MSPR 1
Overview and Introduction to Matlab
MED7 & SMC 7, Copenhagen, 2015

Dr. Hendrik Purwins

Overview

- Basic operations
- Vectors
- Plots
- Reading/writing data from/to files

References:

- https://www.mathworks.com/help/pdf_doc/matlab/getstart.pdf
- Demo videos in Matlab

Matlab

- `help`

Variables

```
a = 10     % assign the number 10 to variable a
b = a^2    % take the square of a
```

Even though a and b are assigned I can make them anything I want.

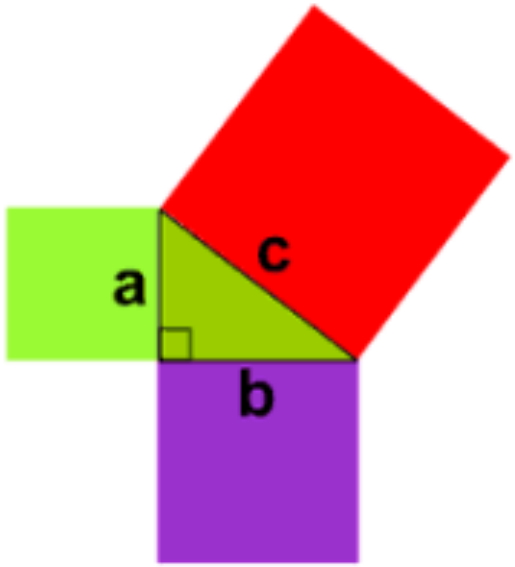
```
a = 'hello'
```

Class Assignment

We have a triangle including a 90 degree angle and the two shortest sides are lengths a=3 and b=4. What is the length of the hypotenuse c? Pythagoras says:

$$c^2 = a^2 + b^2 \Rightarrow c = \sqrt{a^2 + b^2}$$

Calculate by hand and implement in Matlab.



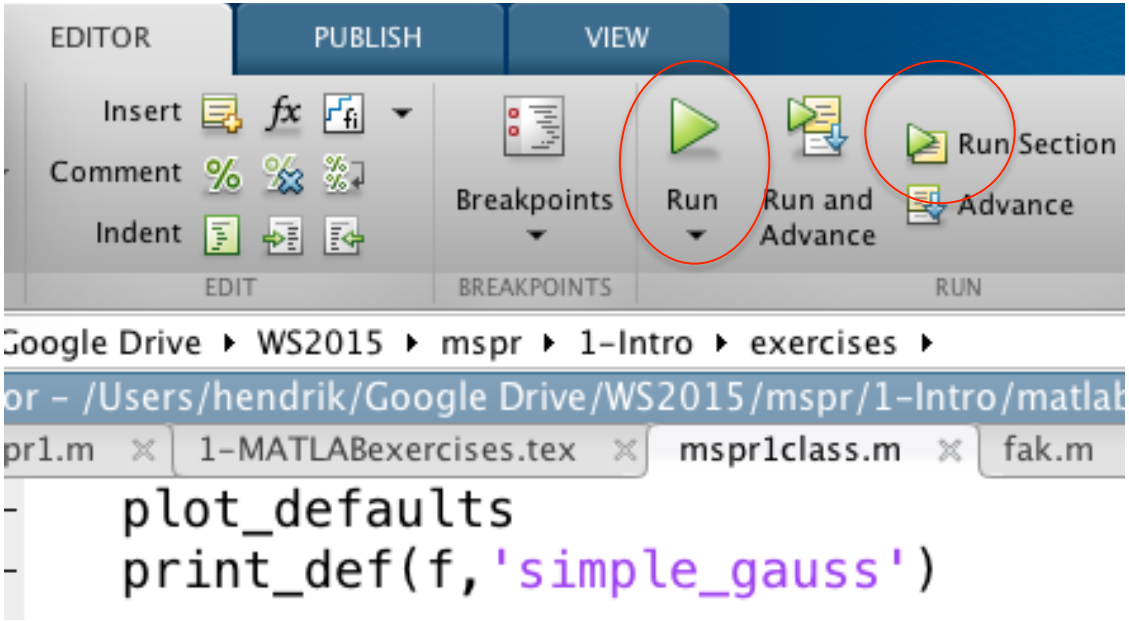
Vectors

```
b = [1; 2; 3]  % make a 3x1 column vector
b = [1, 2, 3]  % make a 1x3 row vector
b = [1 2 3]    % the commas can be left off too
b = b'   % take transpose to make 3x1 vec.
c = b*2     % multiply each element by 2

b.*c  % element-wise multiply
b'*c  % inner (dot) product
b*c'  % outer product
```

Mind the differences!

