CME 307 / MS&E 311: Optimization

Interior Point Methods

Professor Udell

Management Science and Engineering
Stanford

March 1, 2024

with Prof. Luiz-Rafael Santos, UFSC https://lrsantos11.github.io/

Convex optimization

Convex optimization problem

$$\begin{array}{ll}
\text{minimize} & f(x) \\
\text{subject to} & g(x) \leq 0 \\
& Ax = b
\end{array}$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $g : \mathbf{R}^n \to \mathbf{R}^p$ are smooth and convex, $A \in \mathbf{R}^{m \times n}$ is full rank.

Convex optimization

Convex optimization problem

$$\begin{aligned}
\text{minimize} \quad & f(x) \\
\text{subject to} \quad & g(x) \leq 0 \\
& Ax = b
\end{aligned}$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $g : \mathbf{R}^n \to \mathbf{R}^p$ are smooth and convex, $A \in \mathbf{R}^{m \times n}$ is full rank.

KKT conditions:

Convex optimization

Convex optimization problem

$$
\begin{aligned}
\text{minimize} \quad & f(x) \\
\text{subject to} \quad & g(x) \leq 0 \quad s \in \mathbf{R}^p \\
& Ax = b \quad y \in \mathbf{R}^m
\end{aligned}
$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $g : \mathbf{R}^n \to \mathbf{R}^p$ are smooth and convex, $A \in \mathbf{R}^{m \times n}$ is full rank.

KKT conditions:

Convex optimization

Convex optimization problem

$$
\begin{aligned}
\text{minimize} \quad & f(x) \\
\text{subject to} \quad & g(x) \leq 0 \quad s \in \mathbf{R}^p \\
& Ax = b \quad y \in \mathbf{R}^m
\end{aligned}
$$

where $f : \mathbf{R}^n \to \mathbf{R}$, $g : \mathbf{R}^n \to \mathbf{R}^p$ are smooth and convex, $A \in \mathbf{R}^{m \times n}$ is full rank.

KKT conditions:

$$
\nabla f(x) + A^T y + (\nabla g(x))^T s = 0
$$
$$
Ax = b
$$
$$
g(x) \leq 0
$$
$$
s \geq 0
$$
$$
s_j g_j(x) = 0, \quad j = 1, \ldots, p
$$

Outline

IPM for linear and quadratic programs

IPM for Convex nonlinear programming

IPM for Conic Optimization

IPM for linear and quadratic programs

Linear/Quadratic Program

$$
\begin{array}{ll}
\text{minimize} & c^\top x + \frac{1}{2} x^\top Q x \\
\text{subject to} & Ax = b, \\
& x \geq 0,
\end{array}
$$

where $Q \in \mathbf{S}_+^n$, and $A \in \mathbf{R}^{m \times n}$ is full-rank.

IPM for linear and quadratic programs

Linear/Quadratic Program

$$\begin{array}{ll}
\text{minimize} & c^\top x + \frac{1}{2} x^\top Q x \\
\text{subject to} & Ax = b, \\
& x \geq 0,
\end{array}$$

where $Q \in \mathbf{S}_+^n$, and $A \in \mathbf{R}^{m \times n}$ is full-rank.

▶ $\mathcal{P} := \{x \in \mathbf{R}^n \mid Ax = b, x \geq 0\}$ is a polyhedron.

IPM for linear and quadratic programs

Linear/Quadratic Program

$$\begin{array}{ll} \text{minimize} & c^\top x + \frac{1}{2} x^\top Q x \\ \text{subject to} & Ax = b, \\ & x \geq 0, \end{array}$$

where $Q \in \mathbf{S}_+^n$, and $A \in \mathbf{R}^{m \times n}$ is full-rank.

▶ $\mathcal{P} := \{x \in \mathbf{R}^n \mid Ax = b, x \geq 0\}$ is a polyhedron.

▶ If $Q = 0$, problem is a linear program.

IPM for linear and quadratic programs

Linear/Quadratic Program

$$\begin{array}{ll} \text{minimize} & c^\top x + \frac{1}{2} x^\top Q x \\ \text{subject to} & Ax = b, \\ & x \geq 0, \end{array}$$

where $Q \in \mathbf{S}_+^n$, and $A \in \mathbf{R}^{m \times n}$ is full-rank.

- $\mathcal{P} := \{ x \in \mathbf{R}^n \mid Ax = b, x \geq 0 \}$ is a polyhedron.

- If $Q = 0$, problem is a linear program.

  How to solve LP/QP problems?

