EW & EV // λ nur in 1 diag

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

EW wenn $\chi_A = 0$

EV wenn $(A - \lambda_1 I)\cdot v = 0$

Norm & Skalarprodukt

$\langle \vec a, \vec b\rangle = |\vec a||\vec b|\cos(\alpha)$

$\|x+y\| \le \|x\| + \|y\|$

$|\langle a,b\rangle| \le \|a\|\cdot\|b\|$

Matrixstuff

$(A+B)^T = A^T + B^T$

$(AB)^T = B^T A^T$

"unitär" $\Leftrightarrow U^H U = I$

"Symmetrisch" $\Leftrightarrow A^T A = A A^T$

"Orthogonal" $\Rightarrow A^T = A^{-1}$

$\Rightarrow A^T A = A A^T = I$ // Längen und Winkel treu

Symm. + Symm. => Symm.

A, B symm && $AB = BA \Rightarrow AB$ symm

B "kongruent" zu $A \Leftrightarrow \exists P : B = P^T A P$

A "orthogonal" falls Spalten/Zeilen orthonormal

$\Rightarrow \|EW(\text{Ortho})\| = 1$

$\Rightarrow \|\det(\text{Ortho})\| = 1$ (nicht ⇐)

LR-Zerlegung

Existiert nur mit regulären Matrizen (quadratisch)

auf einer Seite + heisst auf d. anderen −

aber scaling ist gleich

System lösen wo $Ax = B$:

1. $Lc = b$ 2. $Rx = c$

denn $Lc = LRx = Ax = b$

[L zahlen Links unten]

[R zahlen rechts oben]

oder: $Lc = Pb$, $Rx = c$, $PA = LR$

Nur Submatrix mit Pivot verändern! Links davon nicht.

$\binom{1}{1}$ links notieren!

Vorteil von LR: für viele b nur ein mal berechnen

Angeblich gilt $A = (v_1\, v_2)^{-1} \begin{pmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{pmatrix}$

"Kommt λ_1 n-mal vor in χ, gibt es (n−rang) linear unabhängige Eigenvektoren"

A "orthogonal" $\Rightarrow \exists U : U^{-1} A U = D$

wobei U Spalten = EV(A), U "unitär"

und D = diagonal EW(A)

Symm => reelle EW

$A^H A$ = Symmetrisch

Matrix als lineare Funktion

$F(x) = Ax$, $G(x) = Bx \Rightarrow F(G(x)) = ABx$

$\ker = \{0\} \Leftrightarrow$ eindeutige Lsg $\Leftrightarrow Ax = 0$ hat nur 0 als Lsg.

$\operatorname{rank} A = \dim \operatorname{im} A$

$\operatorname{rank} BA \le \min(\operatorname{rank} A, \operatorname{rank} B)$

$\dim\ker + \dim\operatorname{im} = \dim(\text{Urbild})$

in Zeilenstufenform sind Zeilen lin. unab.

regulär, bijektiv, invertierbar, vollen Rang, quadratisch, nichtsingulär, linear unabhängig, $\det \ne 0$, alle EW $\ne 0$

Permutationsmatrix

$P \cdot v = \begin{pmatrix}0&0&1\\1&0&0\\0&1&0\end{pmatrix} \cdot v = \begin{pmatrix}v_3\\v_1\\v_2\end{pmatrix}$

Kern

stellt b auf lin. unabhängigen Vektoren falls existent in R^3

Inverse symm

$A^H A \Leftrightarrow \langle Ay, x\rangle = \langle y, Ax\rangle$

reelle EW

"Hermitesch" $A^H = A$

"Schiefsymm." $A^T = -A$

Rank = # Zeilen ≠ 0 in Zeilenstufenform

Inverse berechnen

sign(a) = Vorzeichen von a

auf einer Seite + heisst auf der anderen auch +

Bei orthogonaler Matrix: $A^{-1} = A^T$

λ^{-1} = EW von A^{-1} wenn λ EW von A

$EV(A) = EV(A^{-1})$ $\left[Av = \lambda v \; ; \; A^{-1} v = \frac{1}{\lambda} v \right]$

