Mathematik für InformatikerInnen 2

Frank-Olaf Schreyer

Universität des Saarlandes, SS 2020

Numerische Hauptachsentransformation

Die Themen heute sind

- QR-Verfahren
- Hilbert-Matrizen

Heute werden wir einen Algorithmus besprechen, der es erlaubt, symmetrische Matrizen numerisch zu diagonalisieren.

Sei $A \in \mathbb{R}^{n \times n}$ eine symmetrische Matrix. Die Algorithmen, die wir bisher haben, erlauben es, wenn wir einen Eigenvektor v_1 zu einem Eigenvektor λ_1 von A gefunden haben,

$$W = v_1^{\perp} \text{ und } W \xrightarrow{A'} W$$

zu berechnen und den Algorithmus dann rekursiv mit A' aufzurufen.

Um λ_1 zu bestimmen, haben wir Nullstellen des charakteristischen Polynoms berechnet. Ein andere Option ist Vektoriteration zu verwenden, um einen normierten Eigenvektor v_1 zu bekommen.

Heute werden wir den QR-Algorithmus besprechen, der simultan alle Eigenwerte und die Diagonalisierung numerisch approximativ berechnet.

QR-Verfahren

Sei $A \in \mathbb{R}^{n \times n}$ eine symmetrische Matrix. Wir berechnen rekursiv eine Folge (A_k) von symmetrischen Matrizen wie folgt:

1. $A_1 = A$.
2. Ist A_k gegeben, dann betrachten wir

$$A_k = Q_k R_k$$

die QR-Zerlegung von A_k, in der R_k positive Diagonaleintäge hat, und setzen

$$A_{k+1} = R_k Q_k.$$

Bemerkung. Wegen

$$A_{k+1} = R_k Q_k = Q_k^t A_k Q_k$$

sind sämtliche Matrizen A_k zu A vermöge orthogonaler Matrizen konjugiert. Es gilt

$$A_{k+1} = P_k^t A P_k$$

für

$$P_k = Q_1 Q_2 \cdot \ldots \cdot Q_k \in \mathrm{O}(n).$$

QR-Verfahren

Satz. *Sei $A \in \mathbb{R}^{n\times n}$ eine symmetrische Matrix mit Eigenwerten $\lambda_1, \ldots, \lambda_n \in \mathbb{R}$ mit*

$$|\lambda_1| > |\lambda_2| > \ldots > |\lambda_n| > 0.$$

Dann konvergiert die Folge (A_k) gegen eine Diagonalmatrix mit Einträgen $\lambda_1, \ldots, \lambda_n$ (in der Regel in dieser Reihenfolge). Ist dies die Reihenfolge, dann gilt für die Nicht-Diagonalelemente von $A_k = (a_{ij}^{(k)})$, dass die Konvergenzgeschwindigkeit gegen Null von der Ordnung

$$a_{ji}^{(k)} = a_{ij}^{(k)} \in O(|\frac{\lambda_i}{\lambda_j}|^k) \text{ für } i > j$$

ist.

Bemerkung. Man kann zeigen, dass bei mehrfachen Eigenwerten, etwa $\lambda_j = \lambda_{j+1}$, ebenfalls noch Konvergenz gegen eine Diagonalmatrix vorliegt. Gilt $\lambda_j = -\lambda_{j+1}$, so bleibt eventuell ein 2×2-Block stehen.

Ende Teil 1

Beweis. Wir zeigen zunächst für die k-te Potenz von A

$$A^k = Q_1 \cdot \ldots \cdot Q_k \cdot R_k \cdot \ldots \cdot R_1 = P_k \cdot U_k$$

mit Induktion nach k.
Der Induktionsanfang $k = 1$ ist klar: $A = Q_1 R_1$ ist die QR-Zerlegung von A.
Induktionsschritt $k \to k + 1$:

$$A_{k+1} = Q_{k+1} R_{k+1} = R_k Q_k = Q_k^t A_k Q_k = P_k^t A P_k.$$

Für die Potenzen ergibt sich

$$\begin{aligned} A^{k+1} &= A A^k \overset{I.V.}{=} A P_k U_k = P_k P_k^t A P_k U_k \\ &= P_k Q_{k+1} R_{k+1} U_k = P_{k+1} U_{k+1}. \end{aligned}$$

Da $P_k \in O(n)$ orthogonal ist und U_k eine obere Dreiecksmatrix ist, ist also

$$A^k = P_k U_k$$

die QR-Zerlegung von A^k.

