School of Mathematics

IPMs for Convex Optimization: QP, NLP, SOCP and SDP

Jacek Gondzio
Email: J.Gondzio@ed.ac.uk
URL: http://www.maths.ed.ac.uk/~gondzio

Outline

- **IPM for Quadratic and Nonlinear Programming**
  - quadratic forms, NLP notation
  - duality, Lagrangian, first order optimality conditions
  - primal-dual framework
- **Self-concordant barrier**
- **Second-Order Cone Programming**
  - example cones
  - example SOCP problems
  - logarithmic barrier function
- **Semidefinite Programming**
  - background (linear matrix inequalities)
  - example SDP problems
  - logarithmic barrier function
- **Final Comments**

IPM for Convex QP

Convex Quadratic Programs

Def. A matrix $Q \in \mathcal{R}^{n \times n}$ is positive semidefinite if $x^T Q x \geq 0$ for any $x \neq 0$. We write $Q \succeq 0$.

The quadratic function

$$f(x) = x^T Q \, x$$

is convex if and only if the matrix Q is positive definite.
In such case the quadratic programming problem

$$\begin{array}{ll} \min & c^T x + \frac{1}{2} x^T Q \, x \\ \text{s.t.} & Ax = b, \\ & x \geq 0, \end{array}$$

is well defined.

If there exists a *feasible* solution to it,
then there exists an *optimal* solution.

QP with IPMs

Apply the *usual* procedure:

- replace inequalities with log barriers;
- form the Lagrangian;
- write the first order optimality conditions;
- apply Newton method to them.

Replace the **primal** QP

$$\begin{array}{ll} \min & c^T x + \frac{1}{2} x^T Q\, x \\ \text{s.t.} & Ax \;=\; b, \\ & x \geq 0, \end{array}$$

with the **primal barrier QP**

$$\begin{array}{ll} \min & c^T x + \frac{1}{2} x^T Q\, x - \sum_{j=1}^{n} \ln x_j \\ \text{s.t.} & Ax \;= b. \end{array}$$

First Order Optimality Conditions

Consider the **primal barrier quadratic program**

$$\begin{array}{ll} \min & c^T x + \frac{1}{2} x^T Q\, x - \mu \sum_{j=1}^{n} \ln x_j \\ \text{s.t.} & Ax \;=\; b, \end{array}$$

where $\mu \geq 0$ is a barrier parameter.

Write out the **Lagrangian**

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

First Order Optimality Conditions (cont'd)

The conditions for a stationary point of the Lagrangian:

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

are

$$\begin{array}{rcrcl} \nabla_x L(x, y, \mu) & = & c - A^T y - \mu X^{-1} e + Qx & = & 0 \\ \nabla_y L(x, y, \mu) & = & Ax - b & = & 0, \end{array}$$

where $X^{-1} = diag\{x_1^{-1}, x_2^{-1}, \cdots, x_n^{-1}\}$.
Let us denote

$$s = \mu X^{-1} e, \quad \text{i.e.} \quad XSe = \mu e.$$

The **First Order Optimality Conditions** are:

$$\begin{array}{rcl} Ax & = & b, \\ A^T y + s - Qx & = & c, \\ XSe & = & \mu e. \end{array}$$

Apply Newton Method to the FOC

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

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

where $F : \mathcal{R}^{2n+m} \mapsto \mathcal{R}^{2n+m}$ is an application defined as follows:

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

Actually, the first two terms of it are *linear*; only the last one, corresponding to the complementarity condition, is *nonlinear*.
Note that

$$\nabla F(x, y, s) = \begin{bmatrix} A & 0 & 0 \\ -Q & A^T & I \\ S & 0 & X \end{bmatrix}.$$

Newton Method for the FOC (cont'd)

Thus, for a given point (x, y, s)
we find the Newton direction $(\Delta x, \Delta y, \Delta s)$
by solving the system of linear equations:

$$\begin{bmatrix} A & 0 & 0 \\ -Q & A^T & I \\ S & 0 & X \end{bmatrix} \cdot \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} b - Ax \\ c - A^T y - s + Qx \\ \mu e - XSe \end{bmatrix}.$$

Interior-Point QP Algorithm

Initialize

$$k = 0, \quad (x^0, y^0, s^0) \in \mathcal{F}^0, \quad \mu_0 = \frac{1}{n} \cdot (x^0)^T s^0, \quad \alpha_0 = 0.9995$$

Repeat until optimality

$k = k + 1$
$\mu_k = \sigma \mu_{k-1}$, where $\sigma \in (0, 1)$
$\Delta =$ Newton direction towards μ-center

Ratio test:

$$\alpha_P := \max \{\alpha > 0 : \; x + \alpha \Delta x \geq 0\},$$
$$\alpha_D := \max \{\alpha > 0 : \; s + \alpha \Delta s \geq 0\}.$$

Make step:

$$x^{k+1} = x^k + \alpha_0 \alpha_P \Delta x,$$
$$y^{k+1} = y^k + \alpha_0 \alpha_D \Delta y,$$
$$s^{k+1} = s^k + \alpha_0 \alpha_D \Delta s.$$

From LP to QP

QP problem

$$\begin{array}{ll} \min & c^T x + \frac{1}{2} x^T Q\, x \\ \text{s.t.} & Ax = b, \\ & x \geq 0. \end{array}$$

First order conditions (for barrier problem)

$$\begin{array}{rcl} Ax & = & b, \\ A^T y + s - Qx & = & c, \\ XSe & = & \mu e. \end{array}$$

IPMs for Convex NLP

Convex Nonlinear Optimization

Consider the nonlinear optimization problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & g(x) \leq 0, \end{array}$$

where $x \in \mathcal{R}^n$, and $f : \mathcal{R}^n \mapsto \mathcal{R}$ and $g : \mathcal{R}^n \mapsto \mathcal{R}^m$ are convex, twice differentiable.

