Subject:
Date

بهینه سازی مقاوم

دکتر سرافراز

جلسه اول   دوشنبه   ۱۴۰۲/۱۱/۲۳   2024-02-12

شنبه آمده اند ولی درب کلاس ها بسته بوده است. موضوع درس بهینه سازی مقاوم است. ابتدا مسئله بهینه سازی استاندارد را یادآوری کنیم:

$$\min_x f_0(x)$$
s.t.
$$f_i(x) \leq 0 \quad i=1,\dots,m$$
$$g_i(x) = 0 \quad i=1,\dots,n$$

جواب x^*

می توان مسئله را به فرم دیگری نیز نوشت:

$$\min_{x,t} t$$
$$\text{s.t.} \quad f_0(x) - t \leq 0$$

جواب (x^*, t^*)

به علاوه f_i ها، g_i ها.

در این فرم شکل قید می ماند. فایده فرم دوم

منابع خطا و نایقینی:

(۱) ضرایب مسئله، مثل ضرایب f_0، g_i ها، f_i ها، یا پارامترهای سیستم هستند (مثلاً ضریب سختی فنر) یا ساختار سیستم (مثل پارامترهای قوانین طبیعی) وجود دارند. دومی عموماً دقیق است ولی اولی می تواند نادقیق باشد.

(۲) Estimation of Measurement: سنسورها یا تخمین گرهایی که در سیستم داریم، خطا دارند.

(۳) Implementation: حتی اگر x^* را دقیق محاسبه کرده باشیم، موقع اعمال کردن

Subject:
Date

احتمالاً خطا وجود دارد و $x^*(1+\varepsilon)$ اعمال می‌شود. چرا؟ خطای actuator و

خطای quantization ناشی از گسسته سازی برای شرایط دیجیتال.

در این شرایط که عدم قطعیت وجود دارد Optimization with Uncertainty

مسئله Robust Optimization به وجود می‌آید.

مثال: مسئله Pilot 4 from Netlib

$$\min c^T x \qquad x \in \mathbb{R}^{1000}$$
$$\text{s.t. } Ax \geq b \quad \text{410 constraints}$$
$$\equiv a_i^T x \geq b_i$$

بعد از حل x^* به دست آمده است.

شرط 372، active است یعنی $a_i^T x - b = 0$ است. ولی اگر a_iها فقط

0.1% تغییر کنند، به جای صفر می‌شود $a_i^T x - b \simeq -10^5$ که اصلاً feasible نیست.

مثال بالا تا حدی نادرست بوده چون a_iها را دستی به گونه‌ای تغییر دادیم که

اوضاع خیلی بد شود. ولی اگر $\tilde{a}_i = (1+\xi_i)\, a_i$

$$\xi_i \sim \text{Uniform}\,[-0.001,\ 0.001] \text{ and } V = \frac{b - \tilde{a}_i^T x}{b_i}$$

$$\text{Prob}\{V > 0\} = 0.5,\quad \text{Prob}\{V > 150\%\} = 0.18,$$

$$\text{mean}\{V\} = 125\%$$

نتیجه اخلاقی: تغییرات کوچک می‌تواند موجب تغییرات زیادی شود. پس robust مهم است.

Subject:
Date

برای مسائل دینامیک، قیودی را می‌نویسیم که الزاماً به صورت LMI نوشت.

LMIs in Control Theory   LMI: Linear Matrix Inequality

حتی در Robust Control هم ابتدا مسئله را به LMI تبدیل می‌کنند و بعد آن را

حل می‌کنند. می‌رویم سراغ Robust Optimization و از ساده‌ترین

مسئله که برنامه‌ریزی خطی باشد، شروع می‌کنیم: Linear Programming (LP)

$$\min \; c^T x$$
$$\text{s.t.} \quad Ax \leq b$$

در مسئله استاندارد (A, b, c) پارامترهای معلوم هستند

در حالت Uncertain Linear Optimization، پارامترها دیگر

ثابت نیستند و $(A, b, c) \in U$ هستند. مثلاً $b_i^\circ(1-\varepsilon) \leq b_i \leq b_i^\circ(1+\varepsilon)$

$\varepsilon > 0$ معلوم

می‌توانیم مسئله را بازنویسی کنیم:

$$\min \; c^T x$$
$$\text{s.t.}$$
$$Ax \leq b^\circ(1-\varepsilon)$$

در بازنویسی روبرو، worst-case را

در نظر گرفته‌ام.

عموماً در مسئله robust، می‌خواهیم قید همواره برقرار باشد که معادل این است که

$$\min \; c^T x$$
$$\text{s.t.} \; Ax \leq b \quad \forall (c, A, b) \in U$$

حتی در worst-case قید برقرار باشد.

PAPCO
③ prime

Subject:
Date

برای سادگی نگارش، همیشه هدف را t میگذاریم و قید $f_0(x) - t \leq 0$ هم میگذاریم. در این صورت، تابع هدف نایقینی ندارد و مسئله تبدیل به robust feasible میشود. این جلسه، برای سادگی فرض میکنیم c معلوم است.

پارامترها، ممکن است واقعاً ثابت باشند، یا واقعاً در بازه‌ای تغییر کنند. مثلاً در Auto Pilot هواپیما، وزن مسافران را نمی‌دانیم ولی ثابت است و وزن سوخت، در یک بازه است. (خیلی وقت‌ها، تخمین دقیق آن سخت است.)

$$\min c^T x \quad s.t. \quad Ax \leq b \longrightarrow \min \{ c^T x \mid Ax \leq b, \ \forall (A,b) \in U \}$$

Robust Counterpart $\min \{ c^T x \mid \sup_{(A,b) \in U} \{Ax - b\} \leq 0 \}$

معادل Robust Optimization (RO) همتای

اگر حالت worst-case را در نظر بگیریم، تضمین 100% میدهیم ولی ممکن است خیلی از بهینه فاصله داشته باشد. میتوانیم کمتر سختگیری کنیم:

Stochastic Optimization (SO); chance-constraint

توزیع A, b نیاز است:

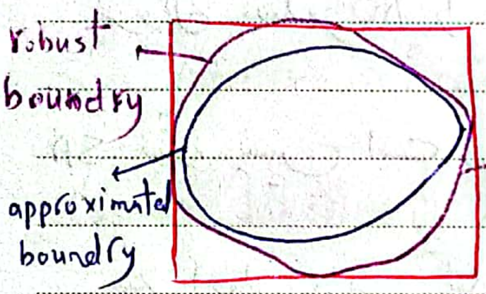
$$\min_{(A,b) \sim P} \{ c^T x : \text{Prob}\{ Ax - b \geq 0 \} \leq \varepsilon \} \qquad \varepsilon << 1$$

شرط معادل $\text{Prob}\{ Ax - b \leq 0 \} \geq 1 - \varepsilon$

pure $2^2 = 4$

حالت SO ، کمتر محافظه کار است ولی لازمه اش این است که پارامترهای A,b واقعاً ماهیت تصادفی داشته باشند و اطلاعات کافی داشته باشیم که بتوانیم Prob حساب کنیم. شرط آخر هم این است که ریسک اش را بپذیریم که ممکن است با احتمال ε، از آن عدول کنیم. بعضی جاها ممکن نیست.

* یک چالش این است که خیلی وقت ها، مسئله از نظر محاسباتی Tractable نیست و حل چند جمله ای ندارد و باز مجبوریم تقریب بزنیم که محافظه کارتر هم می‌شود.



مرز robust در شکل روبرو، از اشتراک قیدهای چرخنده به وجود آمده است ولی

ولی در واقع پیدا کردن این مرز NP hard است

به دست آوردن اش خیلی سخت است، به همین دلیل approximate می‌کنیم.

* چالش دیگر در SO این است که خود پارامترهای توزیع ها را ممکن است دقیق نداشته باشیم. مثلاً بدانیم توزیع اش نرمال است ولی دقیق میانگین را ندانیم و بدانیم در یک بازه است.

$$\min \left\{ c^T x : \operatorname{Prob}_{(A,b)\sim P} \{ Ax - b \leq 0 \} \geq 1-\varepsilon ,\ \forall P \in \mathcal{P} \right\}$$

Distributionally Robust Optimization (DRO) برگرفته از این مسئله.

Subject:
Date

DRO گویا بینابین RO و SO است. در RO: no-data about distribution

در SO: exact data about dist. در DRO: some data about distribution

در درس از LP شروع می‌کنیم. ابتدا RO، بعد SO. اگر وقت شد DRO.

