Elementos de Machine Learning

Diplomado de Análisis Computacional Estadístico de Datos con Python

Impartido por: Ronaldo Canizales

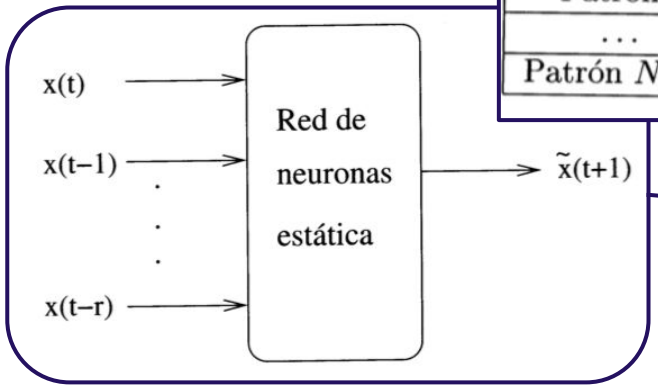
Bloque A

Otras aplicaciones: Series temporales

Predicción de series temporales: un paso en el futuro

$$\tilde{x}(t+1) = \tilde{F}\big(x(t), x(t-1), ..., x(t-r)\big)$$

	Entrada	Salida deseada
Patrón 1	$x(r), x(r-1), ..., x(1), x(0)$	$x(r+1)$
Patrón 2	$x(r+1), x(r), ..., x(2), x(1)$	$x(r+2)$
Patrón 3	$x(r+2), x(r+1), ..., x(3), x(2)$	$x(r+3)$
Patrón 4	$x(r+3), x(r+2), ..., x(4), x(3)$	$x(r+4)$
...	...	...
Patrón $N-r$	$x(N-1), x(N-2), ..., x(N-r), x(N-(r+1))$	$x(N)$



Time step	Value
1	10
2	11
3	18
4	15
5	20

X	Y
?	10
10	11
11	18
18	15
15	20
20	?

Con r = 1

Individual Household Electric Power Consumption

Donated on 8/29/2012

Measurements of electric power consumption in one household with a one-minute sampling rate over a period of almost 4 years. Different electrical quantities and some sub-metering values are available.

Dataset Characteristics	Subject Area	Associated Tasks
Multivariate, Time-Series	Physics and Chemistry	Regression, Clustering

Feature Type	# Instances	# Features
Real	2075259	9

Dataset Information

Additional Information

This archive contains 2075259 measurements gathered in a house located in Sceaux (7km of Paris, France) between December 2006 and November 2010 (47 months).

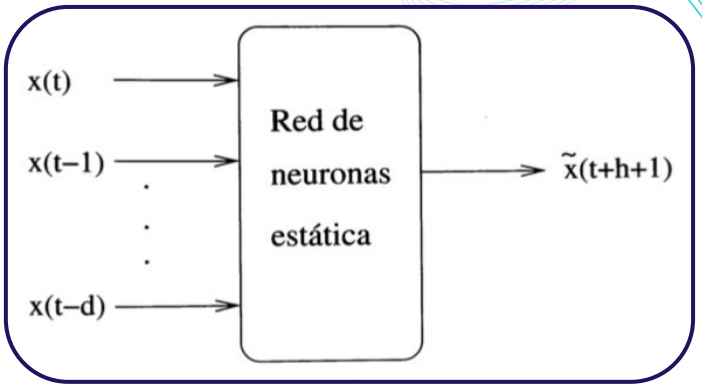
Notes:

1.(global_active_power*1000/60 - sub_metering_1 - sub_metering_2 - sub_metering_3) represents the active energy consumed every minute (in watt hour) in the household by electrical equipment not measured in sub-meterings 1, 2 and 3.