Assumptions:
f and g are convex
$\Rightarrow$ If there exists a **local** minimum then it is a **global** one.
f and g are twice differentiable
$\Rightarrow$ We can use the **second order Taylor approximations**.

Some additional (technical) conditions
$\Rightarrow$ We need them to prove that the point which satisfies the first order optimality conditions is the optimum. *We won't use them in this course.*

Nonlinear Optimization with IPMs

Nonlinear Optimization via QPs:
Sequential Quadratic Programming (SQP).
Repeat until optimality:

- approximate NLP (locally) with a QP;
- solve (approximately) the QP.

Nonlinear Optimization with IPMs:
works similarly to SQP scheme.

However, the (local) QP approximations are not solved to optimality. Instead, only one step in the Newton direction corresponding to a given QP approximation is made and the new QP approximation is computed.

NLP Notation

Consider the nonlinear optimization problem

$$\min \ f(x) \quad \text{s.t.} \quad g(x) \leq 0,$$

where $x \in \mathcal{R}^n$, and $f : \mathcal{R}^n \mapsto \mathcal{R}$ and $g : \mathcal{R}^n \mapsto \mathcal{R}^m$ are convex, twice differentiable.

The vector-valued function $g : \mathcal{R}^n \mapsto \mathcal{R}^m$ has a derivative

$$A(x) = \nabla g(x) = \left[\frac{\partial g_i}{\partial x_j} \right]_{i=1..m,\, j=1..n} \in \mathcal{R}^{m \times n}$$

which is called the **Jacobian** of g.

NLP Notation (cont'd)

The Lagrangian associated with the NLP is:

$$\mathcal{L}(x,y) = f(x) + y^T g(x),$$

where $y \in \mathcal{R}^m, y \geq 0$ are Lagrange multipliers (dual variables).

The first derivatives of the Lagrangian:

$$\begin{aligned} \nabla_x \mathcal{L}(x,y) &= \nabla f(x) + \nabla g(x)^T y \\ \nabla_y \mathcal{L}(x,y) &= g(x). \end{aligned}$$

The **Hessian** of the Lagrangian, $Q(x,y) \in \mathcal{R}^{n \times n}$:

$$Q(x,y) = \nabla^2_{xx} \mathcal{L}(x,y) = \nabla^2 f(x) + \sum_{i=1}^{m} y_i \nabla^2 g_i(x).$$

Convexity in NLP

Lemma 2: If $f : \mathcal{R}^n \mapsto \mathcal{R}$ and $g : \mathcal{R}^n \mapsto \mathcal{R}^m$ are convex, twice differentiable, then the **Hessian** of the Lagrangian

$$Q(x, y) = \nabla^2 f(x) + \sum_{i=1}^{m} y_i \nabla^2 g_i(x)$$

is positive semidefinite for any x and any $y \geq 0$. If f is strictly convex, then $Q(x, y)$ is positive definite for any x and any $y \geq 0$.

Proof: The convexity of f implies that $\nabla^2 f(x)$ is positive semidefinite for any x. Similarly, the convexity of g implies that for all $i = 1, 2, ..., m$, $\nabla^2 g_i(x)$ is positive semidefinite for any x. Since $y_i \geq 0$ for all $i = 1, 2, ..., m$ and $Q(x, y)$ is the sum of positive semidefinite matrices, we have that $Q(x, y)$ is positive semidefinite.

If f is strictly convex, then $\nabla^2 f(x)$ is positive definite and so is $Q(x, y)$.

IPM for NLP

Add slack variables to nonlinear inequalities:

$$\begin{array}{rll} \min & f(x) & \\ \text{s.t.} & g(x) + z & = 0 \\ & z & \geq 0, \end{array}$$

where $z \in \mathcal{R}^m$. Replace inequality $z \geq 0$ with the logarithmic barrier:

$$\begin{array}{lll} \min & f(x) - \mu \sum_{i=1}^{m} \ln z_i & \\ \text{s.t.} & g(x) + z & = 0. \end{array}$$

Write out the **Lagrangian**

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

IPM for NLP

For the **Lagrangian**

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

write the conditions for a stationary point

$$\begin{array}{rcrcl} \nabla_x L(x, y, z, \mu) & = & \nabla f(x) + \nabla g(x)^T y & = & 0 \\ \nabla_y L(x, y, z, \mu) & = & g(x) + z & = & 0 \\ \nabla_z L(x, y, z, \mu) & = & y - \mu Z^{-1} e & = & 0, \end{array}$$

where $Z^{-1} = diag\{z_1^{-1}, z_2^{-1}, \cdots, z_m^{-1}\}$.

The **First Order Optimality Conditions** are:

$$\begin{array}{rcl} \nabla f(x) + \nabla g(x)^T y & = & 0, \\ g(x) + z & = & 0, \\ YZe & = & \mu e. \end{array}$$

Newton Method for the FOC

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

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

where $F : \mathcal{R}^{n+2m} \mapsto \mathcal{R}^{n+2m}$ is an application defined as follows:

$$F(x, y, z) = \begin{bmatrix} \nabla f(x) \; + \; \nabla g(x)^T y \\ g(x) \; + \; z \\ YZe \; - \; \mu e \end{bmatrix}.$$

Note that all three terms of it are *nonlinear*.
(In LP and QP the first two terms were *linear*.)

