Oplossingen NMB1: Kleinste-kwadratenbenadering (deel 1)

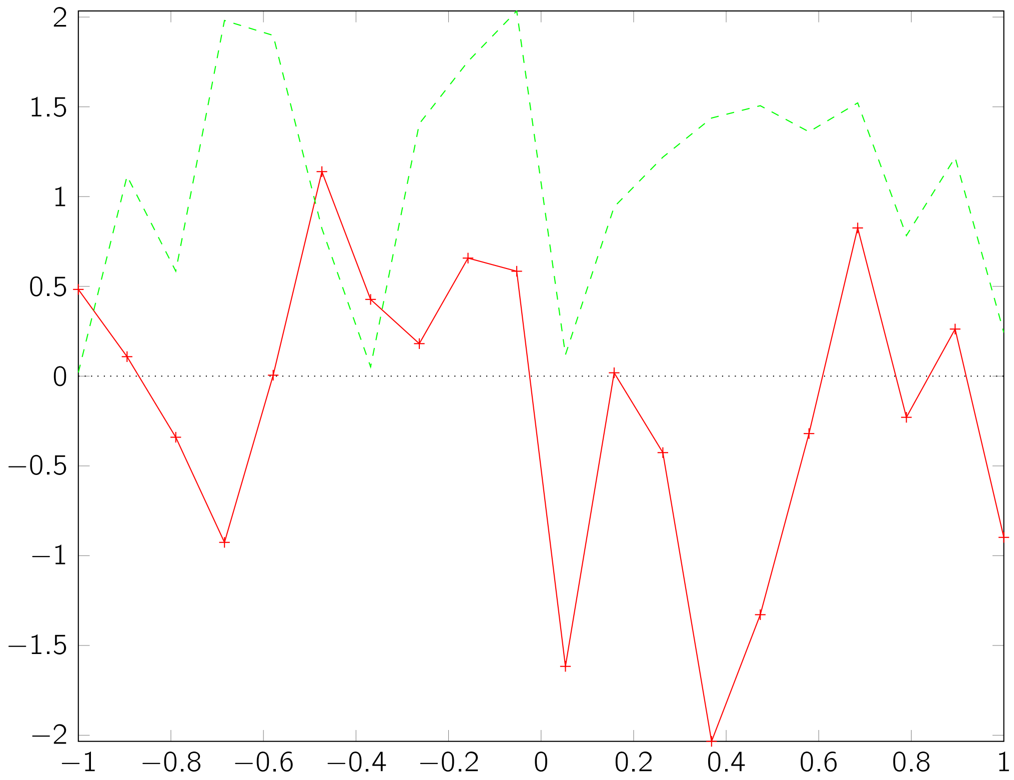
Nico Vervliet, *KU Leuven* *2 mei 2016*

1 Opgave 1:

Oplossing

```
function plotres(x,r,w)
%PLOTRES plots residuals in combination with weights

    hold on

    plot(x,r,'r+-') % Plot residuals
    plot([x(1) x(end)], [0 0], 'k:') % x-axis
    scalefactor = max(abs(r));
    plot(x,w/max(w)*scalefactor,'g--') % Plot scaled weights
    ylim([-scalefactor scalefactor]) % center x-axis

    hold off
end
```

```
% Test function
x = linspace(-1, 1, 20);
r = randn(size(x));
w = rand(size(x));
clf
box on
plotres(x,r,w)
```



2 Opgave 2:

```
function c = kkb1(x, f, w, n)
%KKB1 kleinste-kwadratenveeltermbenadering van maximale graad n

    N = length(x);
    D = diag(sqrt(w));
    A = cumprod([ones(N, 1) x(:,ones(1,n))], 2);
    A = D*A;
    b = D*f;
    c = A \ b;
end
```

```
function c = kkb2(x, f, w, n)
%KKB2 KKB met veeltermen van maximale graad n
%   Stelt discrete kleinste-kwadratenveeltermbenadering op
%   van graad n voor de punten (x_i, f_i) met gewichten w_i,
%   i = 1, ..., N.
    x = x(:); f = f(:); w = w(:);
```

```
    N = length(x);
    X = [ones(N, 1), x(:,ones(1,2*n))];
    X = cumprod(X, 2);
    A = w'*X;
    A = hankel(A(1:n+1), A(n+1:end)');
    b = (w.*f)'*X(:,1:n+1);
    c = A \ b';
end
```

```
function c = kkb2v2(x, f, w, n)
    x = x(:); f = f(:); w = w(:);
    N = length(x);

    W = diag(w);
    A = cumprod([ones(N, 1) x(:,ones(1,n))], 2);
    b = A'*W*f;
    c = (A'*W*A) \ b;
end
```

