行列の固有値問題

桂田 祐史

2002 年 7 月 16 日, 2006 年 5 月 21 日

大幅な書き直しをしたい…が暇がない。困った。困った。
この文書は

http://nalab.mind.meiji.ac.jp/~mk/labo/text/

で読めます。
最近

「固有値問題ノートの補足」
http://nalab.mind.meiji.ac.jp/~mk/labo/text/eigenvalues-add.pdf

というのも書き出しました。
多分骨組みはもう一度 0 から考えて、それに上の二つの文書から必要なものを抜き出して貼り付けていくのでしょう。

目 次

1 一般的な注意

- 次元の低い場合を除いて直接法はない (固有方程式と同値で、それは代数方程式なので、四則と巾根では解けない) 反復法になる。
- 対称な問題と非対称な問題の差は大きい
- 中継地点 (三重対角行列、Hessenberg 形) を経由すべし

1.1 問題の定式化

いま A を N 次正方行列とする。この時

$$Ax = \lambda x, \quad x \neq 0$$

を満たす $\lambda \in \mathbf{C}$ を A の**固有値** (eigenvalue), $x \in \mathbf{C}^N \setminus \{0\}$ を A の固有値 λ に属する**固有ベクトル** (eigenvector) と呼ぶ。

固有値問題 — 固有値、固有ベクトルを求める問題 — は非線形問題であり、有限回の四則演算では解けない ($N \geq 5$ のときは巾根を求める操作を用いても解けない)。これを解くには、何らかの意味での反復法が必要である。

これに類似した問題に、**一般化固有値問題**と**特異値問題**がある。ここでは名前をあげておくだけにとどめるが、将来問題に出会った時に、固有値問題の親戚であると気がつけば良い。いずれの問題も、固有値問題のアルゴリズムを修正したもので解くことが出来る。

一般化固有値問題　行列 A, B が与えられた時、方程式

$$Ax = \lambda Bx, \quad x \neq 0$$

を満たすスカラー λ, ベクトル x を求める問題を一般化固有値問題という。これは 2 次形式の固有値問題などに関係して現れる。$B = I$ (単位行列) の場合は通常の固有値問題になる。応用上は B は多くの場合、正定値対称行列になる。B が正則である場合、上の方程式に B^{-1} をかけて $A' = B^{-1}A$ とおくと

$$A'x = \lambda x, \quad x \neq 0$$

となって、形式的には通常の固有値問題に帰着するが、このやり方は大抵の場合、得策ではない (A, B が対称であっても、A' が対称であるとは限らなくなるなどの理由がある)。簡単なことは次の文書に書いておいた。

> 『一般化固有値問題メモ』
> http://nalab.mind.meiji.ac.jp/~mk/labo/text/generalized-eigenvalue-problem.pdf

特異値問題　一言でいうと正方行列でないような行列に対して固有値問題を一般化したものである。

1.2 線形代数の復習

線形代数学で学んだことをいくつか復習しておこう。

定理 1.1 固有値は、固有方程式と呼ばれる代数方程式

$$\det(\lambda I - A) = 0$$

の根である。従って N 次正方行列の固有値は重複度を込めて数えると N 個ある。

逆に任意の代数方程式に対して、それを固有方程式にもつ行列が存在するので、数学的には「固有値問題は代数方程式の問題と同値である」。

定理 1.2 (i) Hermite 行列の固有値は実数であり、ユニタリ行列で対角化できる。

(ii) 実対称行列の固有値は実数であり、直交行列で対角化できる。

1.3 心構え (どう立ち向かうべきか)

- 固有値を求めるのに、固有方程式を解こうとするのは得策ではない[1]。固有値問題を解くには、以下に説明する固有値問題用の解法を採用すべきである。
- 実際的な観点からは、固有値問題の解法に使用される各種方法の原理を理解し、自分が解こうとしている問題にあった方法 or プログラムを選択できるようになることを目指すのが良い。
- 解こうとしている問題については、以下のことに留意して考えよう。

  (i) 問題は対称 (=行列が実対称行列または Hermite 行列) かどうか。

  (ii) すべての固有値が欲しいのかどうか。

  (iii) 固有ベクトルは必要かどうか。

  (ii), (iii) は要するに「ムダはやめよう」ということである。
- 対称な問題は非対称な問題と比べて解きやすい。

  「対称な問題を解くのはサイエンスであるが、非対称な問題を解くのはアートである。」

  という言葉があるくらいである。

2 解法についての概観

歴史的なことを述べると、以前は対称な問題は Jacobi 法と呼ばれるアルゴリズムで解かれるのが普通であった。しかし Jacobi 法は N が小さな (せいぜい数十) である場合には実用的であるが、N の大きな問題を解くのに採用するのは得策ではない。今では行列の三重対角化や Hessenberg 形への変換を利用した解法が主流である（対称行列の場合は三重対角化、非対称行列の場合は Hessenberg 行列への変換を行なうことになる）。行列を三重対角化 (resp. Hessenberg 化) するとは、相似変換を施して、行列を三重対角行列 (resp. Hessenberg 行列) に変換することをいう。この変換の方法は色々あるが、いずれも $O(N^3)$ の演算量で済む。後で述べるように相似変換で固有値は不変なので、変換後の「簡単な」行列の固有値を求めれ

[1] そもそも次数 n が 4 以下でないと根の公式はないし、代数方程式のための数値解法で解ける問題の範囲は、固有値問題のための数値解法で解ける範囲よりも狭い。

ば、元の行列の固有値が求まったことになる。三重対角行列や Hessenberg 行列に対する固有値問題は、元の行列よりも簡単に解ける、というのが基本的なアイディアである。

色々な細かな技法 (部品) を、利用者の要求に応じて組み合わせることで、望ましい解法が出来上がる。以下、それらの原理を説明する。三重対角化や Hessenberg 化のためのアルゴリズムの解説は次節にまわして、ここでは実際に固有値を求める諸方法を解説する。

2.1 相似変換

固有値問題でもっとも基本的な方法は**相似変換**である。これは行列 A を正則行列 P によって、$P^{-1}AP$ に変換することを意味する。

定理 2.1 相似変換により固有値は不変である。

に注意しよう。A が実対称な場合には P として実直交行列が使われる。この場合 $P^{-1} = {}^tP$ であり、計算が簡単になることの他に、様々な利点がある。

実対称行列の、実直交行列による相似変換は実対称行列である:

$$(P^TAP)^T = P^TA^T(P^T)^T = P^TAP.$$

実直交行列による変換として、次の三つが重要である。

(1) **Givens 変換** — 2 次元平面における回転

(2) **鏡映変換** (Householder 変換) — 超平面に関する対称移動

(3) **QR 変換**

2.2 Jacobi 法 — 温故知新

1960 年代まで主流であった **Jacobi 法**について簡単に説明しよう。これは実対称行列を 2 次元の回転変換により変形していって、対角成分以外の成分の絶対値を小さくしていくというものであり、行列のサイズ N が 10 程度の小さなものであれば現在でも実用的である。

なぜ「対角成分以外の成分の絶対値を小さくしていく」のか？これについて説明しよう。まず次の定理は簡単であるが重要である。

定理 2.2 対角行列の固有値は対角成分である。

一般の行列の場合も次の定理が成り立つ。おおざっぱにまとめると「行列の固有値は対角成分に近く、そのずれは非対角成分の大きさによる」。

定理 2.3 (Gerschgorin の円板定理 (1931)) $A = (a_{ij})$ に対して

$$\Delta_i = \left\{ z \in \mathbf{C}; |z - a_{ii}| \le \sum_{1 \le j \le N, j \ne i} |a_{ij}| \right\} \quad (i = 1, 2, \cdots, N)$$

とおくと、

$$\sigma(A) \equiv A \text{ の固有値全体} \subset \bigcup_{i=1}^{n} \Delta_i.$$

もし Δ_i のうちで k 個が連結成分をなせば、その中に k 個の固有値がある。

証明 $Ax = \lambda x$, $x = (x_1, \ldots, x_N) \ne 0$ とする。今 x の絶対値最大の成分を x_k とする:

$$\max_{i=1,2,\ldots,N} |x_i| = |x_k|.$$

方程式 $Ax = \lambda x$ の第 k 成分を書くと

$$\sum_{j=1}^{N} a_{kj} x_j = \lambda x_k.$$

これから

$$(\lambda - a_{kk}) x_k = \sum_{1 \le j \le N, j \ne k} a_{kj} x_j.$$

(以下 $1 \le j \le N, j \ne k$ を単に $j \ne k$ と書くことにする。) よって

$$|\lambda - a_{kk}| \le \sum_{j \ne k} |a_{kj}| \left| \frac{x_j}{x_k} \right| \le \sum_{j \ne k} |a_{kj}|,$$

したがって $\lambda \in \Delta_k \subset \bigcup_{i=1}^{N} \Delta_i$.

つぎに

$$D = \mathrm{diag}(a_{11}, \cdots, a_{NN}), \quad A_t = (1-t)D + tA \quad (t \in [0,1])$$

とおくと、A_t に対する円板は

$$\Delta_i(t) = \{ z \in \mathbf{C}; |z - a_{ii}| \le t \sum_{j \ne i} |a_{ij}| \}$$

である。A_i の固有値は t について連続的に変化し、$t = 0$ のとき $\Delta_i(0) = \{a_{ii}\}$ にそれぞれ重複個数だけある。t を増加させれば、いくつかの円板が拡大して合流するが、k 個の $\Delta_i(t)$ が合流すれば、その中に A_t の k 個の固有値が存在する。ゆえに $t = 1$ に達したとき、Δ_i の k 個が連結成分をなせば、その中に $A_1 = A$ の k 個の固有値が存在する。 ■

系 2.4 (狭義優対角行列は正則) N 次正方行列 $A=(a_{ij})$ について、

$$|a_{ii}| > \sum_{1\le j\le N, j\ne i} |a_{ij}| \quad (i=1,2,\cdots,N)$$

が成り立てば、A は正則である。

証明 $0 \notin \bigcup_{i=1}^{N} \Delta_i$ より、0 は A の固有値ではない。ゆえに A は正則である。■

なぜか今のカリキュラムの盲点となってしまって、紹介されることがないが、「代数方程式の根は方程式の係数の連続関数である」行列の固有方程式の係数は、行列の成分の連続関数であることは明らかだから、「行列の固有値は行列の成分の連続関数である」。Gerschgorin の定理はあざやかに連続性を見せてくれる。

2.3 巾乗法 (power method)

ここでも問題は対称であると仮定する。

与えられた行列の絶対値最大の固有値を求めるための、累乗法あるいは巾乗法[2]を説明する。

行列 A の固有値 $\{\lambda_i; i=1,2,\cdots,n\}$ は絶対値の順に番号づけられているとする:

$$|\lambda_1| \ge |\lambda_2| \ge \cdots |\lambda_n|.$$

また、$\{u_i; i=1,2,\cdots,n\}$ は $\{\lambda_i\}$ に対応する A の固有ベクトルからなる正規直交基底とする。

ここで話を簡単にするために次の仮定をおく。

$$\text{仮定:} \quad |\lambda_1| > |\lambda_2|.$$

この時、適当な x_0 ($x_0 \notin \mathrm{Span}\{u_2, u_3, \cdots, u_n\}$) を選んで

$$\begin{cases} y_{k+1} & := & Ax_k \\ x_{k+1} & := & y_{k+1}/\|y_{k+1}\| \end{cases}$$

によりベクトル列 $\{x_k; k=0,1,\cdots\}$ を定めると、

$$\lim_{k\to\infty} x_k = \pm u_1$$

となる。実際には十分大きな番号 k を取れば、$x_k = \pm u_1$ とみなしてよい。

大雑把な説明 まず適当な $\{c_i\}$ を選ぶことにより

$$x_0 = \sum_{i=1}^{n} c_i u_i$$

[2]「冪乗法」あるいは「羃乗法」と書くのが正しい?