بحث Robust کلاً بحث جدیدی است (حدود 20 سال اخیر). DRO خیلی جدید

است (حدود 3 سال). به خصوص در مباحث ML و AI جذاب و مهم است.

روال مسائل RO را با تحلیل حساسیت Sensitivity Analysis هم ترکیب کرد ولی

SA بیشتر پسین است و بعد از حل مسئله انجام می‌شود.

منبع اصلی درس کتاب آقای نمیروفسکی Robust Optimization

این که ریاضیات را عمیق شویم (برای پژوهش بهتر است) یا ابزارهای بیشتری

ببینیم را در طول ترم بسته به feedback، flexible هستند.

چند تعریف: Convex Set: $\forall x_1, x_2 \in C: \theta_1 x_1 + \theta_2 x_2 \in C$ s.t. $\theta_1 + \theta_2 = 1$, $\theta_1, \theta_2 \geq 0$

Hyperplane: $\{x \mid a^T x = b\}$ $a \neq 0 \Rightarrow$ Convex

ابرصفحه

Half-space: $\{x \mid a^T x \geq b\} \Rightarrow$ Convex

نیم فضا

PAPCO

dual $2 \times 3 = 6$

Subject:
Date

چند وجهی: تقاطع چند ابرصفحه و نیم‌فضا Polyhedra: $\{x \mid Ax \leq b, Cx = d\}$

مخروط Cone: $\{x \in k, \theta x \in k, \theta \geq 0\}$

Conic Combination: $x_1, x_2 : \theta_1 x_1 + \theta_2 x_2 : \theta_1, \theta_2 \geq 0$

Norm-ball: $\|x - x_c\| \leq r$ $\Rightarrow$ convex

Norm-cone: $\{(x, t) \mid \|x\| \leq t\}$ $\Rightarrow$ convex

S^n: symmetric $n\times n$ matrix. $S^n_{\geq 0}$: positive semi-definit $n\times n$ matrix. $= \{X \in S^n, X \geq 0\}$

S^n_+: positive definite $n\times n$ matrix

$\theta_1 + \theta_2 = 1, \theta_1, \theta_2 \geq 0$

Convex function: $f(\theta_1 x_1 + \theta_2 x_2) \leq \theta_1 f(x_1) + \theta_2 f(x_2)$

درس، پروژه پایانی دارد. میانترم و پایان‌ترم هم دارد. شاید سمینار: (ارائه مقاله روز) هم داشته باشیم. تمرین خیلی نخواهد داشت درس. نگران نمره نباشید.

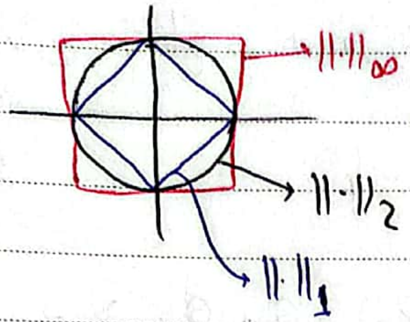
2024-02-17 ۱۴۰۲/۱۱/۲۸ شنبه جلسه دوم

چند جلسه به صورت مختصر، مبحث Convex Optimization را مرور خواهیم کرد.

کره به مرکز x_c و شعاع r. البته نرم های مختلف می‌تواند $\|x - x_c\| \leq r$ داشته باشد که شکل‌های مختلف می‌شود.

PAPCO
 prime

Subject:
Date



برای $p \geq 1$، محدب است

ممکن است رخ دهد.

مقادیر مختلف p، در تعریف norm-cone هم

Convex Cone : $S^n_{\geq 0}$ $x_1, x_2 \in k$

$\Theta_1 k_1 + \Theta_2 k_2 \in k \quad \forall \Theta_1, \Theta_2 \geq 0$

Affine functions: $f: \mathbb{R}^n \to \mathbb{R}^m$, $f(x) = Ax + b$, $A \in \mathbb{R}^{m\times n}$
$b \in \mathbb{R}^m$

شامل حالت های scaling، rotation، addition، ... میشود.

if $X \subseteq \mathbb{R}^n$ is convex, then $f(X) = \{f(x) \mid x \in X\}$ is convex

Proof: $x, y \in X$: $\alpha x + (1-\alpha) y \in X$ $\quad \alpha \in [0,1]$ چون X محدب است داریم،

Assume $f(x), f(y) \in f(X)$, then $\alpha f(x) + (1-\alpha) f(y) \in f(X)$

چون خطی است $f(x) =$

$= f(\underbrace{\alpha x + (1-\alpha) y}_{\in X}) \in f(X)$

برعکس این هم صادق است،

Y is a convex set $\subseteq \mathbb{R}^m$ then $f^{-1}(Y) \subseteq \mathbb{R}^n$ is a convex set.

All functions that satisfy $\begin{cases} f(x+y) = f(x) + f(y) \\ f(\alpha x) = \alpha f(x) \end{cases} \Rightarrow f(x) = Ax$

که به خاطر شرط بالا، دو طرفه Convex بودن را نگه میدارد که خیلی خوب است.

PAPCO

pure $2^3 = 8$

Subject:
Date

$\{x \mid f(x) \leqslant 0\}$, s.t. f is an affine function is convex.

Proof: $f(x_1) \leqslant 0$, $f(x_2) \leqslant 0 \Rightarrow Ax_1 + b \leqslant 0$, $Ax_2 + b \leqslant 0$

$$f(\theta x_1 + (1-\theta)x_2) \overset{?}{\leqslant} 0 \xrightarrow[\text{سمت چپ را}]{\text{ساده می‌کنیم}} A(\theta x_1 + (1-\theta)x_2) + b$$

$$= \underbrace{\theta}_{+} \underbrace{(Ax_1 + b)}_{-} + \underbrace{(1-\theta)}_{+} \underbrace{(Ax_2 + b)}_{-} \leqslant 0$$

اثبات شد.

LMI خیلی کاربرد دارد. Linear Matrix Inequalities (LMIs)

$$\{x \mid x_1 A_1 + x_2 A_2 + \dots + x_n A_n \leqslant B\} \quad A_i, B \in S^P$$

به خاطر شرط بالا، مجموعه محدب می‌شود. کافی است به فرم زیر نوشته شود که شکل بالا باشد،

پس محدب است:

$$\begin{bmatrix} A_1 & A_2 & \cdots & A_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} - B \leqslant 0$$

چرا مجموعه‌های محدب را دوست داریم: ① اشتراک مجموعه‌های محدب، محدب است.

② حل مسئله بهینه‌سازی محدب راحت‌تر است.

چند مثال از LMI ها:

$f: S^n \to S^n \quad A^T X + XA + Q \leqslant 0$

$$f(x) = \begin{bmatrix} X & A^T X \\ XA & X \end{bmatrix} \leqslant 0$$

در کنترل در بررسی پایداری سیستم

$\dot{x} = Ax$ و نوشتن لیاپانوف کاربرد دارد.

Subject:
Date

$\dot{x} = Ax \longrightarrow \exists P = P^T \geq 0$ s.t. $PA + A^T P < 0$ which is equivalent

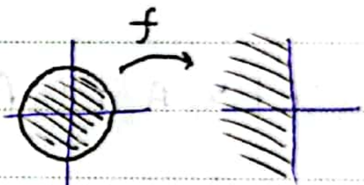
to. $PA + A^T P + Q \leq 0$ where $Q > 0$

Linear Fractional Transformation (LFT)

$$f(x) = \frac{Ax + b}{\underbrace{c^T x + d}_{\text{اسکالر است}}} \;,\quad f : \mathbb{R}^n \longrightarrow \mathbb{R}^m \qquad \mathrm{dom}(f) = \{x \mid c^T x + d > 0\}$$

If X convex, then $f(X)$ convex. If Y convex, $f^{-1}(Y)$ convex.

Example: $f(z) = \frac{2}{T} \frac{z-1}{z+1} \quad z \in \mathbb{C}$



توابع حجم را عوض میکنند، ولی بعد را عوض نمیکنند. با طریق مشابه میتوان نشان داد

که بعد $\mathbb{R}$ با بعد مجموعه $[0,1]$ برابر است.

