Minimizing Quadratic Functions in Constant Time (NIPS’16)

ERATO感謝祭 Season IV

吉田悠一
(国立情報学研究所)

定数時間アルゴリズムとは？

性質検査

- 与えられた入力がある性質$\mathcal{P}$を満たすか判定したい
- 入力が巨大な時は線形時間でさえ遅すぎる

性質を「満たす」か「満たすにはほど遠い」かを
区別すれば良いことにして速くならないか？

性質検査
[Blum et al.'93; Rubinfeld and Sudan'96]

多くの性質が入力長に依存しない計算時間 = 定数時間で
判定できる！

性質検査の研究者としての憂い

理論的に非常に綺麗

- 他の数学との関連 (e.g., 加法的組合せ論)
- 定数時間で検査できる性質の必要十分条件
  - グラフ [Alon et al. SICOMP'09]
  - 有限体上の関数 [Y. STOC'14] [Y. SODA'16]
  - 制約充足問題 [Y. STOC'11] [Chen-Valeriote-Y. FOCS'16]
  - 関係データベース [Chen-Y. 2017]

全く応用されていない

理由（？）

- 離散的な対象ばかり
- 判定問題ばかり

連続最適化問題を解こう！

二次関数最小化の定数時間近似

二次関数最小化

二次関数

$$f(\boldsymbol{v}) = \langle \boldsymbol{v}, A\boldsymbol{v} \rangle + \langle \boldsymbol{b}, \boldsymbol{v} \rangle = \sum_{i,j} A_{ij} v_i v_j + \sum_i b_i v_i$$

の”最小値”の近似を定数時間で行う

<u>応用</u>

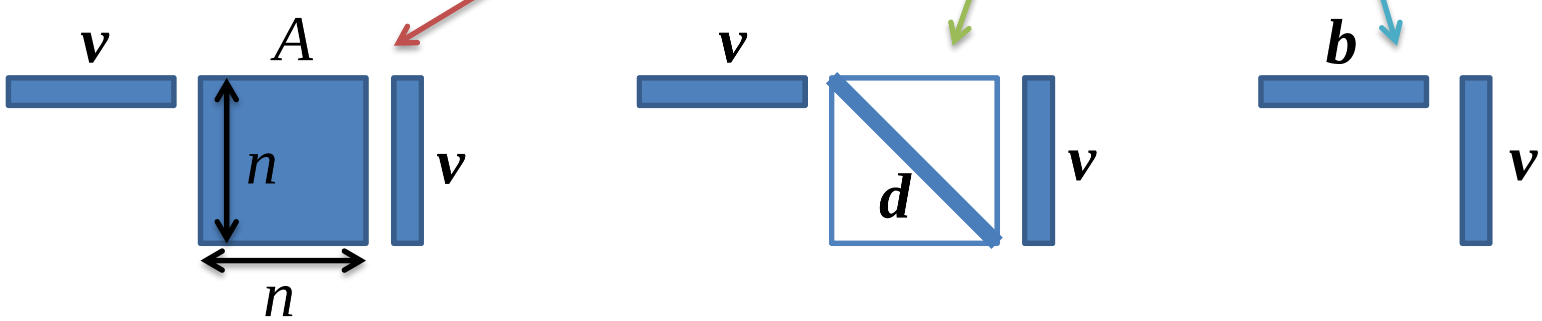
- 線形回帰 (注: 一般には最小解に興味がある)
- ピアソン偽距離(分布間の距離)の推定
- 特徴選択

これらの応用では次元数nはしばしば大きい

問題設定

目的関数

$$p_{n,A,\boldsymbol{d},\boldsymbol{b}}(\boldsymbol{v}) = \frac{1}{n^2}\left(\langle \boldsymbol{v}, A\boldsymbol{v}\rangle + n\langle \boldsymbol{v}, \mathrm{diag}(\boldsymbol{d})\boldsymbol{v}\rangle + n\langle \boldsymbol{b}, \boldsymbol{v}\rangle\right)$$



- $\|A\|_{\max}$ $(:= \max_{i,j}|A_{ij}|) = \|\boldsymbol{d}\|_\infty = \|\boldsymbol{b}\|_\infty = O(1)$ とする

