linear algebra

Includes things like matrices and vectors to optimize our code.

vector is a 1-dimensional array of numbers. column vector: row vector:

matrix is a 2-dimensional array of numbers.

initialization

Identity matrix (If you multiply a matrix with it, you get the original matrix):

M = eye(n); % n is the size of the identity matrix

matrix of zeroes:

M = zeros(n, m); % the matrix will be the size n \* m

matrix of ones:

M = ones(n, m); % the matrix will be the size n \* m

random:

M = rand(n, m); % the matrix will be the size n \* m

custom:

M = [1, 2, 3; 4, 5, 6; 7, 8, 9];

\*replace m with 1 if you want a vector.

matrix-vector multiplication

n \* m m \* 1 n \* 1

new\_M = M \* v;

matrix multiplication

n \* m m \* o n \* o

Just the concatenation of A times column vectors of B.

new\_M = A \* B;

element-wise operations

Add a dot in front of the operator.

A = [1, 2; 3, 4];

B = [5, 6; 7, 8];

A \* B; % matrix multiplication

A .\* B; % multiply each element with the same positions

C = 10;

A .+ C % add all elements in A by 10

other useful functions

A = pinv(B); % returns inverse of B

A = B’ % returns transpose of B

reshape(A, n, m) % reshapes the elements in A into a n \* m matrix