IPM for linear and quadratic programs

Linear/Quadratic Program

$$\begin{aligned} \text{minimize} \quad & c^\top x + \tfrac{1}{2} x^\top Q x \\ \text{subject to} \quad & Ax = b, \\ & x \geq 0, \end{aligned}$$

where $Q \in \mathbf{S}_+^n$, and $A \in \mathbf{R}^{m \times n}$ is full-rank.

- ▶ $\mathcal{P} := \{x \in \mathbf{R}^n \mid Ax = b, x \geq 0\}$ is a polyhedron.
- ▶ If $Q = 0$, problem is a linear program.

How to solve LP/QP problems?

Simplex: vertex to vertex
IPM: go through the middle!

IPM for linear and quadratic programs

Linear/Quadratic Program

$$\begin{array}{ll} \text{minimize} & c^\top x + \frac{1}{2} x^\top Q x \\ \text{subject to} & Ax = b, \\ & x \geq 0, \end{array}$$

where $Q \in \mathbf{S}^n_+$, and $A \in \mathbf{R}^{m \times n}$ is full-rank.

▶ $\mathcal{P} := \{x \in \mathbf{R}^n \mid Ax = b, x \geq 0\}$ is a polyhedron.

▶ If $Q = 0$, problem is a linear program.

How to solve LP/QP problems?

Advantages of vertex solution vs interior solution?

Simplex: vertex to vertex
IPM: go through the middle!

Building blocks of IPM

Ingredients for Interior Point Method

- ▶ Duality theory: Lagrangian function; KKT (first order optimality) condition.
- ▶ Barrier function: logarithmic barrier.
- ▶ Newton's method (and a good linear solver)

Building blocks of IPM

Ingredients for Interior Point Method

- ▶ Duality theory: Lagrangian function; KKT (first order optimality) condition.
- ▶ Barrier function: logarithmic barrier.
- ▶ Newton's method (and a good linear solver)

The reward: fantastic convergence properties!

- ▶ Theoretical: $O(\sqrt{n}\log(1/\varepsilon))$ iterations
- ▶ Practical: $O(\log n \log(1/\varepsilon))$ iterations

(but the per-iteration cost may be high due to the Newton solve: often $O(n^3)$)

IPM: algorithmic template

IPM procedure

- ▶ replace inequalities with log barriers;
- ▶ form the Lagrangian;
- ▶ write down the KKT conditions of the perturbed problem;
- ▶ find one (or more) directions using Newton's method on the KKT system;
- ▶ (decide how to combine the directions and) compute a stepsize.

Duality and KKT conditions

Primal-dual QPs

Primal problem

minimize $\quad c^\top x + \frac{1}{2} x^\top Q x$

subject to $\quad Ax = b$

$\qquad\qquad x \geq 0$

Dual problem

maximize $\quad b^\top y - \frac{1}{2} x^\top Q x$

subject to $\quad A^\top y + s - Qx = c$

$\qquad\qquad s \geq 0$

Duality and KKT conditions

Primal-dual QPs

Primal problem

$$
\begin{aligned}
\text{minimize} \quad & c^\top x + \tfrac{1}{2} x^\top Q x \\
\text{subject to} \quad & A x = b \\
& x \geq 0
\end{aligned}
$$

Dual problem

$$
\begin{aligned}
\text{maximize} \quad & b^\top y - \tfrac{1}{2} x^\top Q x \\
\text{subject to} \quad & A^\top y + s - Q x = c \\
& s \geq 0
\end{aligned}
$$

KKT conditions

$$
\begin{aligned}
A x &= b \\
A^\top y + s - Q x &= c \\
X S e &= 0 \qquad \rhd \text{complementarity:} \quad x \cdot s = 0 \\
(x, s) &\geq 0
\end{aligned}
$$

where $X = \mathbf{diag}(x_1, \ldots, x_n), S = \mathbf{diag}(s_1, \ldots, s_n) \in \mathbf{R}^{n \times n}$, and $e = (1, \ldots, 1) \in \mathbf{R}^n$.
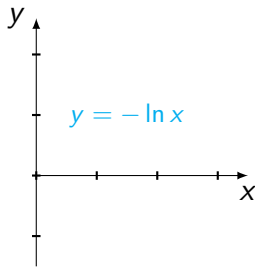
Logarithmic barrier

$-\ln x_j$
replaces the inequality
$x_j \geq 0$

$y = -\ln x$

Logarithmic barrier

$-\ln x_j$
replaces the inequality
$x_j \geq 0$

$y = -\ln x$
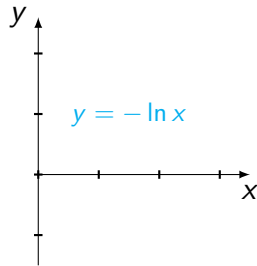
$$\text{minimize} \quad -\sum_{j=1}^{n} \ln x_j \quad \Longleftrightarrow \quad \text{maximize} \quad \prod_{1 \leq j \leq n} x_j$$

$\Longrightarrow$ keeps every entry of x away from 0.

