線形代数学入門で学んでいない データサイエンスのための線形代数

鈴木治郎 [*1]

[*1] 信州大学 全学教育機構

[*1]szkjiro@shinshu-u.ac.jp

キーワード　線形代数，数値計算，次元の呪い

1　はじめに

人工知能技術を含めたデータサイエンスの学習の機運が高まっている中，基礎知識としての数学，とくに確率統計，微分積分学，線形代数学を学必要性も高まっている．ところが線形代数学に焦点をあてたとき，現在ではコンピュータを電卓代わりに利用すれば済むような，行列とベクトルの利用法レベルで止まっているものが多いように思われる．しかし，それではデータサイエンスの適用結果を線形代数学の知識に基づいて解釈するには不十分である．

そこで本稿では，いわゆる線形代数学入門レベルで何が学べるかを概観し，その上で，データサイエンスに関する各種プログラミングツールを活用する上で必要となる数値計算における性質を説明し，最後に分析結果を解釈することへの結びつけをいくつか示唆したいと考える．

2　線形代数学入門の概観

線形代数学入門をひとことで言えば，適当な抽象的線形空間において線形写像を定義し，その写像の特徴量である固有値を用いて写像の核となる空間を記述することである．本稿ではすべて実数上を仮定して，少し記号を導入する．ここで言う行列が何を表すかなど，いくつかの例は後に取り上げる．

$\mathbf{x} \in \mathbb{R}^n$ を n 次元実数ベクトル，行列 A は線型写像 $\mathbb{R}^n \to \mathbb{R}^n$ とする．線形写像 A には後に述べるいくつかの制限をつければ，固有値 $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ を行列 A から計算できて，各 λ_i に対して定まる固有ベクトル $\mathbf{v}_i \in \mathbb{R}^n$ により式

$$A\mathbf{x} = \mathbf{0}$$

の解は

$$\mathbf{x} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i, \qquad \alpha_i \in \mathbb{R}$$

と表すことができる．ここで与えた線形写像への制限は，後に述べるようにデータサイエンス目的ではよくある設定である．

線形代数学入門では，上の性質を目標としていくつかの性質や方法

- 線形写像を行列 A で表現する方法
- 行列 A をユニタリー行列（行列式が ± 1 である行列）による変換（いわゆる基本変形）で対角行列 $diag(\lambda_1, \lambda_2, \ldots, \lambda_n)$ の形に変換する方法（ユニタリー変換）
- 固有多項式を解くことで固有値を求める方法
- 行列式と固有値の関係 $\det A = \prod_{i=1}^{n} \lambda_i$ が成り立つ
- 核空間の次元，つまり階数（ランク）に関する構造定理を与える
- 核空間の幾何学的解釈

などを学ぶことになる．

3　幾何学的解釈による行列入門

線形方程式

$$\alpha x + \beta y + \gamma z = 0, \qquad \alpha, \beta, \gamma \in \mathbb{R}$$

は実 3 次元空間における平面を表す．ここで

$$A_1 = (\alpha_1, \beta_1, \gamma_1), \qquad \mathbf{x} = (x, y, z)^t$$

とおく．記号 ^{t} はベクトルに対する転置を表す．このとき上の方程式は

$$A_1\mathbf{x} = \mathbf{0}$$

と表せる．このベクトル積のことを内積という．ベクトル A_1 のことは係数行列ともいう．もう一つの係数ベクトル $A_2 = (\alpha_2, \beta_2, \gamma_2)$ を用いて方程式

$$A_2\mathbf{x} = \mathbf{0}$$

を考えれば，上の 2 つの平面は，一直線で交わるか，あるいは同一である．この係数行列を積み上げて

$$\begin{pmatrix} A_1 \\ A_2 \end{pmatrix} \mathbf{x} = \mathbf{0}$$

と表すとき，係数ベクトルを積み上げて並べたものを行列という．ユニタリー行列による変換で各行列が移り合うときに，それらは同値であると言い，記号 $\sim$ で表すことにすれば

$$\begin{pmatrix} A_1 \\ A_2 \end{pmatrix} \sim \begin{pmatrix} A_1 \\ \mathbf{0} \end{pmatrix}, \qquad \not\sim \begin{pmatrix} A_1 \\ \mathbf{0} \end{pmatrix}$$

のいずれかである．このように，すべて成分が 0 のベクトルを適当に付加することにより，対象とする行列はすべて同じ行数をもつと仮定すると扱いをしやすい．行数ともとのベクトルの成分数とが等しい行列のことを正方行列という．現在の仮定のもとでは，正方行列は対角成分のみが 0 でない行列に同値になることがわかっているので，実数 $\lambda_1 \geq \cdots \geq \lambda_r > \lambda_{r+1} = 0, \ldots, \lambda_n = 0$ により

$$A \sim \begin{pmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ 0 & \vdots & \ddots & \vdots & 0 \\ 0 & \cdots & 0 & 0 & \lambda_n \end{pmatrix}$$

