Universitat autònoma de Barcelona

Grau en matemàtiques

Memòria de la pràctica 3

Mètodes numèrics

Aleix
Suárez Sayabera

Juny 2024

Índex

1 Introducció i objectius

Aquesta pràctica consisteix a proporcionar una rutina que sigui capaç d'avaluar en un punt, tant el polinomi interpolador d'una taula de valors com el valor de la seva derivada. Tot això amb el mètode de Neville.

Un cop feta aquesta rutina, se'ns demana utilitzar-la per calcular zeros, màxims i mínims de taules astronòmiques.

2 Implementació de la rutina per avaluar el polinomi interpolador

La interpolació consisteix en, si tenim uns certs punts de suport $\{(x_0, y_0), \ldots, (x_n, y_n)\} \subseteq \mathbb{R}^2$, construir una funció que aproximi els valors mitjans d'aquests punts de suport. En aquest cas, volem que donats aquests punts de suport, avaluar una funció interpoladora, que en aquest cas serà un polinomi.

2.1 El mètode de Neville

El mètode de Nevill consisteix a avaluar el polinomi interpolador utilitzant una recurrència.

Suposem que tenim punts de suport $\{(x_0, y_0), \ldots, (x_n, y_n)\} \subseteq \mathbb{R}^2$. On per a un subconjunt de $k+1$ subindexos $\{i_0, \ldots, i_k\} \subseteq \mathbb{N}$, denotem per $P_{i_0,\ldots,i_k}$ el polinomi de grau k que interpola els punts $\{(x_{i_0}, y_{i_0}), \ldots, (x_{i_k}, y_{i_k})\}$. El mètode de Neville per avaluar $P_n(x)$ es basa en aquesta recurrència:

$$P_{i_0,\ldots,i_k}(x) = \frac{(x - x_{i_0})P_{i_1,\ldots,i_k}(x) - (x - x_{i_k})P_{i_0,\ldots,i_{k-1}}(x)}{x_{i_k} - x_{i_0}}, \tag{1}$$

La recurrència es pot derivar,obtenint la següent recurrència per avaluar $P'n(x)$:

$$P'_{i_0,\ldots,i_k}(x) = \frac{P'_{i_1,\ldots,i_k}(x) - P'_{i_0,\ldots,i_{k-1}}(x)}{x_{i_k} - x_{i_0}} + \frac{(x - x_{i_0})P'_{i_1,\ldots,i_k}(x) - (x - x_{i_k})P'_{i_0,\ldots,i_{k-1}}(x)}{x_{i_k} - x_{i_0}}. \tag{2}$$

2.2 Programació del mètode de Neville

La rutina per avaluar el polinomi interpolador ha de tindre un prototipus específic:

```
double intnev_avald(double x, int n, double xi[], double yi[], double *p, double *dp);
```

La funció `intnev_avald` rep els següents paràmetres:

- `double x`:
  És el valor de x en el que volem avaluar el polinomi interpolador.
- `int n`:
  És l'índex del total de punts de suport amb el qual calcularem el polinomi interpolador.
- `double xi[]`:
  És el vector on es guardaran els diferents valors de la component x dels punts de suport.
- `double yi[]`:
  Similarment al vector anterior, solament que en aquest cas estaran les components y dels punts de suport.
- `double *p`:
  És l'apuntador on es desa el valor de $P(x)$.
- `double *dp`;
  Com al cas anterior, és l'apuntador on es desa el valor de la derivada de $P'(x)$.

Per tal de programar això a C, he començat declarant dos apuntadors de mida $n+1$, que són respectivament `P` i `dP`, i els he inicialitzat amb els valors de `yi[]` per a `P` i amb zeros per a `dP`, ja que les derivades s'inicialitzen així al mètode de Neville, tots els $P'i$ són zero.
Després de tindre aquests apuntadors declarats, he començat a escriure el bucle `for` on es calculen per la recurrència de Neville els diferents valors del polinomi interpolador i la seva derivada.
Finalment, he afegit la condició que `*p` i `*dP` puguin ser `NULL`.