Convex functions: ① dom(f) convex

$\alpha \in [0,1]$

② $f(\alpha x + (1-\alpha) y) \leq \alpha f(x) + (1-\alpha) f(y)$

f concave $= -f$ convex Strictly convex $(\leq) \to (<)$

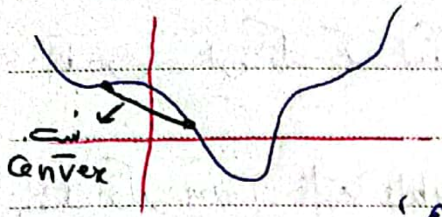
Quasi-convex: ① dom(f) Convex

② $\{x \mid f(x) \leq a\} \; \forall a$: convex

PAPCO

dual $2 \times 5 = 10$

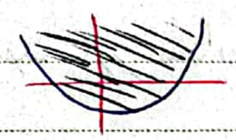
Subject:
Date



تابع روبرو Convex نیست ولی quasi-convex است.

در نمودار تابع Convex، هر دو نقطه ای را به هم وصل کنیم،

$\{(x,y) \mid y \geq f(x)\}$ is Convex iff f is convex بالاتر از خود تابع را اقتصر



از نظر هندسی، مجموعه‌ی فضای بالای تابع f م‌شود.

f Quasi-concave : $-f$ Quasi-convex.

تا اینجا تعاریف و مقدمات گفتیم. حال برویم سراغ یک مسئله بهینه سازی.

Optimization Problem $\min \; f_0(x)$ فرم کلی هر مسائل بهینه سازی.

s.t.

$x \in \mathbb{R}^n$ $f_i(x) \leq 0 \quad i=1,\dots,m$

$f_i : \mathbb{R}^n \to \mathbb{R} \quad i=0,1,\dots,m$ $h_i(x) = 0 \quad i=1,\dots,p$

$h_i : \mathbb{R}^n \to \mathbb{R}$

اگر بخواهیم محدب باشد مسئله (که هدف ما همین است)

Convex opt. problem	Quasi-convex problem
$f_0, f_1, \dots, f_m$ Convex	f_0 ; quasi-convex
$h_i(x) = a_i^T x - b_i$ (affine)	$f_1, \dots, f_m$ Convex
	$h_i = a_i^T x - b_i$

Subject:
Date

با توجه به این نکته که اشتراک مجموعه‌های محدب، محدب است، برای هر دو مسئله ذکر شده، feasible set محدب است. عموماً اصلاً پیدا کردن مجموعه نقاط feasible کار مهمی است. اگر انجام شد، حل مسئله بهینه‌سازی چندان مهم نیست.

در بهینه‌سازی مقاوم، هدف تبدیل کردن مسئله به یک مسئله بهینه‌سازی معمولی است.

SDP = Semi-definite program

$$\min \; c^T x$$

s.t.

$$x_1 F_1 + x_2 F_2 + \cdots + x_n F_n + G < 0 \,, \quad F_i, G \in S^k$$

اگر چند شرط LMI داشته باشیم، می‌توانیم ترکیب شان کنیم در یک شرط.

مثلاً اگر شرط $x_1 H_1 + x_2 H_2 + \cdots + x_n H_n + G' \leq 0$ را هم داشتیم، ترکیب این دو شد:

$$x_1 \begin{bmatrix} F_1 & 0 \\ 0 & H_1 \end{bmatrix} + x_2 \begin{bmatrix} F_2 & 0 \\ 0 & H_2 \end{bmatrix} + \cdots + x_n \begin{bmatrix} F_n & 0 \\ 0 & H_n \end{bmatrix} + \begin{bmatrix} G & 0 \\ 0 & G' \end{bmatrix} \leq 0$$

به همین دلیل، هر جا نوشتیم همان یک شرط را بگذاریم کامل است. در فصل اول درس حالت خطی را بررسی می‌کنیم و در فصل چهار بعد سراغ SDP ها می‌رویم که در کنترل هم خیلی کاربرد دارد.

PAPCO

dual $2^2 \times 3 = 12$

Subject:
Date

★ Scher Complement Lemma

$Q(x) = Q^T(x)$, $S(x)$, $R(x) = R^T(x)$ $\longrightarrow$ Affine functions of x.

$$\begin{bmatrix} Q(x) & S(x) \\ S^T(x) & R(x) \end{bmatrix} > 0 \quad \longleftrightarrow \quad \begin{cases} R(x) > 0 \\ Q(x) - S^T(x) R^{-1}(x) S(x) > 0 \end{cases}$$

ضمنی $Q(x)>0$ هم باید باشد.

اگر بخواهیم مسئله را به فرم LMI تبدیل کنیم، از راست به چپ لم شور استفاده کنیم.

Example1: $P(x) > 0$, $\mathrm{trace}\left(S^T(x) P^{-1}(x) S(x)\right) < 1$ چگونه به

LMI تبدیل کنیم؟ اگر Z را به گونه‌ای انتخاب کنیم که: $\mathrm{trace}(Z) < 1$

می‌شود:

$$\begin{bmatrix} Z & S \\ S^T & P \end{bmatrix} > 0$$

Z متغیر بهینه‌سازی است که خودش را به 1 می‌چسباند.

بهتر است قید نامساوی باشد تا تساوی. قید تساوی بد است.

Example 2: $\|A\|_2 \leq t \quad (t>0) \longleftrightarrow A^T A \leq t^2 I$

$$\longleftrightarrow A^T \frac{1}{t} A \leq t I \longleftrightarrow t I - A^T \frac{1}{t} A \geq 0$$

$$\Leftrightarrow \begin{bmatrix} tI & A^T \\ A & tI \end{bmatrix} \geq 0$$

به عبارتی norm-cone را هم به LMI تبدیل کردیم.

به جای A اگر $A(x)$ بود هم باز همین طور می‌شد نوشت.

Subject:
Date

$\|A(x)\|_2 = x_1 A_1 + \cdots + x_n A_n$ اگر بتوانیم مسئله را به فرم

$\rho(A^T(x) A(x)) = \lambda_{max}(A^T(x) A(x))$ LMI تبدیل کنیم، برنامه‌های

خوبی برای حل آن وجود دارد

که خیلی خوب کارا پیدا حل می‌کنند.

$$\min_{t,x} \; t$$

$$\text{s.t.} \begin{bmatrix} tI & A(x) \\ A^T(x) & tI \end{bmatrix} \succeq 0$$

* S-procedure (S-lemma)

$F_i(x) = x^T A_i x + 2 b_i^T x + c_i, \quad A_i \in S^n, \; i = 0, \dots, m$

آیا $F_i(x) \leq 0$, for all $i = 1, \dots, m$ برای همه Xهای feasible، $F_0(x) \leq 0$

هست یا خیر؟ یک شرط کافی می‌دهد:

Sufficient Condition:

$F_0(x) < 0$, for all x: $F_i(x) \leq 0 \quad i = 1, \dots, m$

if $\quad F_0(x) - \sum \tau_i F_i(x) \leq 0, \quad \tau_i \geq 0, \forall x$

یعنی می‌گوید به جای اینکه m تا نامعادله حل کنیم، با انتخاب τ مناسب،

فقط یک نامعادله حل کنیم، شرط کافی می‌دهد.

if $m = 1 \longrightarrow$ S-procedure neccessary & sufficient.

PAPCO

dual 2×7 =

Subject:
Date

$$\forall x: x^T A_i x + 2 b_i^T x + c_i \le 0 \longleftrightarrow \begin{bmatrix} A_i & b_i \\ b_i^T & c_i \end{bmatrix} \preceq 0$$

پس شرط S-procedure می‌شود،

و دیگر کلاً x ندارد که

$$\begin{bmatrix} A_0 & b_0 \\ b_0^T & c_0 \end{bmatrix} - \sum \tau_i \begin{bmatrix} A_i & b_i \\ b_i^T & c_i \end{bmatrix} \preceq 0$$

کارمان خیلی راحت‌تر می‌شود.

