

Introduction à Matlab

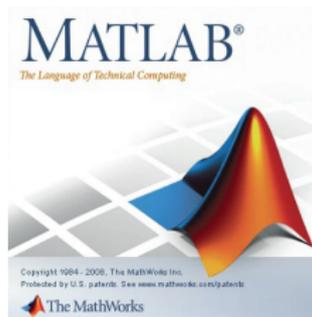
Mathieu Bouffard

Ecole Normale Supérieure de Lyon

mathieu.bouffard@ens-lyon.fr

23 Septembre 2014

Qu'est-ce que Matlab ?



- MATLAB pour MATrix LABoratory, création fin 1970s
- Ensemble d'outils et d'algorithmes efficaces permettant :
 - d'effectuer des calculs simples et complexes, des statistiques...
 - d'afficher et travailler des courbes, images, graphes 1-3D...
 - de concevoir algorithmes et modèles
 - de manipuler et traiter des fichiers
 - ...

Qu'est-ce que Matlab ?

- C'est un langage **interprété** : exécution du code au fur et à mesure (avantage pratique, pas de compilation !)
- Ecriture des instructions directement au clavier dans une fenêtre de commande...
- ...ou dans des scripts et fonctions (*m-files*) pouvant être appelées ultérieurement

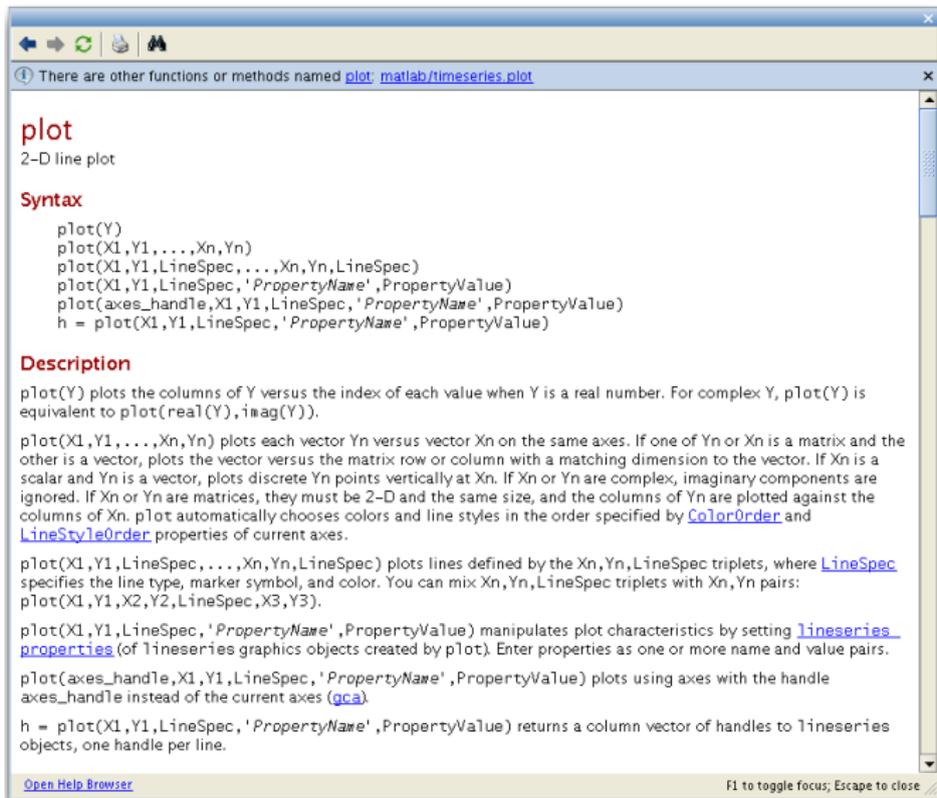
Distribution

- Inconvénient majeur : pas open source, licence chère
- mais généralement plusieurs licences disponibles par laboratoire/université
- Il existe des équivalents gratuits : Octave, Scilab

Pour quoi utiliser Matlab ?

- Matlab est particulièrement utile :
 - pour faire des calculs et des représentations graphiques simples rapidement
 - pour concevoir des algorithmes et modèles ne nécessitant pas de calculs intensifs et de grosses quantités de mémoire
 - pour ouvrir, écrire et manipuler des fichiers
- Matlab n'est **pas** recommandé :
 - pour des représentations graphiques et traitements d'images très complexes bien que cela soit possible (préférer GMT, Paraview...)
 - pour des calculs intensifs et/ou consommateurs de mémoire (C, Fortran...)

- Documentation en ligne (www.mathworks.fr/) et menu help incontournables



The screenshot shows the MATLAB help browser window for the `plot` function. The title bar reads "There are other functions or methods named `plot`: [matlab/timeseries.plot](#)". The main content area is titled "plot" and "2-D line plot".

Syntax

```
plot(Y)
plot(X1,Y1,...,Xn,Yn)
plot(X1,Y1,LineStyle,...,Xn,Yn,LineStyle)
plot(X1,Y1,LineStyle,'PropertyName',PropertyValue)
plot(axes_handle,X1,Y1,LineStyle,'PropertyName',PropertyValue)
h = plot(X1,Y1,LineStyle,'PropertyName',PropertyValue)
```

Description

`plot(Y)` plots the columns of `Y` versus the index of each value when `Y` is a real number. For complex `Y`, `plot(Y)` is equivalent to `plot(real(Y),imag(Y))`.

`plot(X1,Y1,...,Xn,Yn)` plots each vector `Yn` versus vector `Xn` on the same axes. If one of `Yn` or `Xn` is a matrix and the other is a vector, plots the vector versus the matrix row or column with a matching dimension to the vector. If `Xn` is a scalar and `Yn` is a vector, plots discrete `Yn` points vertically at `Xn`. If `Xn` or `Yn` are complex, imaginary components are ignored. If `Xn` or `Yn` are matrices, they must be 2-D and the same size, and the columns of `Yn` are plotted against the columns of `Xn`. `plot` automatically chooses colors and line styles in the order specified by [ColorOrder](#) and [LineStyleOrder](#) properties of current axes.

`plot(X1,Y1,LineStyle,...,Xn,Yn,LineStyle)` plots lines defined by the `Xn,Yn,LineStyle` triplets, where [LineStyle](#) specifies the line type, marker symbol, and color. You can mix `Xn,Yn,LineStyle` triplets with `Xn,Yn` pairs: `plot(X1,Y1,X2,Y2,LineStyle,X3,Y3)`.

`plot(X1,Y1,LineStyle,'PropertyName',PropertyValue)` manipulates plot characteristics by setting [lineseries](#) [properties](#) (of `lineseries` graphics objects created by `plot`). Enter properties as one or more name and value pairs.

`plot(axes_handle,X1,Y1,LineStyle,'PropertyName',PropertyValue)` plots using axes with the handle `axes_handle` instead of the current axes ([gca](#)).

`h = plot(X1,Y1,LineStyle,'PropertyName',PropertyValue)` returns a column vector of handles to `lineseries` objects, one handle per line.

At the bottom of the window, there is a link "Open Help Browser" and the text "F1 to toggle focus; Escape to close".