Getting Started with MATLAB, Python and R

AAEC 6305: Dynamic Economic Optimization - Fall 2019

Matthew Aaron Looney

8/6/2019

Contents

1 Introduction

2 Help

2.1 Getting Help

Language	MATLAB/Octave	Python	R
Browse help interactively	`doc` Octave: `help -i % browse with Info`	`help()`	`help.start()`
Help on using help	`help help` or `doc doc`	`help`	`help()`
Help for a function	`help plot`	`help(plot)` or `?plot`	`help(plot)` or `?plot`
Help for a toolbox/library package	`help splines` or `doc splines`	`help(pylab)`	`help(package='splines')`
Demonstration examples	`demo`		`demo()`
Example using a function			`example(plot)`

2.2 Searching available help documentation

Language	MATLAB/Octave	Python	R
Search help files	`lookfor plot`		`help.search('plot')`
Find objects by partial name			`apropos('plot')`
List available packages	`help`	`help(); modules [Numeric]`	`library()`
Locate functions	`which plot`	`help(plot)`	`find(plot)`
List available methods for a function			`methods(plot)`

2.3 Using interactively

Language	MATLAB/Octave	Python	R
Start session	Octave: `octave -q`	`ipython -pylab` or JupyterLab	RStudio
Auto completion	Octave: `TAB` or `M-?`	`TAB`	
Run code from file	`foo(.m)`	`execfile('foo.py')` or `run foo.py`	`source('foo.R')`
Command history	Octave: `history`	`hist -n`	`history()`
Save command history	`diary on [..] diary off`		`savehistory(file=".Rhistory")`

End session	`exit` or `quit`	`CTRL-D`
		`CTRL-Z # windows`
		`sys.exit()`
			`q(save='no')`

3 Basic programming

3.1 Loading packages

Language	MATLAB/Octave	Python	R
Script file extension	.m	.py	.R
Comment symbol (rest of line)	% Octave: % or #	#	#
Import library functions	% must be in MATLABPATH Octave: % must be in LOADPATH	from pylab import *	library(RSvgDevice)
Eval	string=’a=234’; eval(string)	string="a=234" eval(string)	string <- "a <- 234" eval(parse(text=string))

3.2 Working directory and OS

Language	MATLAB/Octave	Python	R
List files in directory	dir or ls	os.listdir(".")	list.files() or dir()
List script files in directory	what	grep.grep("*.py")	list.files(pattern="\.r$")
Displays the current working directory	pwd	os.getcwd()	getwd()
Change working directory	cd foo	os.chdir(’foo’)	setwd(’foo’)
Invoke a System Command	!notepad Octave: system("notepad")	os.system(’notepad’) os.popen(’notepad’)	system("notepad")

3.3 Debugging and profiling code

Language	MATLAB/Octave	Python	R
Most recent evaluated expression	ans		.Last.value
List variables loaded into memory	whos or who		objects()
Clear variable x from memory	clear x or clear [all]		rm(x)
Print	disp(a)	print a	print(a)

3.4 Conditionals

Language	MATLAB/Octave	Python	R	
if-statement	if 1>0 a=100; end	if 1>0: a=100	if (1>0) a <- 100	
if-else-statement	if 1>0 a=100; else a=0; end			
Ternary operator (if?true:false)			ifelse(a>0,a,0)	$a > 0?a : 0$

3.5 Loops

Language	MATLAB/Octave	Python	R
for-statement	`for i=1:5; disp(i); end`	`for i in range(1,6): print(i)`	`for(i in 1:5) print(i)`
Multiline for statements			

MATLAB/Octave

```
for i=1:5
   disp(i)
   disp(i*2)
end
```

Python

```
for i in range(1,6):
    print(i)
    print(i*2)
```

R

```
for(i in 1:5) {
    print(i)
    print(i*2)
}
```

---

4 File and Data input/output

Language	MATLAB/Octave	Python	R
Reading from a file (2d)	`f = load('data.txt')`	`f = fromfile("data.txt")` `f = load("data.txt")`	`f <- read.table("data.txt")`
Reading from a file (2d)	`f = load('data.txt')`	`f = load("data.txt")`	`f <- read.table("data.txt")`
Reading fram a CSV file (2d)	`x = dlmread('data.csv', ';')`	`f = load('data.csv', delimiter=';')`	`f <- read.table(file="data.csv", sep=";")`
Writing to a file (2d)	`save -ascii data.txt f`	`save('data.csv', f, fmt='%.6f', delimitwrier='te(f;'),file="data.txt")`	
Writing to a file (1d)		`f.tofile(file='data.csv', format='%.6f', sep=';')`	
Reading from a file (1d)		`f = fromfile(file='data.csv', sep=';')`	

---

5 Basic Operators

5.1 Getting help on operator syntax

Language	MATLAB/Octave	Python	R
Help on operator syntax	`help -`		`help(Syntax)`

5.2 Arithmetic operators

Language	MATLAB/Octave	Python	R
Assignment; defining a number	`a=1; b=2;`	`a=1; b=1`	`a<-1; b<-2`
Addition	`a + b`	`a + b` or `add(a,b)`	`a + b`
Subtraction	`a - b`	`a - b` or `subtract(a,b)`	`a - b`
Multiplication	`a * b`	`a * b` or `multiply(a,b)`	`a * b`
Division	`a / b`	`a / b` or `divide(a,b)`	`a / b`

