R, Matlab, Python cheat sheet

|  |  |  |  |
| --- | --- | --- | --- |
| Function | R | Python | Matlab |
| General |  |  |  |
| help on funcname | ?func | help(funcname) | doc funcname |
| variable class/type of x | class(x) | type(x) | class(x) |
| Formated print | print(sprintf(“pi:%$.2f”,pi)) | print("pi:%.2f" % (numpy.pi)) | fprintf(“pi:%.2f”,pi) |
| for loop | for (j in 1:5){  print( sprint(“node:%d”,j))  } | for j in range(5):  print(“node:”+str(j)) | for j=1:5  print(node+j) # not tested  end |
| while loop | while(TRUE){  print(j)  break  } | while True:  print(j)  break | while true  print(j)  break;  end |
| function | f <- function(x,y){ x+y } | def f(x,y):  x+y |  |
|  |  |  |  |
| Stings |  |  |  |
| Assignment | s = ‘hello world’ | s = ‘hello world’ | s = ‘hello world’ |
| Replacement | s = s.gsub(’hello’,’goodbye’,s) | s = s.replace(,’hello’,’goodbye’) | s = strrep(s,’hello’,’goodbye’) |
| Vectors |  |  |  |
| base | 1 based | 0 based | 1 based |
| vector element | v[[a]] (or v[i] usually) | v[i] | v(i) |
| vector | c(1,2,3) | [1,2,3] | [1,2,3] |
| print vector |  |  | flim=sprintf('%.4f ', lim); printf("lim:%s",seed,flim); |
| make x range from -1 to 1 by 0.1 | x <- 0.1\*(-10:10) | x = numpy.arange(-1,1,0.1) | x = -10:0.1:10 |
| dot product | a %\*% b | numpy.dot(a,b) | a .\* b |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| Matrices |  |  |  |
| dimensions | dim(m) | m.shape | size(m) |
| create | m=matrix(c(1,2,3,4),2,2) | m = numpy.array([[1,2],[3,4]]) | m = [1 2;3 4] |
| matrix element i,j | m[i,j] | m[I,j] | m(I,j) |
| 1st row, 2nd col | m[1,2] | m[0][1] | m(1,2) |
| transpose | t(m) | numpy.transpose(m) | m’ |
| flatten array |  |  | reshape(m,[],1) |
|  |  |  |  |
|  |  |  |  |
| Random Numbers |  |  |  |
| seed | set.seed(1234) | random.seed(1234) | rng(1234) |
| 10 randoms between 0&1 | runif(10) | [random.random() for i in range(10)]; | rand(10,1) |
| Plotting |  |  |  |
| plot x vs. y | plot(x,y) | matplotlib.pyplot.plot(x,y) | plot(x,y) |
|  |  |  |  |