Example: پس p تا بیضی که داریم، می‌خواهیم بیضی (کوچکترین) پیدا کنیم که اشتراک آن‌ها را در بر بگیرد.

$$(x-x_i)^T A_i (x-x_i) \le 1 \leftrightarrow \underbrace{x^T A_i x - 2x_i A_i^T x_i + (x_i^T x - 1)}_{\text{Quadratic فرم}} \le 0$$

اسم بیضی هدف می‌گذاریم: $(x-x_0)^T A_0 (x-x_0) \le 1$

کوچکترین بیضی را اگر از نظر حجم بگیریم، می‌شود:

$$\min \det(A_0)$$

$$\text{s.t.} \begin{bmatrix} A_0 & x_0^T A_0 \\ A_0 x_0 & x_0^T x_0 - 1 \end{bmatrix} - \sum \tau_i \begin{bmatrix} A_i & x_i^T A_i \\ A_i x_i & x_i^T x_i - 1 \end{bmatrix} \preceq 0$$

الان هم x_0، هم A_0 متغیر تصمیم هستند که دارند در هم ضرب می‌شوند. $\tau_i \ge 0$

سؤال: اگر اجتماع بود، کوچکترین بیضی در برگیرنده‌اش چگونه می‌شود؟

دنبال این بیضی هستیم.

جلسه بعد، KKT و بهینه‌سازی را می‌گوییم و از هفته بعدش وارد مقادم خواهیم شد.

Subject:
Date

2024-02-19 دوشنبه ۳۰، ۱۱، ۱۴۰۲ جلسه سوم

جلسه قبل، دو ابزار Shor's comp, S-proc را معرفی کردیم به عنوان ابزارهای مهم در RO.

این جلسه و جلسه بعد، در مورد حل مسئله بهینه سازی محدب صحبت خواهیم کرد.

Cone: $C \to x_1, x_2 \in C \Rightarrow \theta_1 x_1 + \theta_2 x_2 \in C \quad \theta_1, \theta_2 \geq 0$

Dual cone: $K \subseteq \mathbb{R}^n \longrightarrow K^* = \{x \mid x^T y \geq 0, \ \forall y \in K\}$

یک cone

convex است

If K is a cone, then $K^{**} = K$

Notation: K cone در ادامه چند notation می‌گوییم که بعداً استفاده خواهیم کرد

$x \succeq_K 0 \longrightarrow x \in K \qquad x \preceq_K 0 \longrightarrow -x \in K$

$x \succeq_K y \longrightarrow x - y \in K \qquad x \succ_K 0 \longrightarrow x \in \text{interior}(K)$

notation $\succeq_K$، تعمیم از $\geq$ معمولی است ($x \in \mathbb{R}^n$, $x \geq 0$ یعنی تک تک درایه‌ها مثبت)

برای $\succeq_K$ یعنی داخل cone قرار می‌گیرد (به جای نیم صفحه $x \geq 0$ یا نیم فضا)

اثبات مخروط بودن: $K_p = \{(x,t) \mid \|x\|_p \leq t\}$

$\|x_1\|_p \leq t_1 \qquad \|x_1 + x_2\|_p \overset{?}{\leq} t_1 + t_2$

$\|x_2\|_p \leq t_2 \qquad \|x_1 + x_2\|_p \leq \|x_1\|_p + \|x_2\|_p \leq t_1 + t_2$ ✓

نامساوی مثلث

PAPCO

$$k_p^* = \{(x,t) \mid \|x\|_q \leqslant t\} = k_q \quad s.t. \quad \frac{1}{p} + \frac{1}{q} = 1$$

یا معادلاً $\max|x_i|$

if $p=1 \rightarrow k_p^* = \{(x,t) \mid |x_i| < t\}$

$$\{(x,t) \mid \sum |x_i| < t\}$$

$k_+ = \{x \in \mathbb{R}^n \mid x \geqslant 0\} \longrightarrow k_+^*$ چه می‌شود؟

می‌شود خودش $k_+^* = k_+$ اثبات:

طبق تعریف

$k_+^* = \{x \mid x^T y \geqslant 0, \forall y \geqslant 0\}$ یک بار $y = (1, 0, \dots, 0) \Rightarrow x_1 \geqslant 0$

〃 $y = (0, 1, 0, \dots, 0) \Rightarrow x_2 \geqslant 0$

$k = \{X \in S^n \mid X \geqslant 0\}$ ماتریس های PSD با تعریف ضرب

$\Rightarrow k^* = k$ داخلی $\langle X, Y \rangle = \mathrm{trace}(X^T Y)$

اثبات: $k^* = \{X \in S^n \mid \mathrm{trace}(X^T Y) \geqslant 0, \forall y \in k\}$

زحمت دارد. باید از خاصیت trace که می‌توان ترتیب را چرخاند استفاده کرد

و تعریف $x^T A x \geqslant 0$ برای $A \geqslant 0$ زحمت کشید. بروم سراغ مسئله بهینه سازی.

$$\begin{aligned} &\min \ f_0(x) \\ &\text{subject to} \quad f_i(x) \leqslant 0 \quad i = 1, \dots, m \\ &\qquad\qquad\quad h_i(x) = 0 \quad i = 1, \dots, p \end{aligned}$$

$x \in \mathbb{R}^n$ decision variable

$x \in X$ domain

جواب مسئله بهینه سازی $\rightarrow p^*$

Subject:
Date

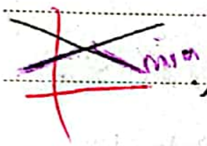
$L: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \longrightarrow \mathbb{R}$ تابع لاگرانژ

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

Lagrange Dual function λ_i ، v_i ضرایب لاگرانژ

$$g(\lambda,v) = \min_{x \in X} \left\{ f_0(x) + \sum \lambda_i f_i(x) + \sum v_i h_i(x) \right\}$$

گویا به ازای xهای مختلف خط یا ابر صفحه (بر حسب λ و v) رسم کرده‌ایم و بعد بین



آن‌ها min گرفته‌ایم که تابعی مقعر Concave به دست می‌دهد.

می‌توان نشان داد $g(\lambda,v) \leq p^*$ است به ازای $\forall \lambda \geq 0, \forall v$

در نتیجه، جواب مسئله دوگان: (Dual Problem) $\max_{\lambda,v} g(\lambda,v)$

s.t. $\lambda \geq 0$

d^* نام بگذاریم، $d^* \leq p^*$ برقرار باشد Weak Duality

مسئله‌ی دوگان، یک مسئله بهینه‌سازی محدب است، مستقل از مسئله اصلی.

Dual problem is a convex problem. با حل این مسئله، می‌توانیم

تخمین دست پایین از جواب مسئله اصلی (p^*) به دست آوریم.

با یک مثال، دوگان را ببینیم. مسئله‌ی Linear Programming را بررسی کنیم.

PAPCO

Subject:
Date

$\min\ c^T x$ (primal) $\quad \mathcal{L}(x,\lambda,v) = c^T x + \lambda^T(Dx - e) + v^T(Ax - b)$

s.t. $Ax = b$

$Dx \leqslant e$

$$\frac{\partial L}{\partial x} = C + D^T\lambda + A^T v = 0$$

در نقطه بهینه باید این شرط برقرار باشد، پس ضرایب x می‌رود.

$g(\lambda, v) = -\lambda^T e - v^T b$

پس اصلاح هم Linear Program شد.

(dual) $\max\limits_{\lambda,v}\ -\lambda^T e - v^T b = -\begin{bmatrix} e^T & v^T \end{bmatrix}\begin{bmatrix}\lambda \\ v\end{bmatrix}$

s.t. $C + D^T\lambda + A^T v = 0$

$\lambda \geqslant 0$

گاهی مسئله دوگان از اصلی ساده‌تر می‌شود.

$\min\ c^T x$ $\quad \mathcal{L}(x,\lambda,v) = c^T x + \lambda^T(Ax - b) + v^T\begin{bmatrix} x_1(1-x_1) \\ \vdots \\ x_n(1-x_n)\end{bmatrix}$

s.t. $Ax \leqslant b$

$x_i \in \{0,1\}$

$\equiv x_i(1-x_i) = 1$

$$\frac{\partial L}{\partial x} = 0 \Rightarrow C + A^T\lambda + \begin{bmatrix} 1-2x_1 & & 0 \\ & 1-2x_2 & \\ 0 & & \ddots \ 1-2x_n\end{bmatrix} v = 0$$

از اینجا x_i به دست می‌آید. و بعد از روی آن g

$$\frac{1}{2}(C + A^T\lambda + v) = \begin{bmatrix} x_1 & 0 \\ 0 & x_n\end{bmatrix} v$$

را به دست می‌آوریم.

به راحتی بشه x_i ها به دست می‌آیند و جاگذاری می‌کنیم تا g به دست آید و مسئله dual به دست آید.

Subject:
Date

$d^* \le p^*$ (weak duality), $d^* = p^*$ (strong duality)

عموماً برای مسائل محدب strong duality برقرار است و برای مسائل غیرمحدب در دنیای واقعی برقرار نیست.