Barrier primal QP

Step 1: replace inequality constraints by barrier

Replace the primal QP

$$\begin{aligned} \text{minimize} \quad & c^\top x + \tfrac{1}{2}x^\top Q x \\ \text{subject to} \quad & Ax = b \\ & x \geq 0 \end{aligned}$$

with the barrier primal QP

$$\begin{aligned} \text{minimize} \quad & c^\top x + \tfrac{1}{2}x^\top Q x - \mu \sum_{j=1}^{n} \ln x_j \\ \text{subject to} \quad & Ax = b \end{aligned}$$

Logarithmic barrier and stationarity

Step 2: remove equality constraints using Lagrangian

$$\mathcal{L}(x, y, \mu) = c^\top x + \frac{1}{2}x^\top Q x - y^\top(Ax - b) - \mu \sum_{j=1}^{n} \ln x_j$$

Logarithmic barrier and stationarity

Step 2: remove equality constraints using Lagrangian

$$\mathcal{L}(x, y, \mu) = c^\top x + \frac{1}{2} x^\top Q x - y^\top (Ax - b) - \mu \sum_{j=1}^{n} \ln x_j$$

A stationary point (x, y, μ) of the Lagrangian satisfies

$$\nabla_x \mathcal{L}(x, y, \mu) = 0 \qquad\qquad = c + Qx - A^\top y - \mu X^{-1} e$$

with $X^{-1} = \textbf{diag}(x_1^{-1}, \ldots, x_n^{-1}) \in \mathbf{R}^{n \times n}, (x_j > 0)$.

KKT conditions for barrier problem

▶ Define $s := \mu X^{-1} e$, which implies $XSe = \mu e$, to get

KKT$_\mu$

$$Ax = b$$
$$A^\top y + s - Qx = c$$
$$XSe = \mu e$$
$$(x, s) > 0$$

KKT conditions for barrier problem

▶ Define $s := \mu X^{-1} e$, which implies $XSe = \mu e$, to get

KKT$_\mu$

$$Ax = b$$
$$A^\top y + s - Qx = c$$
$$XSe = \mu e$$
$$(x, s) > 0$$

KKT

$$Ax = b$$
$$A^\top y + s - Qx = c$$
$$XSe = 0$$
$$(x, s) \geq 0$$

KKT conditions for barrier problem

▶ Define $s := \mu X^{-1} e$, which implies $XSe = \mu e$, to get

KKT_μ KKT

$$Ax = b$$
$$A^\top y + s - Qx = c$$
$$XSe = \mu e$$
$$(x, s) > 0$$

$$Ax = b$$
$$A^\top y + s - Qx = c$$
$$XSe = 0$$
$$(x, s) \geq 0$$

$$\text{KKT}_\mu \to \text{KKT as } \mu \to 0.$$

Central path (LP case)

▶ Parameter μ controls the distance to optimality

$$c^\top x - b^\top y = c^\top x - x^\top A^\top y = x^\top s = n\mu$$

Central path (LP case)

▶ Parameter μ controls the distance to optimality

$$c^\top x - b^\top y = c^\top x - x^\top A^\top y = x^\top s = n\mu$$

▶ Analytic center (μ-center): unique point

$$(x(\mu), y(\mu), s(\mu)), \qquad x(\mu) > 0, \ s(\mu) > 0$$

that satisfies the KKT_μ conditions.

Central path (LP case)

▶ Parameter μ controls the distance to optimality

$$c^\top x - b^\top y = c^\top x - x^\top A^\top y = x^\top s = n\mu$$
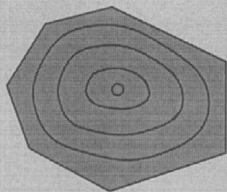
▶ Analytic center (μ-center): unique point

$$(x(\mu), y(\mu), s(\mu)), \qquad x(\mu) > 0, \ s(\mu) > 0$$
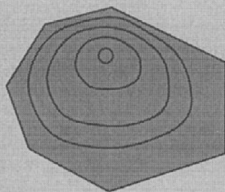
that satisfies the KKT_μ conditions.

