

Pràctica 3: Interpolació polinòmica i integració numèrica

Arnau Mas

24 d'Abril 2018

Problema 1

L'objectiu d'aquest problema és interpolar la funció $f: \mathbb{R} \rightarrow \mathbb{R}$ donada per

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

a l'interval $[-1, 1]$ mitjançant el polinomi interpolador de Lagrange. Es faran servir dos conjunts de nodes diferents per a realitzar la interpolació. En primer lloc, n nodes equidistants dins de l'interval, és a dir, donats per

$$x_k = -1 + \frac{2k}{n}$$

per $k \in \{0, \dots, n-1\}$. Els altres nodes que farem servir seran nodes de Chebyshev, definits com

$$x_k = \cos \frac{2k+1}{n+1} \frac{\pi}{2}$$

per $k \in \{0, \dots, n-1\}$. Es proposa fer la interpolació fent servir 4, 8, 16, 32 i 64 nodes.

El programa `nodes.c` genera una llista amb n nodes equidistants o de Chebyshev. Aquesta llista serveix d'entrada per al programa `prob1.c`, que implementa el mètode de diferències dividides de Newton per a calcular els coeficients del polinomi interpolador de Lagrange per als nodes donats. A més avalua aquest polinomi aplicant la regla de Horner. El fitxer auxiliar `diferencies_dividides.c` conté la implementació de funcions auxiliars per aquests programes, com el càlcul de diferències dividides fent ús de l'expressió recursiva així com una implementació de la regla de Horner per avaluar un polinomi.

— * —

Per a tenir una idea del màxim error que es comet en cada cas hem avaluat tant la funció com el polinomi en un nombre elevat de punts. En particular en els punts donats per $x_k = -0.989 + k \cdot 0.011$ per $k \in \{0, \dots, 180\}$, que són 181 punts repartits de forma equidistant a l'interval on estem interpolant.

Problema 2

Considerem la funció de Bessel de primera espècie d'ordre zero, $J_0(x)$. Volem estimar el valor en l'absissa x^* tal que $J_0(x^*) = 0$, per això realitzarem interpolació inversa de graus 1, 3 i 5, és a dir, si $(x, J_0(x))$ són els punts de la funció de Bessel que ens donen, nosaltres prendrem $(J_0(x), x)$ com a punts per a realitzar la interpolació. De manera que si $p(x)$ és el nostre polinomi interpolador, al avaluar $p(0)$ obtindrem una aproximació de x^* .

El programa `prob2.c` donats uns punts com a entrada, calcula el polinomi interpolador de Lagrange amb el mètode de les diferències dividides de Newton i l'avalua en el punt zero amb la regla de Horner. Els resultats que hem obtingut al realitzar les interpolacions demanades han sigut els següents:

Taula 2.1: Interpolant valors positius de $J_0(x)$ més pròxims al canvi de signe de la funció.

Grau	x^*	$J_0(x^*)$
1	2.404728613882804	$5.03291522479392 \times 10^{-5}$
3	2.404822718113948	$1.47416266795862 \times 10^{-6}$
5	2.404825294785460	$1.36489238329446 \times 10^{-7}$

Taula 2.2: Interpolant valors negatius de $J_0(x)$ més pròxims al canvi de signe de la funció.

Grau	x^*	$J_0(x^*)$
1	2.400077241947102	$2.46750381340249 \times 10^{-3}$
3	2.404149375353531	$3.51087705194527 \times 10^{-4}$
5	2.404216734868258	$3.16108843546827 \times 10^{-4}$

Taula 2.3: Interpolant valors de $J_0(x)$ simètrics al canvi de signe de la funció.

Grau	x^*	$J_0(x^*)$
1	2.404927513002775	$-5.29287203952310 \times 10^{-5}$
3	2.404824021911155	$7.97298995297100 \times 10^{-7}$
5	2.404825653043717	$-4.94996456864148 \times 10^{-8}$

Observem que el millor resultat l'obtenim amb la interpolació de grau 5 de valors simètrics al voltant del canvi de signe de $J_0(x)$. En general les millors són les interpolacions amb els valors positius i amb els simètrics, i la interpolació amb valors negatius és prou dolenta en comparació.

