Mathematic Foundations for Computer Science

Probability and Optimization

Chapter 4: Convex Optimization

Spring 2021

Instructor: Xiaodong Gu

Recall: What is Optimization?

Mathematical Optimization

The task of selecting the "best" configuration of a set of variables from a "feasible" set of configurations.

$$\begin{aligned} &\text{minimize (or maximize)} && f(x) \\ &\text{subject to} && x \in \chi \end{aligned}$$

Continuous Optimization

Optimization problems where feasible set χ is a connected subset of Euclidean space, and f is a continuous function.

$$\begin{aligned} &\text{minimize} && f(x) \\ &\text{subject to} && g_i(x) \le b_i, \text{ for } i \in C. \end{aligned}$$

Standard Form

Finding the minimizer of a function subject to constraints:

$$\begin{array}{lll} \underset{x}{\text{minimize}} & f_0(x) & \\ \text{subject to} & f_i(x) \leq 0 & i = 1, \ldots, m \\ & h_i(x) = 0 & i = 1, \ldots, p \end{array}$$

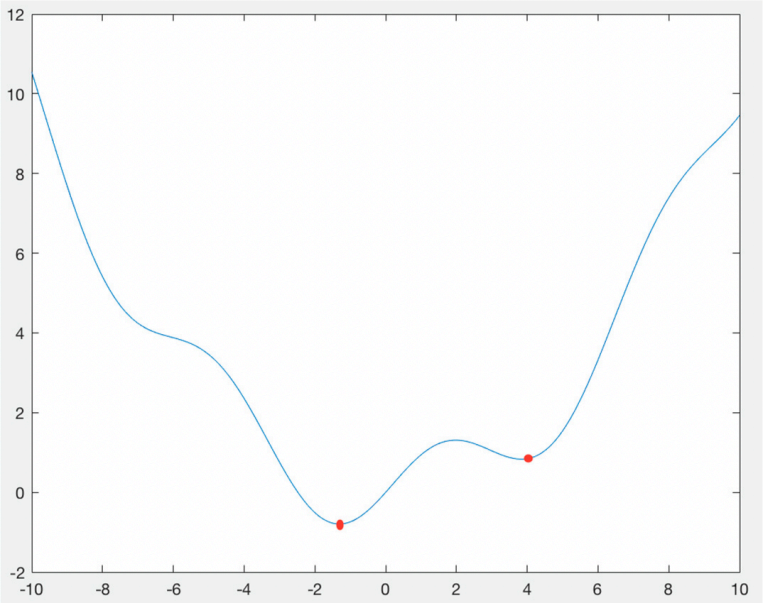
where

$x = (x_1, \ldots, x_n)$ is the optimization variable
$f_0 : \mathbf{R}^n \longrightarrow \mathbf{R}$ is the objective function
$f_i : \mathbf{R}^n \longrightarrow \mathbf{R}, \quad i = 1, \ldots, m$ are inequality constraint functions
$h_i : \mathbf{R}^n \longrightarrow \mathbf{R}, \quad i = 1, \ldots, p$ are equality constraint functions.

- **Goal**: find an optimal solution $x^\star$ that minimizes f_0 while satisfying all the constraints.

Local Minima and Global Minima

- local minima: a solution that is optimal within a neighboring set.
- global minima: the optimal solution among all possible solutions



What kind of functions have local minima == global minima?

Convex Optimization

Convex Optimization

A problem of minimizing a convex function (or maximizing a concave function) over a convex set.

Standard Form:

$$\begin{aligned} &\text{minimize} && f_0(x) \\ &\text{subject to} && g_i(x) \le 0, \text{ for i} \in C_1. \\ & && h_i(x) = 0, \text{ for i} \in C_2. \end{aligned}$$

where f_0, g_i, h_i are convex

Convex optimization problems have local minima == global minima

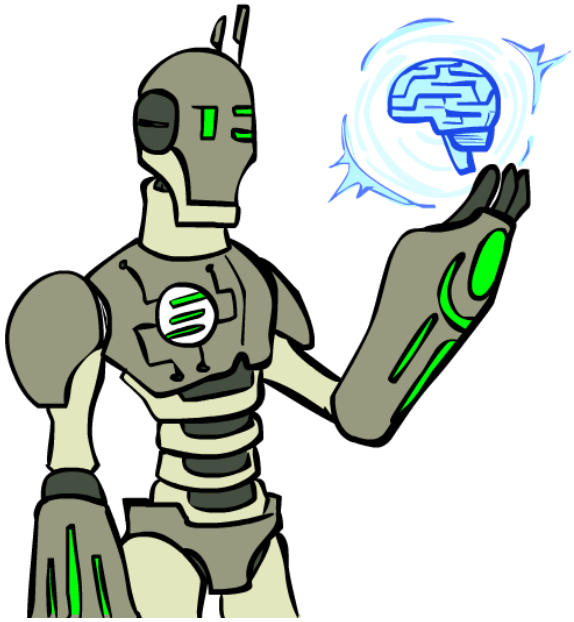
Overview of Chapter 3

Convex Problems

Convex set, convex function, quadratic programming, Lagrange duality

Optimization Algorithms

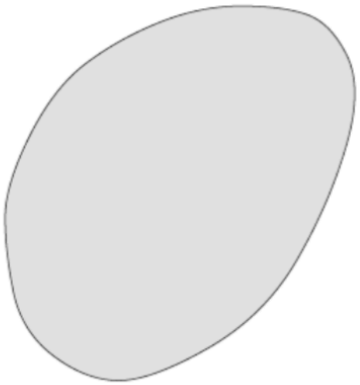
Gradient descend, Newton's method, interior point method,



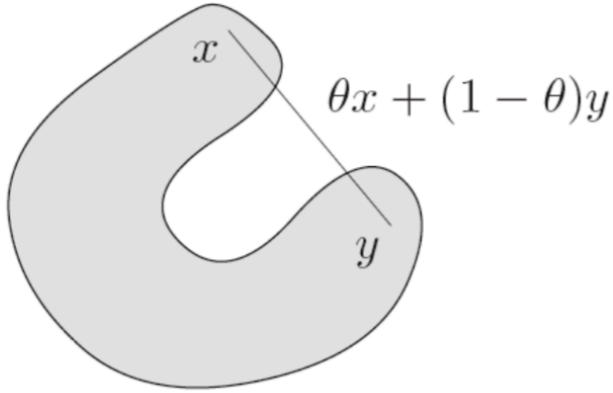
Convex Set

- A set $C \in \mathbf{R}^n$ is said to be **convex** if the line segment between any two points is in the set: for any $x, y \in C$ and $0 \leq \theta \leq 1$,

$$\theta x + (1 - \theta)\, y \in C.$$



convex



non-convex

Examples of Convex Set

- Trivial: $\emptyset$, point, line, etc.
- Hyperplane:

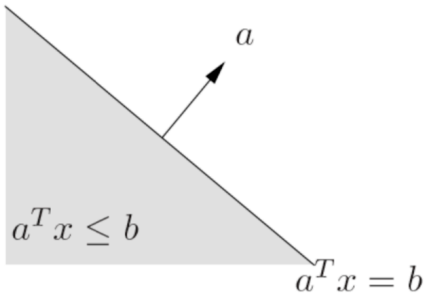
  $C = \{x \mid a^T x = b\}$ where a $\in R^n, b \in R$
- Halfplane:

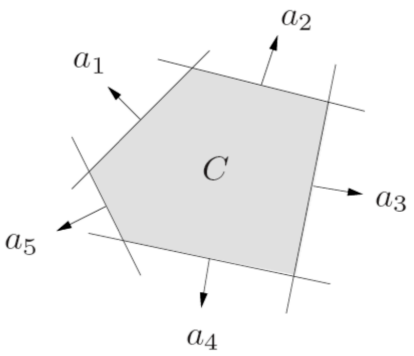
  $C = \{x \mid a^T x \leq b\}$
- Ployhedron:

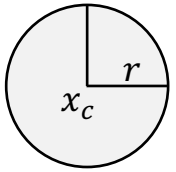
  $C = \{x \mid Ax \leq b, Cx = d\}$

  where $A \in R^{m \times n}, b \in R^m, C \in R^{p \times n}, d \in R^p$
- Euclidean ball:

  $B(x_c, r) = \{x \mid \|x - x_c\|_2 \leq r\}$

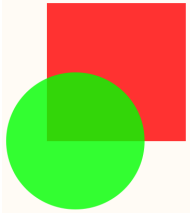






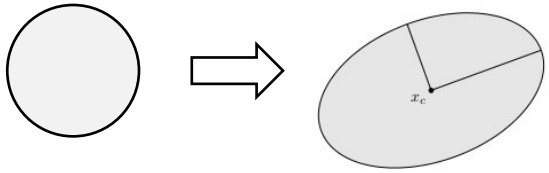
Operations preserving convexity

- Intersection: the intersection of convex sets is convex.



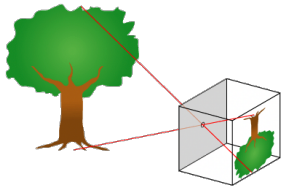
- Affine Maps: if f: $\mathbb{R}^n \rightarrow \mathbb{R}^m$ is an affine function (i.e., $f(x) = Ax+b$) and C is convex, then $f(C) = \{Ax+b : x \in C\}$ is convex.

