

3.24pt

Oefenzitting 2

Nico Vervliet

6 maart 2018

Outline

3.24pt

Pen en papier

Opgave 1

Opgave 2

Computeropgaven

Opgave 3

Opgave 4

Opgave 5

QR van een bandmatrix

$$\begin{bmatrix} \times & \times & 0 & 0 & 0 & 0 & 0 \\ \times & \times & \times & 0 & 0 & 0 & 0 \\ \times & \times & \times & \times & 0 & 0 & 0 \\ 0 & \times & \times & \times & \times & 0 & 0 \\ 0 & 0 & \times & \times & \times & \times & 0 \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times & \times \end{bmatrix}$$

Figuur: Bovenbandbreedte $q = 2$ en onderbandbreedte $p = 2$.

Merk op

We hanteren hier de definitie dat bovenbandbreedte de diagonaal meetelt, terwijl de onderbandbreedte deze niet meetelt.

Vorm QR van bandmatrix

$$\begin{bmatrix} \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times & \times & \times \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times & 0 & 0 & 0 \\ 0 & \times & \times & \times & \times & 0 & 0 \\ 0 & 0 & \times & \times & \times & \times & 0 \\ 0 & 0 & 0 & \times & \times & \times & \times \\ 0 & 0 & 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & 0 & 0 & \times \end{bmatrix}$$

Conclusie

- ▶ Q heeft onderband van p , bovenband van n
- ▶ R is bovendriehoeks, bovenband van $p + q$

Vorm QR van bandmatrix (2)

Vorm van Q

- ▶ Elke kolom q_i is som van q_j met $j \leq i$

Vorm van R

- ▶ $r_{ij} = q_i^* a_j$
- ▶ q_i heeft $p + i$ niet-nullen bovenaan
- ▶ a_j heeft $\max\{0, j - q\}$ nullen bovenaan
- ▶ $r_{ij} \neq 0$ als $j - q < p + i$ of $j - i < p + q$

Complexiteit Householder

for $k = 1, \dots, n$ **do**

$$x = A_{k:m,k}$$

$$v_k = \text{sign}(x_1) \|x\| e_1 + x$$

$$v_k = v_k / \|v_k\|$$

$$A_{k:m,k:n} = A_{k:m,k:n} - 2v_k(v_k^* A_{k:m,k:n})$$

Complexiteit Householder

for $k = 1, \dots, n$ **do**

$$x = A_{k:m,k}$$

$$v_k = \text{sign}(x_1) \|x\| e_1 + x$$

$$v_k = v_k / \|v_k\|$$

$$A_{k:m,k:n} = A_{k:m,k:n} - 2v_k(v_k^* A_{k:m,k:n})$$

+-	*/	√
$2pn$	$3pn$	$2n$
$2p(p+q)n$	$2p(p+q)n$	

Complexiteit Givens

for *elke* $a_{ji} \neq 0$ *met* $i < j$ **do**

Voorbereidende stappen:

$$r = \sqrt{a_{ii}^2 + a_{ji}^2}$$

$$c = a_{ii}/r$$

$$s = -a_{ji}/r$$

Matrixtransformatie:

$$a_{i,i:n} = ca_{i,i:n} - sa_{j,i:n}$$

$$a_{j,i:n} = sa_{i,i:n} + ca_{j,i:n}$$

Complexiteit Givens

for elke $a_{ji} \neq 0$ met $i < j$ **do**

Vorbereidende stappen:

$$r = \sqrt{a_{ii}^2 + a_{ji}^2}$$

$$c = a_{ii}/r$$

$$s = -a_{ji}/r$$

Matrixtransformatie:

$$a_{i,j:n} = ca_{i,j:n} - sa_{j,i:n}$$

$$a_{j,i:n} = sa_{i,j:n} + ca_{j,i:n}$$

+-	*/	√
pn	$4pn$	pn
$2p(p+q)n$	$4p(p+q)n$	

a) Kolom verwijderen

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{bmatrix}$$

a) Kolom verwijderen

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{bmatrix}$$

a) Kolom verwijderen

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \\ 0 & 0 & \times \end{bmatrix}$$

a) Kolom verwijderen

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & \times & \times \\ 0 & 0 & \times \end{bmatrix}$$

Givens rotaties om nullen te maken

- ▶ $Q^T \tilde{A} = H$ (H is in Hessenbergvorm)
- ▶ $Q_1 = Q G_k \cdots G_{n-1}$
- ▶ $R_1 = G_{n-1}^T \cdots G_k^T H$
- ▶ $\tilde{A} = Q_1 R_1$

b) Kolom toevoegen

$$\begin{bmatrix} \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \\ \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times \\ 0 & \times & \times \\ 0 & 0 & \times \\ 0 & 0 & 0 \end{bmatrix}$$

b) Kolom toevoegen

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & 0 & \times \\ 0 & \times & 0 & 0 \end{bmatrix}$$

Extra kolom

- ▶ $Q^T \tilde{A} = H$
- ▶ $w = Q^T z$ (w is de extra kolom in H)

b) Kolom toevoegen

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & \times & 0 & \times \\ 0 & \times & 0 & 0 \end{bmatrix}$$

