

TMV166 Linjär algebra för M

Veckoprogram 5: Egenvärdesproblem

Lay: 5.1–5.4, 5.7–5.8. Egenvärden och egenvektorer.

Förra veckan i **F12** presenterade jag Determinanter från Lay 3.1–3.2 (vi hoppar över 3.3). Den här veckan ska vi öva på detta.

Föreläsningarna kommer att handla om Egenvärdesproblem.

Rekommenderade uppgifter

Determinanter.

Avsnitt	
3.1	PP, 3, 4, 9, 10, 15, 37, 17, 19–21,
3.2	PP, 5, 7, 11, 13, 21, 25, 29 , 9, 15, 17, 19, 27, 28, 32, 39, 41–43

Egenvärdesproblem.

Avsnitt	
5.1	PP, 1, 3, 5, 7, 9, 13, 15 , 17, 19, 21, 22, 25, 29, 31
5.2	PP, 1, 5, 9, 12, 13 , 17, 19, 21, 22, 20
5.3	PP, 1, 5, 7, 3, 11, 15, 17, 21, 22, 23, 27, 31, 32
5.7	1, 3, 5, 7
5.4	, , PP, 1, 3, 5, 9, 11, 15 , 21, 32

OBS! Bortse från frågor som berör sänka, källa eller sadelpunkt i 5.7.

Datorövningar.

I föreläsning F15 ser vi hur man kan använda potensiteration för att hitta en egenvektor, och därmed även tillhörande egenvärde. MATLAB använder istället en metod som bestämmer alla egenvärden och egenvektorer samtidigt. Den är baserad på en generalised form av *QR-iteration*. (Väldigt) enkelt uttryckt så sätter man först $A_1 = A$ och för $k = 1, 2, \dots$, faktoriserar man sedan $A_k = Q_k R_k$ och sätter $A_{k+1} = R_k Q_k$ tills man erhåller egenvärdena på diagonalen av R_k . Varje steg görs implicit för att hålla nere beräkningskostnaden, dvs. Q_k och R_k bildas aldrig explicit.

Via kommandot `eig` kan man få både egenvärden och egenvektorer, på olika sätt:

- $E = \text{eig}(A)$ ger en kolonnmatrix E , vars element är egenvärdena till matrisen A .

- `[V, D] = eig(A)` ger två matriser V och D . Kolonnerna i V är egenvektorerna till A , och D är en diagonalmatrix med motsvarande egenvärden till A på diagonalen, ordnade i samma ordning. Vidare är kolonnerna i V normerade: $\text{norm}(V(:,k)) = 1$. Om A är symmetrisk är kolonnerna dessutom parvis ortogonala, så att $V^T V = I$ (eller $V^{-1} = V^T$).
- `[V, D] = eig(A, 'vector')` ger samma sak som ovan, förutom att D istället blir en kolonnmatrix.

Observera att `[V, D] = eig(A)` ger resultat även om A **inte** är diagonaliserbar. Detta inträffar om A har ett egenvärde med större (algebraisk) multiplicitet än motsvarande egenrumms dimension; i så fall spänner inte egenrummen upp hela \mathbb{R}^n . Egenvärdena räknas upp i D enligt multipliciteten, men motsvarande kolonner i V är inte linjärt oberoende och V är inte inverterbar. Produkten `inv(V)*A*V` ger också ett resultat, som ofta även blir D , men man får förhoppningsvis en varning av typen *Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND = 1.110223e-16*.

Uppgift 1. Bestäm egenvärden och egenvektorer till följande matriser. Undersök om matriserna är diagonaliserbara med reella matriser eller med komplexa matriser eller inte alls. Undersök också om den egenvektormatrisen V är ortogonal eller ej.

$$A_1 = \begin{bmatrix} 1 & 0 & 0 \\ 3 & -1 & 0 \\ 5 & 4 & 0 \end{bmatrix} \quad A_2 = \begin{bmatrix} 5 & 4 & -2 \\ 4 & 5 & 2 \\ -2 & 2 & 8 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad A_4 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

