

K SCHOOL

Máster Data Science

Barcelona, 2019.
Henry Navarro

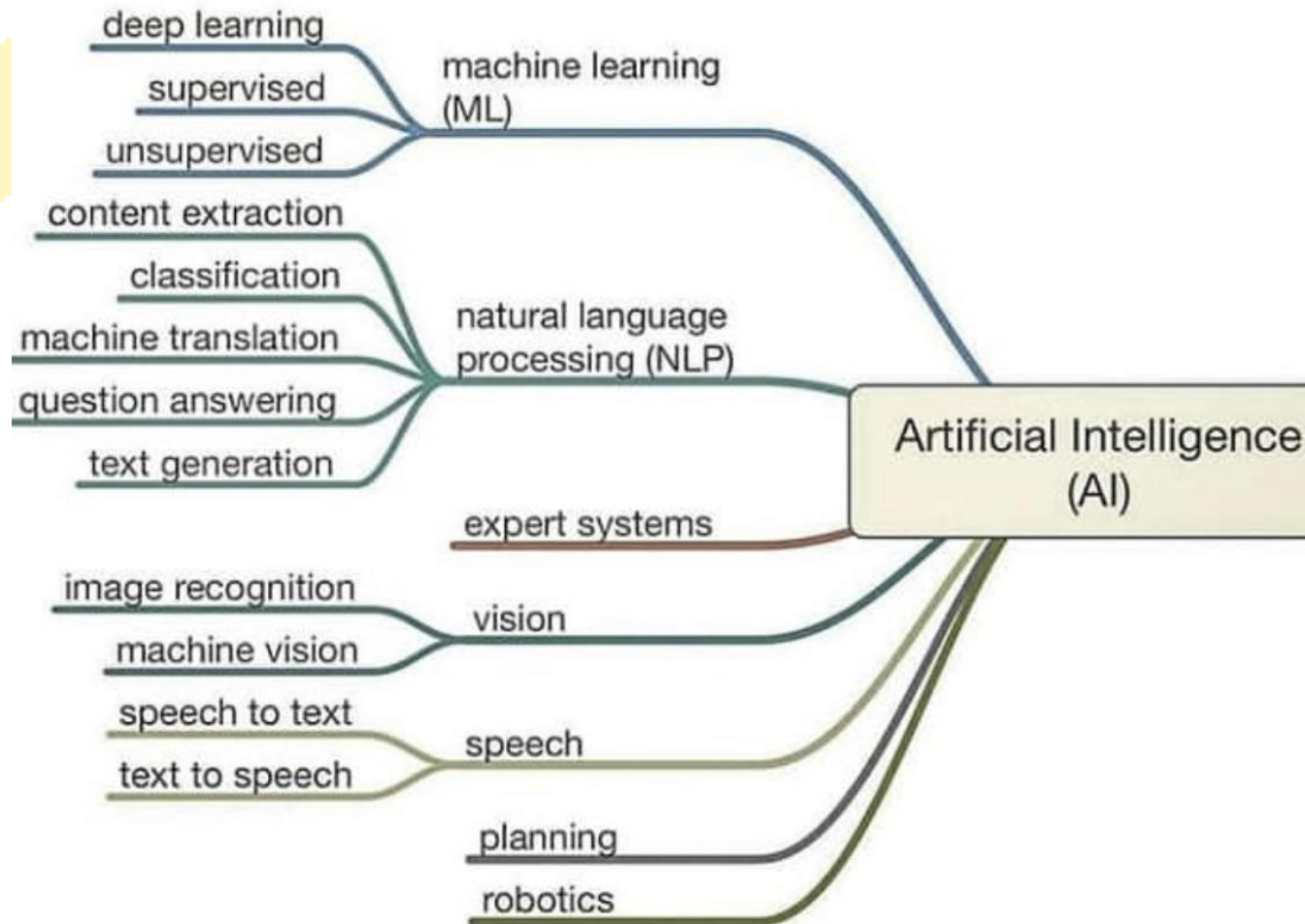




About me

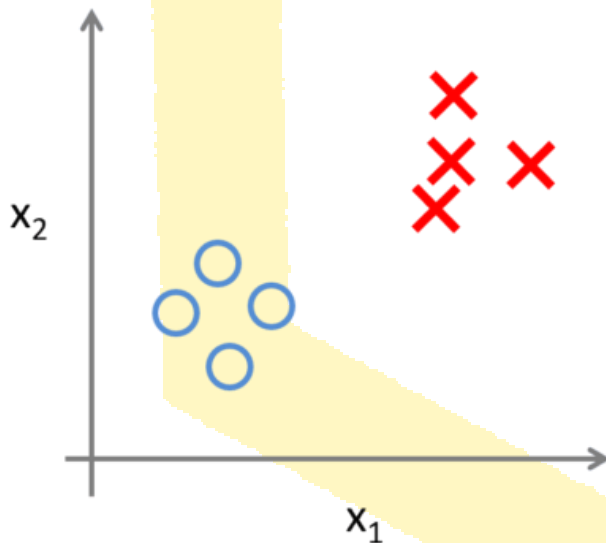
- Ahora: Lead Data Scientist – Altran Innovation.
- Antes: Data Scientist – Equifax, Solutio, Enefgy.
- Profesor Universidad Carlos III de Madrid – Grupo ML4DS & GISC. (<https://goo.gl/9eeHAz>)
- Profesor Escuela de Organización Industrial.
- Data Analyst – Ministerio de Justicia (Venezuela).
- Máster en Ingeniería Matemática – UC3M.
- Licenciado en Matemática – UCV.

Intro unsupervised learning

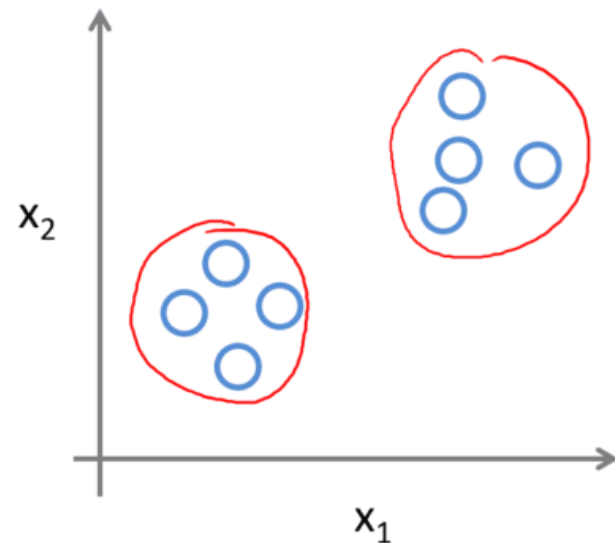


Supervised vs. Unsupervised learning

Supervised Learning

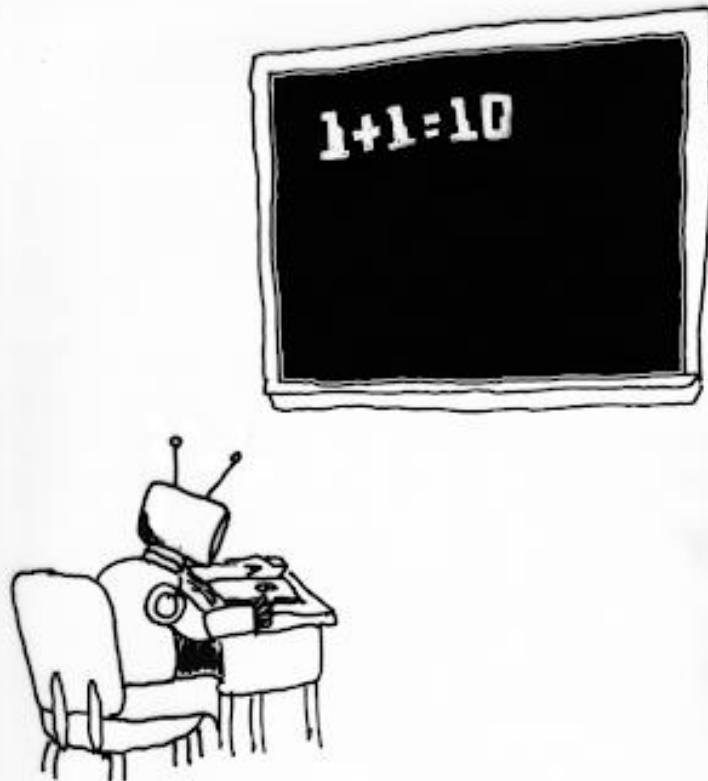


Unsupervised Learning

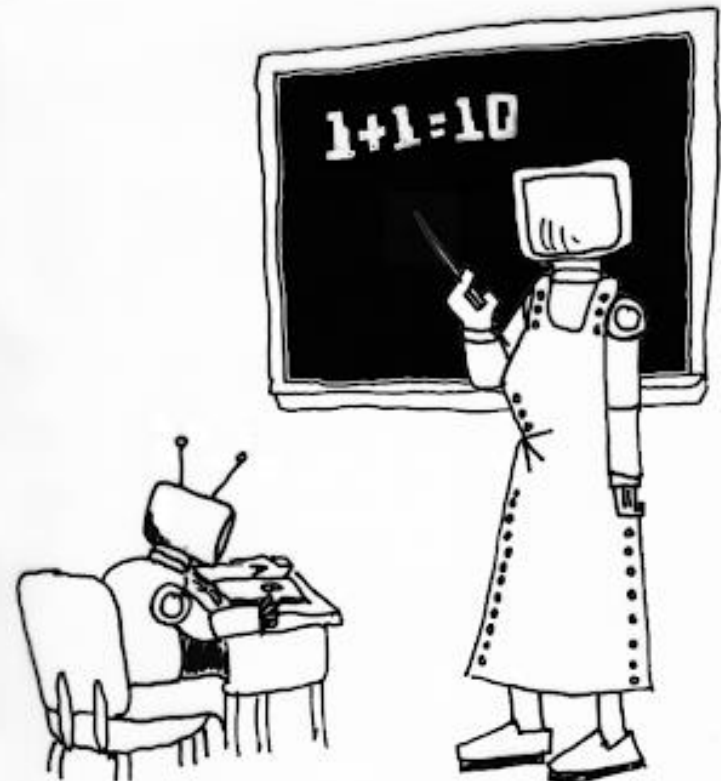


Supervised vs. Unsupervised learning

UNSUPERVISED MACHINE LEARNING



SUPERVISED MACHINE LEARNING



Variable
respuesta/objetivo/
dependiente



supervised learning

y	x1	x2	x3	x4	x5	x6
0	186.176767	32.01013	7.389389	56.68727	171.3361	18.03844
1	159.659374	-95.06651	-83.210200	155.31220	-149.0119	-183.90314
0	44.307132	-167.88587	-90.023000	124.17956	170.8277	30.37569
0	129.380781	-83.71101	193.529927	193.97078	135.2245	-157.56599
0	-7.236501	150.92669	-75.665873	58.89800	-114.4337	-58.16047
0	-13.191041	51.07507	168.874093	-73.05704	-179.1995	-178.97354