Extra kolom en Givens Rotaties

- ▶ $Q^T \tilde{A} = H$
- ▶ $w = Q^T z$
- ▶ $Q_1 = Q G_{k+1} \cdots G_{m-1}$
- ▶ $R_1 = G_{k+1}^T \cdots G_{m-1}^T H$
- ▶ $\tilde{A} = Q_1 R_1$
- ▶ Let op volgorde van de rotaties: van onder naar boven zodat enkel op de diagonaal niet-nullen geïntroduceerd worden.

c) Rij toevoegen

$$\begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} = \begin{bmatrix} \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \end{bmatrix}$$

Uitbreiden Q en R

- ▶ $Q_d = \begin{bmatrix} 1 \\ Q \end{bmatrix}$
- ▶ $Q_d^T \tilde{A} = \begin{bmatrix} w^T \\ R \end{bmatrix} = H$

Matlab

```
format short e;  
% Matrix  
X = rand(5,2); X(2:5,1) = X(2:5,1)*1.e-8;  
% Beide reflecties toepassen op X  
U1 = X(:,1) + norm(X(:,1))*eye(size(X,1),1)  
U2 = X(:,1) - norm(X(:,1))*eye(size(X,1),1)  
H1X = X - 2 / (U1'*U1) * U1 * U1' * X  
H2X = X - 2 / (U2'*U2) * U2 * U2' * X
```

Bespreking

- ▶ H1X is zoals het hoort: enkel eerste element van kolom 1 is niet nul, andere zijn zeer klein (ongeveer $1e-20$ tot $1e-30$)
- ▶ H2X is niet zoals het hoort: de nullen zijn niet echt nul (ongeveer $1e-10$)
- ▶ Het probleem is de slechte conditie van de aftrekking.

Opgave 4

► Voorbeeld:

$$A = \begin{bmatrix} 1 & 1 \\ 0 & \epsilon \\ 0 & 0 \end{bmatrix} \quad (1)$$

met $\epsilon = 10^{-12}$.

Opgave 4

► Voorbeeld:

$$A = \begin{bmatrix} 1 & 1 \\ 0 & \epsilon \\ 0 & 0 \end{bmatrix} \quad (1)$$

met $\epsilon = 10^{-12}$.

Opgave 4

- ▶ Normaalstelsel $A^T A x = A^T b$:

$$A^T A = \begin{bmatrix} 1 & 1 \\ 1 & 1 + \epsilon^2 \end{bmatrix} \approx \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \quad (2)$$

Op een computer is $1 + \epsilon^2 = 1$ vanwege de beperkte machinenauwkeurigheid

- ▶ Householder $Rx = Q^T b$:
 - ▶ $Q = I$
 - ▶ $R = A$

Opgave 4

```
e = 1e-12;  
A = [1 1; 0 e; 0 0];  
b = rand(3,1);  
  
x1 = (A'*A) \ (A'*b)  
R = A  
x2 = R \ b
```


a)

```
function [d,T,ds]=fastgivensQR1d(A)
...
ds=zeros(m,n);
...
for j=1:n
    for i=m:-1:j+1
        %nieuwe alfa,beta en d bepalen
        [alfa,beta,type,newd]=fastgivens1(T(i-1:i,j),d(i-1:i,j));
        % we willen dit weten
        d(i-1:i)=newd;
        ...
    end
    % opslaan
    ds(:,j) = d';
end
```

a)

```
%% Opgave 5a  
figure(1)  
A = randn(100,50);  
[d,~,ds] = fastgivensQR1d(A);  
subplot(1,2,1)  
plot(ds(end,:))
```

b) Oude versie (fastgivens.m)

```
type=1;  
tau=d(1);  
newd(1)=(1+gamma)*d(2);  
newd(2)=(1+gamma)*tau;
```

b) Nieuwe versie (fastgivens.m)

```
if gamma <= 1
    type=1;
    tau=d(1);
    newd(1)=(1+gamma)*d(2);
    newd(2)=(1+gamma)*tau;
else
    type=2;
    alfa=1/alfa;
    beta=1/beta;
    gamma=1/gamma;
    newd(1)=(1+gamma)*d(1);
    newd(2)=(1+gamma)*d(2);
end
```

c)

```
%% Opgave 5a
figure(1)
A = randn(100,50);
[d,~,ds] = fastgivensQR1d(A);
subplot(1,2,1)
plot(ds(end,:))
lim = ylim;

%% Opgave 5c
subplot(1,2,2)
[d,~,ds] = fastgivensQRd(A);
plot(ds(end,:))
ylim(lim)
```

c)

- ▶ Pieken zijn vaak lager (niet altijd zo!)
- ▶ Aangezien de laatste d een combinatie is van twee d waarden, kan de piek terug ongedaan gemaakt worden.

d)

- ▶ Met aanpassing nog steeds groei
- ▶ Observeer M , D en T en herschaal indien overflow dreigt
- ▶ Nog steeds probleem bij niet-volle rang: pivoteer.

e)

```
A = rand(m,n);  
b = rand(m,1);  
[d,T] = fastgivensQR([A,b]);  
x = T(1:n,1:n)\T(1:n,n+1);  
norm(A*x - b) - norm(A*(A\b) - b)
```