From linear to conic optimization

Erling D. Andersen
http://erling.andersen.name
http://www.mosek.com

September 15, 2009

Outline

- Introduction
- Conic optimization.
- Applications of conic optimization.
- Algorithms for conic optimization.
- Some computational results.
- Literature.
- Conclusions.

Introduction

The most successful OR model:

$$
\begin{array}{lllll}
(PO) & \text{min} & c^T x & & \\
& \text{s.t.} & Ax & = & b, \\
& & x \geq 0. & &
\end{array}
$$

Pros:

- Wide applicability.
- Efficient and robust solution algorithms.
- "Simple" (c, A, b).
- Duality theory.

Cons:

- Only linear.

- $x^2, 1/x, \ln(x), \ldots$.

<u>Nonlinear optimization</u>

$$
\begin{array}{llll}
(NO) & \min & f(x) & \\
 & \text{s.t.} & g(x) & \leq 0.
\end{array}
$$

Pros:

- Very general.

Cons:

- Lack of good algorithms.
- Local versus global optimums.
- Convexity (how to check).
- Black box model.
- How to compute gradients and Hessians.
- How to handle f and g in software.

Summary:

- A linear model is restrictive.
- The nonlinear model is too general
- Is there a good compromise?

Conic optimization

$$
\begin{array}{llll}
(CO) & \text{min} & c^T x & \\
 & \text{s.t.} & Ax & = \; b, \\
 & & x \in \mathcal{K}, &
\end{array}
$$

where $\mathcal{K}$ is a convex cone (closed, pointed and solid).

- $\mathcal{K}$ is convex.

- Cone condition:

$$x \in \mathcal{K} \Rightarrow \alpha x \in \mathcal{K}, \forall \alpha \geq 0.$$

- Pointed:

$$\mathcal{K} \cap -\mathcal{K} = 0.$$

- Solid:

$$int \mathcal{K} \neq \emptyset.$$

Some basic cones:

- Linear:

$$\mathcal{K}_l := \{x \in R : \ x \geq 0\}$$

- Quadratic:

$$\mathcal{K}_q := \left\{ x \in R^n : \ x_1 \geq \sqrt{\sum_{j=2}^{n} x_j^2} \right\}$$

- Semi-definite:

$$\mathcal{K}_s := \left\{ x \in R^{\frac{n(n+1)}{2}} : \ \begin{bmatrix} x_1 & \cdots & x_n \\ \vdots & \ddots & \vdots \\ x_n & \cdots & x_{\frac{n(n+1)}{2}} \end{bmatrix} \succeq 0 \right\}$$

Let

$$X := \begin{bmatrix} x_1 & \cdots & x_n \\ \vdots & \ddots & \vdots \\ x_n & \cdots & x_{\frac{n(n+1)}{2}} \end{bmatrix}$$

then $\succeq$ means **symmetry**

$$X = X^T.$$

And **positive semi-definiteness**:

$$y^T X y \geq 0, \forall y$$

or equivalently

$$\lambda_{\text{min}}(X) \geq 0.$$

Cone composition

Assumption:

$$\mathcal{K} = \mathcal{K}^1 \times \ldots \times \mathcal{K}^k$$

where $\mathcal{K}^i$ is of one of the basic cone types.

Example:

$$\{x_1 \geq 0\} \times \{x_2 \geq \|(x_3, x_4)\|\}$$

Comments:

- Are called symmetric or self-scaled cones. (=self-dual and homogeneous cones.)
- Other cones exists but are not symmetric. Open topic.

Conic vision

- Restricted set of cones (≤ 10).
- Cones are simple and easy to specify.
- Convexity is not an issue.
- A lot of structure.
- Nonlinearity is explicit.
- Gradients and Hessians are not an issue.
- Powerful algorithms exists (theory).

Conic duality

The dual cone:

$$\mathcal{K}^* := \{s : \; x^T s \geq 0, \; \forall x \in \mathcal{K}\}.$$

The dual problem:

$$\begin{array}{llrcl} (CO_D) & \text{max} & b^T y & & \\ & \text{s.t.} & A^T y + s & = & c, \\ & & s \in \mathcal{K}^*. & & \end{array}$$

- Most (but not all) of the duality relations holds.

