boost
accelerate your learning

Introducción a la Ciencia de Datos

Sesión 1:

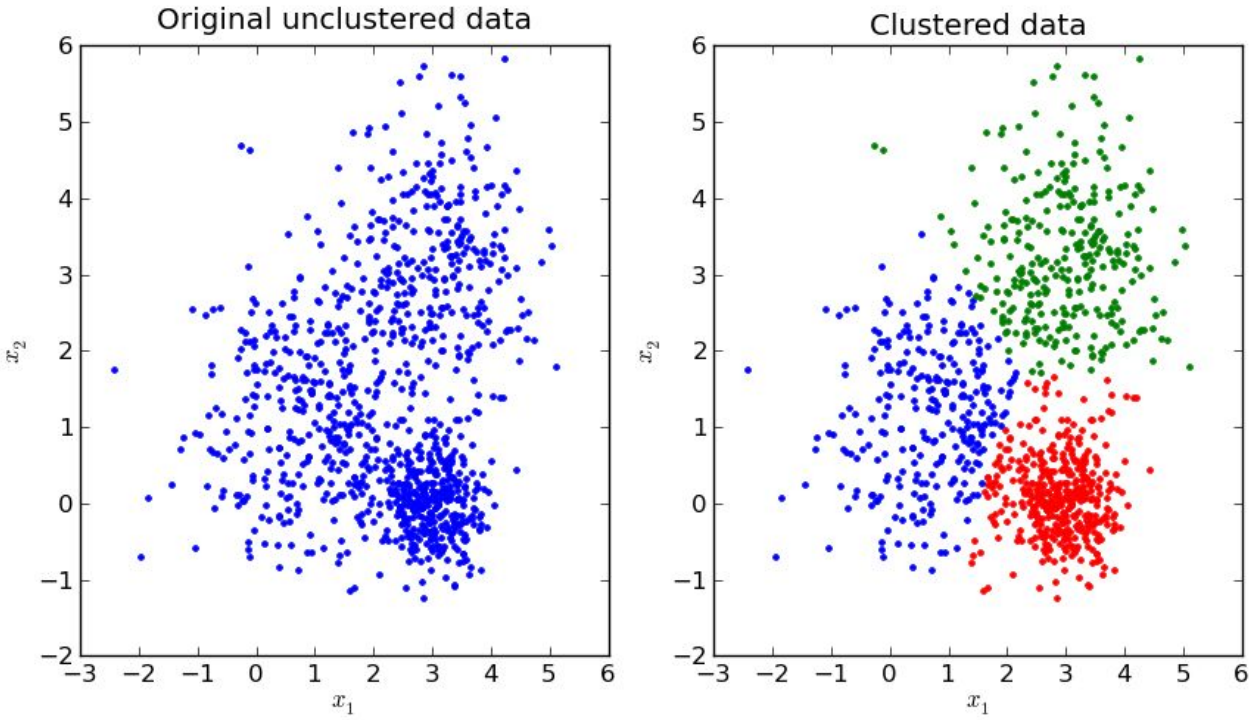
boost
accelerate your learning

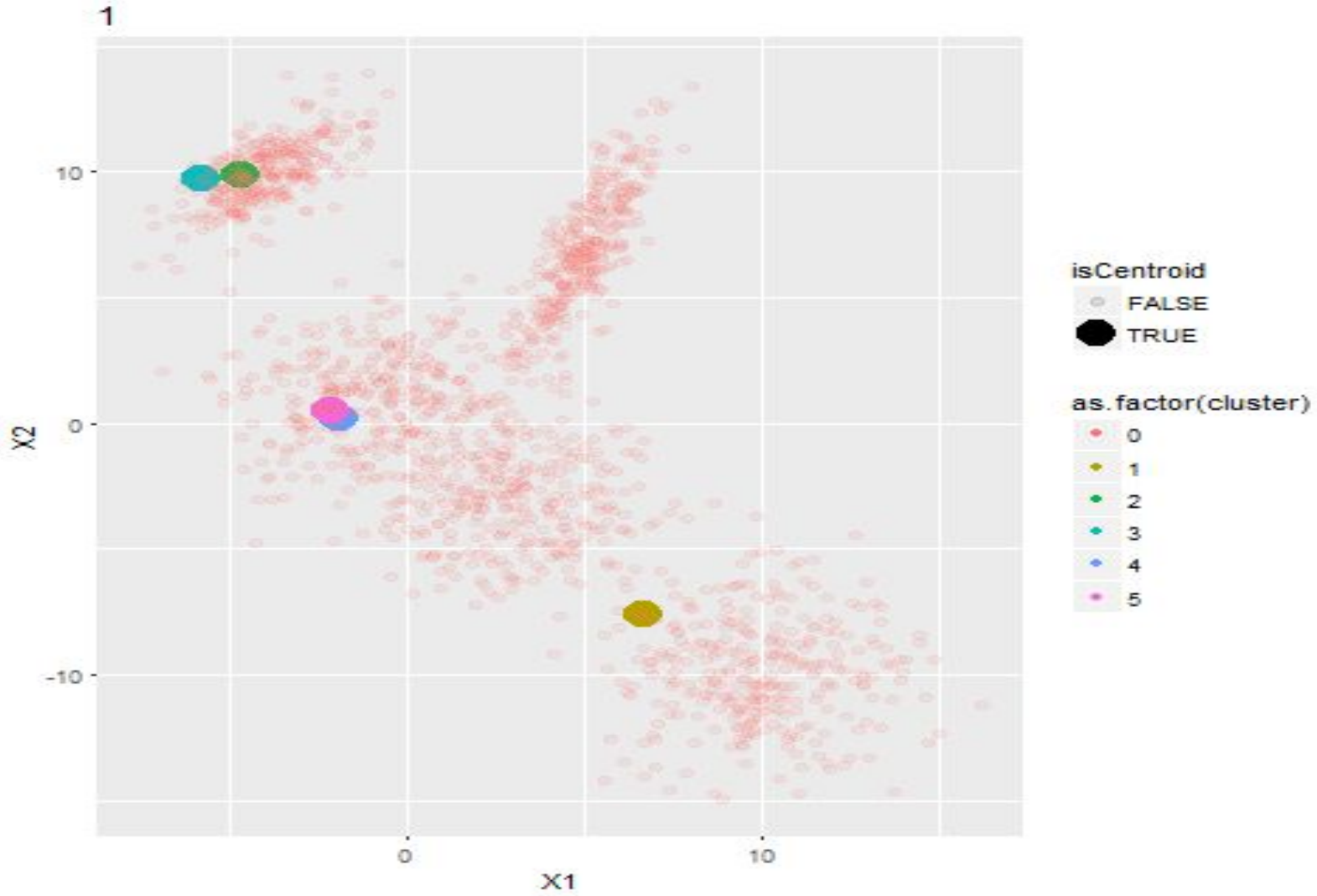
Aprendizaje No-Supervisado

Aprendizaje No-Supervisado

- Se consigue producir conocimiento sin necesidad de explicarle al sistema qué resultados queremos obtener.
- Sin necesidad de que ser supervise la respuesta, se pueden detectar patrones y clasificar la data.
- La máquina puede saber qué tan parecidos son una cantidad de datos con respecto a otra. (clustering)

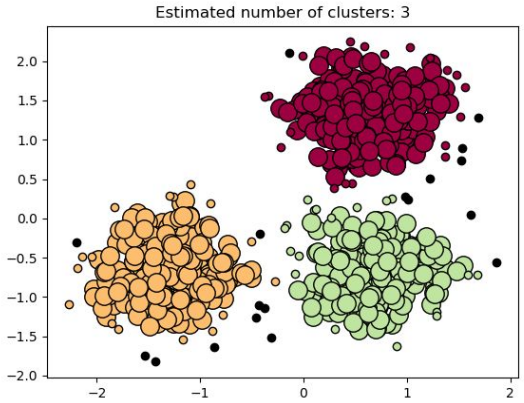
Métodos de Clustering




1
10
0
−10
X2
0
10
X1
isCentroid
FALSE
TRUE
as.factor(cluster)
0
1
2
3
4
5

KMeans

1. Asigna valores centros aleatorios dentro de la muestra.
2. Estos centros forman el primer cluster y se calcula la media del cluster asignándose como nuevo centro.
3. Este centro es el nuevo centroide y calcula los valores más cercanos para toda la muestra, asignándoles a un cluster.
4. Se repiten los pasos 2 y 3 cuantas veces se quiera...