  Example: an ellipsoid is image of a unit ball after an affine map



- Perspective Function: Let P: $\mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ be P(x,t) = x/t, and $S \in \mathbb{R}^{n+1}$ is convex, then $P(S)$ is convex.

  The perspective function scales or normalizes vectors so the last component is one, and then drops the last component.

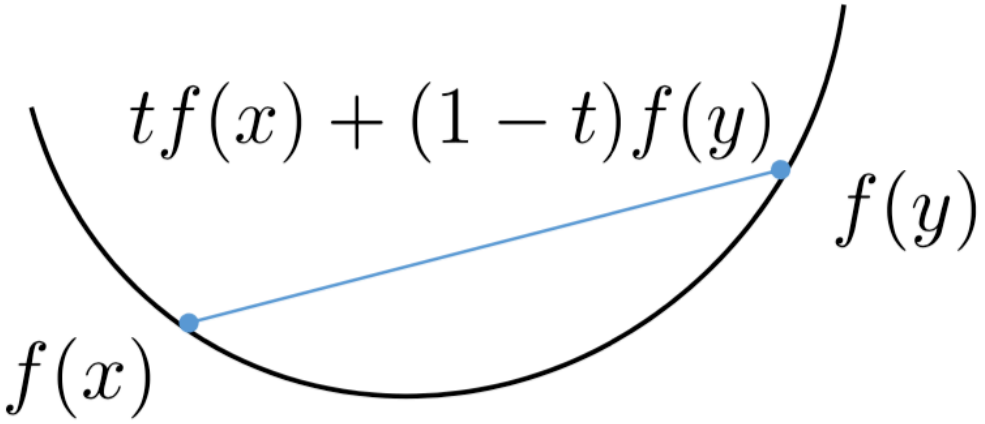


Convex Functions

A function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is convex if for $x, y \in$ **dom** f,

$$f(tx + (1\text{-}t)\, y) \leq t\, f(x) + (1\text{-}\, t)\, f(y),$$

for all $t \in [0, 1]$.



Example of Convex Functions

- Exponential function: e^{ax}
- logarithmic function log(x) is concave
- Affine function: $a^\top x + b$
- Quadratic function: $x^\top Q x + b^\top x + c$ is convex if Q is positive semidefinite (PSD)
- Least squares loss: $\|y - Ax\|_2^2$
- Norm: $\|x\|$ is convex for any norm

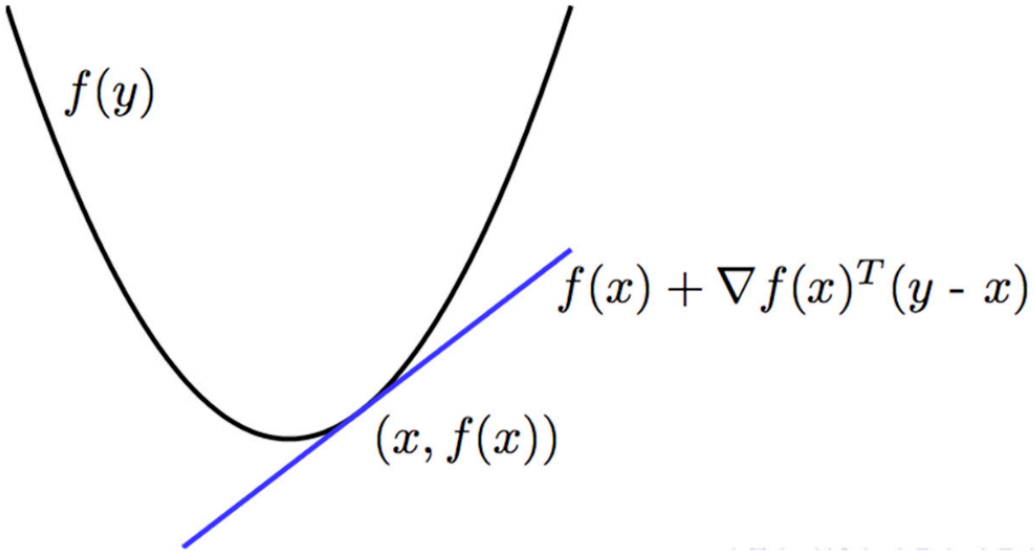
$$\|x\|_2 = \sqrt{\sum_{i=1}^{n} x_i^2} \qquad \|x\|_1 = \sum_{i=1}^{n} |x_i|$$

First Order Convexity Conditions

Theorem

Suppose f is differentiable. Then f is convex if and only if for all x, $y \in \mathbf{dom}\, f$

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



Second Order Convexity Conditions

Suppose f is twice differentiable. Then f is convex if and only if for all $x \in \text{dom } f$

$$\nabla^2 f(x) \succeq 0$$

Properties of convex functions

- If x is a local minimizer of a convex function, it is a global minimizer.
- Suppose f is differentiable and convex. Then, x is a global minimizer of $f(x)$ if and only if $\nabla f(x) = 0$.

- Proof:
  - $\nabla f(x) = 0$. We have

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

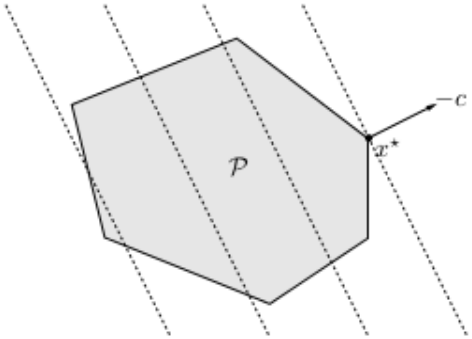
  - $\nabla f(x) \neq 0$. There is a direction of descent.

Convex Optimization Problems

Linear Programming

We have already seen linear programming

$$\begin{array}{ll} \text{minimize} & c^T x \\ \text{subject to} & Ax \leq b \\ & x \geq 0 \end{array}$$



Geometric Programming

Definition

- A monomial (单项式) is a function $f : R^n_+ \to R_+$ of the form
$$f(x) = cx_1^{a_1}x_2^{a_2}\dots x_n^{a_n}$$
where $c \geq 0$, $a_i \in R$.
- A posynomial (正多项式) is a sum of monomials.

A Geometric Program is an optimization problem of the following form

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq b_i, \quad \text{for } i \in \mathcal{C}_1. \\ & h_i(x) = b_i, \quad \text{for } i \in \mathcal{C}_2. \\ & x \succeq 0 \end{array}$$

where f_i's are posynomials, h_i's are monomials, and $b_i > 0$ (wlog 1).

Interpretation

- GP model volume/area minimization problems, subject to constraints.

Example: GP

A manufacturer is designing a suitcase (手提箱)

- Variables: h, w, d
- Want to minimize surface area: $2(hw + hd + wd)$ (i.e. amount of material used)
- Have a target volume: $hwd \geq 5$
- Practical/aesthetic (美学) constraints limit aspect ratio:

$$h/w \leq 2,\ h/d \leq 3$$

- Constrained by airline to $h + w + d \leq 7$

$$\begin{array}{ll} \text{minimize} & 2hw + 2hd + 2wd \\ \text{subject to} & h^{-1}w^{-1}d^{-1} \leq \frac{1}{5} \\ & hw^{-1} \leq 2 \\ & hd^{-1} \leq 3 \\ & h + w + d \leq 7 \\ & h, w, d \geq 0 \end{array}$$

$$\begin{aligned} \widetilde{h} &= \log h \\ \widetilde{w} &= \log w \\ \widetilde{d} &= \log d \end{aligned}$$

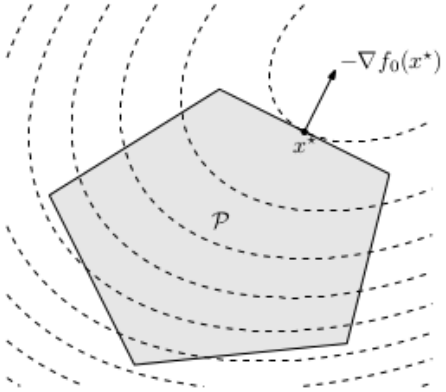
$$\Longrightarrow$$

$$\begin{array}{ll} \text{minimize} & 2e^{\widetilde{h}+\widetilde{w}} + 2e^{\widetilde{h}+\widetilde{d}} + 2e^{\widetilde{w}+\widetilde{d}} \\ \text{subject to} & e^{-\widetilde{h}-\widetilde{w}-\widetilde{d}} \leq \frac{1}{5} \\ & e^{\widetilde{h}-\widetilde{w}} \leq 2 \\ & e^{\widetilde{h}-\widetilde{d}} \leq 3 \\ & e^{\widetilde{h}} + e^{\widetilde{w}} + e^{\widetilde{d}} \leq 7 \end{array}$$

Quadratic Programming (二次规划)

Minimizing convex quadratic function (二次函数) over a polyhedron. Require $P \succcurlyeq 0$.

$$\begin{array}{ll} \text{minimize} & x^T P x + c^T x + d \\ \text{subject to} & Ax \leq b \end{array}$$



$P \succcurlyeq 0$: positive semi-definite, $P \in S^n_+$

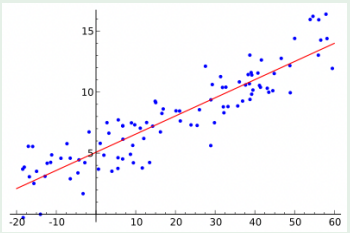
symmetric & all eigenvalues are nonnegative,

