Mineria de dades: PAC4 - Models Agregació

Autor: Nom estudiant

Novembre 2018

# Introducció

## Presentació

Aquesta prova d’avaluació contínua cobreix els Mòduls 5 i 8 (Avaluació de models) del programa de l’assignatura

## Competencies

Les competències que es treballen en aquesta prova són:

* Ús i aplicació de les TIC en l’àmbit acadèmic i professional.
* Capacitat per innovar i generar noves idees.
* Capacitat per avaluar solucions tecnològiques i elaborar propostes de projectes tenint en compte els recursos, les alternatives disponibles i les condicions de mercat.
* Conèixer les tecnologies de comunicacions actuals i emergents així com saber-les aplicar convenientment per dissenyar i desenvolupar solucions basades en sistemes i tecnologies de la informació.
* Aplicació de les tècniques específiques d’enginyeria del programari en les diferents etapes del cicle de vida d’un projecte.
* Capacitat per aplicar les tècniques específiques de tractament, emmagatzematge i administració de dades.
* Capacitat per proposar i avaluar diferents alternatives tecnològiques per resoldre un problema concret.

## Objetius

La correcta assimilació del Mòdul 5: En aquesta PAC treballarem la generació i interpretació d’un model d’agregació basat en particions de dades amb l’eina de pràctiques. No perdrem de vista les fases de preparació de les dades i extracció inicial de coneixement.

## Descripció de la PAC a realitzar

La prova està estructurada en un total d’1 exercici teòric i 2 exercicis teòric / pràctic. Aquests exercicis estan tots relacionats, és necessari fer-los tots per obtenir una valoració satisfactòria.

## Recursos Bàsics

**Material docent proporcionat per la UOC.**

Mòdul 5 i 8 del material didàctic.

**Complementaris**

* Fitxer iris.csv

## Criteris de valoració

**Exercicis teòrics**

Tots els exercicis han de ser presentats de forma raonada i clara, especificant tots i cadascun dels passos que s’hagin dut a terme per a la seva resolució. No s’acceptarà cap resposta que no estigui clarament justificada.

**Exercicis pràctics**

Per totes les PAC és necessari documentar a cada apartat de l’exercici pràctic què s’ha fet i com s’ha fet.

## Format i data de lliurament

El format de lliurament és: usernameestudiant-PECn.html/doc/docx/odt/pdf Data de Lliurament: 21/11/2018 S’ha de lliurar la PAC en la bústia de lliuraments de l’aula

## Nota: Propietat intel·lectual

Sovint és inevitable, en produir una obra multimèdia, fer ús de recursos creats per terceres persones. És per tant comprensible fer-ho en el marc d’una pràctica dels estudis d’Informàtica, Multimèdia i Telecomunicació de la UOC, sempre que això es documenti clarament i no suposi plagi en la pràctica.

Per tant, en presentar una pràctica que faci ús de recursos aliens, s’ha de presentar juntament amb ella un document en què es detallin tots ells, especificant el nom de cada recurs, el seu autor, el lloc on es va obtenir i el seu estatus legal: si l’obra està protegida pel copyright o s’acull a alguna altra llicència d’ús (Creative Commons, llicència GNU, GPL …). L’estudiant haurà d’assegurar-se que la llicència no impedeix específicament el seu ús en el marc de la pràctica. En cas de no trobar la informació corresponent haurà d’assumir que l’obra està protegida per copyright.

Haureu de, a més, adjuntar els fitxers originals quan les obres utilitzades siguin digitals, i el seu codi font si correspon.

# Exemple 1

En aquest exemple anem a generar un conjunt de mostres aleatòries per posteriorment fer servir l’algorisme kmeans per agrupar-les. Es crearan les mostres al voltant de dos punts concrets. Per tant, el lògic serà agrupar en dos clústers ja que a priori, en un problema real, no es coneix com és el nombre correcte de clústers k, anem a provar primer amb dos clústers (el valor òptim) i posteriorment amb 4 i 8 clústers. Per avaluar la qualitat de cada procés d’agrupació anem a fer servir la silueta mitjana. La silueta de cada mostra avalua com de bé o malament està classificada la mostra en el clúster al que ha estat assignada. Per a això s’usa una fórmula que té en compte la distància a les mostres del seu clúster i la distància a les mostres del clúster veí més proper.

A l’hora de provar el codi que es mostra, és important tenir en compte que les mostres es generen de forma aleatòria i també que l’algoritme kmeans té una inicialització aleatòria. Per tant, en cada execució s’obtindrà uns resultats lleugerament diferents.