▶ The curve

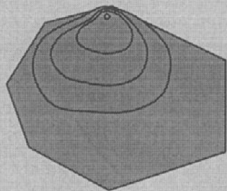$$\mathcal{C}_\mu = \{(x(\mu), y(\mu), s(\mu)) \mid \mu > 0\}$$

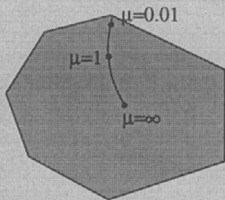is called the primal-dual central path.

(a) μ=∞ (b) μ=1 (c) μ=0.01 (d) central path

μ=0.01
μ=1
μ=∞

Recall Newton's method for nonlinear equation

▶ For $F : \mathbf{R}^n \to \mathbf{R}^n$ smooth, solve $F(x) = 0$.

Recall Newton's method for nonlinear equation

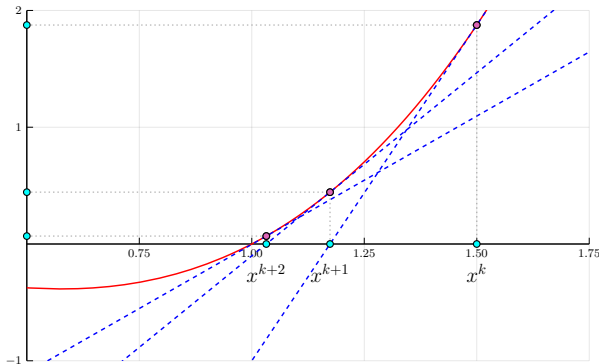▶ For $F : \mathbf{R}^n \to \mathbf{R}^n$ smooth, solve $F(x) = 0$.
▶ Newton's method: define Jacobian $J_F(x)$ so $J_F(x)_{ij} = \frac{\partial F_i}{\partial x_j}$, and iterate

$$x^{k+1} = x^k - \alpha_k J_F(x^k)^{-1} F(x^k)$$

Recall Newton's method for nonlinear equation

- For $F : \mathbf{R}^n \to \mathbf{R}^n$ smooth, solve $F(x) = 0$.
- Newton's method: define Jacobian $J_F(x)$ so $J_F(x)_{ij} = \frac{\partial F_i}{\partial x_j}$, and iterate

$$x^{k+1} = x^k - \alpha_k J_F(x^k)^{-1} F(x^k)$$

Apply Newton Method to KKT$_\mu$

The first order optimality conditions for the barrier problem form a large system of nonlinear equations:

$$F(x, y, s) = 0,$$

where $F : \mathbf{R}^{2n+m} \mapsto \mathbf{R}^{2n+m}$ is defined as

$$F(x, y, s) = \begin{bmatrix} Ax & -b \\ A^\top y + s - Qx & -c \\ XSe & -\mu e \end{bmatrix}$$

Apply Newton Method to KKT$_\mu$

The first order optimality conditions for the barrier problem form a large system of nonlinear equations:

$$F(x, y, s) = 0,$$

where $F : \mathbf{R}^{2n+m} \mapsto \mathbf{R}^{2n+m}$ is defined as

$$F(x, y, s) = \begin{bmatrix} Ax & -b \\ A^\top y + s - Qx & -c \\ XSe & -\mu e \end{bmatrix}$$

▶ The first two blocks are linear.
▶ The last block, corresponding to the complementarity condition, is nonlinear.
▶ Jacobian is

$$J_F(x, y, s) = \begin{bmatrix} A & 0 & 0 \\ -Q & A^\top & I \\ S & 0 & X \end{bmatrix}$$

Interior-point QP Algorithm

IPM Framework

Fix the barrier parameter μ and make *one* (damped) Newton step towards the solution of FOC. Then reduce the barrier parameter μ and repeat.

Interior-point QP Algorithm

IPM Framework

Fix the barrier parameter μ and make *one* (damped) Newton step towards the solution of FOC. Then reduce the barrier parameter μ and repeat.

▶ Given (x_0, y_0, s_0) feasible, $\mu_0 = \frac{1}{n} \cdot (x^0)^\top s^0$

Interior-point QP Algorithm

IPM Framework

Fix the barrier parameter μ and make *one* (damped) Newton step towards the solution of FOC. Then reduce the barrier parameter μ and repeat.

- ▶ Given (x_0, y_0, s_0) feasible, $\mu_0 = \frac{1}{n} \cdot (x^0)^\top s^0$
- ▶ For $k = 1, 2, \ldots$

Interior-point QP Algorithm

IPM Framework

Fix the barrier parameter μ and make *one* (damped) Newton step towards the solution of FOC. Then reduce the barrier parameter μ and repeat.