unsupervised learning

x1	x2	x3	x4	x5	x6
186.176767	32.01013	7.389389	56.68727	171.3361	18.03844
159.659374	-95.06651	-83.210200	155.31220	-149.0119	-183.90314
44.307132	-167.88587	-90.023000	124.17956	170.8277	30.37569
129.380781	-83.71101	193.529927	193.97078	135.2245	-157.56599
-7.236501	150.92669	-75.665873	58.89800	-114.4337	-58.16047
-13.191041	51.07507	168.874093	-73.05704	-179.1995	-178.97354

No tenemos
variable respuesta

Index

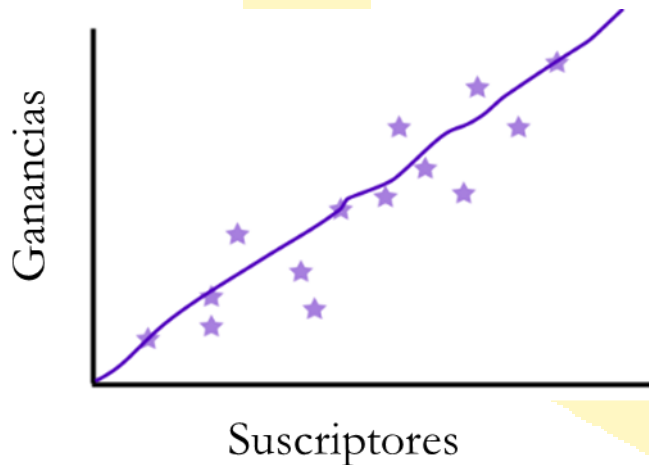
- K-means
- Hierarchical Clustering
- Principal component analysis.
- t-distributed Stochastic Neighbor Embedding
- K-means vs. Hierarchical Clustering

¿Qué hacemos si nuestro conjunto de datos no tiene etiquetas?

- El aprendizaje no supervisado es un grupo de algoritmos de machine learning que funcionan bajo el principio de “verdad sin fundamento”.
- Pensemos en el ejemplo de queremos saber la relación entre el número de suscriptores youtube y las ganancias de un canal de esta plataforma.

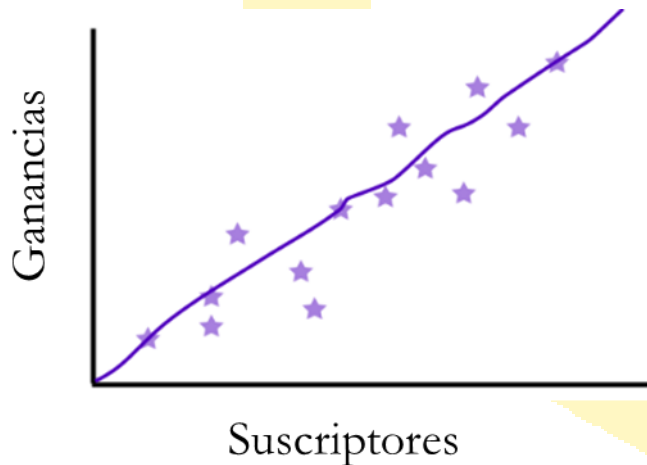
¿Qué hacemos si nuestro conjunto de datos no tiene etiquetas?

- El aprendizaje no supervisado es un grupo de algoritmos de machine learning que funcionan bajo el principio de “verdad sin fundamento”.
- Pensemos en el ejemplo de queremos saber la relación entre el número de suscriptores youtube y las ganancias de un canal de esta plataforma.



¿Qué hacemos si nuestro conjunto de datos no tiene etiquetas?

- El aprendizaje no supervisado es un grupo de algoritmos de machine learning que funcionan bajo el principio de “verdad sin fundamento”.
- Pensemos en el ejemplo de queremos saber la relación entre el número de suscriptores youtube y las ganancias de un canal de esta plataforma.

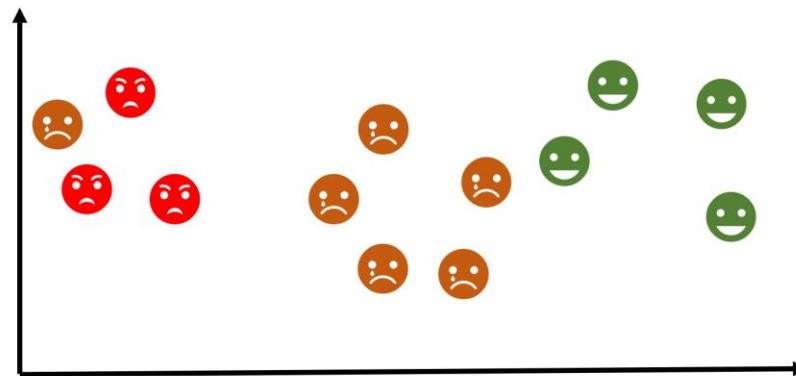


¿Qué hacemos si nuestro conjunto de datos no tiene etiquetas?

- Tal vez no tengamos acceso a los datos salariales, o simplemente estamos interesados en diferentes preguntas. ¡No importa! Lo importante es que no hay una salida con la que coincidir, ni una línea para dibujar que represente una relación.
- Entonces, ¿cuál es exactamente el objetivo del aprendizaje no supervisado? ¿Qué hacemos cuando solo tenemos datos de entrada sin etiquetas?

¿Qué hacemos si nuestro conjunto de datos no tiene etiquetas?

- Tal vez no tengamos acceso a los datos salariales, o simplemente estamos interesados en diferentes preguntas. ¡No importa! Lo importante es que no hay una salida con la que coincidir, ni una línea para dibujar que represente una relación.
- Entonces, ¿cuál es exactamente el objetivo del aprendizaje no supervisado? ¿Qué hacemos cuando solo tenemos datos de entrada sin etiquetas?



Tipos de algoritmos no supervisados

1. Clustering

1. K-means.
2. Hierarchical Clustering.
3. Probabilistic Clustering.

2. Data Compression

1. Principal Component Analysis.
2. Singular Value Decomposition (u otras factorizaciones de matrices).
3. t-Distributed Stochastic Neighbor Embedding.

3. Unsupervised Deep Learning

1. Autoencoders.
2. Anomaly detection.

Conceptos previos

- Distancia
- Scaling
- Missing Value imputation

Métrica

Una métrica $d: X \times X \rightarrow [0, \infty)$ es una función que satisface las siguientes condiciones para cada $x, y \in X$:

- $d(x, y) \geq 0$, no negativa.
- $d(x, y) = 0 \iff x = y$, identidad indecible.
- $d(x, y) = d(y, x)$, simetría.
- $d(x, z) \leq d(x, y) + d(y, z)$, desigualdad triangular.

Métrica (Ejemplos)

- Distancia euclideana.
- Distancia de Minkowski.
- Distancia de Manhattan.
- Distancia de Levenshtein.
- Distancia del infimo y supremo.

Scaling

Es necesario en la mayoría de casos normalizar la escala de valores de las variables para comenzar con el proceso de clustering. Esto se debe a que los valores de las variables de cada observación se representan como coordenadas en el espacio n -dimensional (n es el número de variables) y luego se calculan las distancias entre estas coordenadas. Si estas coordenadas no están normalizadas, puede dar lugar a resultados falsos.

Por ejemplo, supongamos que tenemos unos datos de peso y altura de tres personas: A (6ft, 75kg), B (6ft, 77kg), C (8ft, 75kg). Tendríamos:

- A-B: 2 unidades.
- A-C: 2 unidades.

Tendríamos que ambos pares A-B y A-C son similares.

