F. Bornemann, C. Ludwig 29. October 2015

Assignment 5
A *circulant matrix* $C \in \mathbb{C}^{m\times m}$ has the form

$$C = \begin{pmatrix} c_1 & c_m & \cdots & c_3 & c_2 \\ c_2 & c_1 & \ddots & \vdots & c_3 \\ \vdots & c_2 & \ddots & c_m & \vdots \\ c_{m-1} & \vdots & \ddots & c_1 & c_m \\ c_m & c_{m-1} & \cdots & c_2 & c_1 \end{pmatrix}.$$

Let $c = (c_1, \ldots, c_m)^T$ be the first column of C.

(a) Write a naïve function to calculate $C \cdot v$ using the inputs $c \in \mathbb{C}^m$ and $v \in \mathbb{C}^m$.

(b) Use the fact that C can be diagonalized by the discrete Fourier matrix F_m

$$F_m\, C\, F_m^{-1} = D = \operatorname{diag}(F_m c)$$

to write an efficient function to calculate $C \cdot v$ given c and v as input.

(c) Compare the complexity of the two functions of (a) and (b).

Solution
(a) see zip-file.
(b) We have

$$z = Cv = F_m^{-1} D F_m v$$

and hence the function

```
function z=A5b(c,v)
z = ifft( fft(c).*fft(v) );
end
```

(c) In (a) the complexity is $O(m^2)$ in (b) it is $O(m \log m)$.

Assignment 6
Write a function to generate n Chebyshev points in the interval $[a, b]$.
Hint: Use an affine transformation of $[-1, 1]$ to $[a, b]$ and the known formula for the Chebyshev points on $[-1, 1]$.

Solution
For the interval $[-1, 1]$ $n + 1$ Chebyshev nodes are given by $x_j = \cos(\pi j/n)$ for $j = 0, \ldots, n$. We need a affine bijective transformation $T\colon [-1, 1] \to [a, b]$.
The needed transformation is given by

$$[a, b] \ni y = T(x) = x\frac{b-a}{2} + \Big(a + \frac{b-a}{2}\Big).$$

So we have the function

```
function pts=A6(a,b,n)
% generate n Chebyshev points in interval (a,b).
assert( a<=b );
nm1 = n-1;
x = cos( pi/nm1*(nm1:-1:0) );  % Chebyshev points in (-1,1)
if a>0 || b<0
    len_half = (b-a)/2;
    pts = x*len_half + (a+len_half);
else
    m = (a+b)/2;
    pts = x*(b-m)+m;
end
end
```

Remark: For $0 < a < b$ we don't use `(a+b)/2` in order to avoid overflow.

Assignment 7
Consider the task of finding an interpolation polynomial for the function

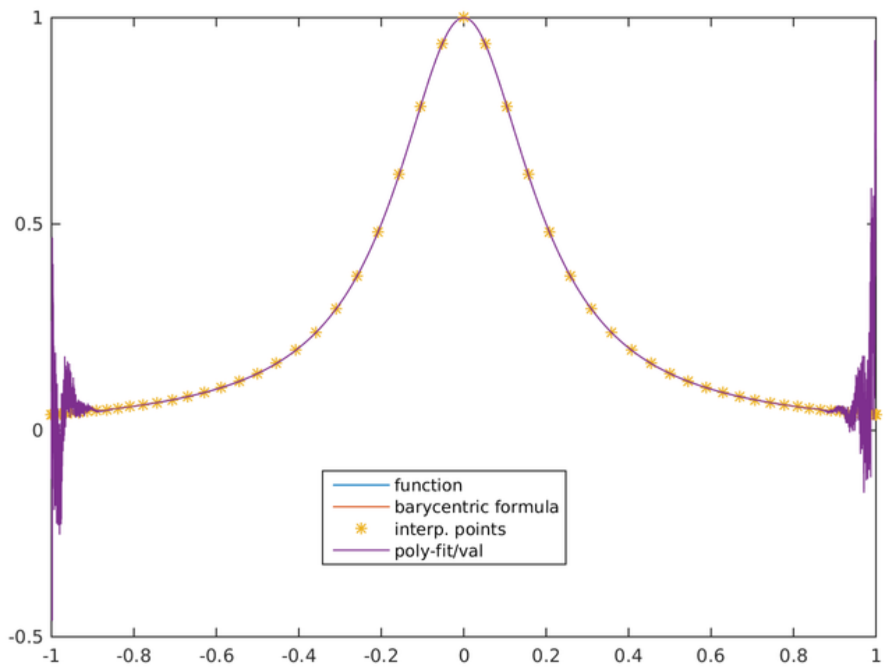
$$f(x) = \frac{1}{1+25x^2}$$

given the values $y_j = f(x_j)$ for the Chebyshev points $x_j = \cos(j\pi/n)$ for $j = 0, \ldots, n$. Extend the Matlab script `ChebyshevInterpolation2.m` to use a second algorithm

$$(x_j, y_j) \longmapsto P = \texttt{polyfit}(x, y, n) \longmapsto yy = \texttt{polyval}(P, xx)$$

to also plot the interpolation polynomial calculated with `polyfit`/`polyval` for $n \in \{50, 60, 70\}$.
Explain the results.

Solution



see zip-file for the code. `polyfit` finds the coeffients a_k of $p(x) = a_0 + a_1 x + \cdots + a_n x^n$ w. r. t. the monomial functions $x \mapsto x^k$. The last map `polyval` is evaluating p given as input the coeffients a_k. If we perturb the a_k then we have

$$p(a + \delta a, x) - p(a, x) = p(\delta a, x) = \sum_k \delta a_k x^k.$$

The condition number of this last map can be a lot larger than the condition number of the interpolation task (at Chebyshev points) which is smaller than 5 for $n \leq 70$. Thus: Even for Chebyshev points `polyfit`/`polyval` is *unstable*.