می‌توان شرایطی روی مسائل محدب گذاشت که شرط تحقق کند ها،

min $f_0(x)$
s.t.
$f_i(x) \le 0$
$Ax = b$

$f_0, f_1, \dots, f_n$ convex، strong duality برقرار باشد.

Slater Condition: اگر feasible set یک نقطه درونی داشته باشد

$$\{x \mid Ax = b,\ f_i(x) < 0,\ \forall i \in \{1, \dots, m\}\} \ne \emptyset$$

(strong duality) اگر شرط slater برقرار باشد، شرط کافی است برای $p^* = d^*$

If slater condition holds and strong duality

$x^* \longrightarrow p^*$ for primal
$(\lambda^*, \nu^*) \rightarrow d^*$ for dual

① $d^* = p^* \Rightarrow g(\lambda^*, \nu^*) = f_0(x^*)$

$$= \min_{x \in X} \left\{ f_0(x) + \sum \lambda_i^* f_i(x) + \nu^{*T}(Ax - b) \right\}$$ اگر min را

$$\le f_0(x^*) + \sum \underbrace{\lambda^*}_{\ge 0} \underbrace{f_i(x^*)}_{\le 0} + \nu^{*T} \underbrace{\cancel{(Ax^* - b)}}_{= 0}$$ برداریم و به جای x، x^* را

که یکی از مقادیر ممکن است را بگذاریم:

باید صفر باشد ← رشود ≤ 0

$f_0(x^*) < f_0(x^*)$ ~ تن

dual $2^2 \times 5 = 20$

Subject:
Date

$\Rightarrow \lambda_i^* f_i(x^*) = 0 \quad i = 1, \dots, m$

$\rightarrow f_i(x^*) < 0 \longrightarrow \lambda_i^* = 0$
$\rightarrow \lambda_i^* > 0 \longrightarrow f_i(x^*) = 0$
Complemetary Slackness شرط

تجمیع این شروط، شروط k.k.T را می‌سازند که شرط‌های لازم بهینه بودن است.

KKT Conditions $\quad f_i$ مشتق‌پذیر، h_i مشتق‌پذیر

① primal feasibility: $f_i(x^*) \leqslant 0, \; h_i(x^*) = 0$

② dual feasibility $\quad \lambda^* \geqslant 0$

③ $\lambda_i^* f_i(x^*) = 0 \quad$ Complementary slackness

④ Stationary $\quad \nabla_x L(x, \lambda, v)\Big|_{x^*, \lambda^*, v^*} = 0$

اگر برای مسئله محدب شروط k.k.T برقرار باشد، $p^* = d^*$. (شرط کافی هم می‌شود)

شروط k.k.T یک مسئله بهینه‌سازی را به مجموعه‌ای از معادلات جبری تبدیل می‌کند.

مثال:
$$\min \; \frac{1}{2} x^T Q x + c^T x \qquad Q \geqslant 0$$
s.t.
$$Ax = b$$
$$Dx \leq e$$

برای مسئله روبرو، kkT را بنویسیم.

PAPCO

Subject:
Date

خیلی وقت‌ها، با تعریف متغیر کمکی، نامساوی $Dx^* \leqslant e \quad , \quad Ax^* = b$ ③

را به مساوی تبدیل می‌کنیم: $Dx^* + S = e$ ① $, \quad S \geqslant 0$ ②

$Dx^* - e = -S$

$\lambda^* \geqslant 0$ ④ که برای شرط‌ها در بعد رتبه خوب است.

$\lambda^* \perp S \Rightarrow \lambda^{*T} S = 0$ ⑤ $\quad \dfrac{\partial L}{\partial x} = Qx^* + c + D^T\lambda^* + A^T v^* = 0$ ⑥

بهترین رفرنس برای Convex Optimization کتاب Boyd است. سایت و اسلاید مفصل هم دارد.

جلسه چهارم شنبه ۱۴۰۲/۱۲/۵ 2024-02-24

مثال: Box Constrained Linear Program

همه x_iها در یک Box هستند.

$$(P_1) \quad \min \; c^T x$$
$$\text{s.t.} \quad Ax = b$$
$$-1 \leqslant x_i \leqslant 1$$

دو مدل می‌توان مسئله را مورد بررسی قرار داد.

می‌توان محدودیت Box را به صورت قید دید یا در

دامنه مسئله اعمال اش کرد.

$$(P_2) \quad \min_{-1 \leqslant x_i \leqslant 1} \; c^T x$$
$$\text{s.t.} \quad Ax = b$$

For P_1: $L(x, \bar{\lambda}, \underline{\lambda}, v) = c^T x + v^T(Ax - b) + \bar{\lambda}^T(x - \vec{1}) + \underline{\lambda}^T(-x - \vec{1})$

dual D_1: $\dfrac{\partial L}{\partial x} = c + A^T v + \bar{\lambda} - \underline{\lambda} = 0$

PAPCO

dual 2x11 = 22

Subject:
Date

Dual 1: $\max \; -v^Tb - \bar{\lambda}^T 1 - \underline{\lambda}^T . 1$

s.t.

$$c + A^T v + \bar{\lambda} - \underline{\lambda} = 0$$

$$\bar{\lambda}, \underline{\lambda} \geq 0$$

مسئله دوگان را مانند قبل می‌توانیم بنویسیم

برای مسئله P2، x_i شرط روی دامنه رو دهد.

$L_2(x, v) = c^T x + v^T (Ax - b)$ Dual: $\max \; -\|A^T v + c\|_1 - b^T v$

$$g(v) = \min_{-1 \leq x_i \leq 1} c^T x + v^T (Ax - b)$$

چگونه پیدا می‌کنیم؟ از $g(v)$ مشتق می‌گیریم و برابر صفر می‌گذاریم که روی مرز می‌افتد.

جلوتر با استفاده از Conic optimization نشان می‌دهیم که چگونه می‌توان به فرم نوشته شده رسید.

در این مسئله، P2 به ظاهر ساده‌تر می‌باشد. شبیه به مسئله فرقی دارد که ما قیدها را در دامنه اعمال کنیم یا در شروط explicit. جواب یکسان خواهد بود.

- Optimization with Conic Inequalities:

$\min \; f_0(x)$

s.t.

$f_i(x) \preceq_{k_i} 0 \qquad i = 1, \dots, m$

$h_i(x) = 0 \qquad i = 1, \dots, p$

یادآوری: k_i مخروط، $f_i(x) \preceq_{k_i} 0$ iff $\forall x \in X, \; -f_i(x) \in k_i$

Subject:
Date

Lagrangian: $L(x, \lambda, v) = f_0(x) + \sum_{i=1}^{m} \langle \lambda_i, f_i(x) \rangle$

$$+ \sum_{j=1}^{p} v_i h_j(x)$$

Dual function: $g(\lambda_1, \dots, \lambda_m, v) = \underset{x \in X}{\min\limits_{(\inf)}} L(x, \lambda_1, \dots, \lambda_m, v)$

Dual problem

$$\max \; g \quad \text{(Concave)}$$

s.t.

$$\lambda_i \succeq_{k_i^*} 0$$

Convex

Concave شرط

مسئله دوگان Convex خواهد بود.

اگر (P) Convex باشد و شرط Slater در آن برقرار باشد، $d^* = p^*$ خواهد بود.

مثال ۲: مسئله Box Constraint را به صورت Conic حل کنیم:

(P_3) $\min \; c^T x$

s.t.