$x^T P x \geq 0$ for all x

Example: QP

Constrained Least Squares

Given a set of measurements $(x_1, y_1), \ldots, (x_m, y_m)$, where $x_i \in R^n$ is the i-th input and $y_i \in R$ is the i-th output, fit a linear function minimizing mean square error, subject to known bounds on the linear coefficients.



$$\text{minimize } \|Ax - b\|_2^2 = x^T A^T A x - 2b^T A x + b^T b$$
$$\text{subject to } l_i \le x_i \le u_i, \text{ for } i = 1,\ldots,n.$$

Conic Optimization Problems (锥优化)

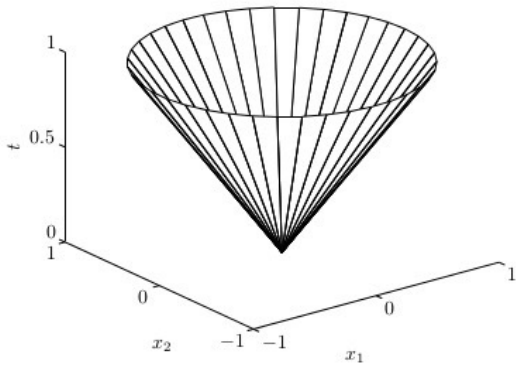
This is an umbrella term for problems of the following form

$$\begin{array}{ll} \text{minimize} & c^\intercal x \\ \text{subject to} & Ax + b \in K \end{array}$$

Where K is a convex cone (e.g. R^n_+, positive semi-definite matrices, etc). Evidently, such optimization problems are convex.

Example: Second Order Cone Programming where K is a second order cone:

$$K = \{(x, t) : \|x\|_2 \le t\}$$



Semi-Definite Programming (SDP)

These are conic optimization problems where the cone in question is the set of positive semi-definite matrices.

$$\begin{aligned} &\text{minimize} && c^T x \\ &\text{subject to} && x_1 F_1 + x_2 F_2 \ldots x_n F_n + G \succcurlyeq 0 \end{aligned}$$

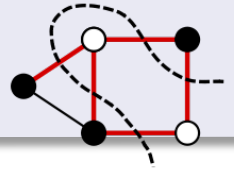
where $F_1, \ldots, F_n$, G are matrices, and $\succcurlyeq$ refers to the positive semi-definite cone S^n_+.

Example: SDP

Max Cut Problem

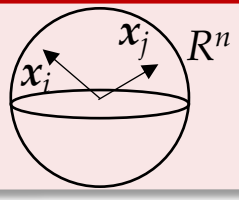
Given an undirected graph G = (V, E), find a partition of V into (S, V\S) maximizing number of edges with exactly one end in S.

$$\begin{aligned} \text{maximize} \quad & \Sigma_{(i,j)\in E}\,(1-x_i x_j)/2 \\ \text{subject to} \quad & x_i \in \{-1, 1\}, \quad \text{for } i \in V. \end{aligned}$$



Vector Program Relaxation

$$\begin{aligned} \text{maximize} \quad & \Sigma_{(i,j)\in E}\,(1-\boldsymbol{x}_i \boldsymbol{x}_j)/2 \\ \text{subject to} \quad & \|\boldsymbol{x}_i\|_2 = 1, \quad \text{for } i \in V. \\ & \boldsymbol{x}_i \in R^n, \quad \text{for } i \in V. \end{aligned}$$



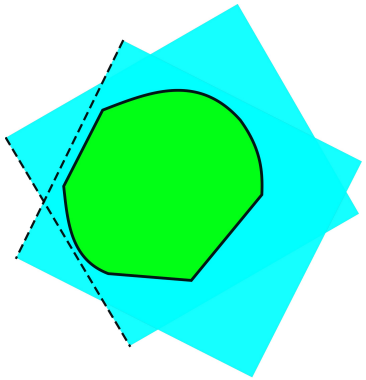
SDP Relaxation

$$\begin{aligned} \text{maximize} \quad & \Sigma_{(i,j)\in E}\,(1-X_{ij})/2 \\ \text{subject to} \quad & X_{ii} = 1, \quad \text{for } i \in V. \\ & X \in S^n_+, \quad \text{for } i \in V. \end{aligned}$$

$$\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^T = \begin{bmatrix} x_1x_1 & \cdots & x_1x_n \\ \vdots & \ddots & \vdots \\ x_nx_1 & \cdots & x_nx_n \end{bmatrix}$$

Duality of Convex Optimization

Geometric Duality of Convex Sets



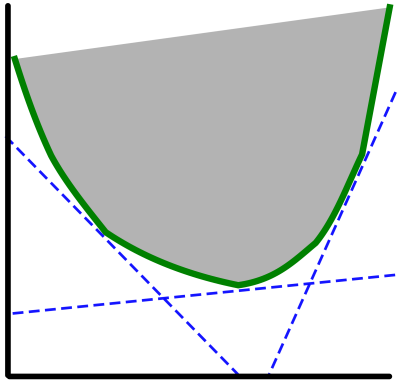
There are two equivalent ways to represent a convex set

- The family of points in the set (standard representation)
- The set of half-spaces containing the set ("dual" representation)

Theorem

A closed convex set S is the intersection of all closed half spaces H containing it.

Geometric Duality of Convex Functions



Theorem

A convex function is the point-wise supremum of all affine functions under-estimating it everywhere.

Recall: Duality in LP

Primal LP

$$\begin{aligned} \text{maximize} \quad & c^T x \\ \text{subject to} \quad & Ax \preccurlyeq b \end{aligned}$$

Dual LP

$$\begin{aligned} \text{minimize} \quad & b^T y \\ \text{subject to} \quad & A^T y \succcurlyeq c \\ & y \succcurlyeq 0 \end{aligned}$$

maximize $2x_1+3x_2$

$y_1 \times 4x_1+8x_2 \le 12$

$y_2 \times 2x_1+x_2 \le 3$

$y_3 \times 3x_1+2x_2 \le 4$

minimize $12y_1+3y_2+4y_3$

subject to $4y_1+2y_2+3y_3 \ge 2$

$8y_1+y_2+2y_3 \ge 3$

$y_1,y_2,y_3 \ge 0$

$(4y_1+2y_2+3y_3)x_1+(8y_1+y_2+3y_3)x_2 \le 12y_1+3y_2+4y_3$

$z^*=2x_1+3x_2 \le (4y_1+2y_2+3y_3)x_1+(8y_1+y_2+2y_3)x_2$

Dual LP relax/soften the constraints by replacing each with a linear "penalty term" or "cost" in the objective.

The Lagrangian

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \le 0, \quad \text{for i = 1,...,m.} \\ & h_i(x) = 0, \quad \text{for i = 1,...,k.} \end{aligned}$$

The basic idea of Lagrangian duality is to relax/soften the constraints by replacing each with a linear "penalty term" or "cost" in the objective.

The Lagrangian Function

$$L(x, \lambda, v) = f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{k} v_i h_i(x)$$

- λ_i is largrange multiplier for i-th inequality constraint
  - required to be nonnegative
- v_i is largrange multiplier for i-th equality constraint
  - allowed to be arbitrary sign

Lagrange Dual Function

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \le 0, \quad \text{for i = 1,...,m.} \\ & h_i(x) = 0, \quad \text{for i = 1,...,k.} \end{aligned}$$

The lagrange dual function gives the optimal value of the primal problem subject to the softened constraints.

The Lagrange Dual Function

$$g(\lambda, v) = \inf_{x \in \mathcal{D}} L(x, \lambda, v) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{k} v_i h_i(x)\right)$$

- Observe that:
  - The infimum is unconstrained (as opposed to the original constrained minimization problem)
  - g is concave regardless of original problem
  - The Lagrange dual can be unbounded (-∞) for some λ and v.

Interpretation: "Soft" Lower Bound

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \le 0, \quad \text{for i = 1,...,m.} \\ & h_i(x) = 0, \quad \text{for i = 1,...,k.} \end{aligned}$$

The Lagrange Dual Function

$$g(\lambda, v) = \inf_{x \in \mathcal{D}} L(x, \lambda, v) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{k} v_i h_i(x)\right)$$

Fact

g(λ, v) is a lower bound on OPT(primal) for every $\lambda \ge 0$ and $v \in R^k$.

Proof

- Every primal feasible x incurs nonpositive penalty by L(x,λ,v)
- Therefore, $L(x^*, \lambda, v) \le f_0(x^*)$
- So $g(\lambda, v) \le f_0(x^*) = \text{OPT(Primal)}$

Interpretation: "Soft" Lower Bound

$$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{subject to} \quad & f_i(x) \le 0, \quad \text{for i = 1,...,m.} \\ & h_i(x) = 0, \quad \text{for i = 1,...,k.} \end{aligned}$$

The Lagrange Dual Function

$$g(\lambda, v) = \inf_{x \in \mathcal{D}} L(x, \lambda, v) = \inf_{x \in \mathcal{D}} \left(f_0(x) + \sum_{i=1}^{m} \lambda_i f_i(x) + \sum_{i=1}^{k} v_i h_i(x)\right)$$

Interpretation

- A "hard" feasibility constraint can be thought of as imposing a penalty of +∞ if violated, and a penalty/reward of 0 if satisfied.
- Largrangian imposes a "soft" linear penalty for violating a constraint, and a reward for slack.
- Lagrange dual finds the optimal subjects to these soft constraints.