2.The dataset contains some missing values in the measurements (nearly 1,25% of the rows). All calendar timestamps are present in the dataset but for some timestamps, the measurement values are missing: a missing value is represented by the

Enlace

Predicción de series temporales: múltiples pasos en el futuro



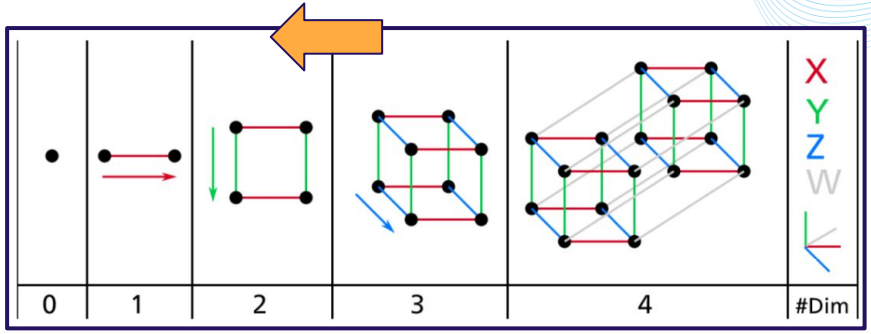
$$\tilde{x}(t+h+1) = \tilde{G}\big(x(t), x(t-1), ..., x(t-d)\big)$$

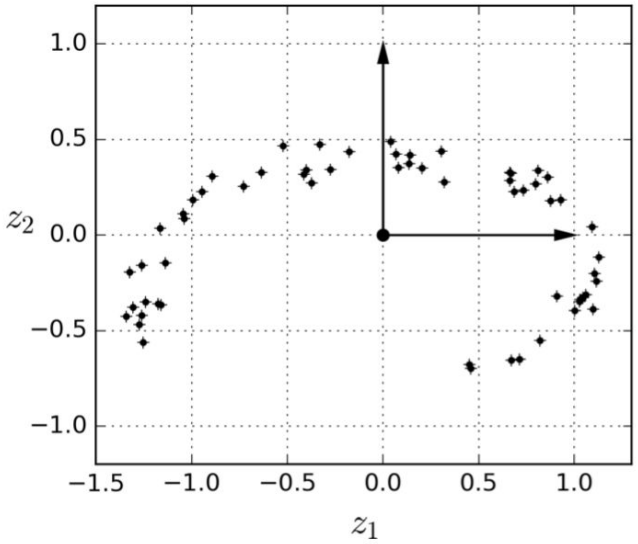
	Entrada	Salida deseada
Patrón 1	$x(d), x(d-1), ..., x(1), x(0)$	$x(d+h+1)$
Patrón 2	$x(d+1), x(d), ..., x(2), x(1)$	$x(d+h+2)$
Patrón 3	$x(d+2), x(d+1), ..., x(3), x(2)$	$x(d+h+3)$
Patrón 4	$x(d+3), x(d+2), ..., x(4), x(3)$	$x(d+h+4)$
...	...	...
Patrón N-d-h	$x(N-h-1), x(N-h-2), ..., x(N-1-h-d)$	$x(N)$

