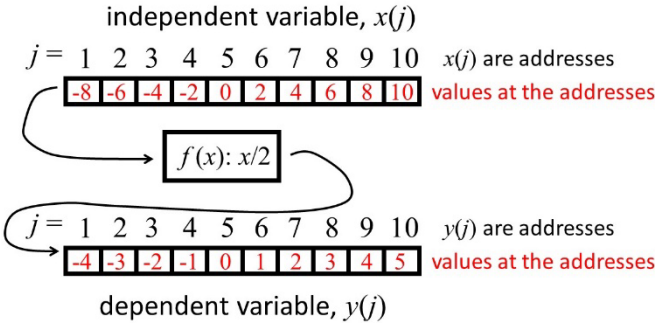
BIOE 198MI Biomedical Data Analysis. Spring Semester 2019.
Lab 1a. Introduction: Variables and Functions

Introduction to MATLAB: Matlab is a numerical computing environment featuring a programming language based on matrix mathematics. It includes a collection of toolboxes containing functions for numerical analysis, statistics, signal and image processing, data acquisition, symbolic math, machine learning, and other modeling and data analysis tools. I will use version R16A, and you are likely to be using R18A or B. There are a few differences but nothing major.

Your objective in this course is to become familiar with writing scripts in Matlab to solve general engineering problems and those in bioengineering. The goal is for you to be able to hear an engineering problem as a story, translate the story into a set of equations, and then interpret the equations into code that generates answers. I plan to lecture as little as possible while providing example problems that we all work out together in class. Coding skill is self-taught and a result of much practice. We expect you to write lab reports that convincingly explain solutions based on clearly visualized results. OK, let's go.

A. Functions, independent and dependent variables

- How do we express and plot `y=f(x)` in Matlab, where the function is to divide x values by 2?




```
x=-8:2:10; %create a row array
y=x/2; %applying f(x) generates y
x,y  %examine the two 1D arrays
plot(x,y) %plot them
xlabel('x');ylabel('y') %label axes
%the independent variable can also be set using
X1=-8;X2=10;dx=2;x=X1:dx:X2; %where the parameters are variables
x=linspace(-8,10,10) %or using X1,X2 and N, total number of points.
```

- Address specific elements of each array by typing `x(2)` or `x(2:4)`.

Exercise: (1) Generate and plot the following (plot the first two in same figure window)

$$x(t) = A\sin(2\pi u_0 t);\quad for\quad A = 2.0,\quad u_0 = 0.2,\quad 0 \le t \le 10\ \ and\ \ \Delta t = 0.01$$

$$y(t) = \frac{A}{2}\sin\left(2\pi u_0 t + \frac{\pi}{2}\right)$$

(2) Create a new figure window using `figure` and replot $y(t)$. Then, on the same graph, plot

$$z(t) = y(t)\exp(-at)\quad for\ \ a = 0.3$$

Finally, use the plot to explain what effect the exponential function has on $y(t)$. Hint: think about how AM radio works.

Summary: The Matlab operations and functions used

- `x=linspace(X1,X2,N)` and `x=(X1:dx:X2)` are similar considering `(X2-X1)/dx=N-1`.
- `x=2*sin(2*pi*0.2*t);` is hardcoding, while `A=2;u=0.2;x=A*sin(2*pi*u*t);` uses variables that adds flexibility when parameters need to change. Also the semicolon `';'` suppresses output to the screen.
- `xlabel('t')` labels the plot. Single quotes indicate text and not a data variable. Careful not all quote marks are the same!
- Executing `plot` twice erases the first plot before plotting the second. Use `hold on` to preserve the first plot within the same window. Or first type `figure` to generate a new window, which will preserve the old window. Use `close 2` to close the second of many plots or `close all` to close all open plot windows.
- Defining the array space for the independent variable will define the array space for the dependent variable. You can create an empty array using `A=zeros(10,10)`, which specifically creates a 10x10 array where all array values are zero.
- Use % to add comments. Use comments especially in longer scripts.
- You need to tell the difference between scalar-scalar, scalar-array, and array-array operations. The `.*` operator can be used to perform an element-by-element multiplication, and the arrays must be the same size. Type `help .` and see what shows.
- For example: `t1=0:10; t2=1:10; x1=sqrt(t1); x2=sqrt(t2); y=x1.*x2;` will give an error. Can you see why? Type `whos` after the error appears to see the array sizes. You can fix using: `t3=[0 t2];x3=sqrt(t3);y=x1.*x3;`

Help: Type `help fft` at the prompt. We will discuss options available there.
Also type `lookfor 'fourier'`. We will discuss the differences and advantages of each.

Exercises:

1. For $x(t) = \sin(2\pi u_0 t)$, apply Matlab to show $\frac{dx}{dt} = \dot{x}(t) = \cos(2\pi u_0 t)$ (numerical derivative for $0 \leq t \leq 10$ where $u_0 = 0.2$). Hint: find out how *for loops* work!

2. Apply Matlab to show that definite integrals $x(t) = \int_0^{2\pi} d\theta \cos\theta = 0$ and $x(t) = \int_0^{\pi/2} d\theta \cos\theta = 1$.

$$\int_0^{2\pi} d\theta \cos\theta = \sin\theta|_0^{2\pi} = \sin(2\pi) - \sin(0) = 0 - 0 = 0$$

Summary: The Matlab operations and functions used

- for loop performs repeated operations while indexing a variable. It has the form

```
for i=n1:dn:n2
   x(i) =
end
```

- defining arrays in for loops using `x=zeros(N,M)` or `x=zeros(N)` or `x=zeros(1,N)` …
- The fundamental theorem of calculus can be demonstrated numerically.
- Difference between `length` and `size`
- `num2str(x)` converts the number x to a text string x for display
- Careful to use radian arguments for sines and cosine