Tema 3. Pete Solare

Responsabil:
Florin Pop (florin.pop@cs.pub.ro)

04.04.2011
Termen de predare: 21.04.2011

1 Descriere Generala

Tema de casa are ca subiect analiza ciclul activitatii solare, care este o variatie periodica a radiatiei solare, variatie care determina schimbarile climatice pe Pamant. Ciclul solar are o durata medie de 11,2 ani, insa se cunosc cicluri solare cu durate intre 8 si 15 ani. Se presupune ca un ciclu solar este determinat de campul magnetic al Soarelui, care se inverseaza o data la 11 ani, un ciclu magnetic complet durand de fapt 22 de ani. Activitatea solara este caracterizata prin numarul de pete solare, numarul de eruptii solare si radiatia solara.

Fisierul `sunspot.dat` (care poate fi incarcat cu `load sunspot.dat`) contine numarul anual de pete solare in perioada 1700 - 1999 ($N = 300$ de ani). Dupa incarcare, variabila `sunspot` este o matrice cu doua coloane. Prima reprezinta anul, a doua numarul de pete solare inregistrate in anul respectiv (numarul este real, deoarece reprezinta rata de emisii spontane care stau la baza petelor solare).

Densitatea de energie emisa in timpul maximului unui ciclu de 11 ani poate fi modelata ca o functie periodica $f : [-\pi, \pi] \to R$,

$$f(x) = \frac{\exp(a\cos(x))}{2\pi I_0(a)},$$

unde a este un parametru ce descrie netezimea functiei, iar $I_0(a)$ este functia Bessel, disponibila in MATLAB (Octave) prin functia `besseli(0,a)`. Pentru aceasta tema de casa se va considera $a = 3$.

2 Obiective

Scopul temei de casa este evaluarea polinoamelor de interpolare si aproximare si a performantelor acestora. Principalele obiective ale temei de casa sunt:

- Analiza convergentei polinoamelor de interpolare.
- Compararea polinoamelor de interpolare si aproximare pentru cazul continuu si pentru cazul discret.
- Estimarea erorii comise in procesul de interpolare.

- Analiza unei problema reale: *periodicitatea petelor solare.*

Notiunile necesare rezolvarii acestei teme de casa sunt: interpolare Lagrange si Newton; interpolare cu functii spline (liniare si cubice de clasa C^2, naturale si tensionate); interpolare trigonometrica (Fourier); interpolare folosing polinoame Cebasev; programare in MATLAB (Octave).

3 Convergenta Polinoamelor de Interpolare

3.1 Evaluare continua [100p]

Scopul primei parti a temei de casa este analiza convergentei algoritmilor de interpolare pentru functia $f(x)$ descrisa in sectiunea 1. Mai concret se va studia daca si cat de repede eroarea de interpolare converge la zero, atunci cand numarul de noduri de interpolare creste. Se vor studia mai multe tipuri diferite de interpolari:

1. Polinomul de interpolare Lagrange;
2. Polinomul de interpolare Newton;
3. Interpolarea folosind functii spline liniare pe intervale, dererminate prin intermediul conditiilor de interpolare Hermite;
4. Interpolarea cu functii spline cubice, naturale;
5. Interpolarea cu functii spline cubice, tensionate, unde derivatele la capete se impun ca fiind $f_1' = \frac{f(x_2)-f(x_1)}{x_2-x_1}$ si $f_N' = \frac{f(x_N)-f(x_{N-1})}{x_N-x_{N-1}}$;
6. Polinomul de aproximare trigonometric (Fourrier). Se vor aplica formulele discrete de calcul ale coeficientilor.

Pentru un anumit interpolant, $p(x)$, acesta poate fi evaluat folosind un numar de puncte $\tilde{x} = linspace(-\pi, \pi, N+1)$ pentru $N = 1000$ si apoi comparat cu valoarea functiei. De asemenea, putem calcula o estimare pentru norma euclidiana a eroarii de interpolare prin insumarea eroarii asupra fiecarei evaluari:

$$E[p(x), N_k] = \left[h \sum_{i=1}^{N+1} |f(\tilde{x}_i) - p_{N_k}(\tilde{x}_i)|^2 \right]^{1/2}, h = \frac{2\pi}{N+1}.$$