と展開されることに注意する。ここで $c_1 \neq 0$ である。

$$A^k x_0 = \sum_{i=1}^n c_i A^k u_i = \sum_{i=1}^n c_i \lambda_i^k u_i = c_1 \lambda_1^k \left\{ u_1 + \sum_{i=2}^n \left(\frac{\lambda_i}{\lambda_1} \right)^k c_i u_i \right\}.$$

ここで $x_k = A^k x_0 / \|A^k x_0\|$ であることに注意すると、$x_k \to \pm u_1$ $(k \to \infty)$ となることが分かる。

2.4 固有ベクトルから固有値を求める方法

誤差がなければ、$Ax = \lambda x$ の適当な ($x_i \neq 0$ となる i に対する) 成分に対応する方程式

$$\sum_{j=1}^n a_{ij} x_j = \lambda x_i$$

の両辺を x_i で割れば λ が求まるが、x が近似的な固有ベクトルでしかない場合には、以下に解説する Rayleigh[3] 商を用いる方法の方が良い。

定義 2.5 (Rayleigh 商) N 次正方行列 A と、$x \in \mathbf{C}^N \setminus \{0\}$ に対して

$$\frac{(Ax, x)}{(x, x)}$$

を A の x に対する **Rayleigh 商**という。

補題 2.6 N 次正方行列 A と $x \in \mathbf{C}^N \setminus \{0\}$ が与えられたとき、$\|Ax - \lambda x\|$ を最小にする λ は A の x に対する Rayleigh 商である。

証明

$$\begin{aligned}
\|Ax - \lambda x\|^2 &= (Ax, Ax) - \lambda (Ax, x) - \overline{\lambda}(x, Ax) + |\lambda|^2 \|x\|^2 \\
&= \|x\|^2 \left[\lambda - \frac{(x, Ax)}{\|x\|^2} \right] \left[\overline{\lambda} - \frac{\overline{(x, Ax)}}{\|x\|^2} \right] + \|Ax\|^2 - \frac{|(x, Ax)|^2}{\|x\|^2} \\
&= \|x\|^2 \left| \lambda - \frac{(x, Ax)}{\|x\|^2} \right|^2 + \|Ax\|^2 - \frac{|(x, Ax)|^2}{\|x\|^2}. \blacksquare
\end{aligned}$$

x を A の近似固有ベクトルとする時、Rayleigh 商

$$\lambda' = \frac{(Ax, x)}{(x, x)}$$

は x に対応する固有値の良い近似になる。粗く言って

$$\text{固有値の誤差} = O(\text{固有ベクトルの誤差}^2)$$

[3] Lord Rayleigh — アルゴンの発見などの業績のある物理学者。

が成り立つ。詳しくは次の命題を見よ。

命題 2.7 A が正規行列 $(AA^* = A^*A)$ で、その固有値を $\lambda_1, \cdots, \lambda_N$ とし、それに対応する固有ベクトルからなる正規直交基底を $x_1, x_2, \cdots, x_N$ とする。いま単位ベクトル v の Rayleigh 商 $\mu = v^*Av$ が

$$\|Av - \mu v\| = \varepsilon, \quad |\lambda_j - \mu| \geq \delta > \varepsilon \quad (j = 2, \ldots, N)$$

を満たせば、次の意味で μ は λ_1 にごく近い:

$$|\lambda_1 - \mu| \leq \frac{\varepsilon^2}{\delta(1 - \varepsilon^2/\delta^2)}.$$

さらに v は次の意味で x_1 に近い: $|c| = 1$ である定数 c に対して

$$\|cv - x_1\|^2 \leq (\varepsilon/\delta)^2 + (\varepsilon/\delta)^4.$$

証明　省略。一松 [2] あるいは [3] を見よ。

2.5　逆反復法

絶対値が最小の固有値 (上の記号で λ_n) を求める方法である。A^{-1} の固有値は $\lambda_1^{-1}, \cdots, \lambda_n^{-1}$ であり、絶対値最大のものは λ_n^{-1} となるから、A^{-1} に巾乗法を適用することにより、λ_n^{-1} が求まり、その逆数を取れば λ_n が得られる。

計算上の注意:　反復の各段階で

$$y_{k+1} := A^{-1}x_k$$

という計算が必要になるが、これは

$$Ay_{k+1} = x_k$$

という y_{k+1} に関する連立一次方程式を解くことにより実行する (いつものことであるが、A^{-1} を計算するのは馬鹿馬鹿しい)。

2.6　シフト法

既に得られている近似固有値の精度を改良したい、あるいは、ある指定した値に最も近い固有値を求める方法である。

行列 A の固有値 λ_i に対する近似固有値 λ_i' が分かっているとしよう。この時、

$$A' = A - \lambda_i' I$$

の固有値は

$$\lambda_1 - \lambda_i', \lambda_2 - \lambda_i', \cdots, \lambda_n - \lambda_i'$$

となる。λ_i' は λ_i の近似値ということで、絶対値が最小なのは $\lambda_i - \lambda_i'$ であると期待できる。よって、A' に対して逆反復法を適用すれば、この値 (それを $\Delta\lambda$ とおこう) が高精度に計算できる。こうして

$$\lambda_i := \lambda_i' + \Delta\lambda$$

により λ_i が求まる。

2.7 Wielandt の減次

行列 $A \in \mathbf{C}^{n\times n}$ の固有値 λ_i $(i = 1, 2, \cdots, n)$ について、

$$|\lambda_1| > |\lambda_2| > |\lambda_3| \geq \cdots \geq |\lambda_n|$$

となっていると仮定する。

2.7.1 対称の場合

最初に簡単のため、A が対称 $A^* = A$ の場合を考える。冪乗法により、λ_1 と対応する固有ベクトル z_1 を求めたとする。λ_i に属する固有ベクトルを z_i として $(i = 2, 3, \cdots, n)$、$z_1, \cdots, z_n$ が正規直交系 $(z_i^* z_j = \delta_{ij})$ であるとする (ただし $i \geq 2$ について、具体的に求めるわけではない)。このとき、

$$B = A - \lambda_1 z_1 z_1^*$$

とおくと、これも対称で、

$$Bz_1 = Az_1 - \lambda_1 z_1 z_1^* z_1 = \lambda_1 z_1 - \lambda_1 z_1 \cdot 1 = 0,$$

$$Bz_i = Az_i - \lambda_1 z_1 z_1^* z_i = \lambda_i z_i - \lambda_1 z_1 \cdot 0 = \lambda_i z_i \quad (i = 2, 3, \cdots, n).$$

ゆえに B について冪乗法を適用すると、λ_2 が得られるはずである。

B と任意に与えられたベクトル v との積は

$$Bv = Av - \lambda_1 z_1 (z_1^* v)$$

と計算できるので、B の成分を求めておく必要はないことに注意しよう。

2.7.2 非対称の場合

A が対角化可能としておく (これは余計な仮定か?)。つまり、λ_i に属する固有ベクトル z_i が存在し、$z_1, \cdots, z_n$ が基底となると仮定する。

よく知られているように、$\overline{\lambda_1}$ は、A の Hermite 共役 A^* の固有値になるが、その固有ベクトル w_1 をシフト法で求めておく (もちろん A が「対称」ならば $\lambda_1 = \lambda_1^*$ で、$w_1 = z_1$ と取れるので、対称の場合にやった操作の一般化になっている)。

必要ならば定数倍することによって、$w_1^* z_i = \delta_{i1}$ となるようにできる。実際、$A^* w_1 = \overline{\lambda}_1 w_1$ に注意して、

$$w_1^* z_i = (z_i, w_1) = (z_i, \overline{\lambda}_1{}^{-1} A^* w_1) = \lambda_1{}^{-1}(Az_i, w_1) = \lambda_1{}^{-1}\lambda_i(z_i, w_1)$$

であるから、

$$\left(1 - \frac{\lambda_i}{\lambda_1}\right) w_1^* z_i = 0.$$

ゆえに $\lambda_i \neq \lambda_1$ ならば

$$w_1^* z_i = 0.$$

一方 $w_1^* z_1 \neq 0$ である。実際、w_1 を $w_1 = c_1 z_1 + \cdots + c_n z_n$ と展開すると (ここで $\{z_i\}$ が基底であることを使っている)、直交性 $w_1^* z_i = 0$ $(i = 2, 3, \cdots, n)$ によって、$0 \neq w_1^* w_1 = c_1 w_1^* z_1$ となるから。

そこで $B = A - \lambda_1 z_1 w_1^*$ とおくと、

$$Bz_1 = Az_1 - \lambda_1 z_1 w_1^* z_1 = \lambda_1 z_1 - \lambda_1 z_1 = 0$$

また

$$Bz_i = Az_i - \lambda_1 z_1 w_1^* z_i = \lambda_i z_i - \lambda_1 z_1 0 = \lambda_i z_i.$$

ゆえに B について冪乗法を適用すると、λ_2 が得られる。

疑問点　A が対角化可能でない場合はどうなるだろう。$w_1^* z_1 \neq 0$ を保証するためだけにこの仮定をおくのは大げさのように思う。A の固有ベクトルが n 個存在しない場合には、B の固有ベクトルとして何が出て来るかすぐには分からないということはある。

3　三重対角化の手法

実対称行列に対する三重対角化の手法を学ぶ。

(1) Givens 法

(2) Householder 法

(3) Lanczos 法

与えられた実対称行列 $A = (a_{ij})$ に対し、適当な正則行列 P を求めて

$$P^{-1}AP = \text{三重対角}$$

とする。特に P として直交行列を取る。

歴史的には、Jacobi 法の変形として Givens 法が最初に現われたが、その後現われた Householder 法は演算回数が約半分となるなど利点が多く、普及している。大型疎行列に対しては Lanczos 法が有力な方法とされている。

3.1 Householder 法

u を $\mathbf{R}^n$ の単位ベクトルとする。u に直交する超平面

$$W_u \equiv \{x \in \mathbf{R}^n; (x,u) = 0\}$$

に関する対称移動を表す変換を U とすると、

$$U = I - 2uu^T$$

である。($\overrightarrow{\mathrm{OP}} = x$ なる点 P から W_u に下ろした垂線の足を Q とすると、$\overrightarrow{\mathrm{QP}} = (x,u)u$. それゆえ、 $Ux = x - 2\vec{QP} = x - 2(x,u)u = (I - 2uu^T)x$. ゆえに $U = I - 2uu^T$.)