$Ax = b$

$-1 \leq x_i \leq 1 \longleftrightarrow -\begin{pmatrix} x \\ 1 \end{pmatrix} \preceq_{k_\infty} 0$

$L(x, v, [\lambda, \sigma]^T) = c^T x + v^T(Ax - b) - \lambda^T x - \sigma . 1$

$\lambda \to \mathbb{R}^n$, $\sigma \to \mathbb{R}$

چون قرار است در ضرب شوند $1 \in \mathbb{R}, x \in \mathbb{R}^n$

PAPCO

Subject:
Date

(D_3): $\max \quad -b^T v - \sigma$

s.t.

$$c + A^T v + \lambda = 0$$

$$+\begin{bmatrix}\lambda \\ \sigma\end{bmatrix} \preccurlyeq_{k_1} 0 \longrightarrow \|\lambda\|_1 \leq \sigma$$

$= k_\infty^*$

دوگان k_∞ می‌شود k_1 چون گفتیم

$$k_p \longrightarrow k_p^* = k_q \quad s.t. \ \frac{1}{p} + \frac{1}{q} = 1$$

مثال ۳: LMI طور : SDP

$$\min \ c^T x$$

s.t.

$$A_1 x_1 + \cdots + A_n x_n \preccurlyeq B$$

مثبت معین بودن ماتریس‌ها

$$A_1, A_2, \ldots, A_n, B \in S_n$$

متقارن

$\preccurlyeq$ برای ماتریس‌ها، بیشتر معین بودن است که شرط conic است و دوال آن دارای این طور است.

$$L = c^T x + \sum_{i=1}^{n} \langle \Lambda, A_i x_i \rangle - \langle \Lambda, B \rangle$$

$$g(\Lambda) = \begin{cases} -\langle \Lambda, B \rangle & c_i + \langle \Lambda, A_i \rangle = 0 \quad \text{for } \forall i \\ -\infty & \text{o.w.} \end{cases}$$

حال چون می‌خواهیم g را max را حساب کنیم، می‌خواهیم در شکل بالا قرار بگیریم.

Dual: $\max \quad -\langle B, \Lambda \rangle$

s.t.

$$c_i + \langle \Lambda, A_i \rangle = 0$$

$$\Lambda \succcurlyeq 0$$

قلمی‌ها را هم می‌توانستیم و راه صورت دو ضابطه را شکل اینجا بنویسیم.

PAPCO

 $= 5^2$ pure

Subject:
Date

KKT conditions for a conic optimization

اگر f_i ، h_i مشتق‌پذیر باشند، شروط زیر، شروط لازم بهینگی خواهند بود:

① $f_i(x^*) \preceq_{k_i} 0$, $h_i(x^*) = 0$ ② $\lambda^* \succeq_{k_i} 0$

③ $\langle \lambda_i^*, f_i(x^*) \rangle = 0$, $\langle v_i^*, h_i(x^*) \rangle = 0$

④ $\nabla_x L \Big|_{x^*, v^*, \lambda^*} = 0$

اگه مسائل را می‌توان به فرم LMI/SDP تبدیل کرد.

مثال: $x^T Q x + b^T x + c \leq 0$, $Q \in S_n^+$

با استفاده از Shor Compliment می‌شود:

$$\equiv \begin{bmatrix} c + b^T x & x^T \\ x & -Q^{-1} \end{bmatrix} \preceq 0$$

برای اینکه محدب شود، باید $c^T x + d \geq 0$ باشد.

مثال: $\|Ax+b\|_2 \leq c^T x + d \equiv (Ax+b)^T(Ax+b) \leq (c^T x+d)^2$

$$\equiv \begin{bmatrix} (c^T x + d) I & Ax+b \\ (Ax+b)^T & c^T x + d \end{bmatrix} \preceq 0$$

Subject:
Date

- Uncertain Linear Optimization Problems:

LO

$$\min_x \{ c^T x + d \mid Ax \le b \} \quad c \in \mathbb{R}^n,\ d \in \mathbb{R},\ A \in \mathbb{R}^{m\times n},\ b \in \mathbb{R}^m$$

در مسئله LP اگر ابهامی روی پارامترهای A، b، c، d نداشته باشیم، d هیچ تاثیری

در حل مسئله ندارد ولی در مسئله uncertain، نمی‌توان به آن استدلال، d را حذف کرد.

Data Matrix $D = \left[\begin{array}{c|c} c^T & d \\ \hline A & b \end{array}\right]$ در مسئله فوق، قید تساوی نگذاشتیم زیرا

می‌توان $Ex = f$ را به صورت

$$\begin{bmatrix} E \\ -E \end{bmatrix} x \le \begin{bmatrix} f \\ -f \end{bmatrix}$$ تبدیل به قیدهای نامساوی کرد: $Ex \ge f$ و $Ex \le f$

گفتیم که ممکن است Data Matrix الان دارای خطا باشد، به چند دلیل

(I) خطای اندازه‌گیری در به دست آوردن پارامترهای سیستم.

(II) برخی از پارامترها از تخمین‌های بیانیه (مانند میزان مصرف در آینده، بازگشت سرمایه، ...)

(III) خطای پیاده سازی. Implementation error. معمولاً این نوع خطا را به خطاهای

نوع سوم تبدیل می‌کنیم و مستقیم خطای نوع سوم را در نظر نمی‌گیریم. مثلاً خطای نوع

سوم که به جای x_i، $x_i + \varepsilon$ اعمال می‌کنیم، تبدیل می‌کنیم روی شرط به صورت:

در صورت نیاز، روی باقی پارامترهای c، d هم اعمال می‌شود. $b_j \to b_j - a_{ji}\,\varepsilon$

PAPCO

$27 = 3^3$ pure

Subject:
Date

اگر از نوع ضرب بود: $x_i \to (1+\varepsilon)x_i \longrightarrow (a_{ji} \to a_{ji}(1+\varepsilon))$

پس مسئلهٔ بهینه سازی میشود به صورت زیر:

LO_u

$$\left\{ \min_x \{ c^T x + d \mid Ax \le b \} \right\}_{(c,d,A,b) \in V \subseteq \mathbb{R}^{(m+1)\times(n+1)}}$$

معمولاً V را به صورت زیر نشان میدهیم:

$$V = \left\{ \begin{bmatrix} c^T & d \\ A & b \end{bmatrix} = \underbrace{\begin{bmatrix} c_0^T & d_0 \\ A_0 & b_0 \end{bmatrix}}_{D_0 \text{ مقدار نامی}} + \sum_{l=1}^{L} z_l \underbrace{\begin{bmatrix} c_l^T & d_l \\ A_l & b_l \end{bmatrix}}_{\text{certain مقادیر داشته}}, \right.$$

$$\left. z = \underbrace{\begin{bmatrix} z_1 \\ \vdots \\ z_L \end{bmatrix}}_{\text{عامل نا\text{ }یقینی که در مجموعه } Z \text{ است}} \in \underbrace{Z}_{\text{known}} \subseteq \mathbb{R}^L \right\}$$

این مدل توصیف برای خیلی مسائل کافی و مناسب است.

اگر Z به صورت روبرو باشد $Z = p + Qy$

داریم:

$$V = \left\{ [D_0 + \sum p_l D_l] + \sum y_k [Q_{lk} D_l], \; y \in Y \right\}$$

به عبارتی میتوانیم با تبدیل خطی، شکل ها را به شکل های ساده تر تبدیل کنیم که تحلیل ساده شود. یعنی اگر بعدا همیشه فرض کردیم مربع واحد است، فرض غیر معقول نیست.

PAPCO

دایره واحد $Z \longrightarrow$ بیضی Y ، مربع واحد $Z \longrightarrow$ متوازی الاضلاع Y

با انتخاب تبدیل مناسب المنت تبدیلات فوق را میتوان اعمال کرد.

مثال:

$$A = \begin{bmatrix} 2-6 & 7-9 \\ 3-7 & 10-12 \end{bmatrix}$$

$$U = \left\{ \begin{bmatrix} c^T & d \\ \begin{bmatrix} 4 & 8 \\ 5 & 11 \end{bmatrix} & b \end{bmatrix} + z_1 \begin{bmatrix} 0 & 0 \\ \begin{bmatrix} 2 & 1 \\ 2 & 1 \end{bmatrix} & 0 \end{bmatrix} \;\middle|\; \|z\|_\infty \leq 1 \right\},$$

میشود به صورت زیر هم نوشت،

$$U = \left\{ \text{"} + z_1 \begin{bmatrix} 0 & 0 \\ \begin{bmatrix} 2 & 0 \\ 0 & 0 \end{bmatrix} & 0 \end{bmatrix} + z_2 \begin{bmatrix} 0 & 0 \\ \begin{bmatrix} 0 & 0 \\ 2 & 0 \end{bmatrix} & 0 \end{bmatrix} + z_3 [\text{مشابه}] + z_4 [\text{مشابه}] \;\middle|\; -1 \leq z_i \leq 1 \right\}$$

جلوتر خواهیم دید که حل با این دو مجموعه U یکی نشود، شاید.