Power, a^b	`a .^ b`	`a ** b`
		`power(a,b)`
		`pow(a,b)`
			`a ^ b`

Remainder	`rem(a,b)`	`a % b`
		`remainder(a,b)`
		`fmod(a,b)`
			`a %% b`

Integer division			`a %/% b`
In place operation to save array creation overhead	Octave: `a+=1`	`a+=b` or `add(a,b,a)`	
Factorial, $n!$	`factorial(a)`		`factorial(a)`

5.3 Relational operators

Language	MATLAB/Octave	Python	R
Equal	`a == b`	`a == b` or `equal(a,b)`	`a == b`
Less than	`a < b`	`a < b` or `less(a,b)`	`a < b`
Greater than	`a > b`	`a > b` or `greater(a,b)`	`a > b`
Less than or equal	`a <= b`	`a <= b` or `less_equal(a,b)`	`a <= b`
Greater than or equal	`a >= b`	`a >= b` or `greater_equal(a,b)`	`a >= b`
Not Equal	`a ~= b`	`a != b` or `not_equal(a,b)`	`a != b`

5.4 Logical operators

Language	MATLAB/Octave	Python	R
Short-circuit logical AND	`a && b`	`a and b`	`a && b`
Short-circuit logical OR	`a || b`	`a or b`	`a || b`
Element-wise logical AND	`a & b` or `and(a,b)`	`logical_and(a,b)` or `a and b`	`a & b`
Element-wise logical OR	`a | b` or `or(a,b)`	`logical_or(a,b)` or `a or b`	`a | b`
Logical EXCLUSIVE OR	`xor(a, b)`	`logical_xor(a,b)`	`xor(a, b)`
Logical NOT	`~a` or `not(a)` Octave: `~a` or `!a`	`logical_not(a)` or `not a`	`!a`
True if any element is nonzero	`any(a)`		
True if all elements are nonzero	`all(a)`		

5.5 Roots and logarithms

Language	MATLAB/Octave	Python	R	
Square root	`sqrt(a)`	`math.sqrt(a)`	`sqrt(a)`	$\sqrt{a}$
Logarithm, base e (natural)	`log(a)`	`math.log(a)`	`log(a)`	$\ln a = \log_e a$
Logarithm, base 10	`log10(a)`	`math.log10(a)`	`log10(a)`	$\log_{10} a$
Logarithm, base 2 (binary)	`log2(a)`	`math.log(a, 2)`	`log2(a)`	$\log_2 a$
Exponential function	`exp(a)`	`math.exp(a)`	`exp(a)`	e^a

5.6 Rounding

Language	MATLAB/Octave	Python	R
Round	`round(a)`	`around(a)` or `math.round(a)`	`round(a)`
Round up	`ceil(a)`	`ceil(a)`	`ceil(a)`
Round down	`floor(a)`	`floor(a)`	`floor(a)`
Round towards zero	`fix(a)`	`fix(a)`	

5.7 Mathematical constants

Language	MATLAB/Octave	Python	R
$\pi = 3.141592$	`pi`	`math.pi`	`pi`
$e = 2.718281$	`exp(1)`	`math.e` or `math.exp(1)`	`exp(1)`

5.8 Pseudo-random number generator

Language	MATLAB/Octave	Python	R
Uniform distribution	`rand(1,10)`	`random.random((10,))` `random.uniform((10,))`	`runif(10)`
Uniform: Numbers between 2 and 7	`2+5*rand(1,10)`	`random.uniform(2,7,(10,))`	`runif(10, min=2, max=7)`
Uniform: 6,6 array	`rand(6)`	`random.uniform(0,1,(6,6))`	`matrix(runif(36),6)`
Normal distribution	`randn(1,10)`	`random.standard_normal((10,))`	`rnorm(10)`

---

6 Basic vector construction

6.1 Vectors

Language	MATLAB/Octave	Python	R
Row vector, $1 \times n$-matrix	`a=[2 3 4 5];`	`a=array([2,3,4,5])`	`a <- c(2,3,4,5)`

Column vector, $m \times 1$-matrix	`adash=[2 3 4 5]';`	`array([2,3,4,5])[:,NewAxis]`
		`array([2,3,4,5]).reshape(-1,1)`
		`r_[1:10,'c']`
			`adash <- t(c(2,3,4,5))`

6.2 Sequences

Language	MATLAB/Octave	Python	R
1,2,3, ... ,10	`1:10`	`arange(1,11, dtype=Float)` `range(1,11)`	`seq(10)` or `1:10`
0.0,1.0,2.0, ... ,9.0	`0:9`	`arange(10.)`	`seq(0,length=10)`
1,4,7,10	`1:3:10`	`arange(1,11,3)`	`seq(1,10,by=3)`
10,9,8, ... ,1	`10:-1:1`	`arange(10,0,-1)`	`seq(10,1)` or `10:1`
10,7,4,1	`10:-3:1`	`arange(10,0,-3)`	`seq(from=10,to=1,by=-3)`
Linearly spaced vector of n=7 points	`linspace(1,10,7)`	`linspace(1,10,7)`	`seq(1,10,length=7)`
Reverse	`reverse(a)`	`a[::-1]` or	`rev(a)`
Set all values to same scalar value	`a(:) = 3`	`a.fill(3), a[:] = 3`	

6.3 Vector concatenation

Language	MATLAB/Octave	Python	R
Concatenate two vectors	`[a a]` `[1:4 a]`	`concatenate((a,a))` `concatenate((range(1,5),a), axis=1)`	`c(a,a)` `c(1:4,a)`