定義 3.1 (鏡映変換、Householder 行列) $\mathbf{R}^n$ の単位ベクトル u に対し、行列 $U = I - 2uu^T$ で定まる線形変換をベクトル u に対する**鏡映変換**、U を **Householder 行列** (または基本直交行列) と呼ぶ。

補題 3.2 (鏡映変換の性質) U を Householder 変換とするとき、

(i) U は直交変換である。

(ii) U は対称変換である。

証明

(i) $UU^T = I$ を計算で確かめてもよいし、U が長さを変えないことから明らかである。

(ii) これは $U = I - 2uu^T$ であることから分かる。 ■

参考　N 次元の任意の直交変換は、N 個以下のベクトルに関する鏡映変換の積に書ける (Cartan)。

補題 3.3 $\|x\| = \|y\|$, $x \neq y$ なる $x, y \in \mathbf{R}^n$ に対して、

$$u = \frac{x-y}{\|x-y\|}, \quad U = I - 2uu^T$$

とおくと $Ux = y$.

証明　明らか。 ■

Householder 行列による三重対角化

$$
\begin{aligned}
A_0 &= A, \\
A_1 &= U_0 A_0 U_0 = \begin{bmatrix} * & * & 0 & \cdots & 0 \\ * & * & * & \cdots & * \\ 0 & * & * & \cdots & * \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & * & * & \cdots & * \end{bmatrix}, \quad U_0 = I - 2u_0 u_0^T, \\
A_2 &= U_1 A_1 U_1 = \begin{bmatrix} * & * & 0 & 0 & \cdots & 0 \\ * & * & * & 0 & \cdots & 0 \\ 0 & * & * & * & \cdots & * \\ 0 & 0 & * & * & \cdots & * \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & * & * & \cdots & * \end{bmatrix}, \quad U_1 = I - 2u_1 u_1^T, \\
& \cdots\cdots\cdots \\
A_{N-2} &= U_{N-3} A_{N-3} U_{N-3} = \begin{bmatrix} * & * & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ * & * & * & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & * & * & * & \ddots & 0 & 0 & 0 & 0 \\ 0 & 0 & * & * & \ddots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \ddots & * & * & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddots & * & * & * & 0 \\ 0 & 0 & 0 & 0 & \cdots & 0 & * & * & * \\ 0 & 0 & 0 & 0 & \cdots & 0 & 0 & * & * \end{bmatrix}, \quad U_{N-3} = I - 2u_{N-3} u_{N-3}^T
\end{aligned}
$$

のように $N-2$ 回の Householder 変換で三重対角化する。この 1 ステップ (A_{k-1} から A_k を作るところ) を発見的に説明しよう。面倒なので A_{k-1} を単に A と書き、

$$
A = \left[\begin{array}{cc|c} \multicolumn{2}{c|}{C} & \begin{array}{c} \mathbf{0} \\ \hline b^T \end{array} \\ \hline \mathbf{0} & \multicolumn{1}{|c|}{b} & B \end{array}\right]
$$

とブロック分けしておく。$u = u_k$ の最初の k 成分を 0 とおく、すなわち

$$
u = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ * \\ \vdots \\ * \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ \\ v \\ \\ \end{pmatrix}, \quad v \in \mathbf{R}^{N-k}
$$

とすると

$$U = I_N - 2uu^T = \begin{bmatrix} I_k & O \\ O & Q \end{bmatrix}, \quad Q = I_{N-k} - 2vv^T$$

となるので、

$$\text{新}\,A = UAU$$

とすると対応するブロックは

$$\begin{aligned} \text{新}\,C &= C, \\ \text{新}\,B &= QBQ, \\ \text{新}\,b &= Qb \end{aligned}$$

目標は

$$\text{新}\,b = Qb = \begin{pmatrix} s \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

の形にすることである。補題 3.3 を考えると $s = \pm\|b\|$ とすればよい。

$$s \text{ の符号} = -b \text{ の第 } 1 \text{ 成分 } (-b_1) \text{ の符号}$$

となるように選ぶと桁落ちが起こりにくい。まとめると

$$(\heartsuit_k) \quad \begin{cases} s = -\mathrm{sign}\,(b_1) \cdot \|b\|, \\ \text{新}\ b = \begin{pmatrix} s \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \\ v = \dfrac{b - \text{新}\ b}{\|b - \text{新}\ b\|}, \\ Q = I_{N-k} - 2vv^T \end{cases}$$

全体としては

for $k := 1$ to $N-2$ do
begin
　　第 k 列の第 $k+1$ 成分以降を b とする。
　　$(\heartsuit_k)$ により s, 新 b, v, Q を作る。
　　新 $B = QBQ$ を計算する。
end

色々な工夫が可能である。まとめると

$$\begin{cases} s & = & -\mathrm{sign}\ (b_1) \times \|b\| \\ w & = & b - \text{新}\ b = \text{第 1 成分は旧}\ b\ \text{の第 1 成分}\ -s,\ \text{その他の成分はそのまま} \\ \|w\|^2 & = & 2\{s^2 - (b\ \text{の第一成分}) \times s\} \\ p & = & \dfrac{Bw}{(\|w\|^2/2)} \\ \alpha & = & w^T p/\|w\|^2 \\ q & = & p - \alpha w \\ \text{新}\ B & = & B - wq^T - qw^T \quad (\text{対称なので半分の計算で OK}) \end{cases}$$

注: この文書の過去の版で、$\|w\|^2$ の計算式の符号を間違えて

$$\|w\|^2 = 2\{s^2 + (b\ \text{の第 1 成分}) \times s\} \quad (\text{この式は間違っている！})$$

のようにしていました。ごめんなさい。■

3.2 Lanczos (ランチョス) 法

(この話題については、『固有値問題ノートの補足』http://nalab.mind.meiji.ac.jp/~mk/labo/text/eigenvalues-add.pdf に説明を書いた。そのうちこちらとマージする。)

実対称行列 A が直交行列 P で三重対角化されたとする:

$$B = P^T A P = \begin{pmatrix} \alpha_1 & \beta_1 & 0 & & \mbox{\Large 0} \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{N-2} & \alpha_{N-1} & \beta_{N-1} \\ \mbox{\Large 0} & & 0 & \beta_{N-1} & \alpha_N \end{pmatrix}.$$

これから $AP = PB$ であるが、$P = (u_1 u_2 \cdots u_N)$ とおくと、

$$A(u_1 u_2 \cdots u_N) = (u_1 u_2 \cdots u_N) \begin{pmatrix} \alpha_1 & \beta_1 & 0 & & \mbox{\Large 0} \\ \beta_1 & \alpha_2 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \beta_{N-2} & \alpha_{N-1} & \beta_{N-1} \\ \mbox{\Large 0} & & 0 & \beta_{N-1} & \alpha_N \end{pmatrix}$$

となるから

$$\begin{aligned} Au_1 &= \alpha_1 u_1 + \beta_1 u_2 \\ Au_2 &= \beta_1 u_1 + \alpha_2 u_2 + \beta_2 u_3 \\ \cdots & \\ Au_i &= \beta_{i-1} u_{i-1} + \alpha_i u_i + \beta_i u_{i+1} \\ \cdots & \\ Au_N &= \beta_{N-1} u_{N-1} + \alpha_N u_N \end{aligned}$$

第 k 行と u_k の内積を作ると

$$\alpha_k = {u_k}^T A u_k. \tag{1}$$

また

$$v_{k+1} := \begin{cases} Au_1 - \alpha_1 u_1 & (k=0) \\ Au_k - (\beta_k u_{k-1} + \alpha_k u_k) & (1 \le k \le N-1) \end{cases} \tag{2}$$

とおくと、$v_{k+1} = \beta_k u_{k+1}$, $\|u_{k+1}\| = 1$ であるから、

$$\beta_k = \pm\|v_{k+1}\| \tag{3}$$
$$u_{k+1} = \frac{v_{k+1}}{\beta_k}. \tag{4}$$

これから、最初に適当に $\|u_1\| = 1$ となる u_1 を選び、以下 $\beta_k \neq 0$ である限り、計算を進めることができる。

3.3 実験プログラムたたき台

実験用行列生成のヒント

```
% ランダム Hessenberg 行列
function ret = rand_h(n)
  ret = triu(rand(n,n)) + diag(rand(n-1,1),-1);

% ランダム三重対角行列
function ret = rand_t(n)
  ret=diag(rand(n,1),0)+diag(rand(n-1,1),1)+diag(rand(n-1,1),-1);

% ランダム実対称三重対角行列
function ret = rand_st(n)
  u=rand(n-1,1);
  ret=diag(rand(n,1),0)+diag(u,1)+diag(u,-1);

% ランダム実対称行列
function ret = rand_s(n)
  a=rand(n,n);
  ret = (a+a')/2;

% QR 変換 (回数指定)
function QR = qr_iteration(a,n)
  QR = a;
  for i=1:n
    [q r]=qr(QR);
    QR=r*q;
  end

% 下三角部分のノルム
function n = norm_l(a)
  n = norm(a-triu(a));
```

beki.m

```
%
% beki.m --- 幂乗法の系統の算法の実験
%

format long

a=[1,1,1;1,2,2;1,2,3]
eigen=eig(a)

maxiter=20; r=zeros(maxiter,1);

disp("幂乗法")
x=ones(3,1);
for k=1:maxiter
  y=a*x; x=y/norm(y); r(k)=x'*(a*x);
end
subplot(1,4,1)
semilogy(1:maxiter,abs(r-eigen(3)))
% 絶対値最大の固有値に属する固有ベクトルを保存しておく
u=x;

disp("逆反復法")
x=ones(3,1);
for k=1:maxiter
  y=a\x; x=y/norm(y); r(k)=x'*(a*x);
end
subplot(1,4,2)
semilogy(1:maxiter,abs(r-eigen(1)))

disp("シフト法")
kinji=0.6;
ap=a-kinji*eye(3,3);
x=ones(3,1);
for k=1:maxiter
  y=ap\x; x=y/norm(y); r(k)=x'*(ap*x)+kinji;
end
subplot(1,4,3)
semilogy(1:maxiter,abs(r-eigen(2)))

x=ones(3,1);
x=x-(u'*x)*u;
for k=1:maxiter
  y=a*x; x=y/norm(y); x=x-(u'*x)*u; r(k)=x'*(a*x);
end
subplot(1,4,4)
semilogy(1:maxiter,abs(r-eigen(2)))
```

householder.m

```
% Householder
function a=householder(A)
  [n,n] = size(A);
  a = A;
  for k=1:n-1
    b = a(k+1:n,k);
    B = a(k+1:n,k+1:n);
    s = - sign(b(1)) * norm(b);
    w = b;
    w(1) = w(1) - s;
    normw2 = 2 * s * (s - b(1));
    p = (B * w) / (normw2/2);
    alpha = (w'*p) / normw2;
    q = p - alpha * w;
    a(k+1:n,k+1:n) = B - w * q' - q * w';
    a(k+2:n,k) = zeros(n-k-1,1);
    a(k,k+2:n) = zeros(n-k-1,1)';
    a(k+1,k) = s;
    a(k,k+1) = s;
  end
```

