

Séance d'exercices n°7

Interpolation polynomiale, polynomiale par morceaux et par splines cubiques

Exercice 1

Exécutez les démos *census* et *spline2d* en tapant *playshow census* et *playshow spline2d* dans la fenêtre de commande. Décortiquez le code de chacun de ces programmes afin de comprendre ce qu'ils font.

Exercice 2

Encodez et exécutez le programme suivant :

```
% Fonction sinus
t=pi/6:0.1:pi*495/180; plot(t,sin(t),'r-','LineWidth',2);
axis([0 10 -2 2]); grid on; hold on; pause
% Interpolation linéaire par morceaux
x=(pi/180)*[30 135 270 330 495];
y=sin(x); plot(x,y,'ko-'); pause
% Interpolation parabolique par morceaux
x1=x(1:3); y1=y(1:3); t1=x(1):0.01:x(3);
p1=polyfit(x1,y1,2); f1=polyval(p1,t1);
plot(t1,f1,'r:','LineWidth',2); pause
x2=x(3:5); y2=y(3:5); t2=x(3):0.01:x(5);
p2=polyfit(x2,y2,2); f2=polyval(p2,t2);
plot(t2,f2,'r:','LineWidth',2); pause
% Interpolation polynomiale de degré 4
p=polyfit(x,y,4); f=polyval(p,t);
plot(t,f,'k-'); pause
% Interpolation par splines cubiques
yy=spline(x,y,t); plot(t,yy,'b')
```

Exercice 3

Les instructions suivantes permettent de générer les coordonnées de 17 points du plan :

```
x = linspace(0,4*pi,17); y = x.*log10(2+sin(x));
```

On demande d'évaluer des approximations de l'expression $x \log_{10}(2+\sin(x))$ aux points d'abscisses entières appartenant à l'intervalle $[1,12]$ et d'évaluer les erreurs commises :

- On approche l'expression donnée par lissage polynomial au moyen de polynômes de degrés respectifs 4, 5, 6, 7 et 16.
- On utilise des approximations linéaire et parabolique par morceaux.
- On fait passer des splines cubiques naturelles par les 17 points donnés.