De fet ja podíem esperar des del principi que la millor interpolació fos la simètrica, ja que és l'única que en l'interval de punts interpoladors conté al x^* , i en general si tenim els punts interpoladors en un interval $[a, b]$ i volem avaluar el polinomi interpolador $p(x)$ en un punt fora d'aquest interval, l'error obtingut pot ser molt gran.

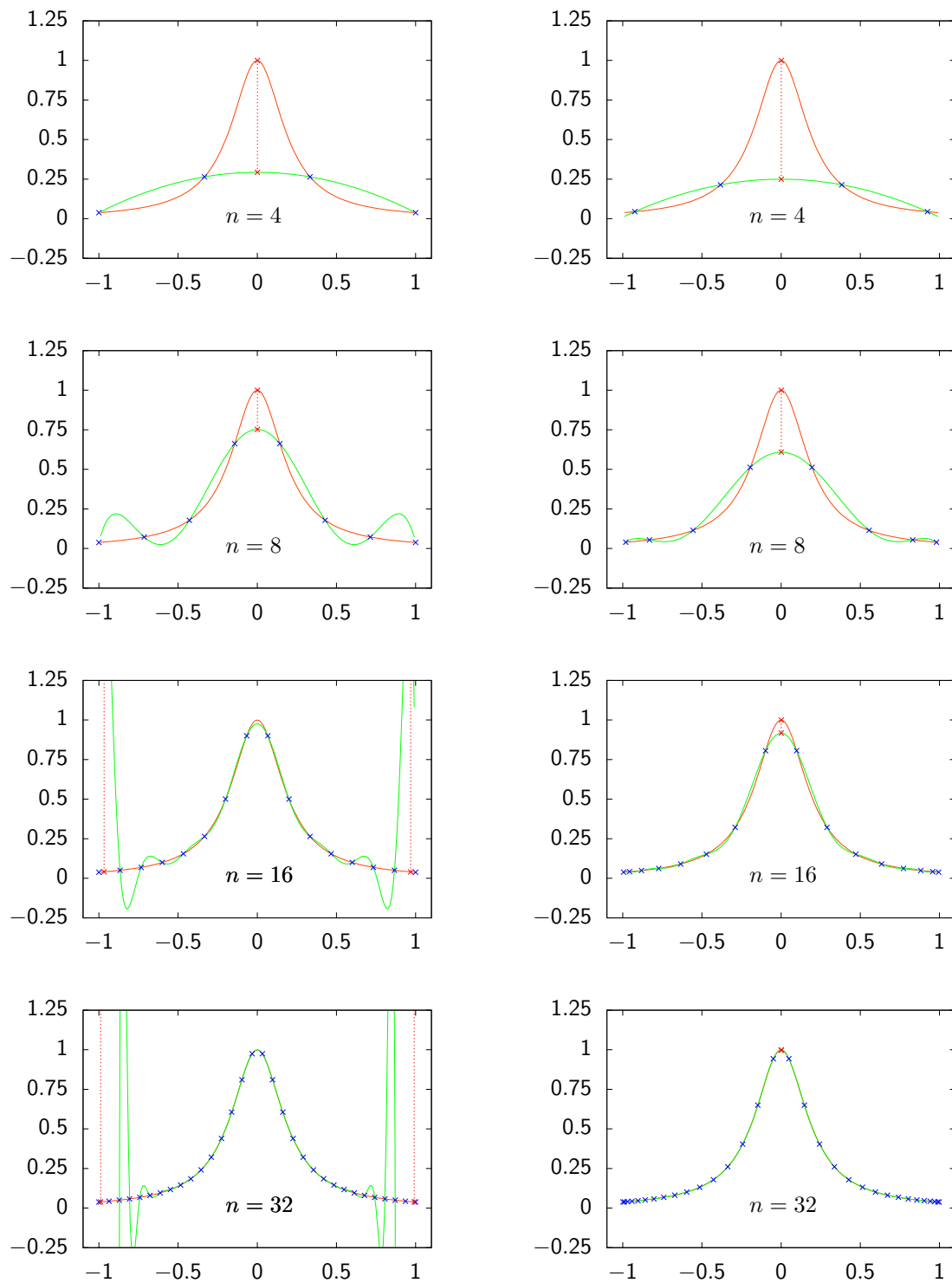


Figura 1.1: Resultat d'interpoliar f fent servir n nodes equidistants (esquerra) i n nodes de Chebyshev (dreta)