Session 3.

INTRODUCTION TO GNU OCTAVE

United Nations Educational, Scientific and Cultural Organization

Intergovernmental Oceanographic Commission

OceanTeacher GLOBAL ACADEMY

Objectives

1. Provide an overview of the GNU Octave Programming Language.
2. Promote the methodology of using Octave in data analysis and graphics.
3. Demonstrate how to manipulate and visualize ocean currents data in Octave.

Outcomes

After taking this session, you should be able to:

1. Identify the location to download and install GNU Octave,
2. Write Octave's syntax and semantics,
3. Develop a greater conceptual understanding of data analysis and graphics using Octave, and
4. Build skills in manipulating ocean currents data through hands-on exercises.

Session Outline

1. Getting Octave
2. Installing Octave on Ubuntu
3. Running Octave
4. Octave Basics
5. Octave Data Types
6. Importing Data
7. Exporting Data
8. Using Functions in Octave
9. Using Octave Packages
10. Base Graphics

What is GNU Octave

1. GNU Octave (mostly MATLAB® compatible) is a free software tool distributed under the terms of the GNU General Public License.
2. It runs on GNU/Linux, macOS, BSD (Berkeley Software Distribution), and Windows.
3. It is interactive and may also be used as batch-oriented language.
4. It has a large, coherent, and integrated collection of tools for data analysis and graphics.

Steps to Install Octave on Ubuntu

1. sudo apt-get upgrade
2. sudo apt-get update
3. sudo apt-get install octave
4. sudo apt-get install liboctave-dev

Note: The above steps were run in the ‘install_software.sh’ script.

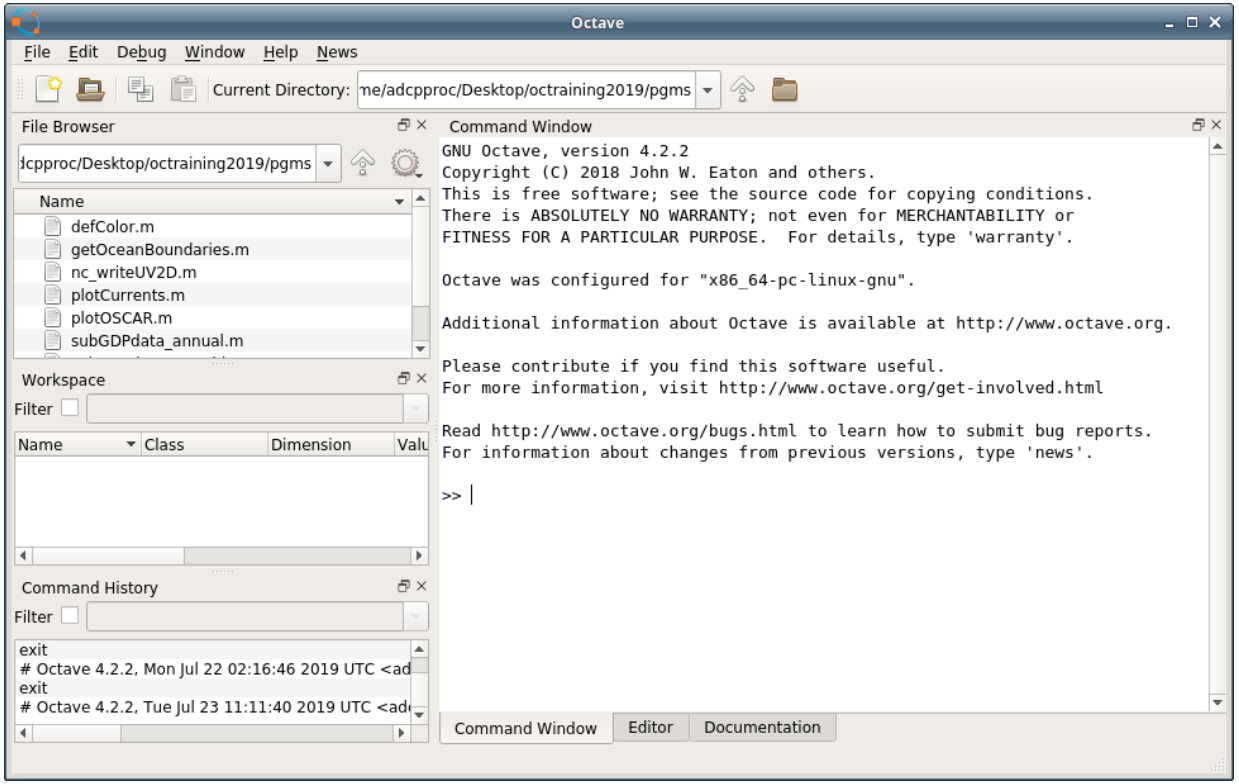
Running/Exiting Octave

1. By default, Octave is started with the shell command '*octave*', or,
2. Type '*octave --no-gui*' at the shell command to start *octave* without GUI.
3. To exit Octave, type "quit", or "exit" at the Octave prompt.

Octave GUI System

The system includes:

