

Numerieke Modelling en Benadering: Chebyshev veeltermen

Sander Prenen

15 april 2020

Inhoudsopgave

1	Inleiding	2
2	Continue kleinste kwadratenbenadering met Chebyshev veeltermen	2
2.1	Beste benadering in V_n	2
2.1.1	Orthogonaliteit van de basisvectoren T_k	2
2.1.2	Beste benadering	3
2.2	Evalueren van de Chebyshev veeltermen	3
2.3	Bepalen van de coëfficiënten	4
2.4	Benaderen van een functie	5
3	Interpolatie in Chebyshev knooppunten	5
3.1	Interpoleren van een functie	6
3.1.1	$\cos(x)$ benaderen	6

1 Inleiding

In dit practicum worden de benaderingseigenschappen van Chebyshev veeltermen van de eerste soort. Deze veeltermen blijken uiterst geschikt voor het benaderen van eindige, reële functies met behulp van de continue kleinste kwadratenbenadering. Ook zijn hun nulpunten vaak een uitstekende keuze voor de abscissa voor veelterminterpolatie. In sectie 2 wordt dieper ingegaan op de benaderingseigenschappen, terwijl in sectie 3 de interpolatie besproken wordt.

2 Continue kleinste kwadratenbenadering met Chebyshev veeltermen

In deze sectie wordt geprobeerd continue functies op eindige, reële intervallen te benaderen aan de hand van Chebyshev-veeltermen van de eerste soort. Dit zijn veeltermen die voldoen aan de volgende voorwaarde: $T_k(x) = \cos(k \arccos(x))$ voor $x \in [-1, 1]$ en $k = 0, 1, 2, \dots$. De veeltermen vormen een basis voor de ruimte V_n , een deelvectorruimte van $C([-1, 1])$.

2.1 Beste benadering in V_n

Indien de basisvectoren $T_k(x)$ orthogonale vectoren zijn, geldt volgende uitdrukking voor de beste benadering $y_n(x)$ voor een functie $f(x) \in C([-1, 1])$:

$$y_n(x) = \sum_{k=0}^n a_k T_k(x) \quad \text{met} \quad a_k = \frac{(f, T_k)}{(T_k, T_k)} \quad (1)$$

Met als scalair product:

$$(f, g) = \int_{-1}^1 \frac{f(x)g(x)}{\sqrt{1-x^2}} dx$$

Om deze formule te kunnen gebruiken, moet dus worden aangetoond dat de basisvectoren orthogonaal zijn. Dit wil zeggen dat alle onderlinge scalaire producten nul zijn, tenzij dat van een basisvector met zichzelf.

2.1.1 Orthogonaliteit van de basisvectoren T_k

In deze sectie wordt de orthogonaliteit van de basisvectoren bekeken. Hiervoor moeten alle onderlinge scalaire producten bepaald worden. Dit wordt gedaan in drie gevallen:

- $k = l = 0$:

$$(T_k, T_l) = \int_{-1}^1 \frac{T_0^2(x)}{\sqrt{1-x^2}} dx = \int_{-1}^1 \frac{1}{\sqrt{1-x^2}} dx = [\arcsin(x)]_{-1}^1 = \pi$$

- $k = l \neq 0$:

$$(T_k, T_l) = \int_{-1}^1 \frac{T_k^2(x)}{\sqrt{1-x^2}} dx = \int_{-1}^1 \frac{\cos^2(k \arccos(x))}{\sqrt{1-x^2}} dx = \frac{2\pi k + \sin(2\pi k)}{4k} = \frac{\pi}{2}$$

- $k \neq l$:

$$\begin{aligned}
(T_k, T_l) &= \int_{-1}^1 \frac{T_k(x)T_l(x)}{\sqrt{1-x^2}} dx = \int_{-1}^1 \frac{\cos(k \arccos(x)) \cos(l \arccos(x))}{\sqrt{1-x^2}} dx \\
&= \frac{1}{2} \int_{-1}^1 \frac{\cos((k-l) \arccos(x))}{\sqrt{1-x^2}} dx + \frac{1}{2} \int_{-1}^1 \frac{\cos((k+l) \arccos(x))}{\sqrt{1-x^2}} dx \\
&= \frac{1}{2} \frac{\sin((k-l)\pi)}{k-l} + \frac{1}{2} \frac{\sin((k+l)\pi)}{k+l} = 0
\end{aligned}$$

Alle scalaire producten zijn nul, behalve die van basisvectoren met zichzelf. De vectoren zijn dus orthogonaal.