Lagrange Dual Problem

This is the problem of finding the best lower bound on OPT(primal) obtained from the Largrange dual function

$$\begin{array}{ll} \underset{\lambda,\nu}{\text{maximize}} & g(\lambda,\nu) \\ \text{subject to} & \lambda \geq 0. \end{array}$$

- Note: this is a convex optimization problem, regardless of whether primal problem was convex
- By convention, sometimes we add "dual feasibility" constraints to impose "nontrivial" lower bounds (i.e., $g(\lambda, v) \geq -\infty$)
- (λ^*, v^*) solving the above are referred to as the dual optimal solution.

Lagrange Dual Problem

Example: Least-Norm Solution of Linear Equations

- Consider the problem

$$\begin{array}{ll} \underset{x}{\text{minimize}} & x^T x \\ \text{subject to} & Ax = b. \end{array}$$

- The Lagrangian is

$$L(x, \nu) = x^T x + \nu^T (Ax - b).$$

- To find the dual function, we need to solve an unconstrained minimization of the Lagrangian. We set the gradient equal to zero

$$\nabla_x L(x, \nu) = 2x + A^T \nu = 0 \Longrightarrow x = -(1/2) A^T \nu$$

Lagrange Dual Problem

Example: Least-Norm Solution of Linear Equations

and we plug the solution in L to obtain g:

$$g(\nu) = L\left(-(1/2)A^T\nu, \nu\right) = -\frac{1}{4}\nu^T AA^T\nu - b^T\nu$$

- The function g is, as expected, a concave function of ν.
- From the lower bound property, we have

$$\text{OPT(primal)} \geq -\frac{1}{4}\nu^T AA^T\nu - b^T\nu \text{ for all } \nu.$$

- The dual problem is the QP

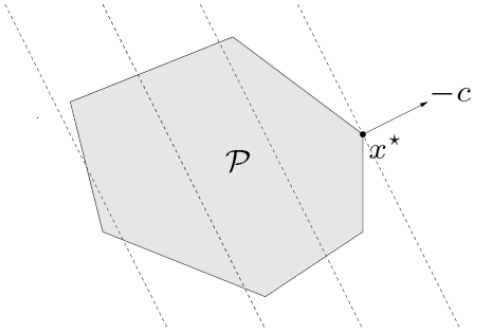
$$\underset{\nu}{\text{maximize}} \quad -\tfrac{1}{4}\nu^T AA^T\nu - b^T\nu.$$

Lagrange Dual Problem

Example: Standard Form LP

- Consider the problem

$$\begin{array}{ll} \underset{x}{\text{minimize}} & c^T x \\ \text{subject to} & Ax = b, \quad x \geq 0. \end{array}$$



- The Lagrangian is

$$\begin{aligned} L(x, \lambda, \nu) &= c^T x + \nu^T (Ax - b) - \lambda^T x \\ &= \left(c + A^T \nu - \lambda\right)^T x - b^T \nu. \end{aligned}$$

- L is a linear function of x and it is unbounded if the term multiplying x is nonzero.

Lagrange Dual Problem

- Hence, the dual function is

$$g(\lambda,\nu) = \inf_x L(x,\lambda,\nu) = \begin{cases} -b^T\nu & c + A^T\nu - \lambda = 0 \\ -\infty & \text{otherwise.} \end{cases}$$

- The function g is a concave function of (λ,ν) as it is linear on an affine domain.

- From the lower bound property, we have

$$\text{OPT(primal)} \geq -b^T\nu \quad \text{if } c + A^T\nu \geq 0.$$

- The dual problem is the LP

$$\begin{array}{ll} \underset{\nu}{\text{maximize}} & -b^T\nu \\ \text{subject to} & c + A^T\nu \geq 0. \end{array}$$

Weak duality

Primal Problem

$$\begin{aligned} &\min\ f_0(x) \\ &\text{s.t.} \\ &f_i(x) \le 0, \quad \forall i = 1, \ldots, m. \\ &h_i(x) = 0, \quad \forall i = 1, \ldots, k. \end{aligned}$$

Dual Problem

$$\begin{aligned} &\max\ g(\lambda, \nu) \\ &\text{s.t.} \\ &\lambda \succeq 0 \end{aligned}$$

Weak Duality

OPT(dual) ≤ OPT(primal)

Proof: g(λ, v) ≤ OPT(primal) for feasible (λ, v)

- Holds for every (even nonconvex) optimization problem
- Duality Gap: difference between optimal dual and primal values

  OPT(primal) – OPT(dual)

- Solving the dual problem may be used to find nontrivial lower bounds for difficult problems.

Strong duality

Even more interesting is when equality is achieved in weak duality which is called strong duality:

Strong Duality

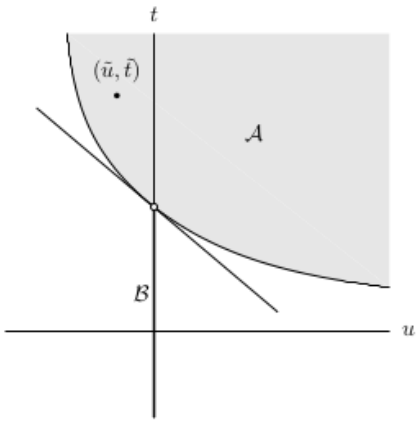
OPT(dual) = OPT(primal)

- Equivalently: there exists a setting of Lagrange multipliers so that g(λ, v) gives a tight lower bound on primal optimal value.
- Very desirable (we can solve a difficult problem by solving the dual)
- In general, does not hold for non-convex optimization problems
- Usually, but not always, holds for convex optimization problems.

Slater's Condition for Strong Duality

Slater's Condition

If the primal is a convex problem, and there exists at least one point x where all inequality constraints are strictly satisfied (i.e., $f_1(x)<0, \ldots, f_m(x)<0$ and $h_1(x)=\ldots h_n(x)=0$). Then strong duality holds.



- A sufficient condition for strong duality.
- Forces supporting hyperplane to be non-vertical
- Can be weakened to requiring strict feasibility only of non-affine constraints

Other Conditions

KKT Conditions

[Optional for after-class reading]

Optimization Algorithms

Unconstrained Minimization

- Gradient descent, Newton's method

Interior-point Methods

Review of Calculus

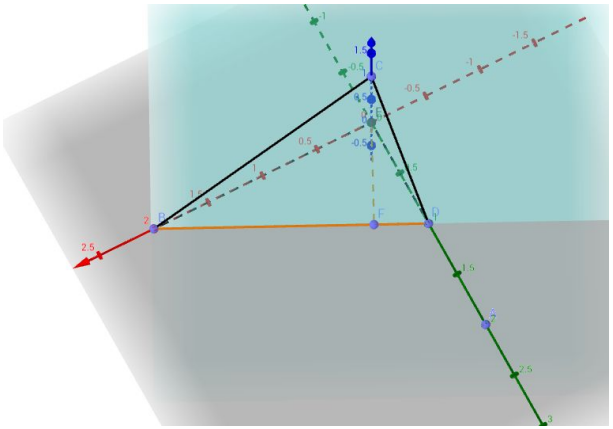
Gradient, Hessian

Gradient

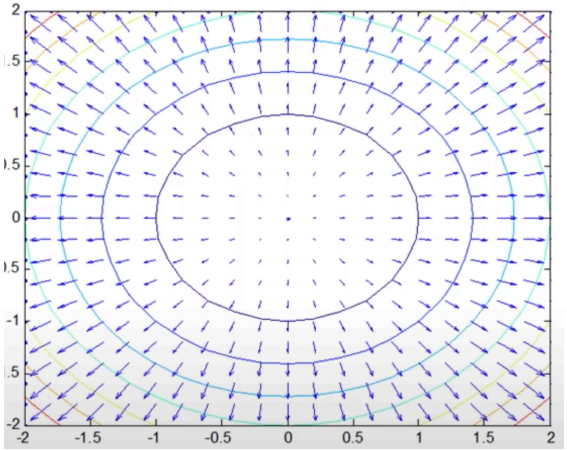
gradient – "direction and rate of fastest increase"

For $f: R^n \to R$, its gradient $\nabla f: R^n \to R^n$ is defined at the point $p = (x_1, \ldots, x_n)$ as the vector

$$\nabla f(p) = \begin{bmatrix} \frac{\partial f}{\partial x_1}(p) \\ \vdots \\ \frac{\partial f}{\partial x_n}(p) \end{bmatrix}$$