1. A command window
2. An editor
3. A debugger
4. A file browser and workspace viewer



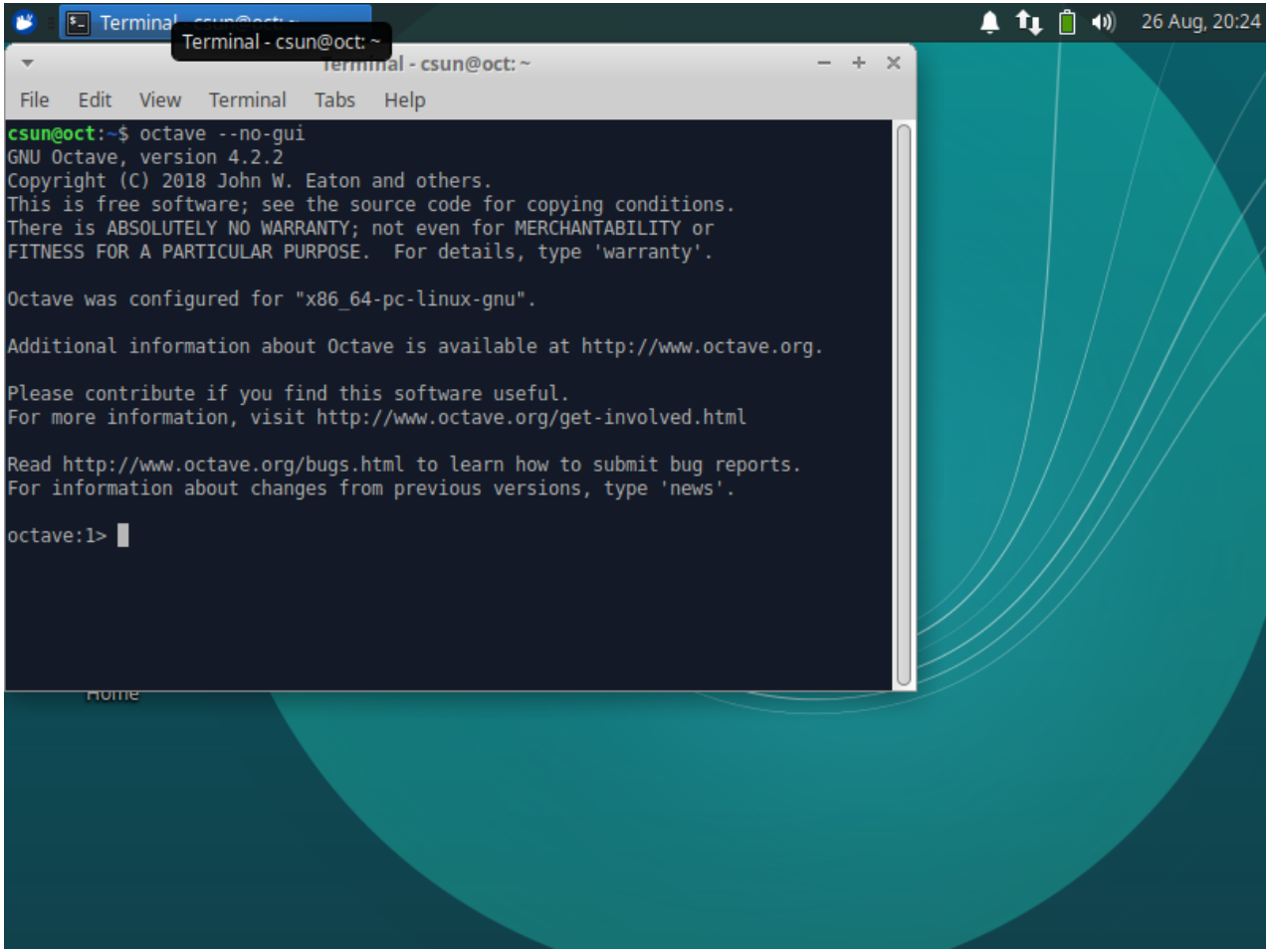
Features of the Octave GUI

- Reading data from files
- Saving and loading variables
- Plotting data
- Customizing plots
- Exporting graphics for use in other applications

Running Octave without GUI

To run Octave without GUI, type
“octave --no-gui” at the system prompt:
$ octave --no-gui

Running Octave without GUI



Octave Packages

- Octave provides field specific features via a set of packages, similar to Matlab's Toolboxes, through Octave-Forge.
- The ‘Octave-Forge’ project is a community-maintained set of packages that can be downloaded and installed in Octave.
- Octave's package system expands its core functionality.

Installing and Removing Packages

- To install a package, use the ***pkg*** command from the Octave prompt by typing:

  pkg install *package_name*, where package_name is the name of the package you want to install.

- To remove a package from the system using the ***pkg*** uninstall command like this

  pkg uninstall *package_name*

Hands-on Exercise: Displaying installed packages

- To query the list of locally installed packages, use the *pkg* command from the Octave prompt by typing: *pkg list*
- Open a terminal on your VM
- Change your working directory to "~/octraining2019/pgms"
- Run octave without GUI
- Type "pkg list" at the octave prompt
- octave > pkg list

Steps Required before Installing Octave Package “NetCDF”

1. sudo apt-get upgrade
2. sudo apt-get update
3. sudo apt install netcdf-bin
4. sudo apt install libnetcdf-dev
5. sudo apt install python-pip
6. pip install netcdf4

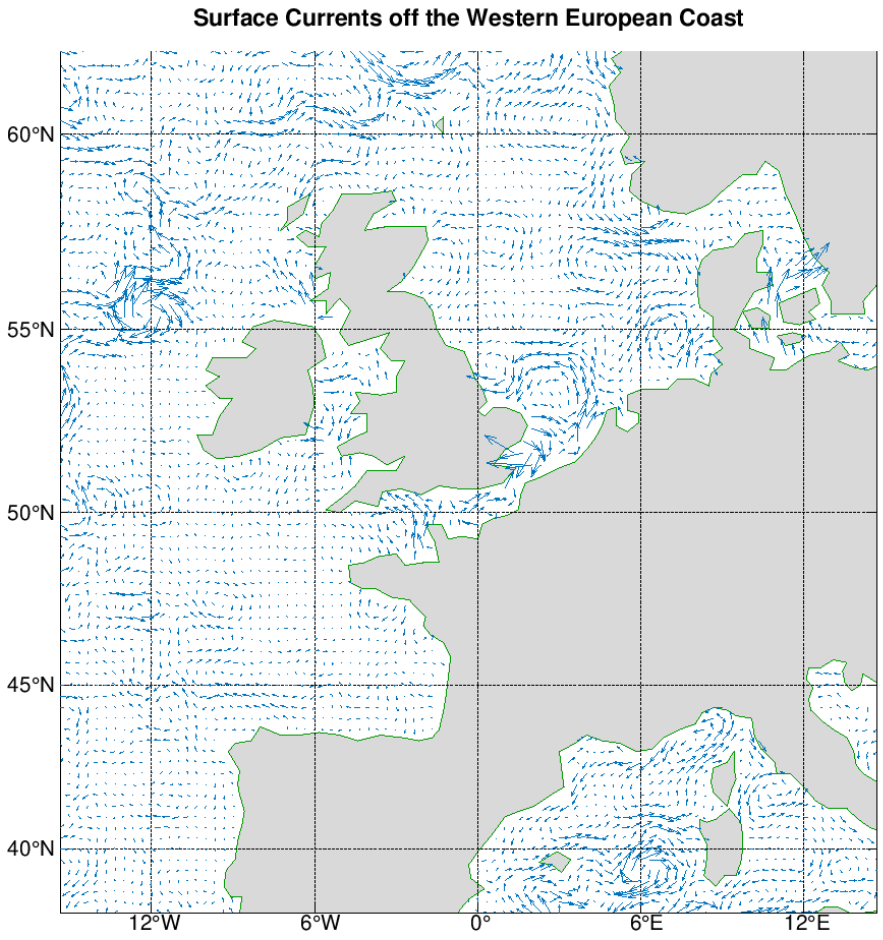
Note: The above steps were run in the ‘install_software.sh’ script.

Hands-On Exercise: Installing and Testing Octave Package

- Open a terminal, type “sudo -s “ at the system prompt, and change directory to “~/octraining2019/packages”
- Run octave without GUI
- Install Octave package, ‘netcdf’ by typing “pkg install ../packages/netcdf-1.0.12.tar.gz” at the Octave command.
- Type “plotOSCAR” at the Octave command window

  octave > plotOSCAR <ENTER>

Output of “plotOSCAR.m”



Hands-On Exercise: Installing Octave Package “io”

- Install Octave Package “io”, using the “io-2.4.12.tar.gz” file located at the “../packages” folder.

  pkg install “../packages/io-2.4.12.tar.gz”

- Check if the “io” package installed correctly

  pkg list

Hands-On Exercise: Installing Octave Package “statistics”

- Install Octave Package “io”, using the “statistics-1.4.1.tar.gz” file located at the “../packages” folder.

  pkg install “../packages/statistics-1.4.11.tar.gz”

- Check if the “io” package installed correctly

  pkg list

- Type “exit” to close “Octave” and exit.

Syntax of Octave

- The syntax of *Octave* resembles that of *MATLAB*.
- An *Octave* program usually runs unmodified on *MATLAB*.
- *MATLAB*, being commercial software, has a larger function set, and so the reverse does not always work, especially when the program makes use of specialized add-on toolboxes for *MATLAB*.