2.3 Validació de la rutina

Per validar aquesta rutina, he creat un programa apart on s'avalua el polinomi interpolador de la funció $\sin(x) \cdot \cos(x)$, en l'interval $[0, \pi]$.
Al programa he dividit l'interval en 9 punts, on aquest llegeix per pantalla el valor x, hi s'imprimeix els resultats de l'interpolació, els valors reals de la funció, i els error. Alguns dels resultats han sigut els següents:

x	error en $f(x)$	error en $f'(x)$
0.1	0.0005	0.03263
0.25	0.0003	0.0818
0.5	0.0001	0.1265
1	$3.8173e^{-05}$	0.0363
1.5	$1.31154e^{-05}$	0.4543
2.5	$7.8854e^{-05}$	0.1621

Taula 1: Valors de l'error segons x

3 Equinoccis de primavera i tardor de l'any 2024

Per tal de donar un ús a la rutina, se'ns demana calcular a partir d'una taula de dades astronòmiques els equinoccis de primavera i tardor de l'any 2024, i els solsticis d'estiu i hivern d'aquest mateix any.
Començant pel càlcul dels equinoccis. Aquests són els moments en que el sol està just a sobre de l'equador terrestre, on el dia i la nit duren el mateix. Aquest fenomen passa dos cops a l'any, un cop a la primavera i un altre a la tardor.

Per tal de calcular aquests equinoccis se'ns ha proporcionat amb un fitxer de text on hi han dades astronòmiques, aquestes dades recullen les posicions del sol, en un sistema de coordenades tal que el pla xy és el pla de l'equador terrestre. Aquest fitxer recull les dades de la posició del sol començant des de l'1 de desembre de 2023 fins al 31 de gener del 2025. Les dades de la posició del sol es recullen en graus de declinació del sol respecte l'equador terrestre.

Per tal de calcular els equinoccis cal fer una utilitat que busqui el dia d'aquesta taula on el grau de declinació sigui zero, ja que grau de declinació zero implica que el sol està just a sobre de l'equador terrestre, i per tant que ens trobem en un equinocci.

3.1 Programació de la utilitat

Per el calcul dels equinoccis se'ns demana programar una utilitat `intaimg` que respongui a la crida:

```
./intaimg n nimg vimg
```

On els paràmetres que rep són els següents:

- `n`:
  És el nombre de punts de suport amb el qual es construeix el polinomi interpolador.
- `nimg`:
  És el `nimg`-èsim canvi de signe de y_j `vimg` a y_{j+1} `vimg`.

- `vimg`:
  És el punt on es farà interpolació inversa per trobar el moment exacte de l'equinocci. On ens interessarà que sigui zero.

Per tal de programar aquesta utilitat, com les dades s'han de llegir per standard input, d'un fitxer amb dues columnes, on no sabem quantes files (dades) té. El primer que he fet ha sigut crear un `int m` que calculi quantes files té el fitxer. Per així, si volem utilitzar la utilitat amb més punts de suport que els que hi ha, doni un error. Seguidament, he declarat dos apuntadors `*xi` i `*yi` amb totes les dades del fitxer.
Després de tindre les dades desades, he creat un bucle per buscar l'índex del `nimg`-èsim canvi de signe, on simplement es comprova si el producte de dos punts consecutius és menor que zero, en cas afirmatiu, es guarda l'índex.
Un cop guardat l'índex, el que cal fer és la interpolació inversa, cal buscar el punt `xi` tal que en aquest moment l'angle del sol sigui nul. Per això, tenint el punt `yi` on hi ha el canvi de signe, fent interpolació inversa centrada en aquest punt, és a dir, agafaré els `n/2` punts de suport següents i anteriors a `yi`, i avaluaré el polinomi interpolador en 0. On això ho faré implementant la rutina `itnev_avald`.
Per tal de tenir una aproximació de l'error, el que faré és avaluar en el punt `x` que doni la utilitat, el polinomi interpolador centrar en aquest punt. On, si la interpolació anterior és correcte, aquest valor ha de ser molt proper a zero.

