ESPECIALIZACIÓN EN DATA SCIENCE

LUIS CHACON MONTALVAN

</revit>

Revolutionary IT Consulting

¿Será bueno tener muchas variables?

¿Todas las variables son diferentes?

¿Todas las variables son diferentes?

¿Si tengo muchos clientes, todos son iguales?

¿Existen variables que no e sirven?

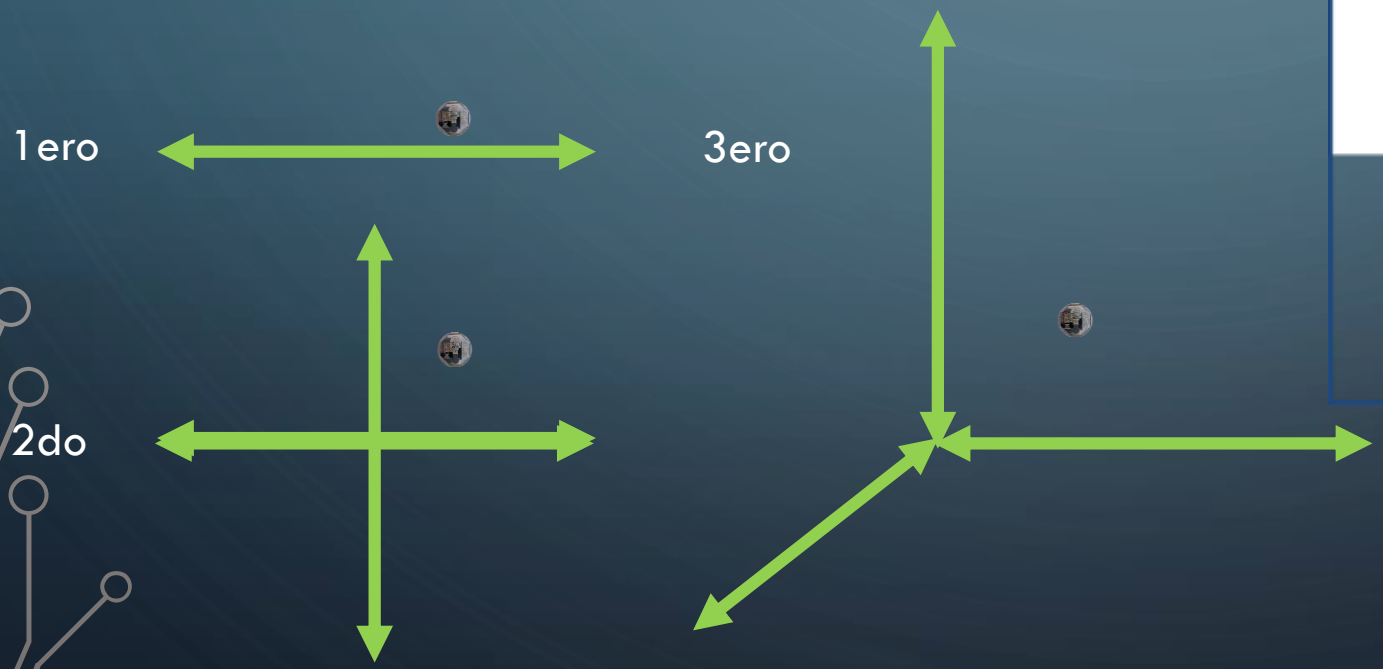
¿Más es mejor?

> *Existen una gran cantidad de factores que podemos tener en cuenta a la hora de seleccionar un algoritmo en* ***inteligencia artificial****, en realidad es casi un arte con mucho contenido científico, y entre los más importantes se encuentra el de decidir qué datos de entrada va a recibir nuestro sistema.*

</revit>
Revolutionary IT Consulting

LA MALDICIÓN DE LA ALTA DIMENSIONALIDAD

El problema más recurrente a la hora de trabajar con atributos redundantes o inservibles se torna “Muy tangible” en el momento en que tienes que procesar la información de tu algoritmo en un computador. Ya que si bien las computadoras son increíblemente más rápidas que hace 3 años, entre más atributos mayor tiempo de procesamiento. Veamos un ejemplo…



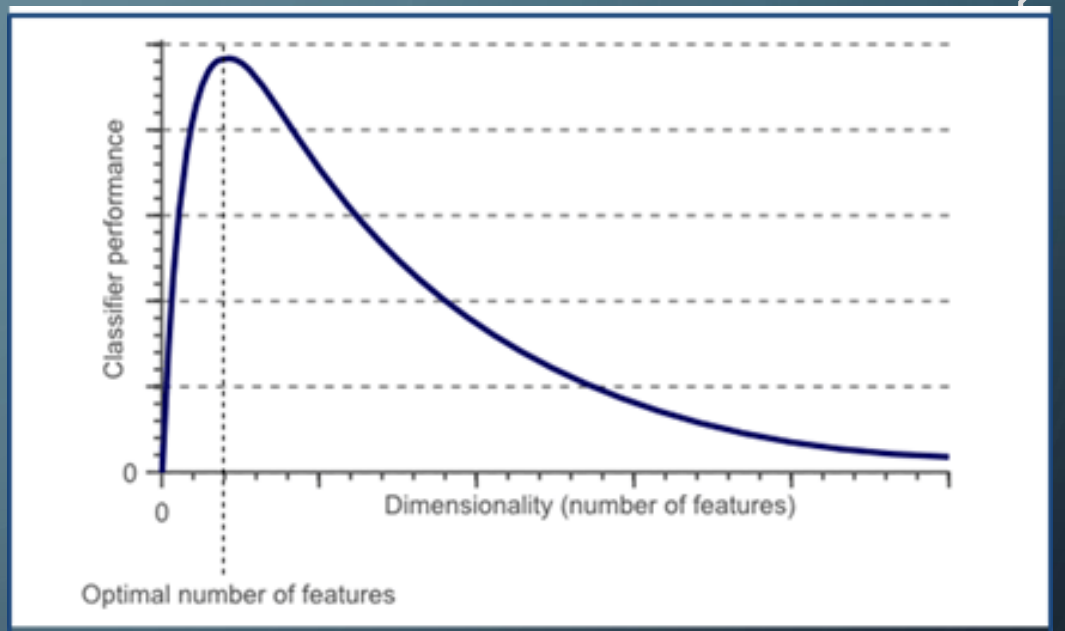


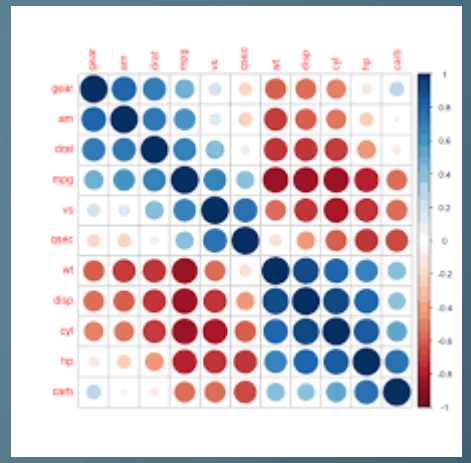
Figura 1. Conforme la dimensionalidad incrementa, el desempeño de nuestro clasificador aumenta, sin embargo tenemos un punto de inflexión al tomar en cuenta muchos atributos, es decir al aumentar su dimensionalidad

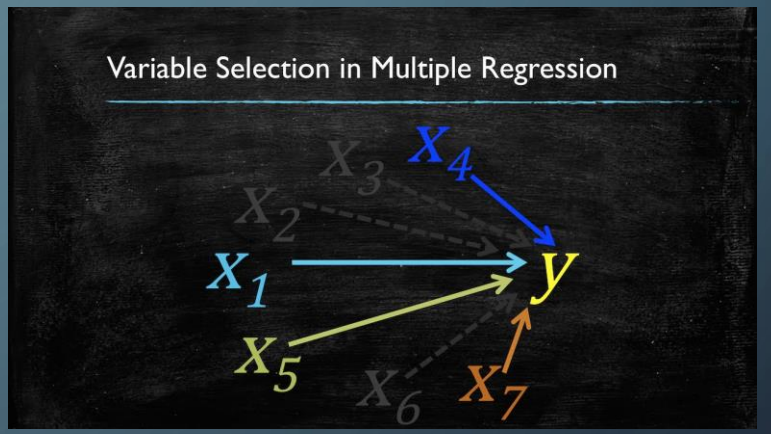
</revit>
Revolutionary IT Consulting

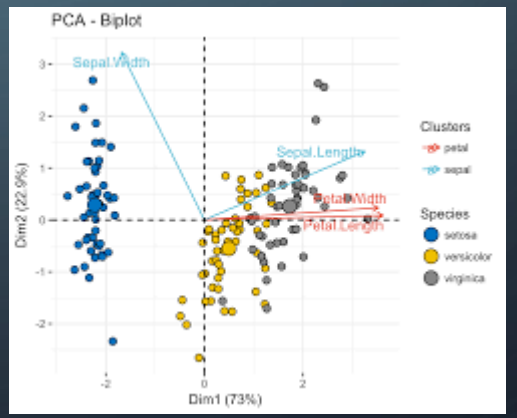
CÓMO EVITAR LA MALDICIÓN

Podemos evitar la maldición de la alta dimensionalidad de los datos, podemos tener en cuenta:

- ✓ Reducir la cantidad de dimensiones
  - Redundancia
  - Correlación
  - ACP
  - Factorial
  - Clustering
- ✓ Seleccionar la cantidad necesaria de atributos
  - Stepwise
  - Random forest
  - Boruta
  - Algoritmos genéticos