と表されるとしてよい．階数（ランク）は 0 でない固有値の個数 r である．

4 数値計算からの視点

4.1 逆行列

誤差を伴う成分からなる正方行列において，行列式が 0 であることを確かめることは事実上不可能である．応用における解釈では，小さな固有値を無視して，大きな固有値に対する固有ベクトルの張る空間を調べることが多く（低次元化の一種），行列式自体が 0 になることを問題にすることはあまりない．

またビッグデータにおいては 0 の成分がたいへん多い（スパース（疎）な行列）が，それらの逆行列を直接に求めるとスパースではない行列になることも多く逆行列を伴わない統計モデリングをすべきである [1].

4.2 固有値

行列 A は，単位行列 I を用いて，その固有値 λ との関係

$$(A - \lambda I)\mathbf{x} = \mathbf{0}$$

をもつ．固有値を求める方法は変数 X に関する固有方程式

$$\det(A - XI) = 0$$

を解くことと，線形代数学入門の教科書（以下，「入門」と略す）では説明されることが多い．コンピュータ利用の数値計算上は，ユニタリー行列による変換で与える方法がふつうである．

データサイエンスの世界では係数行列は誤差を伴う数値であることがふつうである．係数行列は近いのに解が大きく異なる簡単な例をあげる [2].

$$\begin{cases} 3x + 4y & = 7 \\ 3x + 4.00001y & = 7.00001 \end{cases}, \qquad (x, y) = (1, 1)$$

と

$$\begin{cases} 3x + 4y & = 7 \\ 3x + 3.99999y & = 7.00004 \end{cases}, \qquad (x, y) = (7{+}2/3, -4)$$

とは，係数は近くても解は大きく異なっている．固有値の場合，

$$\text{固有方程式の解は近い} \quad \Rightarrow \quad \text{係数は近い}$$

は成り立つが，必要十分条件ではない．このため誤差を伴うデータから直接に方程式を解く方法は好ましくない場合がある．

実際の固有値の計算方法はユニタリー変換による方法，あるいは誤差を前提に繰り返し計算による収束判定を伴うもの（共役勾配法など）が用いられている．Python などのライブラリを直接使う場合には，目的に合ったものを選びたい．

5 データサイエンスで扱う行列

5.1 共分散行列

複数のベクトル同士の共分散から得られる共分散行列は，共分散の性質から対称行列である．さらに共分散行列 Σ の，その転置 Σ^t との積 $\Sigma^t\Sigma$ は正定値行列であり，

固有値は Σ の固有値 λ_i に対して λ_i^2 である．このため共分散行列の性質は正定値行列の性質を用いて確かめられる特徴をもつ．

5.2 ネットワーク行列

たとえば節点 1,2,3 をもつ行列で 1 と 2，1 と 3 は辺で結ばれるが 2 と 3 は結ばれていないとする．この状況は行列を用いて

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

と表せる（自分自身，つまり対角成分も 1 にする場合がある）．節点の番号が各行・各列の番号に対応している．各成分が 0 または 1 であることから，利用する計算機資源に「1 ビット計算機」など，通常の数値計算とは異なったものが用いられることも多い．

5.3 一般化逆行列

逆行列をもたない行列（正則でない行列）に対する理論構築には，特異値分解などがあるが，入門の自然な延長として，これらの手法の利用をデータサイエンスのために書かれているものは多くない [3].

5.4 次元の呪い

ビッグデータにおいては，データとなる各ベクトルの成分数もたいへん大きくなるが，今日の高速コンピュータによっても，従来の計算アルゴリズムで単純に処理できるものでないことが多い．このため多くのデータ処理において，次元を減らす試み（数百分の 1 など）は多くなされてきている．しかし，そうした低次元化においては，適用条件を踏まえて考える必要がある [4].

6 さいごに

以上，簡単ではあるが線形代数学入門の先に学ぶべきことをいくつか触れてきた．データサイエンスを活用する上で，こうした数学的知識を背景に，分析結果の解釈を行おうという契機に本稿がなれば幸いである．

参考文献

[1] 青嶋誠他,『スパース推定法による統計モデリング』，共立出版，2019 年
[2] 新谷尚義,『基礎数学シリーズ 16　数値計算 I』，朝倉書店, 1967 年
[3] 清水昌平編,『データサイエンスのための数学』，講談社，1967 年
[4] 青嶋誠他,『高次元の統計学』，共立出版，2019 年