Vectors, Matricies, Indexing

MATLAB	R	Description
a=[1 2 3 4];	a <- c(1,2,3,4)	Row vector
a=[1;2;3;4] OR a=[1 2 3 4]';	a <- t(c(1,2,3,4))	Column vector
a=[1 2 3; 4 5 6];	matrix(c(1,2,3,4,5,6),nrow=2,byrow=TRUE)	Matrix, enter by row
a=[1 2 3; 4 5 6];	matrix(c(1,4,2,5,3,6),nrow=2)	Matrix, enter by column
a(3)	a[3]	Access element #3
a=[2:7];	a <- 2:7 OR a <- c(2:7)	Vector [2 3 4 5 6 7]
a=[7:-1:2];	a <- 7:2	Vector [7 6 5 4 3 2]
a=[2:3:14];	a <- seq(2,14,3)	Vector [2 5 8 11 14]
x=linspace(a,b,n);	x <- seq(a,b,length.out=n) OR seq(a,b,len=n)	Vector with n equally spaced values between a and b, inclusive
a=zeros(a,1) OR a=zeros(1,a)	a <- rep(0,k)	Vector of length k with all zeros
a=j*ones(a,1) OR a=j*ones(1,a)	a <- rep(j,k)	Vecot of length k with all values j
a=zeros(m,n)	matrix(0,nrow=m,ncol=n) OR matrix(0,m,n)	Matrix of all zeros, size m by n
a=j*ones(m,n)	matrix(j,m,n)	Matrix of all j, size m by n
a=[a1 a2];	a <- cbind(a1,a2)	"Glue" two matricies a1 and a2 together (same # of rows)
[X,Y]=meshgrid(x,y)	m=length(x); n=length(y); X=matrix(rep(x,each=n),nrow=n); Y=matrix(rep(y,m),nrow=n)	X rows are copies of x, Y columns are copies of y
A(:,2)	A[,2]	Column 2 of matrix A
A(7,:)	A[7,]	Row 7 of matrix A
fliplr(A)	t(apply(A,1,rev))	Flip the order of elements in each row of matrix A
flipud(A)	apply(A,2,rev)	Flip the order of elements in each column of matrix A
v(a:end)	v[a:length(v)]	Extract elements of v from positon a to end
No simple way	v[c(-j,-k)]	All but the jth and kth elements of v
A = reshape(A,m,n)	dim(A) <- c(m,n)	Reshape matrix A into an m by n matrix (take elements columnwise from original matrix A)

Cell Array/List

MATLAB	R	Description
v=cell(1,n); v{1}=12; v{2}='hi there'; v{3}=rand(3);	v <- vector('list',n); v[[1]]=12; v[[2]]='hi there'; v[[3]] = matrix(runif(9),3)	Vector of length n capable of containing different data types in different elements (aka cell or list)
w=v{i}	w=v[[i]]	Extract the ith element of cell/list vector
No names associated with elements of cell arrays	names(v)[3] <- 'myrandmatrix'	Set the name of the ith element in a list
No names associated with elements of cell arrays	names(v)	See all names of elements in list
No names associated with elements of cell arrays	names(v) = NULL	Clear all names

Structures/Data Frames

avals=2*ones(1,6); yvals=6:-1:1; v=[1 5 3 2 3 7]; d = struct('a',avals,'y',yvals,'fac',v);	v <- c(1,5,3,2,3,7); d <- data.frame(cbind(a=2, y=6:1,v))	Create a matrix-like object with different named columns (*structure* in MATLAB, *data frame* in R)

Basic Computation

MATLAB	R	Description
a=1; b=2;	a <- 1; b <- 2;	Variable assignment
a+b; a*b; a^b; etc	a+b; a*b; a^b; etc.	Add, subtract, multiply, divide, power
--	a %/% b	Integer division
abs(a); exp(a); log(a); log10(a);	abs(a); exp(a); log(a); log10(a);	absolute value, e, ln, log base 10
sin(a); asin(a); sinh(a); asinh(a);	sin(a); asin(a); sinh(a); asinh(a);	sin, inverse sin, hyperbolic sin, inverse hyperbolic sin
mod(n,k)	n %% k	Remainder
round(x), floor(x), ceil(x)	round(x), floor(x), ceil(x)	Round, round down, round up
sign(x)	sign(x)	Sign of x (+1, 0, or -1)

Matrix Computations

MATLAB	R	Description
dot(x,y)	sum(x*y)	Vector dot product
cross(x,y)	xprod in RSEIS package	Vector cross product
A*B	A %*% B	Matrix multiplication AB
A.*B	A*B	Element-by-element multiplication of A and B
A\b	solve(A,b)	Solve Ax=b
mean(v), mean(A(:))	mean(v) or mean(A)	Mean of all elements in vector or matrix
mean(A), sum(A)	colMeans(A), colSums(A)	Means or sums of columns in a matrix
mean(A,2), sum(A,2)	rowMeans(A), rowSums(A)	Means or sums of rows in a matrix
std(v), std(A(:))	sd(v), sd(c(A))	Standard deviation of all elements in a vector of matrix, normalized by (n-1)
std(A)	sd(A)	Standard deviations of columns of a matrix

Matrix Computations (cont)

std(A,2)	apply(A,1,sd)	Standard deviations of rows of a matrix
min(v), min(A(:))	min(v), min(A)	Minimum of all elements in vector or matrix
min(A)	apply(A,2,min)	Minimum value in each column of A
min(A,[],2)	apply(A,1,min)	Minimum value in each row of matrix A
min(A,c)	pmin(A,c)	Given matrix A and scalar c, compute a matrix where each element is the minimum of c and corresponding element of A
[y,ind]=min(v)	ind = which.min(v)	Find index of the first time min(v) appears in v and store that index as ind
size(A,1)	nrow(A)	Number of rows in A
size(A,2)	ncol(A)	Number of columns in A
size(A)	dim(A)	Dimensions of A, listed as a vector
length(v)	length(v)	Number of elements in vector
numel(A)	length(A)	Number of elements in matrix
sort(v)	sort(v)	Sort values in vector v
[s,idx]=sort(v)	tmp <- sort(v,index,return=TRUE); s <- tmp$s; idx=tmp$ix	Sort values in v, putting sorted values in s and indicies in idx, in teh sense that s[k] = x[idx[k]]
find(v>5)	which(v>5)	List of indicies of each element of v which are greater than 5
[r,c] = find(A>5)	w <- which(A>5, arr.ind=TRUE); r <- w[,1]; c <- w[,2];	Generate r and c giving rows and columns of elements of A which are greater than 5

Relationals and Logicals

MATLAB	R	Description
a == b; a < b; a >= b; etc.	a == b; a < b; a >= b; etc.	eq, gt, lt, gte, lte
a ~= b	a != b	Not equal
a && b; a || b;	a && b; a || b;	AND, OR
~a	!a	Not

GUI Commands

MATLAB	R	Description
clear all	rm(list=ls())	Clear all variables
close all	graphics.off()	Close all figures
clc	ctrl+L	Clear console