El primer que fem és carregar la llibreria clúster que conté les funcions que es necessiten

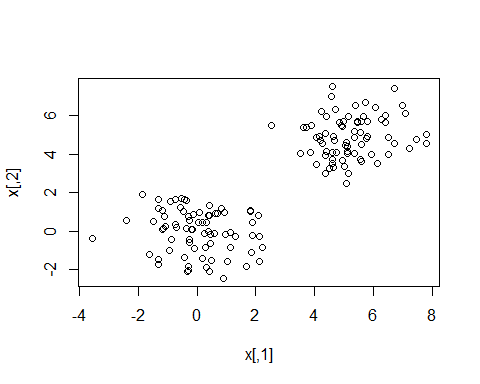
library(cluster)

Generem les mostres de forma aleatòria prenent com a centre els punts [0,0] i [5,5].

n <- 150 # numero de mostres  
p <- 2 # dimension  
  
sigma <- 1 # variància de la distribució  
mean1 <- 0 # centre del primer grup  
mean2 <- 5 # centre del segon grup  
n1 <- round(n/2) # numero de mostres del primer grup  
n2 <- round(n/2) # numero de mostres del segon grup  
  
x1 <- matrix(rnorm(n1\*p,mean=mean1,sd=sigma),n1,p)  
x2 <- matrix(rnorm(n2\*p,mean=mean2,sd=sigma),n2,p)

Ajuntem totes les mostres generades i les mostrem en una gràfica

x <- rbind(x1,x2)  
plot (x)



Com es pot comprovar les mostres estan clarament separades en dos grups. Si es vol complicar el problema es pot modificar els punts centrals (mean1 i mean2) fent que estiguin més propers i/o ampliar la variància (sigma) perquè les mostres estiguin més disperses.

A continuació anem a aplicar l’algorisme kmeans amb 2, 4 i 8 clústers

fit2 <- kmeans(x, 2)  
y\_cluster2 <- fit2$cluster  
  
fit4 <- kmeans(x, 4)  
y\_cluster4 <- fit4$cluster  
  
fit8 <- kmeans(x, 8)  
y\_cluster8 <- fit8$cluster

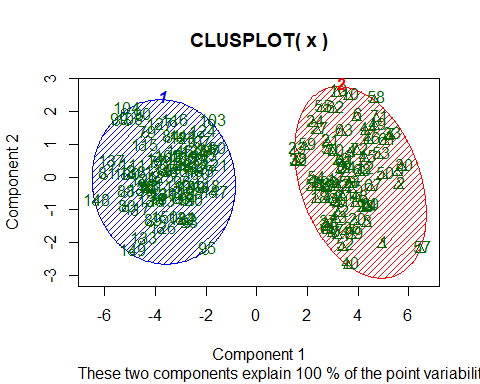
Les variables i\_cluster2, i\_cluster4 i i\_cluster8 contenen per a cada mostra l’identificador del clúster a les quals han estat assignades. Per exemple, en el cas dels k=2 les mostres s’han assignat al cluster 1 o al 2

y\_cluster2

## [1] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [36] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [71] 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [106] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [141] 1 1 1 1 1 1 1 1 1 1

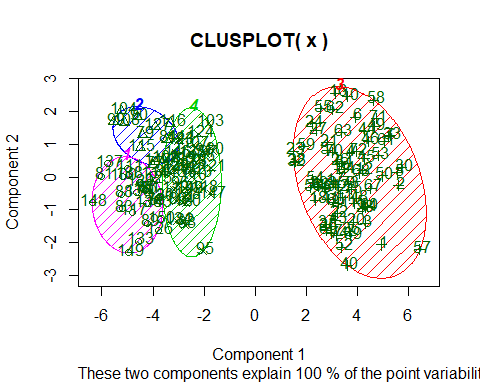
Per veure els clústers podem utilitzar la funció clusplot. Veiem l’agrupació amb 2 clusters

clusplot(x, fit2$cluster, color=TRUE, shade=TRUE, labels=2, lines=0)



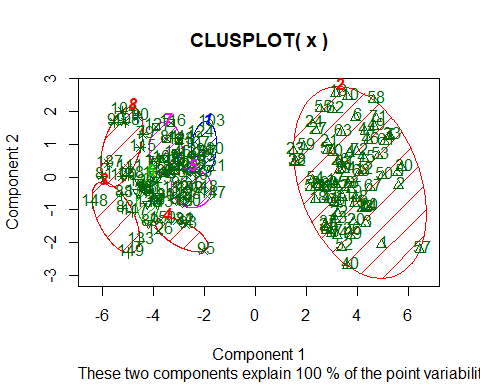
con 4

clusplot(x, fit4$cluster, color=TRUE, shade=TRUE, labels=2, lines=0)