Scaling

Existen varias formas de normalizar los valores de las variables. Uno es la estandarización de la escala completa de todos los valores de las variables, es decir, que $x(i) \in [0,1]$, conocida como min-max normalization, y se obtiene aplicando la siguiente transformación:

$$x(s) = \frac{x(i) - \min(x)}{\max(x) - \min(x)}$$

También puede usarse la que seguramente muchos conocen:

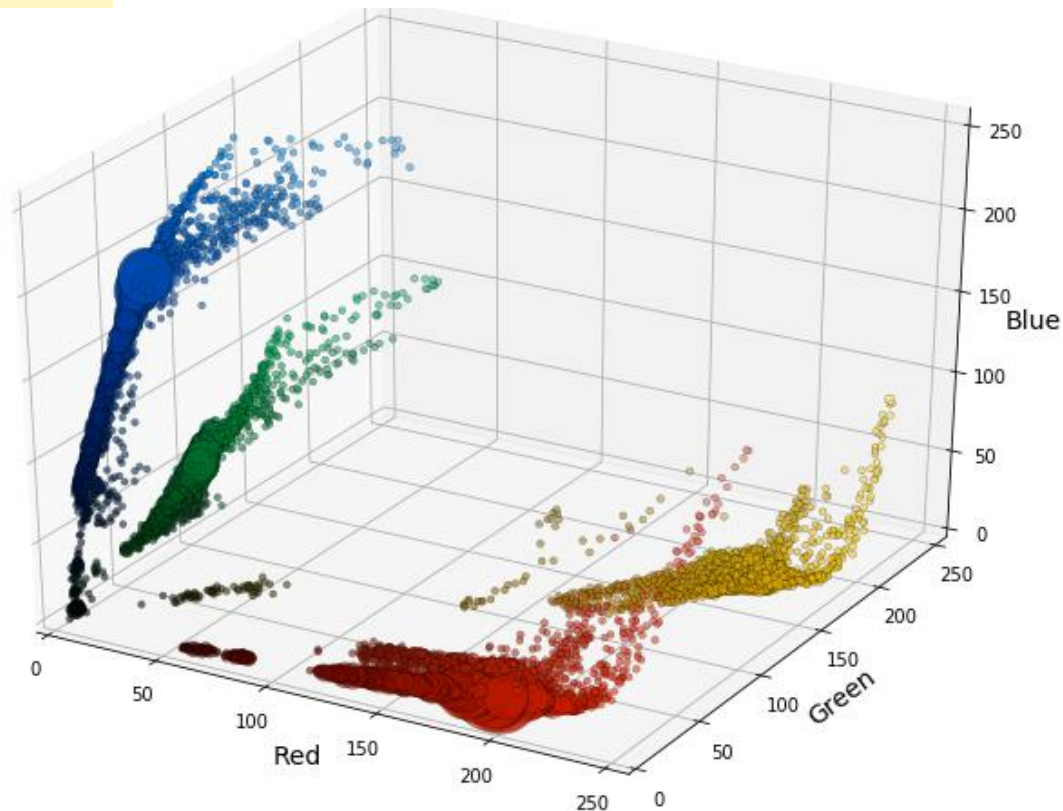
$$x(s) = \frac{x(i) - \mu(x)}{\sigma(x)}$$

Missing Value imputation

Básicamente, "eliminemos" los fucking NA's...



K-means, viejo, sencillo y potente.



K-means clustering

- La idea básica detrás del K-means consiste en definir agrupaciones de modo que se minimice la variación total dentro de la agrupación
- Existen diferentes algoritmos de K-means y modificaciones tales como K-means++, Fuzzy c-Means, X-means (el más nuevo, año 2000).
- El algoritmo estándar es el algoritmo de Hartigan-Wong (1979), que define la variación total dentro del grupo como la suma de las distancias al cuadrado de las distancias euclidianas entre los elementos y el centroide correspondiente.

K-means clustering

Es decir, definimos:

$$W(C_k) = \sum_{x_i \in C_k} (x_i - \mu_k)^2$$

donde:

- x_i es una observación de los datos perteneciente al cluster C_k .
- μ_k es la media de los puntos asignados al cluster C_k

Cada observación (x_i) es asignada a un grupo dado de tal manera que la distancia de la suma de cuadrados (SS) de la observación a sus cluster asignados se minimice (μ_k) se minimiza.

K-means clustering

Definimos *total within-cluster variation* como sigue:

$$tot.withiness = \sum_{k=1}^K W(C_k) = \sum_{k=1}^K \sum_{x_i \in C_k} (x_i - \mu_k)^2$$

La *total within-cluster sum of square* mide la compacidad (es decir, la bondad de ajuste) del clustering y queremos que sea lo más pequeño posible.

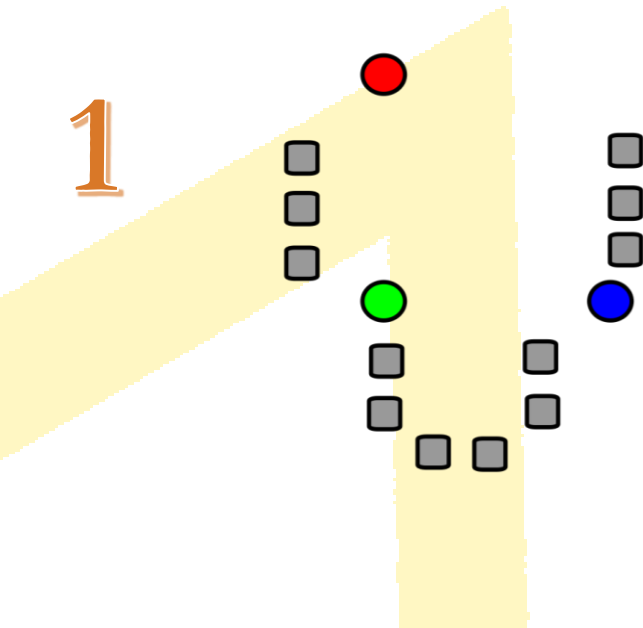
K-means clustering

El algoritmo de K-means se puede resumir de la siguiente manera:

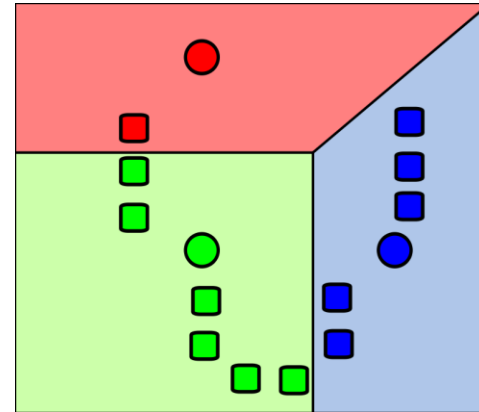
1. Especifique el número de clustering (K) que se crearán (por el Data Scientist)
2. Selecciona aleatoriamente k objetos del conjunto de datos como centros del clustering o medias.
3. Asigna cada observación a su centroide más cercano, basándose en la distancia euclídea entre el objeto y el centroide.
4. Para cada uno de los clusters k, actualiza el centroide del cluster calculando los nuevos valores medios de todos los puntos de datos que se encuentran en el cluster.
5. Minimiza iterativamente el *tot.withiness*. Es decir, repite los pasos 3 y 4 hasta que las asignaciones de clúster dejen de cambiar o se alcance el número máximo de iteraciones.
6. De forma predeterminada, R utiliza 10 como valor predeterminado para el número máximo de iteraciones (suele ser poco).

K-means clustering

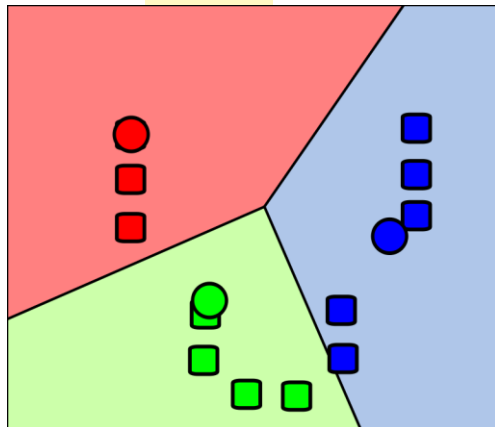
1



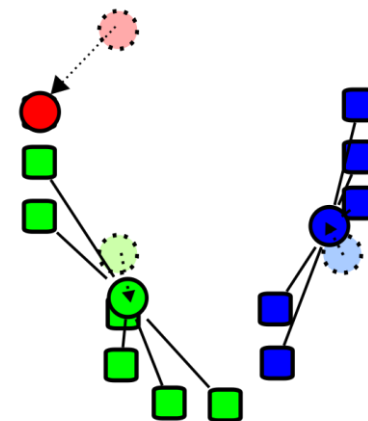
2



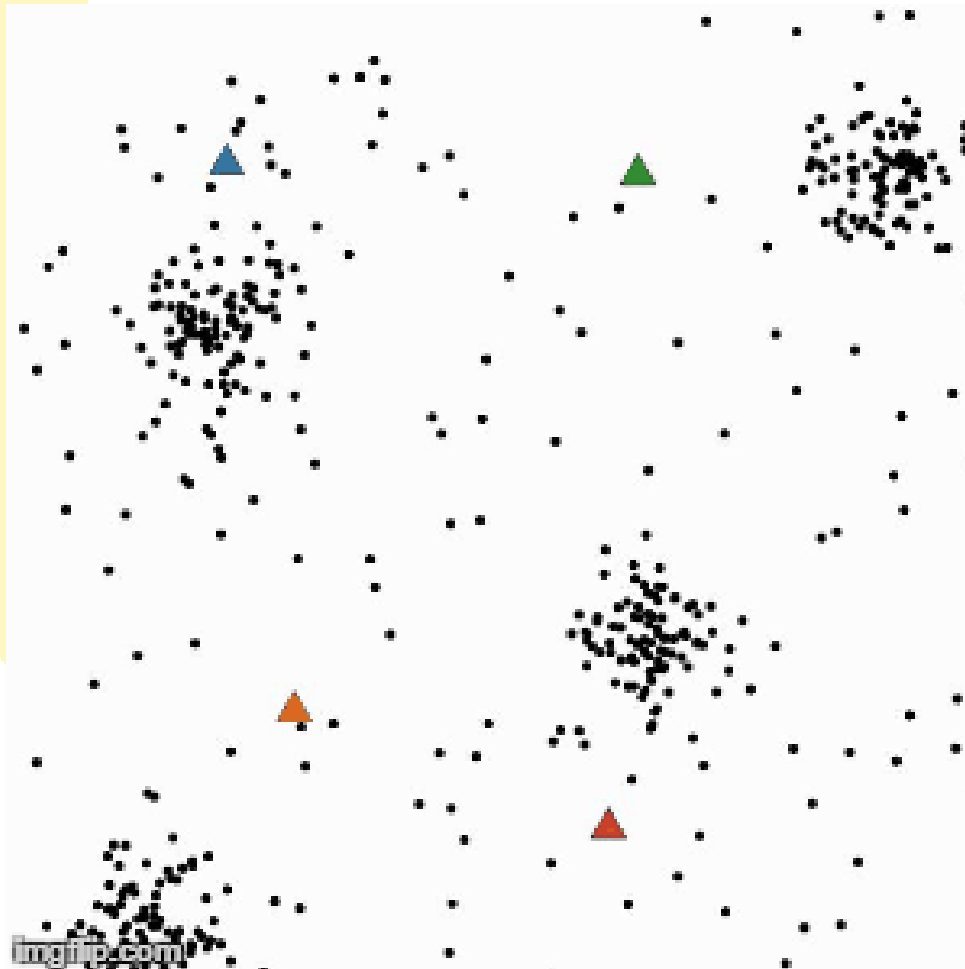
4



3



K-means clustering



Let's do it...



