c^T x + d > 0$. If $C \subseteq \mathrm{dom}(f)$ is convex then so if $f(C)$, and if D is convex then so is $f^{-1}(D)$

Example: conditional probability set

Let U, V be random variables over $\{1, \dots n\}$ and $\{1, \dots m\}$. Let $C \subseteq \mathbb{R}^{nm}$ be a set of joint distributions for U, V, i.e., each $p \in C$ defines joint probabilities

$$p_{ij} = \mathbb{P}(U = i, V = j)$$

Let $D \subseteq \mathbb{R}^{nm}$ contain corresponding conditional distributions, i.e., each $q \in D$ defines

$$q_{ij} = \mathbb{P}(U = i | V = j)$$

Assume C is convex. Let's prove that D is convex. Write

$$D = \left\{ q \in \mathbb{R}^{nm} : q_{ij} = \frac{p_{ij}}{\sum_{k=1}^{n} p_{kj}}, \text{ for some } p \in C \right\} = f(C)$$

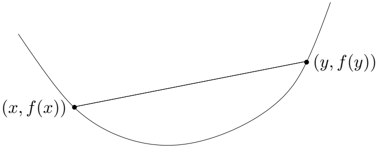
where f is a linear-fractional function, hence D is convex

Convex functions

Convex function: $f : \mathbb{R}^n \to \mathbb{R}$ such that $\mathrm{dom}(f) \subseteq \mathbb{R}^n$ convex, and

$$f(tx + (1-t)y) \le tf(x) + (1-t)f(y) \quad \text{for } 0 \le t \le 1$$

and all $x, y \in \mathrm{dom}(f)$



In words, function lies below the line segment joining $f(x), f(y)$

Concave function: opposite inequality above, so that

$$f \text{ concave} \iff -f \text{ convex}$$

Important modifiers:

- Strictly convex: $f\big(tx + (1-t)y\big) < tf(x) + (1-t)f(y)$ for $x \neq y$ and $0 < t < 1$. In words, f is convex and has greater curvature than a linear function

- Strongly convex with parameter $m > 0$: $f - \frac{m}{2}\|x\|_2^2$ is convex. In words, f is at least as convex as a quadratic function

Note: strongly convex $\Rightarrow$ strictly convex $\Rightarrow$ convex

(Analogously for concave functions)

Examples of convex functions

- Univariate functions:
  - Exponential function: e^{ax} is convex for any a over $\mathbb{R}$
  - Power function: x^a is convex for $a \geq 1$ or $a \leq 0$ over $\mathbb{R}_+$ (nonnegative reals)
  - Power function: x^a is concave for $0 \leq a \leq 1$ over $\mathbb{R}_+$
  - Logarithmic function: $\log x$ is concave over $\mathbb{R}_{++}$

- Affine function: $a^T x + b$ is both convex and concave

- Quadratic function: $\frac{1}{2}x^T Q x + b^T x + c$ is convex provided that $Q \succeq 0$ (positive semidefinite)

- Least squares loss: $\|y - Ax\|_2^2$ is always convex (since $A^T A$ is always positive semidefinite)

- Norm: $\|x\|$ is convex for any norm; e.g., ℓ_p norms,

$$\|x\|_p = \left(\sum_{i=1}^{n} x_i^p\right)^{1/p} \quad \text{for } p \geq 1, \quad \|x\|_\infty = \max_{i=1,\ldots n} |x_i|$$

and also operator (spectral) and trace (nuclear) norms,

$$\|X\|_{\mathrm{op}} = \sigma_1(X), \quad \|X\|_{\mathrm{tr}} = \sum_{i=1}^{r} \sigma_r(X)$$

where $\sigma_1(X) \geq \ldots \geq \sigma_r(X) \geq 0$ are the singular values of the matrix X

- Indicator function: if C is convex, then its indicator function

$$I_C(x) = \begin{cases} 0 & x \in C \\ \infty & x \notin C \end{cases}$$

  is convex

- Support function: for any set C (convex or not), its support function

$$I_C^*(x) = \max_{y \in C} \ x^T y$$

  is convex

- Max function: $f(x) = \max\{x_1, \ldots x_n\}$ is convex

Key properties of convex functions

- A function is convex if and only if its restriction to any line is convex

- Epigraph characterization: a function f is convex if and only if its epigraph

$$\mathrm{epi}(f) = \{(x, t) \in \mathrm{dom}(f) \times \mathbb{R} : f(x) \leq t\}$$

is a convex set

- Convex sublevel sets: if f is convex, then its sublevel sets

$$\{x \in \mathrm{dom}(f) : f(x) \leq t\}$$

are convex, for all $t \in \mathbb{R}$. The converse is not true

- First-order characterization: if f is differentiable, then f is convex if and only if $\mathrm{dom}(f)$ is convex, and

$$f(y) \geq f(x) + \nabla f(x)^T (y - x)$$

  for all $x, y \in \mathrm{dom}(f)$. Therefore for a differentiable convex function $\nabla f(x) = 0 \iff x$ minimizes f

