

計算機実験 I (L4)

疎行列の対角化・特異値分解・回帰分析

藤堂眞治

wistaria@phys.s.u-tokyo.ac.jp

2017/06/21

- 1 疎行列に対する反復法
- 2 特異値分解
- 3 最小二乗法による回帰分析
- 4 今後について

反復法

- 疎行列の場合、行列ベクトル積は高速に行える
- Givens 変換、Householder 変換などを行うと疎行列性が失われる
- 行列ベクトル積のみを用いる反復法が効果的
 - ▶ べき乗法
 - ▶ Lanczos 法

べき乗法 (Power Method)

- 適当なベクトル v_1 から出発する
- v_1 が最大固有ベクトル ξ_1 と直交していないとすると

$$v_1 = c_1 \xi_1 + c_2 \xi_2 + c_3 \xi_3 + \cdots + c_N \xi_N$$

と展開できる ($c_1 \neq 0$)。この両辺に A を次々掛けて行くと

$$v_2 = Av_1 = c_1 \lambda_1 \xi_1 + c_2 \lambda_2 \xi_2 + c_3 \lambda_3 \xi_3 + \cdots + c_N \lambda_N \xi_N$$

$$v_3 = A^2 v_1 = c_1 \lambda_1^2 \xi_1 + c_2 \lambda_2^2 \xi_2 + c_3 \lambda_3^2 \xi_3 + \cdots + c_N \lambda_N^2 \xi_N$$

$\vdots$

$$v_{n+1} = A^n v_1 = c_1 \lambda_1^n \xi_1 + c_2 \lambda_2^n \xi_2 + c_3 \lambda_3^n \xi_3 + \cdots + c_N \lambda_N^n \xi_N$$

$$= c_1 \lambda_1^n \left[\xi_1 + \sum_{k=2}^N \frac{c_k}{c_1} \left(\frac{\lambda_k}{\lambda_1} \right)^n \xi_k \right] \approx c_1 \lambda_1^n \xi_1$$

べき乗法の収束

- べき乗法による固有値

$$\frac{v_{n+1}^T v_{n+1}}{v_{n+1}^T v_n} = \lambda_1 + O\left(\left(\frac{\lambda_2}{\lambda_1}\right)^{2n}\right)$$

- 誤差の収束

$$\frac{v_{n+1}^T v_{n+1}}{v_{n+1}^T v_n} \approx \lambda_1 + e^{-2n \ln(\lambda_1/\lambda_2)}$$

- $1/\ln(\lambda_1/\lambda_2)$ 程度の反復が必要
- λ_1 と λ_2 が近い場合には、反復回数が非常に多くなる

第2固有値・第3固有値...

- 第1固有ベクトル ξ_1 の成分を行列から差し引く (減次)

$$A_1 = A - \lambda_1 \xi_1 \xi_1^T$$

この行列は、固有値 $0, \lambda_2, \lambda_3, \dots, \lambda_N$ を持つ

- 行列 A_1 に対してべき乗法を使うと、固有値 λ_2 と固有ベクトル ξ_2 が得られる
- 第 k 固有値まで求まっている場合

$$A_k = A - \sum_{i=1}^k \lambda_i \xi_i \xi_i^T$$

- 実際には数値誤差のため、ベクトルの直交性は厳密ではない
- 大きい方から数個程度を求めるのが限界

Rayleigh-Ritz の方法

- $N \times N$ 行列 A について、互いに正規直交するベクトル $v_1, v_2, \dots, v_M$ ($M < N$) が張る部分空間の中で「最良の」固有ベクトルを求めたい

- $N \times M$ 行列

$$V = (v_1 v_2 \cdots v_M)$$

を定義すると、 $V^T V = I$ が成り立つ (ただし $VV^T \neq I$)

- 部分空間内のベクトルを $w = \sum_i a_i v_i$ と表すと、 $\frac{w^T A w}{w^T w}$ が極大値を取る (本当の固有ベクトルにできるだけ平行になる) 条件は、

$$\frac{\partial}{\partial a_i} \frac{w^T A w}{w^T w} \sim \sum_j H_{ij} a_j - \lambda a_i = 0$$