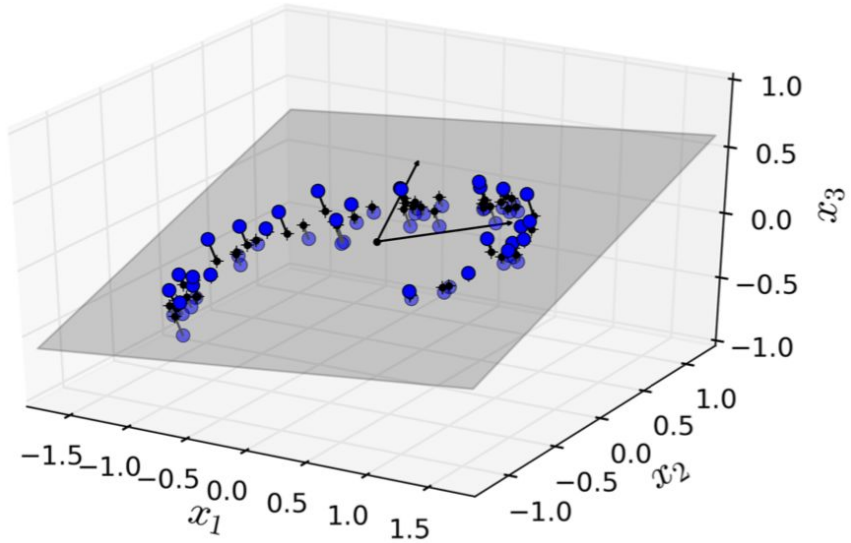
Bloque B

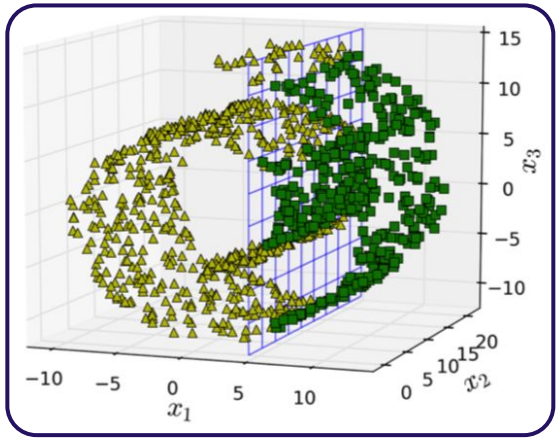
Reducción de la dimensionalidad: Intro al uso de PCA en ML

¿A qué nos referimos?



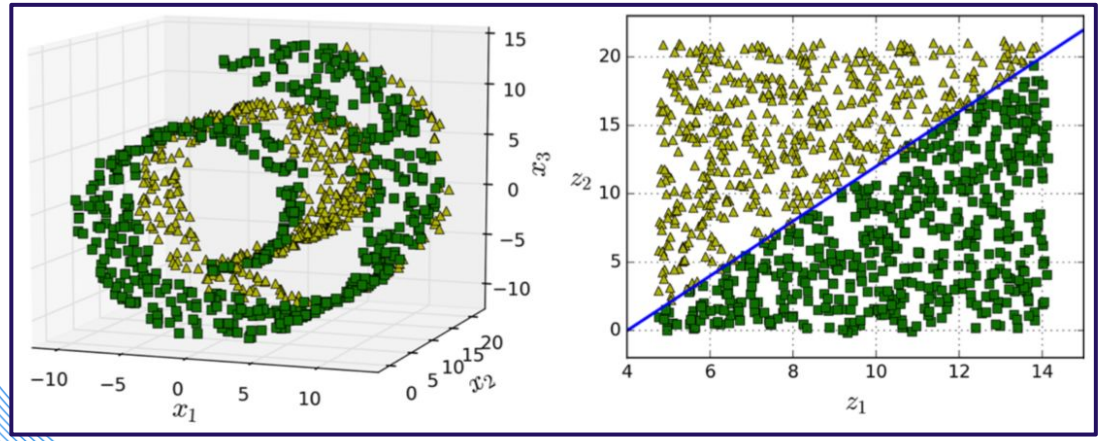






A veces no será necesario

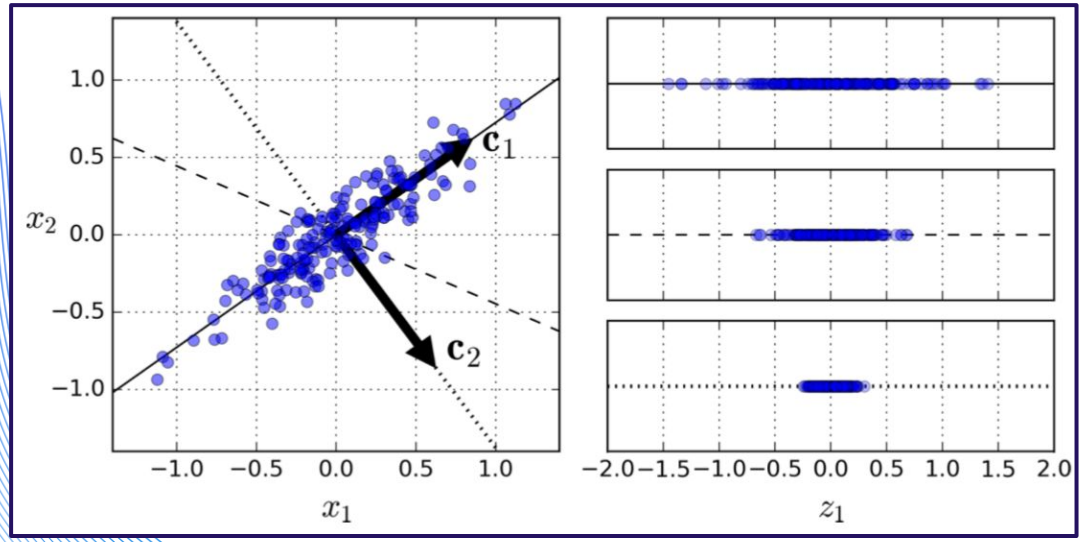
¿Por qué hacerlo en ML?