Examples of Octave's Syntax

- Lines marked like so, 'octave:n>', are lines you type, ending each with a carriage return. Octave will respond with an answer, or by displaying a graph.
- octave: 1> a = rand(2,3)
- Ending a command with a semicolon tells Octave not to print the result of the command. For example,
- octave: 2> a = rand(2,3);

Input conventions I

- Octave is case sensitive.
- All commands can be typed in at the prompt or read from a script.
- Scripts are plain text files with file suffix .m. They are imported by calling the file name without the suffix and behave as if their content was typed in line by line.

Input conventions II

- Semicolon, “;”, separates several commands within a line.
- A command terminated by “;” does not display its result on-screen.
- “,” separates two commands without suppressing on-screen output.
- “...” at the end of the line denotes that an expression continues into the next line.

Input conventions III

- Comments are preceded by %.
- Entire blocks of code can be commented by enclosing the code between matching

  ‘#{’ and **‘#}’** or

  ‘%{’ and **‘%}’**

  markers (NOTE: without single quotes).

Variables and Standard Operations

- ***varname*** = ***expression*** assigns the result of *expression* to *varname*.
- Octave has all the usual mathematical functions +, -, *, /, ^, sin, cos, exp, acos, abs, etc.
- The operators of elementary logic are:

< smaller	<= smaller or equal	&	and
> greater	>= greater or equal	|	or
== equal	~= not equal	~	not

Hands-on Exercise: Syntax Examples

- octave:1> x12 = 1/8, long_name = 'A String'

  x12 = 0.12500

  long_name = A String

- octave:2> sqrt(-1)-i

  ans = 0

- octave:3> x = sqrt(2); sin(x)/x

  ans = 0.69846

Octave Scripts

There is a script named as “doAll.m”:

```
x12 = 1/8, long_name = 'A String'
sqrt(-1)-i
x = sqrt(2); sin(x)/x
```

Hands-on Exercise: Executing Octave Scripts

There is a script doAll, saved in a file named doAll.m:

```
x12 = 1/8, long_name = 'A String'
sqrt(-1)-i
x = sqrt(2); sin(x)/x
```

```
Octave 4:> doAll
x12 =  0.12500
long_name = A String
ans = 0
ans =  0.69846
octave:5>
```

Vector and Matrix Operations

Matrices and vectors are the most important building blocks for programming in Octave.

- Vectors
  - Row vector: v = [1,2,3] or v = [1 2 3]
  - Column vector: v = [1; 2; 3]
- Matrices
  - A = [1 2 3; 4 5 6]

Hands-On Exercise: Basic Matrix Arithmetic

- octave> A = [2,1;3,2;-2,2]

```
A =

   2   1
   3   2
  -2   2
```

- Multiplication by a scalar, c

  octave> c = 3;

  octave> c*A

```
ans =

   6   3
   9   6
  -6   6
```

- Matrix Addition, Subtraction &Multiplication

Hands-On Exercise: Basic Matrix Arithmetic (cont.)

- Transpose of a Matrix

  octave: AT = A’

  AT =

  2 3 -2

  1 2 2

Hands-On Exercise: Matrix Concatenation

Horizontal Concatenation

```
octave> A
    A =
      2   1
      3   2
     -2   2
octave> B = [1,1;3,4;2,2]
    B =
      1  1
      3  4
      2  2
octave> C = [A,B]
```

```
C =
   2   1   1   1
   3   2   3   4
  -2   2   2   2
```

Vertical Concatenation

```
octave> C = [A;B]
    C =
      2   1
      3   2
     -2   2
      1   1
      3   4
      2   2
```

Functions

- Traditionally, functions are also stored in plain text files with suffix .m. In contrast to scripts, functions can be called with arguments, and all variables used within the function are local, i.e., they do not influence variables defined previously.
- Example:

  A function f, saved in the file named “myFunction.m”.

```
function y = myFunction (x)
  y = cos(x/2)+x;
end
```

Global Variables

- **global** name declares name as a global variable.
- Example:

  A function foo in the file named foo.m:

  global N % makes N a global variable; may be set in main file

  function out = foo(arg1,arg2)

  global N % makes local N refer to the global N

  <Computation>

  end

Loops

for-loop

```
for n = 1:10
        [x(n),y(n)]=f(n);
end
```

while-loop

```
while t<T
        t = t+h;
end
```

for-loop backward

```
for n = 10:-1:1
        [x(n),y(n)]=f(n);
end
```

Branching: The *if* statement

First

```
if (condition)
  then-body
endif
```

Second

```
if (condition)
  then-body
else
  else-body
endif
```

Third

```
if (condition)
  then-body
elseif (condition)
  elseif-body
else
  else-body
endif
```

Branching: The *switch* statement

The general form of the switch statement is

```
switch (expression)
    case label-1
      command_list
    case label-2
      command_list
    otherwise
      command_list
endswitch
```

Variables and Data Types

In Octave almost everything is a matrix.

Main matrix Classes:

- Strings: matrices of characters
- Structures: matrices of named fields for data of varying types and sizes
- Logical: matrices of boolean 0/1-values

Variables and Data Types

What about...

1. Vectors or arrays?
   - A matrix with one column or row
2. Scalars?
   - A matrix of dimension 1x1
3. Integers?
   - A double (you never have to worry)
4. Characters:
   - A string of size 1

Variables and Data Types

- Creating a Matrix

  Simply type octave:1> A = [1,2;3,4]

- Creating a Character String

  Simply type octave:2> str = 'Hello World'

- Creating a Structure

  Type for instance: octave:3> data.name = 'sensor 1 front'

Displaying Variables

- Display Variables

  Simply type its name

  – octave:> a

- Suppress Output

  Add a semicolon

  – octave:> a;

  – Applies also to function calls

Base Graphics

- Octave provides plotting through the use of the function call, *graphics_toolkit* (***name***).
- When called with a single input *name* set the default graphics toolkit to ***name***.
- With no inputs, return the current default graphics toolkit.
- Use ‘available_graphics_toolkits()’ to display available graphics toolkits.

Hands-On Exercise: Display Available Graphics Toolkits

- Start Octave without GUI

  $ octave --no-gui

- Type “available_graphics_toolkits()” at the Octave prompt.

  Octave > available_graphics_toolkits()