Sei

$$D = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix} = SAS^t$$

die Hauptachsentransformation von A mit einer orthogonalen Transformationsmatrix $S \in \mathrm{O}(n)$. Dann gilt für die Potenzen

$$A^k = S^t D^k S, \text{ da } A = S^t DS.$$

Wir nehmen nun an, dass $S = L \cdot R$ eine LR-Zerlegung ohne Permutationsmatrix hat.
Dabei verlangen wir nicht, dass die untere Dreiecksmatrix $L = (\ell_{ij})$ Einträge $|\ell_{ij}| \leqslant 1$ für $i > j$ hat.
S hat keine solche Zerlegung genau dann, wenn im Gauß-Algorithmus im i-ten Eliminationsschritt das Diagonalelement Null ist.
Dies ist also nur selten der Fall. (Braucht man eine Permutationsmatrix wirklich, so müssen die entsprechenden Diagonalelemente von D vertauscht werden.)

Es sei also

$$S = LR$$

mit L eine untere Dreiecksmatix und R eine obere Dreiecksmatrix. Dann gilt

$$A^k = S^t D^k LR = S^t (D^k L D^{-k}) D^k R$$

Da $L = (\ell_{ij})$ eine untere Dreiecksmatrix ist, gilt

$$(D^k L D^{-k})_{ij} = \ell_{ij} (\frac{\lambda_i}{\lambda_j})^k \text{ für } i > j$$

und somit

$$D^k L D^{-k} = E + F_k$$

mit $F_k \underset{k \to \infty}{\longrightarrow} 0$. Eingesetzt liefert dies

$$A^k = S(E + F_k) D^k R.$$

Sei nun

$$E + F_k = \tilde{Q}_k \tilde{R}_k$$

die QR-Zerlegung von $E + F_k$, in der $\tilde{R}_k$ positive Diagonaleinträge hat. Dann gilt $\tilde{Q}_k \underset{k \to \infty}{\longrightarrow} E$ und $\tilde{R}_k \underset{k \to \infty}{\longrightarrow} E$.

Weil

$$A^k = S^t \tilde{Q}_k \tilde{R}_k D^k R = P_k U_k$$

gilt, folgt aus der Eindeutigkeit der QR-Zerlegung, dass

$$P_k = S^t \tilde{Q}_k \Sigma_k \text{ und } U_k = \Sigma_k \tilde{R}_k D^k R$$

gilt, wobei Σ_k eine Diagonalmatrix mit Einträgen $(\operatorname{sgn} \lambda_j)^k$ ist. Es folgt

$$Q_k = P_{k-1}^t P_k = (\Sigma_{k-1} \tilde{Q}_{k-1}^t S)(S^t \tilde{Q}_k \Sigma_k) \underset{k\to\infty}{\longrightarrow} \Sigma_{k-1}\Sigma_k = \Sigma_1$$

und

$$\begin{aligned} R_k &= U_k U_{k-1}^{-1} = (\Sigma_k \tilde{R}_k D^k R)(R^{-1} D^{-(k-1)} \tilde{R}_{k-1}^{-1} \Sigma_{k-1}^{-1}) \\ &= \Sigma_k \tilde{R}_k D \tilde{R}_{k-1}^{-1} \Sigma_{k-1} \underset{k\to\infty}{\longrightarrow} \Sigma_k D \Sigma_{k-1} = \Sigma_1 D. \end{aligned}$$