- $\mathcal{K}_l, \; \mathcal{K}_q, \; \mathcal{K}_s$ are self dual i.e.

$$\mathcal{K} = \mathcal{K}^*.$$

Applications

Conic quadratic optimization

Define the rotated quadratic cone:

$$\mathcal{K}_r := \left\{ x \in R^n : \ 2x_1 x_2 \geq \sum_{j=3}^{n} x_j^2, \ x_1, x_2 \geq 0 \right\}$$

Let

$$\begin{aligned} x_1 &= \tfrac{u+v}{\sqrt{2}}, \\ x_2 &= \tfrac{u-v}{\sqrt{2}}, \end{aligned}$$

then

$$2x_1 x_2 \geq \sum_{j=3}^{n} x_j^2 \Leftrightarrow u \geq \sqrt{v^2 + \sum_{j=3}^{n} x_j^2}$$

so the quadratic and rotated quadratic cones are equivalent. (It is easy to verify $v \geq 0$).

Quadratic optimization

$$\begin{array}{lll} \text{min} & 0.5||Q^0x||^2 + c^Tx & \\ \text{s.t.} & 0.5||Q^ix||^2 + a_{i:}x & \leq \; b_i, \forall i = 1, 2, \ldots. \end{array}$$

Conic quadratic equivalent:

$$\begin{array}{llcll} \text{min} & c^Tx + t_0 & & & \\ \text{s.t.} & t_i + a_{i:}x & = & b_i, & \forall i = 1, 2, \ldots, \\ & Q^ix - y^i & = & 0, & \forall i = 0, 1, \ldots, \\ & z_i & = & 1, & \forall i = 0, 1, \ldots, \\ & ||y^i||^2 & \leq & 2t_iz_i, & \forall i = 0, 1, \ldots. \end{array}$$

Because

$$\frac{1}{2}||Q^ix||^2 \leq t_i, \; \forall i = 0, 1, \ldots.$$

Applications:

- Finance.
- Approximation of more general non-linear problems.
- Linear least squares.

Portfolio optimization. An application

- Select a portfolio of assets i.e. stocks, bonds, etc.

- Such that a large return with a low risk is obtained.

- Assumptions:
  - An initial portfolio is available.
  - A single period.
  - One of the assets is risk free i.e. cash.

Formal definition

Parameters:

- A portfolio can consist of n traded assets numbered $1, 2, \ldots$ held over a period of time

- w_j^0 is the initial holding of asset j where $\sum_j w_j^0 > 0$.

- r_j is the return on asset j assumed to be a random variable. r has a known mean $\bar{r}$ and covariance Σ.

Variables:

- x_j is the amount of asset j traded.
  - If $x_j > 0$, then the amount of asset j is increased (by purchasing).
  - If $x_j < 0$, then the amount of asset j is decreased (by selling).

Tradeoff

Observe

- Return (expected return)

$$E[r^T(w^0 + x)] = \bar{r}^T(w^0 + x)$$

- Risk (variance)

$$V[r^T(w^0 + x)] = (w^0 + x)^T \Sigma (w^0 + x)$$

- High return and a small risk i.e. small variance is desired.

- There is a trade-off between return and risk.

- Expected return and variance can be nontrivial to estimate.

- By definition Σ is positive semi-definite and

$$\begin{aligned}\text{Std. dev.} &= \left\| \Sigma^{\frac{1}{2}}(w^0 + x) \right\| \\ &= \left\| L^T (w^0 + x) \right\|\end{aligned}$$

  where L is **any** matrix such that

$$\Sigma = LL^T$$

  i.e. for instance the Cholesky factor.

- A low rank of Σ is advantageous from a computational point of view.

First model:

$$\begin{array}{lll} \text{min} & (w^0 + x)^T \Sigma (w^0 + x) & \\ \text{s.t.} & \bar{r}^T (w^0 + x) & = t, \\ & e^T x & = 0, \end{array}$$

where $e := (1, \ldots, 1)^T$.

Model:

- Minimizes the variance.

- While selecting a portfolio having an expected target return of t.

- Satisfying the budget or self-financing constraint.

- Can clearly be reformulated as a CQO.

Usage:

- Solved for different values of t.
- Investor choose the portfolio that according to his/her preferences has the best relation between risk and return.
- Nobel prize wining Markowitz model.