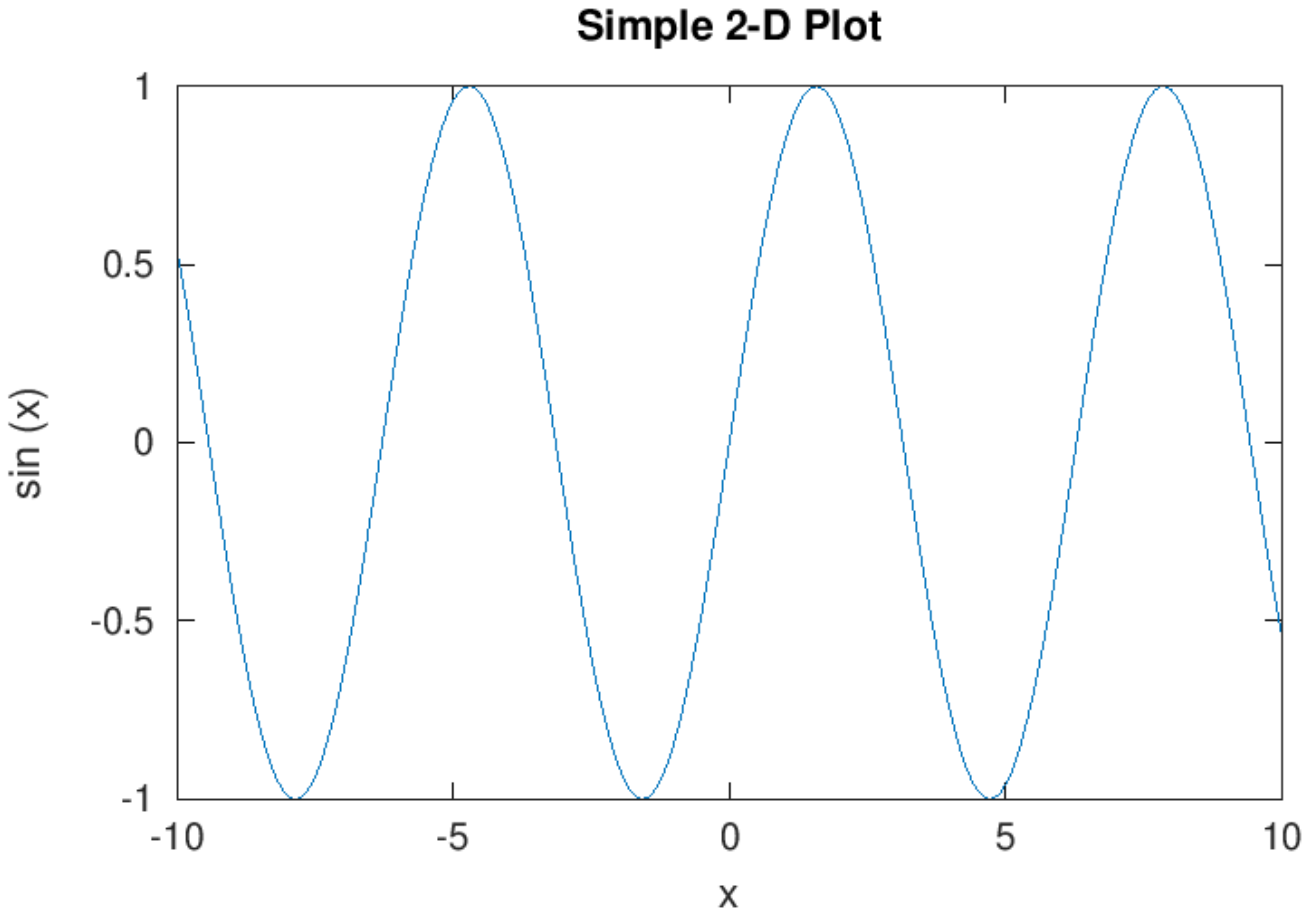
```
ans =
{
  [1,1] = fltk
  [1,2] = gnuplot
  [1,3] = qt
}
```

High-Level Plotting

- Octave provides simple means to create many different types of two- and three-dimensional plots using high-level functions.
- Two-Dimensional Plots
  - The plot function allows you to create simple x-y plots with linear axes. For example,

```
x = -10:0.1:10;
plot (x, sin (x));
xlabel ("x");
ylabel ("sin (x)");
title ("Simple 2-D Plot");
```

Simple Two-Dimensional Plot

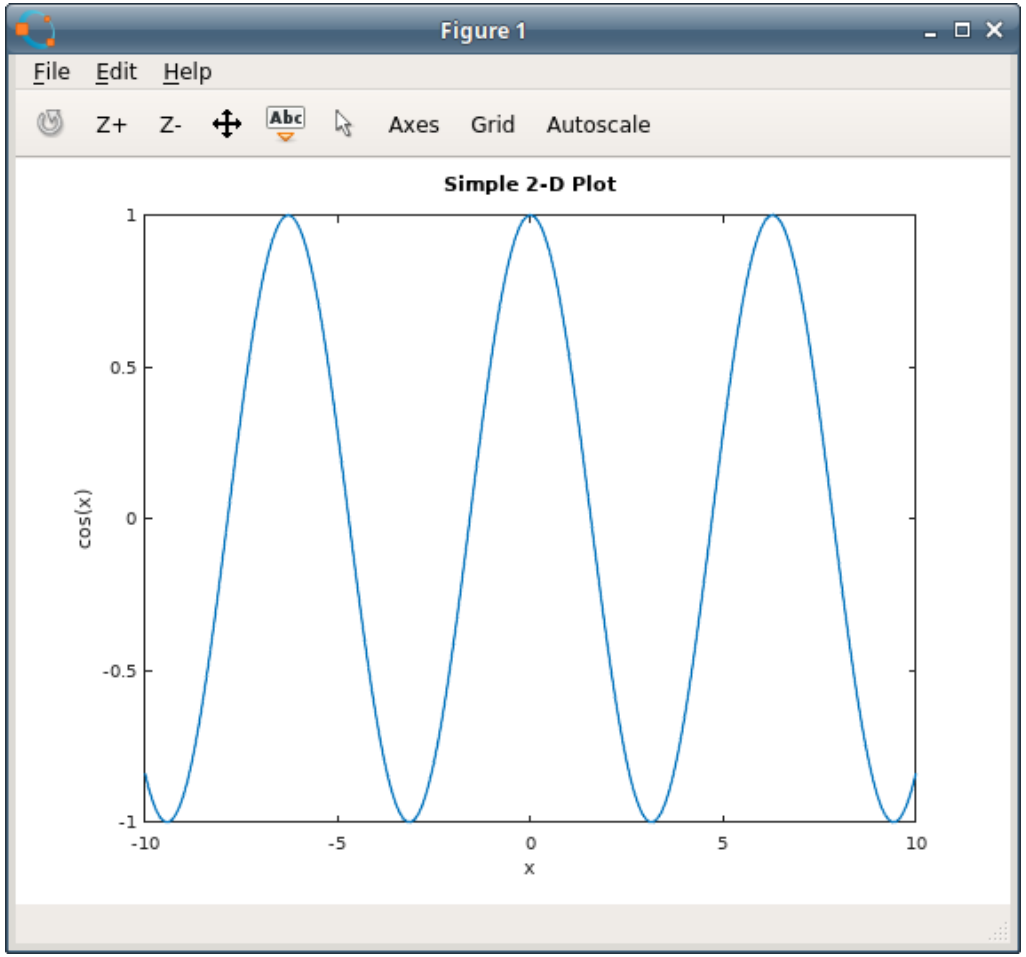


Hands-On Exercise: Creating a simple cosine wave plot

- Adapting the following script to create a simple cosine wave plot.

```
x = -10:0.1:10;
plot (x, sin (x));
xlabel ("x");
ylabel ("sin (x)");
title ("Simple 2-D Plot");
```

