Old school Plotting (and scheming) with Matlab

Utkarsh Upadhyay
Indian Institute of Technology, Kanpur

May 21, 2009

1 What not to forget

Read table 1 for a list of commands you should know for plotting. Remember the *help function_name* will give you detailed help.

1.1 Anonymous functions

Anonymous Functions can be made for immediate use. eg.
*>> f = @(x,y) max([x y] * [[1 2]; [3 4]] + [1 5]);*
Now, f is a function which takes two values and returns the maximum of $x + 3y + 1$ and $2x + 4y + 5$.

1.2 Polynomials

Representation of polynomials is with an array which has the coefficients, eg.

$x^2 + 10x + 25$ is written as [1 10 25]

plot	The cornerstone of all plots in the history of Matlab
hist, histc	For histograms, both plotting and getting the frequency
legend	Putting a legend on the plot
axes	Show axis on a plot
grid	Set the grid settings on an existing plot
axis	Controlling the dimensions of the plot
subplots	Plotting multiple plots
hold	Allows multiple edits on the same plot
figure	Creates figure windows
drawnow	Forcing drawing of graphs (flushing the graphs immediately)
close	Close figure windows
plot3	Plotting 3D one point at a time
meshgrid	Making the XY plane matrices
surf, mesh, surface	Surface plots
contour, contour3	Contour plots
quiver, gradient	Plotting vector (gradient) fields
griddata, griddata3	Plotting with incomplete data (interpolations)
polyval	Get the value of a polynomial at a point
polyfit	Fitting a ploynomial to a list of values
plottools	Point and click solution to all your plotting needs

Table 1: Functions you need to know for Plotting

2 Simple plots

```
>> plot([3 1 2 3], [-1 0 1 -0.5], 'r')
>> hold on
>> plot([1 2 2 1], [1 1 -1 -1], 'g')
>> grid minor
>> hold off
```

Produces figure 1.

2.1 Question 1

Make the following figures:

1. Equilateral triangle
2. Rhombus

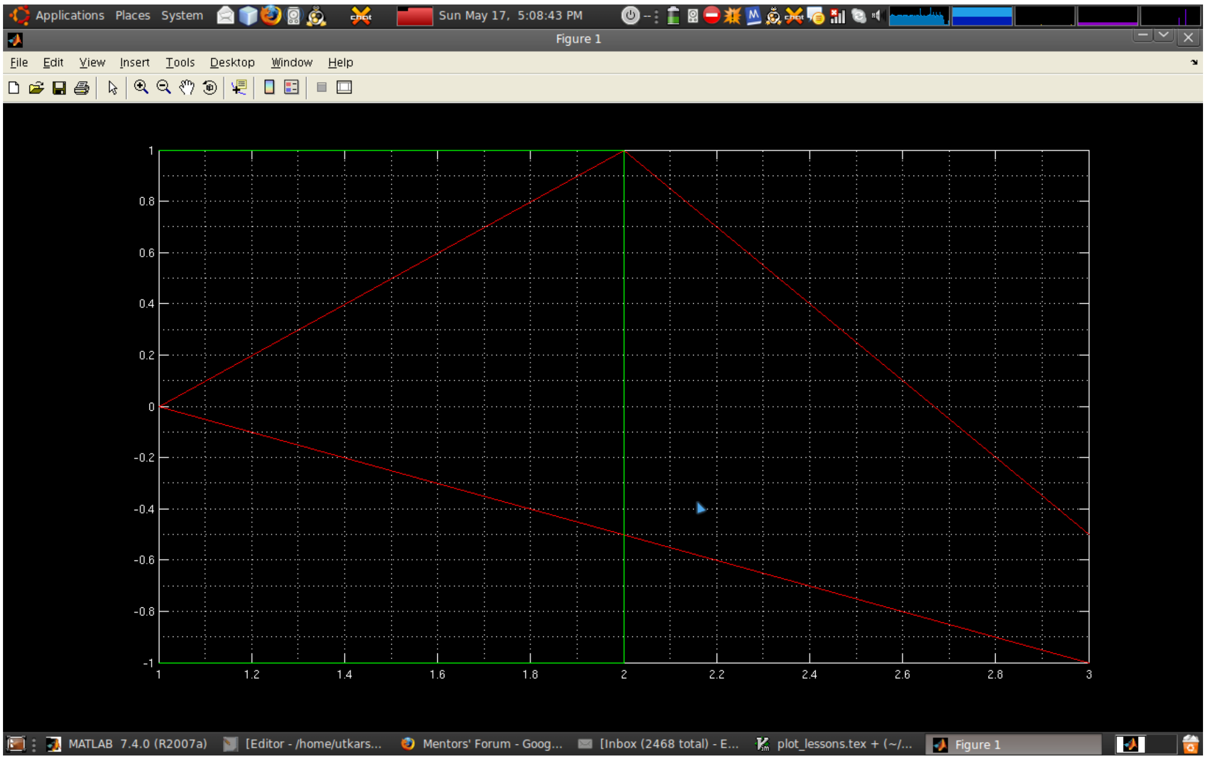

Figure 1: Figure 1

3 Plotting functions

Starting with a plot of $y = mx + c$.

3.1 Evaluating ploynomials

```
>> pol = [4 0 1];
>> x = [-2 -1.2 0 1 5];
>> y = polyval(pol, x)

y =

   17.0000    6.7600    1.0000    5.0000  101.0000

>> f = @(x) 4 * x .* x + 1

f =

    @(x)4*x.*x+1

>> y = f(x)

y =

   17.0000    6.7600    1.0000    5.0000  101.0000
```

>>

3.2 Question 2

Plot the following functions:

- $y = 2x + 1; x \in [-10, 10]$
- $y = 24\,x^4 sin(x) + 4; x \in [-100, 100]$

3.3 Question 2.1

Plot a circle of radius one.

