Facultat d'Informàtica de Barcelona
Departament d'Enginyeria de Sistemes, Automàtica i Informàtica Industrial
UNIVERSITAT POLITÈCNICA DE CATALUNYA

VISIÓ PER COMPUTADOR

Exercici 1 de Laboratori

Facultat d'Informàtica de Barcelona

Manel Frigola
Joan Climent

Barcelona, Febrer de 2021

Daniel Escribano

Exercici 1.

El següent exercici ens ajudarà a practicar amb el Matlab alhora que s'introdueix a un procediment de transformació de dades molt útil pel capítol de classificació d'imatges.
L'objectiu és transformar un núvol de punts (en aquest cas aleatoris), que tinguin més o menys l'aspecte de la figura a), per centrar-los en l'eix horitzontal, tal i com es mostra en la figura b).

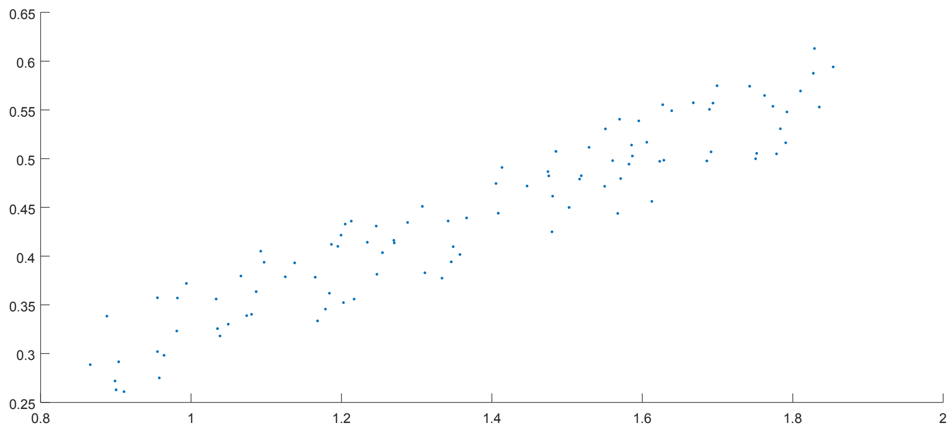


Fig a)

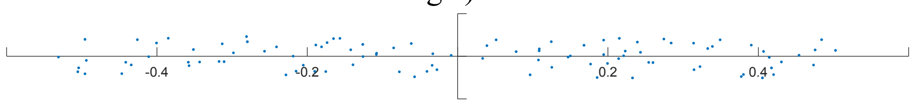


Fig b)

1. Per crear un núvol de punts (x,y) com els de la figura a) farem servir valors aleatoris mitjançant la funció rand.

% 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

2. Per representar-los gràficament és suficient utilitzar la funció *scatter* i ja tenim la figura a)

scatter(x,y);

3. Per obtenir la figura b) cal girar el núvol de punts (x,y) en relació al seu centre de masses amb un angle igual a la seva pendent. Primer de tot centrem el núvol de punts en el zero (restar la seva mitjana).

% Cloud point centering
xp = x - mean(x);
yp = y - mean(y);

4. Calcular l'angle que té el núvol de punts amb l'eix horitzontal. Es podria fer una regressió lineal i calcular l'angle de la recta de regressió. Una altre manera de fer-ho, més complexa però més generalitzable a N dimensions, és utilitzar la covariància i veure quin eix és el que té major variància.

% 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)**);**

He determinat que havia de fer servir el *eigenvector* de la variable que tenia major variància, escollint aquest vector gràcies a la variable *ind* .

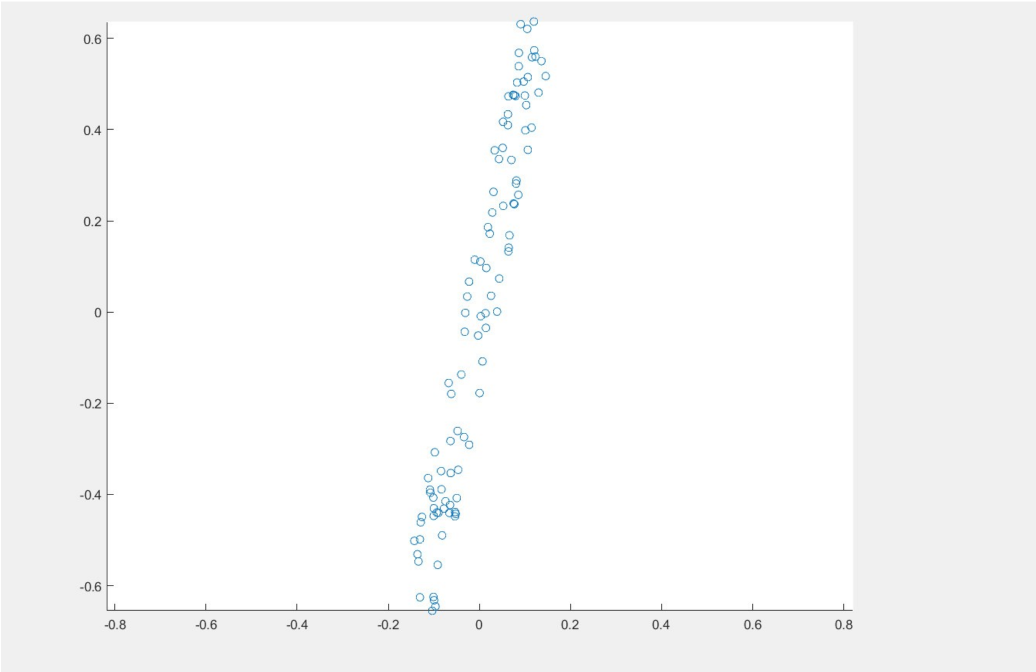
5. Ara, utilitzant una matriu de rotació R, girem els punts un angle theta