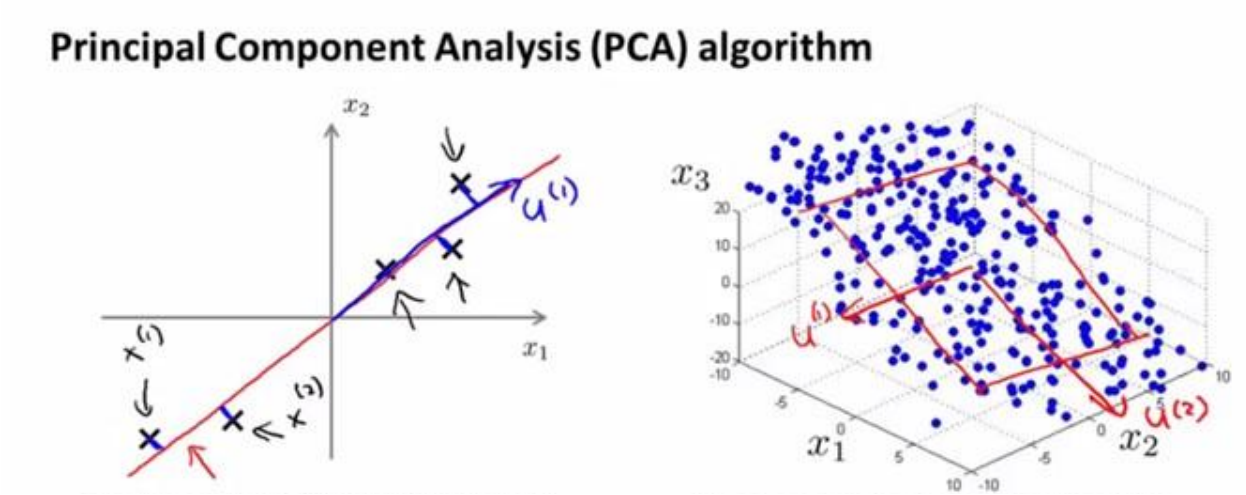


¿Cómo elegimos la mejor cantidad de Clusters para nuestra data?

La respuesta se verá en el Bonus Notebook!

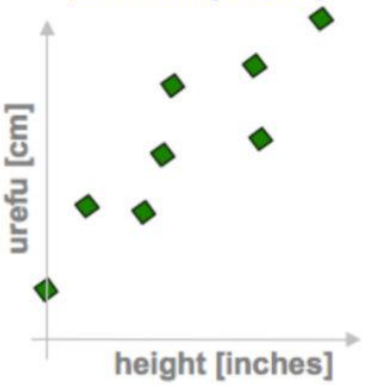
PCA (Principal Component Analysis)

El Análisis de Componentes Principales (PCA) nos ofrece reducir las dimensiones de nuestra data para, principalmente, poder visualizarla con mayor facilidad. Así como para resumir lo más importante de la data. Pero OJO, al reducir las dimensiones también se puede perder información valiosa para el análisis.

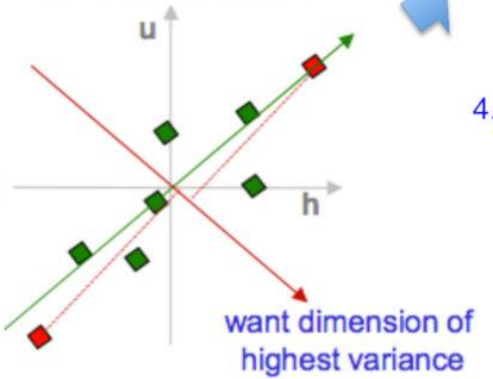


PCA in a nutshell

1. correlated hi-d data
("urefu" means "height" in Swahili)



2. center the points



3. compute covariance matrix

$$\begin{array}{c} \\ h \\ u \end{array}\begin{array}{c} \begin{array}{cc} h & u \end{array} \\ \begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix} \end{array} \rightarrow \mathrm{cov}(h,u) = \frac{1}{n}\sum_{i=1}^{n} h_i u_i$$

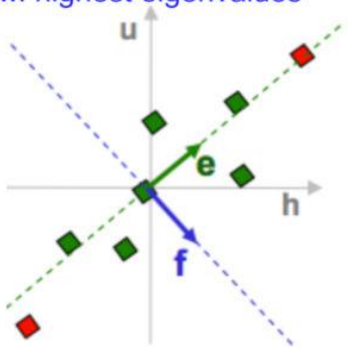
4. eigenvectors + eigenvalues

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}\begin{pmatrix} e_h \\ e_u \end{pmatrix} = \lambda_e \begin{pmatrix} e_h \\ e_u \end{pmatrix}$$

$$\begin{pmatrix} 2.0 & 0.8 \\ 0.8 & 0.6 \end{pmatrix}\begin{pmatrix} f_h \\ f_u \end{pmatrix} = \lambda_f \begin{pmatrix} f_h \\ f_u \end{pmatrix}$$

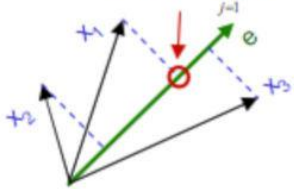
```
eig(cov(data))
```

5. pick m<d eigenvectors
w. highest eigenvalues



6. project data points to
those eigenvectors

$$x_e' = x^T e = \sum_{j=1}^{a} x_{ij} e_j$$



7. uncorrelated low-d data