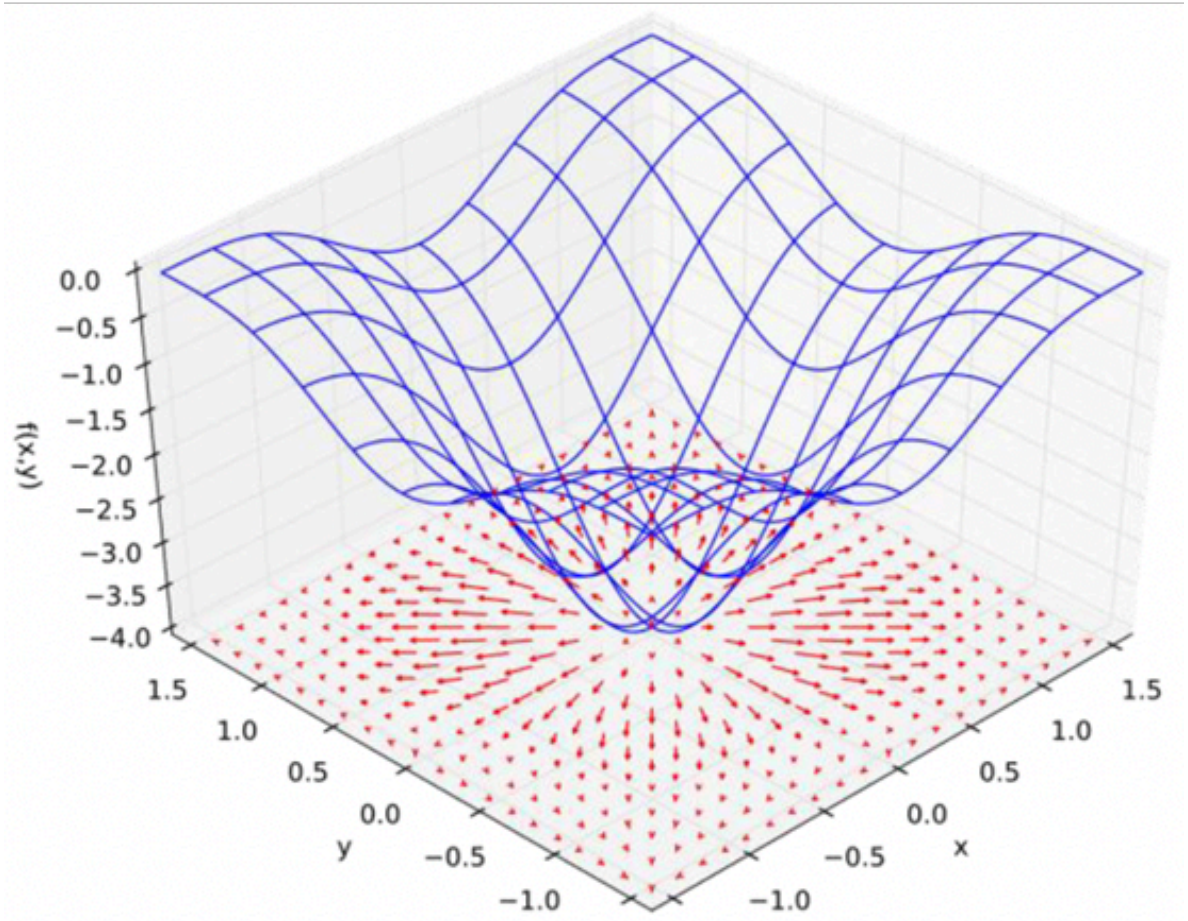
https://www.zhihu.com/question/36301367



Contours for x^2+y^2 with gradients

Gradient

gradients for arbitrary function $f(x,y)$



Gradient Vector

Example

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}, \qquad f(x) = 2x_1^2 x_2 - x_1 x_3^3$$

$$\nabla_x f(x) = \begin{bmatrix} \frac{\partial}{\partial x_1} f(x) \\ \frac{\partial}{\partial x_2} f(x) \\ \frac{\partial}{\partial x_3} f(x) \end{bmatrix} = \begin{bmatrix} 4x_1 x_2 - x_3^3 \\ 2x_1^2 \\ -3x_1 x_3^2 \end{bmatrix}$$

Hessian

A square matrix of second-order partial derivatives of a scalar-valued function.

- The Hessian of $f : \mathbb{R}^d \to \mathbb{R}$ is

$$\nabla^2 f = \begin{pmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_d} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_d \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_d^2} \end{pmatrix}$$

Hessian describes the local curvature (曲率) of a function of many variables

Unconstrained Minimization

$$\text{minimize} \quad f(x)$$

- f convex, twice continuously differentiable (hence $\mathbf{dom}\, f$ open)
- we assume optimal value $p^\star = \inf_x f(x)$ is attained (and finite)

unconstrained minimization methods

- produce sequence of points $x^{(k)} \in \mathbf{dom}\, f$, $k = 0, 1, \ldots$ with

$$f(x^{(k)}) \to p^\star$$

- can be interpreted as iterative methods for solving optimality condition

$$\nabla f(x^\star) = 0$$

Unconstrained Minimization

Algorithms

- Gradient Descend
- Steepest Descend
- Newton's Method
- …

Gradient Descend

The simplest optimization algorithm

- Goal:

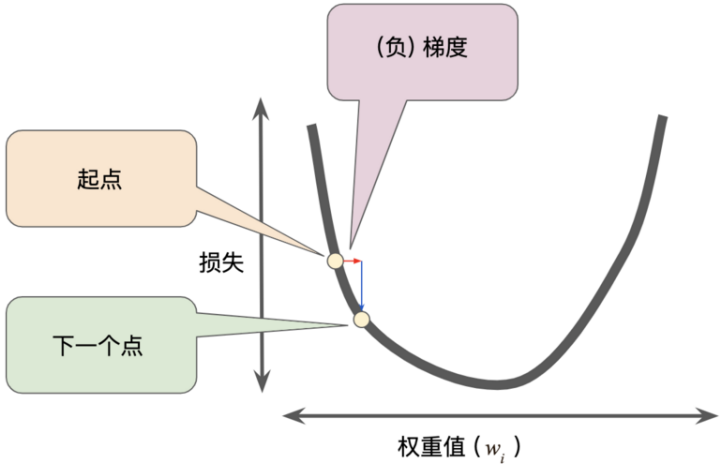
$$\min_x f(x)$$

Unconstrained Optimization

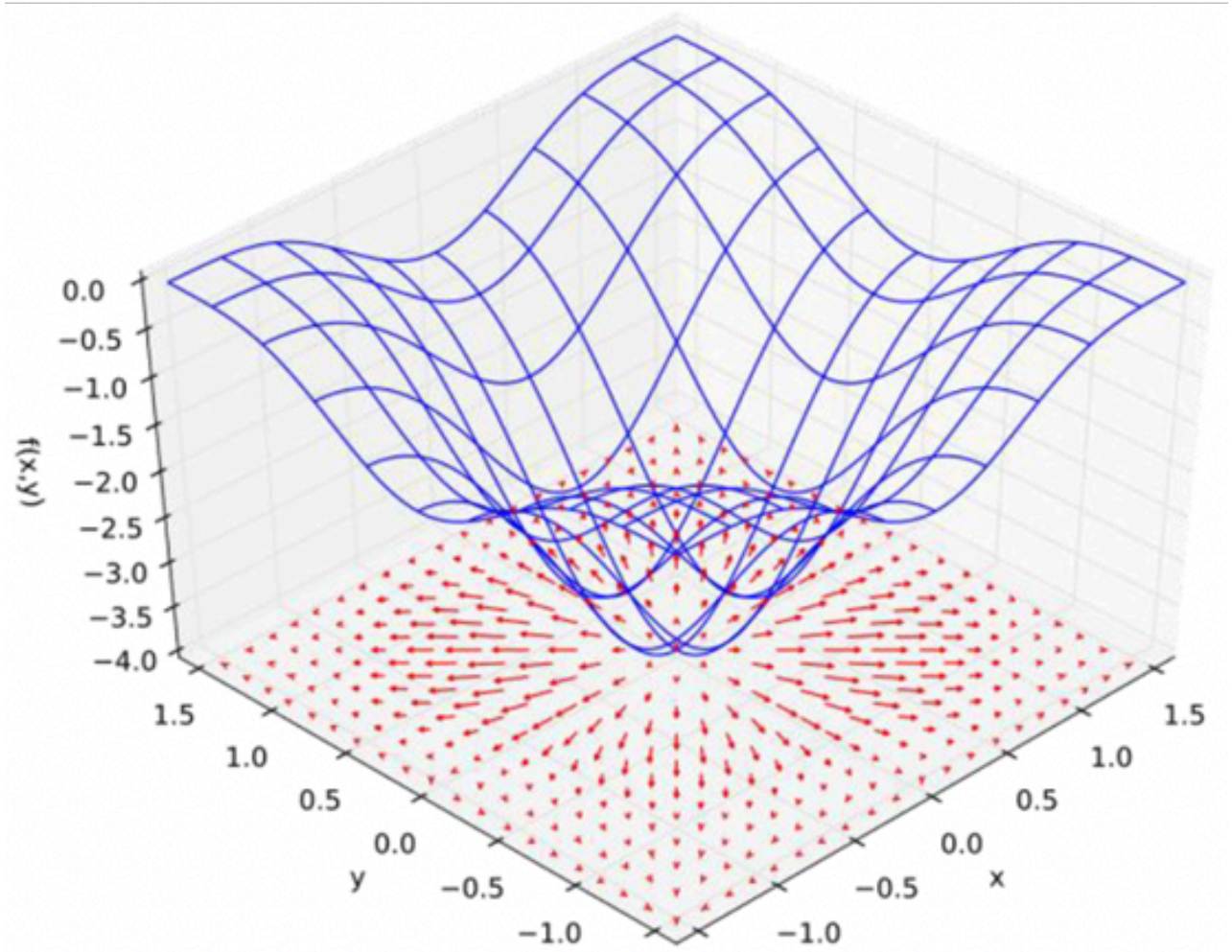
- Iteration:

$$x_{t+1} = x_t - \eta_t \nabla f(x_t)$$

- η_t is step size.