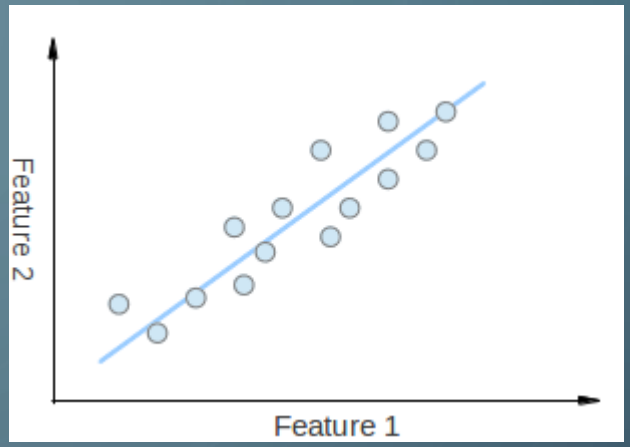




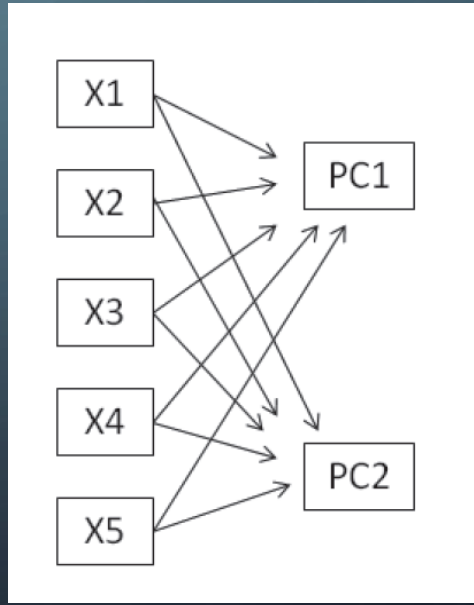
</revit>
Revolutionary IT Consulting

ANÁLISIS DE COMPONENTES PRINCIPALES

El análisis de componente principales ACP, consiste en buscar combinaciones lineales, de las variables originales, que representen lo mejor posible a la variabilidad presente en los datos. De este modo, se puede entender la información contenida en los datos, empleando únicamente, unas pocas combinaciones lineales a las cuales llamaremos componentes principales.



Un aspecto clave en el ACP es la interpretación de las componente principales, ya que esta no es a priori, sino que será deducida tras observar la relación de los componentes principales con las variables iniciales. Esto no siempre es fácil, y será de vital importancia el conocimiento que el experto tenga sobre las variables de estudio.



</revit>
Revolutionary IT Consulting

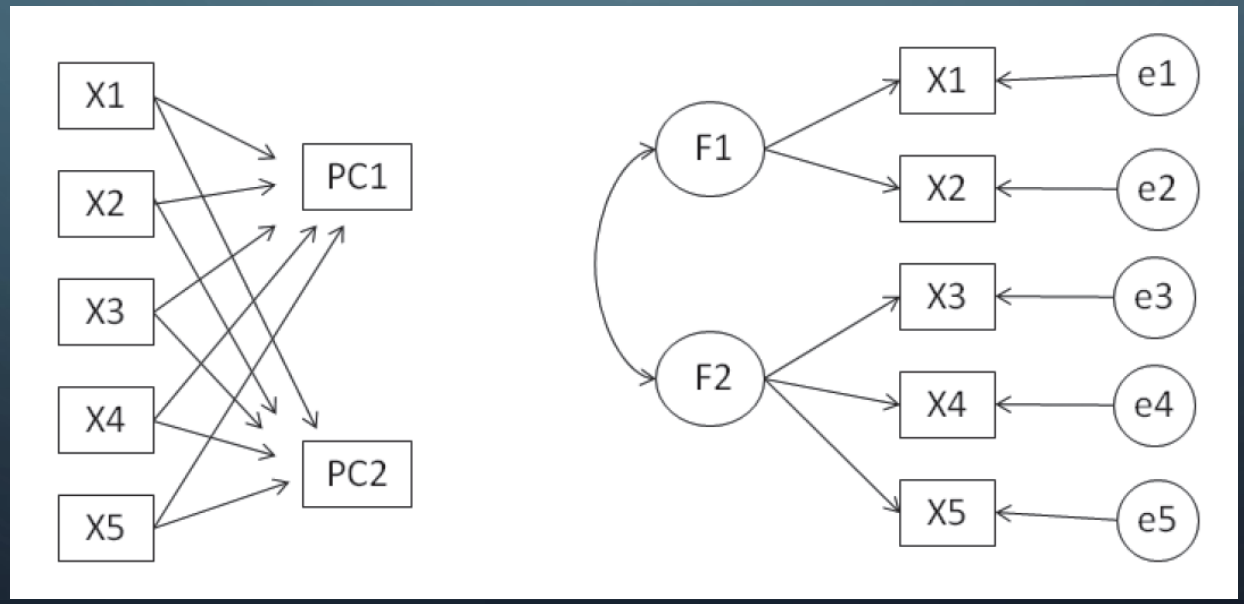
ANÁLISIS DE COMPONENTES PRINCIPALES

Los pasos para un análisis de componentes principales son:

- **Análisis de matriz de correlaciones**: El ACP tiene sentido si las variables están altamente correlacionadas, ya que esto nos indica que hay información redundante, por lo cual, la mayor parte de la información presente puede ser explicada por unos pocos componentes principales (CP).
- **Selección de componentes principales**: La elección de las CP se da de forma que la primera recoja la mayor proporción de la variabilidad original (`información`), la segunda recoja la mayor variabilidad posible que no fue recogida por la primera y así sucesivamente hasta que se recoja la proporción de variabilidad que se considere suficiente.
- **Análisis de la matriz factorial**: Una vez seleccionados los componentes principales, se representan en forma de matriz. Cada elemento de ésta representa los coeficientes factoriales de las variables (`las correlaciones entre las variables y los componentes principales`).
- **Interpretación del CP**: Para que un factor sea fácilmente interpretable debe tener las siguientes características:
  - Los coeficientes factoriales deben ser próximos a 1.
  - Una variable debe tener coeficientes elevados sólo con un factor.
  - No deben existir factores con coeficientes similares.
- **Cálculo de las puntuaciones factoriales**: Son las puntuaciones que tienen los componentes principales para cada caso, que nos permitirán su análisis posterior y su representación gráfica.

ANÁLISIS FACTORIAL

- EL Análisis Factorial es una técnica que consiste en entender la estructura latente que poseen un conjunto de variables observables, cuyo objetivo es encontrar los factores latentes o constructos que expliquen el comportamiento y/o relación de las variables observadas.
- Es necesario mencionar la diferencia entre el Análisis de Componentes Principales y el Análisis Factorial. Esto se puede ver de manera gráfica en la figura siguiente.
- Mientras que un ACP es una combinación lineal de todas las variables observables, el AF consiste en obtener factores latentes que son asumidos causantes de las variables observables.
- Por ejemplo, en la parte derecha de la imagen, se puede ver que F1 es la causa de las variables X1, X2 y F2 es la causa de las variables X3, X4 y X5, los cuales son inferidos a través de las correlaciones entre las variables. Además, los errores (e1, e2, e3, e4 y e5) representan la varianza de las variables observadas no explicadas por los factores.