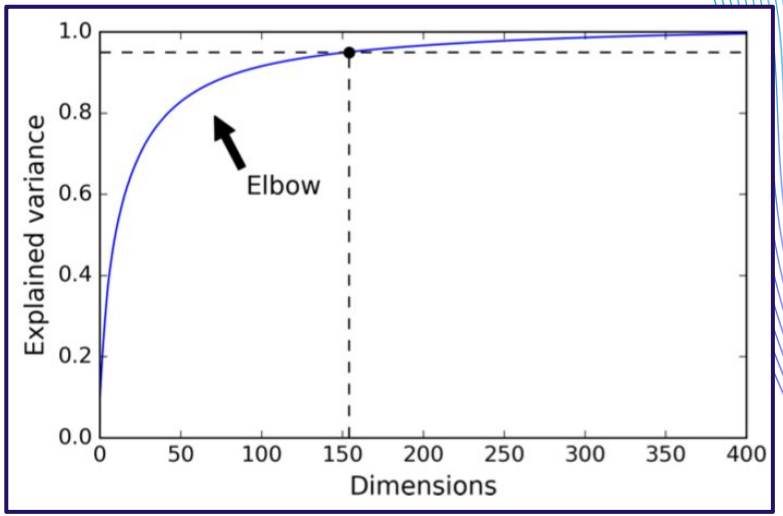


A veces nos ayudará para encontrar un modelo más preciso (o más sencillo).

No basta con reducir dimensiones

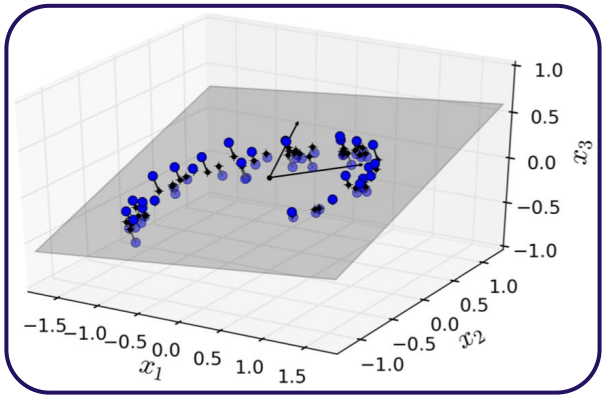
El algoritmo se encarga de encontrar la dimensión que retenga la mayor varianza entre los datos. Siempre se perderá información, pero se desea perder la menor cantidad posible.

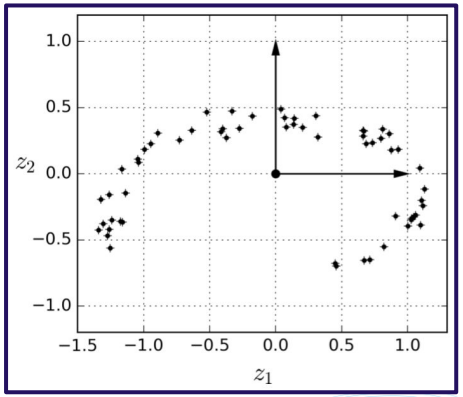




Se desea seleccionar la menor cantidad de dimensiones, sin perder mucha información.