How to Run Code



run
%% section

Type the skript name into the command line
or
Mark code, right mouse click, ‘Evaluate Selection’

Matrices

Column vectors are nx1 matrices, having n rows and 1 column. A nxm matrix has n rows and m columns.

```
B = [1 2 3 4; 5 6 7 8; 9 10 11 12]
B*5
B+5
```

Vectors

There are quick ways to create vectors and matrices.

```
t = [0:9] % create a 10-element vector going from 0 - 9
t = [0:0.1:9.9] % 1x100 vector from 0 - 9.9 by steps of
   0.1
t = [1:1:10;1:2:20] % a 2x10 matrix, 2nd row incremented
   by 2
```

We can see the value of a specific element of t by using index numbers

```
t(1,4) % Value of t in first row, fourth column
t(4,1) % Oops. t only has 2 rows, not 4.
t(0,1) % Oops. Indicies in MATLAB start at 1 not 0.
```

To find the values in an entire column use the colon (:).

```
t(:,4) % Return values for all rows in the fourth column
t(2,:) % Return values of second row for all columns
```

Index into matrices by row then column.

Indexing in MATLAB starts at 1.

Plotting data hasn't been so easy!

```
x = -1:0.01:1; % <--- semi colon suppresses
  output
y = x^2;
plot(y)

plot(x, y, 'b')     % Now the axes are
  marked
hold on;
plot(x, x.^3,'r');
legend('x^2','x^3')     % add a legend
xlabel('x');          % add an x-label
ylabel('y');       % add a y-label
title('x^2 and x^3');  % add a title
```

Class Assignment

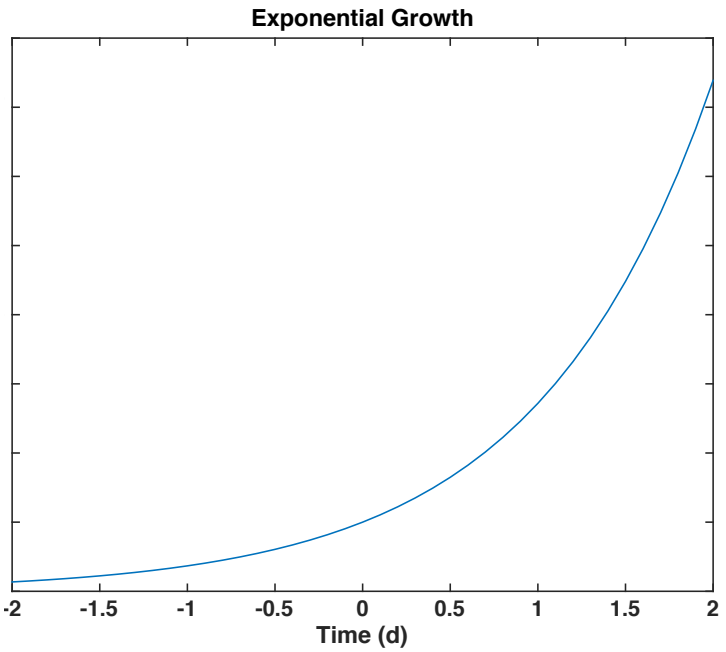
Initial spread of a virus, nuclear chain reaction, savings at constant interest rate, computer processing power all follow exponential growth.

- Mathematically:

$$y(t) = a \cdot e^{\lambda t} \qquad (1)$$

for initial value a time t and growth factor λ

- e is the Euler constant $e = 2.718281828459\ldots$
- Plot the function in the range of -2 until 2 in steps of 0.1

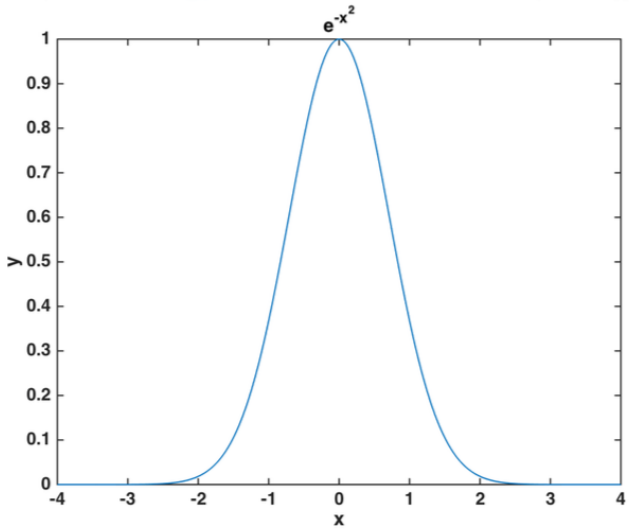


Class Assignment

Plot the function

$$e^{-x^2}$$

for x in the range of -4 and 4 in steps of 00.1.