- Given (x_0, y_0, s_0) feasible, $\mu_0 = \frac{1}{n} \cdot (x^0)^\top s^0$
- For $k = 1, 2, \ldots$
    - $\mu_k = \sigma \mu_{k-1}$, where $\sigma \in (0, 1)$

Interior-point QP Algorithm

IPM Framework

Fix the barrier parameter μ and make *one* (damped) Newton step towards the solution of FOC. Then reduce the barrier parameter μ and repeat.

- Given (x_0, y_0, s_0) feasible, $\mu_0 = \frac{1}{n} \cdot (x^0)^\top s^0$
- For $k = 1, 2, \ldots$
    - $\mu_k = \sigma \mu_{k-1}$, where $\sigma \in (0, 1)$
    - Find Newton direction $(\Delta x^k, \Delta y^k, \Delta s^k)$ by solving

$$
\begin{bmatrix} A & 0 & 0 \\ -Q & A^\top & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} b - Ax^k \\ c - A^\top y^k - s^k + Qx^k \\ \mu_k e - X^k S^k e \end{bmatrix}
$$

Interior-point QP Algorithm

IPM Framework

Fix the barrier parameter μ and make *one* (damped) Newton step towards the solution of FOC. Then reduce the barrier parameter μ and repeat.

- Given (x_0, y_0, s_0) feasible, $\mu_0 = \frac{1}{n} \cdot (x^0)^\top s^0$
- For $k = 1, 2, \ldots$
  - $\mu_k = \sigma \mu_{k-1}$, where $\sigma \in (0, 1)$
  - Find Newton direction $(\Delta x^k, \Delta y^k, \Delta s^k)$ by solving

  $$\begin{bmatrix} A & 0 & 0 \\ -Q & A^\top & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} b - Ax^k \\ c - A^\top y^k - s^k + Qx^k \\ \mu_k e - X^k S^k e \end{bmatrix}$$

  - Find step length α_k so $(x^k + \alpha_k \Delta x^k, y^k + \alpha_k \Delta y^k, s^k + \alpha_k \Delta s^k)$ is feasible.

Interior-point QP Algorithm

IPM Framework

Fix the barrier parameter μ and make *one* (damped) Newton step towards the solution of FOC. Then reduce the barrier parameter μ and repeat.

- Given (x_0, y_0, s_0) feasible, $\mu_0 = \frac{1}{n} \cdot (x^0)^\top s^0$
- For $k = 1, 2, \ldots$
    - $\mu_k = \sigma \mu_{k-1}$, where $\sigma \in (0, 1)$
    - Find Newton direction $(\Delta x^k, \Delta y^k, \Delta s^k)$ by solving

$$\begin{bmatrix} A & 0 & 0 \\ -Q & A^\top & I \\ S^k & 0 & X^k \end{bmatrix} \begin{bmatrix} \Delta x^k \\ \Delta y^k \\ \Delta s^k \end{bmatrix} = \begin{bmatrix} b - Ax^k \\ c - A^\top y^k - s^k + Qx^k \\ \mu_k e - X^k S^k e \end{bmatrix}$$

    - Find step length α_k so $(x^k + \alpha_k \Delta x^k, y^k + \alpha_k \Delta y^k, s^k + \alpha_k \Delta s^k)$ is feasible.
    - Make step $(x^{k+1}, y^{k+1}, s^{k+1}) = (x^k + \alpha_k \Delta x^k, y^k + \alpha_k \Delta y^k, s^k + \alpha_k \Delta s^k)$.

Path-following algorithm

- **Short-step path-following method:** $\mathcal{O}(\sqrt{n})$ complexity result

Path-following algorithm

▶ **Short-step path-following method:** $\mathcal{O}(\sqrt{n})$ complexity result

---

Theorem ([Gondzio, 2012, Thm. 3.1])

Given $\epsilon > 0$, suppose that a feasible starting point $(x^0, y^0, s^0) \in \mathcal{N}_2(0.1)$ satisfies

$$\left(x^0\right)^\top s^0 = n\mu^0, \text{ where } \mu^0 \leq 1/\epsilon^\kappa,$$

for some positive constant κ. Then for some $K = \mathcal{O}(\sqrt{n}\ln(1/\epsilon))$ such that

$$\mu^k \leq \epsilon, \quad \forall k \geq K$$

Path-following algorithm

▶ **Short-step path-following method:** $\mathcal{O}(\sqrt{n})$ complexity result

Theorem ([Gondzio, 2012, Thm. 3.1])