Also

$$A_k = Q_k R_k \underset{k\to\infty}{\longrightarrow} \Sigma_1 \Sigma_1 D = D$$

$\square$

Ende Teil 2

Beispiel. Im folgenden ist A eine ganzzahlige symmetrische Matrix, die ich mir durch Wahl zufälliger Einträge in $[-99, 99] \cap \mathbb{Z}$ verschafft habe.

$$A = \begin{pmatrix} 24 & -89 & -67 & -4 & -87 & -60 \\ -89 & 8 & -7 & -64 & 25 & -44 \\ -67 & -7 & -6 & -8 & -42 & 20 \\ -4 & -64 & -8 & -82 & -63 & -10 \\ -87 & 25 & -42 & -63 & 64 & 50 \\ -60 & -44 & 20 & -10 & 50 & 64 \end{pmatrix}$$

Dann ergibt sich

$$A_{100} = \begin{pmatrix} 209.067 & -.00000490305 & 0 & 0 & 0 & 0 \\ -.00000490305 & -174.62 & 0 & 0 & 0 & 0 \\ 0 & 0 & 93.7371 & 0 & 0 & 0 \\ 0 & 0 & 0 & -64.8425 & .189336 & 0 \\ 0 & 0 & 0 & .189336 & 61.5915 & -.00000237983 \\ 0 & 0 & 0 & 0 & -.00000237983 & -52.9337 \end{pmatrix},$$

wobei alle Einträge vom Betrag $< 10^{-6}$ durch Null ersetzt wurden

$$A_{200} = \begin{pmatrix} 209.067 & 0 & 0 & 0 & 0 & 0 \\ 0 & -174.62 & 0 & 0 & 0 & 0 \\ 0 & 0 & 93.7371 & 0 & 0 & 0 \\ 0 & 0 & 0 & -64.8428 & .00110483 & 0 \\ 0 & 0 & 0 & .00110483 & 61.5918 & 0 \\ 0 & 0 & 0 & 0 & 0 & -52.9337 \end{pmatrix}.$$

Die Eigenwerte von A sind

$$\{209.067, -174.62, 93.7371, -64.8428, 61.5918, -52.9337\}.$$

Der kleinste Quotient $|\frac{\lambda_j}{\lambda_{j-1}}|$ ist $q = \frac{61.5918}{64.8428} \approx 0.95$. Um die Diagonalmatrix mit einem Fehler in der Größenordnung von $< 10^{-9}$ zu bestimmen, brauchen wir wenigstens

$$k = 9 \log 10 / (-\log q) \approx 403$$

Iterationen.

$$A_{403} = \begin{pmatrix} 209.067 & 0 & 0 & 0 & 0 & 0 \\ 0 & -174.62 & 0 & 0 & 0 & 0 \\ 0 & 0 & 93.7371 & 0 & 0 & 0 \\ 0 & 0 & 0 & -64.8428 & 3.22404 \cdot 10^{-8} & 0 \\ 0 & 0 & 0 & 3.22404 \cdot 10^{-8} & 61.5918 & 0 \\ 0 & 0 & 0 & 0 & 0 & -52.9337 \end{pmatrix}$$

Um den Fehler wirklich unter 10^{-9} zu drücken braucht man weitere $\log 32.24/(-\log 0.95) \approx 68$ Iterationen.

Wir können nun auch numerisch den Rang einer Matrix bestimmen. Im folgenden habe ich eine 300×300 ganzzahlige Matrix vom erwarteten Rang 280

$$A = B \cdot C$$

konstruiert, indem ich zufällige ganzzahlige Matrizen $B \in \mathbb{Z}^{300 \times 280}$ und $C \in \mathbb{Z}^{280 \times 300}$ mit Einträgen $[-9, 9] \cap \mathbb{Z}$ gewählt habe. Für die Berechnung der Singulärwertzerlegung

$$A = U \Sigma V$$

hat mein Computer 0.12 Sekunden gebraucht. Die Singulärwerte σ_k für $k = 280, 281$ sind

$$70.0135,\ 4.04966 \cdot 10^{-12}$$

mit einem dramatischen Abfall um den Faktor $5.78412 \cdot 10^{-14}$.