Newton Method for the FOC

Observe that

$$\nabla F(x,y,z) = \begin{bmatrix} Q(x,y) & A(x)^T & 0 \\ A(x) & 0 & I \\ 0 & Z & Y \end{bmatrix},$$

where $A(x)$ is the **Jacobian** of g
and $Q(x,y)$ is the **Hessian** of $\mathcal{L}$.

They are defined as follows:

$$\begin{aligned} A(x) &= \nabla g(x) & \in \mathcal{R}^{m \times n} \\ Q(x,y) &= \nabla^2 f(x) + \sum_{i=1}^{m} y_i \nabla^2 g_i(x) & \in \mathcal{R}^{n \times n} \end{aligned}$$

Newton Method (cont'd)

For a given point (x, y, z) we find the Newton direction $(\Delta x, \Delta y, \Delta z)$ by solving the system of linear equations:

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

Using the third equation we eliminate

$$\Delta z = \mu Y^{-1} e - Ze - ZY^{-1} \Delta y,$$

from the second equation and get

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

Interior-Point NLP Algorithm

Initialize

$k = 0$

(x^0, y^0, z^0) such that $y^0 > 0$ and $z^0 > 0$, $\mu_0 = \frac{1}{m} \cdot (y^0)^T z^0$

Repeat until optimality

$k = k + 1$

$\mu_k = \sigma \mu_{k-1}$, where $\sigma \in (0,1)$

Compute $A(x)$ and $Q(x,y)$

$\Delta =$ Newton direction towards μ-center

Ratio test:

$$\begin{aligned} \alpha_1 &:= \max\{\alpha > 0 : \; y + \alpha\Delta y \geq 0\}, \\ \alpha_2 &:= \max\{\alpha > 0 : \; z + \alpha\Delta z \geq 0\}. \end{aligned}$$

Choose the step: (use trust region or line search) $\alpha \leq \min\{\alpha_1, \alpha_2\}$.

Make step:

$$\begin{aligned} x^{k+1} &= x^k + \alpha\Delta x, \\ y^{k+1} &= y^k + \alpha\Delta y, \\ z^{k+1} &= z^k + \alpha\Delta z. \end{aligned}$$

From QP to NLP

Newton direction for **QP**

$$\begin{bmatrix} -Q & A^T & I \\ A & 0 & 0 \\ S & 0 & X \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta s \end{bmatrix} = \begin{bmatrix} \xi_d \\ \xi_p \\ \xi_\mu \end{bmatrix}.$$

Augmented system for QP

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

From QP to NLP

Newton direction for **NLP**

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

Augmented system for NLP

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

Conclusion:
NLP is a natural extension of QP.

Newton Method and Self-concordant Barriers

Another View of Newton M. for Optimization

Newton Method for Optimization

Let $f : \mathcal{R}^n \mapsto \mathcal{R}$ be a twice continuously differentiable function. Suppose we build a quadratic model $\tilde{f}$ of f around a given point x^k, i.e., we define $\Delta x = x - x^k$ and write:

$$\tilde{f}(x) = f(x^k) + \nabla f(x^k)^T \Delta x + \frac{1}{2} \Delta x^T \nabla^2 f(x^k) \Delta x$$

Now we **optimize the model** $\tilde{f}$ instead of **optimizing** f.
A minimum (or, more generally, a stationary point) of the quadratic model satisfies:

$$\nabla \tilde{f}(x) = \nabla f(x^k) + \nabla^2 f(x^k) \Delta x = 0,$$

i.e.

$$\Delta x = x - x^k = -(\nabla^2 f(x^k))^{-1} \nabla f(x^k),$$

which reduces to the usual equation:

$$x^{k+1} = x^k - (\nabla^2 f(x^k))^{-1} \nabla f(x^k).$$

Self-concordant Functions

There is a nice property of the function that is responsible for a good behaviour of the Newton method.

Def Let $C \in \mathcal{R}^n$ be an open nonempty convex set.
Let $f : C \mapsto \mathcal{R}$ be a three times continuously differentiable convex function.
A function f is called **self-concordant** if there exists a constant $p > 0$ such that

$$|\nabla^3 f(x)[h, h, h]| \leq 2p^{-1/2}(\nabla^2 f(x)[h, h])^{3/2},$$

$\forall x \in C, \forall h : \; x + h \in C$.
(We then say that f is p-self-concordant).

Note that a self-concordant function is always well approximated by the quadratic model because the error of such an approximation can be bounded by the 3/2 power of $\nabla^2 f(x)[h, h]$.

Self-concordant Barriers

Lemma
The barrier function $-\log x$ is self-concordant on $\mathcal{R}_+$.

Proof Consider $f(x) = -\log x$.
Compute
$f^{'}(x) = -x^{-1}$, $f^{''}(x) = x^{-2}$ and $f^{'''}(x) = -2x^{-3}$
and check that the self-concordance condition is satisfied for $p = 1$.

Lemma
The barrier function $1/x^{\alpha}$, with $\alpha \in (0, \infty)$ is not self-concordant on $\mathcal{R}_+$.

Lemma
The barrier function $e^{1/x}$ is not self-concordant on $\mathcal{R}_+$.

