VISIÓ PER

COMPUTADOR

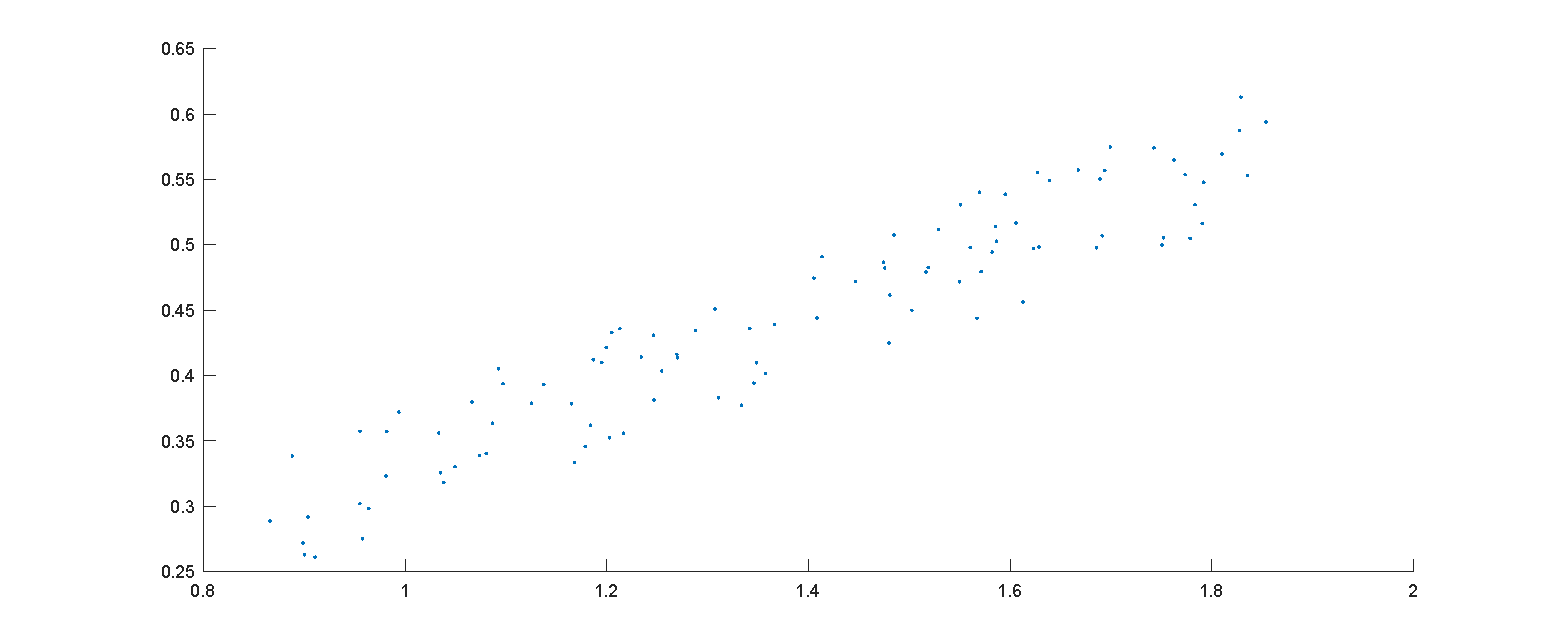
Informe de laboratori 1

Marc Cervilla Rovira

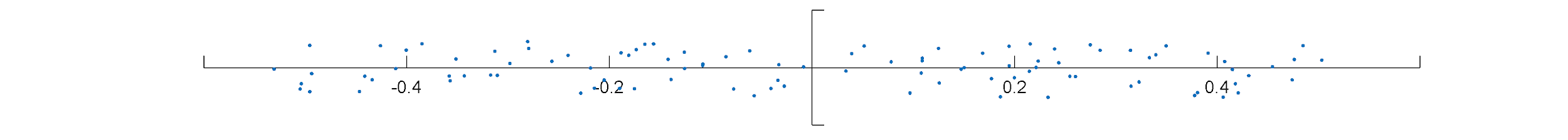
24/02/2021

Introducció:

En aquest primer exercici de laboratori començarem a practicar amb el Matlab a partir del següent procediment de transformació de dades: l’objectiu serà obtenir un conjunt de dades aleatòries amb un pendent semblant al de la figura a) i centrar aquest conjunt a l’eix horitzontal tal com mostra la figura b).



a)



b)

Implementació:

En primer lloc obtenim el núvol de dades:

% Point cloud creation

x = rand(1,100) + rand(); % 100 punts aleatoris amb un offset també aleatori

y = rand().\*x + rand(1,100)/10; % pendent de valor aleatori i offset també aleatori

i utilitzant la funció scatter podem veure el gràfic generat:

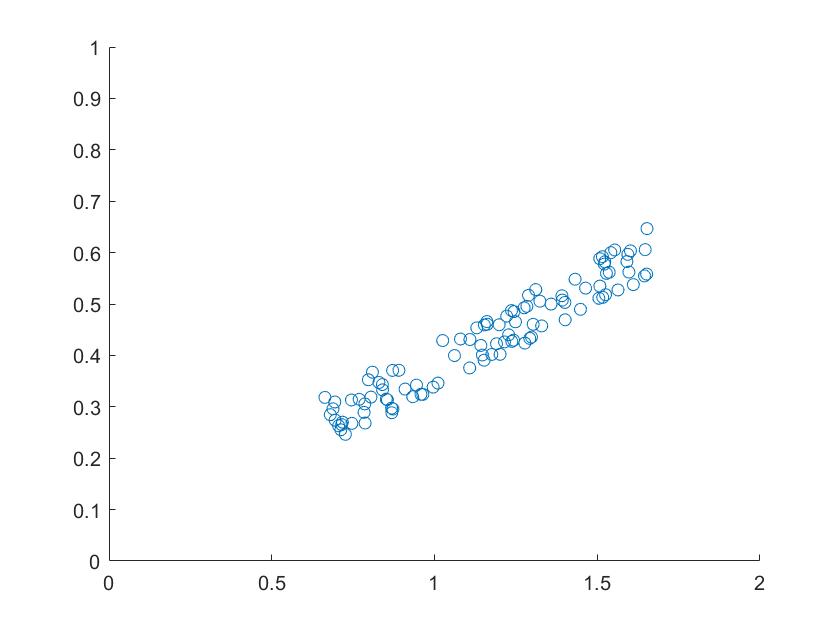


FIGURA 1

A continuació, restem la mitjana de cada vector als diferents punts per centrar-ho a 0.

% Cloud point centering

xp = x - mean(x);

yp = y - mean(y);

Utilitzem la covariança per trobar l’eix amb més variança, per tal de trobar l’angle theta amb el que rotarem les dades. Per treure aquest angle he utilitzat els valors: evectors(ind,1) i evectors(ind,2).

% Covariance and eigen values

c = cov(xp, yp);

[evectors, evalues] = eig(c);

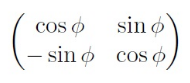
% Determine which dimension has the major variance

[val,ind] = max(diag(evalues));

% Extract the angle of the ‘major axis’

theta = -pi/2-atan2(evectors(ind,1),evectors(ind,2));

Buscant per internet he trobat que la matriu de rotació en sentit horari es defineix:



Per tant generem la matriu, rotem els punts(en sentit horari) i els dibuixem.

% Create clockwise rotation matrix

R = [cos(theta) sin(theta); -sin(theta) cos(theta)]; % heu de buscar a internet que cal posar aquí

% Rotate the points

rp = R \* [xp;yp];

% Draw the points

%scatter(rp(1,:),rp(2,:));

Així doncs ens queda el següent gràfic:

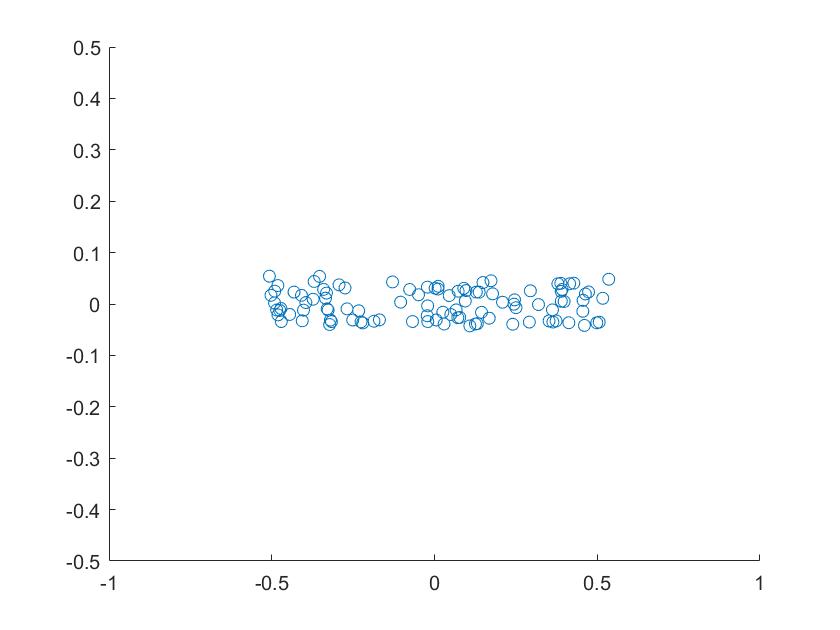


FIGURA 2

Per tant, hem aconseguit passar de la figura inicial amb un núvol de punts amb pendent aleatori a la mateixa figura però centrada a 0 i en l’eix horitzontal. Provant diferents execucions del script obtenim els resultats aplicant el mateix procediment a diferents punts amb diferents pendents.

Per tal d’ajustar els eixos de les figures anteriors per a que no es reajustin he utilitzat el següent codi:

FIGURA 1:

figure(1);

scatter(x,y);

axis([0 2 0 1]);

FIGURA 2:

figure(2);

scatter(rp(1,:),rp(2,:));

axis([-1 1 -0.5 0.5]);

Si volguessim fer un procediment semblant amb una gràfica 3D penso que es podria aconseguir afegint la 3a dimensió tant als punts de la gràfica com a les matrius de rotació i dels eigen values(utilitzant matrius 3x3 en comptes de 2x2), ja que necessitarem 3 vectors per senyalar la direcció del núvol. Tot i això, es complicaria més la visualització de punts en les figures.