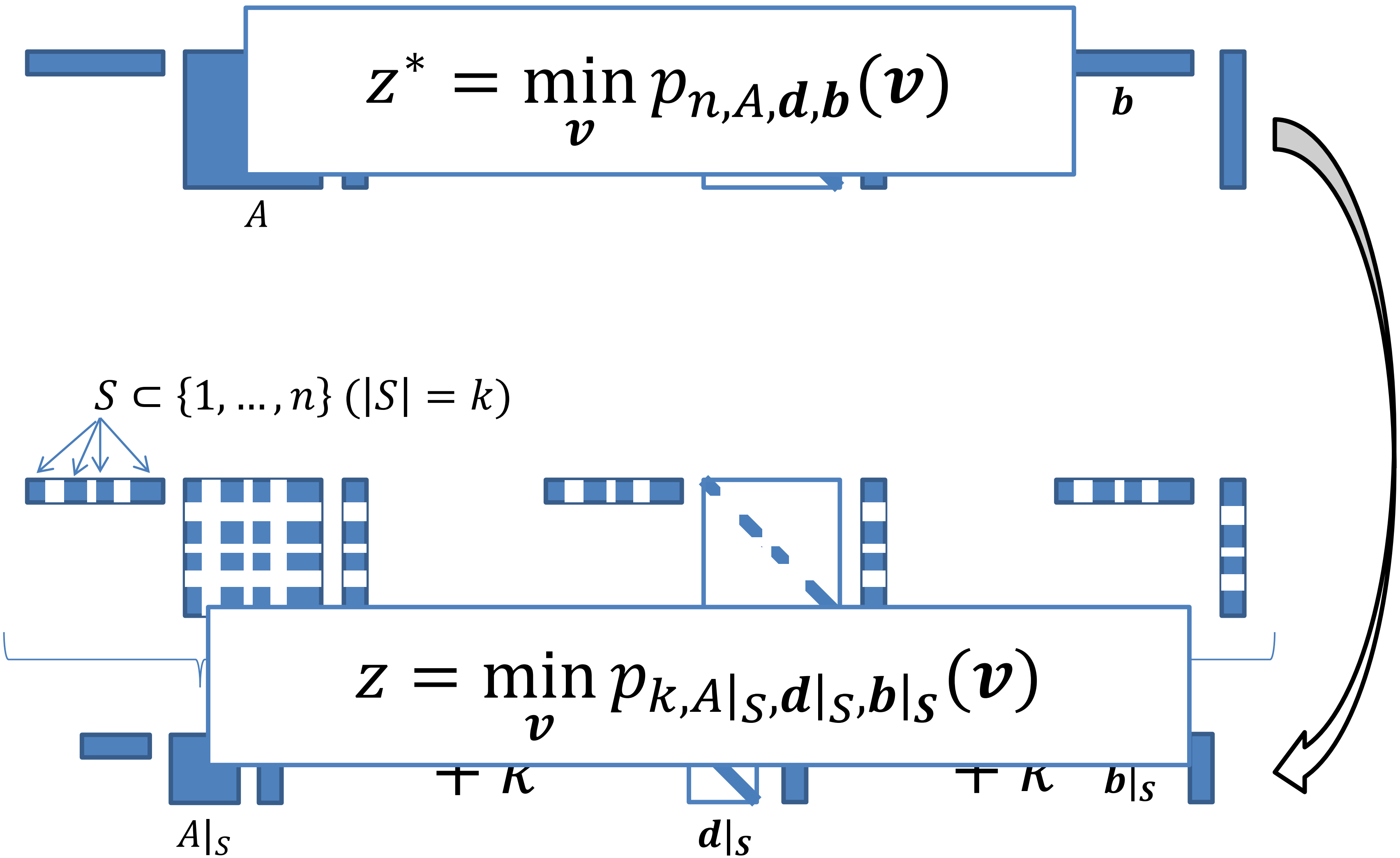
目標

$z^* = \min_{\boldsymbol{v}} p_{n,A,\boldsymbol{d},\boldsymbol{b}}(\boldsymbol{v})$を定数時間で近似する

提案アルゴリズム

もとの問題をサンプリングして厳密に解く（超簡単！）

$$z^* = \min_{\boldsymbol{v}} p_{n,A,\boldsymbol{d},\boldsymbol{b}}(\boldsymbol{v})$$



$$z = \min_{\boldsymbol{v}} p_{k,A|_S,\boldsymbol{d}|_S,\boldsymbol{b}|_S}(\boldsymbol{v})$$

