Université de Lille 1
Master 1 de Mathématiques 2004 / 2005
Mathématiques Assistées par Ordinateur (MAO)

Fonctions et polynômes avec Matlab

Quelques commandes

Dfinition à l'aide de tableaux.
On se donne une suite `X` de `N` abscisses. La suite des images des `X(i)` par une fonction `f` donnée peut se définir comme suit :

- Fonction 'inline'
  Cette commande permet de définir directement une fonction (à valeur scalaire) et n'ayant qu'un petit nombre de variables.

  - Version scalaire
    ```
    >> gaussienne=inline('exp(-x^2/2)')
    angle=
    Inline function:
    gaussienne(x) = exp(-x^2/2)
    >>gaussienne(2)

    ans =

    0.1353
    ```
  - Version vectorielle.
    On adapte la fonction précédente lorsque l'argument est un tableau comme suit :
    ```
    gaussiennev=inline('exp(-x.^2/2)')

    gaussiennev =
    Inline function:
    gaussiennev(x) = exp(-x.^2/2)
    >>X=0:0.1:1; >>gaussiennev(X)

    ans =

    Columns 1 through 7

    1.0000 0.9950 0.9802 0.9560 0.9231 0.8825 0.8353

    Columns 8 through 11

    0.7827 0.7261 0.6670 0.6065
    ```
    On peut dès à présent représenter graphiquement cette gaussienne :

```
Y=gaussiennev(X);% on range les valeurs dans le tableau Y
plot(X,Y)
```

Les polynômes - définition et fonctions élémentaires

- Racines d'un polynôme `roots`
  `ROOTS(C)` définit le polynôme `C(1)*X^N + ... + C(N)*X + C(N+1)`.
- Définition d'un polynôme à partir de ses racines : `poly`
  `POLY(V)` retourne les coefficients du polynôme dont les racines sont ranges dans le tableau `C` ; les coefficients sont rangés suivant les puissances décroissantes.
- Evaluation d'un polynôme `polyval` :
  `Y = POLYVAL(P,X)` retourne l'évaluation du polynôme `P` aux points du tableau `X`.
- Ajustement d'un polynôme à des données :`polyfit`
  `POLYFIT(X,Y,N)` détermine les coefficients du polynôme `P(X)` de degré `N` qui approche `P(X(i))=Y(i)` au sens des moindres carrés.
- Dérivation d'un polynôme `polyder`
  `POLYDER(P)` retourne la dérivée du polynôme `P`
- Multiplication de deux polynômes `conv`
  `C = CONV(A, B)` est la convolution des tableaux `A` et `B`, c'est à dire les coefficients du produit des deux polynômes.
- Division de deux polynômes `deconv`
  `[Q,R] = DECONV(A, B)` est la déconvolution des tableaux `A` et `B` : `B = conv(A,Q) + R`.

Exercices

1. Explorer les commandes `plot`, `fplot` et appliquez-les avec divers options possibles pour représenter une fonction de votre choix.
2. En Matlab, on peut définir un polynôme à partir de ses racines : si $r = [r_1, \cdots, r_n]$ est un vecteur ligne de taille n alors `poly(r)` sera la polynôme $\prod_{i=1}^{n}(x - r_i)$. Le calcul approché de racines de polynôme s'effectue en Matlab avec la commande `roots`.
   Construire les polynômes `poly(r)` avec `r=1:n` pour différentes valeurs de `n` (de 10 à 30, disons) et (re)calculer les racines par `roots`. Que remarque-t-on ?
3. Construction d'un polynôme d'interpolation

   (a) Polynôme d'interpolation dans la base cannonique
   Souvenez-vous du cours d'analyse numérique en DEUG :
   Soient $n+1$ réels <u>deux à deux distincts</u> x_i, $i = 1, \cdots, n+1$. Etant donnés $n+1$ réels y_i, $i = 1, \cdots, n+1$ il exite un unique polynôme $P = \sum_{i=0}^{n} a_i x^i$ tel que

$$P(x_k) = y_k,\ k = 1, \cdots, n+1 \iff \sum_{i=0}^{n} a_i x_k^i = y_k\ k = 1, \cdots, n+1.$$

On a donc

$$\begin{pmatrix} x_1^n & \cdots & x_1^k & \cdots & x_1^2 & x_1 & 1 \\ x_2^n & \cdots & x_2^k & \cdots & x_2^2 & x_2 & 1 \\ \vdots & & & & & & \vdots \\ x_n^n & \cdots & x_n^k & \cdots & x_n^2 & x_n & 1 \end{pmatrix} \begin{pmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_0 \end{pmatrix} = \begin{pmatrix} y_{n+1} \\ y_n \\ \vdots \\ y_1 \end{pmatrix}$$

La matrice de ce système est appelée *matrice de vandermonde* associée aux abscisses x_i. On la construit en Matlab à l'aide de la commande `vander` (consulter sa description avec le "help").
Ecrire un programme matlab qui permette de calculer le polynôme d'interpolation de Lagrange, en utilisant la matrice de vandermonde. Représenter sur un même graphique le polynôme d'interpolation, la fonction d'origine f et les points $(x_i, f(x_i))$.

(b) Ajustement - Moindres carrés.
Soit e_i, $i = 0, \cdots, n$ une base de $\mathcal{P}_n$. Soit $m > n$ et x_j, $j = 0, \cdots, m$ $m+1$ réels deux à deux distincts. On se propose de chercher un polynôme $p(x) = \sum_{i=0}^n a_i e_i$ tel que

$$\sum_{j=0}^m (p_n(x_j) - y_j)^2 w_j \leq \sum_{j=0}^m (q_n(x_j) - y_j)^2 w_j, \ \forall q_n \in \mathcal{P}_n.$$

Le polynôme p_n est la solution au sens des moindres carrés. Après des calculs élémentaires, on aboutit à la caractérisation de p_n :

$$\sum_{k=0}^n a_k \sum_{j=0}^m w_j e_k(x_j) e_i(x_j) = \sum_{j=0}^m w_j y_j e_i(x_j), i = 0, \cdots, n.$$

Ce système d'équations peut se réécrire sous la forme :

$$B^T B a = B^T y.$$

Explorer la commande `polyfit`. Que peut-on dire lorsque $n = m$? En déduire une nouvelle solution de l'exercice précédent. Représenter sur un même graphique le polynôme d'interpolation, la fonction d'origine f et les points $(x_i, f(x_i))$.