Determinar el número óptimo de clusters

Recordemos que es el Data Scientist quien especifica la cantidad de clusters que asignará el algoritmo. Sin embargo, es conveniente hallar el número óptimo de clusters. Tenemos tres métodos clásicos para determinar el número óptimo de clusters:

- Elbow method
- Silhouette method
- Gap statistic

Elbow Method

El *total within-cluster sum of square* (wss) cuantifica qué tan compacto es el clustering y queremos hacerlo tan pequeño como sea posible. Por tanto Podemos seguir el siguiente algoritmo para definir el número óptimo de clusters:

1. Calcula el algoritmo de clustering (por ejemplo, k-means clustering) para diferentes valores de k.
2. Para cada k, calcula el *total within-cluster sum of square* (wss)
3. Realiza un plot de la curva wss respect al número de clusters k.
4. La ubicación de una pequeña curva (elbow) en el plot, es generalmente considerada como un indicador apropiado del número de clusters.

Average Silhouette Method

- En resumen, el *average silhouette method* se enfoca en medir la calidad de un clustering. Esto es, determina qué tan bien se encuentra cada objeto dentro de su grupo.
- Un gran average silhouette indica un buen clustering
- El *average silhouette method* calcula la silueta promedio de observaciones para diferentes valores de k.
- El número óptimo de clusters k es el que maximiza el average silhouette sobre un rango de posibles valores.
- Se define silhouette como:

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

donde $a(i)$, $b(i)$ pueden verse en

[https://en.wikipedia.org/wiki/Silhouette \(clustering\)](https://en.wikipedia.org/wiki/Silhouette_(clustering))

Gap Statistic Method

- El Gap Statistic Method ha sido publicado por R. Tibshirani, G. Walther y T. Hastie (Stanford University, 2001). Podría aplicarse a otros métodos de clustering.
- Involucra simulaciones de Monte Carlo.
- Realiza Bootstrapping para generar B copias del *dataset de referencia*.
- Es el más fundamentado matemáticamente.

Gap Statistic Method

El algoritmo se resume en los siguientes pasos:

- Se realiza el clustering, variando el número de clusters $k = 1, \dots, k_{max}$ y calculando el correspondiente W_k .
- Genera B data sets, via bootstrapping y agrupa (cluster) cada uno de ellos variando el número de clusters $k = 1, \dots, k_{max}$. Calcula el estimated gap statistics mediante la ecuación

$$Gap_n(k) = E_n^* \log(W_k) - \log(W_k)$$

donde E_n^* denota la esperanza bajo un tamaño de muestra n .

- Sea $\bar{w} = \frac{1}{B} \sum_b \log(W_{kb}^*)$, calcula la desviación estándar $sd(k) = \sqrt{\frac{1}{B} \sum_b (\log(W_{kb}^*) - \bar{w})^2}$ y definamos $s_k = sd_k \times \sqrt{1 + \frac{1}{B}}$.
- Elige el total de clusters como el menor k tal que

$$Gap(k) \geq Gap(k+1) - s_{k+1}$$

Let's do it...

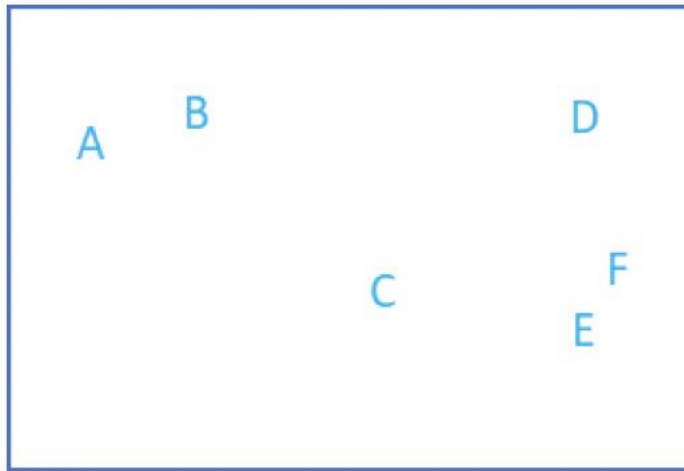


Referencias

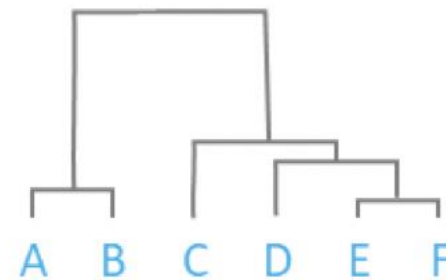
(Las referencias no están escritas formalmente)

- Primer k-means:
http://www.labri.fr/perso/bpinaud/userfiles/downloads/hartigan_1979_kmeans.pdf.
- K*-means: https://www.researchgate.net/publication/222418782_K-Means_A_new_generalized_k-means_clustering_algorithm.
- Gap Statistics: <http://web.stanford.edu/~hastie/Papers/gap.pdf>.
- Libro clustering:
<https://onlinelibrary.wiley.com/doi/book/10.1002/9780470316801>.
- X-means:
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.19.3377&rep=rep1&type=pdf>.

Hierarchical Clustering, un árbol, que no es un árbol

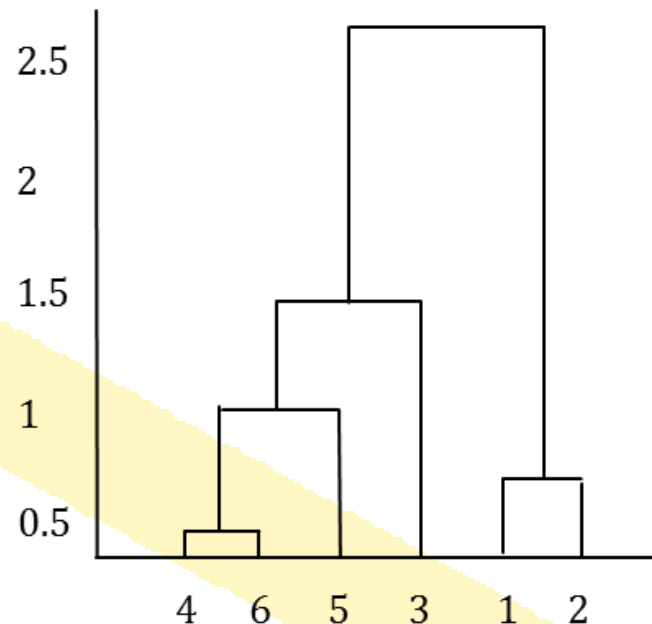


Dendrogram



Dendograma

- En Hierarchical Clustering, se clasifican los objetos en una jerarquía similar a un diagrama en forma de árbol que se denomina dendograma. La distancia de división o unión (Split o merge) (llamada Height o altura) se muestra en el eje y del dendograma. Ejemplo:



Dendograma

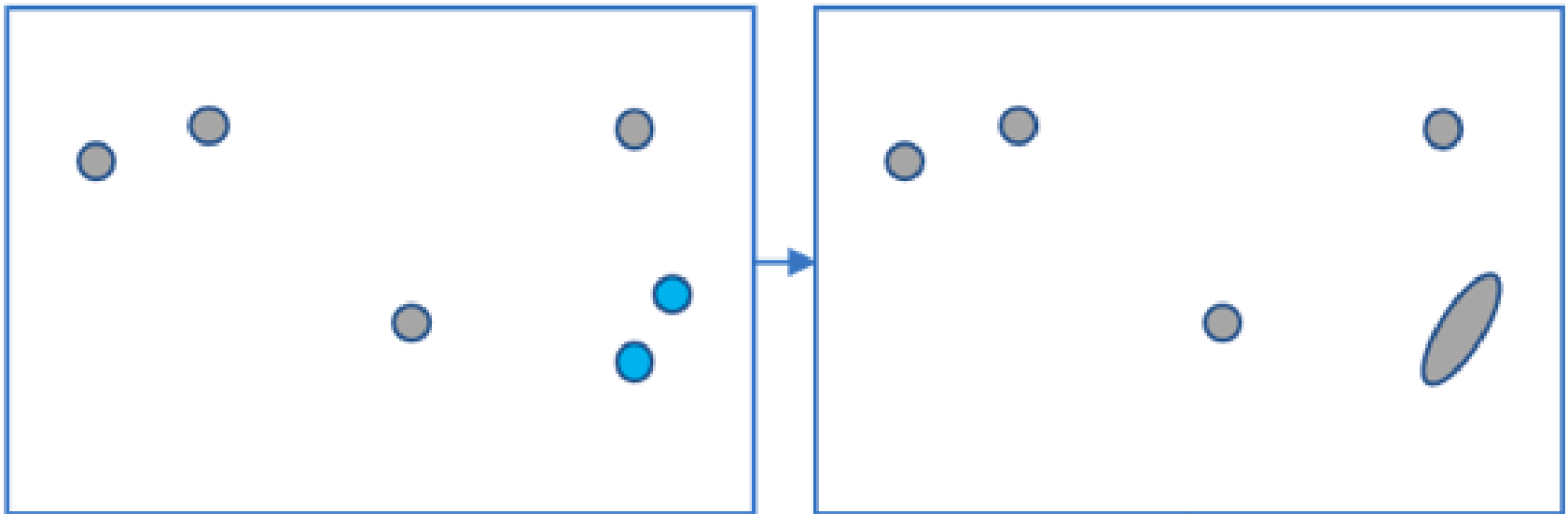
- En la figura anterior, 4 y 6 se encuentran en un mismo cluster, digamos, cluster A, ya que eran los más cercanos en distancia seguidos por los puntos 1 y 2, digamos cluster B.
- Después de eso, 5 se unió en el mismo cluster A seguido de 3 resultando el dendograma final en dos clusters.
- Por último, los dos clusters se unen en uno solo y aquí es donde se detiene el proceso de clustering.
- Una pregunta que podríamos hacernos ahora es ¿cómo decidir cuándo dejar de unir los grupos?