Hyperbolic programming

$$\begin{array}{lll} \min & \sum_j \frac{c_j}{x_j} & \\ \text{s.t.} & Ax & = \; b, \\ & x \geq 0, & \end{array}$$

where $c_j > 0$.

Conic quadratic reformulation:

$$
\begin{array}{lllr}
\text{min} & \sum_j c_j t_j & & \\
\text{s.t.} & Ax & = & b, \\
& z_j & = & \sqrt{2}, \\
& z_j^2 & \leq & 2x_j t_j, \\
& x \geq 0. & &
\end{array}
$$

Applications:

- Equilibrium in TCP networks.
- Stratified sampling.
- Stock optimization models.

Robust linear optimization

Non robust LO:

$$\begin{array}{lrcl} \text{min} & c^T x & & \\ \text{s.t.} & a_{i:}x & \leq & b_i, \ \ \forall i. \end{array}$$

Assume:

$$a_{i:}^T \in \mathcal{E}_i := \{z : \ z = \bar{a}_{i:}^T + H^i y, \ \ \|y\| \leq 1\}$$

where

$$H^i \in \mathbf{R}^{n \times l_i}.$$

Robust version (Ben-Tal and Nemirovski):

$$\begin{array}{lrcl} \text{min} & c^T x & & \\ \text{s.t.} & a_{i:}x & \leq & b_i, \ \ a_{i:}^T \in \mathcal{E}_i, \quad \forall i \end{array}$$

or

$$\begin{array}{lrcl} \text{min} & c^T x & & \\ \text{s.t.} & \bar{a}_{i:}x + \left\|(H^i)^T x\right\| & \leq & b_i, \qquad \forall i. \end{array}$$

Is a CQO.

A statistical interpretation

Assumptions:

- $a_{i:}$ are independent Gaussian random vectors.
- $\bar{a}_{i:}$ is the mean and Σ_i is the covariance matrix.

Problem:

$$\begin{array}{lll} \text{min} & c^T x & \\ \text{s.t.} & \text{Prob}(a_{i:}x \leq b_i) & \geq p, \forall i. \end{array}$$

Equivalent problem:

$$\begin{array}{lrcl} \text{min} & & c^T x & \\ \text{s.t.} & \bar{a}_{i:}x + \Phi^{-1}(p)\left\|\Sigma_i^{1/2}x\right\| & \leq & b_i, \forall i. \end{array}$$

where

$$\Phi(z) := \frac{1}{2\pi}\int_{-\infty}^{z} e^{-t^2/2}dt.$$

Is a CQO for $p \geq 0.5$.

Semi-definite optimization - max cut

Let a symmetric graph be given having edge weights

$$w_{ij} \geq 0.$$

Find a cut or equivalent a partition of the nodes into two disjoint sets

$$(S, \bar{S})$$

such that the sum of weights of crossing edges are maximized.

Let

$$x_j = \begin{cases} 1, & \text{node } j \in S, \\ -1, & \text{otherwise.} \end{cases}$$

Observation

total edge weight − weight of crossing edges

$$= \frac{1}{2} \sum_j \sum_i w_{ij} x_i x_j$$

Hence,

$$\text{weight of cut} = \frac{1}{2}\left(\frac{1}{2}\sum_i \sum_j \left(w_{ij} - w_{ij} x_i x_j\right)\right).$$

Max cut problem

$$\begin{array}{lll} \max & \frac{1}{4}\sum_i \sum_j \left(w_{ij}\left(1 - x_i x_j\right)\right) & \\ \text{s.t.} & x_j^2 & = 1. \end{array}$$

Equivalent problem:

$$\begin{array}{lll} \max & \frac{1}{4}\sum_i \sum_j (w_{ij}(1 - X_{ij})) & \\ \text{s.t.} & X - xx^T & = 0, \\ & X_{ii} & = 1. \end{array}$$

Relaxation:

$$\begin{array}{lrcl} \max & \frac{1}{4}\sum_i \sum_j (w_{ij}(1 - X_{ij})) & & \\ \text{s.t.} & X & \succeq & 0, \\ & X_{ii} & = & 1. \end{array}$$

Comments:

- Very good bound. (Optimal value is within 14% of relaxation).

- Provably good heuristic can be devised.

- Major result in optimization.

- SDO can provide bounds for any quadratic optimization problem.

- Bounds are sometimes surprisingly strong.

