TECHNISCHE UNIVERSITÄT DRESDEN

Introduction to Matlab

Logical indexing and plots

Dario Cuevas and Vahid Rahmati

DRESDEN concept
Exzellenz aus Wissenschaft und Kultur

Logical indexing

- Important function: find
- To find the elements of an array that satisfy a condition, we use logical indexing. For example, for a vector VecX = 2:2:20;.
- idx = find(VecX<7) gives the indices of VecX whose values are smaller than 7.
- VecX(idx1) gives you the values of VecX which are smaller than 7.
- For MatX = magic(5); [idxRow, idxCol] = find(MatX<10); gives the two indices of each element of MatX smaller than 10.
- MatX([idxRow,idxCol]) does **NOT** return the values of MatX which are smaller than 10. To do this, we use logical indexing, for example:
  - MatX(MatX<10)
  - MatX(MatX>5)
  - MatX(MatX~=5)

Logical indexing

- You can replace certain values of a matrix:
  - `MatX(MatX==5) = -1;`
  - `MatX(MatX<=3) = MatX(MatX<=3)+1;`
  - `MatX(MatX>10 & MatX<20) = 15; %More than one condition`

  Hint : note that we used & instead of &&
- You can find all the elements of a row (or column) that satisfy a condition:
  - `MatX(1, MatX(1,:)<20)`
  - `MatX(MatX(:,3)>5,3) = 200;`
- Can you see what this code does?

```
MatX = magic(5);
MatX(MatX>10) = -(1:(numel(MatX(MatX>10))) );
```

Exercises: Create a matrix `MatY = ceil(10*rand(10))`

1. Change the elements of MatY that are smaller than 3 into -1
2. Change the elements of MatY between 4 and 7 into -2
3. Change the positive elements of the first column of MatY into 0.

Plot command

- plot(x,y), where x and y are vectors of the same size. For example
  - x = 1:0.1:10; y = sin(x); plot(x, y)
  - x1 = -pi:0.1:pi; plot(x1, 2*cos(x1))
  - x2 = 1:10; plot(x2, x2.^2, 'red') %or blue, black, b, r, g, p, m, etc…
  - x3 = 0:0.1:pi/2; plot(x3, arctan(x3), 'b*') % color + marker
  - x4 = -10:10; plot(x4, heaviside(x4), '-.');
- List of markers and colors: help plot
- You can put more than one function in a plot:
  - plot(x, sin(x), 'g', x, cos(x), 'red')

Plot command

- Example: x = 0:0.1:2*pi; plot(x, 2*sin(0.5*x));
- Useful properties

Property	What it does…
title	Sets title for the plot figure
xlabel/ylabel	Gives a label to each axis
legend	Creates a floating legend
axis([x1, x2, y1, y2])	Changes the range of the plot
axis equal/square/tight	Changes the aspect ratio of the plot
grid on/off	Turn the grid on or off
LineWidth	Changes the width of the plot line

- get(gca) gives a list of all the things you can change

Plot-Hold commands

- To draw multiple plots in one plot:
  1) A combination of plot and hold commands can be used in order to add the new plot to the current one:

```
plot(x1,y1); % first plot
hold on ;  % the first plot will be retained
plot(x2,y2);  % the new plot is added on top of the first plot
plot(x3,y3); % the third plot is added on top of both previous plots
hold off; % the retaining of the previous plots is stopped
plot(x4,y5); % the previous plots were erased, an only this plot is displayed
```

- Hint 1: the vectors x1, x2, x3, and x4 can have different number of elements, and ranges.
- Hint 2: the final range of x and y axes account for their presented smallest and biggest values among the plots.

Plot simoltaneously

- To draw multiple plots in one figure:
  2) The plots can be drawn simultaneously :

```
plot(x1,y1,x2,y2); % draws the plot(x2,y2) on top of the plot(x1,y1)
```

- Hint 3: the hints 1 & 2 also apply here.
- Example1:
  - x1 = -pi:0.1:pi; y1 = cos(x1); x2 = -pi/2:0.05:pi/2; y2=sin(x2);
    plot(x1,y1,'-.ored',x2,y2, '-*green')
    plot(x2,y2,'-.ored',x1,y1, '-*green')
    plot(x1, cos(x1), ' -.ored',x2,y2, ' -*green')
  - x1 = -pi:0.1:pi; y1 = cos(x1); x2 = -pi/2:0.05:pi/2; y2=sin(x2);
    clf;
    plot(x1,y1, '-.ored',x2,y2, '-*green');
    hold on
    plot(x1/3,x1/3,'sblack‘)

Subplot command

- To draw multiple plots in one figure we use subplot(m, n, plot number) command, dividing the figure into m rows and n columns, thus m*n plots will be available.
- Hint 1: the plots are numbered row-wise.
- Example:
  - x1 = -pi:0.1:pi; y1 = cos(x1);
    subplot(3,2,1);
    plot(x1,y1,'-.ored',x2,y2, '-*green');
    title('cos function');
    x2 = -pi/2:0.05:pi/2; y2=sin(x2);
    subplot(3,2,2);
    plot(x2,y2,'-.ored',x1,y1, '-*green')
    ylabel('something! ')
    subplot(3,2,5);
    imagesc(magic(5));
    title('MAGIC MATRIX');

Exercises

1. The vectors x=[0:0.2:10] and z=[-5:0.1:5] are given. Plot the following functions on top of each other in one plot: y1 = exp(-x), y2= (z^2)/10, y3=sin(x). The second function should be drawn in red, and third in green. Set the labels for x and y axes as “x label [unit]” and “y label [unit]”, respectively. Activate the grid lines. Set the title as “my first plot”.
2. Using a for loop, create the vectors of random numbers:
   x = Mean+ StandardDeviation*randn(1,100)
   for mean={-4,-2, 0, 2, 4} and StandardDeviation = 0.5, and draw plot(1:100, x) for each vector x. The resulting curves should be displayed in one plot. Hint: you can use hold commands. Using the command figure create a new figure, and plot there a white noise (zero mean) for samples 1:500 (instead of 1:100) and StandardDeviation of 1. Increase the line width to 2.