Hierarchical clustering

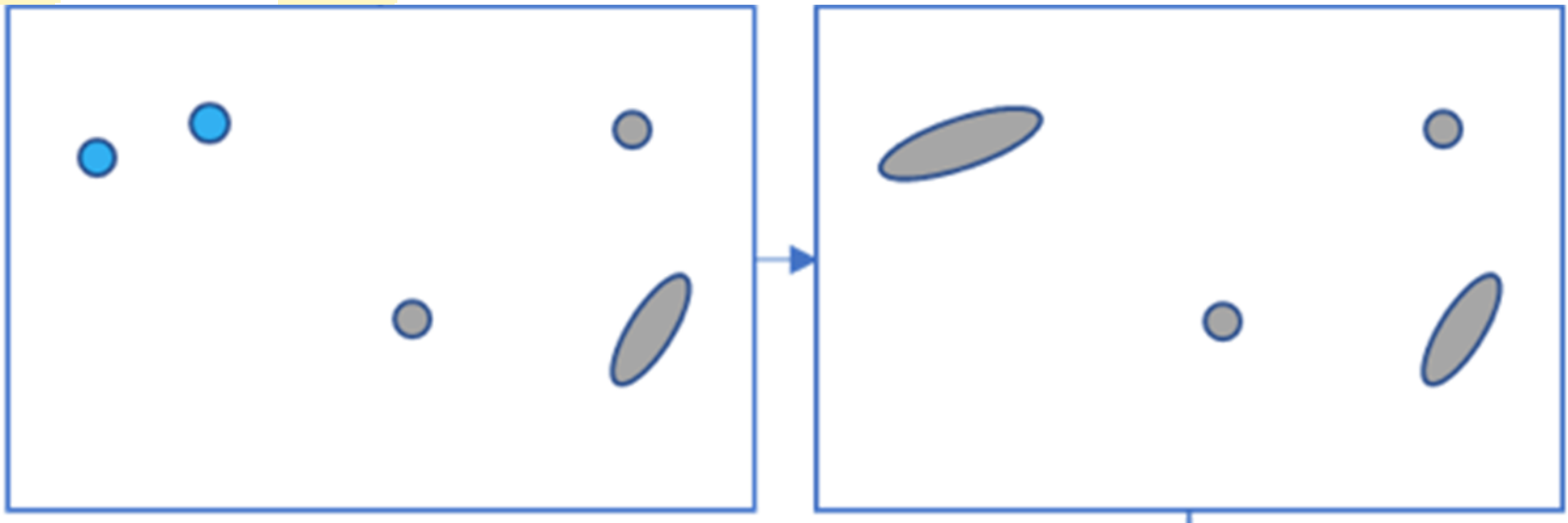
En resumen, el algoritmo de hierarchical clustering es el siguiente:

- Comienza por calcular la distancia entre cada par de puntos de observación y almacenarla en una matriz de distancia.
- Posteriormente pone cada punto en su propio grupo.
- Luego comienza a unir los pares de puntos más cercanos en función de las distancias basándose en la matriz de distancia y, como resultado, la cantidad de agrupaciones se reduce a 1.
- Vuelve a calcular la distancia entre el nuevo cluster y los antiguos y los almacena en una nueva matriz de distancia.
- Por último, repite los pasos 2 y 3 hasta que todos los clusters se fusionan en uno solo.

