Algorithmische Methoden in der Numerik - Uebung2

Felix Dreßler (k12105003)
Elisabeth Köberle (k12110408)
Eva Zahn (k12110309)

14. Juni 2022

1 Aufgabe a - QRFact

```
function [A, D, pi , k ] = QRFact (A)

[m,n] = size(A);


pi = 1:n; %p=pi
si = zeros(n,1);
%siq = zeros(n,1);

D = zeros(min(m,n),1);

nq = n;

for j = n:-1:1
    si(j) = dot(A(:,j),A(:,j));
    if si(j) == 0
        temp1 = pi(j); %alternative (maybe less efficient) [pi(j),pi(nq)] = deal(pi(nb),
            pi(j));
        pi(j) = pi(nq);
        pi(nq) = temp1;
        nq = nq-1;
    end
end
siq = si;

for i = 1:nq%different loop iterator than in script, here i is k

    [val,piv] = max(si(pi(i:nq))./siq(pi(i:nq)));

    if val <= -1
        k =i-1;
        return;
    end

    temp1 = pi(i);
    pi(i) = pi(piv);
    pi(piv) = temp1;

    si(pi(i)) = dot(A(:,pi(i)),A(:,pi(i)));

    if si(pi(i)) < m * eps^2 * siq(pi(i))
        k = i-1;
        return;
    end

    if sign(A(i,(pi(i)))) == 0 %to compensate for the different sign function
        D(i,1) = sqrt(si(pi(i)));
    else
        D(i,1) = -sign(A(i,(pi(i)))) * sqrt(si(pi(i)));
    end

    A(i,pi(i)) = A(i, pi(i)) - D(i,1);

    for j = i+1:nq
        gamma = dot(A(i:m,pi(j)),A(i:m,pi(i))) / (-D(i,1)*A(i,pi(i))); %dot() is
            scalarproduct
        A(i:m, pi(j)) = A(i:m, pi(j)) - gamma * A(i:m, pi(i));
```

```
56          si(pi(j)) = si(pi(j)) - A(i, pi(j))^2;
57          if si(pi(j)) < m * eps * siq(pi(j))
58              si(pi(j)) = dot(A(l,pi(j)),A(l,pi(j)));
59          end
60      end
61  end
62  k = nq;
63  end
```

2 Aufgabe b - QRSolve

Unter Verwendung von den in *Aufgabe c* berechneten Q und R wurde in *Aufgabe b* den Vektor x berechnet.

```
function [ x ] = QRSolve (B,D, p , k , b )

%[A,d,p,k] = QRFact(B);
[m,n] = size(B);

if k < n
    x = zeros(n:1);
else

    Q = CompQ(B,p,k);
    R = CompR(B,D,p,k);

    Qt = transpose(Q);
    Rt = transpose(R);

    c= Qt * b;

    x = Rt * c;
end
end
```

3 Aufgabe c

3.1 CompR

```
function [R] = CompR(B,D, p , k )

R= triu ( B (: , p ));
R= full ( spdiags (D ,0 , R) );

end
```

3.2 CompQ

```
function [Q] = CompQ(B, p , k )

[m ,~]= size (B);
Q= eye (m );

for j =1: k

    v= zeros (m ,1) ;
    if(j >1)
        v (1: j -1) =0;
    end
    v(j :m)=B (j:m ,p(j) );

    P= eye (m) -(2/ dot (v ,v)) *( v* transpose (v ));
    Q=Q *P;
end
end
```

4 Tests