Determinante

$\ne 0$ heisst voller Rang

1 Zeilentausch => $\det \cdot (-1)$

Skalierung einer Zeile/Spalte um a => $\det \cdot a$

Addition (a· Zeile) auf Zeile => bleibt gleich

$\det(AB) = \det(A)\det(B) = \det(BA)$

$\det(A^T) = \det(A)$

$\det(A^{-1}) = \det(A)$

$\det(A^k) = \det(A)^k$

$\det(C^{-1}AC) = \det(A)$

$\det(k\cdot A) = k^n \cdot \det(A)$

Für $A = \begin{pmatrix}B & 0\\ 0 & C\end{pmatrix}$ gilt $\det(A) = \det(B)\cdot\det(C)$

Householder-Matrizen

$H = I - 2vv^T$ ← Normalenvektor auf Ebene

Spiegelt an Hyperebene

$H = H^T = H^{-1}$ $H^2 = I$

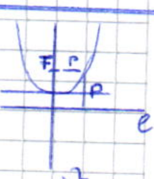
Hat EW (1), (−1) da H orthogonal ist

(n−1) mal, 1 mal

LR mit Pivotisierung

L nur Zeilen unter diago. vertauschen, immer max(in Spalte) als Pivot

Kegelschnitte



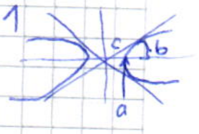
Parabel: $x^2 = 2py$, $\varepsilon = 1$

Ellipse: $\frac{(x-x_0)^2}{a^2} + \frac{(y-y_0)^2}{b^2} = 1$

Abstand d. Brennpunkte vom Mittelpunkt $\varepsilon = \sqrt{a^2 - b^2}$

Kreis: $\varepsilon = 0$

Hyperbel: $\frac{x^2}{a^2} - \frac{y^2}{b^2} = 1$



Scheitelp. ± a

Asymptot: $y = \frac{\pm b}{a} x + q$

surjektiv, injektiv

Für jedes Y min. ein X => $F: X \mapsto Y$ surjektiv

Für jedes Y max. ein X => $F: X \mapsto Y$ injektiv

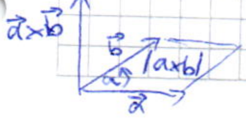
$F = Ax$

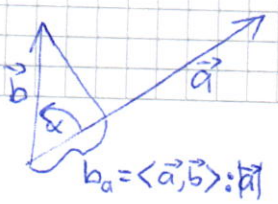
A vollen Zeilenrang => surjektiv

A vollen Spaltenrang => injektiv

Kreuzprod.

$|\vec a \times \vec b| = |a||b|\sin(\alpha)$





$b_a = \langle \vec a, \vec b\rangle : |\vec a|$

Trägheit

$(p, r-p, n-r)$ heisst Trägheit und ist für alle kongruenten gleich

Basen

Ist B eine Basis?

Sei $B = \{p_0, p_1, p_2, p_3\}$ und $B' = \{q_0, q_1, q_2, q_3\}$

$p_0(t) = 1$, $p_1(t) = t$, $p_2(t) = t^2$, $p_3(t) = t^3$

$q_0(t) = t-1$, $q_1(t) = t+1$, $q_2(t) = (t-1)^3$, $q_3(t) = (t+1)^3$

B hat 4 Lin. unabhängige Elemente in P_3.
Dim $P_3 = 4 \Rightarrow$ span $B = P_3 \Rightarrow B$ ist Basis von P_3.

Ist B' eine Basis?

Wenn sich ein Polynom $p_B = a_0 p_0 + a_1 p_1 + a_2 p_2 + a_3 p_3$ als $p_{B'} = b_0 q_0 + \ldots$ darstellen lässt.

B' durch B ausdrücken: $\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = \underbrace{\begin{pmatrix} -1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ -1 & 3 & -3 & 1 \\ 1 & 3 & 3 & 1 \end{pmatrix}}_{A} \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$

Aus $p_B = (a_0 \ a_1 \ a_2 \ a_3) \begin{pmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{pmatrix}$ folgt (p-Vektor ersetzen) $p_B = (a_0 \ a_1 \ a_2 \ a_3) A^{-1} \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = (b_0 \ b_1 \ b_2 \ b_3) \begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = p_{B'}$ □

$\Rightarrow$ Wenn $A_{B \to B'}$ invertierbar ist, ist B' eine Basis.