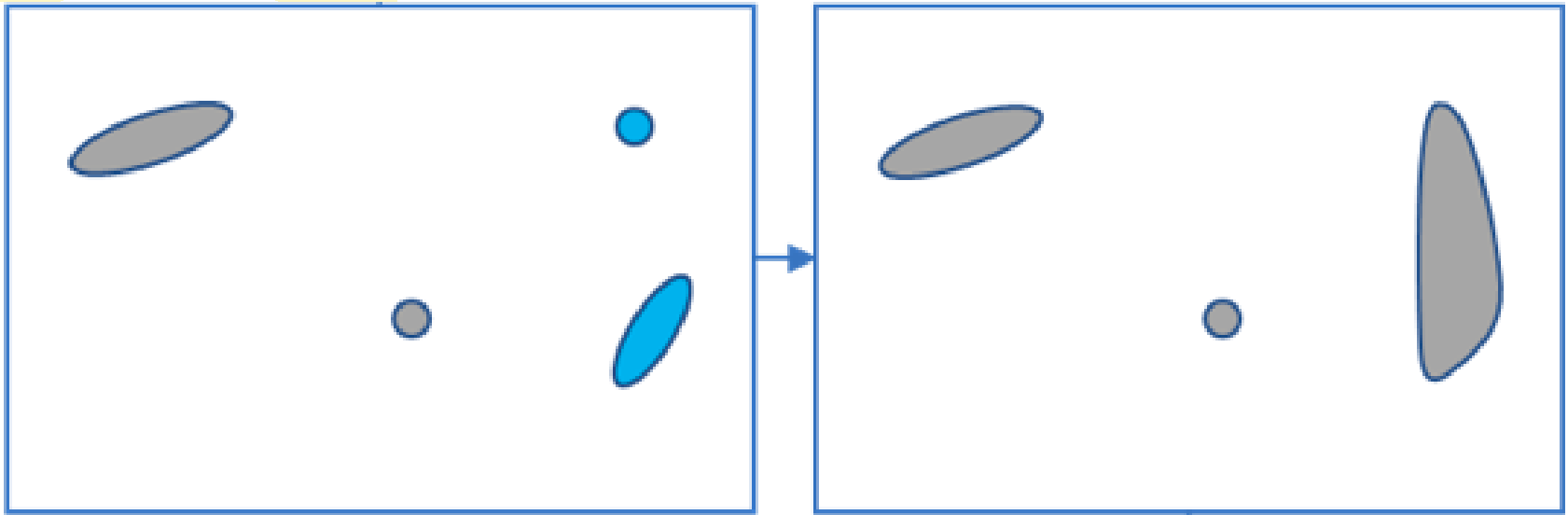
Hierarchical clustering



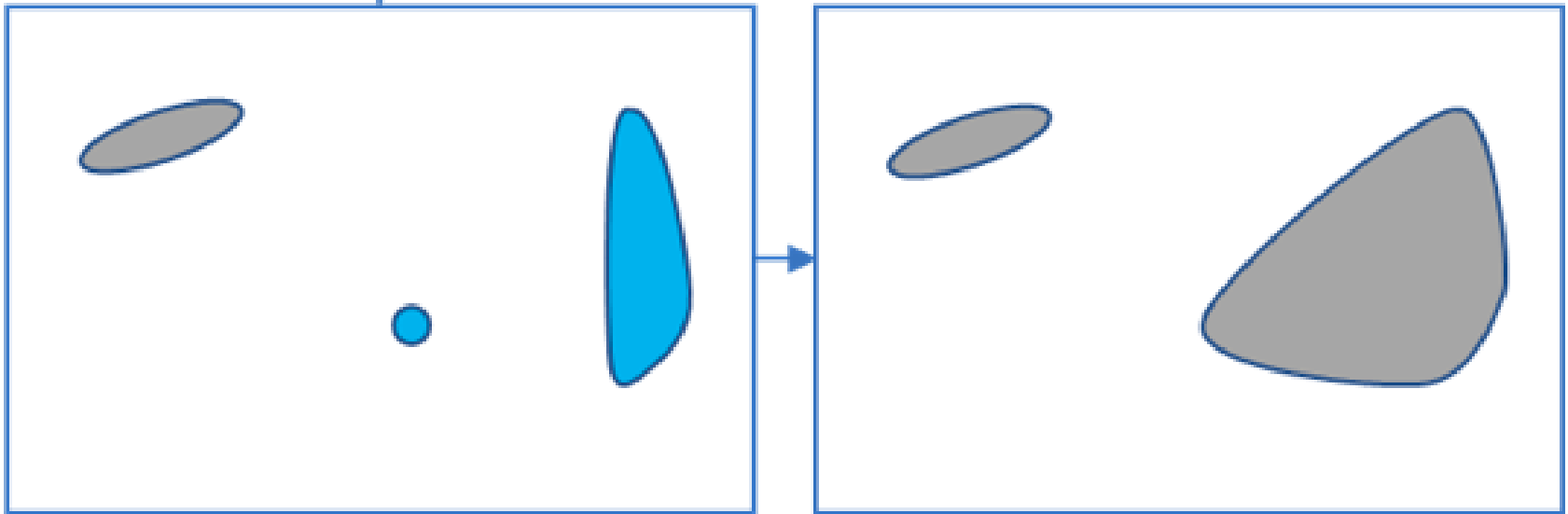
Hierarchical clustering



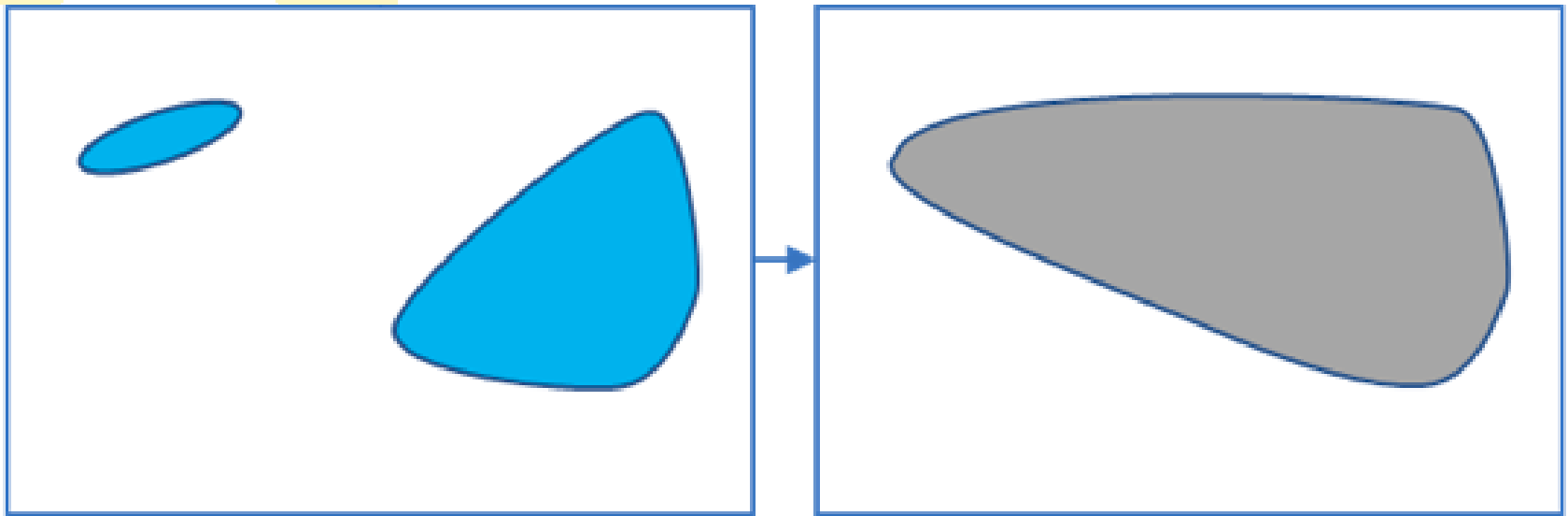
Hierarchical clustering



Hierarchical clustering

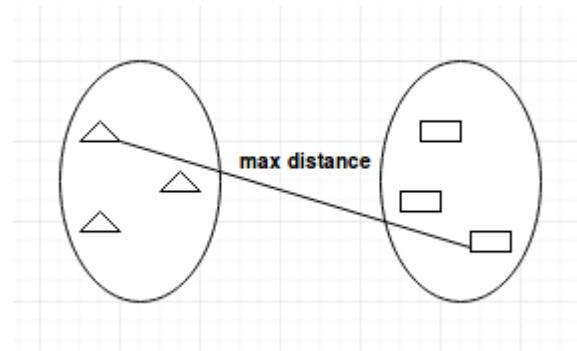


Hierarchical clustering

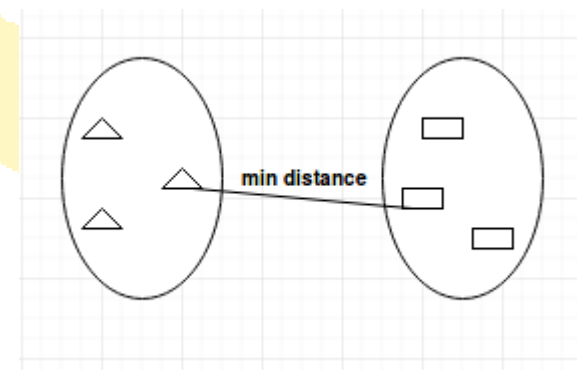


Linkage Methods

Complete-linkage: calcula la distancia máxima entre agrupaciones antes de unir.

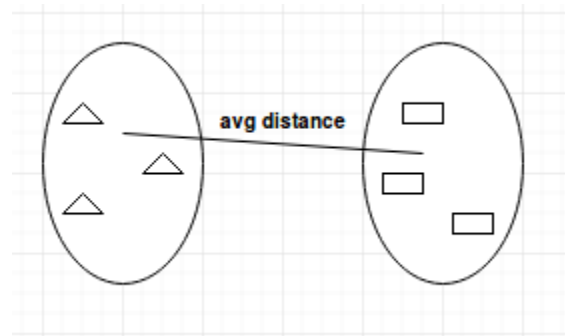


Single-linkage: calcula la distancia mínima entre los cluster antes de unirse. Este linkage se puede usar para detectar outliers, ya que se unirán al final.

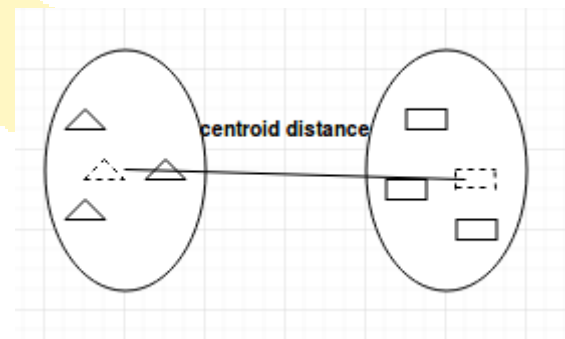


Linkage Methods

Average-linkage: calcula la distancia promedio entre clusters antes de unirlos.



Centroid-linkage: encuentra el centroide del cluster 1 y el centroide del cluster 2, y luego calcula la distancia entre los dos antes de unirse.



Bondad de ajuste del cluster

- La parte más importante en cualquier proyecto donde se aplique de unsupervised learning es el análisis de los resultados. Después de haber realizado el clustering utilizando cualquier algoritmo y cualquier conjunto de parámetros, debemos asegurarnos que lo hicimos correctamente. ¿Pero cómo lo determinamos?
- Existen muchas medidas para esto, pero quizás la más popular sea el Índice de Dunn. El índice de Dunn es la relación entre las distancias mínimas entre clusters y el diámetro máximo entre clusters. El diámetro de un grupo es la distancia entre sus dos puntos más alejados. Para tener clusters bien separados y a la vez compactos, debemos aspirar a un índice de Dunn alto.

Let's do it...

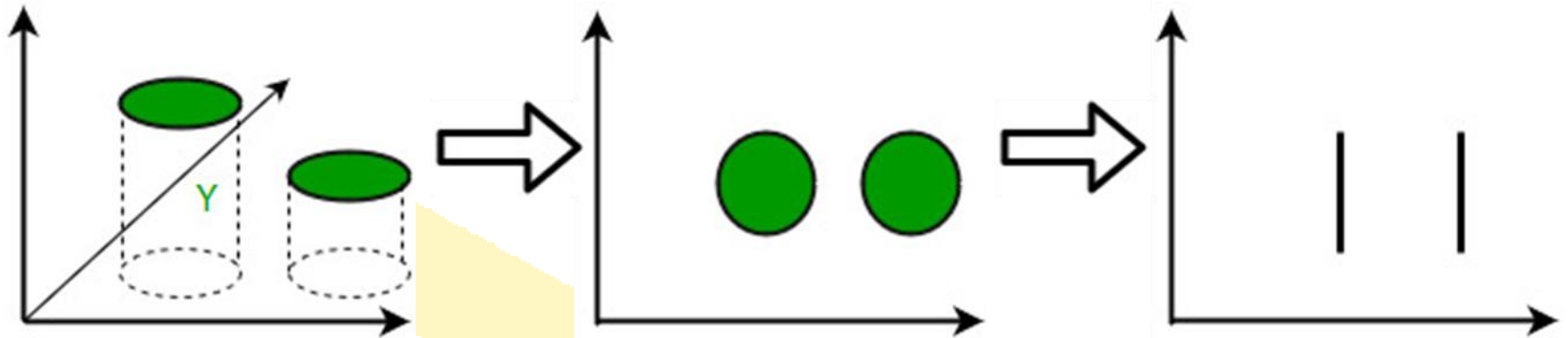


Referencias

(Las referencias no están escritas formalmente)

- Libro clustering:
<https://onlinelibrary.wiley.com/doi/book/10.1002/9780470316801>.
- Lecture notes interesantes:
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.19.3377&rep=rep1&type=pdf>.
- Matemáticas para clustering:
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.531.5291&rep=rep1&type=pdf>

Dimensionality reduction, ¿en cuántas dimensiones podemos visualizar datos?



Dimensionality reduction Motivación

- Reducir nuestra la dimensión de nuestro conjunto de datos.
- Supongamos que nuestro dataset tiene cien columnas (es decir, características) o podría ser una matriz de puntos que conforman una esfera en el espacio tridimensional.
- Dimensionality reduction consiste en reducir el número de columnas por ejemplo, veinte o llevar la esfera a un espacio bidimensional.
- Eso está muy bien, pero ¿por qué nos debe importar? ¿Por qué deberíamos eliminar 80 columnas de nuestro conjunto de datos cuando podríamos incorporarlo directamente a nuestro algoritmo de aprendizaje automático y dejar que se encargue del resto?