Given $\epsilon > 0$, suppose that a feasible starting point $(x^0, y^0, s^0) \in \mathcal{N}_2(0.1)$ satisfies

$$\left(x^0\right)^\top s^0 = n\mu^0, \ where \ \mu^0 \leq 1/\epsilon^\kappa,$$

for some positive constant κ. Then for some $K = \mathcal{O}(\sqrt{n}\ln(1/\epsilon))$ such that

$$\mu^k \leq \epsilon, \quad \forall k \geq K$$

▶ θ-neighborhood of the central path:
$\mathcal{N}_2(\theta) := \{(x, y, s) \in \mathcal{F}^0 \mid \|XSe - \mu e\| \leq \theta\mu\}$, with $\mu = \frac{1}{n}x^\top s$.

Path-following algorithm

▶ **Short-step path-following method:** $\mathcal{O}(\sqrt{n})$ complexity result

Theorem ([Gondzio, 2012, Thm. 3.1])

Given $\epsilon > 0$, suppose that a feasible starting point $(x^0, y^0, s^0) \in \mathcal{N}_2(0.1)$ satisfies

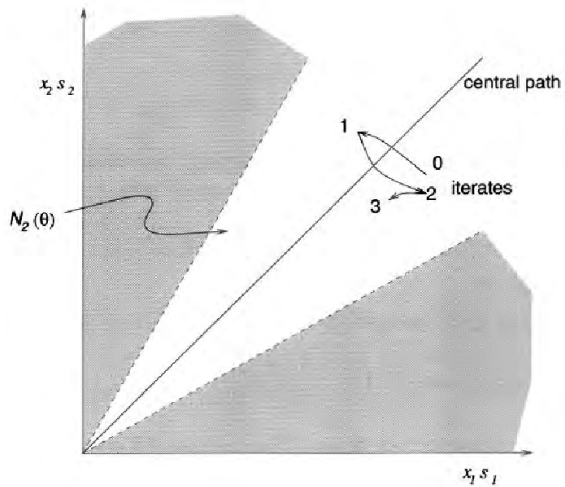$$\left(x^0\right)^\top s^0 = n\mu^0, \text{ where } \mu^0 \leq 1/\epsilon^\kappa,$$

for some positive constant κ. Then for some $K = \mathcal{O}(\sqrt{n}\ln(1/\epsilon))$ such that

$$\mu^k \leq \epsilon, \quad \forall k \geq K$$

▶ θ-neighborhood of the central path:
  $\mathcal{N}_2(\theta) := \{(x, y, s) \in \mathcal{F}^0 \mid \|XSe - \mu e\| \leq \theta\mu\}$, with $\mu = \frac{1}{n}x^\top s$.
▶ Slow progress towards optimality

Augmented system

Newton direction

$$\begin{bmatrix} A & 0 & 0 \\ -Q & A^\top & I \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} b - Ax \\ c - A^\top y - s + Qx \\ \mu_k e - XSe \end{bmatrix} =: \begin{bmatrix} \xi_p \\ \xi_d \\ \xi_\mu \end{bmatrix}$$

use last (complementarity) block to solve for Δs as a function of Δx.

Augmented system

Define $\Theta = XS^{-1}$ (ill-conditioned!). Then Δx and Δy solve the Newton system $\iff$

$$\begin{bmatrix} -Q - \Theta^{-1} & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \xi_d - X^{-1}\xi_\mu \\ \xi_p \end{bmatrix}$$

▶ Newton system is nonsymmetric.
▶ Augmented system is symmetric but indefinite.

Normal equations

Augmented system

$$\begin{bmatrix} -\Theta^{-1} & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \xi_d - X^{-1}\xi_\mu \\ \xi_p \end{bmatrix} =: \begin{bmatrix} g \\ \xi_p \end{bmatrix}$$

Normal equations

Eliminate Δx to arrive at the *Normal equations*

$$(A\Theta A^\top)\Delta y = A\Theta g + \xi_p$$

Normal equations

Augmented system

$$\begin{bmatrix} -\Theta^{-1} & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} \xi_d - X^{-1}\xi_\mu \\ \xi_p \end{bmatrix} =: \begin{bmatrix} g \\ \xi_p \end{bmatrix}$$

Normal equations

Eliminate Δx to arrive at the *Normal equations*

$$(A\Theta A^\top)\Delta y = A\Theta g + \xi_p$$

▶ $A\Theta A^\top$ is symmetric and positive semidefinite. (Finally!)
▶ Normal equations in QP $(A(Q + \Theta)A^\top)\Delta y = g$ are generally nearly dense, even when A and Q are sparse.
▶ LP: Normal equations are often used.
▶ QP: usually use the indefinite augmented system.