Funktional (als Matrix) bzgl. Basenpaar (A, B) darstellen.

1. $f: A \mapsto B$ $\quad f(p_A)(t) = p_A(t+1) - p_A(t)$
   für p_{A_0} bis p_{A_n} berechnen: $f(p_{A_0}) = 0$, $f(p_{A_1}) = 1$ ...
2. Ergebnisse in B ausdrücken: $f(p_{A_0}) = 0 = 0$, $f(p_{A_1}) = 1 = p_{B_0}$ ...
3. Matrix bilden sodass $f = aCp_B$ von A auf B führt
   $f(a_0 p_{A_0} + a_1 p_{A_1} + a_2 p_{A_2} + a_3 p_{A_3}) = (a_0 \ a_1 \ a_2 \ a_3) \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 2 & 0 & 0 \\ 1 & 3 & 3 & 0 \end{pmatrix} \begin{pmatrix} p_{B_0} \\ p_{B_1} \\ p_{B_2} \\ p_{B_3} \end{pmatrix}$
4. evt ausrechnen $\Rightarrow f(p_A) = a_1 p_0 + a_2 p_0 + a_2 2 p_1 + a_3 p_0 + 3 a_3 p_1 + 3 a_3 p_2$

// Bsp von B' nach B von oben

Alternativ Basis A als Matrix abhängig von existenter basis aus drücken: $f() = (\ldots)(p)$. Um polyn. von $4 \to 6$ verwandeln Matrix senaten

Generell

$B \cdot v_B = A \cdot v_A = V = A \cdot T_{B \to A} v_B$

: meistens B ortho. $= B^{-1} = B^T$

Finde A, die $f\left(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}\right)$ in Standardbasis darstellt

$f(x) = Ax = \left((f(e_1)), (f(e_2)), (f(e_3)) \right) \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$

Basentransformation $B_e \to B_1$

$B_1 = \left\{ \begin{pmatrix} a \\ b \\ c \end{pmatrix}, \begin{pmatrix} a_2 \\ b_2 \\ c_2 \end{pmatrix}, \begin{pmatrix} a_3 \\ b_3 \\ c_3 \end{pmatrix} \right\} \Rightarrow T_{B_e \to B_1} = \begin{pmatrix} a & a_2 & a_3 \\ b & b_2 & b_3 \\ c & c_2 & c_3 \end{pmatrix}$

Koordinatentransformation $\xi_A \to \xi_B$

Option 1: $T_{\xi_A \to \xi_B} = T^{-1}_{B_A - B_B}$

Option 2: $a_1 = u b_1 + v b_2 + w b_3 = \begin{pmatrix} u \\ v \\ w \end{pmatrix}$
$a_2 = \ldots$
$a_3 = \ldots$

$\Rightarrow v_B = T_{\xi_A \to \xi_B} \cdot v_A = \begin{pmatrix} u & u_2 & u_3 \\ v & v_2 & v_3 \\ w & w_2 & w_3 \end{pmatrix} \cdot v_A$

$f(v_b) = A v_b$
$f(v_c) = A T_{\xi_b \to c} v_c$
äquivalent

möglich iff A hermitesch ($A = A^T$)
möglich wenn positiv Definit

Cholesky LR ohne Zeilentausch

$\Rightarrow D = \text{diag}(R) \Rightarrow A = LDL^T$
falls D positiv $\Rightarrow D^{\frac{1}{2}} = \sqrt{\ }$ der diagelemente $\Rightarrow \tilde{R} = D^{\frac{1}{2}} L^T$
$\Rightarrow A = \tilde{R}^T \tilde{R}$

Basentransformation through Projektion

$\left\langle \begin{pmatrix} 7 \\ 5 \\ 3 \end{pmatrix}, \text{Basis} \right\rangle$ Projiziert den Vektor auf die Basis. Wenn Basis normalisiert ist, gilt es keine Längenverzerrung

$\Rightarrow \begin{pmatrix} \langle v, b_x \rangle \\ \langle v, b_y \rangle \\ \langle v, b_z \rangle \end{pmatrix} = V_b$

Transponieren

$\begin{pmatrix} a & b \\ c & d \\ e & f \end{pmatrix}^T = \begin{pmatrix} a & c & e \\ b & d & f \end{pmatrix}$

Komplexes Skalarprod.

