SISTEMAS INTELIGENTES ARTIFICIALES

Prof.: Esp. Ing. Agustín Fernandez

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Utilizaremos WEKA para realizar las practicas:
  - Su ultima versión estable es la 3.8
  - Desarrollado por la universidad de Waikato: waikato.ac.nz
  - Sitio web de weka: https://www.cs.waikato.ac.nz/ml/weka/
  - Según sus creadores:
    - Weka es un software de aprendizaje automático de código abierto (probado) al que se puede acceder a través de una interfaz gráfica de usuario, aplicaciones de terminal estándar o una API de Java. Se usa ampliamente para la enseñanza, la investigación y las aplicaciones industriales, contiene una gran cantidad de herramientas integradas para tareas estándar de aprendizaje automático y, además, brinda acceso transparente a cajas de herramientas conocidas como scikit-learn, R y Deeplearning4j.

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- WEKA permite realizar ML sin programar si así se lo desea:
  - Disponible para Win, Linux y MacOS
  - Permite diversas fuentes de datos (ASCII, JDBC)
  - Interfaz visual
  - Diferentes herramientas de minería de datos: reglas de asociación, agrupación, clasificación y regresión.
  - Manipulación de datos.
  - Combinación de modelos.

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- **Los archivos que WEKA utiliza son de tipo arff, por ejemplo weather.numeric.arff:**

```
@attribute outlook {sunny, overcast, rainy}
@attribute temperature numeric
@attribute humidity numeric
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,85,85,FALSE,no
sunny,80,90,TRUE,no
overcast,83,86,FALSE,yes
rainy,70,96,FALSE,yes
rainy,68,80,FALSE,yes
rainy,65,70,TRUE,no
overcast,64,65,TRUE,yes
sunny,72,95,FALSE,no
sunny,69,70,FALSE,yes
rainy,75,80,FALSE,yes
sunny,75,70,TRUE,yes
overcast,72,90,TRUE,yes
overcast,81,75,FALSE,yes
rainy,71,91,TRUE,no
```

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- **Los archivos que WEKA utiliza son de tipo arff, por ejemplo weather.nominal.arff:**

```
@attribute outlook {sunny, overcast, rainy}
@attribute temperature {hot, mild, cool}
@attribute humidity {high, normal}
@attribute windy {TRUE, FALSE}
@attribute play {yes, no}

@data
sunny,hot,high,FALSE,no
sunny,hot,high,TRUE,no
overcast,hot,high,FALSE,yes
rainy,mild,high,FALSE,yes
rainy,cool,normal,FALSE,yes
rainy,cool,normal,TRUE,no
overcast,cool,normal,TRUE,yes
sunny,mild,high,FALSE,no
sunny,cool,normal,FALSE,yes
rainy,mild,normal,FALSE,yes
sunny,mild,normal,TRUE,yes
overcast,mild,high,TRUE,yes
overcast,hot,normal,FALSE,yes
rainy,mild,high,TRUE,no
```

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

WEKA posee 4 entornos de trabajo:

- **Simple CLI**: Entorno consola para invocar directamente con java a los paquetes de WEKA
- **Explorer**: Entorno visual que ofrece una interfaz grafica para el uso de los paquetes.
- **Experimenter**: Entorno centrado en la automatización de tareas de manera que se facilite la realización de experimentos a gran escala.
- **KnowledgeFlow**: Permite generar proyectos de minería de datos mediante la generación de flujos de información.

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Primer ejemplo con WEKA:
  - Seleccione el entorno Explorer
  - Luego abra el archivo **weather.nominal.arff** (se encuentra en %wekapath%/data/) desde el botón Open File...
  - Observe que muestra WEKA al abrir el archivo: ¿Qué se observa al hacer click sobre cada atributo?
  - Vaya a la pestaña «Classify» y seleccione el clasificador J48 (se encuentra dentro de trees), con la opción «Use training set» haga click en «Start»
  - Los resultados aparecerán en el marco derecho y si selecciona el nombre del resultado en el marco izquierdo y luego hace click derecho podrá ver el árbol generado entre otras cosas.
  - Realice una investigación con su grupo para lograr explicar del cuadro derecho de resultados el siguiente item: Kappa statistic

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Ejercicio con WEKA: Utilice el archivo drug1n.arff
  - Se trata de predecir el medicamento mas adecuado para un nuevo paciente en función de 6 atributos: **Age** (numerico), **Sex** (nominal), **BP** (Blood Pressure, nominal: alta, baja, normal), **Cholesterol** (nominal: alto, normal), **Na** (nivel de sodio en sangre, numérico) y **K** (nivel de potasio en sangre, numerico). Los datos indican cual ha sido, de entre 5 medicamentos (drugA, drugB, drugC, drugX, drugY) el mejor para cada uno de los pacientes que forman el conjunto de entrenamiento.
    1. **Compruebe precisión y matriz de Confucion para los clasificadores ZeroR y OneR (se encuentran en el grupo Rules)**
    2. **Aplique J48 y compare con los resultados anteriores. Utilice para ello 3 métodos: «training set», «cross validation» y «supplied test set» (para este ultimo use drug2n.arff). Anote los resultados mas significativos: numero de ejemplares clasificados correcta e incorrectamente, precisión y matrices de confusión. Visualice el árbol generado.**
    3. **Intente combinar atributos para mejorar el árbol generado.**