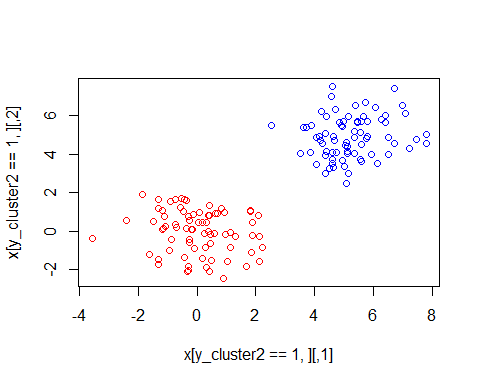
y con 8

clusplot(x, fit8$cluster, color=TRUE, shade=TRUE, labels=2, lines=0)



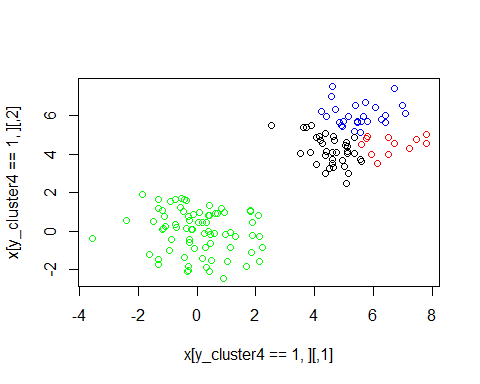
També podem visualitzar el resultat del procés d’agrupament amb el següent codi per al cas de 2 clústers

plot(x[y\_cluster2==1,],col='blue', xlim=c(min(x[,1]), max(x[,1])), ylim=c(min(x[,2]), max(x[,2])))  
points(x[y\_cluster2==2,],col='red')



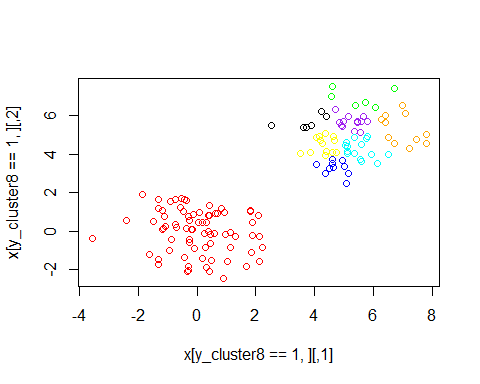
per a 4

plot(x[y\_cluster4==1,],col='blue', xlim=c(min(x[,1]), max(x[,1])), ylim=c(min(x[,2]), max(x[,2])))  
points(x[y\_cluster4==2,],col='red')  
points(x[y\_cluster4==3,],col='green')  
points(x[y\_cluster4==4,],col='black')



i per a 8

plot(x[y\_cluster8==1,],col='blue', xlim=c(min(x[,1]), max(x[,1])), ylim=c(min(x[,2]), max(x[,2])))  
points(x[y\_cluster8==2,],col='red')  
points(x[y\_cluster8==3,],col='green')  
points(x[y\_cluster8==4,],col='black')  
points(x[y\_cluster8==5,],col='yellow')  
points(x[y\_cluster8==6,],col='purple')  
points(x[y\_cluster8==7,],col='cyan')  
points(x[y\_cluster8==8,],col='orange')



Ara anem a avaluar la qualitat del procés d’agregació. Per a això usarem la funció silhouette que calcula la silueta de cada mostra

d <- daisy(x)   
sk2 <- silhouette(y\_cluster2, d)  
sk4 <- silhouette(y\_cluster4, d)  
sk8 <- silhouette(y\_cluster8, d)

La funció silhouette retorna per a cada mostra, el clúster on ha estat assignat, el clúster veí i el valor de la silueta. Per tant, calculant la mitjana de la tercera columna podem obtenir una estimació de la qualitat de l’agrupament

mean(sk2[,3])

## [1] 0.7283502

mean(sk4[,3])

## [1] 0.524232

mean(sk8[,3])

## [1] 0.4889355

Com es pot comprovar, agrupar amb dos clústers és millor que en 4 o en 8, la qual cosa és lògic tenint en compte com s’han generat les dades.