Rayleigh-Ritz の方法

■ 行列

$$H_{ij} = v_i^T A_{ij} v_j$$

に対する固有値問題: $Ha = \lambda a$

■ λ : もとの行列の近似固有値 (Ritz 値)

■ Va : もとの行列の近似固有ベクトル (Ritz ベクトル)

■ 最大固有値に対する良い近似固有値が欲しい場合 $\Rightarrow$ $v_1, v_2, \dots, v_M$ を最大固有ベクトルになるべく近い (しかし、 互いに直交する) ベクトルに選べば良い

Lanczos 法

- 初期 (ランダム) ベクトル v_1 に加えて

$$Av_1, Av_2, \dots, A^{M-1}v_1$$

を正規直交化して $v_1, v_2, \dots, v_M$ を作る (Krylov 部分空間)

- 部分空間での Ritz 値を固有値の近似値とする
- $A^n v_1$ はどんどん最大固有ベクトルに近づいていくので、
 $M \ll N$ でも良い近似固有値が得られると期待される
- Lanczos 法 (Arnoldi 法)

Lanczos 法

- 正規化された初期 (ランダム) ベクトル v_1 から出発する
- $v_2, v_3, \dots$ を生成する

$$v_2 = (Av_1 - \alpha_1 v_1) / \beta_1$$

$$v_3 = (Av_2 - \alpha_2 v_2) / \beta_2$$

$\vdots$

ここで

$$\alpha_i = v_i^T A v_i$$

$$\beta_i = |Av_i - \beta_{i-1} v_{i-1} - \alpha_i v_i|$$

と選ぶ

Lanczos 法

- $v_1, v_2, v_3, \dots, v_{M+1}$ は正規直交
- 漸化式を書き換えると

$$Av_1 = \alpha_1 v_1 + \beta_1 v_2$$

$$Av_2 = \beta_1 v_1 + \alpha_2 v_2 + \beta_2 v_3$$

$$Av_3 = \beta_2 v_2 + \alpha_3 v_3 + \beta_3 v_4$$

$$\vdots$$

$$Av_M = \beta_{M-1} v_{M-1} + \alpha_M v_M + \beta_M v_{M+1}$$

Lanczos 法

■ 行列で表現すると

$$A(v_1 v_2 \cdots v_M) = (v_1 v_2 \cdots v_M v_{M+1}) \begin{pmatrix} \alpha_1 & \beta_1 & & & & \\ \beta_1 & \alpha_2 & \beta_2 & & & \\ & \beta_2 & \alpha_3 & \beta_3 & & \\ & & \beta_3 & \alpha_4 & \beta_4 & \\ & & & \ddots & \ddots & \ddots \\ & & & & \beta_{M-1} & \alpha_M \\ & & & & & \beta_M \end{pmatrix}$$

両辺に左から $(v_1 v_2 \cdots v_M)^T$ をかけると

$$(v_1 v_2 \cdots v_M)^T A (v_1 v_2 \cdots v_M)$$

は 3 重対角行列となることがわかる

Lanczos 法

- 原理的には、 N ステップ目で $\beta_N = 0$ となり、3 重対角化が完了する
- 実際には、数値誤差のため $v_1, v_2, v_3 \dots$ の直交性が崩れる
 - ▶ M を大きくしすぎると、おかしい固有値が出てくる
 - ▶ 全ての固有値が欲しい場合には Householder 法を使う
- Lanczos 法では、大きな固有値に対応する固有ベクトルにできるだけ近いものから部分空間を作っていく
 - ▶ 100 万次元以上の行列の場合でも $M = 100 \sim 200$ 程度で最初の数個の固有値は精度良く求まる
- 必要な操作は、行列とベクトルの積、ベクトルの内積・スケーリング・和のみ
 - ▶ 疎行列の場合、非常に効率が良い