Simple 2-D Plot of a Cosine Wave



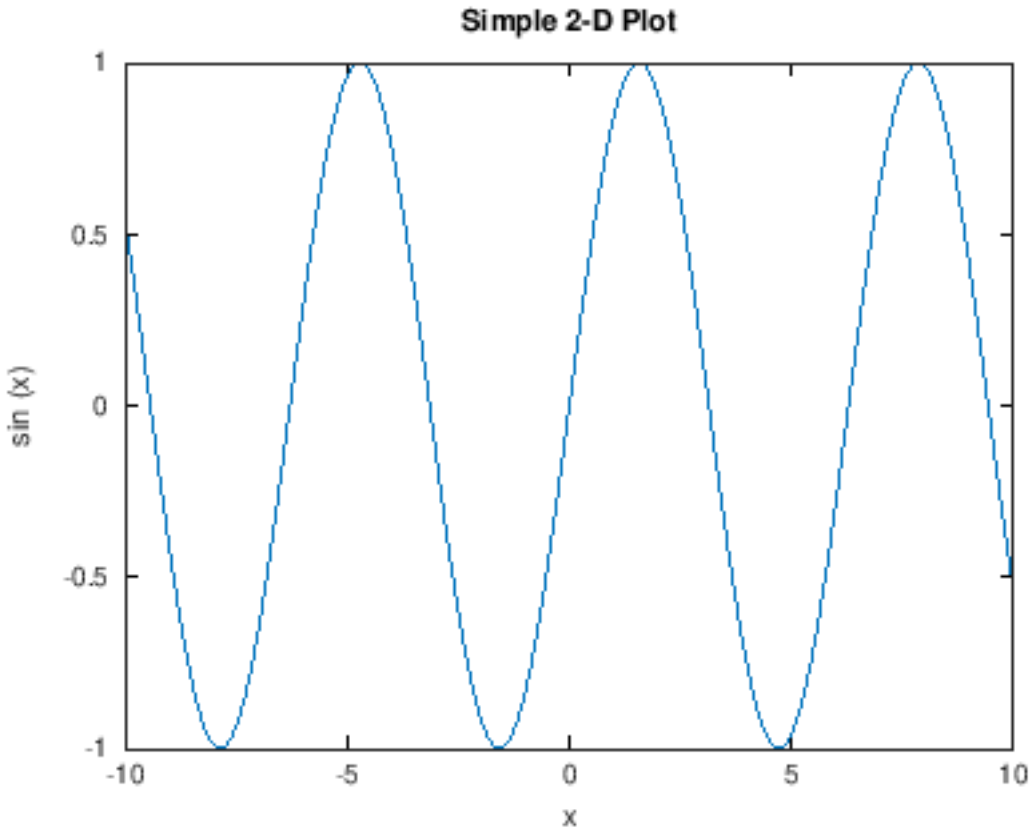
Printing and Saving Plots

- The *print* command allows you to save plots in a variety of formats. A complete example showing the capabilities of text printing using the -dpng option is:

Script “printSineWave.m”

```
close all;
clear;
graphics_toolkit('gnuplot');
xstart = 100; ystart = 100;
xsize  = 400; ysize = 320;
width  = num2str(xsize); %Convert numerical data to strings
height = num2str(ysize); %Convert numerical data to strings
resolution = strcat('-S',width,',',height);
hf = figure ('visible','on','Position',[xstart,ystart,xsize,ysize]);
x = -10:0.1:10;
plot (x, sin (x));
xlabel ("x");
ylabel ("sin (x)");
title ("Simple 2-D Plot");
print('sineWave.png','-dpng',resolution);
```

Output of “printSineWave.m”



Hands-On Exercise: Creating plots without displaying

- Modify the Octave script, ‘printSineWave.m’, located in the ‘pgms’ directory, to set ‘visible’ ‘off’, to create a plot without displaying.

Hands-On Exercise: Q & A

How do I run an octave script?

- To execute a script from within Octave, just type its name without the .m extension. Thus, if you have a script called foo.m , just type foo from within the Octave command prompt to execute it. You must make sure that the script is in your current working directory or in Octave's load path.

Hands-On Exercise: Q & A

What is octave Forge?

- Octave Forge is a central location for collaborative development of packages for GNU Octave. The Octave Forge packages expand Octave's core functionality by providing field specific features via Octave's package system.

It’s time for a break!!!

BACKUP SLIDES

Reading Assignments

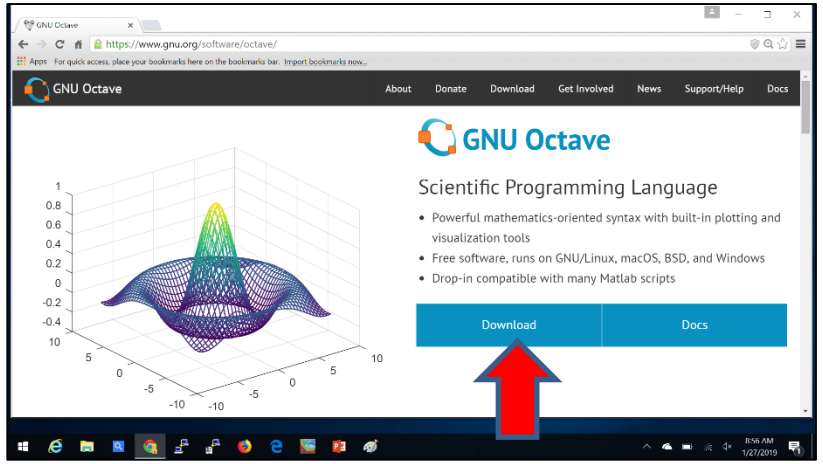
- https://octave.org/doc/v4.2.1/index.html
- http://math.jacobs-university.de/oliver/teaching/iub/resources/octave/octave-intro/octave-intro.html
- http://www.philender.com/courses/multivariate/notes/matoctave.html
- http://www.yanivplan.com/files/tutorial3matrixop.pdf

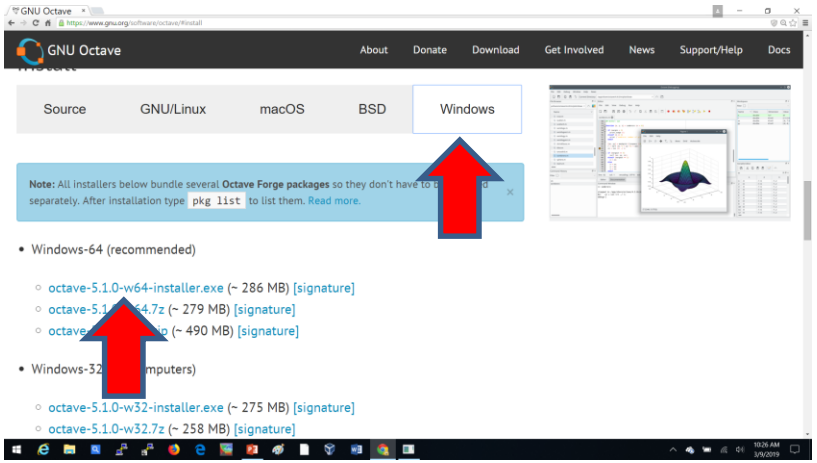
Downloading Octave

- The latest released version of Octave is always available from https://www.gnu.org/software/octave/download.html, or
- The previous released version of Octave is always available from ftp://ftp.gnu.org/gnu/octave.