Hiperparámetro: n_components






```python
from sklearn.decomposition import PCA

pca = PCA(n_components = 2)
X2D = pca.fit_transform(X)
```

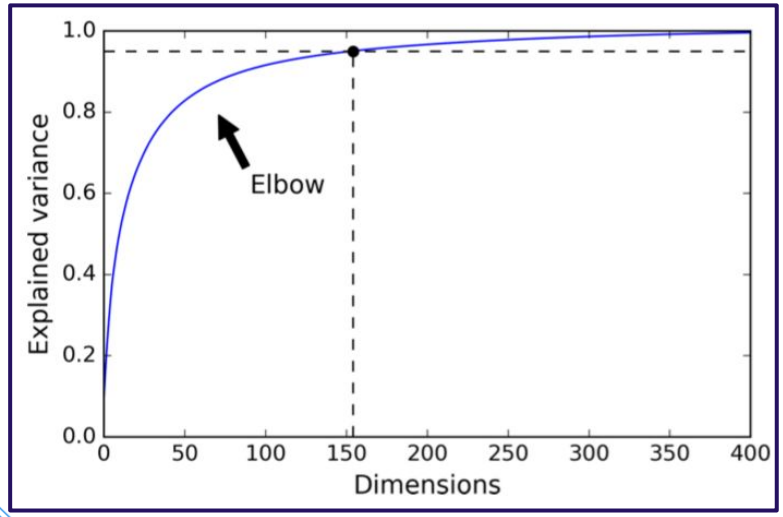
```
>>> print(pca.explained_variance_ratio_)
array([ 0.84248607,  0.14631839])
```

Para fijar una cantidad de dimensiones, hay que proveer un número natural mayor a uno

El 1.2% restante es el “precio a pagar” por buscar un modelo más sencillo de entrenar

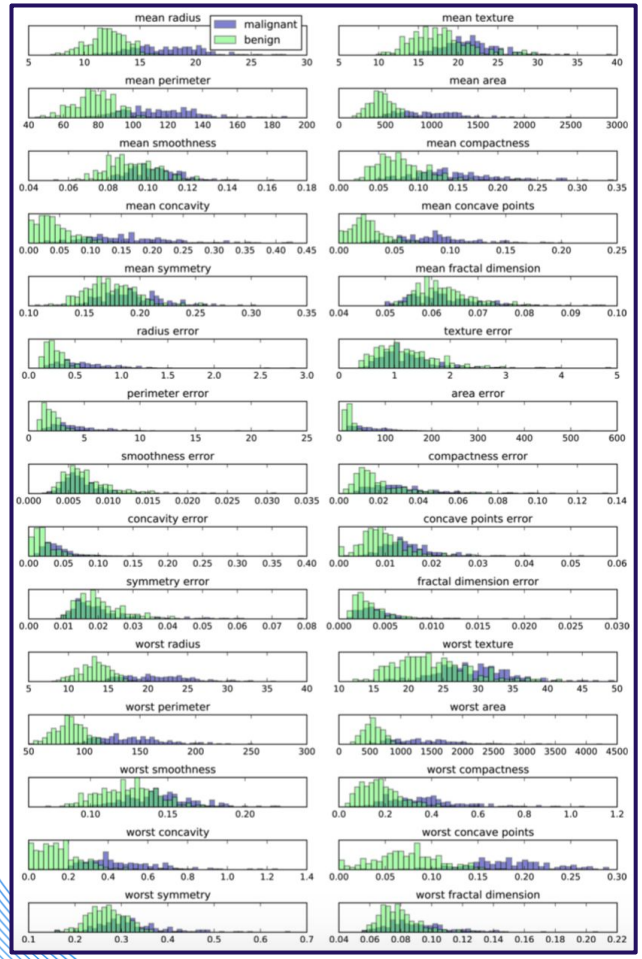
Hiperparámetro: n_components

En cambio, si lo que nos interesa es preservar cierto porcentaje de varianza (lo más utilizado)




```
pca = PCA(n_components=0.95)
X_reduced = pca.fit_transform(X)
```