- تعریف: x_0 is a robust feasible solution for LO_U, if

$$A x_0 \leq b \quad \text{for all } (c, d, A, b) \in U$$

- تعریف: For a candidate x_0, $\hat{c}(x)$ is called a robust value:

$$\hat{c}(x_0) = \sup_{(c,d,A,b) \in U} c^T x_0 + d$$

Robust Optimization

در مسائل RO، به دنبال x^* ای هستیم که اولاً robust-feasible باشد و ثانیاً robust-value کمینه داشته باشد. داریم worst-case را بررسی می‌کنیم

همتای مقاوم Robust Counterpart

$$\min_{x} \left\{ \hat{c}(x) = \sup_{(c,d,A,b)\in U} \left[c^T x + d\right] \;\middle|\; Ax \le b,\ \forall (c,d,A,b)\in U \right\}$$

قبلاً گفتیم که،

$$\text{LO: } \min\ c^T x + d \quad \text{s.t. } Ax \le b \quad \longleftrightarrow \quad \min\ t \quad \text{s.t. } c^T x + d \le t,\ Ax \le b$$

با این تکنیک، تابع هدف دیگر عدم قطعیت نخواهد داشت.

به همین دلیل، از این به بعد در هر مسئله بهینه سازی مقاوم برای LP، فرض می‌کنیم تابع هدف معلوم است.

Robust Counterpart (RC):

$$\min_{x} \left\{ c^T x \mid Ax \le b,\ \forall (A,b)\in U \right\}$$

در ادامه سعی می‌کنیم شرط را ساده‌تر بنویسیم و به حل آن بپردازیم.

یعنی با معادل سازی، تا الان عدم قطعیت در تابع هدف را برداشتیم، شرط‌های تساوی را هم برداشتیم، شرط باقی مانده را هم ساده‌تر می‌نویسیم به صورت $a_i^T x \le b_i$ و بعد این مسئله ساده‌تر را حل می‌کنیم.

در این درس، خیلی Optimization حل نمی‌کنیم و فرض می‌کنیم بلدیم و فقط هدفمان این است که مسئله RO را به مسئله Convex Opt. تبدیل کنیم و بعد آن را دیگر در این درس حل نمی‌کنیم.

Subject:
Date

جلسه پنجم دوشنبه ۱۴۰۲/۱۲/۷ 2024-02-26

تا اینجا گفتیم می‌توانیم مسئله استاندارد LP را به فرم روبرو بنویسیم، بدون این که کلیت از دست برود.

$$\min c^T x$$
$$Ax \leq b \quad \{A, b\} \in U$$

حال می‌خواهیم شرط $Ax \leq b$ را ساده‌تر کنیم.

سطر i ام شرط $\quad Ax \leq b \longleftrightarrow (Ax)_i \leq b_i \quad i = 1, \dots, m$

$$\longleftrightarrow a_{i*}^T x \leq b_i \quad i = 1, \dots, m \quad [a_{i*}, b_i] \in U_i$$

یعنی همه‌ی الان‌ها که معادل سطرهاست.

$[A, b] \in U \quad [a_{i*}, b_i] \in U_i \longrightarrow U_i$ projection of U.

از روی U، $\hat{U}$ را می‌سازیم $\quad \Rightarrow \hat{U} = U_1 \times U_2 \times \dots \times U_m$

$$\begin{matrix} Ax \leq b \\ [A, b] \in \hat{U} \end{matrix} \quad \equiv \quad \begin{matrix} Ax \leq b \\ [A; b] \in U \end{matrix}$$

مثال، چگونه تفکیک کنیم؟ $\quad X = \{x_1 \geq z_1, \; x_2 \geq z_2 \mid z \in U\}$

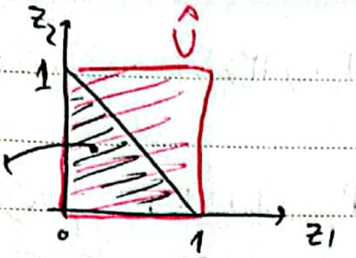
$$z \in U = \{z \mid z_1 + z_2 \leq 1, \; z_1, z_2 \geq 0\}$$

$$\begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} x \leq \begin{bmatrix} -z_1 \\ -z_2 \end{bmatrix} \qquad \left\{ \begin{matrix} b_1 + b_2 \geq -1 \\ b_1 \leq 0, \; b_2 \leq 0 \end{matrix} \right\} = U$$

سطر اول $\quad -x_1 \leq -z_1$

$$\Rightarrow x_1 \geq z_1$$

محدوده Z :



projection on z_1 : $[0,1]$

، z_2 : $[0,1]$

شرط اول : $x_1 \geq z_1$

$0 \leq z_1 \leq 1$: U_1

$\hat{U} = U_1 \times U_2$

شرط دوم $x_2 \geq z_2$

$0 \leq z_2 \leq 1$ U_2

اگر مسئله را برای U حل کنیم یا برای $\hat{U}$ حل کنیم، به جواب یکسان خواهیم رسید.

می‌توان ثابت کرد (اینجا ثابت نکرده‌ام.) به همین دلیل، بدون از دست رفتن

کلیت مسئله، از این به بعد می‌توانیم فقط یک شرط بگذاریم.

هدف چیست؟ ما یک مسئله داریم با نایقینی مثل $f_1(x) \leq 0$ که در دل خود
$f_2(x) \leq 0$
نایقینی دارد. می‌خواهیم هر شرط را، معادل یک شرط بدون نایقینی بنویسیم و بعد

آن‌ها را تک‌تک و جداگانه حل کنیم و قضیه می‌گوید جواب یکی می‌شود.

x : robust_feasible

به صورت ریاضی وار:

$$\forall [A,b] \in U : Ax \leq b \leftrightarrow (Ax)_i \leq b_i$$

$$\leftrightarrow a_{i*}^T x \leq b_i \quad \forall [a_{i*}, b_i] \in U_i = Proj(U).$$

$$\leftrightarrow Ax \leq b \quad \forall [A;b] \in U_1 \times U_2 \times \dots \times U_m$$

Subject:
Date

چون قرار است شرط به ازای همه z های عضو U برقرار بماند.

قضیه دیگر می‌گوید اگر U_i را به $Conv(U_i)$ تبدیل کنیم، محدوده feasible تغییر نمی‌کند.

به صورت ~~...~~ ریاضی، یعنی اگر x نقطه‌ی robust feasible باشد، یعنی

$(Ax)_i \leq b_i$ به ازای $[a_{i+}; b_i] \in U_i$، در این صورت $(Ax)_i \leq b_i$

خواهد بود به ازای $[a_{i+}, b_i] \in Conv(U_i)$.

$$[\bar{a}_i ; \bar{b}_i] = \sum_{j=1}^{J} \lambda^j [a_i^j ; b_i^j] \quad s.t. \quad \text{که}:$$

$$\sum \lambda_j = 1, \quad \lambda_j \geq 0$$

$$[a_i^j, b_i^j] \in U_i$$

از اینجا

$$(a_i^j)^T x \leq b_i^j$$

تساوی‌ها × λ^j چون مثبت است شکل نمی‌شود

$$\lambda^j (a_i^j) \leq \lambda^j b_i^j \xrightarrow{\text{sum over } j} \sum_j \lambda^j (a_i^j) \leq \sum \lambda^j b_i^j$$

$\rightarrow \bar{a}_i^T x \leq \bar{b}_i \longrightarrow x$ is robust feasible for $Conv(U_i)$

به طریق مشابه به صورت مرزی، می‌توانیم مرز محدوده را هم اضافه کنیم، یعنی

به جای U_i که Convex hull U_i را گذاشته بودیم، مرزش را هم اضافه کنیم.

~~...~~ آخر سر یعنی Convex & Closed می‌شود.

PAPCO

33 = 3×11 dual

Subject:
Date

$U_i = \left\{ [a_i ; b_i] = [a_i^0 ; b_i^0] + \sum_{l=1}^{L} z_l [a_i^l ; b_i^l],\ z \in Z \right\}$: نهایتاً می‌رسیم به

where Z is convex and closed.

تبدیلات :

$$a_1 x_1 + a_2 x_2 = b \iff \begin{cases} a_1 x_1 + a_2 x_2 \leq b \\ a_1 x_2 + a_2 x_2 \geq b \end{cases}$$

این تبدیل هم برای حالت certain و هم حالت uncertain جواب می‌دهد.