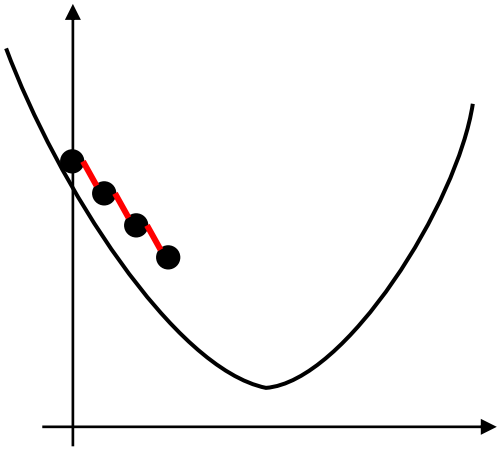


Gradient Descend

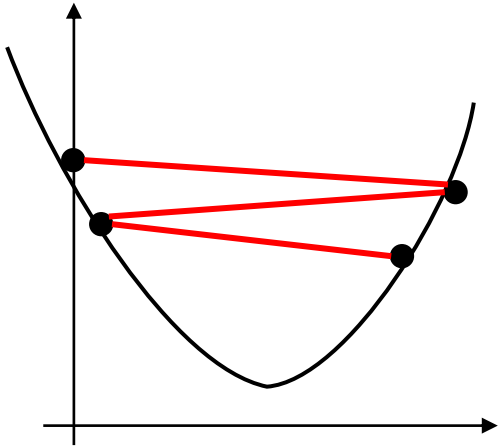


How to Choose Step Size?

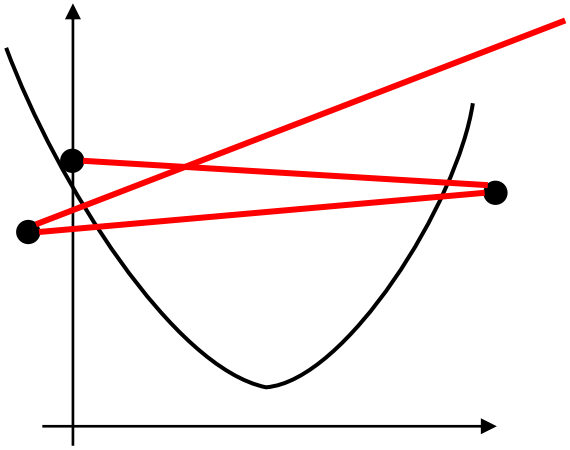
- If step size is too big, the value of function can diverge.
- If step size is too small, the convergence is very slow.



η too small:
slow progress



η too large:
oscillations

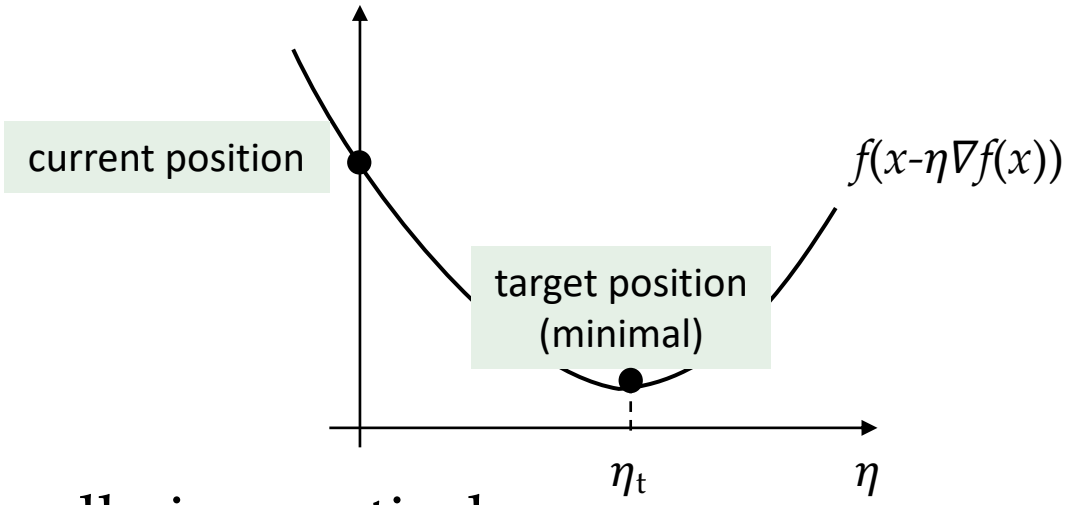


η much too large:
instability

On the Step Size: Exact Line Search

- Idea: the best step directly leads the function value to its minimal

$$\eta_t = \arg\min_{\eta} \; f(x - \eta \nabla f(x))$$



- Usually impractical.

On the Step Size: Backtracking Line Search

(回溯线搜索)

Using a relaxed tangent to approximate the upper bound of the minimal.

- Let $\alpha \in (0, 1/2], \beta \in (0, 1)$. Start with $\eta = 1$ and multiply $\eta = \beta\eta$ until

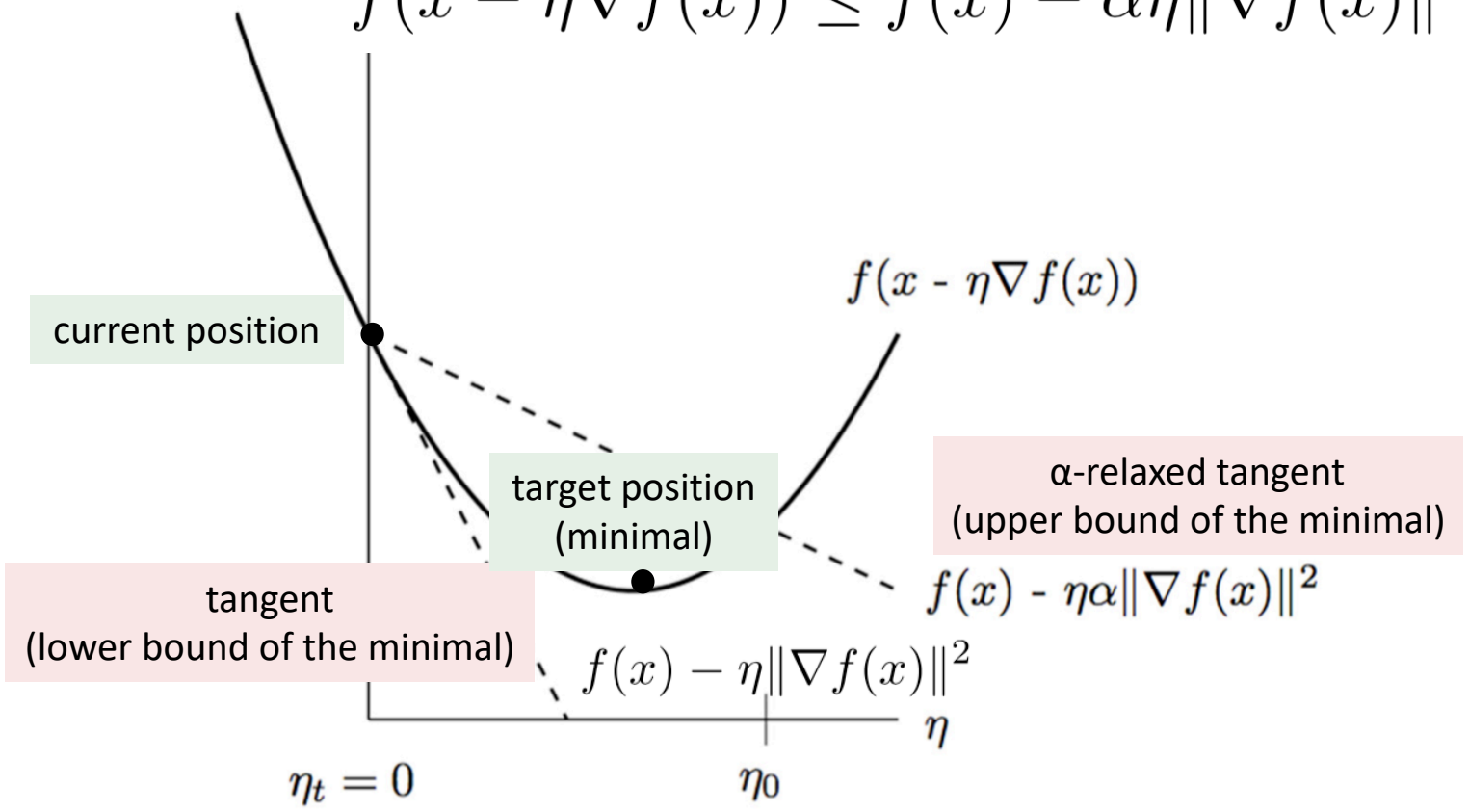
$$f(x - \eta \nabla f(x)) \leq f(x) - \alpha\eta \|\nabla f(x)\|^2$$

- Work well in practice.