La maldición de la dimensionalidad

The curse of dimensionality

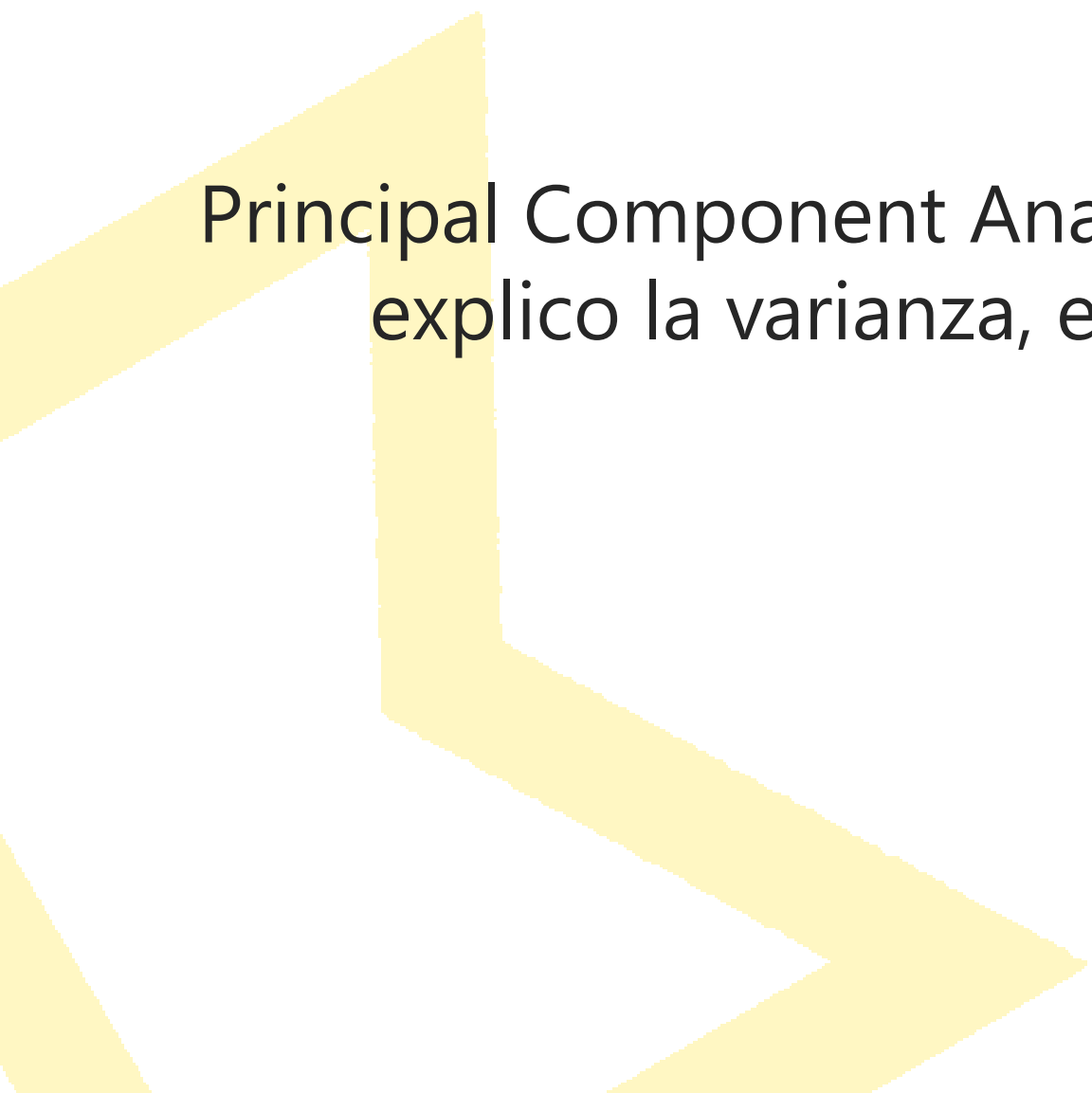
- La maldición de la dimensionalidad se refiere a todos los problemas que surgen cuando se trabaja con datos en las dimensiones superiores, que no existían en las dimensiones inferiores.
- A medida que aumenta el número de variables, el número de observaciones también aumenta proporcionalmente.
- Cuantas más variables tengamos, mayor número de observaciones necesitaremos para tener para que todas las combinaciones de valores de variables estén bien representadas en nuestra muestra.

Dimensionality reduction Motivación

- A medida que aumenta el número de variables, el modelo se vuelve más complejo.
- Cuanto mayor sea el número de variables, más posibilidades hay de que tengamos overfitting.
- Un modelo de machine learning que está entrenado con un gran número de variables, depende cada vez más de los datos con los que fue entrenado.
- Menos variables en nuestro conjunto de entrenamiento, implican menos suposiciones que nuestro modelo tendrá que hacer.

Dimensionality reduction ventajas

- Menos datos “engañosos” se traduce en una mejora en la precisión del modelo.
- Menos dimensiones significan menos coste computacional, que se traduce en un algoritmo que se entrenará más rápido.
- Menos dimensiones permiten el uso de algoritmos no aptos para un gran número de dimensiones.
- Elimina las variables redundantes y ruido.

A large, abstract yellow geometric shape, resembling a stylized 'L' or a corner, is positioned on the left side of the slide, extending from the top left towards the bottom right.

Principal Component Analysis (PCA): si no explico la varianza, estoy fuera...

PCA

- Se tienen p variables X_1, X_2, \dots, X_p sobre una muestra de n observaciones. Nuestra matriz viene dada por:

$$X = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix}$$

PCA

- Problema: ¿Podemos describir la “información” contenida en estos datos mediante algún conjunto de variables menor que el de variables originales?
- Idea: Si una variable es función de otras, contiene información redundante.
- Por tanto, si las p variables observadas están fuertemente correlacionadas, será posible sustituirlas por menos variables sin gran pérdida de “información”.

PCA

Sean $X = [X_1, \dots, X_p]$ y $S = \text{Cov}(X)$ su matriz de covarianzas.

Como $S \geq 0$ y simétrica, su descomposición espectral es:

$$S = T\Lambda T'.$$

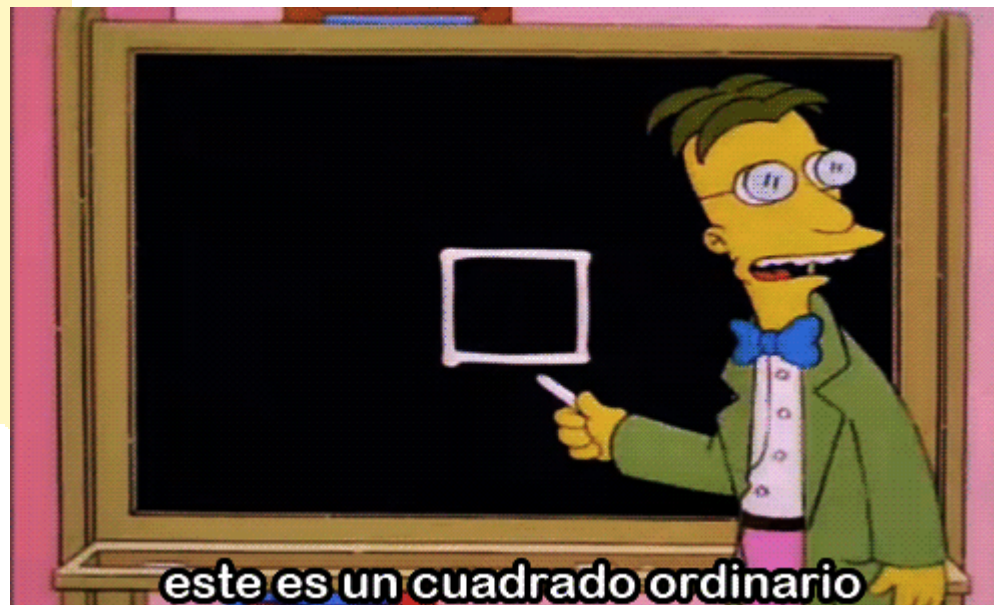
Las **componentes principales** de X son las nuevas variables

$$Y_j = Xt_j, \quad j = 1, \dots, p.$$

Donde t_j es el j -ésimo autovector de S .

PCA: matemáticas.

¡¡AL PIZARRÓN!!



PCA: Ejemplo “a mano”

La Tabla siguiente contiene información sobre chalets construidos por diez constructoras

Promotora	X1=Duración		X2=Precio medio
	media hipoteca		
1	87		3
2	143		9
3	189		18
4	190		8
5	205		9
6	147		11
7	188		25
8	373		27
9	126		13
10	257		34

PCA: Ejemplo "a mano"

Vector de medias y matriz de covarianzas

$$\bar{x} = (19.05 \quad 1.57)' \quad S = \begin{bmatrix} 56.9685 & 5.1705 \\ ? & 0.89471 \end{bmatrix}$$

Autovalores y autovectores de S :

$$\Lambda = \text{diag}(57.4413 \quad 0.4213), T = \begin{bmatrix} 0.9958 & -0.0911 \\ 0.0911 & 0.9958 \end{bmatrix}$$

Por tanto, las CP vienen dadas por:

$$Y_1 = 0.9958 X_1 + 0.0911 X_2, Y_2 = -0.0911 X_1 + 0.9958 X_2$$

Y los porcentajes de variabilidad explicados por cada componente son:

$$\frac{57.4413}{57.8626} \cdot 100 = 99.27\%, \quad \frac{0.4213}{57.8626} \cdot 100 = 0.73\%$$

PCA: más matemáticas.

¡¡AL PIZARRÓN!!

Debo reducir la dimensionalidad de mi dataset
Debo reducir la dimensionalidad de mi dataset
Debo reducir la dimensionalidad de mi dataset
Debo reducir la dimensionalidad de mi dataset
Debo reducir la dimensionalidad de mi dataset
Debo reducir la dimensionalidad de mi dataset
Debo reducir la dimensionalidad de mi dataset
Debo reducir la dimensionalidad de mi dataset



Let's do it...



PCA: Desventajas

Asume linealidad en los datos, esto es, no es fuerte cuando es aplicado a datos no lineales.

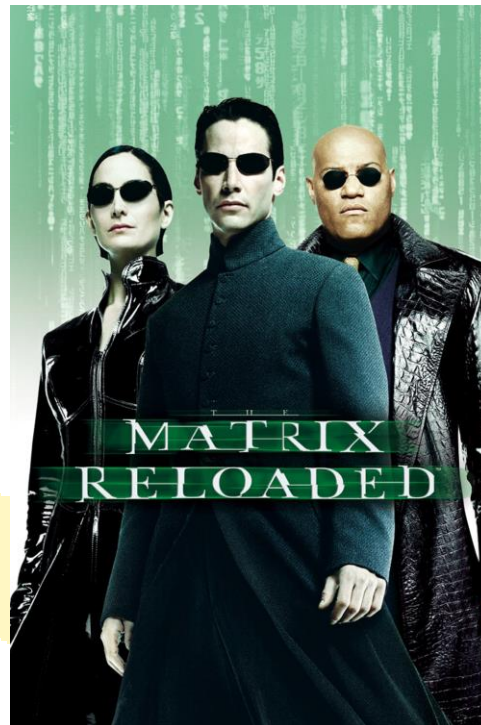
Por eso, existen otros métodos mucho más fuertes, que pueden ser aplicados a casi todo tipo de datos.