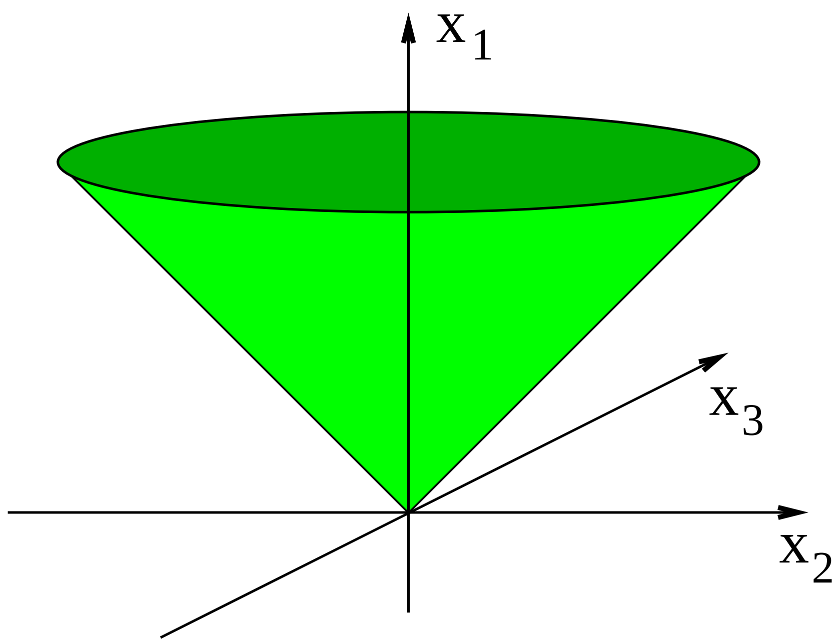
Use self-concordant barriers in optimization

Second-Order Cone Programming (SOCP)

Cones: Background

Def. A set $K \in \mathcal{R}^n$ is called a cone if for any $x \in K$ and for any $\lambda \geq 0$, $\lambda x \in K$.

Convex Cone:



Example:

$$K = \{x \in \mathcal{R}^n : x_1^2 \geq \sum_{j=2}^{n} x_j^2,\, x_1 \geq 0\}.$$

Example: Three Cones

R_+:

$$R_+ = \{x \in \mathcal{R} : x \geq 0\}.$$

Quadratic Cone:

$$K_q = \{x \in \mathcal{R}^n : x_1^2 \geq \sum_{j=2}^{n} x_j^2,\ x_1 \geq 0\}.$$

Rotated Quadratic Cone:

$$K_r = \{x \in \mathcal{R}^n : 2x_1 x_2 \geq \sum_{j=3}^{n} x_j^2,\ x_1, x_2 \geq 0\}.$$

Matrix Representation of Cones

Each of the three most common cones has a matrix representation using orthogonal matrices T and/or Q.
(Orthogonal matrix: $Q^T Q = I$).

Quadratic Cone K_q. Define

$$Q = \begin{bmatrix} 1 & & & & \\ & -1 & & & \\ & & -1 & & \\ & & & \ddots & \\ & & & & -1 \end{bmatrix}$$

and write:

$$K_q = \{x \in \mathcal{R}^n : x^T Q x \geq 0,\ x_1 \geq 0\}.$$

Example: $x_1^2 \geq x_2^2 + x_3^2 + \cdots + x_n^2.$

Matrix Representation of Cones (cont'd)

Rotated Quadratic Cone K_r. Define

$$Q = \begin{bmatrix} 0 & 1 & & & \\ 1 & 0 & & & \\ & & -1 & & \\ & & & \ddots & \\ & & & & -1 \end{bmatrix}$$

and write:

$$K_r = \{x \in \mathcal{R}^n : x^T Q x \geq 0,\ x_1, x_2 \geq 0\}.$$

Example: $2x_1 x_2 \geq x_3^2 + x_4^2 + \cdots + x_n^2.$

Matrix Representation of Cones (cont'd)

Consider a linear transformation $T : \mathcal{R}^2 \mapsto \mathcal{R}^2$:

$$T_2 = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}.$$

It corresponds to a rotation by $\pi/4$. Indeed, write:

$$\begin{bmatrix} z \\ y \end{bmatrix} = T_2 \begin{bmatrix} u \\ v \end{bmatrix}$$

that is

$$z = \frac{u+v}{\sqrt{2}}, \quad y = \frac{u-v}{\sqrt{2}}$$

to get

$$2yz = u^2 - v^2.$$

Matrix Representation of Cones (cont'd)

Now, define

$$T = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & & & \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \end{bmatrix}$$

and observe that the rotated quadratic cone satisfies

$$Tx \in K_r \quad \text{iff} \quad x \in K_q.$$

Example: Conic constraint

Consider a constraint:

$$\frac{1}{2}\|x\|^2 + a^T x \le b.$$

Observe that $g(x) = \frac{1}{2}x^T x + a^T x - b$ is convex hence the constraint defines a convex set.
The constraint may be reformulated as an intersection of an affine (linear) constraint and a quadratic one:

$$\begin{aligned} a^T x + z &= b \\ y &= 1 \\ \|x\|^2 &\le 2yz, \;\; y, z \ge 0. \end{aligned}$$

Example: Conic constraint (cont'd)