# Exemple 2

A continuació anem a veure un altre exemple de model d’agregació. Per a això farem servir el fitxer iris.csv. Aquesta base de dades es troba descrita en <https://archive.ics.uci.edu/ml/datasets/iris>. Com es pot comprovar, aquestes dades estan pensades a problemes de classificació supervisada on es pretén classificar cada tipus de flor en un de les tres classes existents. Com en aquest exemple anem a usar un mètode no supervisat, transformarem el problema supervisat original en un no supervisat. Per a consegir-ho no usarem la columna class, que és la variable que es vol predir. Per tant, intentarem trobar agrupacions usant únicament els quatre atributs que caracteritzen a cada flor.

Carreguem les dades i ens quedem únicament amb les quatre columnes que defineixen cada flor

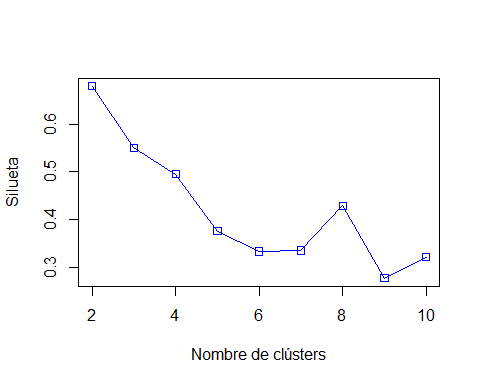
iris\_data<-read.csv("./iris.csv", header=T, sep=",")  
attach(iris\_data)  
x <- iris\_data[,1:4]

Com a priori no coneixem el nombre optimo de clusters, provem amb diversos valors

d <- daisy(x)   
resultados <- rep(0, 10)  
for (i in c(2,3,4,5,6,7,8,9,10))  
{  
 fit <- kmeans(x, i)  
 y\_cluster <- fit$cluster  
 sk <- silhouette(y\_cluster, d)  
 resultados[i] <- mean(sk[,3])  
}

Mostrem en una gràfica els valors de les siluetes mitjana de cada prova per comprovar que nombre de clusters és el millor

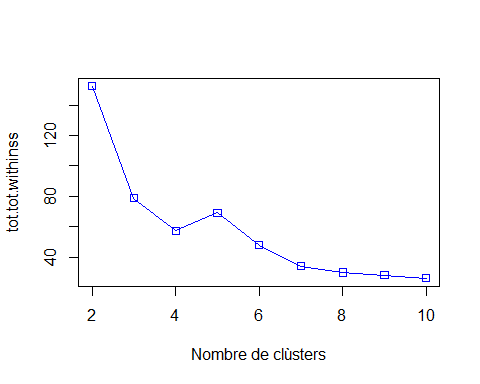
plot(2:10,resultados[2:10],type="o",col="blue",pch=0,xlab="Nombre de clústers",ylab="Silueta")



Encara que el valor a priori esperat és k=3 ja que que el conjunt original té 3 classes, el millor valor que s’obté és k=2.

Una altra manera d’avaluar quin és el millor nombre de clústers és considerar com a millor model aquell que ofereix la menor suma dels quadrats de les distàncies dels punts de cada grup pel que fa al seu centre (withinss), amb la major separació entre centres de grups (betweenss). Com es pot comprovar és una idea conceptualment similar a la silueta. Una manera comuna de fer la selecció del nombre de clusters consisteix en aplicar el mètode “elbow” (colze), que no és més que la selecció del nombre de clústers sobre la base de la inspecció de la gràfica que s’obté en iterar amb el mateix conjunt de dades per a diferents valors del nombre de clústers. Es triarà el valor que es troba en el “colze” de la corba

resultados <- rep(0, 10)  
for (i in c(2,3,4,5,6,7,8,9,10))  
{  
 fit <- kmeans(x, i)  
 resultados[i] <- fit$tot.withinss  
}  
plot(2:10,resultados[2:10],type="o",col="blue",pch=0,xlab="Nombre de clùsters",ylab="tot.tot.withinss")



En aquest cas el nombre òptim de clusters són 4 que és quan la corba comença a estabilitzar-se.