□

Spänningsmatrisen

Spänningsmatrisen $S = \begin{bmatrix} \sigma_x & \tau_{xy} & \tau_{xz} \\ \tau_{yx} & \sigma_y & \tau_{yz} \\ \tau_{zx} & \tau_{zy} & \sigma_z \end{bmatrix}$ beskriver normalspänningar σ och skjuvspänningar

τ i plan parallella med koordinatplanen, genom en kropp, ett kontinuum, som påverkas av inre och yttre krafter. Egenvektorer till matrisen S är huvudspänningsriktningarna och motsvarande egenvärden är normalspänningen i plan vinkelräta mot egenvektorn. I dessa plan är skjuvspänningen noll. Matrisen S är alltid symmetrisk och därmed alltid diagonaliserbar.

Spänningsvektorn s på en viss snittyta med enhetsnormalvektor n ges av $s = Sn$.

Normalspänningen på snittytan ges av längden av projektionen av spänningsvektorn på planets enhetsnormal, $\sigma = n^T S n$. Skjuvspänningen på snittytan är längden av spänningsvektorns projektion på snittytan. Denna beräknas enkelt med Pythagoras sats: $\tau^2 = \|s\|^2 - \sigma^2$.

Om n är en normerad egenvektor till S så är $s = Sn = \lambda n$, där λ är motsvarande egenvärde. I det fallet är $\sigma = \|s\| = \lambda$ och $\tau = 0$. Därav begreppet huvudspänning och huvudspänningsriktning.

Uppgift 2. Spänningstillståndet i en punkt Q i en kropp har beräknats med finita-elementmetoden och uttrycks i ett kartesiskt koordinatsystem (x, y, z) med spänningsmatrisen

$$S = \begin{bmatrix} 30 & 0 & 10 \\ 0 & 30 & 10 \\ 10 & 10 & 30 \end{bmatrix} \text{ MPa.}$$

- a) Beräkna normal- och skjuvspänning på en snittyta med normalvektor $n = \frac{1}{\sqrt{5}} [1 \ 2 \ 5]^T$.
- b) Beräkna huvudspänningar och huvudspänningsriktningar.

□

Uppgift 3. Skriv en MATLAB-funktion som beräknar det dominerande egenvärdet (om sådant egenvärde finns) och en motsvarande egenvektor x till matrisen A med potensmetoden. Funktionen ska ha deklarationen

```
function [lambda,x]=potensmetoden(A,x0,tol)
```

Här är x_0 en startvektor och tol en tolerans för stoppvillkoret $|x_{k+1} - x_k| \leq tol$. Testa med matriser från föregående uppgifter. Förklara varför, om det inte fungerar. □

Uppgift 4. Skriv en MATLAB-funktion som beräknar det egenvärde som är närmast talet α (om sådant egenvärde finns) och en motsvarande egenvektor x till matrisen A med inversa potensmetoden. Funktionen ska ha deklarationen

```
function [lambda,x]=inversapotensmetoden(A,x0,alpha,tol)
```

Här är x_0 en startvektor och tol en tolerans för stoppvillkoret $|x_{k+1} - x_k| \leq tol$. Testa med matriser från föregående uppgifter. Förklara varför, om det inte fungerar. □

Facit

```
function [lambda,x1]=potensmetoden(A,x0,tol)
    x=x0;
    delta=2*tol;
    while delta>tol
        y=A*x;
        y=y/norm(y);      % Normera y
        y=sign(x'*y)*y;   % Se till att y och x inte motriktade
        delta=norm(x-y);
        x=y;
    end
    lambda=y'*A*y;
    x1=x;
end
% x0=rand(size(A,1),1)
```

```
function [lambda,x1]=inversapotensmetoden(A,x0,alpha,tol)
    x=x0;
    delta=2*tol;
    I=eye(size(A),'like',A); % 'like' ger gles format om A är gles
    B=(A-alpha*I);
    while delta>tol
        y=B\x;           % y=(A-alpha I)^(-1)x
        y=y/norm(y);     % Normera y
        y=sign(x'*y)*y;  % Se till att y och x inte motriktade
        delta=norm(x-y);
        x=y;
    end
    lambda=y'*A*y;      % Rayleigh-kvoten
    x1=x;
end
```