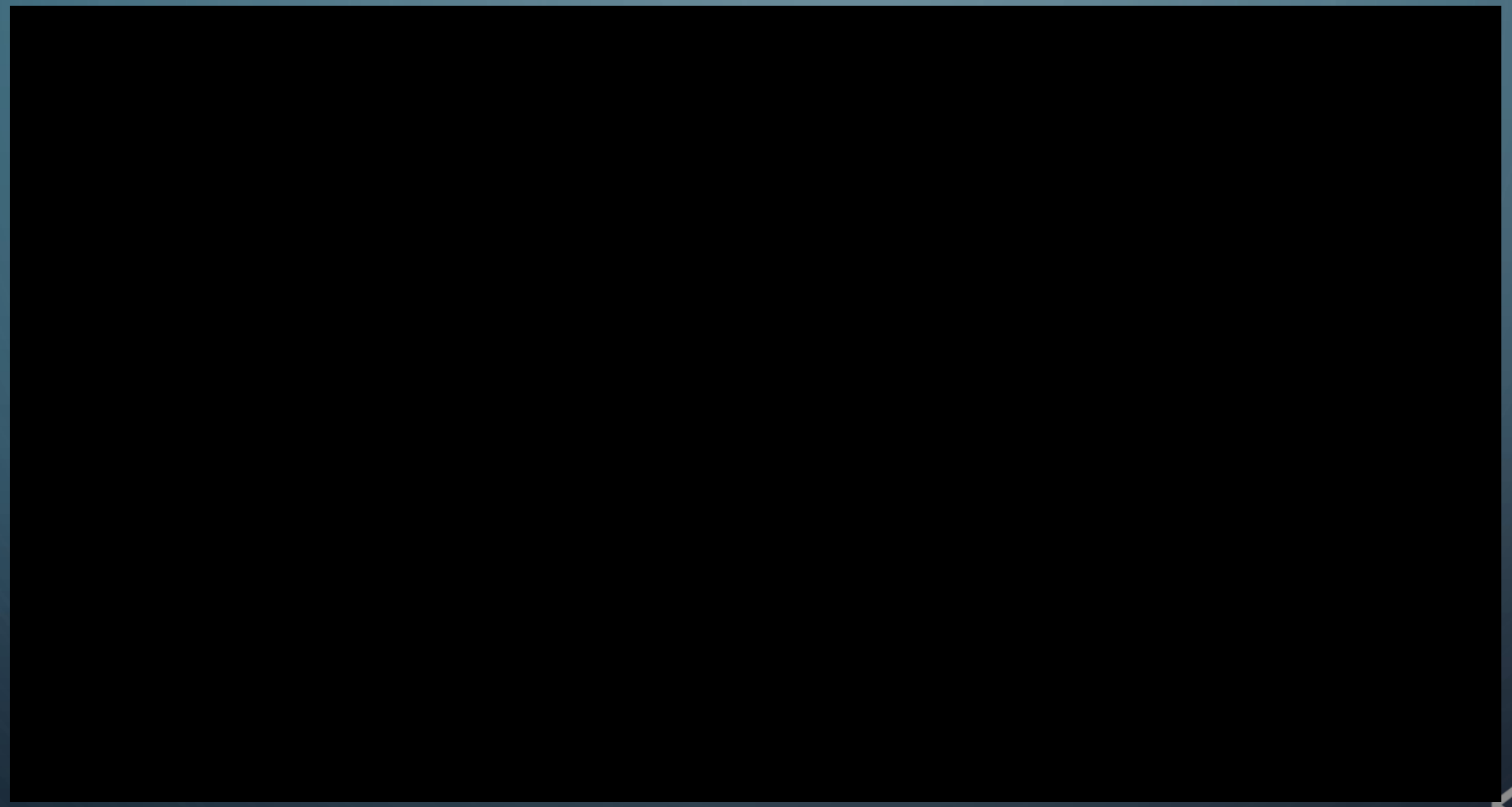
</revit>
Revolutionary IT Consulting

ANÁLISIS FACTORIAL

Los pasos para un análisis factorial son:

- **Análisis de matriz de correlaciones**: Permite apreciar si existe una estructura oculta en los datos; es decir, si existen subconjuntos claros de las variables.
- **Determinar el número de factores**: La elección radica en utilizar el diagrama de sedimentación considerando un análisis paralelo con los enfoques de CP y AF, además del criterio de Kaiser-Harrislas.
- **Extraer los factores comunes**: Existen diversos métodos como máxima verosimilitud, eje principal iterado, mínimos cuadrados ponderados, mínimos cuadrados ponderados generalizados y residual mínimo. Los estadísticos preferimos utilizar el enfoque de máxima verosimilutd debido a que se basa en un modelo estadístico bien definido. Sin embargo, si este método falla en la convergencia, se recomienda utilizar el enfoque de eje principal iterado.
- **Rotación de factores**: La rotación es necesaria para poder interpretar mejor los factores. Comúnmente se emplea la rotación varimax.
- **Cálculo de las puntuaciones factoriales**: Son las puntuaciones que tienen los factores para cada variable. A diferencia de los obtenidos en un ACP, en el AF las puntuaciones sólo pueden ser estimadas.

IA – MACHINE LEARNING – DEEP LEARNING



</revit>
Revolutionary IT Consulting

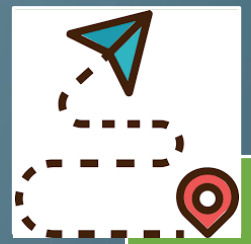
INTRODUCCIÓN AL MACHINE LEARNING

¿Qué es?

- El ***machine learning*** es un campo que se deriva de la IA. Consiste en desarrollar procesos que permitan a las máquinas aprender por sí solas a partir de los datos.
- No aprenden por sí mismas, sino que están programadas para adaptar su algoritmo conforme reciben información.

¿Cómo?

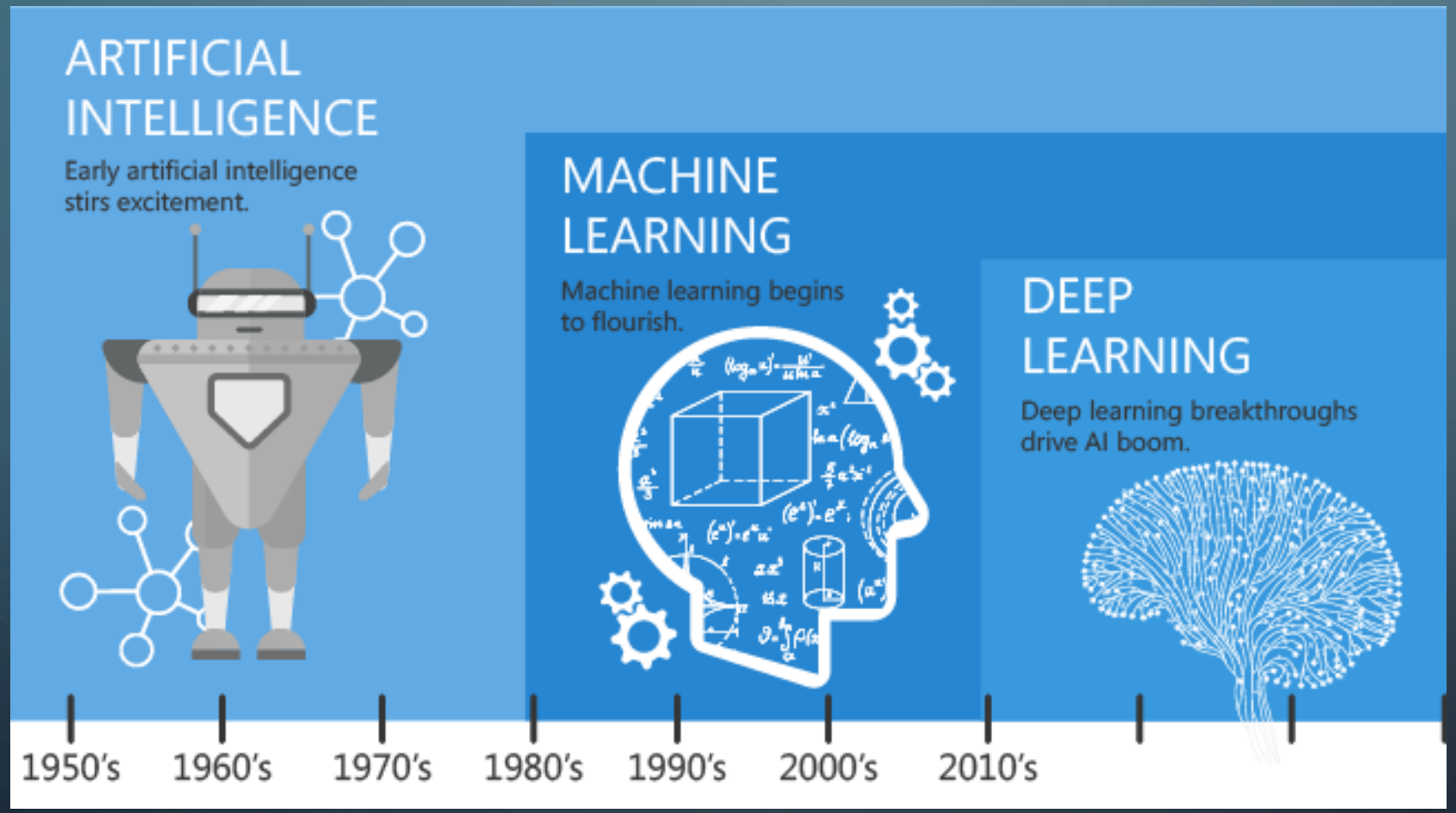
- No se ingresa manualmente las ordenes.
- Los algoritmos ML resuelven por su cuenta las adversidades que tienen al hacer inferencias a partir de los datos, y mientras más datos tengan, mejores serán los resultados.
- La clave está en que puede reencausar los resultados parciales que va obteniendo



¿Hacia donde vamos?

- Vamos en camino de complementar el trabajo humano con el trabajo de la máquina.
- Especializar las aplicaciones según rubro o actividades específicas. (Sistema experto).
- Las aplicaciones en el trabajo abarcan todo, desde mejorar las experiencias minoristas en las tiendas con IoT hasta aumentar la seguridad con datos biométricos hasta predecir y diagnosticar enfermedades