- Potentially computational expensive. Why?

- A (highly) important technique of the future?

References

- Lectures on Modern Convex Optimization: Analysis, Algorithms, and Engineering Applications. Aharon Ben-Tal and Arkadi Nemirovski.

- http://www.stanford.edu/~boyd/

Optimality conditions

Usual duality holds (almost).

Weak duality:

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

if (x, y, s) is a primal-dual feasible solution.

Strong duality holds in most cases i.e.:

$$c^T x - b^T y = x^T s = 0$$

if and only (x, y, s) is a primal-dual optimal solution.

Potential problems!

- Duality gap can occur.

- Non-attainment:

$$\begin{array}{ll} \text{min} & \frac{1}{x} \\ \text{st} & x \geq 0. \end{array}$$

Optimality conditions:

$$\begin{aligned} Ax &= b, \\ A^T y + s &= c, \\ c^T x - b^T y &= 0, \\ x \in \mathcal{K},\ s \in \mathcal{K}^*. \end{aligned}$$

Primal infeasibility condition:

$$\begin{aligned} b^T y &> 0, \\ A^T y + s &= 0, \\ s \in \mathcal{K}^*. \end{aligned}$$

Dual infeasibility condition:

$$\begin{aligned} c^T x &< 0, \\ Ax &= 0, \\ x \in \mathcal{K}. \end{aligned}$$

Algorithms

Interior-point methods

Barrier approach:

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

Let

$$B(x^k) \to -\infty$$

for x^k approaching the boundary of $\mathcal{K}$.

Solve

$$\begin{array}{lcl} \min & c^T x - \mu B(x) & \\ \text{s.t.} & Ax & = & b, \end{array}$$

for small $\mu > 0$.

Barriers

Linear cone:

$$\ln(x)$$

Quadratic cone:

$$\ln(x_1^2 - \|x_{2:n}\|^2) = \ln(x_1) + \ln\left(x_1 - \frac{\|x_{2:n}\|^2}{x_1}\right)$$

Semi-definite cone:

$$\ln(\det(X))$$

Optimality conditions

Lagrange function:

$$L(x,y) := c^T x - \mu B(x) - y^T(Ax - b).$$

First-order optimality conditions:

$$\begin{array}{rcccl} \nabla_x L(x,y) & = & c - \mu \nabla B(x) - A^T y & = & 0, \\ \nabla_y L(x,y) & = & -Ax + b & = & 0. \end{array}$$

Define

$$s := \mu \nabla B(x)$$

then

$$\begin{array}{rcl} A^T y + s & = & c, \\ Ax & = & b, \\ s & = & \mu \nabla B(x) \end{array}$$

Study

$$s = \mu \nabla B(x)$$

Linear case:

$$s = \mu x^{-1}$$

Quadratic case:

$$s = \mu X^{-1} e_1$$

$X := \mathrm{mat}\,(x)$ i.e.

$$V := \mathrm{mat}\,(v) = \begin{bmatrix} v_1 & v_{2:n}^T \\ v_{2:n} & v_1 I \end{bmatrix}.$$

Semi-definite case:

$$S = \mu X^{-1}.$$

Modified

Linear case:

$$xs = \mu.$$

Quadratic case:

$$Xs = \begin{bmatrix} x^T s \\ x_1 s_{2:n} + s_1 x_{2:n} \end{bmatrix} = \mu e_1.$$

Semi-definite case:

$$XS = \mu.$$

Complementarity conditions

Let $\mu = 0$!

Primal-dual algorithms

Primal-dual optimality:

$$\begin{aligned} A^T y + s &= c, \\ Ax &= b, \\ XS &= \mu. \end{aligned}$$

WARNING: Sloppy notation but you get the idea!

One Newton step

$$\begin{aligned} A^T d_y + d_s &= c - A^T y^0 - s^0, \\ A d_x &= b - A x^0, \\ X d_s + S d_x &= -XS + \mu. \end{aligned}$$

for suitable chosen μ and starting point.

Comments:

- Newton step is not well-defined always.

- Requires (Nesterov-Todd) scaling. Exists only for symmetric cones.

- Leads to a powerful primal-dual algorithm.

- Polynomial complexity (solution may not be rational).

- Hard to generalize to nonsymmetric cones.