Referencias

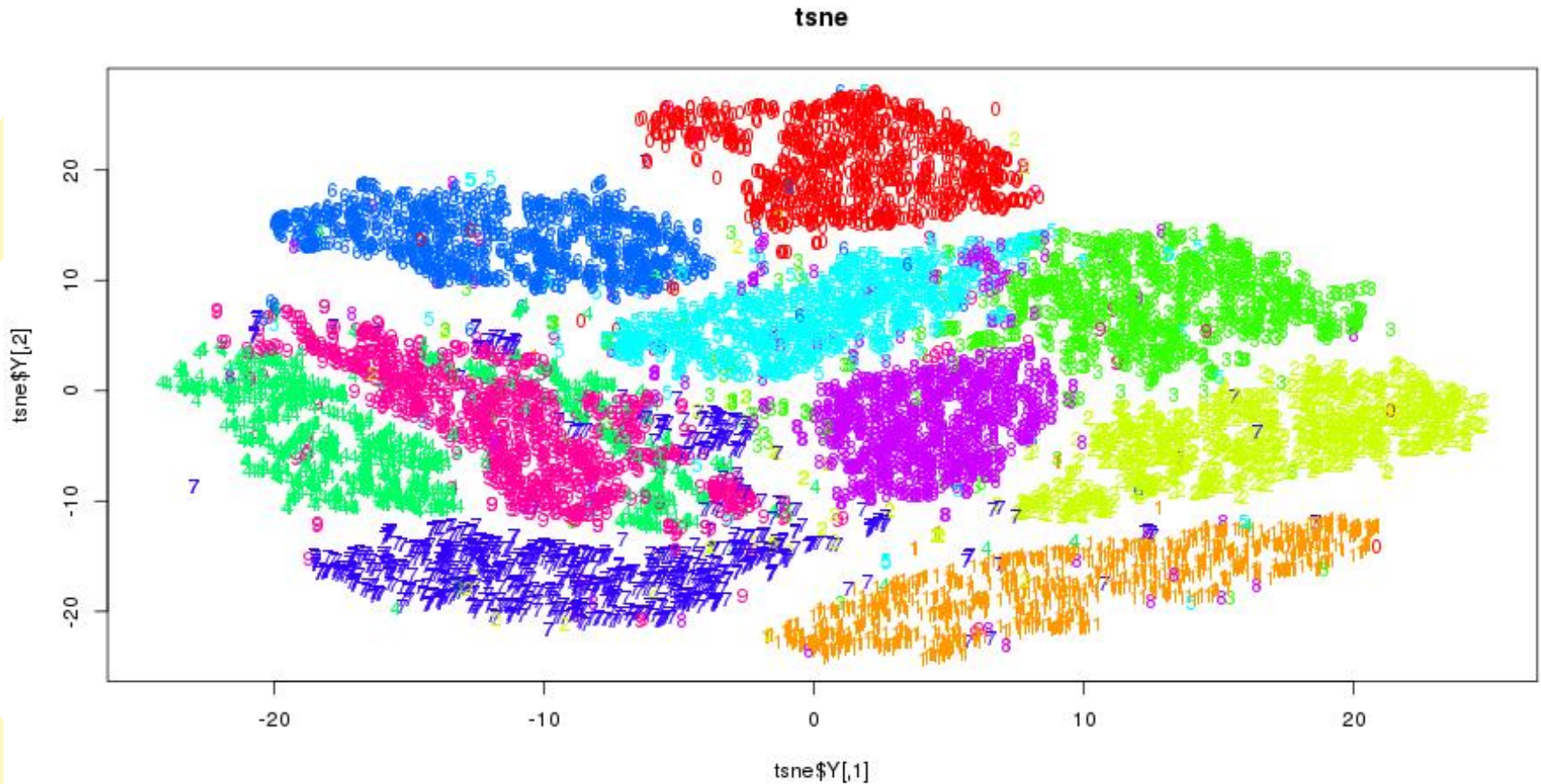
(Las referencias no están escritas formalmente)

- Libro exclusivo de PCA:
[http://cda.psych.uiuc.edu/statistical_learning_course/Jolliffe%20I.%20Principal%20Component%20Analysis%20\(2ed.,%20Springer,%202002\)\(518s\)_MVsa.pdf](http://cda.psych.uiuc.edu/statistical_learning_course/Jolliffe%20I.%20Principal%20Component%20Analysis%20(2ed.,%20Springer,%202002)(518s)_MVsa.pdf)

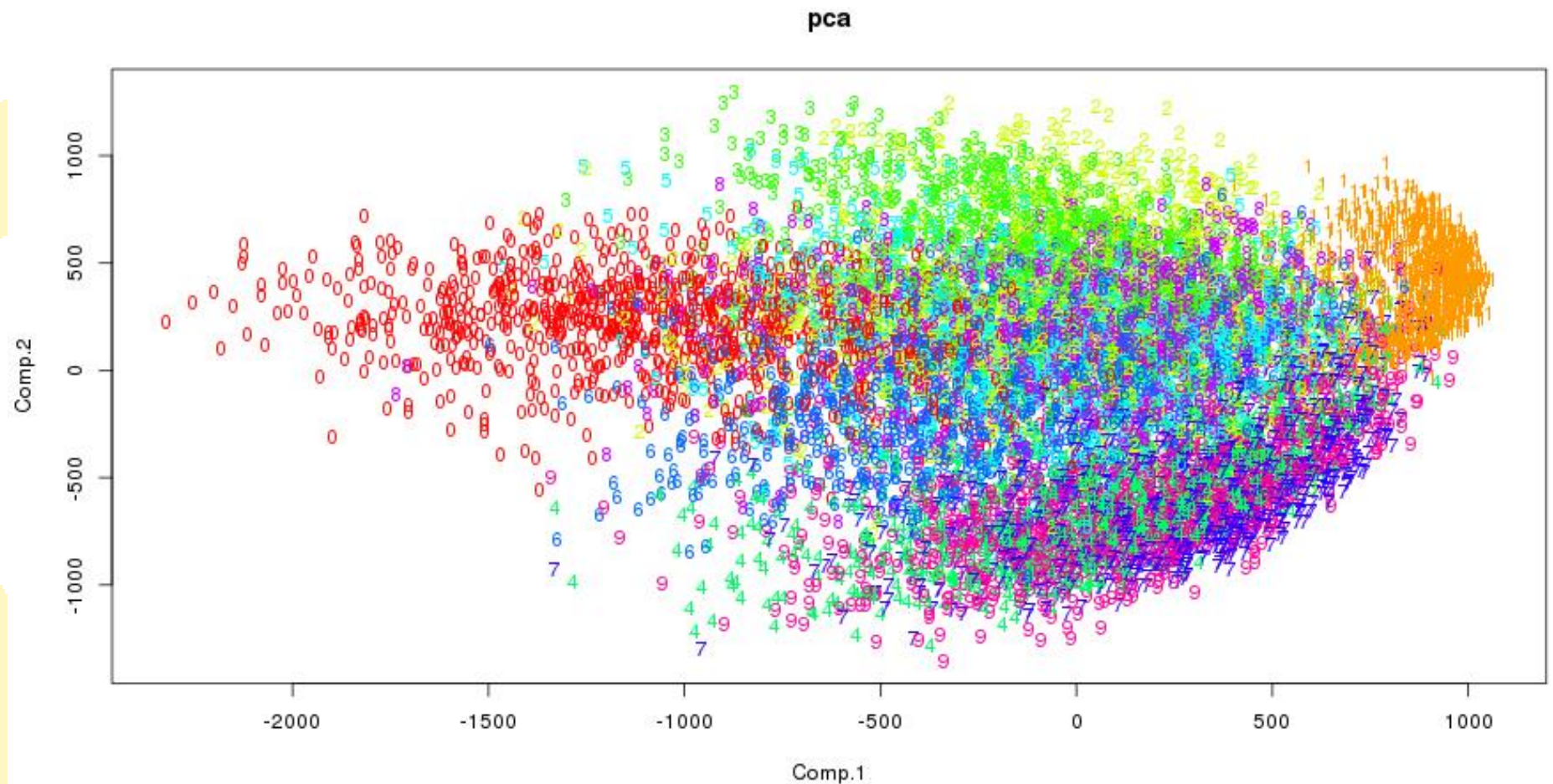
t-distributed Stochastic Neighbor Embedding: dimensionality reduction reloaded...



t-SNE vs- PCA



t-SNE vs- PCA



t-SNE

- Al igual que el PCA, es una técnica de proyección, o reducción de dimensión, que se aplica con la intención de visualizar N variables en 2.
- Uno de los resultados de t-SNE es una matriz de dos dimensiones (puede ser de tres también), donde cada observación (fila) representa un dato de entrada. Luego podemos aplicar un clustering y agrupar los casos según su distancia en este nuevo mapa de 2 dimensiones.
- t-SNE mapea los datos multidimensionales a un espacio dimensional inferior.
- Después de este proceso, las variables de entrada ya no son identificables, y no puede hacerse ninguna inferencia basada únicamente en la salida del t-SNE. Por lo tanto, es principalmente una técnica de exploración y visualización de datos.

t-SNE: overview

- Comienza calculando la probabilidad de similitud de puntos en el espacio de multidimensional y calculando la probabilidad de similitud de puntos en el espacio de baja dimensión (puede ser 2 o 3). La similitud de los puntos se calcula como la probabilidad condicional de que un punto A elegiría el punto B como su vecino si los vecinos se seleccionaran en proporción a su densidad de probabilidad bajo una distribución Gaussiana (distribución normal) centrada en A.
- Luego trata de minimizar la diferencia entre estas probabilidades condicionales (o similitudes) en el espacio de dimensión superior y de dimensión inferior para una representación perfecta de los datos de entrada de dimensión inferior.
- Para medir la minimización de la suma de la diferencia de probabilidad condicional, t-SNE minimiza la suma de la divergencia de Kullback-Leibler, para ello, utiliza gradient descent.

Referencias

(Las referencias no están escritas formalmente)

- Imagen t-SNE vs PCA: <https://www.kaggle.com/puyokw/clustering-in-2-dimension-using-tsne/data>
- Página oficial t-SNE: <https://lvdmaaten.github.io/tsne/>
- Artículo de t-SNE en la revista JMLR:
<http://www.jmlr.org/papers/volume9/vandermaaten08a/vandermaaten08a.pdf>.
- ¿cómo usar t-SNE de manera efectiva?:
<https://distill.pub/2016/misread-tsne/>

Let's do it...



Fin...