なお、授業のレポートである福澤 [7] も参考になるかも。

4 二分法

二分法 (**bisection method**, **Strum method**) を解説する。

- これは実対称行列専用の方法である (中間値の定理を用いるので、固有多項式が実係数で、根がすべて実数でないといけない)。
- 前節で説明した方法を用いて、与えられた実対称行列を三重対角化しておくことで、二分法を効率的に実行することが出来る。4.1 では最初から三重対角行列に対する説明になっている。
- 固有値は固有多項式の根であるが、Strum 列の理論によって、ある区間内の固有値の個数を計算することが出来る。このことと、いわゆる二分探索 (binary search) の方法を組み合わせて得られるのが、固有値計算手法としての二分法である。

4.1 伝統的な説明

$$T = \begin{pmatrix} a_1 & b_1 & 0 & & \text{\huge 0} \\ b_1 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{N-2} & a_{N-1} & b_{N-1} \\ \text{\huge 0} & & 0 & b_{N-1} & a_N \end{pmatrix}$$

を実対称三重対角行列とする。

$b_k \neq 0$ $(k=1,2,\cdots,N-1)$ と仮定する。もしある k に対して $b_k=0$ ならば

$$T = \begin{pmatrix} T' & O \\ O & T'' \end{pmatrix}$$

となり、T', T'' の固有値を求める問題に帰着できるから、一般性は失わない。

$p_k(\lambda)$ を $\lambda I - T$ の k 次首座小行列式とする $(k=0,1,\cdots,N)$。すなわち

$$p_k(\lambda) := \begin{cases} \det(\lambda I_k - T_k) & (k=1,2,\cdots,N) \\ 1 & (k=0). \end{cases}$$

ただし、

$$I_k = k \text{ 次の単位行列}, \quad T_k = T \text{ の } k \text{ 次首座小行列} = \begin{pmatrix} a_1 & b_1 & 0 & & \multicolumn{1}{c}{\text{\huge 0}} \\ b_1 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{k-2} & a_{k-1} & b_{k-1} \\ \text{\huge 0} & & 0 & b_{k-1} & a_k \end{pmatrix}.$$

すぐ分かることは、

補題 4.1 ($p_k(\lambda)$ の計算は簡単)

$$\begin{cases} p_0(\lambda) &=& 1 \\ p_1(\lambda) &=& \lambda - a_1 \\ p_{k+1}(\lambda) &=& (\lambda - a_{k+1})p_k(\lambda) - {b_k}^2 p_{k-1}(\lambda) \quad (k=1,2,\cdots,N-1) \\ p_N(\lambda) &=& \det(\lambda I - T). \end{cases}$$

補題 4.2 $\{p_k(\lambda)\}_{k=0}^N$ は Strum 列である。すなわち

(i) $p_0(\lambda)$ は定符号。

(ii) $p_k(\lambda)$ の根は有限個 $(k=1,2,\ldots,N)$.

(iii) $p_k(\lambda), p_{k+1}(\lambda)$ は共通根を持たない。

(iv) $p_k(\lambda_0)=0 \Longrightarrow p_{k-1}(\lambda_0)p_{k+1}(\lambda_0)<0$ $(k=1,2,\cdots,N-1)$.

(v) $p_N(\lambda_0)=0 \Longrightarrow p'_N(\lambda_0)p_{N-1}(\lambda_0)>0$.

証明 Strum 列については別に一つの章をもうけて説明するので、そこで述べることにする。
■

符号の変化数 $\{p_i(\lambda)\}_{i=0}^N$ を Strum 列とするとき、$p_N(x) \neq 0$ なる x に対して

$$N(x) \overset{\text{def.}}{\equiv} \text{“}\{p_0(x), p_1(x), \cdots, p_N(x)\}\text{” の}\textbf{符号の変化数}$$

とおく。例えば

$p_0(x)$	$p_1(x)$	$p_2(x)$	$p_3(x)$	$p_4(x)$	$p_5(x)$	$p_6(x)$	$p_7(x)$	$p_8(x)$	$p_9(x)$	$p_{10}(x)$
$+$	$-$	$-$	$-$	0	$+$	$+$	$+$	$+$	$-$	$-$

では $N(x) = 3$.

この符号の変化数は (特別な注意をせずに) 数値的に安定して計算できる。つまり絶対値が非常に小さくて、符号の判別がつきにくい場合も、「符号の変化数」そのものは疑いがなく計算できる。例えば

$p_{k-1}(x)$	$p_k(x)$	$p_{k+1}(x)$
$+$	絶対値小	$-$

または

$p_{k-1}(x)$	$p_k(x)$	$p_{k+1}(x)$
$-$	絶対値小	$+$

において $p_k(x)$ が正であっても負であっても 0 であっても符号の変化数の計算にとっては影響がない。注意すべきは Strum 列の条件 (iv) から

$p_{k-1}(x)$	$p_k(x)$	$p_{k+1}(x)$
$+$	絶対値小	$+$

または

$p_{k-1}(x)$	$p_k(x)$	$p_{k+1}(x)$
$-$	絶対値小	$-$

のような場合 (もしこうなったら符号の変化数の計算がむつかしい) が起こり得ないことである。

Strum の定理によって、$p_N(a)p_N(b) \neq 0$ なる $[a, b]$ において、$[a, b]$ 内の零点の個数は $N(a) - N(b)$ であることが分かる。

4.2 Sylvester の慣性律による説明

二分法を Sylvester の慣性律を用いて説明するのは、森・杉原・室田 [5] による。

線形代数でおなじみの **Sylvester の慣性律**「行列の符号数は座標変換で変化しない」を復習しよう。正則行列 M に対して

$$\pi(M) := M \text{ の固有値のうち正のものの個数},$$
$$\zeta(M) := M \text{ の固有値のうち } 0 \text{ のものの個数},$$
$$\nu(M) := M \text{ の固有値のうち負のものの個数}$$

とおく。

定理 4.3 (Sylvester の慣性律による説明) A を 実対称行列、S を正則行列で $B = S^T AS$ とするとき

$$\pi(A) = \pi(B), \quad \zeta(A) = \zeta(B), \quad \nu(A) = \nu(B).$$

任意に選んだ $\sigma \in \mathbf{R}$ に対して、$A - \sigma I$ を LDU 分解する、すなわち

$$A - \sigma I = LDL^T, \quad L \text{ は下三角行列}, \quad D \text{ は対角行列}.$$

このとき定理から

$$\pi(A-\sigma I)=\pi(D),\quad \zeta(A-\sigma I)=\zeta(D),\quad \nu(A-\sigma I)=\nu(D)$$

であるが、$\pi(D)$, $\zeta(D)$, $\nu(D)$ はすぐに分かる。従って、行列 A の固有値について、任意に与えられた実数 σ よりも大きいもの、小さいものの個数が勘定できることになる。

A が三重対角行列であれば、LDU 分解は $O(n)$ の計算量で済ませられることを注意しておく。

5 QR 法

この節の内容は主に一松 [2] による。

5.1 正則行列の QR 分解

$A\in GL(n;\mathbf{C})$ の列ベクトルを $\boldsymbol{a}_1,\cdots,\boldsymbol{a}_n$ とする:

$$A=(\boldsymbol{a}_1\ \boldsymbol{a}_1\ \cdots\boldsymbol{a}_n). \tag{5}$$

この $\boldsymbol{a}_1,\cdots,\boldsymbol{a}_n$ に Gram-Schmidt の直交化法を施して $\boldsymbol{q}_1,\cdots,\boldsymbol{q}_n$ を作る。つまり

$$\boldsymbol{v}_k:=\boldsymbol{a}_k-\sum_{i=1}^{k-1}\langle\boldsymbol{a}_k,\boldsymbol{q}_i\rangle\boldsymbol{q}_i,\quad \boldsymbol{q}_k\overset{\text{def.}}{=}\frac{\boldsymbol{v}_k}{\|\boldsymbol{v}_k\|}\quad (k=1,2,\cdots,n)$$

として計算するわけだが[4]、中間変数 r_{ik} $(1\le i\le k\le n)$ を導入して、次のように書き換えておこう。

$$\boldsymbol{v}_k:=\boldsymbol{a}_k-\sum_{i=1}^{k-1}r_{ik}\boldsymbol{q}_i,\quad r_{ik}:=\langle\boldsymbol{a}_k,\boldsymbol{q}_i\rangle, \tag{6}$$

$$\boldsymbol{q}_k:=\frac{\boldsymbol{v}_k}{r_{kk}},\quad r_{kk}:=\|\boldsymbol{v}_k\|\quad (k=1,2,\cdots,n). \tag{7}$$

さて

$$Q:=(\boldsymbol{q}_1\ \boldsymbol{q}_2\ \cdots\ \boldsymbol{q}_n),\quad R:=\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n}\\ & r_{22} & \cdots & r_{2n}\\ & & \ddots & \vdots\\ \mathbf{0} & & & r_{nn}\end{pmatrix}$$

とおくと、Q は unitary 行列で、

$$A=QR.$$

ここまで A は複素行列として計算してきたが、実行列である場合は、Q, R も実行列 (したがって Q は実直交行列) である。

[4] これは線型代数の教科書の、Gram-Schmidt の直交化法の説明に載っている式であるが、数値計算に用いるには問題も多いとのこと。

定義 5.1 (QR 分解) 正則行列 $A \in GL(n; \mathbf{C})$ に対して、

$$A = QR \quad (Q \text{ は unitary 行列}, R \text{ は対角線分がすべて正の上三角行列})$$

となる Q, R を見い出すことを A を **QR 分解**すると言う。

上の議論から任意の正則行列は QR 分解可能であることが分かったが、実はこれは一意的である。

命題 5.2 (QR 分解の一意性) 任意の $A \in GL(n; \mathbf{C})$ に対して、QR 分解はただ一つしかない。

証明 $A = QR$ は

$$\boldsymbol{a}_k = \sum_{i=1}^{k} r_{ik}\boldsymbol{q}_i \quad (k = 1, 2, \cdots, n)$$