Entonces se debe proveer un número real en el rango entre cero y uno



Ejemplo de aplicación: detección de cáncer de mama

```
from sklearn.datasets import load_breast_cancer
cancer = load_breast_cancer()

scaler = StandardScaler()
scaler.fit(cancer.data)
X_scaled = scaler.transform(cancer.data)
```

30 variables de entrada

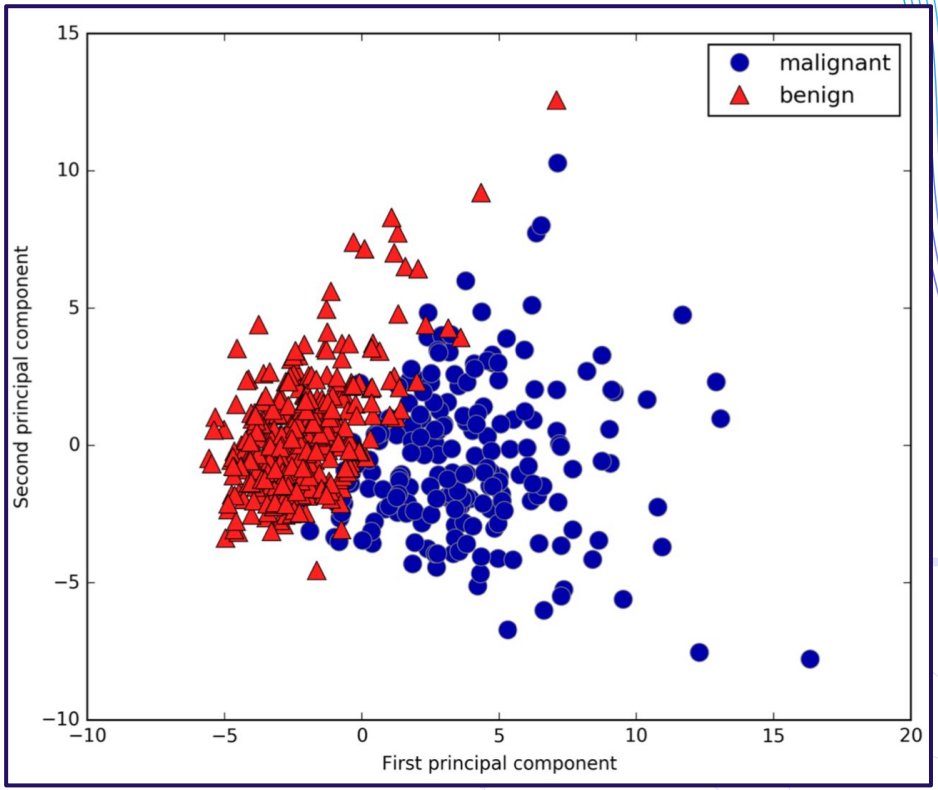
¡Con solo 2 componentes, tenemos un muy buen conjunto de datos!

```python
from sklearn.decomposition import PCA
# keep the first two principal components of the data
pca = PCA(n_components=2)
# fit PCA model to breast cancer data
pca.fit(X_scaled)

# transform data onto the first two principal components
X_pca = pca.transform(X_scaled)
print("Original shape: {}".format(str(X_scaled.shape)))
print("Reduced shape: {}".format(str(X_pca.shape)))
```

```
Original shape: (569, 30)
Reduced shape: (569, 2)
```

```python
# plot first vs. second principal component, colored by class
plt.figure(figsize=(8, 8))
mglearn.discrete_scatter(X_pca[:, 0], X_pca[:, 1], cancer.target)
plt.legend(cancer.target_names, loc="best")
plt.gca().set_aspect("equal")
plt.xlabel("First principal component")
plt.ylabel("Second principal component")
```



Bloque C

Aprendizaje no supervisado: clusterización

Clusterización

Pertenece al **aprendizaje no supervisado**, es una técnica que nos permite **descubrir estructuras ocultas** en los datos.