Now, substitute:

$$z = \frac{u+v}{\sqrt{2}}, \quad y = \frac{u-v}{\sqrt{2}}$$

to get

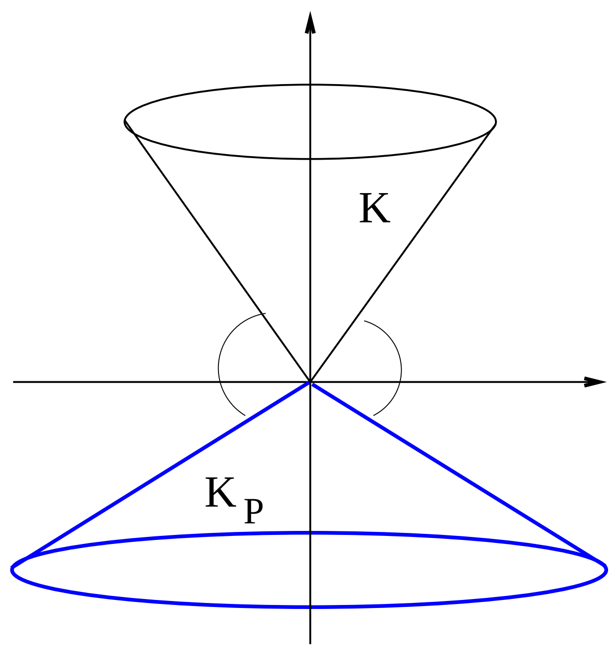
$$\begin{aligned} a^T x + \frac{u+v}{\sqrt{2}} &= b \\ u - v &= \sqrt{2} \\ \|x\|^2 + v^2 &\leq u^2. \end{aligned}$$

Dual Cone

Let $K \in \mathcal{R}^n$ be a cone.

Def. The set:
$$K_* := \{s \in \mathcal{R}^n : s^T x \geq 0, \; \forall x \in K\}$$
is called the **dual** cone.

Def. The set:
$$K_P := \{s \in \mathcal{R}^n : s^T x \leq 0, \; \forall x \in K\}$$
is called the **polar** cone (Fig below).



Conic Optimization

Consider an optimization problem:

$$\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax = b, \\ & x \in K, \end{array}$$

where K is a convex closed cone.

We assume that

$$K = K^1 \times K^2 \times \cdots \times K^k,$$

that is, cone K is a product of several individual cones each of which is one of the three cones defined earlier.

Primal and Dual SOCPs

Consider a **primal** SOCP

$$\begin{array}{rl} \min & c^T x \\ \text{s.t.} & Ax = b, \\ & x \in K, \end{array}$$

where K is a convex closed cone.

The associated **dual** SOCP

$$\begin{array}{rl} \max & b^T y \\ \text{s.t.} & A^T y + s = c, \\ & s \in K_*. \end{array}$$

Weak Duality:
If (x, y, s) is a primal-dual feasible solution, then

$$c^T x - b^T y = x^T s \geq 0.$$

IPM for Conic Optimization

Conic Optimization problems can be solved in polynomial time with IPMs.

Consider a quadratic cone

$$K_q = \{(x,t) : x \in \mathcal{R}^{n-1}, t \in \mathcal{R},\ t^2 \geq \|x\|^2,\ t \geq 0\},$$

and define the (convex) **logarithmic barrier function** for this cone $f : \mathcal{R}^n \mapsto \mathcal{R}$

$$f(x,t) = \begin{cases} -\ln(t^2 - \|x\|^2) & \text{if } \|x\| < t \\ +\infty & \text{otherwise.} \end{cases}$$

Theorem:
$f(x,t)$ is a self-concordant barrier on K_q.

Exercise: Prove it in case $n = 2$.

Semidefinite Programming (SDP)

SDP: Background

Def. A matrix $H \in \mathcal{R}^{n \times n}$ is positive semidefinite if $x^T H x \geq 0$ for any $x \neq 0$. We write $H \succeq 0$.

Def. A matrix $H \in \mathcal{R}^{n \times n}$ is positive definite if $x^T H x > 0$ for any $x \neq 0$. We write $H \succ 0$.

We denote with $\mathcal{SR}^{n \times n}$ ($\mathcal{SR}_{+}^{n \times n}$) the set of symmetric and symmetric positive semidefinite matrices.

Let $U, V \in \mathcal{SR}^{n \times n}$. We define the inner product between U and V as $U \bullet V = trace(U^T V)$, where $trace(H) = \sum_{i=1}^{n} h_{ii}$.

The associated norm is the Frobenius norm,
written $\|U\|_F = (U \bullet U)^{1/2}$ (or just $\|U\|$).

Linear Matrix Inequalities

Def. *Linear Matrix Inequalities*

Let $U, V \in \mathcal{SR}^{n \times n}$.

We write $U \succeq V$ iff $U - V \succeq 0$.

We write $U \succ V$ iff $U - V \succ 0$.

We write $U \preceq V$ iff $U - V \preceq 0$.

We write $U \prec V$ iff $U - V \prec 0$.

Properties

1. If $P \in \mathcal{R}^{m \times n}$ and $Q \in \mathcal{R}^{n \times m}$, then $trace(PQ) = trace(QP)$.

2. If $U, V \in \mathcal{SR}^{n \times n}$, and $Q \in \mathcal{R}^{n \times n}$ is orthogonal (i.e. $Q^T Q = I$), then $U \bullet V = (Q^T U Q) \bullet (Q^T V Q)$.
More generally, if P is nonsingular, then
$U \bullet V = (P U P^T) \bullet (P^{-T} V P^{-1})$.

3. Every $U \in \mathcal{SR}^{n \times n}$ can be written as $U = Q \Lambda Q^T$, where Q is orthogonal and Λ is diagonal. Then $UQ = Q\Lambda$.
In other words the columns of Q are the eigenvectors, and the diagonal entries of Λ the corresponding eigenvalues of U.