一般の非正方行列の場合

- 特異値分解 (SVD: Singular Value Decomposition)
- 任意の $m \times n$ 実行列 A は

$$A = U\Lambda V^T$$

の形に (一意に) 分解できる

- U : ($m \times k$) 行列 (列ベクトルは互いに正規直交)
- V : ($n \times k$) 行列 (列ベクトルは互いに正規直交)
- Λ : $\text{diag}(\lambda_1, \lambda_2, \dots, \lambda_k)$ ($\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_k > 0$) 特異値
- ベクトル表示 (行列をランク 1 の行列で分解)

$$A = \sum_{\ell=1}^k \lambda_{\ell} u_{\ell} v_{\ell}^T$$

特異値分解の例

$$\begin{pmatrix} 1 & 2 & 3 \\ 6 & 4 & 5 \\ 8 & 9 & 7 \\ 10 & 11 & 12 \end{pmatrix} = \begin{pmatrix} -0.14 & -0.62 & -0.05 \\ -0.34 & 0.37 & 0.81 \\ -0.55 & 0.54 & -0.58 \\ -0.75 & -0.44 & 0.06 \end{pmatrix} \times \begin{pmatrix} 25.35 & 0 & 0 \\ 0 & 2.15 & 0 \\ 0 & 0 & 1.71 \end{pmatrix} \begin{pmatrix} -0.56 & -0.59 & -0.59 \\ 0.68 & 0.09 & -0.73 \\ 0.48 & -0.81 & 0.35 \end{pmatrix}$$

完全特異値分解 (full SVD)

$$\begin{pmatrix} 1 & 2 & 3 \\ 6 & 4 & 5 \\ 8 & 9 & 7 \\ 10 & 11 & 12 \end{pmatrix} = \begin{pmatrix} -0.14 & -0.62 & -0.05 & -0.77 \\ -0.34 & 0.37 & 0.81 & -0.29 \\ -0.55 & 0.54 & -0.58 & -0.29 \\ -0.75 & -0.44 & 0.06 & 0.48 \end{pmatrix}$$

$$\times \begin{pmatrix} 25.35 & 0 & 0 \\ 0 & 2.15 & 0 \\ 0 & 0 & 1.71 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} -0.56 & -0.59 & -0.59 \\ 0.68 & 0.09 & -0.73 \\ 0.48 & -0.81 & 0.35 \end{pmatrix}$$

- U, V が直交行列になるように、基底ベクトルを追加
- こちらを「SVD」、もともとの分解を「thin SVD」と呼ぶ場合も多い

LAPACK による特異値分解

- 倍精度実行列の特異値分解 dgesvd http://www.netlib.org/lapack/explore-html/d8/d2d/dgesvd_8f.html
- Fortran による関数宣言

```
subroutine dgesvd(character JOBU, character JOBVT,
  integer M, integer N,
  double precision, dimension(lda, *) A,
  integer LDA, double precision, dimension(*) S,
  double precision, dimension(ldu, *) U, integer LDU,
  double precision, dimension(ldvt, *) VT,
  integer LDVT,
  double precision, dimension( * ) WORK, integer LWORK,
  integer INFO )
```

固有値分解との関係

- 半正定値実対称行列の場合
「固有値分解」と「特異値分解」は等価
- 一般の (非正方) 実行列 A の場合
 - ▶ 完全 SVD $A = U\Lambda V^T$ を考えると
 - ▶ $B = A^T A = V\Lambda U^T U\Lambda V^T = V\Lambda^2 V^T$ は実対称行列
固有値: λ_ℓ^2 、固有ベクトル V
 - ▶ $C = AA^T = U\Lambda V^T V\Lambda U^T = U\Lambda^2 U^T$ も実対称行列
固有値: λ_ℓ^2 、固有ベクトル U

特異値分解が役に立つ例

- 連立一次方程式の最小二乗解
- 行列の低ランク近似
- 画像圧縮

特異な (特異に近い) 連立一次方程式

- ランク r の $m \times n$ 行列 A の full SVD を考える

$$A = (U_1 U_2) \begin{pmatrix} \Lambda_1 & 0 \\ 0 & 0 \end{pmatrix} (V_1 V_2)^T = U_1 \Lambda_1 V_1^T$$