Ambos algoritmos (reducción de la dimensionalidad y clusterización) permiten **resumir** nuestros datos.

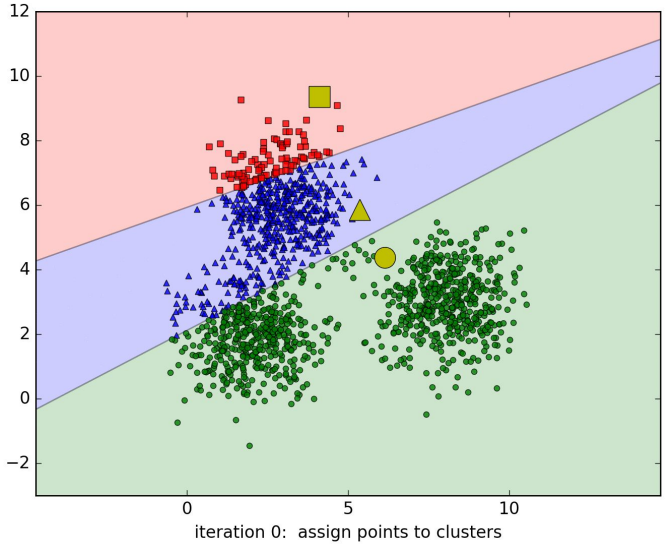
> PCA **comprime** nuestros datos mediante la representación de ellos en nuevas (y menor cantidad) de características, mientras que simultáneamente captura la mayor cantidad de información relevante. De forma similar, la clusterización, es una forma de reducir el volumen de datos y **encontrar patrones**. Lo logra mediante la categorización de la data, no mediante la creación de nuevas variables.

Clusterización

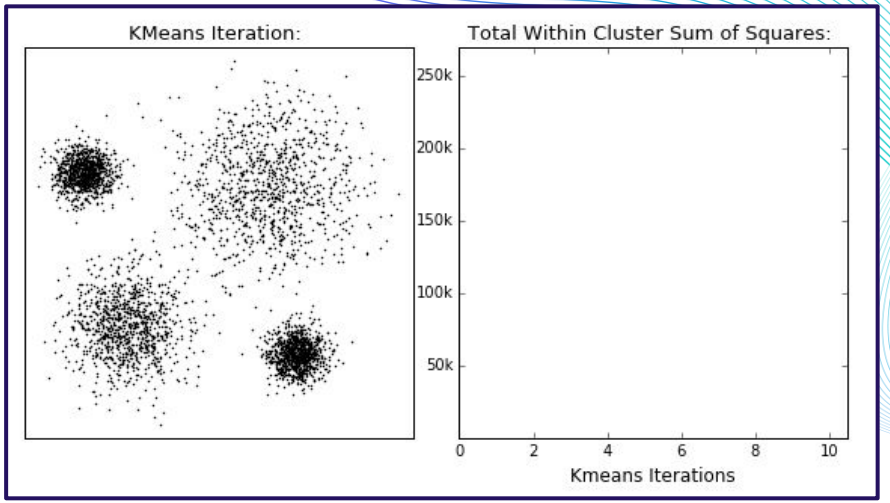
El objetivo de la clusterización es **encontrar una forma natural de agrupar nuestros datos**, de tal manera que los datos que pertenecen a un subgrupo (llamado clúster) son **más similares entre sí** que con los datos de los demás subgrupos.

Es un algoritmo no supervisado, ya que **no se sabe de antemano** cuántos subgrupos (clústers) se formarán, ni cuántos datos pertenecerán a cada uno de ellos.

También permite la **categorización automática** de nuevos datos mediante la regla que ha aprendido.



Algoritmo de k-medias



El objetivo es **encontrar k centroides** y asignar uno de ellos a cada registro, de tal forma que se **minimice** la varianza intra-clúster (llamada **inercia**). Usualmente se utiliza la distancia Euclidiana (distancia entre dos puntos), pero es posible ocupar otras métricas.