4. If $U \in \mathcal{SR}^{n \times n}$ and $U = Q \Lambda Q^T$, then
$trace(U) = trace(\Lambda) = \sum_i \lambda_i$.

Properties (cont'd)

5. For $U \in \mathcal{SR}^{n\times n}$, the following are equivalent:
   (i) $U \succeq 0$ ($U \succ 0$)
   (ii) $x^T U x \geq 0, \forall\, x \in \mathcal{R}^n$ ($x^T U x > 0, \forall\, 0 \neq x \in \mathcal{R}^n$).
   (iii) If $U = Q \Lambda Q^T$, then $\Lambda \succeq 0$ ($\Lambda \succ 0$).
   (iv) $U = P^T P$ for some matrix P ($U = P^T P$ for some square nonsingular matrix P).

6. Every $U \in \mathcal{SR}^{n\times n}$ has a square root $U^{1/2} \in \mathcal{SR}^{n\times n}$.
Proof: From Property 5 (ii) we get $U = Q \Lambda Q^T$.
Take $U^{1/2} = Q \Lambda^{1/2} Q^T$, where $\Lambda^{1/2}$ is the diagonal matrix whose diagonal contains the (nonnegative) square roots of the eigenvalues of U, and verify that $U^{1/2} U^{1/2} = U$.

Properties (cont'd)

7. Suppose

$$U = \begin{bmatrix} A & B^T \\ B & C \end{bmatrix},$$

where A and C are symmetric and $A \succ 0$.
Then $U \succeq 0$ ($U \succ 0$) iff $C - BA^{-1}B^T \succeq 0$ ($\succ 0$).
The matrix $C - BA^{-1}B^T$ is called the *Schur complement* of A in U.

Proof: follows easily from the factorization:

$$\begin{bmatrix} A & B^T \\ B & C \end{bmatrix} = \begin{bmatrix} I & 0 \\ BA^{-1} & I \end{bmatrix} \begin{bmatrix} A & 0 \\ 0 & C - BA^{-1}B^T \end{bmatrix} \begin{bmatrix} I & A^{-1}B^T \\ 0 & I \end{bmatrix}.$$

8. If $U \in \mathcal{SR}^{n\times n}$ and $x \in \mathcal{R}^n$, then $x^T U x = U \bullet xx^T$.

Primal-Dual Pair of SDPs

Primal

$$\begin{array}{ll} \min & C \bullet X \\ \text{s.t.} & A_i \bullet X = b_i, \;\; i = 1..m \\ & X \succeq 0; \end{array}$$

Dual

$$\begin{array}{ll} \max & b^T y \\ \text{s.t.} & \sum_{i=1}^m y_i A_i + S = C, \\ & S \succeq 0, \end{array}$$

where $A_i \in \mathcal{SR}^{n \times n}$, $b \in \mathcal{R}^m$, $C \in \mathcal{SR}^{n \times n}$ are given;
and $X, S \in \mathcal{SR}^{n \times n}$, $y \in \mathcal{R}^m$ are the variables.

Simplified notation:

Primal

$$\begin{array}{ll} \min & C \bullet X \\ \text{s.t.} & \mathcal{A}X = b, \\ & X \succeq 0; \end{array}$$

Dual

$$\begin{array}{ll} \max & b^T y \\ \text{s.t.} & \mathcal{A}^* y + S = C, \\ & S \succeq 0. \end{array}$$

Theorem: Weak Duality in SDP

If X is feasible in the primal and (y, S) in the dual, then

$$C \bullet X - b^T y = X \bullet S \geq 0.$$

Proof:

$$\begin{aligned} C \bullet X - b^T y &= (\sum_{i=1}^{m} y_i A_i + S) \bullet X - b^T y \\ &= \sum_{i=1}^{m} (A_i \bullet X)\, y_i + S \bullet X - b^T y \\ &= S \bullet X = X \bullet S. \end{aligned}$$

Further, since X is positive semidefinite, it has a square root $X^{1/2}$ (Property 6), and so

$$X \bullet S = trace(XS) = trace(X^{1/2} X^{1/2} S) = trace(X^{1/2} S X^{1/2}) \geq 0.$$

We use Property 1 and the fact that S and $X^{1/2}$ are positive semidefinite, hence $X^{1/2} S X^{1/2}$ is positive semidefinite and its trace is nonnegative.

SDP Example 1: Minimize the Max. Eigenvalue

We wish to choose $x \in \mathcal{R}^k$ to minimize the maximum eigenvalue of $A(x) = A_0 + x_1 A_1 + \ldots + x_k A_k$, where $A_i \in \mathcal{R}^{n \times n}$ and $A_i = A_i^T$.
Observe that

$$\lambda_{max}(A(x)) \leq t$$

if and only if

$$\lambda_{max}(A(x) - tI) \leq 0 \quad \Longleftrightarrow \quad \lambda_{min}(tI - A(x)) \geq 0.$$

This holds iff

$$tI - A(x) \succeq 0.$$

So we get the SDP in the *dual* form:

$$\begin{array}{ll} \max & -t \\ \text{s.t.} & tI - A(x) \succeq 0, \end{array}$$

where the variable is $y := (t, x)$.

Logarithmic Barrier Function

Define the **logarithmic barrier function** for the cone $\mathcal{SR}_+^{n\times n}$ of positive definite matrices.
$f : \mathcal{SR}_+^{n\times n} \mapsto \mathcal{R}$

$$f(X) = \begin{cases} -\ln\det X & \text{if } X \succ 0 \\ +\infty & \text{otherwise.} \end{cases}$$

Let us evaluate its derivatives.
Let $X \succ 0, H \in \mathcal{SR}^{n\times n}$. Then

$$\begin{aligned} f(X+\alpha H) &= -\ln\det[X(I+\alpha X^{-1}H)] \\ &= -\ln\det X - \ln(1+\alpha trace(X^{-1}H) + \mathcal{O}(\alpha^2)) \\ &= f(X) - \alpha X^{-1} \bullet H + \mathcal{O}(\alpha^2), \end{aligned}$$

so that $f'(X) = -X^{-1}$ and $Df(X)[H] = -X^{-1} \bullet H$.