と書ける。$k = 1$ についての条件

$$\boldsymbol{a}_1 = r_{11}\boldsymbol{q}_1$$

から

$$r_{11} = |r_{11}| = \frac{\|\boldsymbol{a}_1\|}{\|\boldsymbol{q}_1\|} = \|\boldsymbol{a}_1\|,$$

$$\boldsymbol{q}_1 = \frac{\boldsymbol{a}_1}{r_{11}}.$$

次に $k = 2$ についての条件

$$\boldsymbol{a}_2 = r_{12}\boldsymbol{q}_1 + r_{22}\boldsymbol{q}_2$$

から、まず $\boldsymbol{q}_1$ との内積を取って、

$$r_{12} = \langle \boldsymbol{a}_1, \boldsymbol{q}_1 \rangle.$$

これから

$$\boldsymbol{v}_2 := \boldsymbol{a}_2 - r_{12}\boldsymbol{q}_1$$

は計算できて、

$$\boldsymbol{v}_2 = r_{22}\boldsymbol{q}_2$$

であるから、

$$r_{22} = |r_{22}| = \frac{\|\boldsymbol{v}_2\|}{\|\boldsymbol{q}_2\|} = \|\boldsymbol{v}_2\|, \quad \boldsymbol{q}_2 = \frac{\boldsymbol{v}_2}{r_{22}}.$$

以下 $k = 3, \cdots, n$ と順に同様に計算できる。■

別証明 A の二つの QR 分解

$$A = Q_1R_1, \quad A = Q_2R_2$$

があれば、$Q_1R_1 = Q_2R_2$ から

$$Q_2^*Q_1 = R_2R_1^{-1}. \tag{8}$$

この等式の左辺は unitary 行列である。実際

$$(Q_2^* Q_1)(Q_2^* Q_1)^* = Q_2^* Q_1 Q_1^* Q_2^{**} = Q_2^* Q_2 = I.$$

また (8) の右辺は上三角行列で、対角成分はすべて正である。

unitary かつ上三角かつ対角成分がすべて正という行列は単位行列に限られることは容易に証明できる。ゆえに

$$Q_2^* Q_1 = R_2 R_1^{-1} = I.$$

これから

$$Q_1 = Q_2, \quad R_1 = R_2. \blacksquare$$

注意 5.3 (QR 分解を利用した連立 1 次方程式の解法) $A = QR$ という QR 分解があるとき、

$$Ax = b$$

は

$$QRx = b$$

であるから

$$x = R^{-1}(Q^* b).$$

上三角行列の逆行列をかける計算は、乗算 $n^2/2$ 回程度の計算量で計算できるから、x は乗算 $3n^2/2$ 回程度の計算量で計算できることが分かる。QR 分解は $O(n^3)$ 程度の計算量で求まることは分かるから、Gauss の消去法に基づく LU 分解とオーダーだけは匹敵する計算法である。(決して有利な方法ではないが。) ■

5.2 QR 変換

$$A = QR$$

という QR 分解が得られたとき、順番を変えて掛け算した行列 A_1 を作る。

$$A_1 = RQ$$

これは $R = Q^* A$ より

$$A_1 = Q^* A Q$$

となるので、実は A を Q で相似変換したものである。これを **QR 変換**と言う。

固有値を求めるために QR 変換を用いる場合、まず Hessenberg 行列に相似変換した後に使われることが多い。

- Hessenberg 行列の QR 変換は Hessenberg 行列である。
- 実対称行列の QR 変換は実対称行列である。
  ($\because$ 対称行列の直交行列による相似変換は実対称行列であるから。)
- Hermite 行列の QR 変換は Hermite 行列である。
- 三重対角行列の QR 変換は三重対角行列とは限らないが、実対称三重対角行列の QR 変換は実対称三重対角行列である (実対称三重対角 == 実対称かつ Hessenberg だから)。

5.3 QR 法の原理

行列 A が与えられたとき、

$$
(9) \qquad \begin{aligned}
&A_0 = A = Q_0R_0, \quad A_1 = R_0Q_0,\\
&A_1 = Q_1R_1, \quad A_2 = R_1Q_1,\\
&A_2 = Q_2R_2, \quad A_3 = R_2Q_2,\\
&\quad \cdots\\
&A_k = Q_kR_k, \quad A_{k+1} = R_kQ_k,\\
&\quad \cdots
\end{aligned}
$$

と QR 変換を繰り返すと、A_m の対角線の下側の成分はすべて 0 に収束する。

注意 5.4 (たまにある誤解を正す) このことを「A_k は上三角行列に収束する」と言うことがあるが、対角線の上にある成分が特定の値に収束するわけではない。言い換えると、特定の上三角行列 U があって、

$$\lim_{k\to\infty} A_k = U$$

となるわけではない。■

一般性の追求はほどほどにして、次の定理を証明しよう。

定理 5.5 (QR 法の原理) $A \in GL(n;\mathbf{C})$ で、その固有値 λ_j $(j = 1, 2, \cdots, n)$ はすべて相異なり、

$$|\lambda_1| > |\lambda_2| > \cdots > |\lambda_n|$$

と仮定する。このとき $k \to \infty$ とすると、A_k の対角線より下のすべての成分は 0 に収束し、A_k の対角成分は λ_1, $\cdots$, λ_n に収束する。

注意 5.6 実対称行列については、上の定理の仮定はまあまあ一般的であるが、非対称な行列については、虚根を持たないことが必要になるので (虚根を持てば、複素共役となるので、絶対値は等しくなってしまう)、あまり一般的な仮定とは言えない。

証明

ステップ 1 (A^{k+1} の QR 分解) まず

主張 1

$P_k := Q_0 Q_1 \cdots Q_{k-1} Q_k$ とおくと

$$A_k = P_{k-1}^* A P_{k-1}. \tag{10}$$

実際、

$$\begin{aligned}
&A_1 = R_0 Q_0 = (Q_0^* A) Q_0 = Q_0^* A Q_0,\\
&A_2 = R_1 Q_1 = (Q_1^* A_1) Q_1 = Q_1^* (Q_0^* A Q_0) Q_1 = Q_1^* Q_0^* A Q_0 Q_1,\\
&A_2 = R_2 Q_2 = (Q_2^* A_2) Q_2 = Q_2^* (Q_1^* Q_0^* A Q_0 Q_1) Q_2 = Q_2^* Q_1^* Q_0^* A Q_0 Q_1 Q_2,\\
&\cdots\\
&A_k = R_k Q_k = (Q_k^* A_{k-1}) Q_k = Q_k^* (Q_{k-1}^* \cdots Q_1^* Q_0^* A Q_0 Q_1 \cdots Q_{k-1}) Q_k,\\
&\cdots
\end{aligned}$$

となる (帰納法で証明すべきかもしれないが)。

主張 2

$U_k := R_k R_{k-1} \cdots R_1 R_0$ とおくと $A^{k+1} = P_k U_k$ であり、これは A^{k+1} の QR 分解である。

まず

$$\begin{aligned}
P_k U_k &= (Q_0 Q_1 \cdots Q_{k-1} Q_k)(R_k R_{k-1} \cdots R_1 R_0) = (Q_0 Q_1 \cdots Q_{k-1})(Q_k R_k)(R_{k-1} \cdots R_1 R_0)\\
&= P_{k-1} A_k U_{k-1}
\end{aligned}$$

で、主張 1 より $P_{k-1} A_k = A P_{k-1}$ だから、

$$P_k U_k = A P_{k-1} U_{k-1}.$$

この式を再帰的に用いて、

$$P_k U_k = A P_{k-1} U_{k-1} = A(A P_{k-1} U_{k-2}) \cdots = A^k P_0 U_0 = A^k Q_0 R_0 = A^k A = A^{k+1}.$$

一方、P_k は unitary 行列の積であるから unitary 行列、U_k は対角線分が正である上三角行列の積であるから対角成分が正である上三角行列である。ゆえに $A^{k+1} = P_k U_k$ は A^{k+1} の QR 分解である。

ステップ 2 (A_k の挙動) A は固有値がすべて相異なるから対角化が可能である。すなわち $\exists M \in GL(n; \mathbf{C})$ s.t.

$$M^{-1} A M = \Lambda, \quad \Lambda = \mathrm{diag}(\lambda_1, \cdots, \lambda_n).$$

M を $M = QR$ と QR 分解、M^{-1} を $M^{-1} = LU$ (ただし L の対角成分はすべて 1) と LU 分解して、

$$A^{k+1} = M \Lambda^{k+1} M^{-1} = (QR) \Lambda^{k+1} (LU) = QR(\Lambda^{k+1} L \Lambda^{-(k+1)}) \Lambda^{k+1} U. \tag{11}$$

ここで $\Lambda^{k+1}L\Lambda^{-(k+1)}$ は下三角行列で、対角成分はすべて 1 に等しく、$i>j$ のとき (i,j) 成分は

$$\ell_{ij}\left(\frac{\lambda_i}{\lambda_j}\right)^{k+1}$$

に等しい (ただし ℓ_{ij} は L の (i,j) 成分)。そこで

$$\Lambda^{k+1}L\Lambda^{-(k+1)}=I+E_k$$

とおくと、$\lim_{k\to\infty}E_k=O$. 上式を (11) 代入して、

$$\begin{aligned}A^{k+1}&=QR(\Lambda^{k+1}L\Lambda^{-(k+1)})\Lambda^{l+1}U\\&=QR(I+E_k)(R^{-1}R)(\Lambda^{l+1}U)=QR(I+E_k)R^{-1}(R\Lambda^{l+1}U)\\&=Q(I+RE_kR^{-1})(R\Lambda^{l+1}U)\end{aligned}\tag{12}$$

この $I+RE_kR^{-1}$ を QR 分解する:

$$I+RE_kR^{-1}=S_kT_k.\tag{13}$$

$I+RE_kR^{-1}\to I$ $(k\to\infty)$ で、QR 分解の連続性は明らかだから、

$$\lim_{k\to\infty}S_k=I,\quad \lim_{k\to\infty}T_k=I.$$

(13) を (12) に代入して、

$$A^{k+1}=Q(S_kT_k)(R\Lambda^{k+1}U)=(QS_k)(T_kR\Lambda^{k+1}U).\tag{14}$$

この右辺は unitary 行列と、上三角行列の積の形になっている。

ステップ 3 (QR 分解との関係) $U=(u_{ij})$ として、

$$D_1=\operatorname{diag}\left(\frac{\overline{\lambda_1}}{|\lambda_1|},\frac{\overline{\lambda_2}}{|\lambda_2|},\cdots,\frac{\overline{\lambda_n}}{|\lambda_n|}\right),\quad D_2=\operatorname{diag}\left(\frac{\overline{u_{11}}}{|u_{11}|},\frac{\overline{u_{22}}}{|u_{22}|},\cdots,\frac{\overline{u_{nn}}}{|u_{nn}|}\right)$$