K-medias se encarga de encontrar un mínimo para una “k” dada, de la siguiente manera:

1. **Se elige una cantidad de clústers.**
2. **Algunos puntos se eligen aleatoriamente como los centroides de los clústers.**
3. **Cada dato se asigna al clúster de cuyo centro esté más cercano.**
4. **El centroide del clúster se actualiza con la media de los puntos asignados a él.**
5. **Los pasos 3 al 4 se repiten hasta que todos los centroides se mantengan sin cambios.**

Hiper parámetros de k-medias:

Cantidad de clústers

- Clústers (y centroides) a generar.

Cantidad máxima de iteraciones

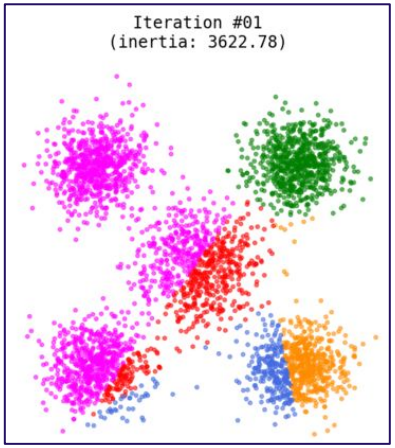
- Para **una ejecución** del algoritmo.

```
from sklearn.cluster import KMeans
#Fit with k-means
k_means = KMeans(n_clusters=nclust)
k_means.fit(X)
```

“Número inicial”

- Cantidad de veces que el algoritmo se ejecutará, utilizando **diferentes semillas** para la generación de los **centroides** El resultado final será el que mejor rendimiento haya tenido (en términos de la **inercia** .
- En la librería sklearn, el algoritmo se ejecuta **al menos 10 veces, con diferentes valores iniciales**.

Características de k-medias

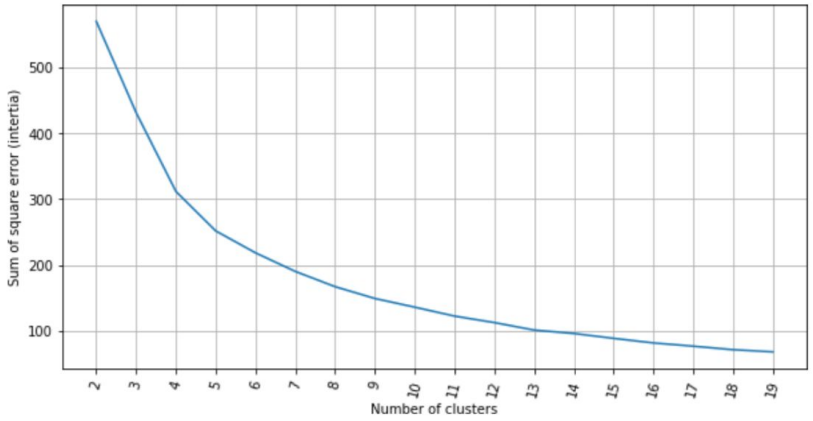


Ventajas

- Su principal ventaja radica en su simplicidad, su amplio rango de aplicabilidad, rápida velocidad de convergencia y su escalabilidad para conjuntos grandes de datos.

Desventajas

- Falta de garantía para encontrar el mínimo global (sin tener que ejecutar el algoritmo una considerable cantidad de veces).
- Puede ser sensible a valores atípicos.



Wholesale customers

Donated on 3/30/2014

The data set refers to clients of a wholesale distributor. It includes the annual spending in monetary units (m.u.) on diverse product categories

Dataset Characteristics
Multivariate

Subject Area
Business

Associated Tasks
Classification, Clustering

Feature Type
Integer

 # Instances
440

 # Features
7

Dataset Information

Has Missing Values?
No

Enlace

¡Gracias por su atención!

Datos de contacto:
rcanizales@uca.edu.sv
rcanizal@colostate.edu
www.linkedin.com/in/ronaldo-canizales/
https://x.com/ArmandoCodigos