U_1 : $m \times r$, U_2 : $m \times (m - r)$, Λ_1 : $r \times r$, V_1 : $n \times r$, V_2 : $n \times (n - r)$

- $r \neq n$ のとき、方程式 $Ax = b$ は無限個の解をもつ、あるいは解なし
- 無限個の解をもつ場合 ($U_1 U_1^T b = b$) の一般解

$$x = V_1 \Lambda_1^{-1} U_1^T b + V_2 z$$

z は $(n - r)$ 次元の任意のベクトル

一般化逆行列と最小二乗解

- V_1 と V_2 の列ベクトルは全て直交する。一般解のうちノルム $|x|^2$ が最小となるのは $z = 0$ のとき

$$x = A^\dagger b \quad A^\dagger = V \begin{pmatrix} \Lambda^{-1} & 0 \\ 0 & 0 \end{pmatrix} U^T$$

- $A^\dagger$: Moore-Penrose の一般化逆行列 (1/0 を 0 とおくことに相当)
- $Ax = b$ が解を持たない場合にも最良解を与える
- 零に非常に近い特異値がある場合も、それらを零とみなすことで数値的に安定した解がえられる

行列の低ランク近似

- ランク r ($r < k$) の行列のうち、行列 A を「最も良く近似」する $\tilde{A}$ を選ぶ
- 「最も良く近似」 = フロベニウスノルム $\|A - \tilde{A}\|_F$ を最小化

$$\|X\|_F^2 \equiv \sum_{ij} x_{ij}^2$$

- 特異値のうち大きなものから r 個とり、残りを零とした

$$\tilde{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_r, 0, \dots, 0)$$

を使い

$$\tilde{A} = U\tilde{\Lambda}V^T$$

を作れば良い (Eckart-Young の定理)

行列の低ランク近似の例

$$\begin{pmatrix} -0.14 & -0.62 & -0.05 \\ -0.34 & 0.37 & 0.81 \\ -0.55 & 0.54 & -0.58 \\ -0.75 & -0.44 & 0.06 \end{pmatrix} \begin{pmatrix} 25.35 & 0 & 0 \\ 0 & 2.15 & 0 \\ 0 & 0 & 0 \end{pmatrix} \\ \times \begin{pmatrix} -0.56 & -0.59 & -0.59 \\ 0.68 & 0.09 & -0.73 \\ 0.48 & -0.81 & 0.35 \end{pmatrix} = \begin{pmatrix} 1.04 & 1.93 & 3.03 \\ 5.33 & 5.12 & 4.52 \\ 8.47 & 8.21 & 7.34 \\ 9.95 & 11.1 & 12.0 \end{pmatrix}$$

- それなりに悪くない近似が得られる
- 誤差 (フロベニウスノルム) = 1.71 (無視した特異値の二乗和の平方根)

Eckart-Young の定理の証明

- A を近似する行列 X (ランク $\leq r$) を考えると

$$\|A - X\|_F^2 = \sum_{ij} (a_{ij} - x_{ij})^2 = \text{tr}[(A - X)(A - X)^T]$$

を最小化すればよい。 $UU^T = VV^T = I_k$ より

$$\begin{aligned} \|A - X\|_F^2 &= \text{tr}[UU^T(A - X)VV^T(A - X)^T] \\ &= \text{tr}[U^T(A - X)V(U^T(A - X)V)^T] \\ &= \text{tr}[(\Lambda - G)(\Lambda - G)^T] \quad (G \equiv U^T X V) \\ &= \sum_i^k (\lambda_i - g_{ii})^2 + \sum_i \sum_{j \neq i} g_{ij}^2 \end{aligned}$$

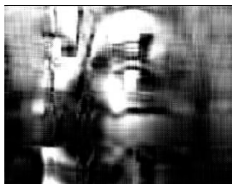
ランク k 以下で、 $\|A - X\|_F^2$ を最小化するには、 $g_{ii} = \lambda_i$ ($i = 1 \cdots r$)、それ以外は全て零とすればよい。