Outline

IPM for NLP

► Convex NLP

$$\text{minimize} \quad f(x) \qquad \text{subject to} \quad g(x) \leq 0$$

$$\text{minimize} \quad f(x) \qquad \text{subject to} \quad g(x) + z = 0, z \geq 0$$

IPM for NLP

▶ Convex NLP

$$\text{minimize} \quad f(x) \qquad \text{subject to} \quad g(x) \leq 0$$

$$\text{minimize} \quad f(x) \qquad \text{subject to} \quad g(x) + z = 0, z \geq 0$$

▶ Replace inequality $z \geq 0$ with logarithmic barrier

$$\text{minimize} f(x) - \mu \sum_{i=1}^{m} \ln(z_i) \quad \text{subject to} \quad g(x) + z = 0$$

IPM for NLP

- Convex NLP
$$\text{minimize} \quad f(x) \qquad \text{subject to} \quad g(x) \leq 0$$
$$\text{minimize} \quad f(x) \qquad \text{subject to} \quad g(x) + z = 0, z \geq 0$$

- Replace inequality $z \geq 0$ with logarithmic barrier
$$\text{minimize} f(x) - \mu \sum_{i=1}^{m} \ln(z_i) \quad \text{subject to} \quad g(x) + z = 0$$

- Write out Lagrangian
$$L(x, y, z, \mu) = f(x) + y^{\top}(g(x) + z) - \mu \sum_{i=1}^{m} \ln(z_i)$$

IPM for NLP

▶ Write conditions for stationary point

$$\nabla_x L(x, z, y) = \nabla f(x) + J_g(x)^\top y = 0$$
$$\nabla_y L(x, z, y) = g(x) + z = 0$$
$$\nabla_z L(x, z, y) = y - \mu Z^{-1} e = 0$$

IPM for NLP

▶ Write conditions for stationary point

$$\nabla_x L(x, z, y) = \nabla f(x) + J_g(x)^\top y = 0$$
$$\nabla_y L(x, z, y) = g(x) + z = 0$$
$$\nabla_z L(x, z, y) = y - \mu Z^{-1} e = 0$$

▶ Write KKT system

$$\nabla f(x) + J_g(x)^\top y = 0,$$
$$g(x) + z = 0$$
$$YZe = \mu e$$

Newton for KKT of NLP

▶ Apply Newton method for KKT system

Newton for KKT of NLP

▶ Apply Newton method for KKT system

▶ Jacobian matrix of KKT system

$$J_F(x, z, y) = \begin{bmatrix} Q(x, y) & J_g(x)^\top & 0 \\ J_g(x) & 0 & I \\ 0 & Z & Y \end{bmatrix}$$

where $Q(x, y) = \nabla^2 f(x) + \sum_{i=1}^m y_i \nabla^2 g_i(x)$ is the Hessian of L

Newton for KKT of NLP

▶ Apply Newton method for KKT system

▶ Jacobian matrix of KKT system

$$J_F(x, z, y) = \begin{bmatrix} Q(x,y) & J_g(x)^\top & 0 \\ J_g(x) & 0 & I \\ 0 & Z & Y \end{bmatrix}$$

where $Q(x,y) = \nabla^2 f(x) + \sum_{i=1}^m y_i \nabla^2 g_i(x)$ is the Hessian of L

▶ Newton step for KKT system

$$\begin{bmatrix} Q(x,y) & J_g(x)^\top & 0 \\ J_g(x) & 0 & I \\ 0 & Z & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - z \\ \mu e - YZe \end{bmatrix}$$

From QP to NLP

▶ Newton direction for NLP

$$\begin{bmatrix} Q(x,y) & J_g(x)^\top & 0 \\ J_g(x) & 0 & I \\ 0 & Z & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - z \\ \mu e - YZe \end{bmatrix}$$

From QP to NLP

▶ Newton direction for NLP

$$\begin{bmatrix} Q(x,y) & J_g(x)^\top & 0 \\ J_g(x) & 0 & I \\ 0 & Z & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - z \\ \mu e - YZe \end{bmatrix}$$

▶ Augmented system for NLP

$$\begin{bmatrix} Q(x,y) & J_g(x)^\top \\ J_g(x) & -ZY^{-1} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - \mu Y^{-1} e \end{bmatrix}$$

From QP to NLP

▶ Newton direction for NLP

$$
\begin{bmatrix} Q(x,y) & J_g(x)^\top & 0 \\ J_g(x) & 0 & I \\ 0 & Z & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - z \\ \mu e - YZe \end{bmatrix}
$$