Getting Octave

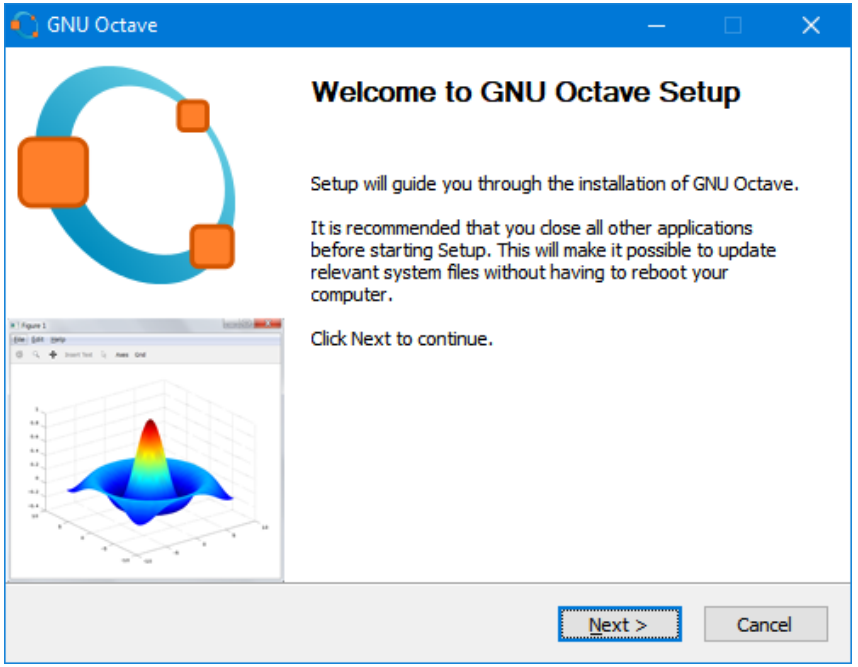
1. Download Octave for Windows from https://www.gnu.org/software/octave/
2. Click on “Download”
3. Select “Windows”
4. Right-Click on “octave-5.1.0-w64-installer.exe ”
5. Select “Save Link As …” to save the installer to your local hard drive.





Installing Octave for Windows

1. Double-click on the installer with the filename of “octave-5.1.0-w64-installer.ext”.
2. Follow the instructions on the screen and Leave all default settings.
3. The installer creates two shortcuts on the desktop for you by default.



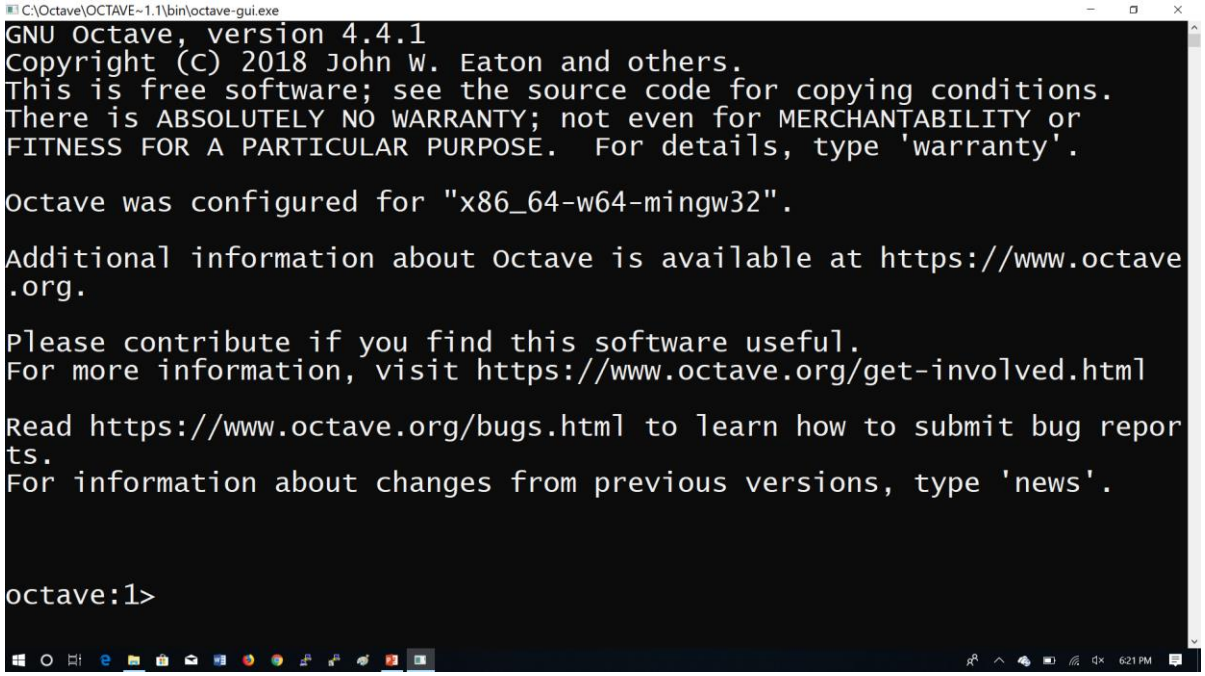
CLI=Command Line Interface
GUI=Graphical User Interface

Versions of the Octave Executable

1. Octave Command Line Interface (Octave-CLI) runs from a command line such as a 'Command Prompt' window running the standard shell such as cmd.exe.
2. Octave Graphical User Interface (Octave-GUI) runs as a standard Windows GUI executable and provides an Octave console in its own window.

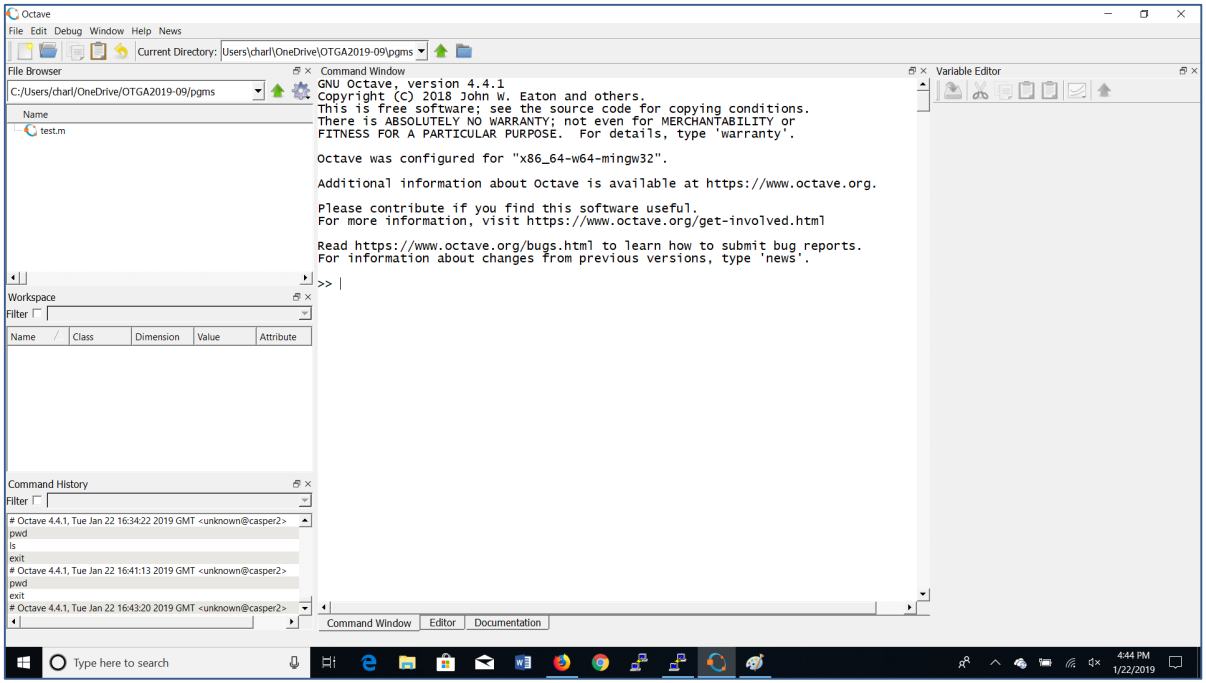
Running Octave-CLI

Double-click on the “Octave-CLI” shortcut to open a 'Command Prompt' window.