特異値分解による画像圧縮

$k = 3$



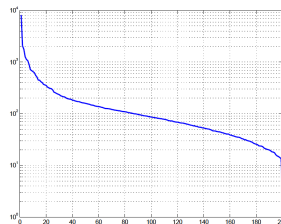
$k = 10$



$k = 20$



$k = 40$



<http://www.na.scitec.kobe-u.ac.jp/~yamamoto/lectures/cse-introduction2009/cse-introduction2009.html>

最小二乗法によるフィッティング

- 説明変数 (例: 電圧): $x_1, x_2, x_3, \dots$,
- 観測値 (例: 電流): $y_1, y_2, y_3, \dots$,
- 単回帰モデル: $y = a + bx + \epsilon$ (ϵ : ノイズ)
- 未知母数: a, b
- 最小二乗法: 残差 $R(a, b) = \sum_i (y_i - (a + bx_i))^2$ を最小化

$$\frac{\partial R}{\partial a} = -2 \sum_i (y_i - (a + bx_i)) = 0$$

$$\frac{\partial R}{\partial b} = -2 \sum_i (y_i - (a + bx_i))x_i = 0$$

回帰分析の一般化

- 基底関数: $\phi_j(x)$ ($j = 1 \cdots M$)
- モデル: $y(x) = \sum_j \phi_j(x) w_j + \epsilon$
- 残差: $R(\mathbf{w}) = \sum_i (y_i - \sum_j \phi_j(x_i) w_j)^2$

$$\frac{\partial R}{\partial w_k} = -2 \sum_i (y_i - \sum_j \phi_j(x_i) w_j) \phi_k(x_i) = 0$$

- 計画行列 (design matrix) $\Phi_{ij} = \phi_j(x_i)$ を導入すると

$$R(\mathbf{w}) = \|\mathbf{y} - \Phi \mathbf{w}\|^2$$

- 最小二乗解: $\Phi^t \Phi \mathbf{w} = \Phi^t \mathbf{y} \Rightarrow \mathbf{w} = (\Phi^t \Phi)^{-1} \Phi^t \mathbf{y}$

リッジ回帰 (Ridge Regression)

- 基底関数の数 (例: 多項式の次数) を増やしすぎると過学習 (over-fitting) が生じる
- 正則化最小二乗法 (λ は非負の定数)

$$\begin{aligned} R(\mathbf{w}) &= \sum_i (y_i - \sum_j \phi_j(x_i) w_j)^2 + \lambda \sum_j w_j^2 \\ &= |\mathbf{y} - \Phi \mathbf{w}|^2 + \lambda \mathbf{w}^t \mathbf{w} \end{aligned}$$

- 最小二乗解

$$(\Phi^t \Phi + \lambda I) \mathbf{w} = \Phi^t \mathbf{y} \Rightarrow \mathbf{w} = (\Phi^t \Phi + \lambda I)^{-1} \Phi^t \mathbf{y}$$

計算機環境

- 教育用計算機システム
- photon クラスタ
 - ▶ 卒業まで利用可 (希望すれば大学院でも)
 - ▶ バッチシステムを使えば、かなり大規模な計算も可能
- 計算機利用・シミュレーションに関する質問は今後も歓迎
`computer@exa.phys.s.u-tokyo.ac.jp`
- バージョン管理システムに関する講習を企画中

計算機実験 II

- 3 年冬学期、金曜 5 限
- 全 7 回程度
- 計算機実験 I で身に付けた知識をもとに、より高度な数値計算手法・アルゴリズムを学び、物理学における具体的な問題への応用を通して実践的な知識と経験を身につける。
 - ▶ 数値対角化と量子力学 (講義と実習)
 - ▶ モンテカルロ法と統計物理 (講義と実習)
 - ▶ 最適化と実験データ解析 (講義と実習)
 - ▶ スパコンと並列計算 (講義と実習)
- 実習はグループに分かれて課題に取り組む。