INTRODUCCIÓN AL MACHINE LEARNING



IBM WATSON

SAS ML

</revit>

Revolutionary IT Consulting

MACHINE LEARNING – TIPOS DE PROBLEMAS

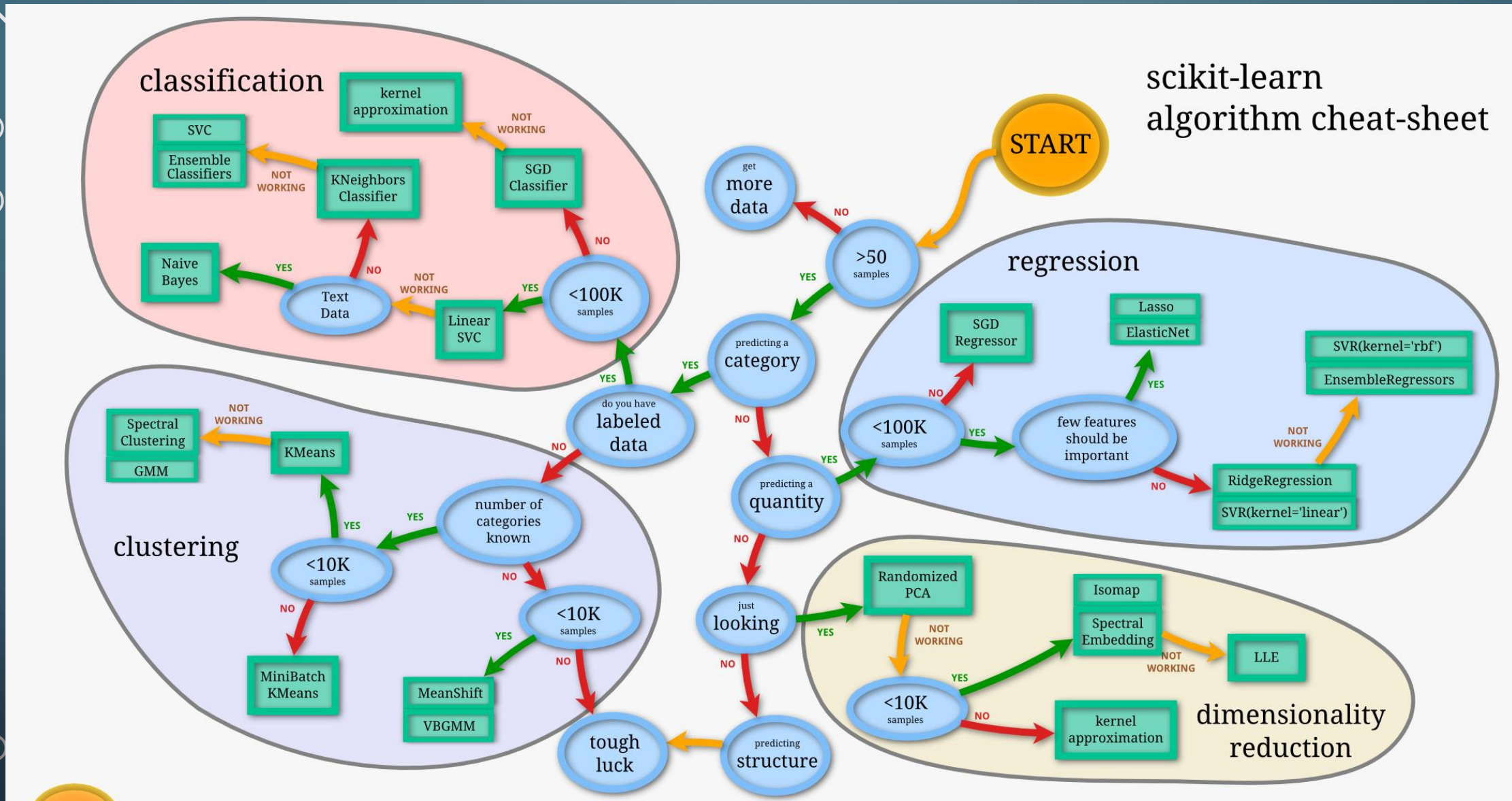
	Variable Continua	Variable categórica
Supervisado	Regresión	Clasificación
No Supervisado	Reducción de la dimensionalidad	Clustering

</revit>
Revolutionary IT Consulting

MACHINE LEARNING – TIPOS DE ALGORITMOS

Regresión *Numérico, Supervisado*	Precio de una vivienda
Clasificacion *Categórico, Supervisado*	Cliente compra/no compra
Clustering *Categórico, No Supervisado*	Segmentación de clientes
Reducción de Dimensionalidad *Numérico, No Supervisado*	Reducción de imágenes para clasificacion

</revit>
Revolutionary IT Consulting


scikit-learn
algorithm cheat-sheet
START
get more data
>50 samples
predicting a category
do you have labeled data
predicting a quantity
just looking
predicting structure
tough luck
<100K samples
Linear SVC
Text Data
Naive Bayes
KNeighbors Classifier
Ensemble Classifiers
SVC
SGD Classifier
kernel approximation
classification
<100K samples
SGD Regressor
Lasso
ElasticNet
few features should be important
SVR(kernel='rbf')
EnsembleRegressors
RidgeRegression
SVR(kernel='linear')
regression
number of categories known
<10K samples
KMeans
Spectral Clustering
GMM
MiniBatch KMeans
<10K samples
MeanShift
VBGMM
clustering
Randomized PCA
Isomap
Spectral Embedding
LLE
<10K samples
kernel approximation
dimensionality reduction
YES
NO
NOT WORKING
Back
scikit learn

AGRUPAMIENTO - CLUSTERING

Partimos de algo que es cierto, no todos somos iguales, incluso dentro de grupos o clases ya establecidas se forman, por diferentes motivos, subgrupos. Como ejemplo práctico, no todos los consumidores de un retail son iguales por lo cual no es justo que a todos los analicemos de la misma manera o que les ofrezcamos la misma oferta.

Por ello es necesario agrupar aquellas observaciones, elementos de la población de estudio, que en función a una medida de asociación o similitud se parecen o están más cercanos. Lo ideal es que los elementos de un mismo grupo sean lo más parecido posible entre sí y lo más diferentes posible entre grupos.

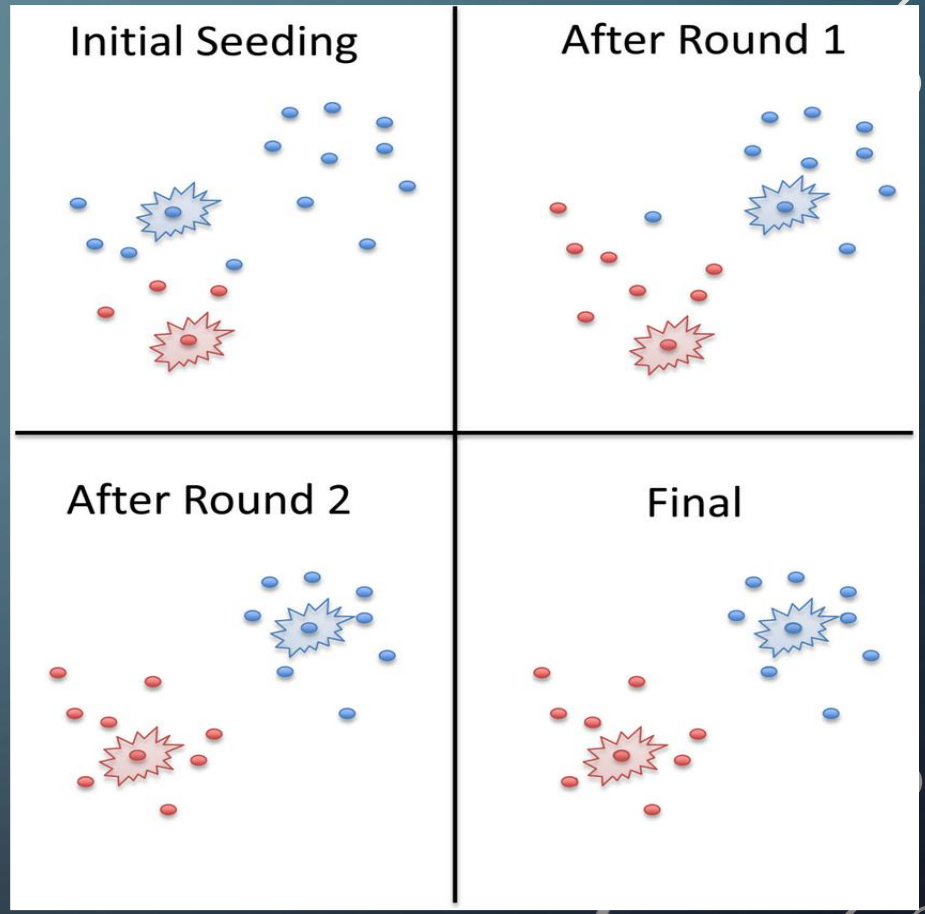
Un ser inteligente no puede tratar todos los objetos que ve como una entidad única a diferencia de cualquier otra cosa en el universo. Tiene que poner los objetos en categorías de modo que pueda aplicar su conocimiento logrado con gran esfuerzo sobre objetos similares encontrados en el pasado al objeto en cuestión.

— Steven Pinker (1997)