2.1.2 Beste benadering

Doordat de basisvectoren orthogonaal zijn, kan formule (1) gebruikt worden. In deze sectie worden de coëfficiënten a_k bepaald.

$$\begin{aligned}
a_k &= \frac{(f, T_k)}{(T_k, T_k)} = \frac{1}{(T_k, T_k)} \int_{-1}^1 \frac{f(x)T_k(x)}{\sqrt{1-x^2}} dx = \frac{1}{(T_k, T_k)} \int_{-1}^1 \frac{f(x) \cos(k \arccos(x))}{\sqrt{1-x^2}} dx \\
&= \frac{1}{(T_k, T_k)} \int_{\pi}^0 \frac{f(\cos \theta) \cos(k\theta) (-\sin(\theta))}{\sqrt{1-\cos^2(\theta)}} d\theta = \frac{1}{(T_k, T_k)} \int_0^{\pi} f(\cos \theta) \cos(k\theta) d\theta
\end{aligned}$$

Dit geeft dus volgende uitdrukking voor a_k gebruikmakend van de uitdrukkingen in sectie 2.1.1:

$$a_k = \begin{cases} \frac{1}{\pi} \int_0^{\pi} f(\cos \theta) d\theta, & k = 0 \\ \frac{2}{\pi} \int_0^{\pi} f(\cos \theta) \cos(k\theta) d\theta, & k > 0 \end{cases} \quad (2)$$

2.2 Evalueren van de Chebyshev veeltermen

Om de beste benadering numeriek te bepalen, moet vergelijking (1) geëvalueerd worden. De functie `evalCheb` geeft een vector $v = (f_1, f_2, \dots, f_N) \in \mathbb{R}^N$ terug. Deze vector wordt bekomen uit inputvectoren $a = (a_0, a_1, \dots, a_n) \in \mathbb{R}^{n+1}$ en $x = (x_1, x_2, \dots, x_N) \in \mathbb{R}^N$ op de volgende manier:

$$f_i = y_n(x_i) = a_0 T_0(x_i) + a_1 T_1(x_i) + \dots + a_n T_n(x_i)$$

In Listing 1 wordt de MATLAB code voor deze berekening weergegeven.

```

1 function v = evalCheb(a,x)
2
3 % Lengte van de vectoren bepalen
4 n = length(a);
5 N = length(x);
6
7
8 % Vector v invullen
9 v = zeros(1,N);
10 for i = 1:1:N % v loop
11     z = x(i);
12     v(i) = 0;
13     for j = 1:1:n % a en T loop
14         v(i) = v(i) + a(j) * cos((j-1) * acos(z));
15     end
16 end

```

Listing 1: evalCheb.m

2.3 Bepalen van de coëfficiënten

Voor het bepalen van de coëfficiënten a_k kan vergelijking (2) gebruikt worden. Deze vergelijking kan met behulp van de trapeziumregel voor numerieke integratie met een discretizatie bestaande uit n intervallen, benaderd worden door:

$$a_k \approx \frac{1}{n}f(1) + \frac{2}{n} \sum_{l=1}^{n-1} f\left(\cos\left(\frac{l\pi}{n}\right)\right) \cos\left(k\frac{l\pi}{n}\right) + \frac{(-1)^k}{n}f(-1), \quad 0 < k \leq n$$