On the Step Size: Backtracking Line Search

The α-relaxed tangent approximates an upper-bound of the minimal

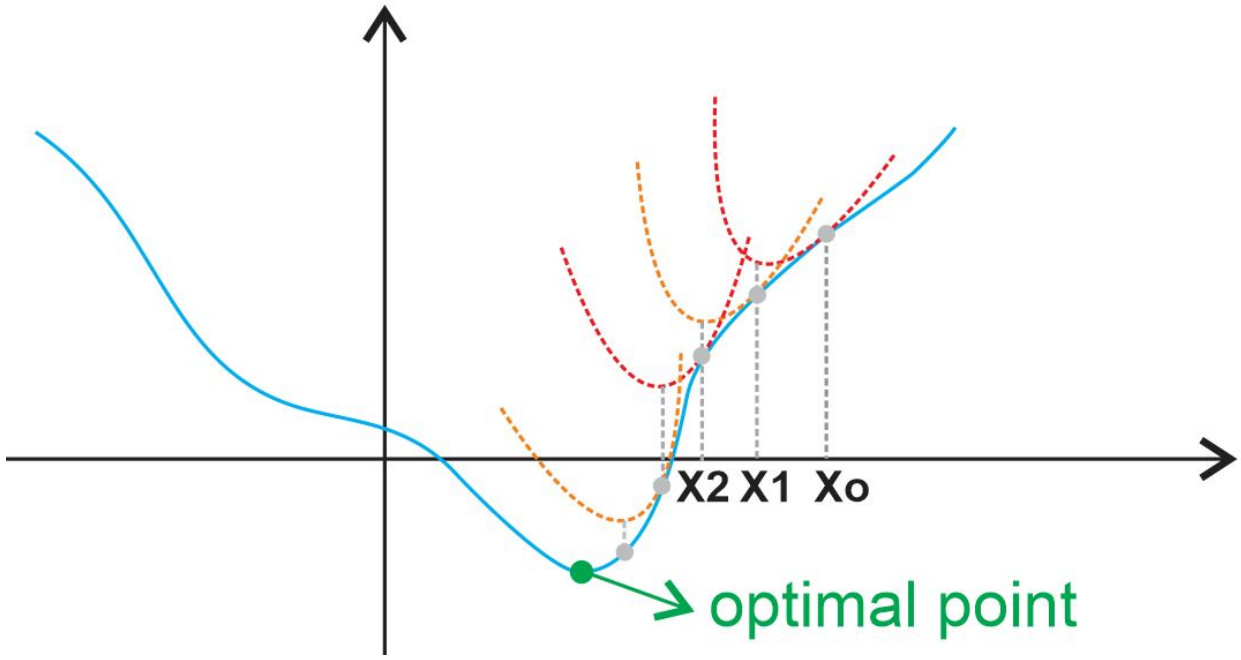
$$f(x - \eta \nabla f(x)) \leq f(x) - \alpha\eta \|\nabla f(x)\|^2$$



Newton's Method

The next step moves to the minimal of the second order approximation of f

$$x_{t+1} = x_t - [\nabla^2 f(x_t)]^{-1} \nabla f(x_t)$$



Newton's Method

interpretations

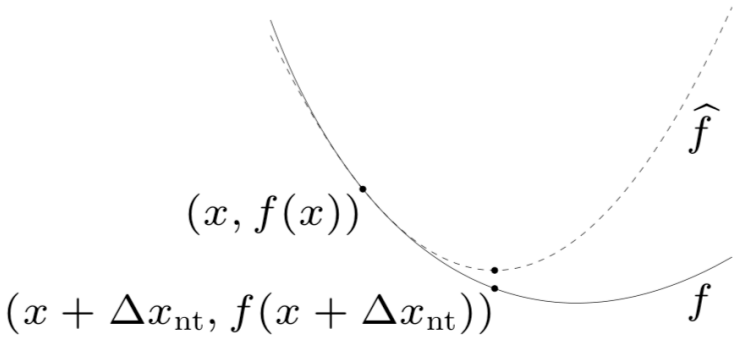
- $x + \Delta x_{\text{nt}}$ minimizes second order approximation

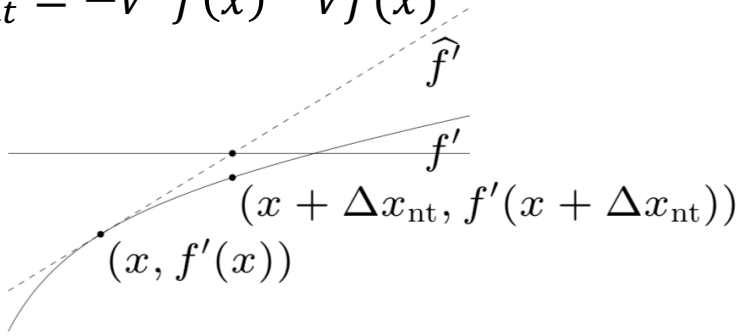
$$\widehat{f}(x+v) = f(x) + \nabla f(x)^T v + \frac{1}{2} v^T \nabla^2 f(x) v$$

- $x + \Delta x_{\text{nt}}$ solves linearized optimality condition

$$\nabla f(x+v) \approx \nabla \widehat{f}(x+v) = \nabla f(x) + \nabla^2 f(x) v = 0$$

$$\Rightarrow v^* = \nabla_{nt} = -\nabla^2 f(x)^{-1} \nabla f(x)$$





Interior-point Methods

Unconstrained minimization → Constrained minimization

- inequality constrained minimization
- logarithmic barrier function and central path
- barrier method

Inequality constrained minimization

What about minimization with inequality constrains?

$$\begin{array}{ll} \text{minimize} & f_0(x) \\ \text{subject to} & f_i(x) \leq 0, \quad i = 1, \ldots, m \\ & Ax = b \end{array}$$

where f_i's are assumed to be convex & twice continuously differentiable

Trapping Points within the Feasible Set

The inequality constraints can be made implicit by rewriting (1) as

$$\boxed{\begin{aligned}&\min\ f_0(x)\\ &\text{s.t. } Ax = b,\ \ f_i(x) \le 0\end{aligned}} \quad \Rightarrow \quad \boxed{\begin{aligned}&\min\ f_0(x) + \sum_{i=1}^{m} I_-(f_i(x))\\ &\text{s.t. } Ax = b\end{aligned}}$$

where $I_-(u) = 0$ for $u \le 0$, $I_-(u) = \infty$ otherwise.

But I_- is not differentiable.

The basic idea: approximate I_- by some differentiable function.

Logarithmic Barrier Function

Approximate $I\_$ by

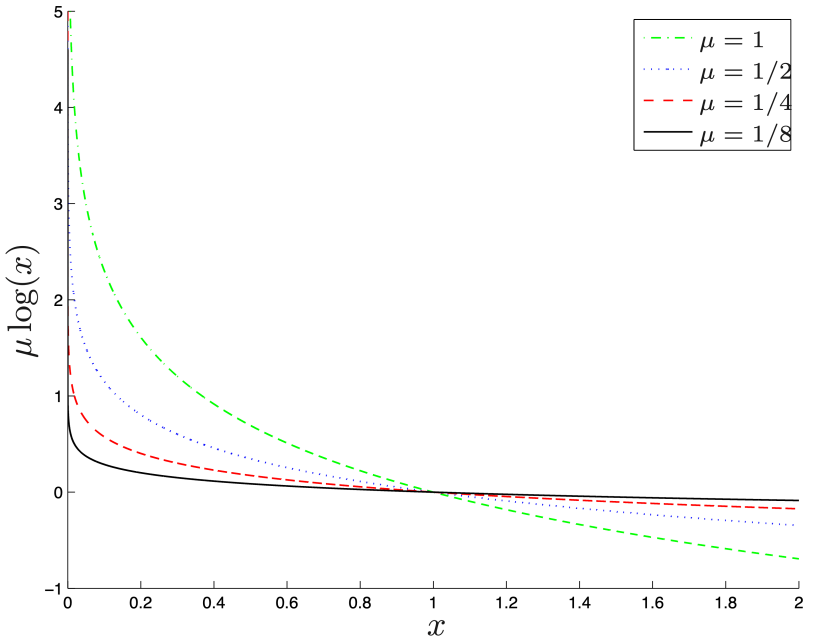
$$I\_(\mathrm{u}) = -\mu \log(-\mu), \quad \textbf{dom}\, I\_ = \{x \in \mathrm{R} \mid x < 0\}$$