$\langle av, bu \rangle = \bar{a} \langle v, bu \rangle$

komplex   "normal"

Linearität

$f(ax) = af(x)$; $f(x+y) = f(x) + f(y)$

zusammen: $f(ax+y) = af(x) + y$

⚠ beim normieren: Skalare ganz ignorieren: $\mathrm{norm}\left(\frac{1}{3}\begin{pmatrix}2\\1\end{pmatrix}\right) = \frac{1}{\cancel{3}\sqrt{6}}\begin{pmatrix}2\\1\end{pmatrix}$

Gram-Schmidt Version Skript — orthonormiert

$$b_1 = \frac{a_1}{\|a_1\|}$$

$$\tilde{b}_k = a_k - \sum_{j=1}^{k-1} \langle b_j, a_k \rangle b_j$$

$$= a_k - \langle b_1, a_k \rangle b_1 - \langle b_2, a_k \rangle b_2 - \dots$$

$$b_k = \frac{\tilde{b}_k}{\|\tilde{b}_k\|}$$

Version Wikipedia — orthogonalisiert

$$V_1 = b_1$$

$$V_2 = b_2 - \frac{\langle V_1, b_2 \rangle}{\langle V_1, V_1 \rangle} V_1$$

$$V_3 = b_3 - \frac{\langle V_1, b_3 \rangle}{\langle V_1, V_1 \rangle} V_1 - \frac{\langle V_2, b_3 \rangle}{\langle V_2, V_2 \rangle} V_2$$

⚠ Falls Skalarprod. neu definiert ist Betrag $\sqrt{\langle x, x \rangle}$

QR-Zerlegung

Q: orthogonal
R: obere Dreiecksmatrix.

1. mit Gram-Schmidt orthonormierung aus Spalten von A die Spalten von Q erstellen.
2. $R = \begin{pmatrix} \langle b_1, a_1 \rangle & \langle b_1, a_2 \rangle & \langle b_1, a_3 \rangle \\ 0 & \langle b_2, a_2 \rangle & \langle b_2, a_3 \rangle \\ 0 & 0 & \langle b_3, a_3 \rangle \end{pmatrix}$
3. Falls zum kleinste Quadrate lösen // $Rx = Q^H y$ für $Ax = y$
   i) $Q^H y$ bestimmen // $Q^{-1} = Q^H$
   ii) x bestimmen
4. Residuenvektor bestimmen: q_j = Spalten von Q vor dem Normieren
   $r = y - \sum_{j=1}^{n} q_j \langle q_j, y \rangle$ //also nur orthogonale Spalten, nicht normal
   $= y - QQ^H y$ // $A = m \times n = \downarrow_m \overset{n}{\rightarrow}$
   // QQ^H nicht $= I$ weil vor dem Normieren

Wenn Skalarprodukt neu definiert als $\langle x, y \rangle = xWy$
gilt statt $Q^T Q = I$ $\boxed{Q^T W Q = I}$
Herleitung: $Ax = b \iff QRx = b \iff \underbrace{Q^T W Q}_{I} Rx = Q^T W b \iff Rx = Q^T W b$
oder: Residuenvektor rechnen, neues SP benutzen, $A = QR$ und $Ax = y - r$
$\Rightarrow QRx = y - (y - QQ^T Wy) \iff QRx = QQ^T Wy \iff Rx = Q^T Wy$

Normalengleichung

Anwendbar wenn A lin. abh. Spalten hat und Rang von A ≤ Höhe.

ungenauer bei grossen Matrizen, auch für Least squares

$$Ax = A(A^H A)^{-1} A^H b$$

$$\boxed{A^H A x = A^H y}$$

1. so ausrechnen dann $\left(\frac{m}{n}\right)x = \left(\frac{m}{n}\right)y$ lösen
2. profit

3. Diagonalisieren $\Rightarrow D = (\dots)$, $U = (v_1)(v_2)(v_3)$ normalisierte EV
4. neue koord y: $y^T D y + (d\ e) U \begin{pmatrix} d \\ e \end{pmatrix} + f$

Als Q(A) notieren, mit quadratisch ergänzen Binome bilden, diese durch $\tilde{x}$ und $\tilde{y}$ ersetzen