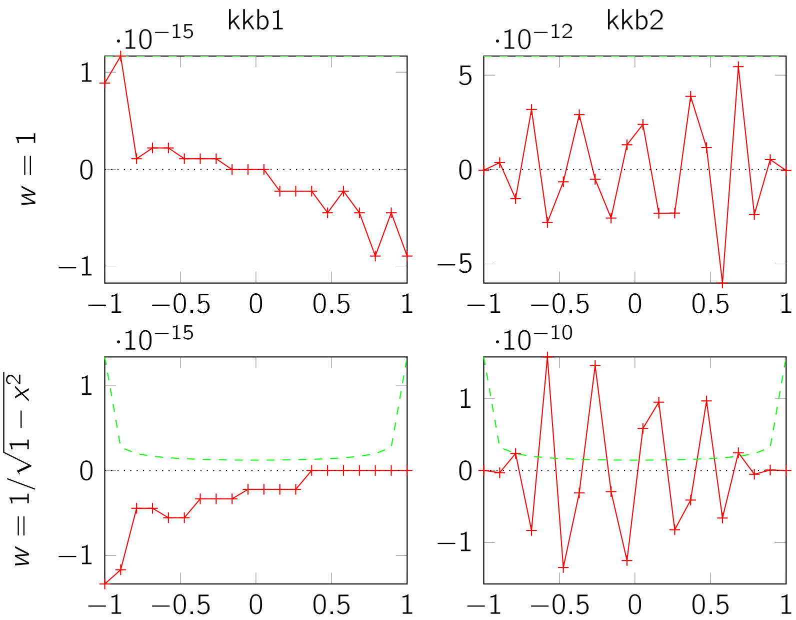
Verschil tussen `kkb2` en `kkb2v2`: voor lage N is de tweede versie sneller, voor hogere N is de eerste versie sneller.

3 Opgave 3:

```
%% generate data
x = linspace(-1, 1, 20)';
f = exp(x);
w = 1./sqrt(1-x.^2);
w(1) = 11;
w(end) = 11;
n = 5;
c1 = kkb1(x,f,w,n);
c2 = kkb2(x,f,w,n);
```

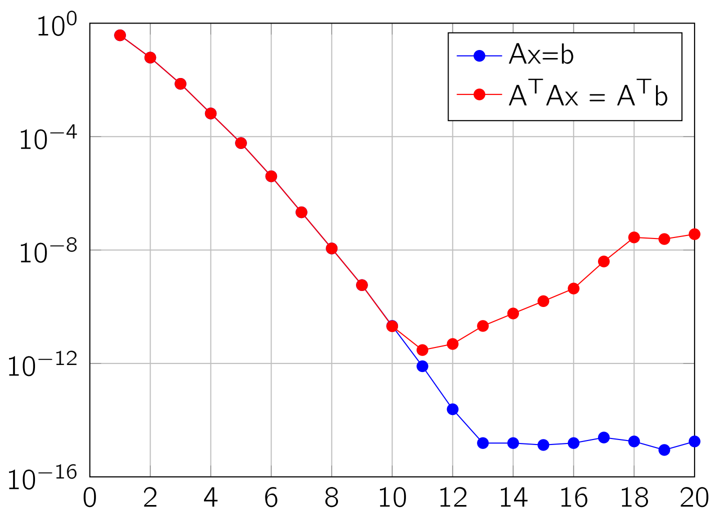
```
%%
figure();
n = 15;
w = ones(size(x));
c = kkb1(x,f,w,n);
y = polyval(c(end:-1:1), x);
r = y-f;
subplot(2,2,1);
```

```
plotres(x,r,w);
title('kkb1')
ylabel('$w=1$','Interpreter','LaTex')

c = kkb2(x,f,w,n);
y = polyval(c(end:-1:1), x);
r = y-f;
subplot(2,2,2);
plotres(x,r,w);
title('kkb2')

w = 1./sqrt(1-x.^2);
w(1) = 11;
w(end) = 11;
c = kkb1(x,f,w,n);
y = polyval(c(end:-1:1), x);
r = y-f;
subplot(2,2,3);
plotres(x,r,w);
ylabel('$w=1/\sqrt{1-x^2}$','Interpreter','LaTex')

c = kkb2(x,f,w,n);
y = polyval(c(end:-1:1), x);
r = y-f;
subplot(2,2,4);
plotres(x,r,w);
```



Als we `kkb1` gebruiken, dan zijn de fouten steeds van de ordegrootte van de machineprecisie. Als we `kkb2` gebruiken, dan is dit niet langer het geval. Het gebruik van de gewichtsfunctie $w = 1/\sqrt{1-x^2}$ zorgt ervoor dat de fouten aan de rand harder benadeeld worden, waardoor deze kleiner worden.

```
%% Last plot
N = 20;
residual = zeros(1, N);
residual2 = zeros(1, N);
for n = 1:N
    c = kkb1(x,f,w,n);
    y = polyval(c(end:-1:1), x);
    residual(n) = max(abs(y-f));

    c = kkb2(x,f,w,n);
    y = polyval(c(end:-1:1), x);
    residual2(n) = max(abs(y-f));
end
figure;
semilogy(1:N, residual, 'b.-', 1:N, residual2, 'r.-');
legend('Ax=b', 'A^TAx  = A^Tb', 'Interpreter', 'LaTex')
grid on;
```



De afrondingsfouten nemen bij hoge graad toe in geval van `kkb2`. Het conditiegetal wordt immers gekwadrateerd.

4 Opgave 4:

Voor opgave 4 wordt geen oplossing gegeven gezien de gelijkenis met de vraag op het practicum NMB. Een hints voor het practicum: Varieer het punten en de graad van de veeltermen. Bespreek wat er gebeurt.