Alle anderen Quotienten $\frac{\sigma_{k+1}}{\sigma_k}$ für $k < 280$ sind > 0.7. Die Singulärwerte von unserem speziellen A fallen langsam ab von

$$22585.8, \ldots, 70.0135$$

mit einem durchschnittlichen Quotient von

$$\left(\frac{\sigma_{280}}{\sigma_1}\right)^{\frac{1}{279}} = .979862.$$

Die Quotienten der Rundungsfehlersingulärwerte $\sigma_{281}, \ldots, \sigma_{300}$ streuen etwas mehr.

Ende Teil 3

Hilbert-Matrizen

Definition. Die Matrix

$$H_n = \begin{pmatrix} \frac{1}{1} & \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots & \frac{1}{n+1} \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots & \frac{1}{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \frac{1}{n+2} & \cdots & \frac{1}{2n-1} \end{pmatrix}$$

heißt n-te Hilbert-Matrix.

Satz. *H_n ist positiv definit.*

Beweis. Wir betrachten den Isomorphimus

$$\varphi\colon \mathbb{R}^n \to \mathbb{R}[t]_{\leqslant n-1}, \quad e_i \mapsto t^{i-1}$$

und das Skalarprodukt

$$\langle f, g\rangle = \int_0^1 f(t)g(t)dt$$

auf $\mathbb{R}[t]_{\leqslant n-1} \subset \mathbb{R}[t] \subset C^0[0,1]$. Es gilt:

$$\langle t^{i-1}, t^{j-1}\rangle = \int_0^1 t^{i+j-2}dt = \left[\frac{1}{i+j-1}t^{i+j-1}\right]_0^1 = \frac{1}{i+j-1}$$

Es folgt

$$H_n = (\langle\varphi(e_i), \varphi(e_j)\rangle)$$

und

$$x^t H_n x = \int_0^1 |\sum_{k=1}^n x_k t^{k-1}|^2 dt > 0 \quad \forall x \in \mathbb{R}^n \backslash \{0\}$$

$\square$

Die Konditionen von Hilbert-Matrizen sind sehr schlecht. Sie wachsen exponentiell mit n:

n	2	3	4	5	6	7
$\kappa(H_n)$	$1.92 \cdot 10$	$5.24 \cdot 10^2$	$1.55 \cdot 10^4$	$4.77 \cdot 10^5$	$1.49 \cdot 10^6$	$4.75 \cdot 10^8$

Hilbert-Matrizen werden häufiger zum Testen von numerischen Algorithmen verwendet. Da die Inversen der Hilbert-Matrizen ganzzahlig sind,

$$\begin{pmatrix} 4 & -6 \\ -6 & 12 \end{pmatrix}, \begin{pmatrix} 9 & -36 & 30 \\ -36 & 192 & -180 \\ 30 & -180 & 180 \end{pmatrix}, \begin{pmatrix} 16 & -120 & 240 & -140 \\ -120 & 1200 & -2700 & 1680 \\ 240 & -2700 & 6480 & -4200 \\ -140 & 1680 & -4200 & 2800 \end{pmatrix},$$

kann man Rundungsfehler gut entdecken. Die allgemeine Formel für die Einträge der Inversen lautet:

$$(H_n^{-1})_{ij} = (-1)^{i+j}(i+j-1)\binom{n+i-1}{n-j}\binom{n+j-1}{n-i}\binom{i+j-2}{i-1}^2.$$

