test\_hyperplane.m

%%

%%Hyperplane in 2D

close all;

a = [1;-1];

b = [1,2];

drawhyperplane2D(a,b);

drawhyperplane2D.m

%%This function plot hyperplane(line) of given a(column) and b(row) vector in 2D

function x2 = drawhyperplane2D(a,b)

%a1x1+a2x2 -a1b1-a2b2 = 0

%If we leave x2 alone: x2 = -a1/a2\*x1 - transpose(a)\*transpose(b)

%I defina m as coeff. of x1, n as the constant.

m = -a(1)/a(2);

n = transpose(a)\*transpose(b)/a(2);

%I set the appropriate range

x1 = -3:0.1:3;

%Value of x2 for given x1 vector

x2 = m\*x1 +n;

%Ploting the graph

plot(x1,x2);

%Labeling the aces

xlabel('x1')

ylabel('x2')

hold on

%Ploting normal vector in b1,b2 point

quiver(b(1),b(2),a(1),a(2));

test\_hyperplane.m

%%

%Hyperplane in 3D

close all;

A = [2 ; -1 ; 2];

B = [1 ; 2 ; 1] ;

drawhyperplane3D(A,B);

drawhyperplane3D.m

%%This function plots hyperplane of given a(column )and b(column) vectors in 3D.

function x3 = drawhyperplane3D(a,b)

%a1x1 + a2x2 + a3x3 - a1b1 - a2b2 - a3b4 = 0

%If we leave x3 alone: x3 = -a1/a3\*x1-a2/a3\*x2+transpose(a)\*transpose(b)/a3

% I define coeff. of x1 as 'm1', coeff. of x2 as 'm2', the constant as 'n'

n = transpose(a)\*b/a(3);

m1 = -a(1)/a(3);

m2 = -a(2)/a(3);

%The plot 3D graph I generated the mesh in a enough range of X and Y

[x1,x2] = meshgrid(b(1)-5:b(1)/5:b(1)+5);

%I calculated the x3 values for given XY mesh.

x3 = m1\*x1 + m2\*x2+ n ;

%Ploting the function

surf(x1,x2,x3);

%Labeling axes

xlabel('x1')

ylabel('x2')

zlabel('x3')

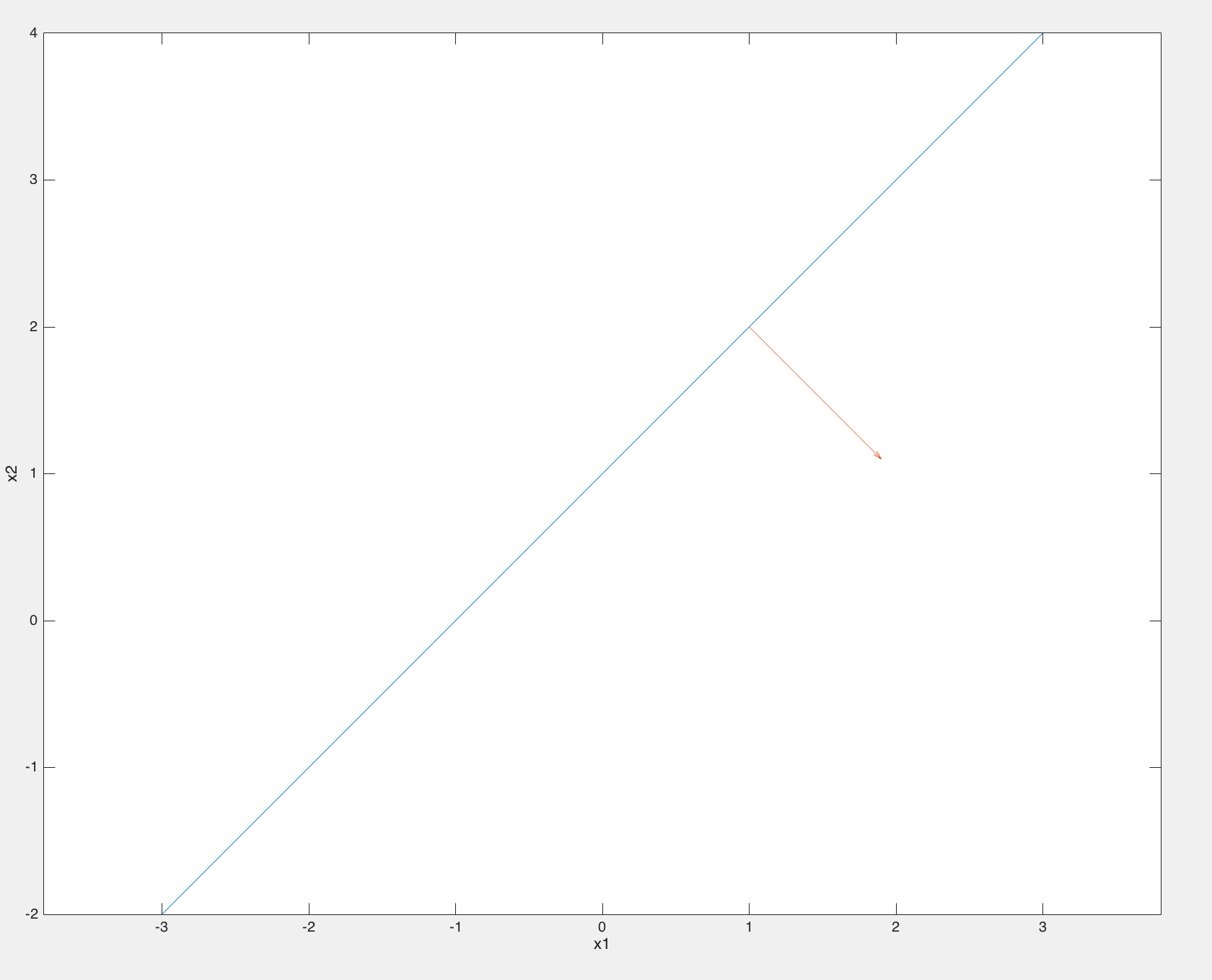
hold on

%I plot the normal vector in b1,b2,b3 point, and largen a bit.

quiver3( b(1),b(2),b(3),a(1),a(2),a(3),2, 'linewidth', 5, 'MaxHeadSize', 0.9, 'color', 'red');

%Appropriate view angle is set

view(62,20)



hyperplane2D