Hieruit volgt dat de coëfficiënten a_k kunnen berekend worden als de DCT van de rij:

$$f(z_{0,n}), f(z_{1,n}), \dots, f(z_{n,n}) \quad (3)$$

Hierin is $z_{l,n} = \cos(l\pi/n)$. Deze punten zijn de extrema van $T_k(x)$ aangezien dit de nulpunten zijn van de eerste afgeleide van $T_k(x)$.

$$\frac{dT_k(x)}{dx} = \frac{d}{dx} \cos(k \arccos(x)) = -\sin(k \arccos(x)) \cdot k \cdot -\frac{1}{\sqrt{1-x^2}}$$

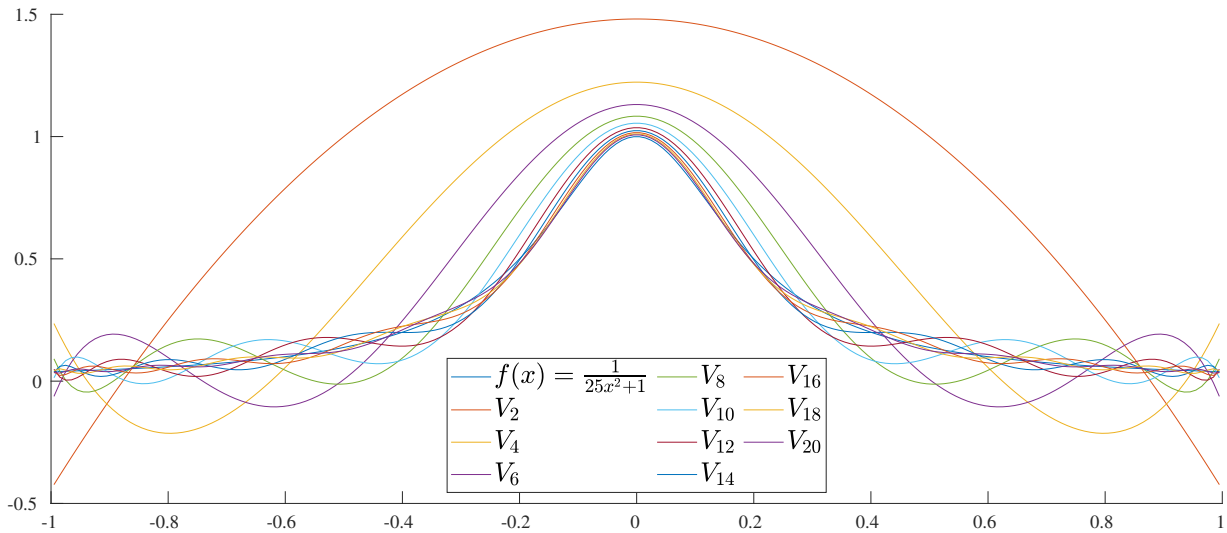
De afgeleide wordt nul als $k = 0$ of $\sin(k \arccos(x)) = 0$. In het tweede geval herleidt dit zich tot:

$$k \arccos(x) = l\pi \Leftrightarrow x = \cos\left(\frac{l\pi}{k}\right)$$

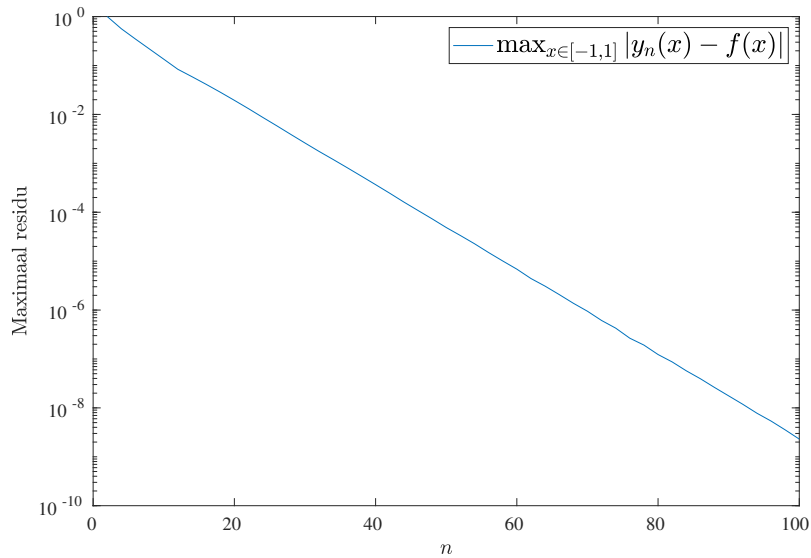
In de functie `approxCheby` wordt de DCT van de rij in vergelijking (3) bepaald. De code van deze functie kan gevonden worden in Listing 2 weergegeven.

```
1 function a = approxCheby(f,n)
2
3 % z vector maken
4 z = zeros(1,n);
5 for i = 1:1:n+1
6     z(i) = cos((i-1) * pi / n);
7 end
8
9 % f evalueren
10 v = zeros(1,n);
11 for i = 1:1:n+1
12     v(i) = f(z(i));
13 end
14
15 % DCT berekenen
16 w = fliplr(v(2:end-1));
17 v_even = [v w];
18 V = fft(v_even)/n;
19 V = real(V(1:n+1));
20 V = [V(1)/2 V(2:end)];
21
22 a = V;
```

Listing 2: `approxCheby.m`



Figuur 1: De beste benaderingen van de Runge functie in de ruimtes V_2 tot V_{20} .



Figuur 2: De maximale fout voor de benadering van de Runge functie in de ruimtes V_2 tot V_{100} .

2.4 Benaderen van een functie

De algoritmes die in Listing 1 en 2 te zien zijn, kunnen gebruikt worden bij het benaderen van een willekeurige functie in $C([-1, 1])$. Om een functie te benaderen worden de coëfficiënten a_k bepaald met `approxCheby`. Deze coëfficiënten worden meegegeven aan `evalCheb`. De punten waarin de benadering geëvalueerd moet worden meegegeven in de x vector. In dit geval wordt gekozen voor 200 equidistante punten tussen -1 en 1 .

In figuur 1 worden de beste benaderingen voor de Runge functie in ruimtes V_2 tot V_{20} weergegeven. De Runge functie heeft als voorschrift $f(x) = \frac{1}{25x^2 + 1}$. Figuur 2 toont de maximale afwijking $d(y_n, f) = \max_{x \in [-1, 1]} |y_n(x) - f(x)|$, in functie van de graad van de benaderende veelterm.

3 Interpolatie in Chebyshev knooppunten

Veeltermen van graad $n - 1$ kunnen een functie interpoleren in n interpolatiepunten. De interpolatiepunten x_k , voor $k = 1, 2, \dots, n$, kunnen vrij gekozen worden. Er wordt gezocht naar

een veelterm $g(x)$ van graad $n - 1$ die de functie $f(x)$ interpoleert zodanig dat de volgende voorwaarde voldaan is:

$$g(x_i) = f(x_i), \quad i = 1, \dots, n$$

De veelterm $g(x)$ kan worden voorgesteld in de vorm $g(x) = \sum_{k=0}^{n-1} c_k \psi_k(x)$. Hierin zijn $\psi_k(x)$ de basisfuncties, in dit geval dus $T_k(x)$. De interpolatievoorwaarden leiden tot volgend lineair stelsel:

$$Mc = B$$

De oplossing van dit stelsel zijn de coëfficiënten c_k . De elementen M_{ij} van de matrix M op rij i en kolom j worden gegeven door:

$$M_{ij} = \psi_j(x_i)$$

en de elementen B_i van de vector B worden gegeven door:

$$B_i = f(x_i).$$

In de functie `interpolate` worden de coëfficiënten c_k en het conditiegetal van de matrix M berekend. Als invoer verwacht de functie de interpolatiepunten x_k en de te benaderen functie $f(x)$. De code van deze functie is terug te vinden in Listing 3.

```

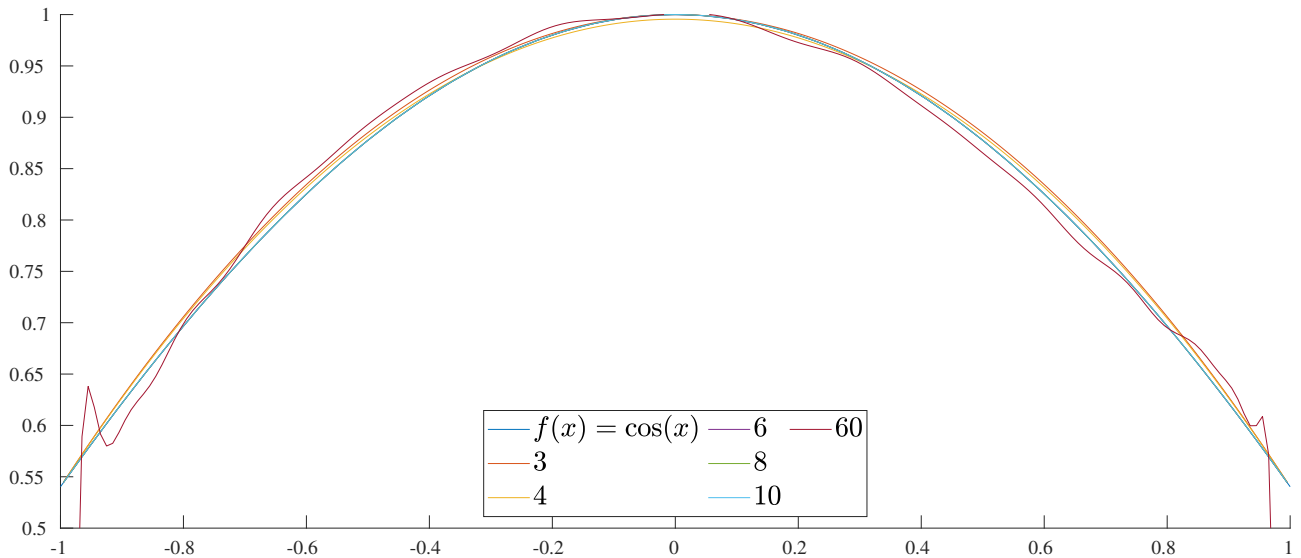
1 function [c, kappa] = interpolate(x,f)
2
3 % n bepalen
4 n = length(x);
5
6 % B vector opstellen
7 B = zeros([n,1]);
8 for i = 1:1:n
9     B(i) = f(x(i));
10 end
11
12 % M opstellen
13 M = zeros(n);
14 for i = 1:1:n
15     for j = 1:1:n
16         M(i,j) = cos((j-1) * acos(x(i)));
17     end
18 end
19
20 % Conditiegetal
21 kappa = cond(M);
22
23 % QR-factorisatie
24 [Q, R] = qr(M);
25 c = R \ transpose(Q) * B;
```