- Second-order characterization: if f is twice differentiable, then f is convex if and only if $\mathrm{dom}(f)$ is convex, and $\nabla^2 f(x) \succeq 0$ for all $x \in \mathrm{dom}(f)$

- Jensen's inequality: if f is convex, and X is a random variable supported on $\mathrm{dom}(f)$, then $f(\mathbb{E}[X]) \leq \mathbb{E}[f(X)]$

Operations preserving convexity

- Nonnegative linear combination: $f_1, \dots f_m$ convex implies $a_1 f_1 + \dots + a_m f_m$ convex for any $a_1, \dots a_m \geq 0$
- Pointwise maximization: if f_s is convex for any $s \in S$, then $f(x) = \max_{s \in S} \; f_s(x)$ is convex. Note that the set S here (number of functions f_s) can be infinite
- Partial minimization: if $g(x, y)$ is convex in x, y, and C is convex, then $f(x) = \min_{y \in C} \; g(x, y)$ is convex

Example: distances to a set

Let C be an arbitrary set, and consider the maximum distance to C under an arbitrary norm $\|\cdot\|$:

$$f(x) = \max_{y \in C} \; \|x - y\|$$

Let's check convexity: $f_y(x) = \|x - y\|$ is convex in x for any fixed y, so by pointwise maximization rule, f is convex

Now let C be convex, and consider the minimum distance to C:

$$f(x) = \min_{y \in C} \; \|x - y\|$$

Let's check convexity: $g(x, y) = \|x - y\|$ is convex in x, y jointly, and C is assumed convex, so apply partial minimization rule

More operations preserving convexity

- Affine composition: if f is convex, then $g(x) = f(Ax + b)$ is convex
- General composition: suppose $f = h \circ g$, where $g : \mathbb{R}^n \to \mathbb{R}$, $h : \mathbb{R} \to \mathbb{R}$, $f : \mathbb{R}^n \to \mathbb{R}$. Then:
  - f is convex if h is convex and nondecreasing, g is convex
  - f is convex if h is convex and nonincreasing, g is concave
  - f is concave if h is concave and nondecreasing, g concave
  - f is concave if h is concave and nonincreasing, g convex

  How to remember these? Think of the chain rule when $n = 1$:

$$f''(x) = h''(g(x))g'(x)^2 + h'(g(x))g''(x)$$

- Vector composition: suppose that

$$f(x) = h\big(g(x)\big) = h\big(g_1(x), \ldots g_k(x)\big)$$

where $g : \mathbb{R}^n \to \mathbb{R}^k$, $h : \mathbb{R}^k \to \mathbb{R}$, $f : \mathbb{R}^n \to \mathbb{R}$. Then:

  - f is convex if h is convex and nondecreasing in each argument, g is convex
  - f is convex if h is convex and nonincreasing in each argument, g is concave
  - f is concave if h is concave and nondecreasing in each argument, g is concave
  - f is concave if h is concave and nonincreasing in each argument, g is convex

Example: log-sum-exp function

Log-sum-exp function: $g(x) = \log(\sum_{i=1}^k e^{a_i^T x + b_i})$, for fixed a_i, b_i, $i = 1, \ldots k$. Often called "soft max", as it smoothly approximates $\max_{i=1,\ldots k} \, (a_i^T x + b_i)$

How to show convexity? First, note it suffices to prove convexity of $f(x) = \log(\sum_{i=1}^n e^{x_i})$ (affine composition rule)

Now use second-order characterization. Calculate

$$
\begin{aligned}
\nabla_i f(x) &= \frac{e^{x_i}}{\sum_{\ell=1}^n e^{x_\ell}} \\
\nabla_{ij}^2 f(x) &= \frac{e^{x_i}}{\sum_{\ell=1}^n e^{x_\ell}} 1\{i = j\} - \frac{e^{x_i} e^{x_j}}{(\sum_{\ell=1}^n e^{x_\ell})^2}
\end{aligned}
$$

Write $\nabla^2 f(x) = \mathrm{diag}(z) - zz^T$, where $z_i = e^{x_i}/(\sum_{\ell=1}^n e^{x_\ell})$. This matrix is diagonally dominant, hence positive semidefinite

References and further reading

- S. Boyd and L. Vandenberghe (2004), "Convex optimization", Chapters 2 and 3
- J.P. Hiriart-Urruty and C. Lemarechal (1993), "Fundamentals of convex analysis", Chapters A and B
- R. T. Rockafellar (1970), "Convex analysis", Chapters 1–10,