とおく。これを使って (14) を

$$A^{k+1}=(QA_kD_2^*D_1^{k*})[(D_1^kD_2)(T_kR)(D_1^kD_2)^*(D_1\Lambda)^k(D_2U)]$$

のように書き換えると、これは A^{k+1} の QR 分解になる。すでに $A^{k+1}=P_kU_k$ という QR 分解が得られていたので、QR 分解の一意性から、

$$P_k=QS_kD_2^*D_1^{k*}.\tag{15}$$

ステップ 4　(10) と (15) から

$$A_{k+1} = P_k^* A P_k = D_1^k D_2 S_k^* (Q^* A Q) S_k D_2^* D_1^{k*}.$$

$$A = M \Lambda M^{-1} = (QR)\Lambda(R^{-1}Q^*)$$

より

$$Q^* A Q = R \Lambda R^{-1}.$$

この行列は (右辺に注目することで) 上三角行列で、かつ対角線上に $\lambda_1, \cdots, \lambda_n$ がこの順にならんでいる。$k \to \infty$ のとき、$S_k \to I$ だから

$$A_{k+1} \to D_1^k (D_2 R \Lambda R^{-1} D_2^*) D_1^{*k}$$

この右辺は上三角行列で、対角成分は $\lambda_1, \lambda_2, \cdots, \lambda_n$ である。■

この定理から $k \to \infty$ とするときの収束の速さは

$$\left(\frac{\lambda_i}{\lambda_j}\right)^k$$

という因子に関わることが分かる。

5.4　QR 分解のアルゴリズム

QR 分解の存在証明として、線型代数の本でおなじみの Gram-Schimidt の直交化法を用いたが、QR 分解を実際に計算するには、以下のような工夫をする。

5.4.1　修正 Gram-Schmidt の方法

5.4.2　Givens 変換を用いる方法

5.4.3　鏡映変換を用いる方法

5.5　QR 法の歴史

1961 年に Francis が複素行列の固有値の計算法として導入したもの (一松 [2])。

5.6　Octave での実験

MATLAB や Octave には `qr()` という QR 分解をする関数があって、QR 法の実験が簡単にできる。

以下、実験をする際の参考に、小プログラム集を。

```
# qr_t.m --- QR 変換
function QR = qr_t(a)
  [q r]=qr(a);
  QR=r*q;
endfunction
# qr_iteration.m --- QR 変換の反復
function QR = qr_iteration(a,n)
  QR = a;
  for i=1:n
    [q r]=qr(QR);
    QR=r*q;
  end
endfunction
# norm_l.m --- 下三角部分 (対角線含まず) のノルム
function n = norm_l(a)
  n = norm(a-triu(a));
endfunction
# rand_h.m --- テスト用 Hessenberg 行列
function ret = rand_h(n)
  ret = triu(rand(n,n)) + diag(rand(n-1,1),-1);
endfunction
# rand_t.m --- テスト用三重対角行列
function ret = rand_t(n)
  ret=diag(rand(n,1),0)+diag(rand(n-1,1),1)+diag(rand(n-1,1),-1);
endfunction
# rand_s.m --- テスト用対称行列
function ret = rand_s(n)
  a=rand(n,n);
  ret = (a+a')/2;
endfunction
# rand_st.m --- テスト用対称三重対角行列
function ret = rand_st(n)
  u=rand(n-1,1);
  ret=diag(rand(n,1),0)+diag(u,1)+diag(u,-1);
endfunction
# rand_hermite.m --- テスト用 Hermite 行列
function ret = rand_hermite(n)
  a=rand(n,n)+i*rand(n,n);
  ret = (a+a')/2;
endfunction
```

6 特異値

いつか勉強して書きたい。

日本語の解説はかつては、一松 [2], 一松 [3] くらいしか見当たらなかったが、今では色々ある。

7 固有値計算のためのパッケージ

(準備中)

A　参考書

森・杉原・室田 [5], 一松 [2], [3] 以外には、名取 [9], 戸川 [6], シャトラン [1] など。

B　Strum の方法

Strum の方法とは、実軸上の区間 $[a,b]$ における多項式 $f(x)$ の実根の個数に関する有名な Strum の定理をもとにした多項式の根の計算法であり、数値計算法のあちこちで顔を出す。ここで簡単にまとめておく。なお、以下の記述は森 [4], 一松 [2] から適当に抜き書きしたものであるが、実は高木 [8] の第 3 章の方が詳しいかもしれない (暇があったら書き直そう)[5]。

B.1　スツルムの定理

定義 B.1 (Strum 列) $[a,b]$ を $\mathbf{R}$ の区間とするとき、次の 4 つの条件を満足する実係数多項式列

$$f_0(x)=f(x), f_1(x), f_2(x), \cdots, f_\ell(x)$$

は区間 $[a,b]$ において **Strum 列**をなすという。

(1) $\forall x\in[a,b]$ に対して、隣り合う二つの多項式 $f_k(x)$, $f_{k+1}(x)$ は同時に 0 にはならない。

(2) $x_0\in[a,b]$, $k\in\{1,2,\cdots,\ell-1\}$ において $f_k(x_0)=0$ となれば、$f_{k-1}(x_0)f_{k+1}(x_0)<0.$

(3) 列の最後の多項式 $f_\ell(x)$ は $[a,b]$ において一定の符号を持つ。

(4) $x_0\in[a,b]$ において $f(x_0)=0$ ならば $f'(x_0)f_1(x_0)>0.$

[5] 何となく、すごく高尚なことが書いてあるのではないかと、中を見もしないで敬遠していたのだが…損をした気分でいます。

定義 B.2 (符号変化の回数) 実係数多項式の列 $f(x) = f_0(x), f_1(x), \cdots, f_\ell(x)$ が区間 $[a,b]$ で Strum 列をなすとき、$f(x)$ の根でない $x \in [a,b]$ に対して、関数値の列

$$f_0(x), f_1(x), \cdots, f_\ell(x)$$

を左から右に見ていったときの**符号の変化の回数** $N(x)$ が定義できる。

- x において、すべての k につき $f_k(x) \neq 0$ が成り立つ場合の $N(x)$ の定義は明らか。
- ある k について $f_k(x) = 0$ となった場合は、仮定 (Strum 列の条件 (3) と x が $f(x)$ の根でないという仮定) から $k = 0, \ell$ はありえず $1 \leq k \leq \ell - 1$ で、その場合も Strum 列の条件 (2) によって、$f_{k-1}(x)f_{k+1}(x) < 0$ となることから、向う三軒両隣では

$f_{k-1}(x)$	$f_k(x)$	$f_{k+1}(x)$
$+$	0	$-$

または

$f_{k-1}(x)$	$f_k(x)$	$f_{k+1}(x)$
$-$	0	$+$

のいずれかのパターンしかありえない。そこで、この 3 項で 1 回符号変化したと数えることにする。

例 B.3 $+, +, -, 0, +, +, -$ では 3 回と数える。

定理 B.4 (Strum) 実係数多項式の列 $f(x) = f_0(x), f_1(x), \cdots, f_\ell(x)$ は区間 $[a,b]$ で Strum 列をなし、$f(a)f(b) \neq 0$ であるとする。このとき、$f(x) \neq 0$ なる $x \in [a,b]$ に対して、関数値の列

$$f_0(x), f_1(x), \cdots, f_\ell(x)$$

を左から右に見ていったときの符号の変化の回数を $N(x)$ とすると、方程式 $f_0(x) = 0$ の区間 $[a,b]$ における解の個数は $N(a) - N(b)$ である。

この定理によって、区間 $[a,b]$ 内に存在する $f(x) = 0$ の解を数を知ることができるが、二分探索の技法と組み合わせることで、解が存在する区間を好きなだけ小さくすることが出来、その区間内の適当な点を解の近似値として採用するという近似解法ができる。これを **Strum の方法**と呼ぶ。

証明 $f(x) = 0$ の解の個数は有限個である。そのうち $[a,b]$ に含まれるものを大きさの順に並べて

$$x_1 < x_2 < \cdots < x_n$$

とする。$n+1$ 個の区間

$$I_0 = [a, x_1), \quad I_1 = (x_1, x_2), \quad I_2 = (x_2, x_3), \quad I_{n-1} = (x_{n-1}, x_n), \quad I_n = (x_n, b]$$

の和集合において ($f(x)$ は 0 とならないので) $N(x)$ が定義できる。以下では

(1) 各区間 I_j において $N(x)$ は定数である:

$$\exists\{n_j\}_{j=0}^{n} \quad \text{s.t.} \quad N(x) = n_j \quad (x \in I_j,\, j = 0, 1, \cdots, n)$$

(2) $n_{j+1} - n_j = -1$ $(j = 0, 1, \cdots, n-1)$

であることを証明する (この二つから容易に $N(a) - N(b) = n$ が導かれる)。

(1) の証明 I_j 内のある部分区間 J において、すべての多項式 $f_k(x) \neq 0$ とすると、$f_k(x)$ の符号は J で一定であるので (中間値の定理から、符号が変化するには 0 にならないといけない)、$N(x)$ の値は変化しないことが分かる。

そこで、ある多項式 $f_k(x)$ の解 $x_* \in I_j$ を通過した場合を考える。Strum 列の条件から $k \neq \ell$, I_j の取り方から $k \neq 0$ であるから、$1 \leq k \leq \ell - 1$ である。Strum 列の条件 (2) から $f_{k-1}(x_*)f_{k+1}(x_*) < 0$ であるが、連続性から x_* の十分小さな近傍 V を取れば、そこで $f_{k-1}(x)$ と $f_{k+1}(x)$ は定符号となり、V 上で $f_{k-1}(x)f_{k+1}(x) < 0$ がなりたつ。それゆえ、列

$$f_0(x), f_1(x), \cdots, f_{k-1}(x), f_k(x), f_{k+1}(x), \cdots, f_\ell(x)$$

の部分列

$$f_{k-1}(x), f_k(x), f_{k+1}(x)$$

の符号については次の二つのいずれか一方だけしか起こらない。

(a)	$f_{k-1}(x)$	$f_k(x)$	$f_{k+1}(x)$	(b)	$f_{k-1}(x)$	$f_k(x)$	$f_{k+1}(x)$
	V 上 $-$		V 上 $+$		V 上 $+$		V 上 $-$

それゆえ、($f_k(x)$ の符号が何であっても) 任意の $x \in V$ について、部分列 $f_{k-1}(x)$, $f_k(x)$, $f_{k+1}(x)$ における符号の変化は 1 として $N(x)$ の値に算入される。これから x_* の十分小さな近傍で $N(x)$ の値に変化はないことが分かる。ゆえに $N(x)$ は I_j 上で定数である。

(2) の証明 $f(x) = 0$ の解 x_j において考える。Strum 列の条件 (4) は $f'(x_j)$ と $f_1(x_j)$ が同符号であることを示している。例えば $f'(x_j) > 0$ の場合、十分小さな $\varepsilon > 0$ を取ると、

- $f(x) < 0$ $(x \in [x_j - \varepsilon, x_j))$, $f(x_j) = 0$, $f(x) > 0$ $(x \in (x_j, x_j + \varepsilon])$.
- $f_1(x) > 0$ $(x \in (x_j - \varepsilon, x_j + \varepsilon))$.