Legendre-Polynome

Vermöge des Skalarprodukts

$$\langle f, g\rangle = \frac{1}{2}\int_{-1}^{1} f(t)g(t)dt$$

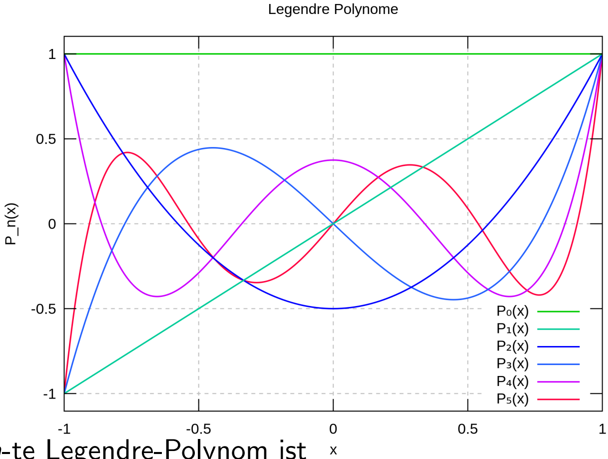
auf $\mathbb{R}[t]$ erhält man aus dem System linear-unabhängiger Vektoren $\{1, t, t^2, t^3, \ldots\}$ mit dem Gram-Schmidt-Verfahren ein Orthogonalsystem von Polynomen

$$\begin{aligned}
P_0(t) &= 1 & P_1(t) &= t \\
P_2(t) &= \tfrac{1}{2}(3t^2 - 1) & P_3(t) &= \tfrac{1}{2}(5t^3 - 3t) \\
P_4(t) &= \tfrac{1}{8}(35t^4 - 30t^2 + 3) & P_5(t) &= \tfrac{1}{8}(63t^5 - 70t^3 + 15t) \\
&\vdots & &\vdots
\end{aligned} .$$

vermöge

$$\tilde{P}_k(t) = t^k - \sum_{\ell=0}^{k-1} \frac{\langle t^k, P_\ell(t)\rangle}{\langle P_\ell(t), P_\ell(t)\rangle} P_\ell(t) \text{ und } P_k(t) = \frac{\tilde{P}_k(t)}{\tilde{P}_k(1)}.$$

Um Wurzeln zu vermeiden, haben wir hierbei auf die Normierung verzichtet. Stattdessen verlangen wir $P_k(1) = 1$; $\|P_k(t)\|^2 = \frac{1}{2k+1}$.



Das n-te Legendre-Polynom ist

$$P_n(x) = \sum_{k=0}^{\lfloor \frac{n}{2} \rfloor} (-1)^k \frac{(2n-2k)!}{(n-k)!(n-2k)!k!2^n} x^{n-2k}.$$

Dass sich dabei die geraden und ungeraden Polynome nicht vermischen, liegt daran, dass bzgl. des Skalarprodukts

$$\langle f, g\rangle = \frac{1}{2}\int_{-1}^{1} f(t)g(t)dt$$

die Gram-Matrizen sich zerlegen. Z.B. für $\mathbb{R}[t]_{\leqslant 7}$ ist

$$(\langle t^k, t^\ell\rangle)_{\substack{k=0,\dots,7\\ \ell=0,\dots,7}} = \begin{pmatrix} 1 & 0 & \frac{1}{3} & 0 & \frac{1}{5} & 0 & \frac{1}{7} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{5} & 0 & \frac{1}{7} & 0 & \frac{1}{9} \\ \frac{1}{3} & 0 & \frac{1}{5} & 0 & \frac{1}{7} & 0 & \frac{1}{9} & 0 \\ 0 & \frac{1}{5} & 0 & \frac{1}{7} & 0 & \frac{1}{9} & 0 & \frac{1}{11} \\ \frac{1}{5} & 0 & \frac{1}{7} & 0 & \frac{1}{9} & 0 & \frac{1}{11} & 0 \\ 0 & \frac{1}{7} & 0 & \frac{1}{9} & 0 & \frac{1}{11} & 0 & \frac{1}{13} \\ \frac{1}{7} & 0 & \frac{1}{9} & 0 & \frac{1}{11} & 0 & \frac{1}{13} & 0 \\ 0 & \frac{1}{9} & 0 & \frac{1}{11} & 0 & \frac{1}{13} & 0 & \frac{1}{15} \end{pmatrix}.$$

Es folgt:

$$\mathbb{R}[t] = \mathbb{R}[t^2] \oplus t\mathbb{R}[t^2]$$

ist eine orthogonale direkte Summe.

Ende Teil 4