Numerical results

- MOSEK v5.0.0.121.
- Linux server.

Problems

			Presolved	
Name	Constraints	Variables	Constraints	Variables
1	6874	6797	2818	4148
2	5868	9612	5867	9611
3	902	2710	900	2707
4	6086	14711	6086	14711
5	5	6	1	2
6	2276	2691	766	1274
7	460	18295	459	18294
8	406	15897	405	15896
9	24	32	23	31
10	5868	11533	5867	11532
11	6223	17766	301	11844
12	698	1049	698	1049
13	6224	35532	302	29610
14	402	11886	402	11886
15	50	34	13	30
16	298	348	169	215
17	14745	84709	14744	84708
18	19	23	5	9
19	16200	65885	16199	49413
20	1793	1942	1270	1422
21	97680	162001	46560	78482
22	64800	261365	64799	196023
23	123	2383	121	2379
24	123	2641	122	2638
25	2526	4977	1717	4168
26	8337	18238	5744	15645
27	4843	9744	3289	8190
28	18086	37887	12430	32231
29	915	3176	120	2379
30	123	4195	120	4190
31	13611	24445	13587	22609

Optimized problem:

Name	Con- straints	Quad. cones	Varia- bles	Cone var.
1	4148	1365	6843	4025
2	5868	1923	9612	9611
3	900	4	2709	1808
4	6086	2943	14711	14711
5	1	1	4	3
6	766	270	1460	1164
7	459	9	18295	1118
8	405	5254	15897	15897
9	23	1	32	24
10	5867	3844	11532	11532
11	301	5922	17766	17766
12	698	1	1049	350
13	302	11844	35532	35532
14	402	3962	11886	11886
15	14	5	33	25
16	210	57	344	238
17	14744	28236	84708	84708
18	5	3	13	11
19	16199	16471	65884	65884
20	1270	487	1422	1422
21	46560	32400	110882	97199
22	64799	65341	261364	261364
23	121	793	2379	2379
24	122	839	2639	2637
25	2525	1	4976	2475
26	8336	1	18237	8236
27	4842	1	9743	4742
28	18085	1	37886	17885
29	913	793	3172	2379
30	120	839	4191	4191
31	17208	3614	30741	13609

Accuracy and efficiency:

Name	Primal obj.	Sig. fig.	Iter.	Time
1	-5.3902456253e+03	10	26	0.4
2	-2.3284506027e-01	9	11	1.9
3	-1.8019712354e-01	7	19	11.7
4	-1.0334780253e-02	9	19	11.6
5	-1.4142135621e-01	10	8	0.0
6	9.6279652799e+06	7	31	0.1
7	-4.1497838732e-05	9	22	80.2
8	-5.5745820525e-05	9	34	73.9
9	-1.0983618747e+00	9	9	0.0
10	-1.5563030613e-02	10	57	7.3
11	-3.5226365277e+00	13	17	29.7
12	1.8428687768e-04	9	11	0.0
13	-3.1812503728e+00	12	26	97.4
14	-1.0068176825e-02	10	24	46.2
15	1.3482698063e+02	9	14	0.0
16	3.3972095441e+00	9	9	0.0
17	1.9171403326e+04	8	44	7.1
18	4.0973429125e-01	10	7	0.0
19	-6.5943460525e+00	10	21	6.2
20	-2.2755393309e+01	9	18	0.1
21	-9.2771622627e-01	9	17	29.6
22	-6.6394902197e+00	10	21	39.3
23	-5.0703094413e-02	10	19	1.0
24	-1.0256950693e-01	9	11	0.6
25	7.8520415550e+00	7	23	0.2
26	2.7330945916e+01	6	73	2.8
27	6.7165032259e+01	9	29	0.6
28	5.1811966331e+01	8	37	4.2
29	-1.3012270287e+01	9	17	1.0
30	-1.6289715302e+00	7	16	2.0
31	-3.2774412605e-05	9	25	2.9

Software you can try out:

- MOSEK (see http://www.mosek.com/)
- SeDuMi (see http://sedumi.mcmaster.ca/).
- Benchmarks and more links:

  `http://plato.asu.edu/bench.html`

Conclusions

- Conic optimization is an exciting extension of LOs.

- Capable of solving large problems.

- Conic quadratic optimization is already useful (in business).

- Semi-definite optimization has great potential.