これから

$$n_{j+1} - n_j = N(x_j + 0) - N(x_j - 0) = -1$$

であることが分かる。$f'(x_j) < 0$ の場合も同様の議論で $n_{j+1} - n_j = -1$ であることが分かる。
■

B.2 ユークリッドの互除法による Strum 列の生成

多項式 $f(x) \in \mathbf{R}[x]$ が与えられたとき、$f_0(x) = f(x)$ と $f_1(x) = f'(x)$ から Euclid の互除法を行い、関数列 $f_0(x), f_1(x), \cdots, f_\ell(x)$ を作る:

$$f_0(x) := f(x), \quad f_1(x) := f'(x), \tag{16}$$

$$\begin{aligned}
f(x) &= q_1(x) f_1(x) - f_2(x), \quad \deg f_2(x) < \deg f_1(x),\\
f_1(x) &= q_2(x) f_2(x) - f_3(x), \quad \deg f_3(x) < \deg f_2(x),\\
f_2(x) &= q_3(x) f_3(x) - f_4(x), \quad \deg f_4(x) < \deg f_3(x),\\
&\vdots\\
f_{k-1}(x) &= q_{k-1}(x) f_k(x) - f_{k+1}(x), \quad \deg f_{k+1}(x) < \deg f_k(x),\\
&\vdots\\
f_{\ell-2}(x) &= q_{\ell-1}(x) f_{\ell-1}(x) - f_\ell(x), \quad \deg f_\ell(x) < \deg f_{\ell-1}(x),\\
f_{\ell-1}(x) &= q_\ell(x) f_\ell(x).
\end{aligned} \tag{17}$$

(普通の互除法と異なり、$f_{k-1}(x)$ を $f_k(x)$ で割ったときの普通の剰余の (-1) 倍を $f_{k+1}(x)$ とすることに注意しよう。そうする理由は以下の (18) を成立させるためである。)

よく知られているように $f_\ell(x)$ は $f(x)$ と $f'(x)$ の最大公約多項式であるから、$f(x) \in \mathbf{R}[x]$ が重根を持たない場合、$f_\ell(x) \equiv$ 定数 $(\neq 0)$ となることに注意しよう。以下この場合に $f(x) = 0$ の解 (根) を求めることを考える。$f(x)$ が重根を持つ場合は $f(x)$ の代わりに $g(x) = f(x)/f_\ell(x)$ を考えることで同様の議論ができる。

定理 B.5 $f(x) \in \mathbf{R}[x]$ が重根を持たないとき、$f(a)f(b) \neq 0$ を満たす任意の区間 $[a, b]$ において、$f(x)$ と $f'(x)$ から互除法で作った剰余列 (詳しくは (16), (17) 参照)

$$f(x) = f_0(x), f'(x) = f_1(x), f_2(x), \cdots, f_\ell(x)$$

は Strum 列をなす。

証明 まず $f_\ell(x) \equiv$ 定数 であるから、Strum 列の条件 (3) は満たされている。次にある $x_0 \in [a, b]$, ある $k \in \{0, 1, \cdots, \ell - 1\}$ に対して

$$f_k(x_0) = f_{k+1}(x_0) = 0$$

となったとすると、式 (17) から $f_{k+2}(x_0)$ 以降の $f_j(x_0)$ もすべて 0 になり、特に $f_\ell(x_0) = 0$. これは $f_\ell(x)$ が定数関数 $(\neq 0)$ であることに矛盾する。ゆえに Strum 列の条件 (1) が満たされる。次にある点 x_0, ある $k \in \{1, 2, \cdots, \ell - 1\}$ に対して $f_k(x_0) = 0$ となったとすると、式 (17) から

$$f_{k-1}(x_0) = -f_{k+1}(x_0). \tag{18}$$

ゆえに Strum 列の条件 (2) も満たされる (条件 (3) から上式の値は 0 にならないことに注意)。最後に x_0 が $f(x)$ の根であるとき、$f(x)$ が重根を持たないという仮定から $f'(x_0)\neq 0$ で、

$$f'(x_0)f_1(x_0)=f'(x_0)^2>0$$

となり、条件 (4) も満たされる。■

B.3　3 重対角行列の固有多項式と Strum 列

$$T=\begin{pmatrix} a_1 & b_1 & 0 & & \\ b_1 & a_2 & b_2 & & \text{\Large 0} \\ & \ddots & \ddots & \ddots & \\ & & b_{N-2} & a_{N-1} & b_{N-1} \\ \text{\Large 0} & & 0 & b_{N-1} & a_N \end{pmatrix}$$

を実対称三重対角行列とする。

注意 B.6 (対称性の仮定について) 実は以下の議論で T の対称性はあまり本質的でなく、対角線をはさんで対称の位置にある成分が同符号、すなわち T の第 (i,j) 成分を t_{ij} と書くとき、

$$t_{k,k-1}t_{k-1,k}>0\quad k=2,\cdots,N$$

が成り立つことが本質的であるという指摘が一松先生の本にあった。なるほど。行列を Hessenberg 形にする算法を施すとき、特に与えられた行列が実対称であった場合には、結果が実対称三重対角になってしまうということで、(以下 Strum の方法を用いて固有値を求める段階になると) 実対称性はある意味では必要性はあまりないのだが、最初に実対称性がないと三重対角化できないわけで、まあ必ずついてくる「おまけ」ということでしょう。■

以下 $b_k\neq 0$ $(k=1,2,\cdots,N-1)$ と仮定する。もしある k に対して $b_k=0$ ならば

$$T=\begin{pmatrix} T' & O \\ O & T'' \end{pmatrix}$$

とブロック分けでき、T', T'' の固有値を求める問題に帰着できるから、一般性は失われない。

$p_k(\lambda)$ を $\lambda I-T$ の k 次首座小行列式とする $(k=0,1,\cdots,N)$。すなわち

(19)
$$p_k(\lambda):=\begin{cases}\det(\lambda I_k-T_k) & (k=1,2,\cdots,N)\\ 1 & (k=0).\end{cases}$$

ただし、

$$I_k=k\text{ 次の単位行列},\quad T_k=\begin{pmatrix} a_1 & b_1 & 0 & & \\ b_2 & a_2 & b_1 & & \text{\Large 0} \\ & \ddots & \ddots & \ddots & \\ & & b_{k-2} & a_{k-1} & b_{k-1} \\ \text{\Large 0} & & 0 & b_{k-1} & a_k \end{pmatrix}.$$

すぐ分かる命題を二つ。

補題 B.7 (漸化式) (19) で定義された $\{p_j(\lambda)\}_{j=0}^N$ について、以下が成り立つ。

$$\begin{cases} p_0(\lambda) &=& 1, \\ p_1(\lambda) &=& \lambda - a_1, \\ p_{k+1}(\lambda) &=& (\lambda - a_{k+1})p_k(\lambda) - {b_k}^2 p_{k-1}(\lambda) \quad (k = 1, 2, \cdots, N-1), \\ p_N(\lambda) &=& \det(\lambda I - T). \end{cases}$$

証明 行列式の展開定理を用いる。■

補題 B.8 (Strum 列であること) (19) で定義された $\{p_j(\lambda)\}_{j=0}^N$ を逆順に並べた多項式列

$$p_N(\lambda), p_{N-1}(\lambda), \cdots, p_1(\lambda), p_0(\lambda)$$

は任意の閉区間 $[a, b]$ において Strum 列である。すなわち

(1) 隣り合う二つの多項式 $p_k(\lambda)$, $p_{k+1}(\lambda)$ は共通根を持たない。

(2) ある $\lambda_0 \in \mathbf{R}$ において $p_k(\lambda_0) = 0$ ならば $p_{k-1}(\lambda_0)p_{k+1}(\lambda_0) < 0$ $(k = 1, 2, \cdots, N-1)$.

(3) 列の最後の多項式 $p_0(\lambda)$ は $\mathbf{R}$ において定符号である。

(4) ある $\lambda_0 \in \mathbf{R}$ において $p_N(\lambda_0) = 0$ ならば $p'_N(\lambda_0)p_{N-1}(\lambda_0) > 0$.

証明

(1) もしも $p_k(\lambda_0) = p_{k+1}(\lambda_0) = 0$ とすると、漸化式から ${b_k}^2 p_{k-1}(\lambda_0) = 0$. $b_k = 0$ と仮定したから $p_{k-1}(\lambda_0) = 0$. これを繰り返すと

$$0 = p_{k+1}(\lambda_0) = p_k(\lambda_0) = p_{k-1}(\lambda_0) = \cdots = p_2(\lambda_0) = p_1(\lambda_0) = p_0(\lambda_0).$$

これから

$$p_0(\lambda_0) = 0$$

これは $p_0(\lambda) \equiv 1$ に矛盾する。

(2) $p_k(\lambda_0) = 0$ を漸化式に代入すると $p_{k+1}(\lambda_0) = -{b_k}^2 p_{k-1}(\lambda_0)$. 前項より左辺 $\neq 0$. これから $p_{k+1}(\lambda_0)$, $p_{k-1}(\lambda_0)$ は異符号である。

(3) $p_0(\lambda) \equiv 1$ であるから明らか。

(4) 漸化式

$$p_k(\lambda) = (\lambda - a_k)p_{k-1}(\lambda) - {b_{k-1}}^2 p_{k-2}(\lambda) \tag{20}$$

を微分すると、

$$p'_k(\lambda) = p_{k-1}(\lambda) + (\lambda - a_k)p'_{k-1}(\lambda) - {b_{k-1}}^2 p'_{k-2}(\lambda). \tag{21}$$

(21) $\times p_{k-1}(\lambda) -$ (20) $\times p'_{k-1}(\lambda)$ より

$$p'_k(\lambda)p_{k-1}(\lambda) - p_k(\lambda)p'_{k-1}(\lambda) = b_{k-1}^2\left(p'_{k-1}(\lambda)p_{k-2}(\lambda) - p_{k-1}(\lambda)p'_{k-2}(\lambda)\right) + p_{k-1}(\lambda)^2$$

