3 Ausgleichsrechnung

Sei $Q \in \mathbb{R}^{N\times N}$ orthogonal, d.h. $Q^T Q = I_N$. Dann ist $\kappa_2(Q) = 1$.

(3.1) Zu $v \in \mathbb{R}^N$ und $k \neq n$ mit $v[k]^2 + v[n]^2 > 0$ existiert eine *Givens-Rotation* $G \in \mathbb{R}^{N\times N}$ mit

$$\begin{pmatrix} G[k,k] & G[k,n] \\ G[n,k] & G[n,n] \end{pmatrix} = \begin{pmatrix} c & s \\ -s & c \end{pmatrix}, \qquad c^2 + s^2 = 1\,,$$

und $G[j][j] = 1$ für $j \neq k,n$ und $G[i][j] = 0$ sonst, so dass $e_n^T G v = 0$ gilt:
Für $|v[n]| > |v[k]|$ setze $\tau = \frac{v[k]}{v[n]}$, $s = \frac{1}{\sqrt{1+\tau^2}}$, $c = s\tau$, sonst setze $\tau = \frac{v[n]}{v[k]}$, $c = \frac{1}{\sqrt{1+\tau^2}}$, $s = c\tau$.
Es gilt $G = c(e_k e_k^T + e_n e_n^T) + s(e_k e_n^T - e_n e_k^T) + \sum_{j\neq k,n} e_j e_j^T$.

(2.10) Zu $v \in \mathbb{R}^N$, $v \neq 0_N$, existiert eine *Householder-Spiegelung* $H = I_N - \frac{2}{w^T w} w w^T \in \mathbb{R}^{N\times N}$ mit $w \in \mathbb{R}^N$, $w[1] = 1$, so dass $Hv = \sigma e_1$ mit $\sigma \in \mathbb{R}$ und $Hw = -w$ gilt:
Falls $v[1] > 0$, setze $\sigma = -|v|_2$, sonst setze $\sigma = |v|_2$. Dann definierte $w = \frac{1}{v[1]-\sigma}(v - \sigma e_1)$.

(3.3) Zu $A \in \mathbb{R}^{K\times N}$ existiert eine QR-Zerlegung $A = QR$ mit einer orthogonalen Matrix $Q \in \mathbb{R}^{K\times K}$ und eine oberen Dreiecksmatrix $R \in \mathbb{R}^{K\times N}$, d.h. $QQ^T = I_K$ und $R[k+1:K,k] = 0_{K-k}$ für $k = 1,\dots,K$.

(3.4) Sei $A \in \mathbb{R}^{K\times N}$ und $b \in \mathbb{R}^K$. Dann gilt:

$$x \in \mathbb{R}^N \text{ minimiert } |Ax - b|_2 \qquad \Longleftrightarrow \qquad A^T A x = A^T b\,.$$

Berechnung der Householder-Vektoren

```
function [v,beta] = householder(y)
N = length(y);
s = y(2:N)' * y(2:N);
if N == 1
  s = 0;
end;
v = [1;y(2:N)];
if s == 0
  beta = 0;
else
  mu = sqrt(y(1)^2 + s);
  if y(1) <= 0
    v(1) = y(1) - mu;
  else
    v(1) = -s/(y(1) + mu);
  end;
  beta = 2*v(1)^2/(s + v(1)^2);
  v = v / v(1);
end;
return;
```

QR-Zerlegung

Karlsruhe Institute of Technology

```
[M,N] = size(A);
for m = 1:min(N,M-1)
  [v,beta] = householder(A(m:M,m));
  if beta ~= 0
    w = beta * v' * A(m:M,m:N);
    A(m:M,m:N) = A(m:M,m:N) - v * w;
    A(m+1:M,m) = v(2:M-m+1);
  end;
end;

for m = 1:min(N,M-1)
  v = [1;A(m+1:M,m)];
  beta = 2 / (v' * v);
  if beta ~= 2
     b(m:M) = b(m:M) - beta*(v'*b(m:M)) * v;
  end;
end;
for n=min(N,M):-1:1
  x(n) = (b(n) - A(n,n+1:N) * x(n+1:N)) / A(n,n);
end;
```