Newtonova metóda interpolácie

Miroslav Kurka
Dept. of Biophysics
Pavol Jozef Šafárik University in Košice
Slovakia

12. novembra 2022

1 Úloha

(a) Majme tabuľku nameraných dát. Nájdite Newtonov interpolačný polynóm s doprednými diferenciami. Upravte ho na tvar $P(x) = a_0 + a_1x + a_2x^2 + ... + a_nx^n$ a porovnajte s aproximačným polynómom 6-tého stupňa zo zadania č.1. Ako sa líšia?

t_i	2	2.5	3	3.5	4	4.5	5
y_i	7.14	6.56	5.98	5.55	5.71	6.01	6.53

(b) Generujte maticu náhodných čísiel 1000 krát 1000 ($A = rand(1000, 1000)$) a vypočítajte hodnotu polynómu $P(A)$ jednak priamo a jednak pomocou Hornerovej schémy. Porovnajte rýchlosť výpočtov s využitím príkazov tic a toc.

1.1 Teória

Majme tvar polynomu:

$$N_p = C_0 + C_1(x - x_0) + C_2(x - x_0)(x - x_1) + ... + C_n(x - x_0)...(x - x_{n-1})$$

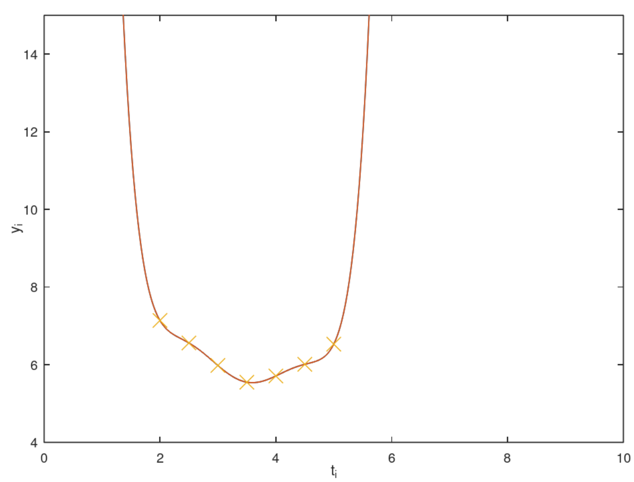
Tento polynóm definujeme ako Newtonov polynóm. Pri riešení úlohy dosadzujeme interpolačné body do polynómu. Na získanie správneho polynómu potrebujeme zistiť koeficienty. V jednoduchšiom prípade ak interpolačné body sú navzájom vzdialené tou istou vzdialenosťou h, nazyvame ich ekvidištančné a je možné využiť metódu dopredných diferencií, kde postupne výpočítavame dopredné diferencie spôsobom $\Delta y_0 = y_1 - y_0, \Delta^2 y_0 = \Delta y_1 - \Delta y_0, \Delta^2 y_1 = \Delta y_2 - \Delta y_1$. Následne dosádzame tieto hodnoty do $C_i = \frac{\Delta^i y_0}{i!(x_1 - x_0)^i}$. Takto získame koeficienty pre náš polynóm.[1]

1.1.1 Algoritmus

V prvom kroku inicializujeme maticu dopredných diferencií, kde prvý stĺpec sú hodnoty y_i. Ďaľej napĺňame maticu do trojuholníkového tvaru, ten dosiahneme indexovanim cez druhú for slučku od 1 po $n - j + 1$, kde $+1$ je kvôli indexovaniu matlab polí od 1. V ďalšom kroku inicializujeme pole koeficientov a faktoríal cez vsatvanú funkciu. Následne napĺňame pole koeficientov, prvá hodnota je zo vzorca rovná y_1, ďalej cez for slučku výpočítavame daľšie hodnoty podľa vzorca $C_i = \frac{\Delta^i y_0}{i!(x_1 - x_0)^i}$, kde -1 sú opäť kvôli indexácií. Z balíčka *symbolic* využijeme funkciu *syms*, ktorá slúži na definíciu funkcií. Zadefinujeme newtonov polynóm s doplnenými koeficientami a interpolačnými bodmi. Funkciou simplify zjednodušíme. Funkciou *matLabFunction* "castneme"typ premennej do function handle, tento krok je potrebný na druhú časť zadania. V druhej časti vytvoríme maticu 1000x1000 náhodných čísel. V prvom kroku prechádzame maticu a dosadzujeme cez dor slučky čísla do polynómu, kde funckiou feval výpočítavame polynóm. V druhom kroku robíme to isté len využívame funckiu horner na úpravu polynomu do hornerového tvaru. Pri obidvoch metódach meriame čas pomocou tic toc funckie

1.2 Výsledky

Z grafu vidíme, že interpolačný polynóm prechádza presne bodmi v čom sa líší s polynómom z prvého zadania. Polynóm LSM[1] z prvého zadania neprechádza presne zadanými bodmi, jedná sa o aproximáciu nie o interpoláciu. Newtonov polynóm prechádza bodmi ale má napríklad nevýhodu vlnivosti. Pri meraní



rýchlosti výpočtov ak je veľkosť matice 10x10 tak je rozdiel metód pár stotín sekúnd avšak ak je veľkosť matice 1000x1000 rozdiel časov je razantný. Pre klasický výpočet je to 17.6557 sekúnd. Pre výpočet cez hornerov tvar je tento čas zredukovaný na 0.000777006 sekúnd.

[1]Polynóm v grafe nie je z dôvodu, že mi nepresne vychádza v prvej úlohe.

1.3 Záver

Dosiahli sme dostatočne presného Polynómu na interpoláciu našich dát. Rozdiel medzi LSM a Newtonovou metódu je v prístupe metód k bodom zadaných dát. Pri skumaní rýchlosti výpočtov bolo zistené, že za pomoci hornerovеho tvaru polynómu je dosiahnúté výrazné zrýchlenie a to z dôvodu, že pri hornerovom tvare zredukujeme počet násobení z $(n^2 + n)/2$ na n násobení[2].

Literatúra

[1] Buša, J., Pirč, V. and Schrötter, Š. (no date) *Numerické metódy, pravdepodobnosť a matematická štatistika*, p. 263.

[2] `https://en.wikipedia.org/wiki/Horner%27s_method#Efficiency`