という漸化式が得られる。ここで

$$q_k(\lambda) := p'_k(\lambda)p'_{k-1}(\lambda) - p_k(\lambda)p_{k-1}(\lambda)$$

とおくと、漸化式は

$$q_k(\lambda) = p_{k-1}(\lambda)^2 + b_{k-1}^2 q_{k-1}(\lambda) \quad (k = 2, 3, \cdots, N).$$

となる。ところで

$$q_1(\lambda) = p'_1(\lambda)p_0(\lambda) - p_1(\lambda)p'_0(\lambda) = p'_1(\lambda) = 1\cdot 1 - (\lambda - \alpha_1)\cdot 0 = 1 > 0$$

であるから、以下帰納的に

$$q_k(\lambda) > 0 \quad (k = 2, 3, \cdots, N)$$

が示せる。特に

$$q_N(\lambda) = p'_N(\lambda)p_{N-1}(\lambda) - p_N(\lambda)p'_{N-1}(\lambda) > 0$$

であるが、$p_N(\lambda) = 0$ であるから

$$p'_N(\lambda)p_{N-1}(\lambda) > 0. \quad \blacksquare$$

符号の変化数の計算に関する注意　$p_N(a) \neq 0$ なる a に対して

$N(a) :=$ "$\{p_0(a), p_1(a), \cdots, p_N(a)\}$" の**符号の変化数**

とおく (逆順であっても符号の変化数は同じになる)。例えば

$p_0(a)$	$p_1(a)$	$p_2(a)$	$p_3(a)$	$p_4(a)$	$p_5(a)$	$p_6(a)$	$p_7(a)$	$p_8(a)$	$p_9(a)$	$p_{10}(a)$
$+$	$-$	$-$	$-$	0	$+$	$+$	$+$	$+$	$-$	$-$

では $N(a) = 3$.

この符号の変化数は (特別な注意をせずに) 数値的に安定して計算できる。つまり絶対値が非常に小さくて、符号の判別がつきにくい場合も、「符号の変化数」そのものは疑いがなく計算できる。例えば

$p_{k-1}(a)$	$p_k(a)$	$p_{k+1}(a)$
$+$	絶対値小	$-$

または

$p_{k-1}(a)$	$p_k(a)$	$p_{k+1}(a)$
$-$	絶対値小	$+$

において $p_k(a)$ が正であっても負であっても 0 であっても符号の変化数の計算にとっては影響がない。注意すべきは Strum 列の条件 (iv) から

$p_{k-1}(a)$	$p_k(a)$	$p_{k+1}(a)$
$+$	絶対値小	$+$

または

$p_{k-1}(a)$	$p_k(a)$	$p_{k+1}(a)$
$-$	絶対値小	$-$

のような場合 (もしこうなったら符号の変化数の計算がむつかしい) が起こり得ないことである。

B.4　直交多項式の作る Strum 列

(ここは単なる覚え書き。後で肉付けするかもしれない。)

$w\colon [a,b] \to \mathbf{R}$ は連続で、有限個の点で 0 になる他は正で、条件

$$\sup_{k\in\mathbf{N}} \int_a^b x^k w(x)\,dx < \infty$$

を満たすような関数とする。このとき $[a,b]$ 上の実数値連続関数全体の集合に

$$(u,v)_w := \int_a^b u(x)v(x)w(x)\,dx$$

で定義される内積を導入して、内積空間としたものを $H_w(a,b)$ とする。また $(\cdot,\cdot)_w$ に付随するノルムを $\|\cdot\|_w$ と書く:

$$\|u\|_w = \sqrt{(u,u)_w}.$$

自然数 n に対して、関数列 $\{1,x,\cdots,x^n\}$ から Gram-Schimidt の直交化法によって得られる直交多項式系を $\{p_0(x),p_1(x),\cdots,p_n(x)\}$ とする。

命題 B.9 $H_w(a,b)$ において関数列 $\{1,x,\cdots,x^n\}$ から Gram-Schimidt の直交化法によって得られる直交多項式系を $\{p_0(x),p_1(x),\cdots,p_n(x)\}$ とし、$p_n(x)$ の最高次の係数を μ_n とすると、

$$\inf\{\|q\|_w; q(x)\in\mathbf{R}[x], \deg q(x)=n, q(x) \text{ の最高次係数} = 1\} = \|p_n/\mu_n\|_w.$$

命題 B.10 $H_w(a,b)$ において関数列 $\{1,x,\cdots,x^n\}$ から Gram-Schimidt の直交化法によって得られる直交多項式系を $\{p_0(x),p_1(x),\cdots,p_n(x)\}$ とすると、$p_n(x)$ の根はすべて単根で、区間 $[a,b]$ の内部にある。

命題 B.11 $H_w(a,b)$ において関数列 $\{1,x,\cdots,x^n\}$ から Gram-Schimidt の直交化法によって得られる直交多項式系を $\{p_0(x),p_1(x),\cdots,p_n(x)\}$ とすると、次のような 3 項漸化式が存在する。

$$p_k(x) = (\alpha_k x + \beta_k)p_{k-1}(x) - \gamma_k p_{k-2}(x) \quad (k=1,2,\cdots).$$

ただし $p_{-1}(x)\equiv 0$ とし、

$$\begin{aligned}\alpha_k &= \frac{\mu_k}{\mu_{k-1}}, \quad \beta_k = -\frac{\alpha_k (xp_{k-1},p_{k-1})_w}{\lambda_{k-1}},\\ \gamma_k &= \frac{\alpha_k(xp_{k-1},p_{k-2})_w}{\lambda_{k-2}} = \frac{\mu_k\mu_{k-2}\lambda_{k-1}}{\mu_{k-1}^2\lambda_{k-2}}.\end{aligned}$$

ただし、$\lambda_k = (p_k,p_k)_w$, $\mu_k = p_k(x)$ の最高次係数。

命題 B.12 $H_w(a,b)$ において関数列 $\{1, x, \cdots, x^n\}$ から Gram-Schimidt の直交化法によって得られる直交多項式系を $\{p_0(x), p_1(x), \cdots, p_n(x)\}$ とすると (最高次の係数が正であるようにしておくと)、

$$p_n(x), p_{n-1}(x), \cdots, p_1(x), p_0(x)$$

は区間 $[a,b]$ において Strum 列をなす。

B.5 一般化された Strum 列

一松先生の本にあった Strum 列の定義の条件は、森先生の本よりも少し緩い条件であった (つまり一般化してあると言える)。明示していないけれど $p_k(\lambda)$ の連続性は仮定されている、と思う。

次の性質を持つ関数列 $\{p_k(\lambda)\}_{k=0}^n$ を **Strum 系**と言う。

(1) 各 $p_k(\lambda)$ の解は有限個; $p_k(\lambda_0) = p_{k+1}(\lambda_0) = 0$ はありえない。

(2) $p_k(\lambda_0) = 0$ のとき $p_{k-1}(\lambda_0)p_{k+1}(\lambda_0) < 0$.

(3) $p_0(\lambda)$ は定符号。

もちろん対応する Strum の定理の結論は少し複雑になる。

定理 B.13 (一般化された Strum の定理) $\{p_k(\lambda)\}_{k=0}^n$ を Strum 系とする。$p_n(a)p_n(b) \neq 0$ $(a < b)$ のとき、

$$N(a) - N(b) = \sum_{p_n(\lambda_0)=0} \chi(\lambda_0).$$

ただし $\chi(\lambda_0)$ は、λ を λ_0- から λ_0+ に変化させるときの $p_n(\lambda)/p_{n-1}(\lambda)$ の符号変化を表す量で

$$\chi(\lambda_0) := \begin{cases} 1 & (- \text{ から } + \text{ に変化}) \\ 0 & (\text{符号の変化なし}) \\ -1 & (+ \text{ から } - \text{ に変化}). \end{cases}$$

証明 略 ■

系 B.14 $N(a) \neq N(B)$ ならば $[a,b]$ 内に $p_n(\lambda) = 0$ の解がある。

系 B.15 (実対称三重対角行列の固有値問題) n 次実対称三重対角行列

$$T = \begin{pmatrix} a_1 & b_1 & 0 & & \mbox{\Large 0} \\ b_1 & a_2 & b_2 & & \\ & \ddots & \ddots & \ddots & \\ & & b_{n-2} & a_{n-1} & b_{n-1} \\ \mbox{\Large 0} & & 0 & b_{n-1} & a_n \end{pmatrix}, \quad b_k \neq 0 \quad (k = 1, 2, \cdots, n-1)$$

から $p_k(\lambda) = \det(\lambda I_k - T_k)$ (T_k は T の第 k 次首座行列) として作った Strum 列 $p_n(\lambda)$, $p_{n-1}(\lambda)$, $\cdots$, $p_1(\lambda)$, $p_0(\lambda) \equiv 1$ について、n 個の固有値はすべて実単根であり、各固有値 λ_0 において、つねに $\chi(\lambda_0) = 1$ である。ゆえに $p_n(a)p_n(b) \neq 0$ となる $a < b$ に対して

$$N(a) - N(b) = [a, b] \text{ 内の } p_N(\lambda) = 0 \text{ の根の個数} = [a, b] \text{ 内の } T \text{ の固有値の個数}.$$

そして $p_k(\lambda)$ の隣接した根の中間に、必ず $p_{k-1}(\lambda) = 0$ の根がある $(k = 1, 2, \cdots, n)$。

証明 略。■

参考文献

[1] F. Chatelin. *Valeurs propres de matrices*. Masson, Paris, 1988. (邦訳) F. シャトラン著, 伊理 正夫, 伊理由美 訳, 行列の固有値問題, シュプリンガー・フェアラーク東京 (1993).

[2] 一松信(ひとつまつしん). 数値解析. 朝倉書店, 1982/10.

[3] 一松信. 代数学入門第一〜三課. 近代科学社, 1992, 1992, 1994.

[4] 森正武(まさたけ). 数値解析 第 2 版. 共立出版, 2002/2/25. 第 1 版は 1973 年に出版された。

[5] 森正武, 杉原正顯(まさあき), 室田一雄(むろた かずお). 線形計算. 岩波講座 応用数学. 岩波書店, 1994.

[6] 戸川隼人(はやと). マトリクスの数値計算. オーム社, 1971.

[7] 福澤誠人. 応用数理実験課題レポート 実対称行列の三重対角化. http://nalab.mind.meiji.ac.jp/~mk/labo/text/members/20030904fukuzawa-report.pdf, 2003.

[8] 高木貞治. 代数学講義 改訂新版. 共立出版, 1965.

[9] 名取亮. 線形計算. 朝倉書店, 1993.