Reading Text /Binary

```
x =textread('series.txt');    %Read text
save('series.mat', 'x') ;     %  save
  variable  as binary
clear
load series.mat          % reload variable x
```

Readin with textscan

- `fid = fopen('data1');`
- `data = textscan(fid,'%s %f','delimiter',',');`
- `fclose(fid);`

- `%f: float`
- `%s: string`

The Iris Data Set I

- Data set introduced by Sir Ronald Fisher in 1936 as an example of discriminant analysis
- Edgar Anderson collected data to quantify morphological variations of three Iris flower species

Iris setosa

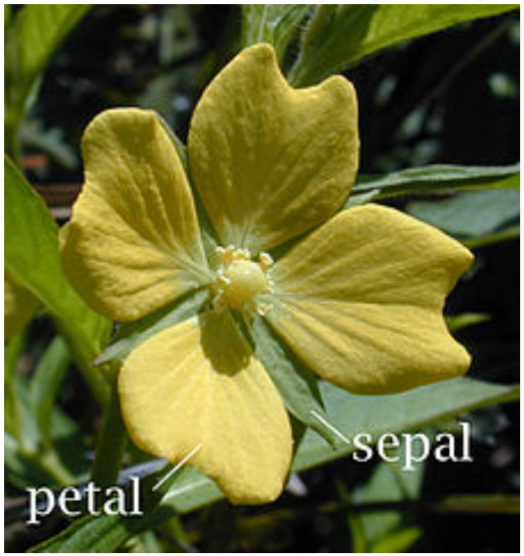
Iris versicolor

Iris virginica

`http://en.wikipedia.org/wiki/Iris_flower_data_set`

The Iris Data Set II

For the 50 samples per each species 4 morphological features are measured



- sepal length
- sepal width
- petal length
- petal width

Data set is a 'classical' data set to demonstrate classification and clustering algorithms.

`http://en.wikipedia.org/wiki/Sepal`

Class Assignment

- Download iris data from the UCL machine learning repository
- Read in iris data with textscan.
- Plot sepal length vs petal length in a scatter plot
- How can you interprete the data?
- Plot histograms of sepal length and of petal width
- Plot a bivariate histogram using hist3 with petal length and petal width.

PRTools

- http://www.37steps.com/software/
- Put into your matlab path!
- `[data{3} data{4}]`: feature vector
- `data{5}:` class labels
- `z=prdataset([data{3} data{4}],data{5},'featlab',...`
- `['Sepal Length';'Sepal Width ';'Petal Length'; 'Petal Width '],...`
- `'name','Fisher Iris');`
- `%`*strings behind 'featlab' need all to have the same the same length!*
- For plotting with class labels:

```
figure(1); scatterd(z);
```