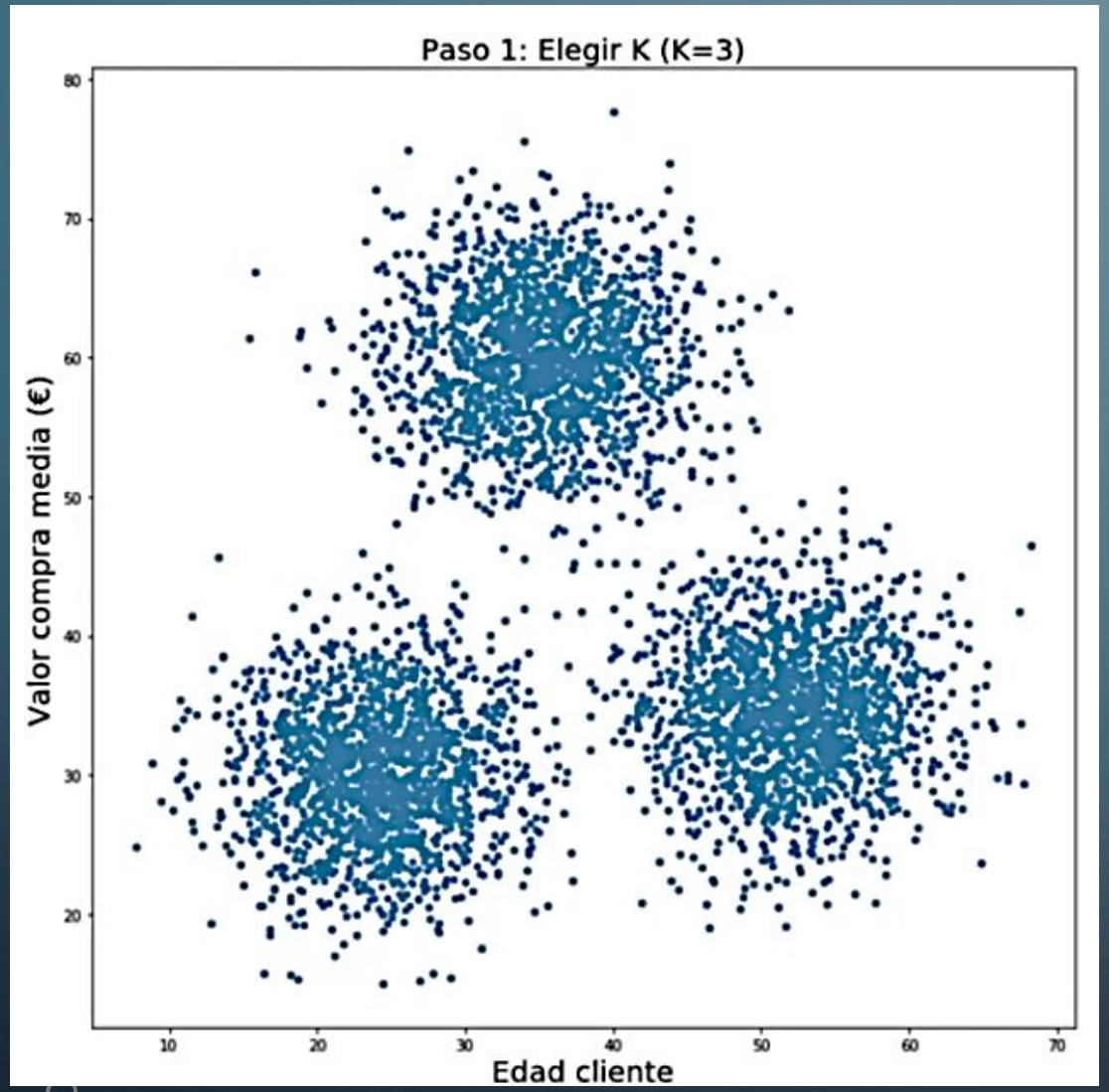
</revit>
Revolutionary IT Consulting

K MEANS

- Es un enfoque simple y elegante para particionar un conjunto de datos en K grupos distintos y no superpuestos.
- Es un método de agrupamiento heurístico con número de clases conocido (k) que se basa en las distancias entre centroides (cada centroide es el centro de un grupo) para generar el agrupamiento en k clusters previamente solicitados.
- El algoritmo no es determinista por lo que cada ejecución podría generar resultados muy variados dependiendo del método de elección de los centroides, además de que podría alcanzar un estado en el que nunca cumpla la condición de parada y por tanto nunca converja.



¿CÓMO FUNCIONA EL ALGORITMO K-MEANS?

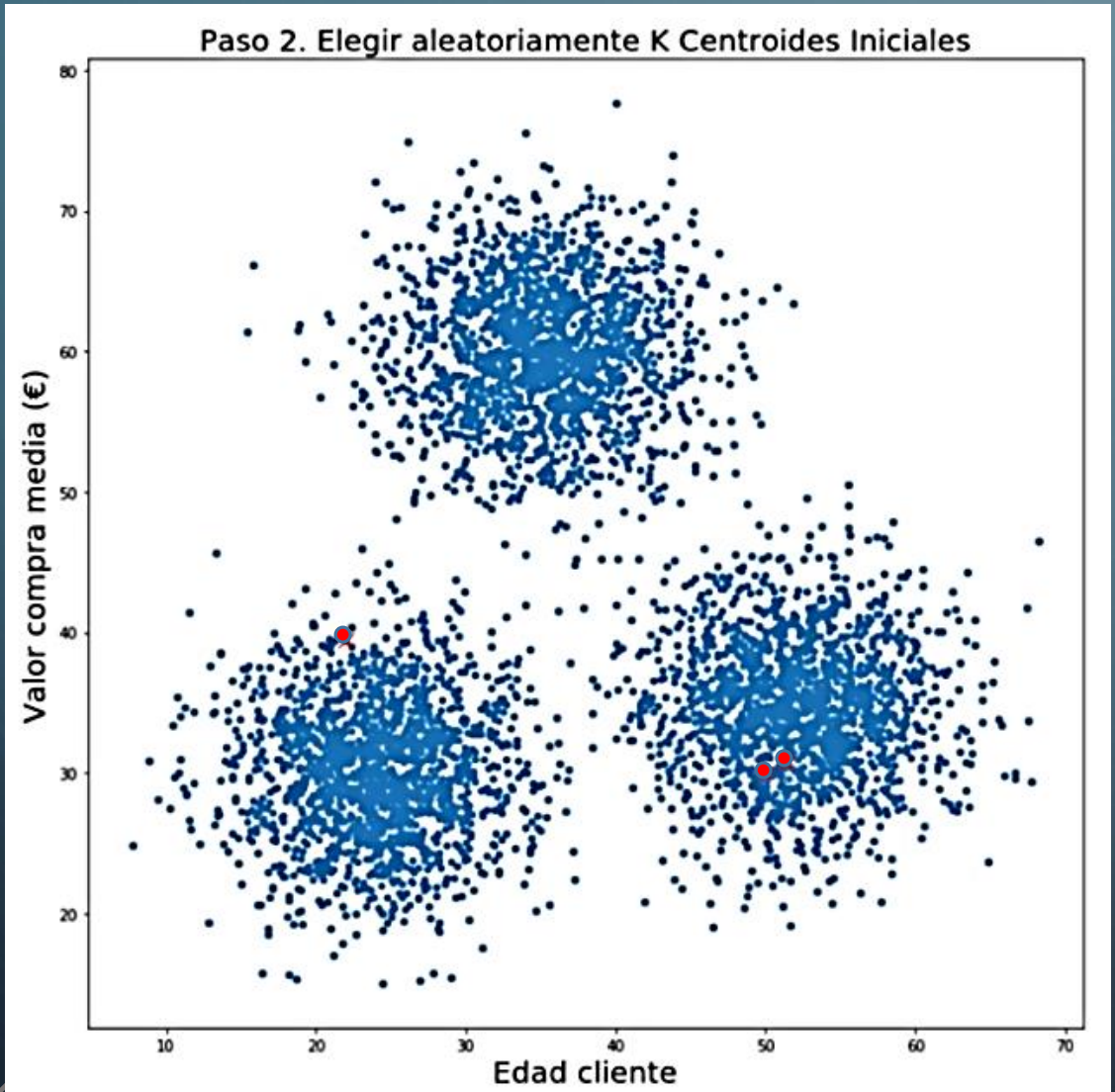


Paso 1. Debemos elegir K

- K es un hiperparámetro que tenemos que indicar a K-Medias. Es Decisión nuestra el decidir en cuantos clusters queremos dividir los datos.
- Para nuestro ejemplo trabajaremos con K = 3. (no siempre es tan obvio)

</revit>
Revolutionary IT Consulting

¿CÓMO FUNCIONA EL ALGORITMO K-MEANS?

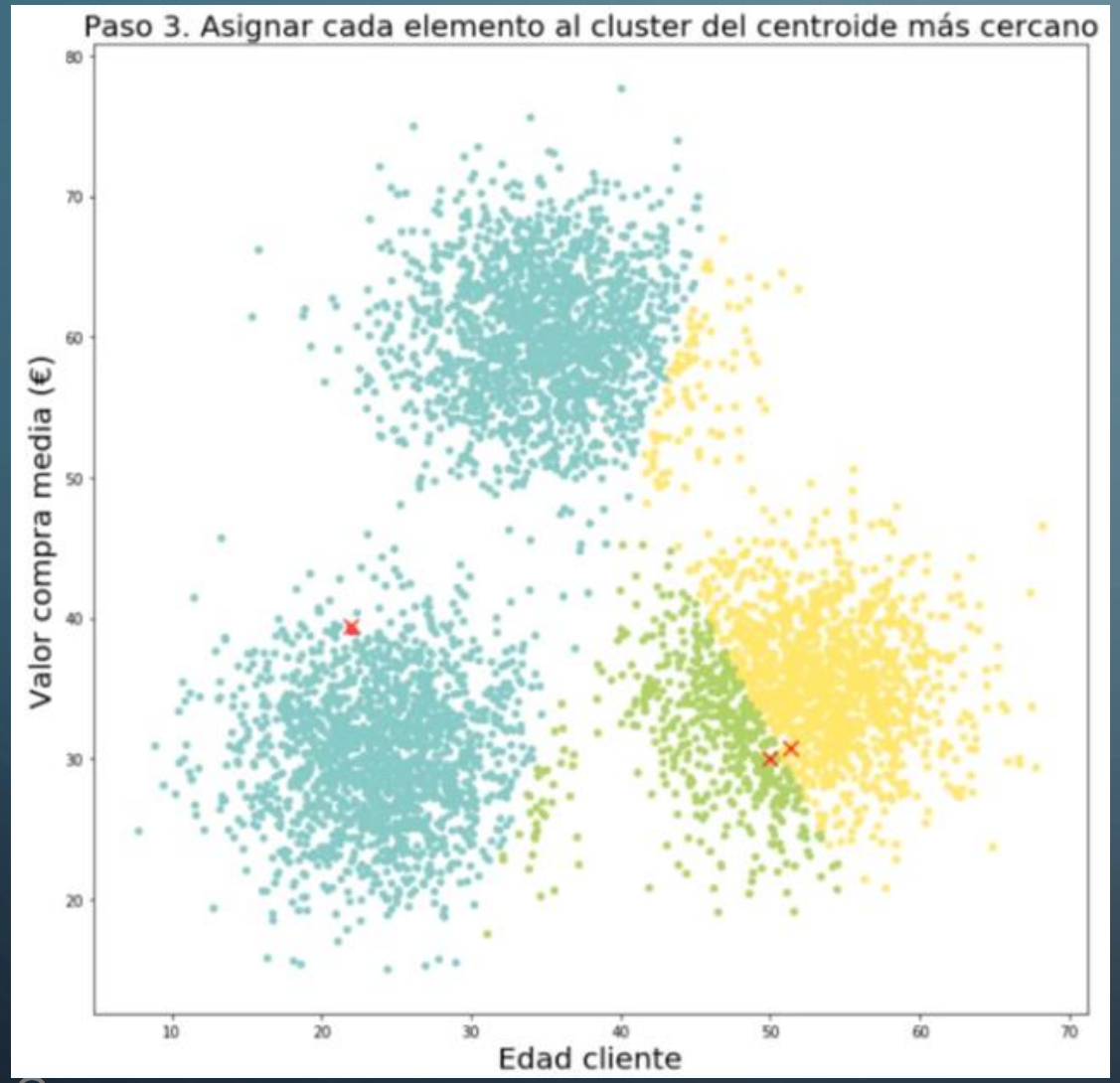


Paso 2. Asignar K centrodies al azar

- Centroides son los puntos que están en el centro de cada cluster

</revit>
Revolutionary IT Consulting

¿CÓMO FUNCIONA EL ALGORITMO K-MEANS?