4 3D plotting

4.1 meshgrid

Useful for easy calculation of Z matrix.

```
>> x = -2:2;
>> y = -3:3;
>> [X Y] = meshgrid(x,y)

X =

    -2    -1     0     1     2
    -2    -1     0     1     2
    -2    -1     0     1     2
    -2    -1     0     1     2
    -2    -1     0     1     2
    -2    -1     0     1     2
    -2    -1     0     1     2


Y =

    -3    -3    -3    -3    -3
    -2    -2    -2    -2    -2
    -1    -1    -1    -1    -1
     0     0     0     0     0
     1     1     1     1     1
     2     2     2     2     2
     3     3     3     3     3

>>
```

Now Z can be calculated comparatively easily, using Vectorization and dot operators. For $z = x\,e^{y^2-xy}$, we have:

```
>> Z = X .* exp ( Y .^ 2 - X .* Y );
>> surf(X,Y,Z);
>> contour(X,Y,Z,100);
>> contour3(X,Y,Z,100);
>> mesh(X,Y,Z);
>> surf(X,Y,Z);
```

4.2 Question 3

Plot $z = x\,e^{y^2-xy}$ and $z = x + xy$ for a finer range of x and y.

5 More and better 3D

3D plots can also be made one point at a time, an exact replica of *plot* from 2D to 3D. Plot3 example:

```
>> t = 0:pi/50:10*pi;
>> plot3(sin(t),cos(t),t);
```

5.1 Question 4

Plot $z = sin\theta$ on the unit circle, i.e. $x^2 + y^2 = 1$.
Hint: Recall XII maths, and use the results already discussed.

6 Multiple plots

Subplot function can be used to partition the plot space into a grid, eg. $subplot(n, m, idx)$ divides the screen into $n \times m$ sections and selects idx^{th} plot to work on, in the column major form. One can change the $m \times n$ to select overlapping parts of the plot screen.

```
>> t = 0:pi/50:10*pi;
>> subplot(2,1,1)
>> plot3(sin(t),cos(t),t);
>> subplot(2,1,2)
>> plot(1:10,(1:10) .^ 2)
>> grid on
>> subplot(2,1,1)
>> grid minor
```

Results in figure 2.

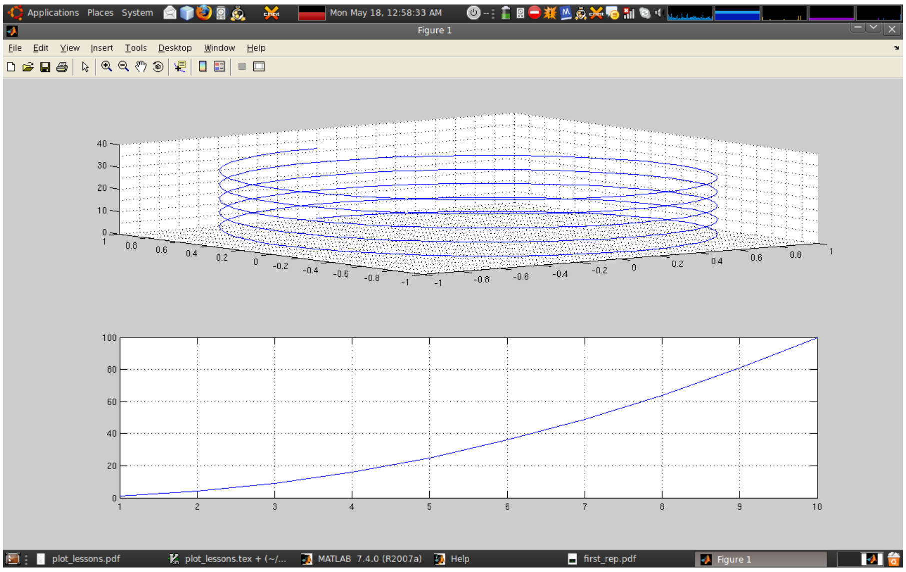

Figure 2: Subplots

6.1 Question 5

Plot the following in a single graph $\theta \in [0, 2\pi]$:

1. $sin\ \theta$ in the place 1, when we have a 2×2 board.
2. $cos\ \theta$ in the place 2, when we have a 2×2 board.
3. $tan\ \theta$ in the place 2, when we have a 2×1 board.

7 Vector fields

```
>> [x,y] = meshgrid(-2:.2:2,-1:.15:1);
>> z = x .* exp(-x.^2 - y.^2); [px,py] = gradient(z,.2,.15);
>> contour(x,y,z), hold on
>> quiver(x,y,px,py), hold off, axis image
```

7.1 Question 6

Consider the function $z = max(-x - y, x + 5y + 5, x^2 + 2y^2 + 1)$.

Draw its contour as well as gradient plot near the points $(0, 0)$ and see its behaviour. Can you guess the point of minima?

7.2 Question 7

For testing speed of convergence of algorithms, a particular function is notorious (banana function).

$z = f(x, y) = 100 * (y - x^2)^2 + (1 - x)^2$

Plot the function contour as well as its gradient in the region $x \in [-4, 4]$ and $y \in [-2, 10]$ to find out why. Keep the contour plot dense while the gradient graph should be sparse enough to determine the arrows clearly. Also, using the function values, or otherwise, find its minimum point.