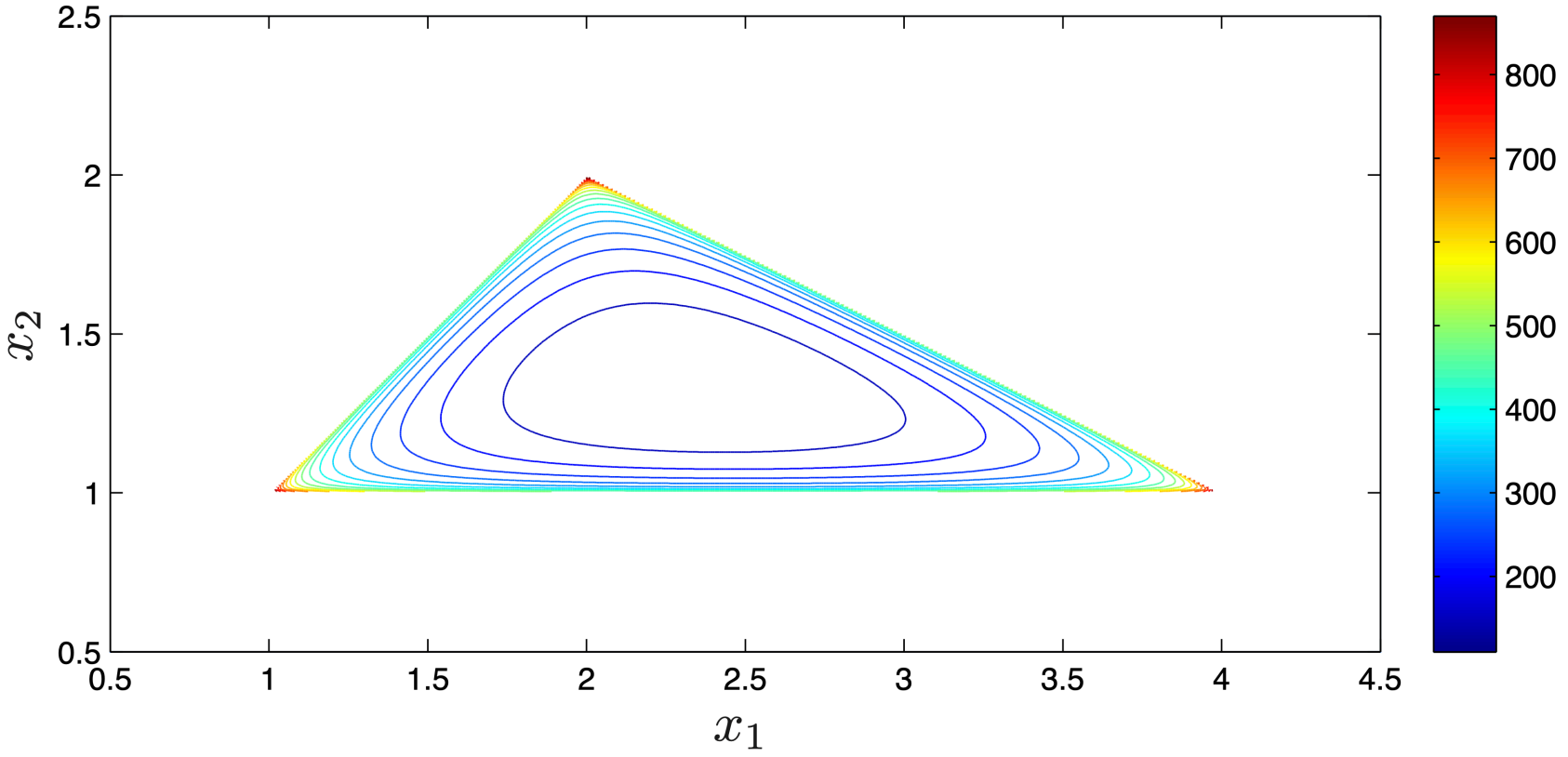
where $\mu > 0$ is a parameter that controls the accuracy of the approx.

Example: a single inequality $x \geq 0$.

The corresponding approx. is

$$-\mu \log(x)$$





Example: A set of linear inequalities $b_i - a^T_i x \le 0$, i = 1,...,m.

The corresponding approx. is

$$-\mu \sum_{i=1}^{m} \log(a_i^T x - b_i)$$

Logarithmic Barrier Function

The original problem can be approximated as:

$$\begin{aligned} \min \ & f_0(x) + \mu\phi(x) \\ \text{s.t.} \ & Ax = b \end{aligned}$$

min f(x) s.t. Ax=b is equivalent to $\min_z$ f(A+b+Fz)

with $$\phi(x) = -\sum_{i=1}^{m} \log(-f_i(x)), \quad \mathbf{dom}\, \phi = \{x \mid f_1(x) < 0, \ldots, f_m(x) < 0\}$$

φ is called the logarithmetic barrier function. Some nice properties:
– φ is convex (by composition).
– φ is twice differentiable:

$$\begin{aligned} \nabla\phi(x) &= \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla f_i(x) \\ \nabla^2\phi(x) &= \sum_{i=1}^{m} \frac{1}{f_i(x)^2} \nabla f_i(x) \nabla f_i(x)^T + \sum_{i=1}^{m} \frac{1}{-f_i(x)} \nabla^2 f_i(x) \end{aligned}$$

This means that the objective fn. is convex and twice differentiable.

Central Path

The log barrier approximation

$$\begin{aligned} \min\ & f_0(x) + \mu\phi(x) \\ \text{s.t.}\ & Ax = b \end{aligned} \qquad (*)$$

- is an accurate approximation of the original problem for $\mu \to 0$ (with $\mu > 0$)
- becomes difficult to minimize as $\mu \to 0$ (at least for Newton's method)

The basic idea: start with a large μ, and iteratively reduce μ until a desired solution accuracy is reached.

Define $x^*(u)$ to be the solution of (*).

Central path $\{x \mid x = x^*(u),\ u > 0\}$

is the collection of optimal points for various μ. A property:

$$f_0(x^*(\mu)) - p^* \le m\mu$$

Central Path

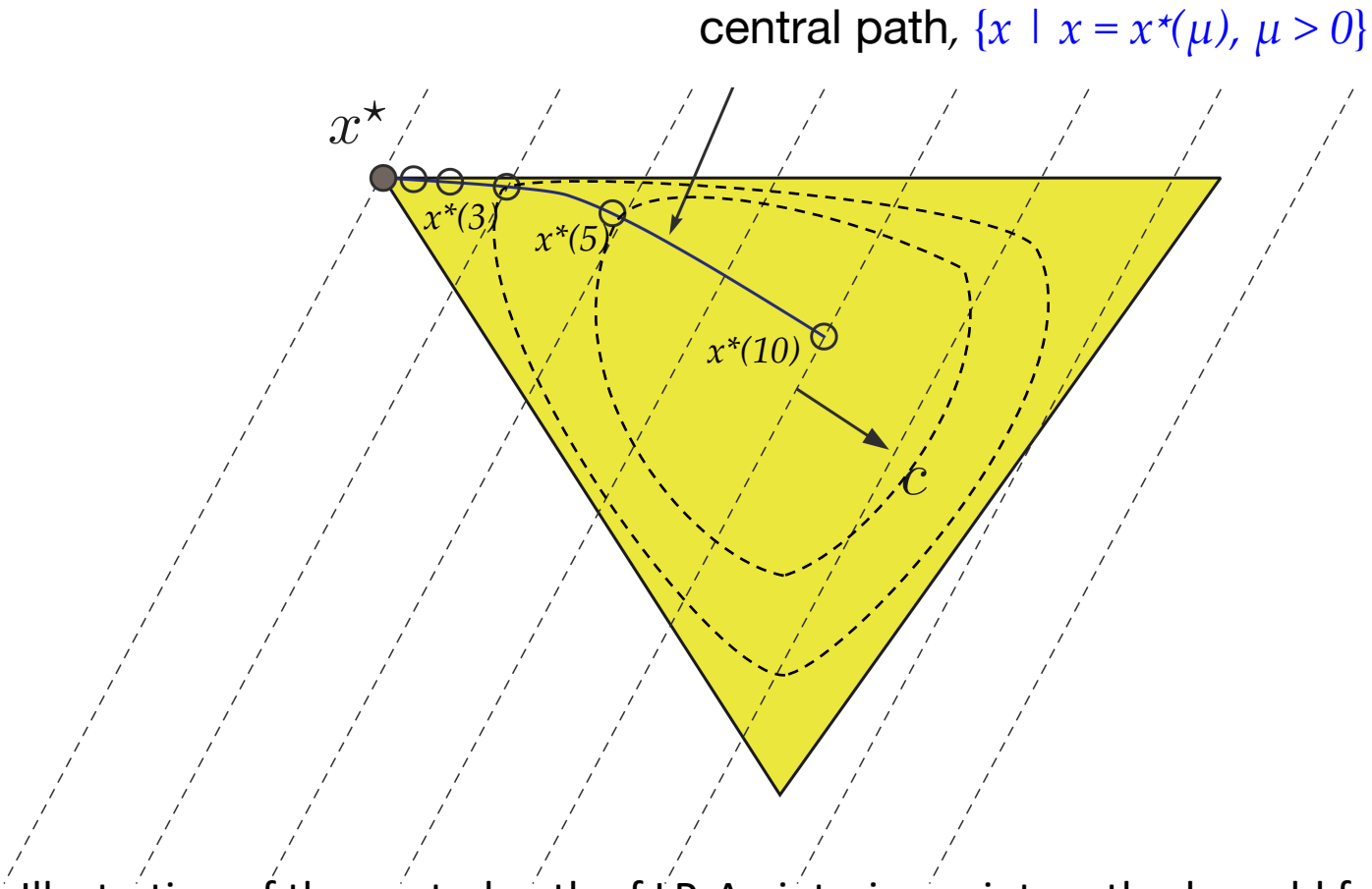


Illustration of the central path of LP. An interior-point method would follow the central path to iteratively approach the optimal solution.

Barrier Method (Path Following Method)

Path Following Method

given an initial strictly feasible point x, μ, $\epsilon > 0$, & $\beta < 1$.

repeat

1. *Centering step.* Starting at x, use Newton's method to solve
$$x^\star(\mu) = \min_x \mu f_0(x) + \phi(x) \text{ s.t. } Ax = b,$$
2. *Update.* $x := x^\star(\mu)$.
3. *Stopping criterion.* **quit** if $m\mu < \epsilon$.
4. *Target shifting.* $\mu := \beta\mu$.

- **Short-step path following:** choose β close to 1.
  – small number of Newton steps per outer iteration
  – large number of outer iterations
- **Long-step path following:** choose a small β.
  – increased number of Newton steps per outer iteration
  – smaller number of outer iterations