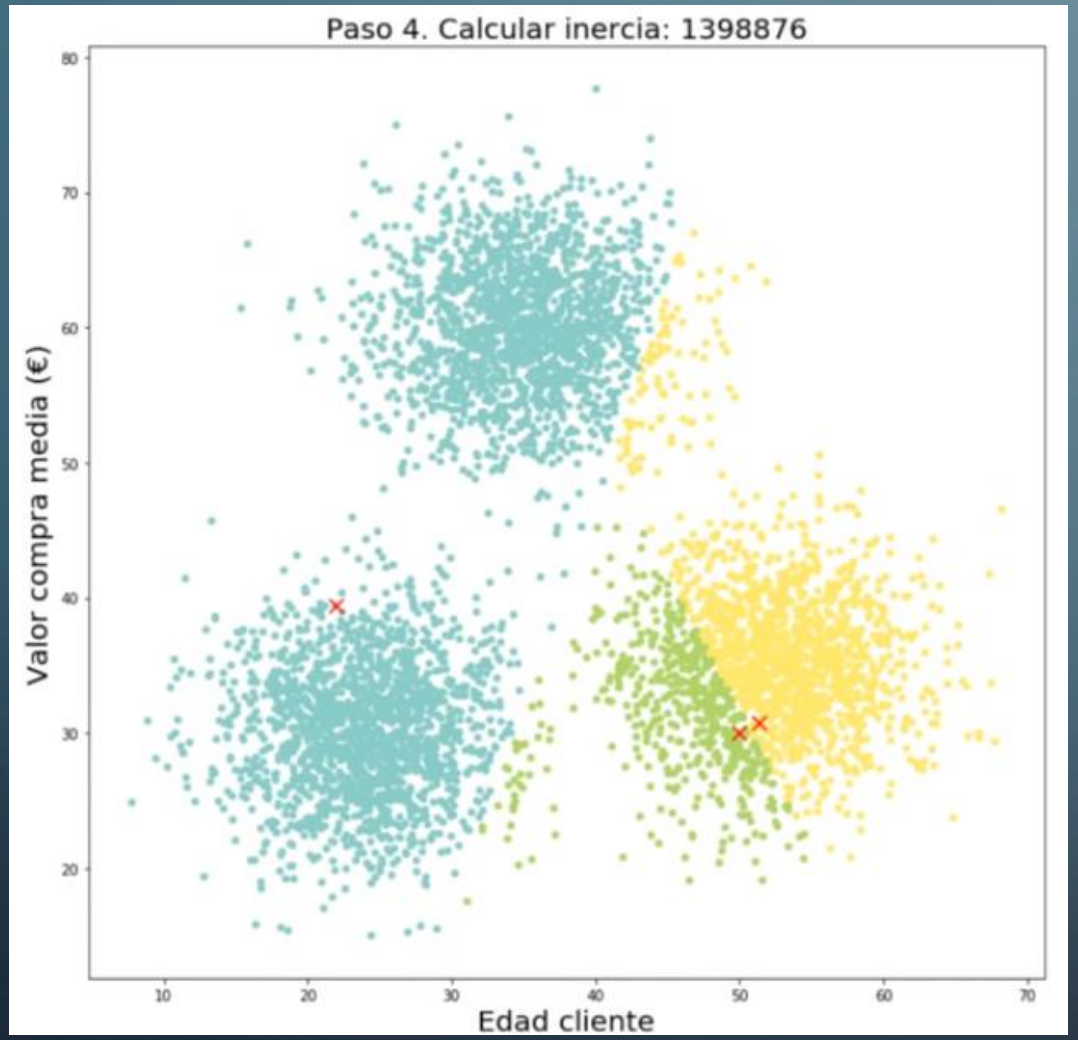
Paso 3. Asignar cada observación al cluster más cercano

- Se puede emplear muchas distancias, pero la más común es la distancia euclidea

$$\sqrt{\sum_{i=1,n} |x_i - y_i|^2}$$

</revit>
Revolutionary IT Consulting

¿CÓMO FUNCIONA EL ALGORITMO K-MEANS?



Paso 4. Calcular la Inercia

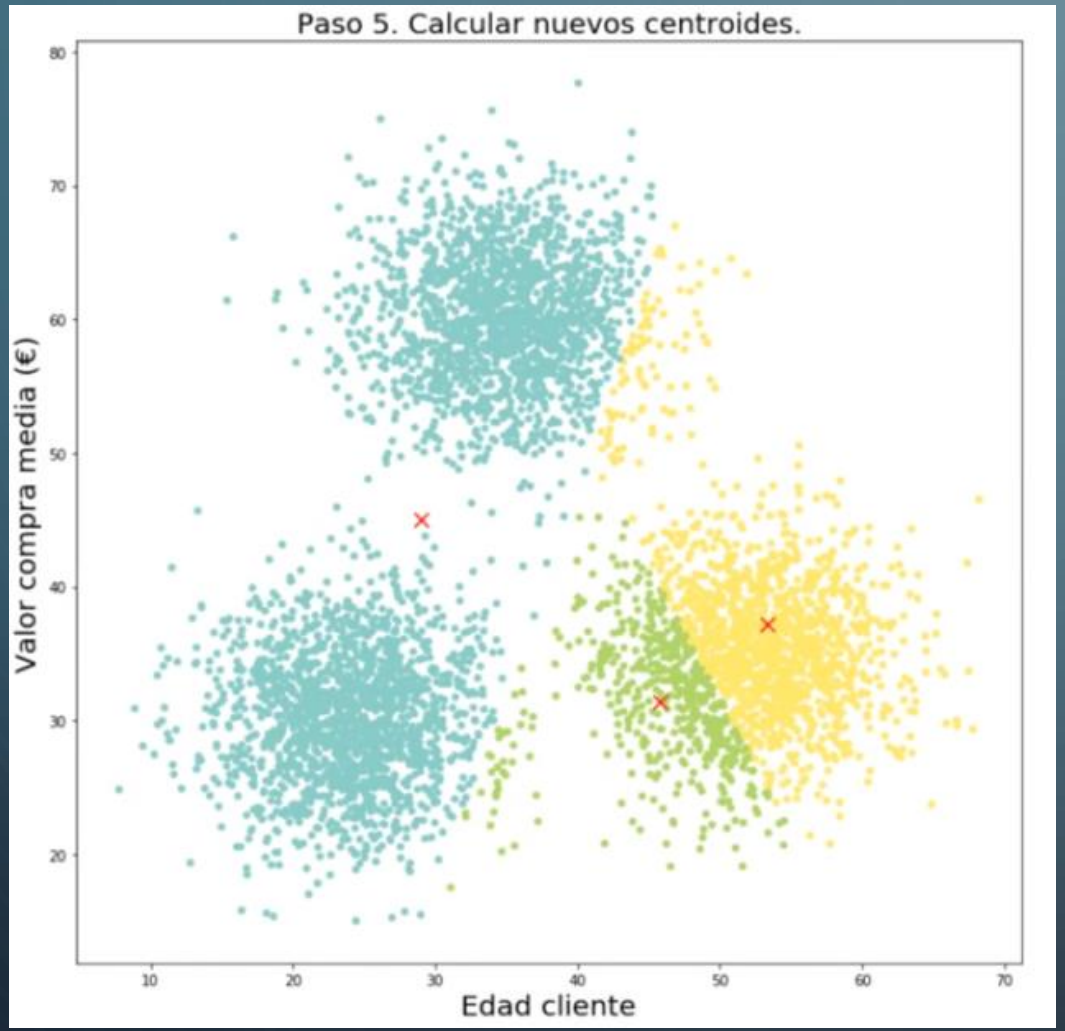
- La condición de parada del algoritmo Kmedias es la Inercia de los clusters, medida como:

$$\sum_{i=0,n} min(||x_j - \mu_i||^2), \mu_j \epsilon C$$

- Es decir, la sumatoria de los cuadrados de las distancias de cada observación al centroide más cercano (el cluster C al que pertenecen)
- Si la Inercia ha llegado a su mínimo o es menor que un valor especificado anteriormente, el algoritmo finaliza

</revit>
Revolutionary IT Consulting

¿CÓMO FUNCIONA EL ALGORITMO K-MEANS?