Pentru un numar diferit de noduri de interpolare, $N_k = 2^k$, $k = 2, 3, ...$, se va calcula eroarea $E[p(x), N_k]$. Se va scrie o functie MATLAB (specifica pentru functia $f(x)$) care primeste ca parametru tipul interpolantului si va intoarce numarul de noduri de interpolare, N, pentru care interpolantul converge. Daca acesta nu converge se va intoarce `inf`. Un interpolant $p(x)$ converge daca $E[p(x), N_k]$ este descrescatoare (dupa un numar de pasi initial) o data cu cresterea lui k si, dupa un numar de pasi N_k, daca $|E[p(x), N_{k+1}] - E[p(x), N_k]| < eps$. In caz contrar, interpolantul nu converge.

Fisierul `eval_interpolator_c.m` care implementeaza functia ceruta va fi de forma:

```
function N = eval_interpolator_c(tip, eps)
% eval_interpolator - determina cat de repede converge un polinom
    de interpolare
% tip - 1 lagrange
%       2 newton
%       3 linear spline
%       4 natural
%       5 cubic spline
%       6 fourrier
% eps - toleranta acceptata pentru convergenta
```

ATENTIE! Forma propusa este orientativa. Puteti aduaga atat valori noi de intrare, daca sunt necesare, cat si valori de iesire.

Comentati pe cat de repede converge eroarea catre zero si care interpolant este cel mai exact. Toate comentariile vor fi incluse in prima sectiune a fisierul `readme.txt`.

3.2 Evaluare discreta [40p]

Pentru a doua parte a temei de casa se considera functia discreta f_i ale carei $N(=300)$ valori se vor incarca din fisierul `sunspot.dat`. Se vor analiza toti cei 6 interpolanti descrisi in sectiunea anterioara.

Va trebui sa descrieti modul de evaluare a erorii, explicand de ce nu se poate aplica aceeasi metoda ca in cazul continuu. Toate explicatiile, precum si formula de calcul a erorii vor fi incluse in a doua sectiune a fisierul `readme.txt`.

Rescrieti functia de la evaluarea continua pentru evaluarea discreta, de forma `function N = eval_interpolator_d(tip, eps)`.

ATENTIE! Forma propusa este orientativa. Puteti aduaga atat valori noi de intrare, daca sunt necesare, cat si valori de iesire.

3.3 Evaluarea temei de casa [20p]

Se cere scrierea unui script de test, `test.m`, care va afisa o matrice de 2 linii si 6 coloane cu valorile lui N determinate pentru toti cei 6 interpolanti propusi pentru a fi evaluati in cazul continuu (prima linie din rezultat) si in cazul discret (a doua linie din rezultat).

3.4 Reprezentarea grafica [40p]

Se cerea realizarea unui **singur** grafic cu doua subgrafice (se va folosi functia `subplot`[1]) pentru reprezentarea celor doua cazuri analizate. Primul subplot va reprezenta grafic functie $f(x)$ si cei 6 interpolanti folosind ca noduri de reprezentare punctele alese pentru evaluarea erorii. Al doilea subplot va reprezenta functia f_i si polinoamele de interpolare in cele 300 de puncte in care se cunosc valorile functiei.

Realizarea acestui grafic se va face prin apelul unui script `test_grafic.m`

[1] `http://octplot.sourceforge.net/subplots.html`

4 Detalii de implementare

Urmatoarele detalii privind implementarea temei de casa sunt obligatorii:

- Existenta celor doua fisiere de test `test.m` si `test_grafic.m` care vor avea comportamentul descris.
- Tema de casa va contine doar cele 4 fisiere `.m` mentionate. Toata implementarea se va face intr-o singura functie si nu vor exista fisiere sau functii aditionale. Pentru aceasta puteti folosi instructiunea `switch`[2] in care `case`-urile vor avea valorile de la 1 la 6 (pentru fiecare polinom de interpolare).
- Existenta fisierului `readme.txt` cu explicatiile solicitate este obligatorie.
- **NU** se vor folosi functii MATLAB (Octave) specifice pentru interpolari.
- Se pot folosi functiile existente pentru polinoame[3]: polyout, polyval, conv, deconv, roots, polyderiv, polyinteg.

[2] `http://www.network-theory.co.uk/docs/octave3/octave_87.html`
[3] `http://en.wikibooks.org/wiki/Octave_Programming_Tutorial/Polynomials`