3.2 Resultats dels equinoccis

Posant a prova aquesta utilitat. Utilitzant com a dades les del fitxer `horizons.txt`. S'han trobat dos canvis de signe diferents. El primer canvi de signe correspon al dia Julià 110.12920, tenint en compte, que, en el fitxer el dia julià zero, és l'1 de desembre de 2023. Per tant, aquest dia julià correspon al dia 19 de març de 2024. I l'equinocci de primavera de 2024 és aquest dia.
El següent zero en la gràfica és al punt 296.5307214, que fent la conversió, correspon al dia 21 de setembre de 2024. I, per tant, l'equinocci de tardor de 2024 segons les dades d'horizon, serà en aquesta data.

4 Solsticis d'estiu i hivern de l'any 2024

De forma similar abans, ara se'ns demana calcular el moment dels solsticis d'estiu i hivern. Aquests dos punts són els moments en el que el sol té una declinació màxima respecte a l'equador. I són els dies en que el dia durarà més o menys, depenent de l'hemisferi.
Aquests dos moments corresponen als punts màxims de les dades de `horizons.txt`, i, per tant, per trobar aquests punts, el que volem és trobar els zeros de la derivada.

4.1 Programació de la utilitat

L'útilitat ha de respondre a la crida:

```
./intext n next tol ixrr
```

- `n`:
  És el nombre de punts de suport amb el quual es construeix el polinomi interpolador.
- `next`:
  És el `next`-èsim extrem de la corba.
- `tol i ixxr`:
  Com cal cercar un zero, cal utilitzar una rutina per fer-ho. Com implementaré la rutina `biseccio`, cal passar una tolerància `tol` i l'índex de xarrera `ixxr`.

El primer que he fet en programar aquesta utilitat ha sigut declarar la funció que li passaré a `biseccio`, per fer això, com a la pràctica passada, hem de declarar amb `struct` els paràmetres que li hem de passar a la funció, els valors dins `*prm`, en el meu programa aquesta funció es diu `f_bis`.

Com a `biseccio` hem d'avaluar el polinomi característic, la meva `f_bis` ha de rebre els paràmetres necessaris per poder utilitzar `itnev_avald` i avaluar la derivada.
Després d'aquesta declaració, per utilitzar `biseccio`, he fet com a l'inici de la utilitat anterior, he creat un paràmetre `m` que és el nombre de files de l'arxiu que estem llegint. I he guardat en dos `doubles` de mida `m` els valors de totes les dades del fitxer.
Seguidament, he creat un bucle on es busca l'índex del `next`-èsim canvi de signe en la derivada. Per fer això he fet córrer un índex que comprovava una per una les dades del fitxer, i es guardava l'índex si aquests valors del fitxer passaven de ser creixents a decreixents. Si l'índex trobava el `next`-èsim canvi de signe en la derivada, aleshores es guardava com `index_zero`.
El següent que he fet ha sigut crear un índex `k`, que serà el valor de la posició inicial de la interpolació. Tot això per centrar-la en la posició de l'índex del zero de la derivada. Aquesta l'he calculat així, `k` = `index_zero` - $n/2$.
Seguidament a això, he creat dos apuntadors `*x_intp` i `*y_intp`, on es guarden els valors dels punts de suport començant per l'índex `k`.
Finalment he aplicat `biseccio` a l'interval [`xi[index_zero]`,`xi[index_zero+1]`], i he trobat el `next`-èsim zero de la derivada.

4.2 Resultats dels solsticis

Ara per calcular els dos solsticis, només hem de buscar amb la utilitat anterior els dos extrems de la corba de horizons.

El primer zero del fitxer correspon a una data de l'any 2023, per tant, he de buscar els dos zeros corresponents a 2024, que són els dos següents al primer.
El segon zero del fitxer correspon al dia Julià 203.937500, on fent la conversió que he fet anteriorment, correspon al dia 20 de juny de 2024, i com a resultat aquesta és la data del solstici d'estiu d'aquest any.
El tercer zero correspon al solstici d'hivern, on la utilitat ha retornat el valor 386.062500, que correspon al dia 20 de desembre de 2024, que hi seria la data corresponent al solstici d'hivern de 2024.