主定理

[定理] $\boldsymbol{v}^*, \widetilde{\boldsymbol{v}}^*$を元の問題/サンプル後の問題の最適解とし、$K_\infty = \max\{\|\boldsymbol{v}^*\|_\infty^2, \|\widetilde{\boldsymbol{v}}^*\|_\infty^2\}$とする。
任意の$\epsilon \in (0,1)$ に対し、サンプル数$k = 2^{O(1/\epsilon^2)}$とすると、確率0.99で以下を満たす。

$$|z^* - z| \leq \epsilon K_\infty$$

厳密　近似

手法	計算量	注意
厳密計算	$O(n^3)$	
確率勾配法	$O(nt), t \in \mathbb{N}$	t: 反復回数
Nystrom近似	$O(nk^2 + k^3)$	k: ランク
提案法	$\boldsymbol{O}\left(\boldsymbol{k}(\boldsymbol{\epsilon})^{3}\right) = \boldsymbol{O}(\mathbf{1})$	

アルゴリズムの正しさの証明概要

[目標] 十分大きなkに対して

$$\min_{\boldsymbol{v}\in\mathbb{R}^n} p_{n,A,\boldsymbol{d},\boldsymbol{b}}(\boldsymbol{v}) \approx \min_{\boldsymbol{v}\in\mathbb{R}^k} p_{k,A|_S,\boldsymbol{d}|_S,\boldsymbol{b}|_S}(\boldsymbol{v})$$

元問題とサンプル後の問題が「近い」と言いたい

$$A \approx A|_S,\ \boldsymbol{d} \approx \boldsymbol{d}|_S,\ \boldsymbol{b} \approx \boldsymbol{b}|_S$$

次元が違うので普通のノルムでは距離が測れない

Graph limit theory (Graphon)を用いて同じ空間に埋め込む

Dikernel

(可測)関数$\mathcal{W}$: $[0,1]^2 \to \mathbb{R}$をDikernelと呼ぶ

任意の行列$A \in \mathbb{R}^{n\times n}$からDikernel $\mathcal{A}$: $[0,1]^2 \to \mathbb{R}$が作れる

3×3行列

$$\begin{pmatrix} 9.4 & -0.4 & 0.5 \\ 1.0 & 0.7 & -0.7 \\ 5.2 & -0.4 & 7.2 \end{pmatrix}$$

Dikernel

(0,0) (0,1)

9.4	-0.4	0.5
1.0	0.7	-0.7
5.2	-0.4	7.2

(1,0) (1,1)

距離が定義可能

5×5行列

$$\begin{pmatrix} -2.4 & -3.9 & 2.5 & 4.7 & 9.4 \\ -0.7 & 0.2 & -6.1 & -2.2 & -2.6 \\ 0.1 & 7.0 & 3.6 & 5.0 & -8.4 \\ 0.4 & 3.9 & -6.6 & -1.4 & -8.3 \\ 6.8 & -5.4 & 6.2 & -7.3 & 1.3 \end{pmatrix}$$

Dikernel

(0,0) (0,1)

-2.4	-3.9	2.5	4.7	9.4
-0.7	0.2	-6.1	-2.2	-2.6
0.1	7.0	3.6	5.0	-8.4
0.4	3.9	-6.6	-1.4	-8.3
6.8	-5.4	6.2	-7.3	1.3

(1,0) (1,1)

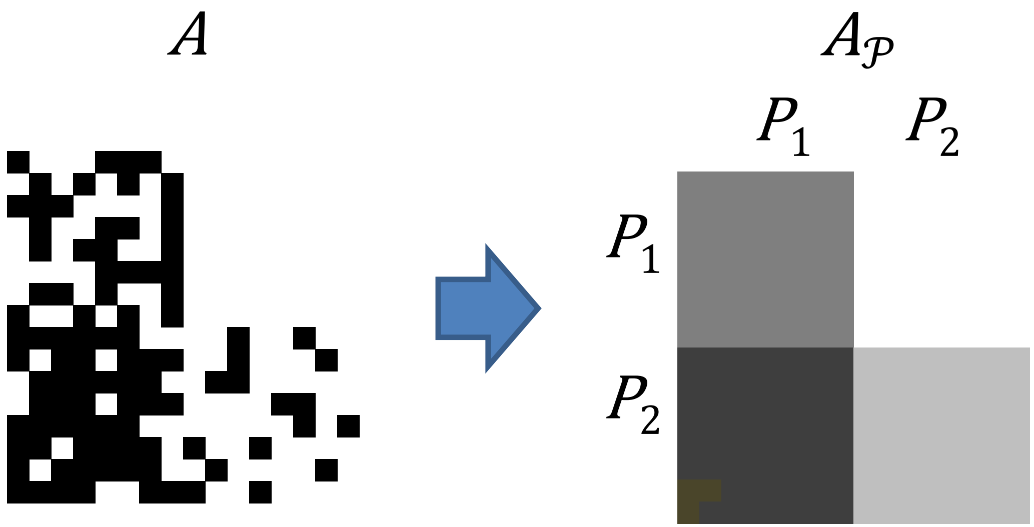
主補題

[補題] kが大きい → $\|\mathcal{A}-\mathcal{A}|_S\|_\square$は小さい

- $\mathcal{A}|_S$: $A|_S$に対応するDikernel
- カットノルム: $\|\mathcal{W}\|_\square = \sup_{S,T\subseteq[0,1]} \left|\int_S \int_T \mathcal{W}(x,y)dxdy\right|$