Projektion

$P^2 = P$ = Bild bleibt bei wiederholten Anwenden gleich

Orthogonalprojektion: $\ker P \perp \operatorname{im} P$

$P(y - Py) = Py - P^2 y = Py - Py = 0$

Äquivalenzen

2×2 Invertieren

$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$

$A^{-1} = \frac{1}{\det(A)} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$

Singulärwertzerlegung siehe Karton

$A^H A$ ist hermitesch & positiv semidefinit

$A^H A V = V \Sigma_n^2$ ← unitäre n×n

EV von $A^H A$ sind orthogonal

$V^H V = \mathbb{I} = U^H U$ ← Spalten ortho

A ist m×n

Wenn $A = USV^H$ dann $AA^H = US^2U^H$

und $A^H A = VS^2V^H$

=> Wenn $A^H = A$ ist $U = V$, EV paarweise orthogonal, SVD = "EW-Zerlegung"

$A = USV^T$ => daraus kann man U bestimmen (U, V unitär)

Hermitesche Matrizen

$A^H = A$

reelle EW

EV paarweise orthogonal in C^n oder identisch

EV basis von C^n

Diagonalisierbar sodass $U^H A U = \Lambda$

Diagonalisieren

Anwendung: $A = SDS^{-1}$ (S: EV, D: $\lambda_1 < \lambda_2$)

Voraussetzungen:

1. das char Polynom zerfällt vollständig in Linearfaktoren. // kein (λ^2+a) übrig
2. geom. Vielfachh. = algebraisch
   d.h. dim Eigenraum = Vielfachheit einer Nst.

DGL Transformationsmethode

1. notiere y-Vektor $\begin{pmatrix} y \\ y' \\ \dots \\ y^{(n-1)} \end{pmatrix}$

Bsp: $y'' = 2y + y'$

$$\begin{pmatrix} y \\ y' \\ \dots \\ y^{n-1} \end{pmatrix}' = \begin{pmatrix} y' \\ y'' \\ \dots \\ y^n \end{pmatrix} = A \cdot \begin{pmatrix} y \\ y' \\ \dots \\ y^{n-1} \end{pmatrix} = \begin{pmatrix} 0 & 1 & \dots & 0 \\ 2 & 1 & \dots & 0 \\ a & b & \dots & d \end{pmatrix} \begin{pmatrix} y \\ y' \\ \dots \\ y^{n-1} \end{pmatrix}$$

2. EW und EV von A bestimmen // falls diagonalisierbar

$\Lambda = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix}$ $V = \left(\begin{pmatrix} v_1 \end{pmatrix}, \begin{pmatrix} v_2 \end{pmatrix}, \begin{pmatrix} v_3 \end{pmatrix} \right)$

// Skalierung von V irrelevant, weil nachher eh · Const gerechnet wird

3. $\boxed{y\text{-Vektor}} = V e^{\Lambda x} c = V \begin{pmatrix} e^{\lambda_1 x} & & \\ & e^{\lambda_2 x} & \\ & & e^{\lambda_3 x} \end{pmatrix} = V \begin{pmatrix} c_1 e^{\lambda_1 x} \\ c_2 e^{\lambda_2 x} \\ c_3 e^{\lambda_3 x} \end{pmatrix}$ => $y = \dots$, $y' = \dots$, $y'' = \dots$

4. Falls Anfangswerte gegeben: $\begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = V^{-1} \begin{pmatrix} y(0) \\ y'(0) \\ y''(0) \end{pmatrix}$

// Falls Startgleichungen von Form $y_1' = a y_2 + b y_1$, $y_2' = c y_2 + d y_1$ => selbes Vorgehen aber mit y-Vektor $= \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$

// Falls ≠0 sondern = a => löse mit =0, am Schluss setze y = a
setze Ableitungen ein $y' = 0, y'' = 0$
und löse nach a
=> $y = y_h + a$

Maschinengenauigkeit

eps: $\left|\frac{x - \text{runde}(x)}{x}\right| \le eps$

eps ≈ kleinste positive Gleitzahl (in Praxis)