برای شرط نامساوی :

$$a_1 x_1 + a_2 x_2 \leq b \iff \begin{cases} a_1 x_1 + a_2 x_2 + s = b \\ s \geq 0 \end{cases}$$

در حالت certain می‌توانیم این تبدیل را انجام دهیم ولی آیا در حالت uncertain هم ممکن است؟

x is robust feasible if $\forall a_1, a_2, b \in U$ s.t. $a_1 x_1 + a_2 x_2 \leq b$

$\longleftrightarrow \quad \forall a_1, a_2, b \in U \quad \exists s \geq 0 \quad a_1 x_1 + a_2 x_2 + s_2 = b$

ولی دومی (که برای حالت certain استفاده می‌کردیم) می‌شود معادل این که

$\exists s \geq 0 : \forall a_1, a_2, b \in U \longrightarrow a_1 x_1 + a_2 x_2 + s = b$

در اولی، s تابعی از (a_1, a_2, b) می‌شود. در دومی اول s را پیدا می‌کند.

پس شرط دوم، شرط محافظه‌کارانه‌تری است و معادل نیست.

dual: $2 \times 17 =$ 34

Subject:
Date

خیلی وقت‌ها در مسائل Robust، می‌رسیم به مسئلهٔ روبرو: $\forall \dots \exists \dots$

ولی حل آن NP-hard است و از سر اجبار، جابجا می‌کنیم تا بشود $\exists \dots \forall \dots$

که تقریب محافظه‌کارانه‌تری است ولی برای اینکه بتوانیم حل کنیم، چاره‌ای نیست.

مسئله پس تبدیل شد به:

$$\min \; c^T x$$
$$\text{s.t.} \quad a_i^T x \leq b_i \qquad [a_i ; b_i] \in V$$

مسئلهٔ روبرو، معادل این است که بخواهیم

∞ تا شرط را چک کنیم. خیلی وقت‌ها این مسئله NP-hard است. یعنی در زمان

چندجمله‌ای قابل حل نیست.

(روش کلاسیک اثبات NP-hard بودن یک مسئله این است که مسئله‌مان را تبدیل می‌کنیم

و نشان می‌دهیم معادل یک مسئلهٔ شناخته‌شدهٔ NP-hard است.)

مثال: مجموعهٔ شدنی بیابید: $\{\|Px - q\|_1 \leq 1\}$

P is known. $q \in \{q = Bz, \; \|z\|_\infty \leq 1\}$

B is known, and Positive semi-definite.

حتی نمی‌خواهد feasible set این را پیدا کنید. فقط جواب (هر چه که باشد)

آیا $x = 0$، robust feasible هست یا نه؟ (یعنی به ازای همهٔ مقادیر ممکن q شرط برقرار

PAPCO

 $= 5 \times 7$ dual

$x=0 \Rightarrow \|Bz\|_1 \leqslant 1 \quad \forall \|z\|_\infty \leqslant 1$

می‌توان ثابت کرد که این شرط معادل است با (با استفاده از رابطه روبرو) $\|u\|_1 = \max\limits_{\|\eta\|_\infty \leqslant 1} \eta^T u$

$$\max \eta^T . Bz \leqslant 1$$

$$\forall \|\eta\|_\infty , \|z\|_\infty \leqslant 1$$

می‌توان ثابت کرد که max در $z=\eta$ رخ می‌دهد، پس می‌شود:

$$\max z^T B z \leqslant 1 \quad \forall \|z\|_\infty \leqslant 1$$

قبلاً ثابت شده است که این مسئله که رسیدیم، NP-hard است، پس مسئله اصلی NP-hard است که اصلاً خوب نیست، چون در عمل حل نمی‌شود.

مجبوریم که شرط مسئله را تبدیل به یک سری شروط معادل کنیم و بعد تقریب بزنیم تا بتوانیم حل کنیم

یک سری متغیر u اضافه می‌کنیم تا دستمان باز شود:

$$X^+ \subseteq \mathbb{R}^n_x \times \mathbb{R}^m_u$$

X^+ بازنمایی $X \subseteq \mathbb{R}^n_x$ است اگر

یعنی تقریب فعلاً نزده‌ام

تصویر X^+ روی صفحه $\mathbb{R}^n_x$ دقیقاً همان X باشد. به صورت ریاضی:

$$x \in X \longleftrightarrow \exists u \in \mathbb{R}^m_u : (x,u) \in X^+$$

فعلاً تقریب و تخمین نزده‌ام. با مثال بهتر مشخص می‌شود.

Subject:
Date

مثال: اگر به مدل 1، معادل بگیریم، تعداد شروط $|x_1|+\cdots+|x_n| \leqslant 1$

معادل است

خیلی زیاد میشود که قابل حل نیست. مجموعاً تا 2^n محدودیت میشود که خیلی زیاد است.

با

مدل ①

$$\begin{cases} x_1+\cdots+x_n \leqslant 1 \\ -x_1+\cdots+x_n \leqslant 1 \\ x_1-x_2+\cdots+x_n \leqslant 1 \\ \vdots \end{cases}$$

مدل 2 را در نظر بگیریم:

مدل ②:

$$\begin{cases} -u_1 \leqslant x_1 \leqslant u_1 \\ -u_2 \leqslant x_2 \leqslant u_2 \\ \vdots \\ -u_n \leqslant x_n \leqslant u_n \\ \sum u_i \leqslant 1 \end{cases}$$

$2n+1$ شرط است. مدل 2، گویا بازنمایش معادلات قبلی است و

$\mathbb{R}^n \leftarrow \text{Proj}\{\mathbb{R}_n \times \mathbb{R}_u\}$

تعریف هر x^* که در مسئله اصلی صدق کند، حتماً (x^*,u) ای پیدا میشود که در مدل 2 صدق میکند. و هر (x^*,u) که در معادلات مدل 2 صدق کند، x^* حتماً در مسئله اصلی صدق میکند.

$$X = \{x \mid \exists v: (x,u) \in X^+\} \qquad X = \text{Proj}(X^+)$$

با این تعریف، میتوان مسئله اصلی را بازنویسی کرد:

$$\min_x \{f(x);\ x \text{ satisfies } S_i,\ i=1,\dots,m\}$$

معادل است با

$$\min_{x, v_1,\dots,v_m} \{f(x);\ (x,v_i) \text{ satisfies } S_i^+,\ i=1,\dots,m\}$$

با بررسی یک یا چند مثال، دقیق‌تر متوجه منظور خواهیم شد.

Subject:
Date

مثال: مسئله استاندارد RO را که داشتیم، $\{a^T x \le b\}$, $[a;b] \in U$

مرخواهیم شرط مربع را بازنویسی کنیم:

$$U = \{[a;b] = [a^0;b^0] + \sum_{l=1}^{L} z_l [a^l;b^l]\}$$

شرط

$$Z = \{z \in \mathbb{R}^L \mid \|z\|_\infty \le 1\}$$

robust feasibility را

می‌نویسیم:

$$a^T x \le b \quad \forall \|z\|_\infty \le 1$$

$$(a^0)^T x + \sum z_l (a^l)^T x \le b^0 + \sum z_l b^l \quad \forall \|z\|_\infty \le 1$$

$$\sum z_l [(a^l)^T x - b^l] \le b^0 - (a^0)^T x \quad \forall \|z\|_\infty \le 1$$

①

$$\max_{\|z\|_\infty \le 1} \sum z_l [(a^l)^T x - b^l] \le b^0 - (a^0)^T x$$

$\equiv$

$|z_1| \le 1, \dots, |z_L| \le 1$

② $\displaystyle\sum_{l=1}^{L} |(a^l)^T x - b^l| \le (a^0)^T x - b^0$

با تبدیلاتی که اعمال کردیم، دیگر $\forall$ ها از بین رفت و الان یک شرط داریم که همگی certain هستند. با تکنیک گفته شده، abs را هم بر می‌داریم.

که تبدیل شد به:

$$-u_l \le (a^l)^T x - b^l \le u_l$$

$$\sum u_l \le (a^0)^T x - b^0$$

$2L - 1$ شرط معلوم می‌شود

certain constraint

و می‌توانیم بدهیم به solver های استاندارد و حل اش کنیم.

PAPCO

dual $2 \times 19 = 38$

Subject:
Date

در ① : وقتی قرار است به ازای $\forall z$ ها برقرار باشد، می‌توانیم برای $\max_z$ بگیریم باید برقرار باشد.

در ② : برای رسیدن به max، باید z_i به مرزش بچسبد، یا $+1$ یا -1 بسته به اینکه ضریب اش مثبت باشد یا منفی که با قدر مطلق نشان دادیم اش.