▶ Augmented system for NLP

$$
\begin{bmatrix} Q(x,y) & J_g(x)^\top \\ J_g(x) & -ZY^{-1} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - \mu Y^{-1} e \end{bmatrix}
$$

▶ Need to compute $Q(x,y)$ and $J_g(x)$ at each iteration

From QP to NLP

▶ Newton direction for NLP

$$\begin{bmatrix} Q(x,y) & J_g(x)^\top & 0 \\ J_g(x) & 0 & I \\ 0 & Z & Y \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - z \\ \mu e - YZe \end{bmatrix}$$

▶ Augmented system for NLP

$$\begin{bmatrix} Q(x,y) & J_g(x)^\top \\ J_g(x) & -ZY^{-1} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} = \begin{bmatrix} -\nabla f(x) - J_g(x)^\top y \\ -g(x) - \mu Y^{-1} e \end{bmatrix}$$

▶ Need to compute $Q(x,y)$ and $J_g(x)$ at each iteration

▶ Caveat: use trust region method to choose stepsize.

Outline

Self-concordant function

Definition

Function f is *self-concordant* if for some constant $M_f \geq 0$, the inequality

$$\nabla^3 f(x)[u, u, u] \leq M_f \|u\|_{\nabla^2 f(x)}^{3/2}$$

holds for any $x \in \operatorname{dom} f$ and $u \in R^n$.

▶ A self-concordant function is always well approximated by a quadratic model because the error of such an approximation can be bounded by the $\|u\|_{\nabla^2 f(x)}^{3/2}$

Self-concordant function

Definition

Function f is *self-concordant* if for some constant $M_f \geq 0$, the inequality

$$\nabla^3 f(x)[u, u, u] \leq M_f \|u\|^{3/2}_{\nabla^2 f(x)}$$

holds for any $x \in \operatorname{dom} f$ and $u \in R^n$.

▶ A self-concordant function is always well approximated by a quadratic model because the error of such an approximation can be bounded by the $\|u\|^{3/2}_{\nabla^2 f(x)}$

Self-concordant function

Definition

Function f is *self-concordant* if for some constant $M_f \geq 0$, the inequality

$$\nabla^3 f(x)[u, u, u] \leq M_f \|u\|_{\nabla^2 f(x)}^{3/2}$$

holds for any $x \in \text{dom} f$ and $u \in R^n$.

▶ A self-concordant function is always well approximated by a quadratic model because the error of such an approximation can be bounded by the $\|u\|_{\nabla^2 f(x)}^{3/2}$

Theorem ([Boyd and Vandenberghe, 2004, Section 11.5])

Newton's method with line search finds an ε approximate solution in less than $T := \text{constant} \times (f(x_0) - f^\star) + \log_2 \log_2 \frac{1}{\varepsilon}$ iterations.

Log-barrier is self-concordant

Theorem

The barrier function $-\ln(x)$ *is self-concordant in* $\mathbf{R}_+$.

Log-barrier is self-concordant

Theorem

The barrier function $-\ln(x)$ is self-concordant in $\mathbf{R}_+$.

Proof.

Consider $f(x) = -\ln(x)$, then

$$f'(x) = -\frac{1}{x}, \quad f''(x) = \frac{1}{x^2}, \quad f'''(x) = -\frac{2}{x^3}$$

Log-barrier is self-concordant

Theorem

The barrier function $-\ln(x)$ is self-concordant in $\mathbf{R}_+$.

Proof.

Consider $f(x) = -\ln(x)$, then

$$f'(x) = -\frac{1}{x}, \quad f''(x) = \frac{1}{x^2}, \quad f'''(x) = -\frac{2}{x^3}$$

Compute and check that self-concordance holds with $M_f = 2$. $\qquad\square$

▶ $-\ln(1/x^{\alpha})$, with $\alpha \in (0, \infty)$ is not self-concordant in $\mathbf{R}_+$.

Log-barrier is self-concordant

Theorem

The barrier function $-\ln(x)$ is self-concordant in $\mathbf{R}_+$.

Proof.

Consider $f(x) = -\ln(x)$, then

$$f'(x) = -\frac{1}{x}, \quad f''(x) = \frac{1}{x^2}, \quad f'''(x) = -\frac{2}{x^3}$$

Compute and check that self-concordance holds with $M_f = 2$. $\qquad\square$

- $-\ln(1/x^\alpha)$, with $\alpha \in (0, \infty)$ is not self-concordant in $\mathbf{R}_+$.
- $\exp(1/x)$ is not self-concordant in $\mathbf{R}_+$.