Logarithmic Barrier Function (cont'd)

Similarly

$$\begin{aligned} f'(X + \alpha H) &= -[X(I + \alpha X^{-1} H)]^{-1} \\ &= -[I - \alpha X^{-1} H + \mathcal{O}(\alpha^2)] X^{-1} \\ &= f'(X) + \alpha X^{-1} H X^{-1} + \mathcal{O}(\alpha^2), \end{aligned}$$

so that $f''(X)[H] = X^{-1} H X^{-1}$

and $D^2 f(X)[H, G] = X^{-1} H X^{-1} \bullet G$.

Finally,
$f'''(X)[H, G] = -X^{-1} H X^{-1} G X^{-1} - X^{-1} G X^{-1} H X^{-1}$.

Logarithmic Barrier Function (cont'd)

Theorem: $f(X) = -\ln \det X$ is a convex barrier for $\mathcal{SR}_+^{n\times n}$.

Proof: Define $\phi(\alpha) = f(X+\alpha H)$. We know that f is convex if, for every $X \in \mathcal{SR}_+^{n\times n}$ and every $H \in \mathcal{SR}^{n\times n}$, $\phi(\alpha)$ is convex in α.

Consider a set of α such that $X+\alpha H \succ 0$. On this set

$$\phi''(\alpha) = D^2 f(\bar{X})[H,H] = \bar{X}^{-1} H \bar{X}^{-1} \bullet H,$$

where $\bar{X} = X + \alpha H$.

Since $\bar{X} \succ 0$, so is $V = \bar{X}^{-1/2}$ (Property 6), and

$$\begin{aligned}\phi''(\alpha) &= V^2 H V^2 \bullet H = trace(V^2 H V^2 H) \\ &= trace((VHV)(VHV)) = \|VHV\|_F^2 \geq 0.\end{aligned}$$

So ϕ is convex.

When $X \succ 0$ approaches a singular matrix, its determinant approaches zero and $f(X) \to \infty$.

Solving SDPs with IPMs

Replace the **primal SDP**

$$\begin{array}{ll} \min & C \bullet X \\ \text{s.t.} & \mathcal{A}X = b, \\ & X \succeq 0, \end{array}$$

with the **primal barrier SDP**

$$\begin{array}{ll} \min & C \bullet X + \mu f(X) \\ \text{s.t.} & \mathcal{A}X = b, \end{array}$$

(with a barrier parameter $\mu \geq 0$).
Formulate the Lagrangian

$$L(X, y, S) = C \bullet X + \mu f(X) - y^T(\mathcal{A}X - b),$$

with $y \in \mathcal{R}^m$, and write the first order conditions (FOC) for a stationary point of L:

$$C + \mu f'(X) - \mathcal{A}^* y = 0.$$

Solving SDPs with IPMs (cont'd)

Use $f(X) = -\ln\det(X)$ and $f'(X) = -X^{-1}$.
Therefore the FOC become:

$$C - \mu X^{-1} - \mathcal{A}^* y = 0.$$

Denote $S = \mu X^{-1}$, i.e., $XS = \mu I$.
For a positive definite matrix X its inverse is also positive definite.

The FOC now become:

$$\begin{aligned} \mathcal{A}X &= b, \\ \mathcal{A}^* y + S &= C, \\ XS &= \mu I, \end{aligned}$$

with $X \succ 0$ and $S \succ 0$.

Then apply Newton method to the FOC.

The Rank Minimization Problem

$$\begin{array}{ll} \min & \text{rank}(X) \\ \text{s.t.} & \mathcal{A}(X) = b \end{array}$$

$X \in \mathcal{R}^{n \times n}$ is the unknown and the linear map $\mathcal{A} : \mathcal{R}^{n \times n} \to \mathcal{R}^m$ and the vector $b \in \mathcal{R}^m$ are given.

- NP-hard problem
- Applications: matrix completion
  (Netflix problem, triangulation from incomplete data), nonnegative factorization, control and system theory, image compression.

A Rank Minimization Heuristic

$$\begin{array}{ll} \min & \text{rank}(X) \\ \text{s.t.} & \mathcal{A}(X) = b \end{array} \quad \underbrace{\Rightarrow}_{\text{heuristic}} \quad \begin{array}{ll} \min & \|X\|_* \\ \text{s.t.} & \mathcal{A}(X) = b \end{array}$$

where $\|\cdot\|_*$ denotes the *nuclear norm* (the sum of singular values).