Beweis Cauchy-Schwarz

für beliebiges α gilt: $0 \le \langle \alpha x + y, \alpha x + y\rangle = \alpha\bar{\alpha}\langle x,x\rangle + \bar{\alpha}\langle x,y\rangle + \alpha\overline{\langle x,y\rangle} + \langle y,y\rangle$. Für $x=0$ gilt $=0$, für $x \neq 0$ wählen wi[r] $\alpha = -\frac{\langle x,y\rangle}{\langle x,x\rangle}$ es folgt $0 \le |\langle x,y\rangle|^2 - |\langle x,y\rangle|^2 - |\langle x,y\rangle|^2 + \langle x,x\rangle\langle y,y\rangle \Leftrightarrow |\langle x,y\rangle|^2 \le \langle x,x\rangle\langle y,y\rangle \Leftrightarrow |\langle x,y\rangle| \le \|x\|\|y\|$

Beweis Dreiecksungleichung

$\|x \pm y\|^2 = \langle x \pm y, x \pm y\rangle = \langle x,x\rangle \pm 2\langle x,y\rangle + \langle y,y\rangle$

Aus Schwarz folgt $\|x \pm y\|^2 \le \|x\|^2 + 2\|x\|\|y\| + \|y\|^2$
$= (\|x\| + \|y\|)^2$

Konditionszahl

bei ortho Matrix: $K_2 = 1$

sagt aus wie sich ein kleiner Fehler im Objekt aufs Verfahren auswirkt

Def: $K_2(A) = (\max(\sqrt{\lambda} \mid \lambda \text{ EW von } A^*A)) : (\min(\text{dasselbe}))$
durch

grosses K_2 => schlecht

regulärer Fall: Invertierbar → K_2 ist ein Quotient

singulärer Fall: nicht Invertierbar → $K = 0$ oder unendlich

fast singulär $\begin{pmatrix} 1 & 0 \\ 0 & \varepsilon \end{pmatrix}$: $\det = \varepsilon$ => det d. Inverse ist riesig

ε nahe an 0

Def: $K_1 = \|A\|_1 \cdot \|A^{-1}\|_1$
wobei $\|A\|_1 = \max(\text{Summe einer Spalte})$

Def: $K_\infty = \|A\|_\infty \cdot \|A^{-1}\|_\infty$
wobei $\|A\|_\infty = \max(\text{Summe einer Zeile})$

Zählt von 1 aus für Matrixzeilen

A(h,:) = Zeile bei h

Swap rows: A([1,3],:) = A([3,1],:)

Matlab

A \ b löst Ax=b, nach kleinste Quadrate bei Überbestimmtheit

.^ ist wie ^ aber für nicht-Matrizen

e^A = expm(A)

Symbolische Variabel (also ohne Wert): syms x

EW und EV bestimmen [v,λ] = eig(A) //wenn nur λ gewünscht, benutze [~,λ] =

Konditionszahl cond(A)

Graph in logarithmischer Skala auf beiden Achsen: loglog(X,Y,Z,A,...)

Einheitsmatrix mit grösse von A eye(size(A)) = eye(n,m)

Maximum: max(abs(A(n:N,n))) eye(n,n) = eye(n)

$A = {}_2\{\overset{3}{\square}$ => size(A) => [2;3]

Alle in Zeile B(1,:) = [2,3]
B(2,:) = 3

k Modulo 2 mod(k,2)

nur hole size(A,1)

Nullen A=zeros(n,n)

Einsen A=ones(n,n)

Matlab

`function [B] = myFnct(A,c)`	Funktion von (A, c) returns B
`[m,n] = size(A)`	define m as height, n as width of A
`if (m<n) x=2; else x=3; end`	If in one line
`x=(('a'~='b')·2)   %is 0 if 'a'=='b'`	=> results in x=2
`for i=1:x`	from 1 to x (including)
`  if(false) return; end`	true und false kleingeschrieben
`end`	
`if (false) x=0; disp('lulz')`	print lulz, strings are single '
`elseif (A==eye(2))`	
`B = A`	assign returnvalue
`end`	
`end`	

A' entspricht A^T

`@(x) x.^2` ist anonyme funktion

funktion kann normal als parameter übereicht werden

`global A; A = ...` in jeder Funktion wo A global sein soll => dort A ändern ändert es überall

Skalarprodukt $\langle a,b \rangle$: `dot(a,b)` | Betrag: `abs(a)`

arccos : `acos( b./c)`

sort in xth dimension (row)