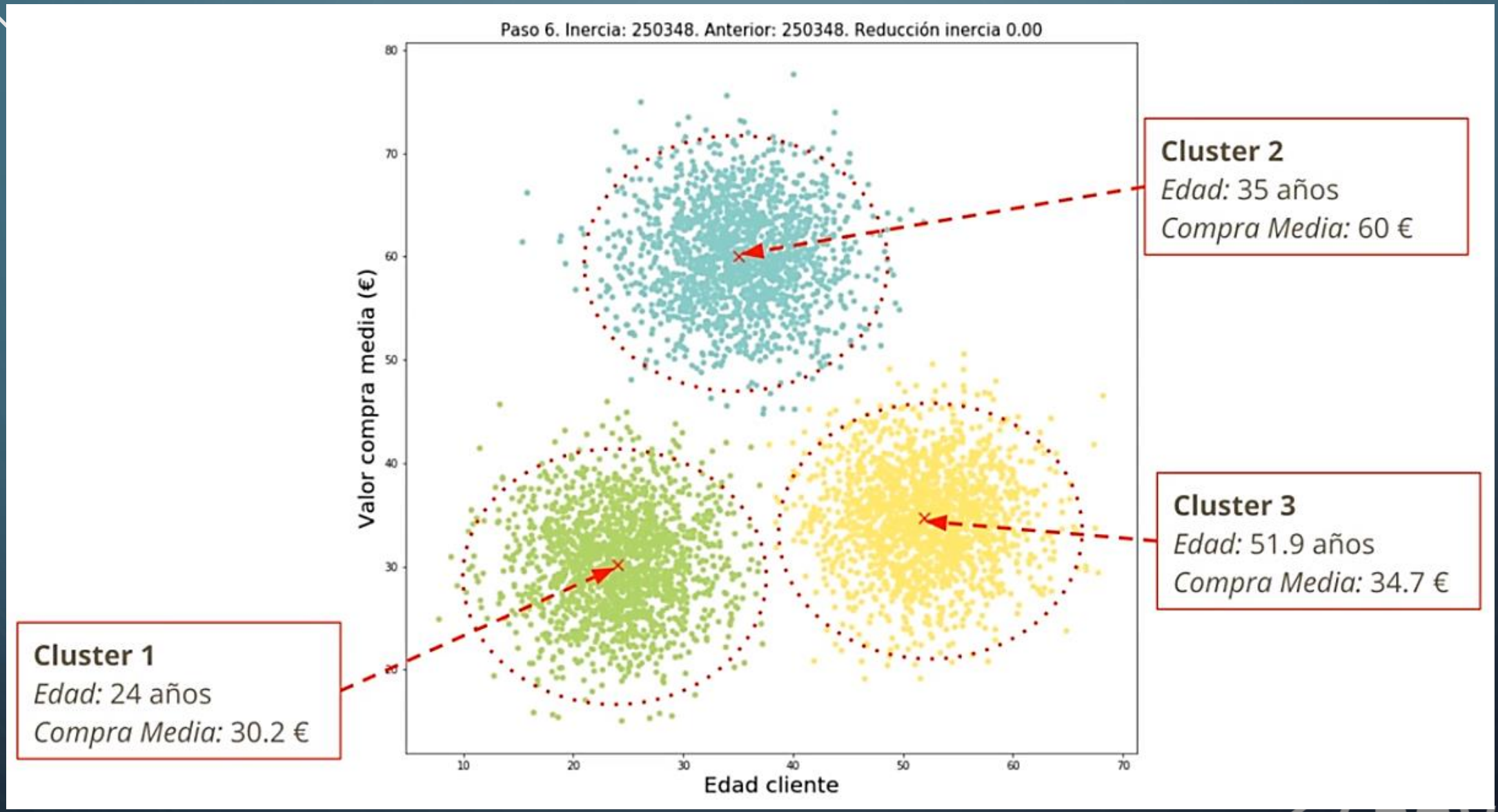


Paso 5. Calcular los nuevos centroides

- Los centroides se calculan ahora como las medias de los puntos que pertenecen a cada centroide del paso anterior (de ahí el nombre k-means)

- Se vuelve al paso 3

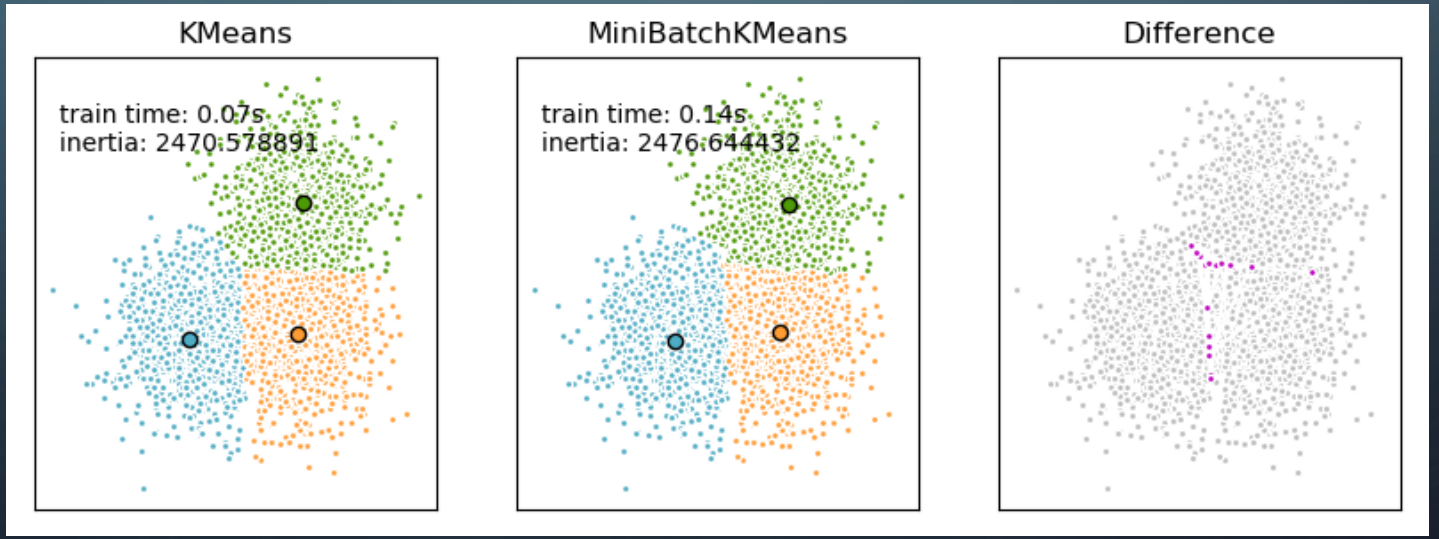
</revit>
Revolutionary IT Consulting



MINI-BATCH K-MEANS

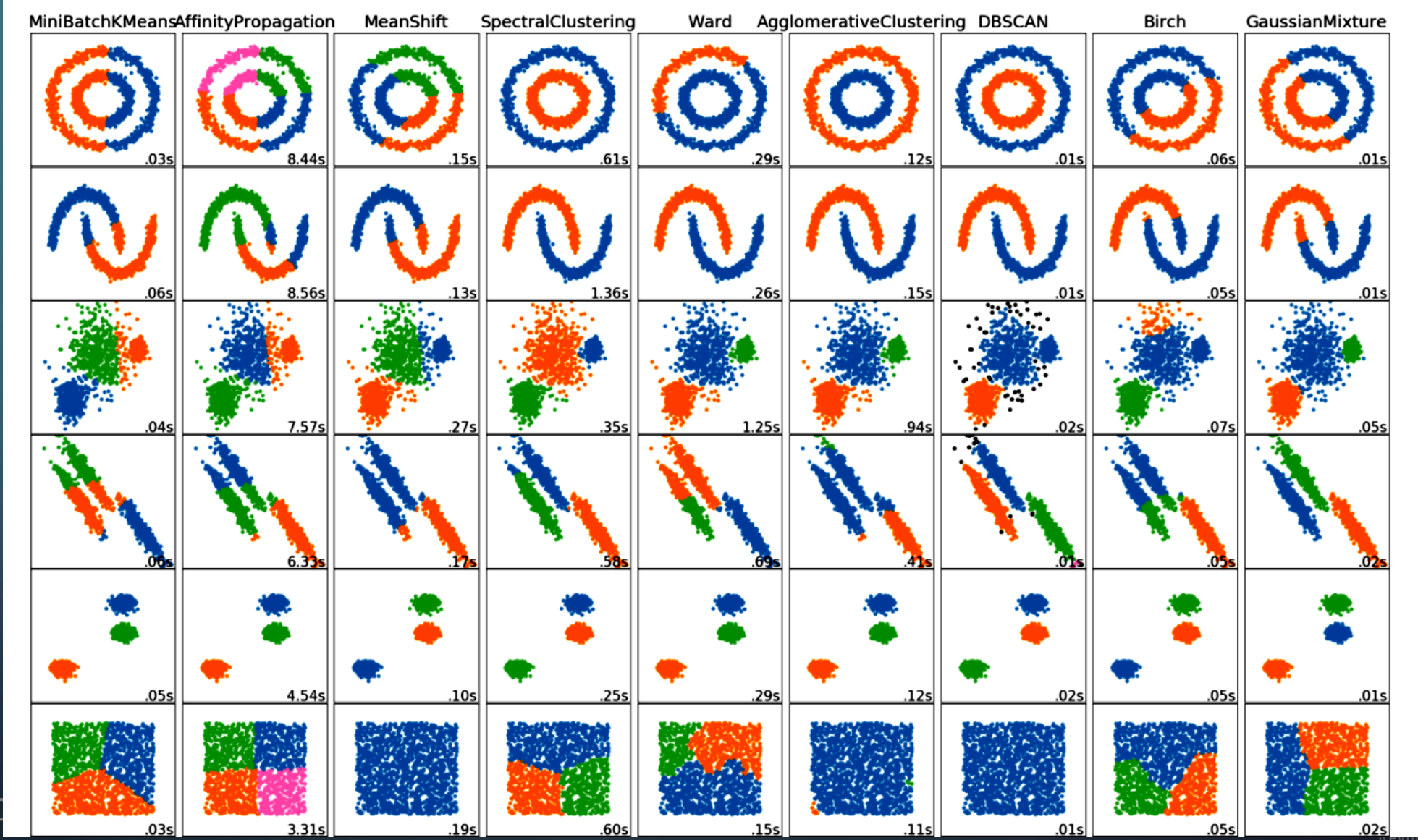
Es una variante del algoritmo k-means que utiliza mini-lotes para reducir el tiempo de cálculo. Los mini lotes son subconjuntos de los datos de entrada, muestreados aleatoriamente en cada iteración de entrenamiento. Estos mini lotes reducen drásticamente la cantidad de cálculos necesarios para converger a una solución local. A diferencia de otros algoritmos que reducen el tiempo de convergencia de los k-means, los mini-batch k-means producen resultados mas rápidos a cambio de tan solo un poco de precisión en comparación con el algoritmo estándar. El algoritmo itera básicamente dos pasos principales, similar al k-means.