- Convex optimization problem
- Special case:
  if X=$diag(x)$, the problem reduces to ℓ_1-*norm minimization*:

$$\begin{array}{ll} \min & \text{card}(x) \\ \text{s.t.} & Ax = b \end{array} \quad \underbrace{\Rightarrow}_{\text{heuristic}} \quad \begin{array}{ll} \min & \|x\|_1 \\ \text{s.t.} & Ax = b \end{array}$$

SDP formulation

Primal-dual convex formulation (heuristic)

$$\begin{array}{ll} \min & \|X\|_* \\ \text{s.t.} & \mathcal{A}(X) = b \end{array} \qquad \begin{array}{ll} \max & b^T y \\ \text{s.t.} & \|\mathcal{A}^*(y)\| \leq 1 \end{array}$$

Primal-dual SDP formulation

$$\begin{array}{ll} \min & \frac{1}{2}(Tr(W_1) + Tr(W_2)) \\ \text{s.t.} & \begin{bmatrix} W_1 & X \\ X^T & W_2 \end{bmatrix} \succeq 0 \\ & \mathcal{A}(X) = b \end{array} \qquad \begin{array}{ll} \max & b^T y \\ \text{s.t.} & \begin{bmatrix} I_m & \mathcal{A}^*(y) \\ \mathcal{A}^*(y)^T & I_n \end{bmatrix} \succeq 0 \end{array}$$

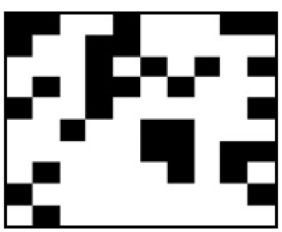
where $y \in \mathcal{R}^m, W_1, W_2 \in \mathcal{R}^{n \times n}$,
$\mathcal{A}^* : \mathcal{R}^m \to \mathcal{R}^{n \times n}$ is the adjoint of $\mathcal{A}$,
$\| \cdot \|$ denotes the operator norm (the maximum singular value).

Netflix Problem (Matrix Completion)

Recommender systems

The Netflix Prize ($1M): In 2006, Netflix held the first Netflix Prize competition to find a better program to predict user preferences and beat its existing Netflix movie recommendation system by at least 10%.

- Given 100 million ratings on a scale of 1 to 5, predict 3 million ratings to highest accuracy
- 17770 total movies x 480189 total users $\Rightarrow$ over 8 billion total ratings

B = 

B_{ij} known for black cells, unknown for white
Row index: *movie*, Column index: *user*
Find low-rank W such that $W \approx B$.

The Matrix Completion Problem

A small number of entries of a matrix $B \in \mathcal{R}^{\hat{m}\times\hat{n}}$ is known:
all entries $B_{i,j}$, with $(i,j) \in \Omega$, where $|\Omega| = m \ll \hat{m}\hat{n}$.

Find an approximation $W \in \mathcal{R}^{\hat{m}\times\hat{n}}$ of B such that:

- W has small rank, and
- W and B agree on Ω.

Matrix Completion Problem

$$\begin{array}{ll} \min & \text{rank}(W) \\ \text{s.t.} & W_{ij} = B_{ij}, \quad \forall (i,j) \in \Omega. \end{array}$$

SDP Relaxation of Matrix Completion

SDP Relaxation

$$\begin{array}{lll} \min & \frac{1}{2}(Tr(W_1) + Tr(W_2)) & \leftrightarrow C \bullet X \\ \text{s.t.} & \begin{bmatrix} W_1 & W \\ W^T & W_2 \end{bmatrix} \succeq 0 & \leftrightarrow X \succeq 0 \\ & W_{ij} = B_{ij} \ (i,j) \in \Omega & \leftrightarrow A_l \bullet X = b_l \end{array}$$

$W \in \mathcal{R}^{\hat{m}\times\hat{n}}$, $W_1 \in \mathcal{R}^{\hat{m}\times\hat{m}}$, $W_2 \in \mathcal{R}^{\hat{n}\times\hat{n}}$ unknowns, B_{ij}, $(i,j) \in \Omega$ given

- $C = I_n$, $X = \begin{bmatrix} W_1 & W \\ W^T & W_2 \end{bmatrix} \in \mathcal{R}^{n\times n}$, with $n = (\hat{m} + \hat{n})$.
- $A_l = \frac{1}{2}\begin{bmatrix} 0 & \Theta^{ij} \\ (\Theta^{ij})^T & 0 \end{bmatrix}$, $l = 1, \ldots, m$, where for each $(i,j) \in \Omega$

  $\Theta^{ij} \in \mathcal{R}^{\hat{m}\times\hat{n}}$: $(\Theta^{ij})_{st} = \begin{cases} 1 & \text{if } (s,t) = (i,j) \\ 0 & \text{else} \end{cases}$ (A_l of rank 2).

Logarithmic Barrier Function

for the cone $\mathcal{SR}_+^{n\times n}$ of positive definite matrices, $f : \mathcal{SR}_+^{n\times n} \mapsto \mathcal{R}$

$$f(X) = \begin{cases} -\ln\det X & \text{if } X \succ 0 \\ +\infty & \text{otherwise.} \end{cases}$$

LP: Replace $x \geq 0$ with $-\mu \sum_{j=1}^{n} \ln x_j$.

SDP: Replace $X \succeq 0$ with $-\mu \sum_{j=1}^{n} \ln \lambda_j = -\mu \ln(\prod_{j=1}^{n} \lambda_j)$.

Nesterov and Nemirovskii,
Interior Point Polynomial Algorithms in Convex Programming: Theory and Applications, SIAM, Philadelphia, 1994.

Lemma The barrier function $f(X)$ is self-concordant on $\mathcal{SR}_+^{n\times n}$.

Interior Point Methods:

- Logarithmic barrier functions for LP, QP, SOCP and SDP
  Self-concordant barriers
  → polynomial complexity (predictable behaviour)
- Unified view of optimization
  → from LP via QP to NLP, SOCP, SDP
- Efficiency
  - good for SOCP
  - problematic for SDP because solving the problem of size n involves linear algebra operations in dimension n^2

    → and this requires n^6 flops!

Use IPMs in your research!