Numerieke Modellering en Benadering: Practicum 1

Ellen Anthonissen Marte Biesmans

vrijdag 21 april 2016

Opgave 1

De Householder transformatiematrix

$$F = I - 2\frac{vv^*}{v^*v}$$

heeft als eigenwaarden -1 en 1 en als eigenvectoren respectievelijk v en w met $w \perp v$. Deze resultaten zijn als volgt bekomen: De Householder transformatiematrix F is symmetrisch

$$F^* = (I - 2\frac{vv^*}{v^*v})^* = I - 2\frac{vv^*}{v^*v} = F$$

en unitair

$$F^*F = FF^* = (I - 2\frac{vv^*}{v^*v})(I - 2\frac{vv^*}{v^*v}) = I - 4\frac{vv^*}{v^*v} + 4\frac{v(v^*v)v^*}{(v^*v)^2} = I.$$

Omdat F unitair is, moeten de eigenwaarden van F op de complexe eenheidscirkel gelegen zijn. Omdat F reëel en symmetrisch is, zijn de eigenwaarden reële getallen. Hieruit volgt dat de eigenwaarden enkel ± 1 kunnen zijn.

Als we nu Fv uitrekenen, bekomen we

$$Fv = v - 2\frac{vv^*}{v^*v}v = -v.$$

Hieruit volgt dat v en eigenvector is bijhorende bij de eigenwaarde -1. Neem nu w met $w \perp v$ en we rekenen Fw uit, dan bekomen we

$$Fw = w - 2\frac{vv^*}{v^*v}w = w,$$

want $v^*w = 0$. Hieruit volgt das w een eigenvector is bijhorende bij de eigenwaarde 1.

Geometrisch gezien komt dit overeen met een spiegeling over de w-as. Neem een vector a en ontbind die in een compontent volgens de v-as en een component volgens de w-as. De component volgens de v-as zal vermenigvuldigt worden met -1 en die volgens de w-as met 1. Zo bekomen we een spiegeling rond de w-as.

n \ κ	1	10^4	10^8
10	0,0044	0,0022	0,0025
100	0,1883	0,1403	0,1306
1000	28,2691	27,725	27,9583

Tabel 1: De snelheid van de expliciete methode

n \ κ	1	10^4	10^8
10	0,0050	0,0015	0,0013
100	0,0115	0,0131	0,0163
1000	7,9605	7,7967	7,8586

Tabel 2: De snelheid van de impliciete methode

n \ κ	1	10^4	10^8
10	10^{-16}	10^{-13}	10^{-9}
100	10^{-15}	10^{-13}	10^{-9}
1000	10^{-14}	10^{-13}	10^{-9}

Tabel 3: De ordegrootte van de relatieve fout van de expliciete methode

n \ κ	1	10^4	10^8
10	10^{-16}	10^{-13}	10^{-9}
100	10^{-15}	10^{-13}	10^{-9}
1000	10^{-14}	10^{-13}	10^{-9}

Tabel 4: De ordegrootte van de relatieve fout van de impliciete methode

n \ κ	1	10^4	10^8
10	10^{-16}	10^{-13}	10^{-9}
100	10^{-15}	10^{-13}	10^{-10}
1000	10^{-15}	10^{-13}	10^{-10}

Tabel 5: De ordegrootte van de verhouding van de norm van het residu op de b-vector van de expliciete methode

n \ κ	1	10^4	10^8
10	10^{-16}	10^{-13}	10^{-9}
100	10^{-15}	10^{-13}	10^{-10}
1000	10^{-15}	10^{-13}	10^{-10}

Tabel 6: De ordegrootte van de verhouding van de norm van het residu op de b-vector van de impliciete methode

Opgave 2

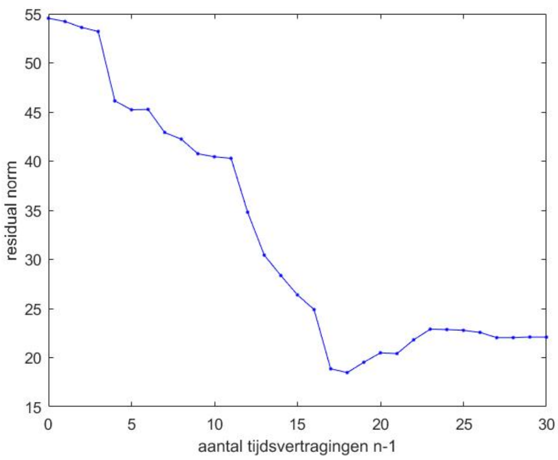
EVENTUEEL MATLAB CODE

Opgave 3

Het kleinste kwadraten probleem $\|b - Ax\|_2$ ziet er uit als volgt:

$$A = \begin{bmatrix} u_n & u_{n-1} & \cdots & u_1 \\ u_{n+1} & u_n & \cdots & u_2 \\ \vdots & \vdots & \ddots & \vdots \\ u_m & u_{m-1} & \cdots & u_{m-n+1} \end{bmatrix}, b = \begin{bmatrix} y_n \\ y_{n+1} \\ \vdots \\ y_m \end{bmatrix},$$

met u_n de n-de waarde uit de vector van de ingangen uit `model.mat`. Dit stelsel $Ax = b$ lossen we op naar x met behulp van een van de algoritmes uit de vorige opgave. Om dit model te valideren stellen we op analoge wijze een matrix B op met de ingangen uit `validate.mat`. We vergelijken de verschillende modellen met behulp van de waarde $\|y - Bx\|_2$ met y de uitgangen uit `validate.mat`. Voor verschillende tijdsvertragingen levert dit de waardes uit figuur 1. Hieruit blijkt dat 18 tijdsvertragingen het meest interessant is, of $n = 19$.



Figuur 1: Validatie van het model bij verschillende tijdsvertragingen

Opgave 4

Neem een vector $x \in \mathbb{R}^n$. Schrijf x als lineaire combinatie van de eigenvectoren van A $q_1, q_2 \ldots q_n$ met bijhorende eigenwaarden $\lambda_1, \lambda_2 \ldots \lambda_n$:

$$x = \sum_{j=1}^{n} a_j q_j,$$

dan is het Rayleigh quotiënt van x:

$$r(x) = \frac{\sum_{j=1}^{n} a_j^2 q_j \lambda_j}{\sum_{j=1}^{n} a_j^2}.$$

Het Rayleigh quotiënt is onafhankelijk van de schaal van x, dus stel $\|a\| = \|[\, a_1 \; a_2 \; \cdots \; a_n \,]^T\| = 1$, dan is $\sum_{j=1}^{n} a_j^2 = 1$. Dan wordt

$$r(x) = \sum_{j=1}^{n} a_j^2 q_j \lambda_j.$$

Stel nu dat $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, dan is het Rayleigh quotiënt maxiaal voor $a = e_1$ met de waarde λ_{max} en minimaal voor $a = e_n$ met de waarde λ_{min}. Dus het Rayleigh quotiënt bevint zich in het interval $[\, \lambda_{min}, \lambda_{max} \,]$.

Ook is het Rayleigh quetiënt een continu voor $a \neq 0$, dus elke waarde tussen λ_{min} en λ_{max} wordt bereikt voor een x.

Opgave 5

'relatie tussen 4 methodes' Het QR-algoritme met of zonder shifts berekent alle eigenwaardes. 'uitspraak toelichten' 'convergentiegedrag tonen van 4 methodes'

Opgave 6

We maken gebruik van de ongelijkheid

$$\frac{\|e_n\|_A}{\|e_0\|_A} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^n.$$

Voor $n = 10$ wordt dit

$$\frac{\|e_{10}\|_A}{\|e_0\|_A} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{10}.$$

Gebruik makende van de gegevens $\|e_0\|_A = 1$ en $\|e_{10}\|_A = 2 \times 2^{-10}$, bekomen we

$$9 \leq \kappa.$$

We vinden dus een ondergrens voor κ.

Voor $n = 20$ wordt de ongelijkheid

$$\frac{\|e_{20}\|_A}{\|e_0\|_A} \leq 2\left(\frac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right)^{20}.$$

Gebruik makende van het gegeven $\|e_0\|_A = 1$ en het berekende $9 \leq \kappa$, bekomen we

$$\|e_{20}\|_A \leq 2 \times 2^{-20}.$$

We vinden dus een bovengrens voor $\|e_{20}\|_A$.

Opgave 7

Opgave 8

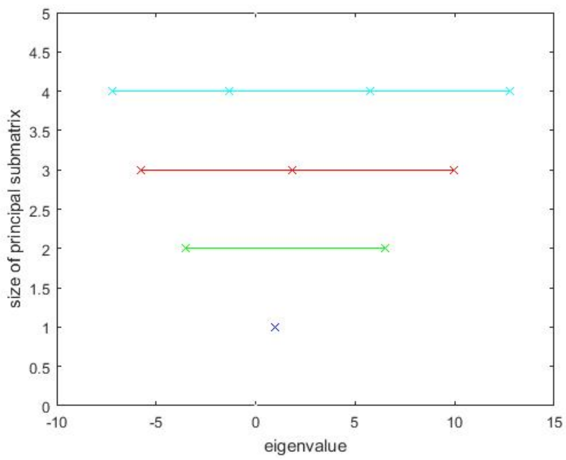
Opgave 9

De interlace eigenschap van een tridiagonale, symmetrische en reële matrix A luidt als volgt. Voor $A \in \mathbb{R}^{n\times n}$, en $A^{(1)}, A^{(2)} \dots A^{(n)}$ de principale vierkante submatrices van dimensie $1, 2 \dots n$, geldt dat de eigenwaarden van deze submatrices interlacen. Dit wil zeggen dat $\lambda_j^{(k+1)} \leq \lambda_j^{(k)} \leq \lambda_{j+1}^{(k+1)}$. Als A irreduceerbaar is (en dus geen nul heeft op een nevendiagonaal), dan worden de ongelijkheden strikte ongelijkheden. Dit laatste is belangrijk voor de bisectie-methode.