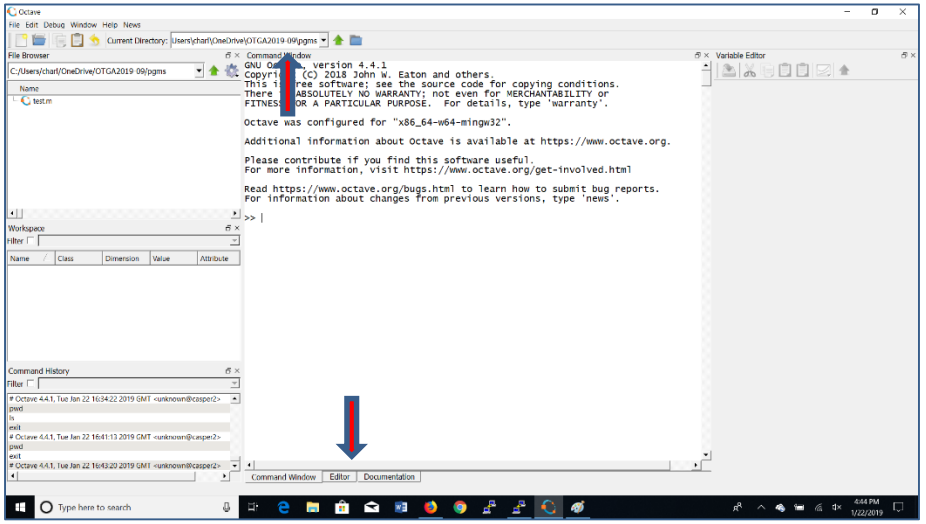
Running Octave-GUI

Double-click on the “Octave-GUI” shortcut you prepared at installation

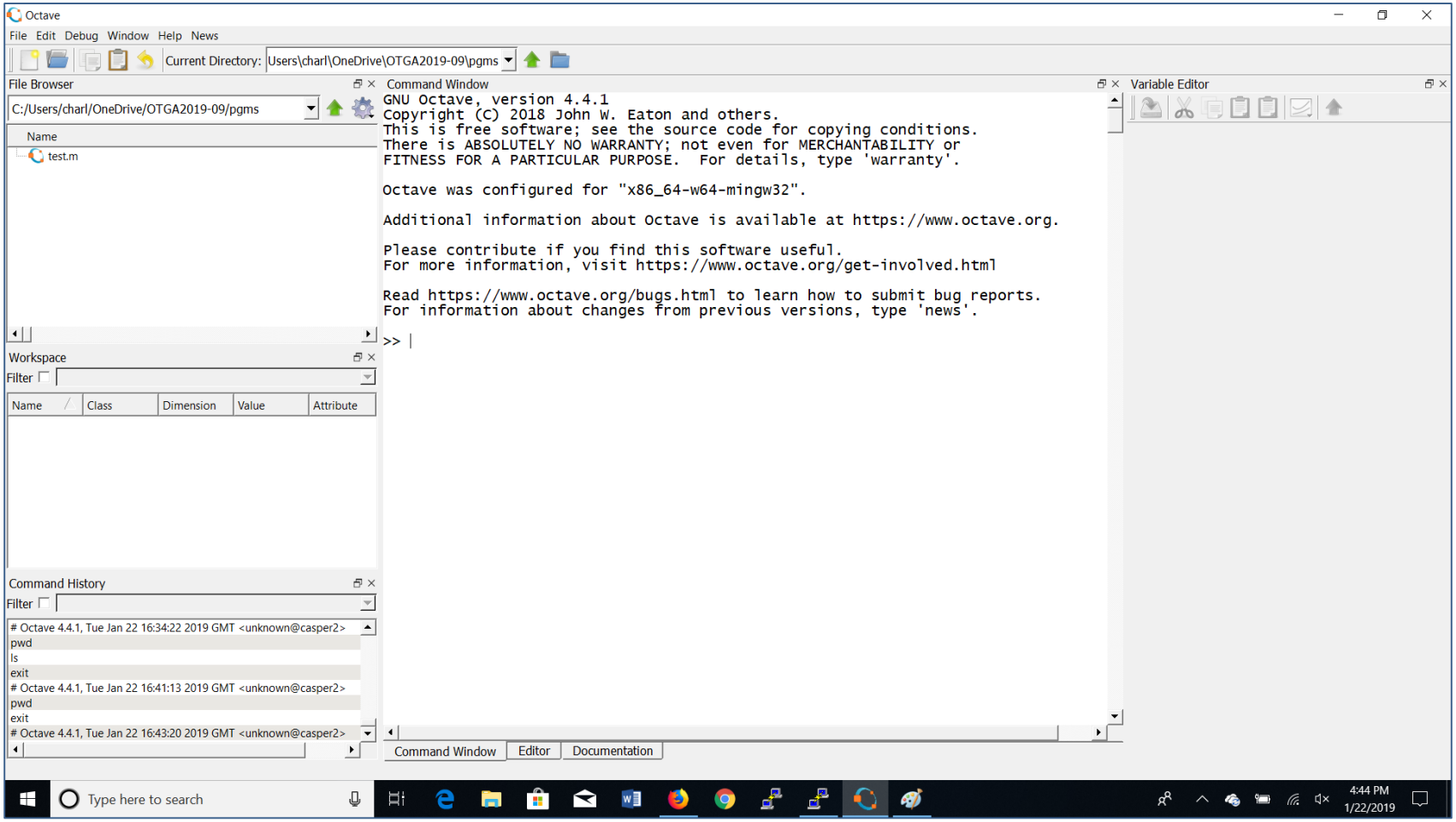


Octave-GUI Main Windows

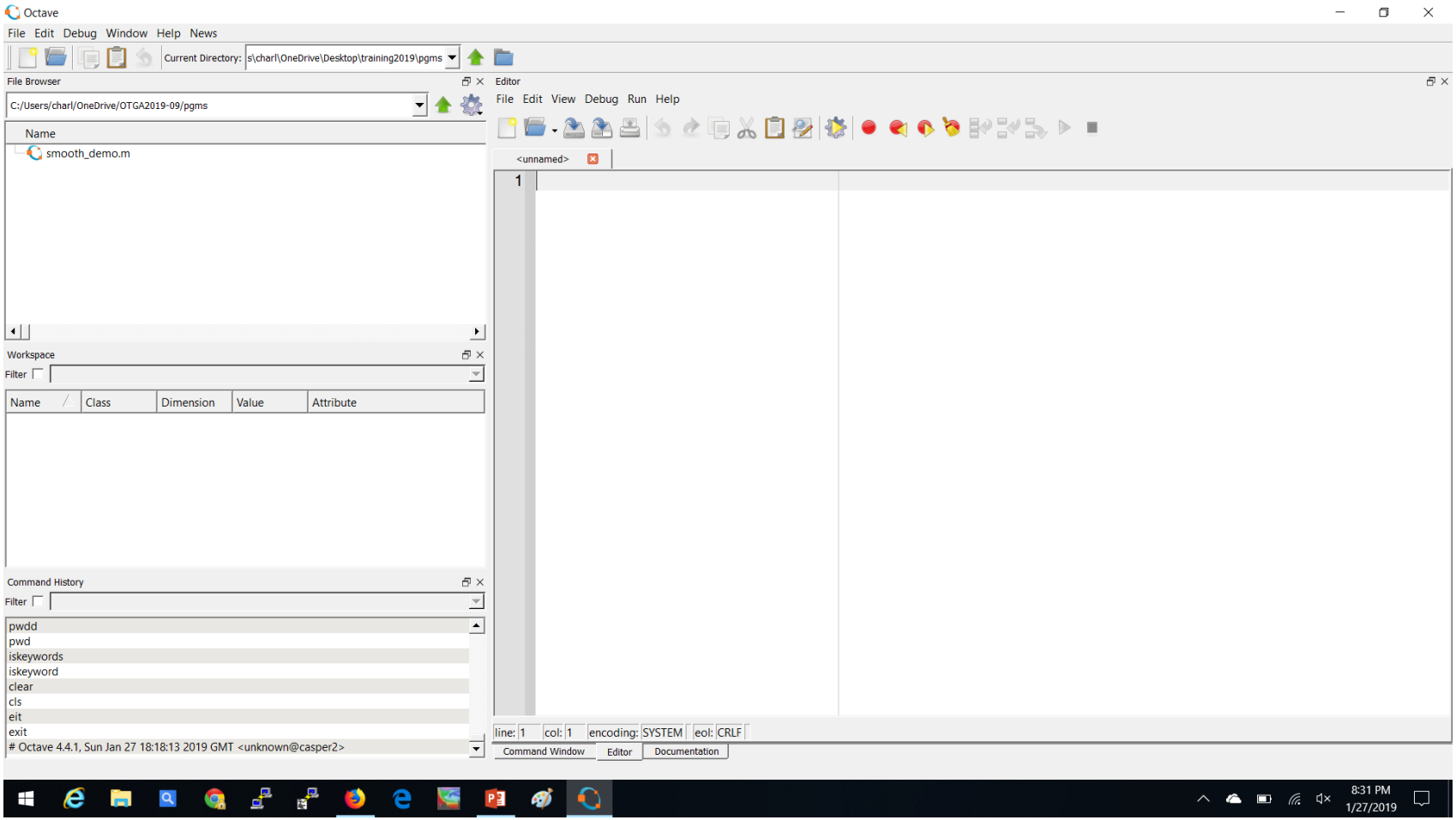
Three windows, “Command Window”, “Editor”, and “Document”, are available by default. They can be opened either via Window menu or from tabs at the bottom of the Octave main window.