`srted = sortrows(B,x)`

sort in 2nd column

`srted = sortrows(B',2)'`

`[L,U,P]=lu(A)` — LR mit Pivotisieren

A ähnlich zu B
wenn $B = S^{-1} A S$
↑ regulär

Schur-Zerlegung

$$C = T D T^H$$

↑ obere Dreiecksmatrix
↑ unitär

- Existiert für n×n Matrix immer
- Transformation ist nicht eindeutig
- Diagonalelemente von D = EW von D = EW von C
- C orthogonal =>
  - D ist Diagonalmatrix
  - Spalten von T = ~~W~~EV von C
  - Schurzerlegung heisst dann auch "Spektralzerlegung"
- C positiv definit =>
  - Schurzergegung(C) = Singulärwertzerlegung von C

do {

1. Bestimme λ_1 und v_1 von C_i
2. Bestimme $(n-i)$ weitere Vektoren die orthogonal Basis zu K^{n+1-i} bilden

while ($i \in [1, \ldots, n-1]$)

$T_i = [v_1, w_2, \ldots, w_{n-i+1}]$

3. $(T_i)^{Tr} C_i T_i = \begin{pmatrix} \lambda_1 & ? \\ 0 & C_{i+1} \end{pmatrix}$

4. $T = T_1 \cdot \hat{T}_2 \cdot \ldots \cdot \hat{T}_{n-1}$

wobei $\hat{T}_i = \begin{vmatrix} 1 & 0 & \cdots & & 0 \\ 0 & 1 & & & \\ \vdots & & \ddots & & \\ \vdots & & 1 & 0 & 0 \\ 0 & & 0 & T_i & \\ 0 & & 0 & & \end{vmatrix}$

Matlab vorgehen

Interpolate:

1. calculate x=A\b for correct solution
2. approx with a polynomial

```
for i=1:n
    p=p+x(i)*space.^(i-1);

end
```

Matrixpotenz

```
if k==0
  Returnvalue = eye(size(B))
elseif k==2
  Retval = A*A
elseif mod(k,2)==0
  Retval = potenz(k/2,A)
else
  Retval = potenz(k-1,A)*A
end
```

LR mit Pivotisierung

```
N= size(A,1); p=(1:N)';
for n=1:N-1
  [r,m] = max(abs(A(n:N,n)));      Spaltenweise Pivotsuche
  m=m+n-1;
  if abs(A(m,n))<eps
     error('LR-Zerlegung existiert nicht');
  end;
  if(m~=n)                         Verhindere Spaltentausch falls unnötig
  A([n m],:)=A([m n],:); p([n m])=p([m n])
```

	0	$\frac{\pi}{6}$ 30°	$\frac{\pi}{4}$ 45°	$\frac{\pi}{3}$ 60°	90° $\frac{\pi}{2}$
sin	0	$\frac{1}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{3}}{2}$	1
cos	1	$\frac{\sqrt{3}}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{1}{2}$	0
tan	0	$\frac{\sqrt{3}}{3}$	1	$\sqrt{3}$	$\pm\infty$
cot	$\pm\infty$	$\sqrt{3}$	1	$\frac{\sqrt{3}}{2}$	0

Stichwortsuche

Datenbank:
$$\begin{array}{c|ccc} & \text{Wort }1 & 2 & 3 \\ \hline \text{File }1 & 1 & 0 & 0 \\ \text{" }2 & 0 & 1 & 1 \\ \text{" }3 & 1 & 1 & 0 \end{array}$$

Anfrage: $q=(1\ 1\ 1)$

=> für jede Zeile SP nehmen, kleinsten Abweichungswinkel berechnen

=> in diesem Bsp hat (1 0 0) den kleinsten winkel

LR

```
n= size(A,1); % height
L = zeros(n);
for k = 1:n          % transform A into upper triangular
                     % Form and create L

  L(k+1:n,k) = A(k+1:n,k)*(1/A(k,k))
  for l= k+1:n
     A(l,:)=A(l,:)-L(l,k)*A(k,:);
  end
end

% add diag. elems
L=L+eye(n);

% R is row-reduced A now
R=zeros(n);
for k=1:n
  R(1:k,k)=A(1:k,k);
end
```