主補題の証明: (弱)正則性補題

[定理] 任意の行列$A \in \mathbb{R}^{n\times n}$は$\{1,2, \ldots, n\}$ の定数個の部分への等分割$\mathcal{P}=(P_1, \ldots, P_k)$を持ち、以下を満たす。



カットノルム
$\|\mathcal{A} - \mathcal{A}_{\mathcal{P}}\|_\square$
が小さい

ブロック内を平均化

主補題の証明

[主補題] kが大きい → $\|\mathcal{A} - \mathcal{A}|_S\|_\square$ は小さい

$\mathcal{S}$は$\mathcal{P}$の各部分から十分な個数のサンプルを得る

$\Downarrow$

$\mathcal{A}|_S$は$\mathcal{A}_\mathcal{P}$の良い近似

$\Downarrow$

$\mathcal{A}_\mathcal{P}$も$\mathcal{A}$の良い近似なので$\|\mathcal{A} - \mathcal{A}|_S\|_\square$は小さい

実験：ピアソン偽距離の近似

分布間のピアソン偽距離のカーネルを用いた近似

以下の問題に帰着される

$$\min_{v\in\mathbb{R}^n} \frac{1}{2}\langle \boldsymbol{v}, H\boldsymbol{v}\rangle - \langle \boldsymbol{h}, \boldsymbol{v}\rangle + \frac{\lambda}{2}\|\boldsymbol{v}\|_2^2$$

- $H, \boldsymbol{h}$: カーネル関数から決まる行列とベクトル
- λ: ℓ^2正則化パラメータ

平均の異なるガウシアンの偽距離の近似で実験

Nystrom近似と比較

精度

Table 2: Pearson divergence: absolute approximation error.

	k	$n = 500$	1000	2000	5000
Proposed	20	0.0027 ± 0.0028	0.0012 ± 0.0012	0.0021 ± 0.0019	0.0016 ± 0.0022
	40	0.0018 ± 0.0023	0.0006 ± 0.0007	0.0012 ± 0.0011	0.0011 ± 0.0020
	80	0.0007 ± 0.0008	0.0004 ± 0.0003	0.0008 ± 0.0008	0.0007 ± 0.0017
	160	0.0003 ± 0.0003	0.0002 ± 0.0001	0.0003 ± 0.0003	0.0002 ± 0.0003
Nyström	20	0.3685 ± 0.9142	1.3006 ± 2.4504	3.1119 ± 6.1464	0.6989 ± 0.9644
	40	0.3549 ± 0.6191	0.4207 ± 0.7018	0.9838 ± 1.5422	0.3744 ± 0.6655
	80	0.0184 ± 0.0192	0.0398 ± 0.0472	0.2056 ± 0.2725	0.5705 ± 0.7918
	160	0.0143 ± 0.0209	0.0348 ± 0.0541	0.0585 ± 0.1112	0.0254 ± 0.0285

kの意味

- 提案手法 = サンプル数
- Nystrom近似 = ランク

同じkだとNystrom近似よりも精度が良い

- Nystrom近似は二次関数最小化の為に設計されていない

実行時間

Table 1: Pearson divergence: runtime (second).

	k	$n = 500$	1000	2000	5000
Proposed	20	0.002	0.002	0.002	0.002
	40	0.003	0.003	0.003	0.003
	80	0.007	0.007	0.008	0.008
	160	0.030	0.030	0.033	0.035
Nyström	20	0.005	0.012	0.046	0.274
	40	0.010	0.022	0.087	0.513
	80	0.022	0.049	0.188	0.942
	160	0.076	0.116	0.432	1.972

- 同じkだとNystromよりも100～1000倍高速

まとめ

二次関数最小化問題に対して理論保証のある定数時間アルゴリズムを与えた

その後の展開

- テンソルのTucker分解
- 固有値計算

今後の目標

- 定数 or polylog時間で解ける連続最適化問題の特徴付け
- 疎なデータへの対応