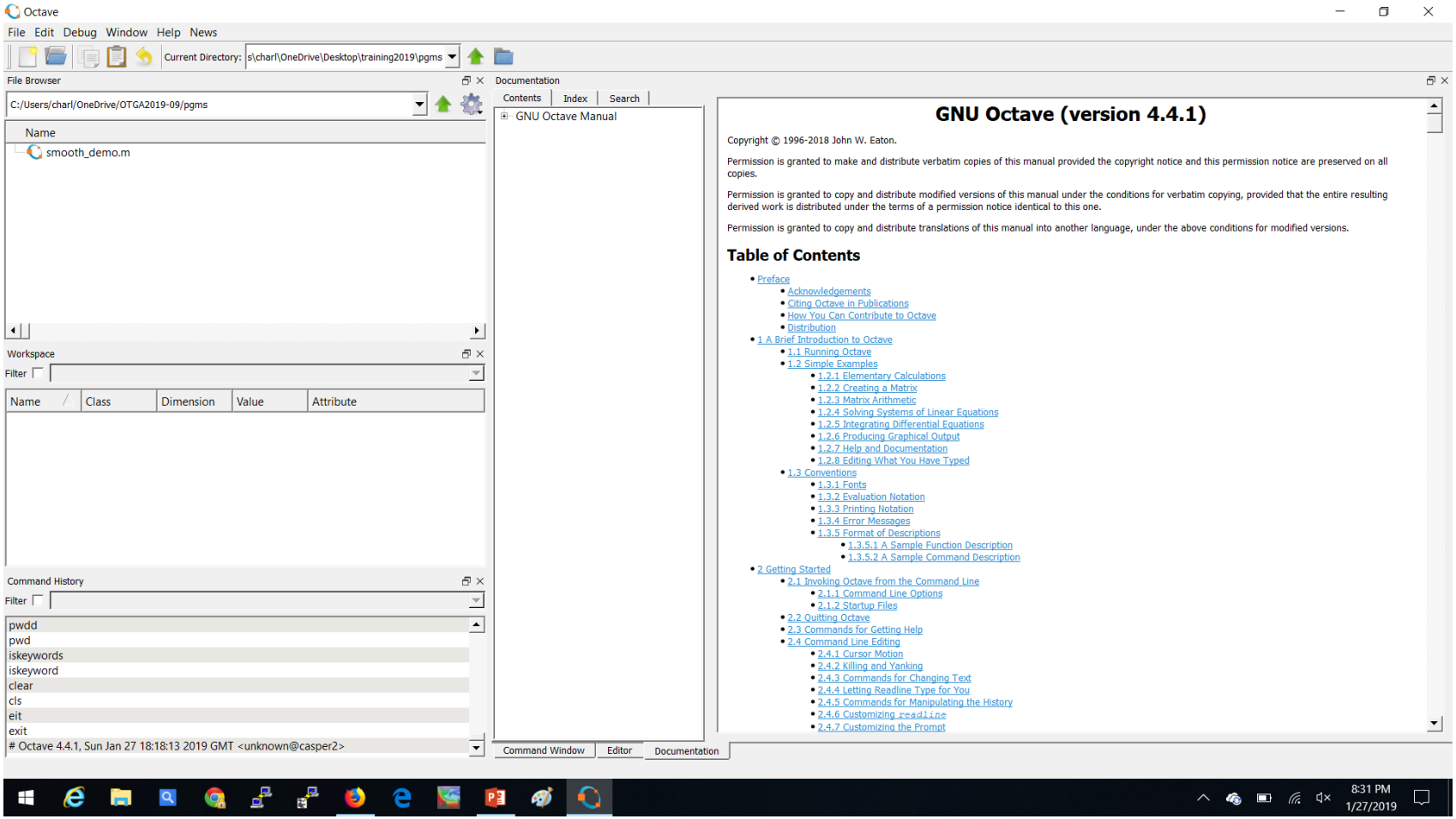
Octave-GUI Command Window



Octave Editor Window



Octave Document Window



References

- Introduction to GNU Octave
  - http://math.jacobs-university.de/oliver/teaching/iub/resources/octave/octave-intro/octave-intro.html
- https://www.tutorialspoint.com/matlab/index.htm
- Introduction to GNU Octave - Wytheville Community College - VCCS
  - http://www.wcc.vccs.edu/sites/default/files/Introduction-to-GNU-Octave.pdf