Listing 3: interpolate.m

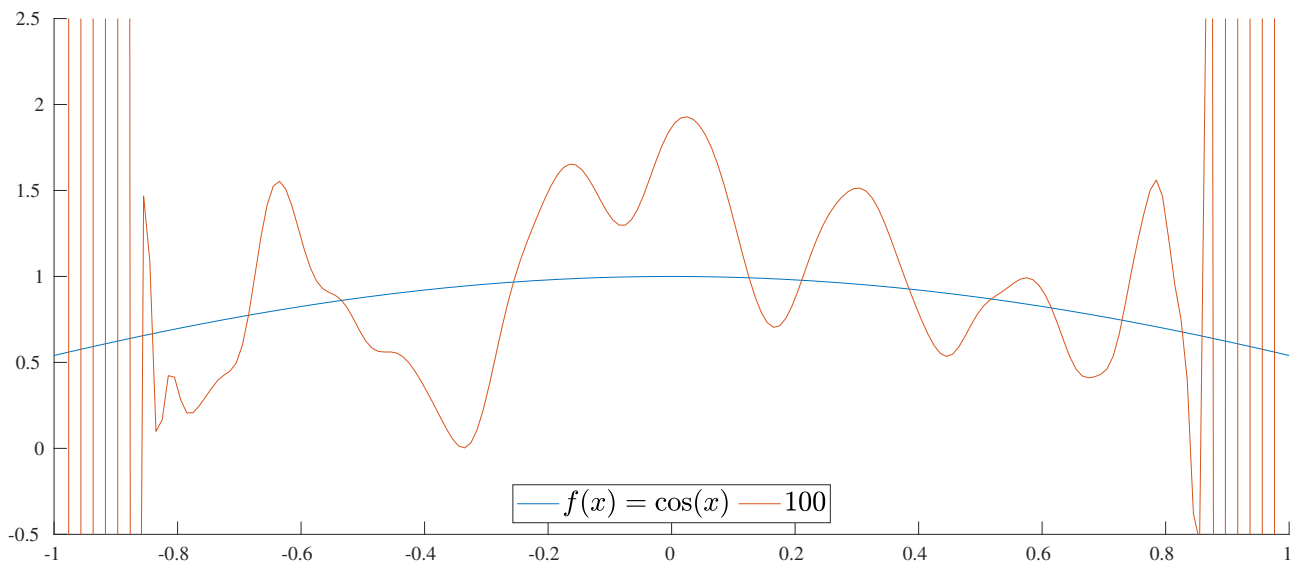
3.1 Interpoleren van een functie

3.1.1 $\cos(x)$ benaderen

De functie `interpolate` wordt gebruikt om de functie $f(x) = \cos(x)$ te benaderen. De coëfficiënten c_k worden meegegeven aan de functie `evalCheb` om de interpolant $g(x)$ te kunnen plotten. Voor de interpolatiepunten x_k worden eerst equidistanten punten gekozen. De



Figuur 3: Benaderingen van de functie $f(x) = \cos(x)$ met een lage n en equidistante interpolatiepunten

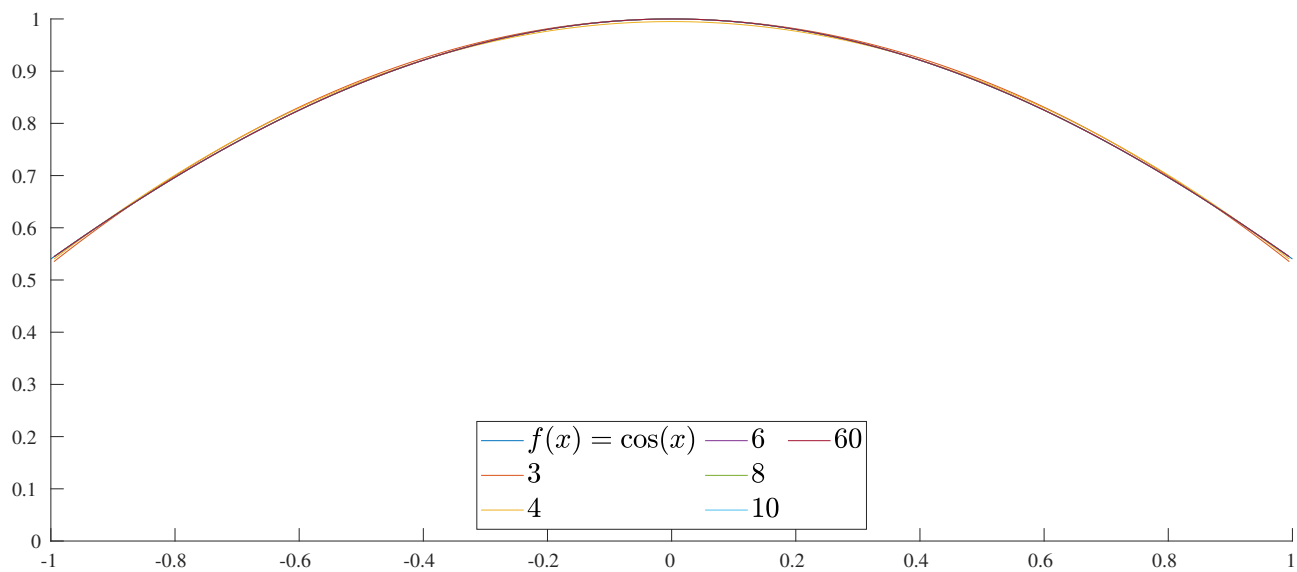


Figuur 4: Benadering van de functie $f(x) = \cos(x)$ met een hoge n en equidistante interpolatiepunten

