# OEFENINGEN NUMERIEKE WISKUNDE

OEFENZITTING 16 (PC): HET BEREKENEN VAN EIGENWAARDEN

Voor het uitwerken van de opgaven moet je `.m`-bestanden gebruiken die je kan vinden op Toledo

## Beschrijving van de Matlab-bestanden

- `m = machten(A, x0, N)` past de methode van de machten met scalering toe. Hierbij is `m` het tupel `[x, mu, e]`. Na oproep van `[x, mu, e] = machten(A, x0, N)` worden `N` iteraties uitgevoerd met de methode van de machten op de matrix `A` met `x0` als startvector. Als resultaat krijg je eerst een tekening te zien, met daarop de opeenvolgende benaderingen van de eigenwaarde in modulus (aangeduid met een +) en de fouten (aangeduid met een *) t.o.v. de modulus van de laatst berekende benadering voor de eigenwaarde. Tenslotte krijg je de numerieke waarden voor de benaderde eigenvector `x`, de modulus `mu` van de bijhorende eigenwaarde en de foutenvector `e` te zien.
- `l = invmachten(A, x0, lambda, N)` past de methode der inverse machten toe. Hierbij is `l` een vector met de opeenvolgende benaderingen voor de eigenwaarde die wordt gevonden door de methode der inverse machten toe te passen op de matrix $(\sigma I - A)$ met `x0` als startvector en waarbij $\sigma$ wordt ingegeven door de parameter `lambda`. Men kan zo de eigenwaarde vinden waarvoor $\sigma$ een benadering is. Hoe beter de schatting $\sigma$ voor de eigenwaarde, hoe sneller de methode zal convergeren.

## Oefeningen

**Probleem 1. (Methode van de machten)** Pas de methode van de machten toe om de dominante eigenwaarde te berekenen van onderstaande matrices $A$ en $B$. Gebruik hierbij als startvector `x0` achtereenvolgens $[1\ 0\ 0]^T$ en $[1\ 1\ 1]^T$.

$$A = \begin{pmatrix} 1002 & -2001 & 1000 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}, \quad B = \begin{pmatrix} 1 & -3 & 5 \\ -1 & -7 & 11 \\ -1 & -9 & 13 \end{pmatrix}.$$

Verklaar telkens het convergentiegedrag. Lees hiertoe paragraaf 5 pag. 101 in het tweede deel van de cursus. Verwijder de normalisatiestap uit het programma `machten.m` en doe een aantal iteraties met matrix $A$ en startvector $[1\ 1\ 1]^T$. Wat concludeer je?

**Probleem 2. (Methode der inverse machten)** Gebruikt `invmachten.m` om van bovenstaande matrix $B$ met startvector `x0` $= [1\ 0\ 0]$ sneller de eigenwaarden te vinden door aan $\sigma$ een benadering van de gezochte eigenwaarde toe te kennen. Doe een aantal experimenten voor verschillende waarden van $\sigma$ en verklaar telkens het convergentiegedrag.

**Probleem 3. (Rayleigh quotiënt iteratie)** Bewijs de volgende stelling (oef 6.1 p. 103):

*Als $\lambda$ een eigenwaarde is van $A \in \mathbb{R}^{n\times n}$ en $X \in \mathbb{R}^{n\times 1}$ een bijhorende eigenvector, dan is*

$$\lambda = \frac{X^T A X}{X^T X}.$$

*Men noemt deze uitdrukking een Rayleigh quotiënt.*

Bij Rayleigh quotiënt iteratie kiest men in elke stap van de inverse machtmethode

$$\lambda = \lambda_k = \frac{q_k^T A q_k}{q_k^T q_k}.$$

Implementeer dit algoritme en bereken een eigenwaarde van

$$A = \begin{pmatrix} 1 & 0.5 & 2 \\ 2 & 0 & 3 \\ -1 & 2 & 4 \end{pmatrix}$$

met startvector `x0` $= [1\ \ 1\ \ 1]^T$ en initiële schatting voor $\lambda = 5$. Vergelijk de resultaten die u bekomt met de inverse machtmethode. Ga na dat de Rayleigh quotiënt iteratie kwadratisch convergeert. Voor hermitische matrices convergeert de methode zelfs kubisch.
Bij het uitvoeren van je programma zal Matlab een waarschuwing geven wanneer het aantal iteratiestappen te hoog is. Hoe verklaar je dit?

**Probleem 4. (Deelruimte-iteratie)** Schrijf een functie `lam = deelruimte(A, x0, N)` die de methode van de machten met meerdere startvectoren implementeert. Hier is `A` $\in \mathbb{R}^{n\times n}$ de matrix waarvan je $m \leqslant n$ dominante eigenwaarden wil berekenen, `x0` $\in \mathbb{R}^{n\times m}$ een matrix waarvan de kolommen overeen komen met de startvectoren en `N` het aantal iteratiestappen. De output is een matrix `lam` $\in \mathbb{R}^{N\times m}$, waarvan de rijen de benaderende dominante eigenwaarden na elke iteratiestap weergeven. Test je algoritme op een aantal voorbeelden. Praktische info:

- Als een matrix `A` meer rijen heeft dan kolommen, berekent het commando `[q, r] = qr(A, 0)` een “economy size” $QR$-factorisatie, waarbij de dimensies van `q` en `A` gelijk zijn.
- Een testmatrix $T$ met gekende eigenwaarden kan je als volgt aanmaken: $T = WDW^{-1}$ met $W$ een willekeurige matrix (commando `rand`) en $D$ een diagonaalmatrix met de eigenwaarden (commando `diag`).

**Probleem 5.** Pas de methode van Rayleigh toe op matrix $\begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}$ met startwaarden $x0 = [5\ \ -3]^T$ en $\lambda = 4$. Convergeert de methode naar de dichtstbijgelegen eigenwaarde (zoals de methode van de inverse machten)? Verklaar! Opmerking: de startvector is *geen* eigenvector!

**Probleem 6.** Werkt de methode van de machten nog steeds wanneer de matrix niet diagonaliseerbaar is, i.e. wanneer de eigenvectoren geen basis vormen? Probeer bijvoorbeeld eens

$$A = \begin{bmatrix} 5 & \sqrt{2} & 1 \\ 0 & 6 & 0 \\ 1 & \sqrt{2} & 5 \end{bmatrix}.$$

Geef een algemeen bewijs.