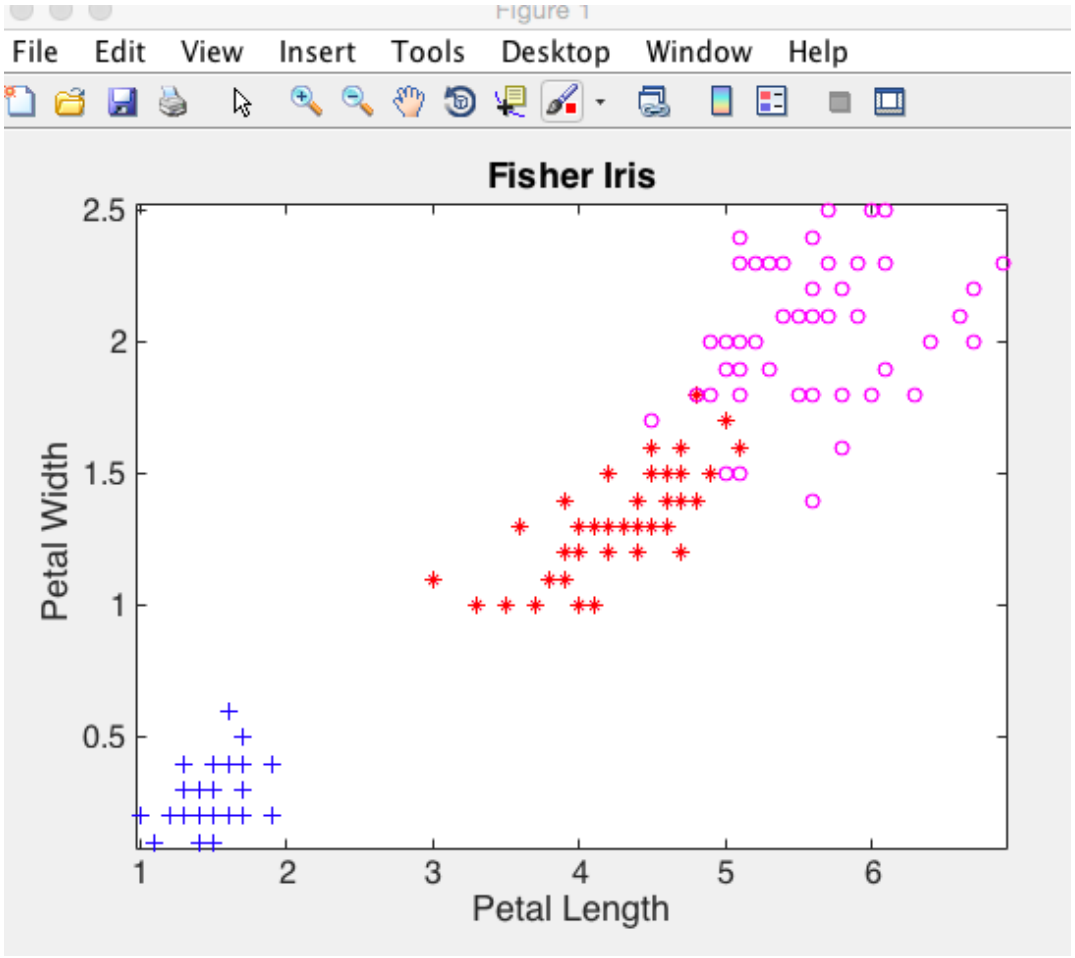


Figure 1
File Edit View Insert Tools Desktop Window Help
Fisher Iris
Petal Width
Petal Length
2.5
2
1.5
1
0.5
1
2
3
4
5
6

Class Assignment

- Do scatter plots of all combination of 2 features in the data. Select the one that separates best the data.
- Interprete the data!

Reading/Playing/Writing Sounds!

```
instr='bass'
[snd sr]=audioread([instr '.wav']);          % load sound
   file
len=length(snd);
plot((1:len)/sr,snd);
sound(snd, sr);                  % play sound
sound(snd/2, sr);
wavwrite(snd/2,sr, 'bass_low');    % save sound with
   lesser volume
```

CLASS ASSIGNMENT

Load the bass.wav into Matlab. Cut out the first bass note from the sound file, plot it

(Time in seconds vs. Amplitude) and play it from Matlab.

Reading Images!

```
imagedata = imread('knit.jpg');
whos   % A 'whos' command reveals size information:
imfinfo('knit.jpg') % Here it says 'BitDepth: 8'.

imagesc(imagedata);
imagesc(fliplr(imagedata)); % flips images right/left

colormap('gray');
```

Creating For-Loops and Functions

```
function y=poly(x,c)
   y=0;
   for i=1:length(c),
    y=c(i)*x.^(i-1);
   end
```

```
x=[-2:0.1:2]
c=[0 0 2 1];
y=poly(x,c);
figure
plot(x,y)
```

Class Assignment

- Create a function
- fak(n)
- That calculates n*(n-1)….*2

Creating Multioutput Functions

```
function [add, sub, mul, div] = math(a, b)
    if (nargin ~=2)
       disp('Usage: [add, sub, mul, div] = math(a,b)'); return;
    end
    % Otherwise continue
    if (nargout >= 1)
       add = a + b;
    elseif (nargout >= 2)
       sub = a - b;
    elseif (nargout >= 3)
       mul = a*b;
    elseif (nargout >= 4)
       div = a/b;
    end
    % If nargout == 0 then just display results!
    if (nargout == 0)
       add = a + b; sub = a - b; mul = a*b; div = a/b;
       fprintf('add = %4.2f\n', add);
       fprintf('sub = %4.2f\n', sub);
       fprintf('mul = %4.2f\n', mul);
       fprintf('div = %4.2f\n', div);
    end
```

How to Look at High-Dimensional Data?

- Linear Algebra
- Principal Component Analysis

References

- Bob Sturm's MSPR lecture 2012