També es pot fer servir la funció kmeansruns del paquet fpc que executa l’algorisme kmeans amb un conjunt de valors i selecciona el valor del nombre de clusters que millor funcioni d’acord a dos criteris: la silueta mitjana (“asw”) i Calinski-Harabasz (“ch”).

library(fpc)  
fit\_ch <- kmeansruns(x, krange = 1:10, criterion = "ch")   
fit\_asw <- kmeansruns(x, krange = 1:10, criterion = "asw")

Podem comprovar el valor amb el que s’ha obtingut el millor resultat i també mostrar el resultat obtingut per a tots els valors de k usant tots dos criteris

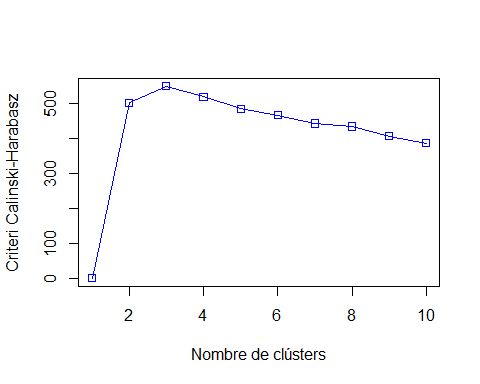
fit\_ch$bestk

## [1] 3

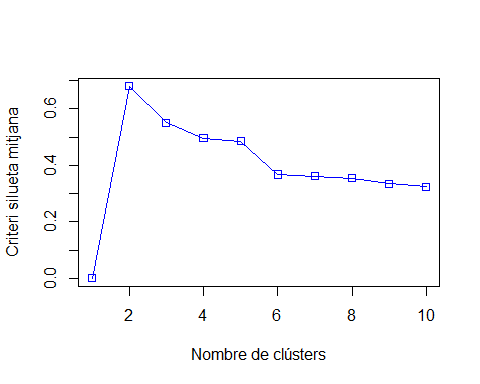
fit\_asw$bestk

## [1] 2

plot(1:10,fit\_ch$crit,type="o",col="blue",pch=0,xlab="Nombre de clústers",ylab="Criteri Calinski-Harabasz")



plot(1:10,fit\_asw$crit,type="o",col="blue",pch=0,xlab="Nombre de clústers",ylab="Criteri silueta mitjana")



Els resultats són molt semblats als quals hem obtingut anteriorment. Amb el criteri de la silueta mitjana s’obtenen dos clústers i amb el Calinski-Harabasz s’obtenen 3.

Com s’ha comprovat, conèixer el nombre òptim de clusters no és un problema fàcil. Tampoc ho és l’avaluació dels models d’agregació.

# Exercicis

## Exercici 1:

Tenint en compte el projecte que has definit a la PAC1, creieu que els mètodes d’agregació són el mètode més adequat per aconseguir algun dels objectius que us heu proposat? Justifica adequadament la resposta.

Si la resposta és no, proposar un nou projecte on si sigui adequat usar models d’agregació.

### Resposta 1:

Escriu aquí la resposta a la pregunta

## Exercici 2:

Prenent com a punt de partida els exemples mostrats, realitzar un estudi similar amb un altre conjunt de dades. Poden ser dades reals del vostre àmbit laboral o d’algun repositori de dades d’Internet. Mireu per exemple: [http://www.ics.uci.edu/~mlearn/\*MLSummary.html](http://www.ics.uci.edu/~mlearn/*MLSummary.html).

A l’hora de triar la base de dades tingues en compte que sigui apropiada per a problemes no supervisats i que els atributs siguin també apropiats per al seu ús amb l’algorisme kmeans.

No us oblideu de la fase de preparació i anàlisi de dades. ### Resposta 2: > Escriu aquí la resposta a la pregunta

## Exercici 3:

Busca informació sobre altres mètodes d’agregació diferents al Kmeans. Partint de l’exercici anterior prova el funcionament d’almenys 2 mètodes diferents i compara els resultats obtinguts en tots dos exercicis.

### Resposta 3:

Escriu aquí la resposta a la pregunta

# Rúbrica

## Exercici 1

* 20%. S’explica de forma clara si l’ús de mètodes d’agregació és adequat per aconseguir els objectius proposats en el cas plantejat en les PAC1. Si no ho és, es proposa un cas on si ho és.

## Exercici 2

* 3%. Se selecciona una base de dades adequada per a l’ús de kmeans.
* 7%. S’expliquen els camps de la base de dades, preparació i anàlisis de dades.
* 10%. S’aplica l’algorisme d’agrupament de forma correcta.
* 10%. Es proven amb diferents valors de k.
* 5%. S’obté una mesura del bé que és l’agrupament.
* 10%. S’expliquen les conclusions que s’obtenen.
* 5%. Es presenta el codi i és fàcilment reproduïble.

## Exercici 3

* 10%. Es prova un algorisme diferent al kmeans.
* 10%. Es prova un altre algorisme diferent al kmeans.
* 10%. Es comparen els resultats del kmeans i els altres dos mètodes provats en aquest exercici.