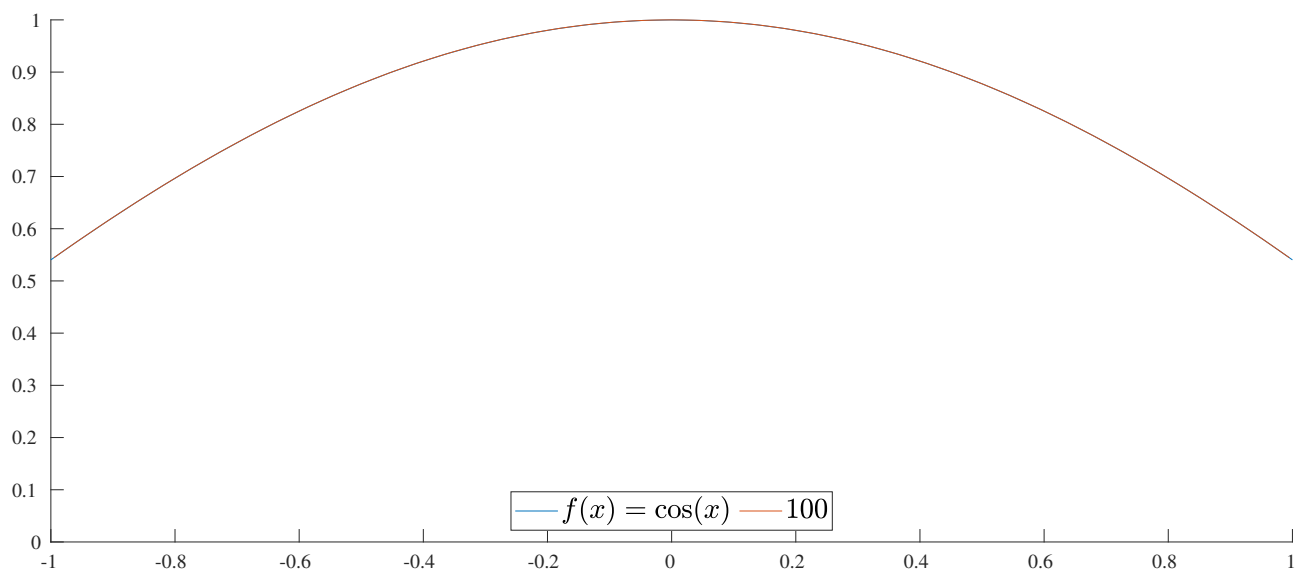
resultaten worden voor verschillende n -waarden getoond in figuren 3 en 4. Naarmate n stijgt vertoont de benadering aan de randen meer afwijking ten op zichte van $f(x)$. Bij een te hoge n zoals in figuur 4 is de benadering heel slecht doordat de gebruikte methode niet stabiel is bij equidistante interpolatiepunten. Door de interpolatiepunten zo te kiezen dat ze de nulpunten zijn van de k -de Chebyshev veelterm, wordt de methode wel stabiel. Deze nulpunten zijn van de vorm:

$$x_i = \cos\left(\frac{\pi(2i-1)}{2k}\right), \quad i = 1, \dots, k$$

In figuren 5 en 6 zijn de interpolatieveeltermen met deze interpolatiepunten te zien. De fout aan de rand is ook veel kleiner zoals in figuur ... te zien is.



Figuur 5: Benadering van de functie $f(x) = \cos(x)$ met een lage n



Figuur 6: Benadering van de functie $f(x) = \cos(x)$ met een hoge n