% Create clockwise rotation matrix
R = [cos(theta) -sin(theta); sin(theta) cos(theta)**];**

% Rotate the points
rp = R * [xp;yp];

Buscant a internet he trobat que la matriu de rotació apropiada era aquesta. He extret aquesta informació del enllaç https://alyssaq.github.io/2015/visualising-matrices-and-affine-transformations-with-python/#rotating

Tot i que en l'enunciat ens demanava crear una matriu de rotació en sentit horari, he decidit deixar una matriu en sentit de rotació antihorari, seguint les indicacions de la pàgina web. En realitzar la matriu de rotació en sentit horari ([cos(theta) sin(theta); -sin(theta) cos(theta)]) obtenia plots semblant al següent:

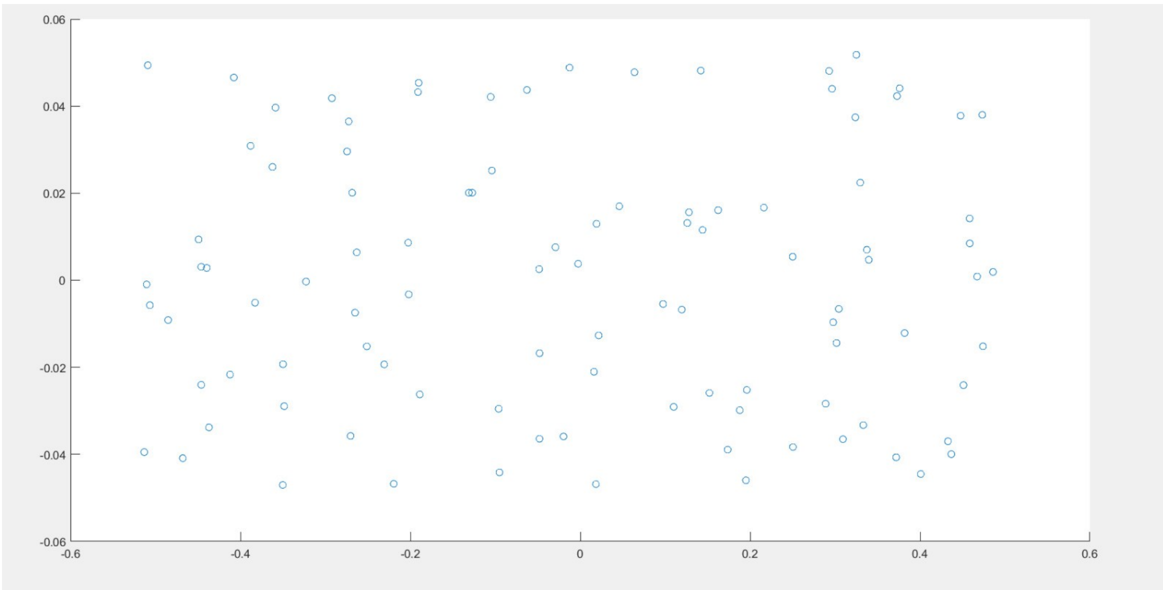


En veure que la majoria dels plots tendien a quedar en posició vertical, he decidit deixar la matriu en sentit antihorari ja que obtenia plots que tendien a quedar en posició horitzontal i l'exercici ens demana modificar el nuvol de punts perquè s'assembli a la imatge b).

6. Finalment, fem servir una altre vegada la funció *scatter* per representar la figura b)

% Draw the points
figure
scatter(rp(1,:),rp(2,:));
axis('equal');

Per aconseguir una imatge semblant a la imatge b) necessitava afegir la funció *axis('equal'),* ja que si no afegia aquesta els eixos de la figura estaven descompensats, com es mostra en la imatge següent:



Daniel Escribano

Proveu amb diferents núvols de punts amb diferents inclinacions. Ajusteu les propietats dels eixos de les figures per a que no es re-escalin les dades.
Reflexioneu si es podria fer quelcom similar amb un núvol de punts 3D.
Feu un informe amb el que heu fet i entregueu-lo a Atenea.

En referència a si es podrien fer coses similars amb un núvols de punts 3D, jo trobo que si.

En aquest exercici hem aplicat PCA(Principal Component Analysis), que consisteix en agafar els ¨components principals¨ i treballar amb ells. Aquests components principals són els *eigenvectors* de la matriu de covariància, vectors que ens indiquen direccions. Els *eigenvalues*, d'altra banda, ens diuen quina es la magnitud de la variància associada al *eigenvector*.
Llavors, el *eigenvector* amb un major *eigenvalue* ens determina la direcció en que hi ha la major variància en les dades.

Així doncs, fent ús del *eigenvector* amb major *eigenvalue* i una matriu de covariància de 3 dimension, podem treballar de forma similar en un núvol de punts 3D.