- En el primer paso, las muestras se extraen al azar, para formar un mini lote. Estos se asignan al centroide más cercano.
- En el segundo paso, los centroides se actualizan, a diferencia de k-means, esto se hace por muestra. Para cada muestra en el mini-lote, el centroide asignado se actualiza tomando el promedio de transmisión de la muestra y todas las muestras anteriores asignadas a ese centroide. Esto tiene el efecto de disminuir la tasa de cambio para un centroide a lo largo del tiempo. Estos pasos se realizan hasta que se alcanza la convergencia o un numero predeterminado de iteraciones.



</revit>
Revolutionary IT Consulting

DBSCAN



DBSCAN

DBSCAN (Density-based spatial clustering of applications with noise) es un algoritmo de clustering propuesto en 1996 que es más avanzado que KMedias, ya que funciona mejor para clusters con geometrías más complejas.

KMedias se basa en el concepto de centroide (como el centro geométrico de un cluster) e intenta reducir la distancia de los puntos de un cluster al mismo.

DBSCAN se basa en el concepto de densidad de cluster e intenta agrupar los puntos que están más juntos.

DBSCAN

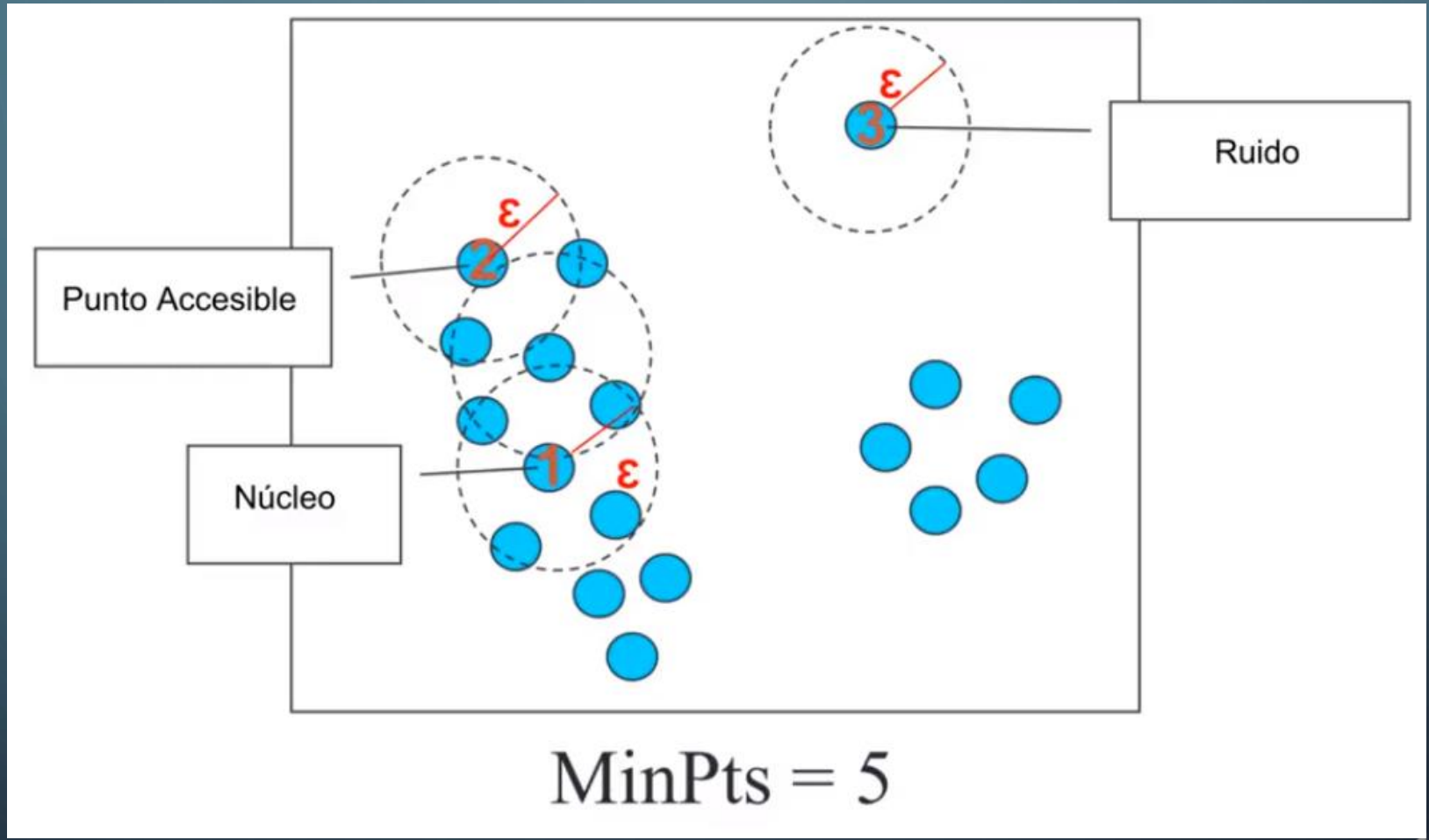
DBSCAN precisa 2 hiperparámetros.

- **MinPts**, el mínimo número de puntos alrededor de un punto para considerarlo un núcleo
- **Ɛ (eps),** el radio alrededor de cada punto para considerar puntos cercanos

DBSCAN define 3 tipos de puntos:

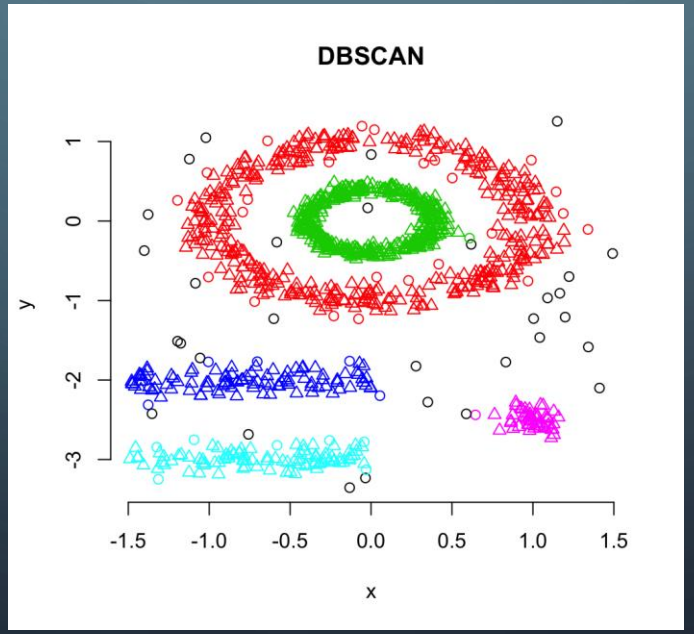
- **Núcleos** (core points). Aquellos puntos que tienen un número de puntos igual o mayor que **MinPts** en un radio de **Ɛ** alrededor.
- **Alcanzables** (reachable points). Aquellos puntos que no son núcleos pero que tienen un núcleo dentro de su radio **Ɛ**
- **Ruido** (noise). Aquellos puntos que no son ni núcleos ni puntos alcanzables

DBSCAN



DBSCAN ALGORITMO

- 1. Encontrar los vecinos de cada punto en su radio ε (eps), e identificar como núcleos aquellos puntos con más vecinos que minPts.
- 2. Agrupar aquellos núcleos que esten a distancia ε (eps) o menor, considerándolos parte del mismo cluster.
- 3. Asignar a cada punto que no sea núcleo a un cluster cercano si el cluster es vecino del punto. Aquellos puntos que no tengan ningún cluster cercano se consideran ruido.



</revit>
Revolutionary IT Consulting

DBSCAN VENTAJAS Y DESVENTAJAS

Ventajas:

- DBSCAN no precisa indicarle el número de clusters a buscar como hiperparámetro
- Al usar el criterio de densidad de puntos (y no distancias a un centro geométrico), DBSCAN es capaz de encontrar clusters que tengan formas complejas
- DBSCAN es robusto a outliers
- DBSCAN está diseñado para funcionar de forma óptima en bases de datos

Inconvenientes:

- DBSCAN no es determinista, y los resultados pueden variar según el orden en el que se evalúen las observaciones para aquellos puntos que estén a distancia de dos o más núcleos. Existe una versión renovada del algoritmo HDBSCAN* que evita ésto.

</revit>

Revolutionary IT Consulting