Als voorbeeld nemen we de tridiagonale, symmetrische en reële matrix

$$A = \begin{bmatrix} 1 & 5 & 0 & 0 \\ 5 & 2 & 6 & 0 \\ 0 & 6 & 3 & 7 \\ 0 & 0 & 7 & 4 \end{bmatrix}.$$

We zien duidelijk op figuur 2 dat de eigenwaarden van een principale submatrix tussen de eigenwaarden van de principale submatrix van een dimensie groter liggen.



Figuur 2: De eigenwaarden van de opeenvolgende principale submatrices van A

Opgave 10

Een algoritme in pseudo-code dat alle eigenwaarden van een symmetrische, tridiagonale en reële matrix in het interval [a,b] berekent tot op een bepaalde tolerantie met behulp van de bisectie-methode, is te vinden in algoritme 1.

Deze methode werd uitgeschreven in MATLAB (zie bijgevoegde MATLAB-code) en werd toegepast op enkele symmetrische, tridiagonale en reële matrices.

Algorithm 1 Bisectie-methode

queue = [[a,b)]
while queue niet leeg **do**
  Haal eerste interval uit de queue
  $Sturm_{links}$ = aantal tekenwisselingen in de Sturm-rij van A-linkergrens*I
  $Sturm_{rechts}$ = aantal tekenwisselingen in de Sturm-rij van A-rechtergrens*I
  #eigenwaarden = $Sturm_{rechts}$ - $Sturm_{links}$
  midden = 0.5*(rechtergrens - linkergrens)
  if #eigenwaarden == 1 **then**
    if midden < tolerantie **then**
      Voeg linkergrens+midden toe als eigenwaarde
    else
      Voeg [linkergrens,linkergrens+midden] en [linkergrens+midden,rechtergrens] vooraan bij in de queue
    end if
  else if #eigenwaarden > 1 **then**
    Voeg [linkergrens,linkergrens+midden] en [linkergrens+midden,rechtergrens] vooraan bij in de queue
  end if
end while

Als eerste voorbeeld nemen we de tridiagonale, symmetrische en reële matrix $A \in \mathbb{R}^{n \times n}$ met $n = 4$.

$$A = \begin{bmatrix} 1 & 5 & 0 & 0 \\ 5 & 2 & 6 & 0 \\ 0 & 6 & 3 & 7 \\ 0 & 0 & 7 & 4 \end{bmatrix}$$

Als we de eigenwaarden van deze matrix willen berekenen in het interval [-2,6) tot op een tolerantie 10^{-2}, dan vinden we we eigenwaarden [-1,3359375; 5,7734375]. Deze komen overeen met de 'exacte' eigenwaarden [-1,32970777874292; 5,77851991186833] berekend met de functie eig(A) van MATLAB. Als we beide resultaten afronden tot op twee cijfers na de komma, zien we inderdaad dat de eigenwaarden gevonden zijn tot op de gegeven absolute fout.

Als tweede voorbeeld genereren we de random symmetrische, tridiagonale en reële matrix $B \in \mathbb{R}^{n \times n}$ MATLAB met $n = 10$. De diagonalen zijn berekend met de functie rand(n) en rand($n-1$). De elementen van de matrix B zijn hier afegrond tot op 4 cijfers na de komma.

$$B = \begin{bmatrix} 0,4170 & 0,2068 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,2068 & 0,9718 & 0,6539 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0,6539 & 0,9880 & 0,0721 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0,0721 & 0,8641 & 0,4067 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0,4067 & 0,3889 & 0,6669 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,6669 & 0,4547 & 0,9337 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0,9337 & 0,2467 & 0,8110 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0,8110 & 0,7844 & 0,4845 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,4845 & 0,8828 & 0,7567 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,7567 & 0,9137 \end{bmatrix}$$

Deze matrix heeft in totaal 10 eigenwaraden. Als we nu enkel geïnteresseerd zijn in de eigenwaarden tussen 0 en 1 tot op minstens 10^{-10} nauwkeurig, vinden we met behulp van de bisecite-methode de eigenwaarden [0,155349893786479; 0,212609510694165; 0,511023909028154; 0,768950582772959]. Deze komen overeen met de 'exacte' eigenwaarden [0,155349893779548; 0,212609510637192; 0,511023909027886; 0,768950582751513] berekend met de functie eig(B) in MATLAB. We zien dat de resultaten bekomen door de bisectie-methode inderdaad juist zijn tot op de gegeven tolerantie.