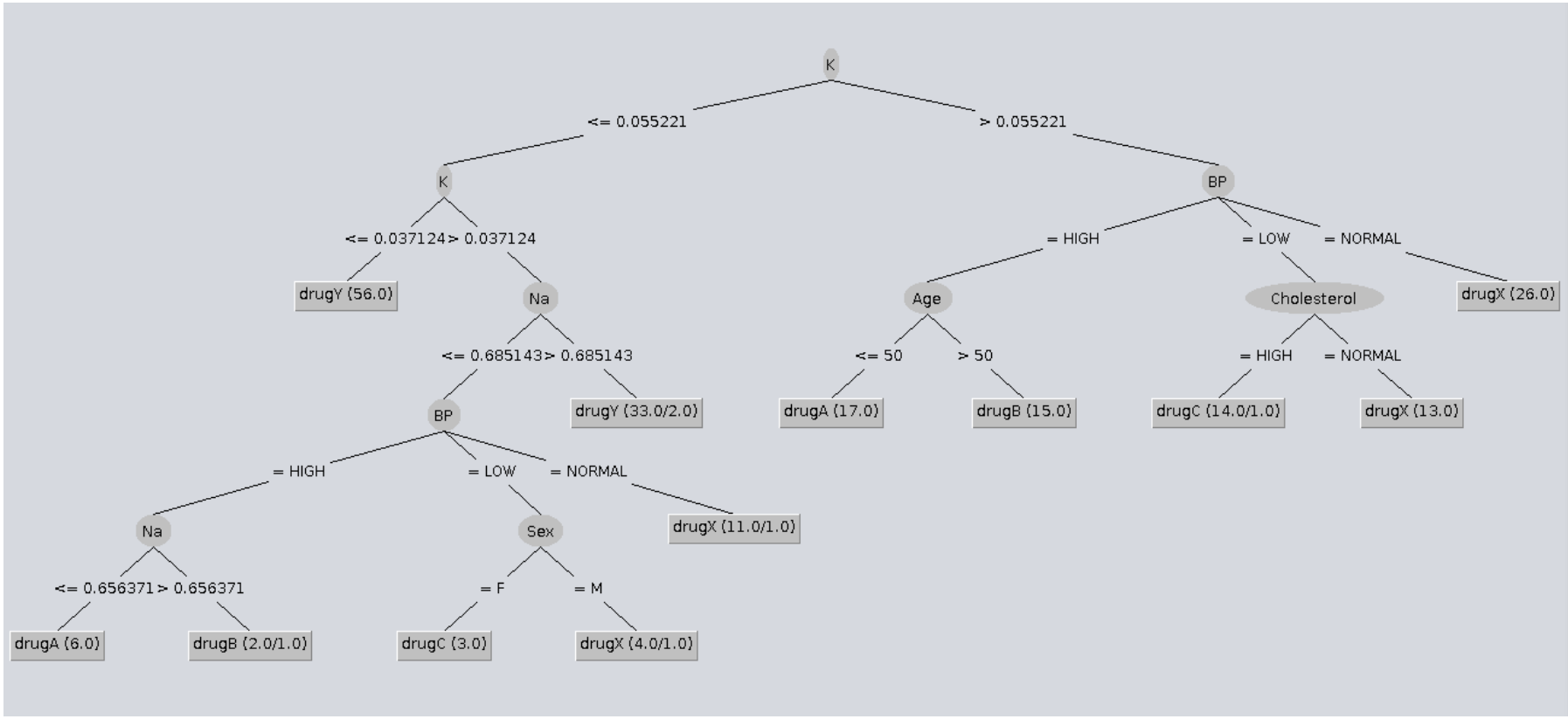
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Salida WEKA:

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     drugA
               1,000    0,005    0,941      1,000    0,970      0,968    1,000     0,996     drugB
               1,000    0,005    0,941      1,000    0,970      0,968    0,998     0,952     drugC
               0,963    0,014    0,963      0,963    0,963      0,949    0,994     0,980     drugX
               0,956    0,018    0,978      0,956    0,967      0,940    0,993     0,987     drugY
Weighted Avg.  0,970    0,013    0,970      0,970    0,970      0,954    0,995     0,985

=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
 23  0  0  0  0 |  a = drugA
  0 16  0  0  0 |  b = drugB
  0  0 16  0  0 |  c = drugC
  0  0  0 52  2 |  d = drugX
  0  1  1  2 87 |  e = drugY
```

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Salida WEKA:



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Ejemplos colab:
  - Weka:
    - https://colab.research.google.com/drive/1x3gZqA7TeYLKhN8ugm0aHYylb2BpImIj?usp=sharing
    - https://colab.research.google.com/drive/1pL0DkfUe4Rom_3_B4RAven9CUXtfI9ik?usp=sharing
  - Python:
    - https://colab.research.google.com/drive/1T7igMVYG_84buPFomXgBxH8Vgl6NmbAg?usp=sharing
    - https://colab.research.google.com/drive/1wNe4Tx8XRCl0uvY8nAzVrY5KRSQS2tiy?usp=sharing

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Matriz de confusión:

Es una herramienta que permite la visualización del desempeño de un algoritmo que se emplea en aprendizaje supervisado.

		PREDICCIÓN	
		POSITIVO	NEGATIVO
CLASE REAL	POSITIVO	True Positives (TP)	False Negatives (FN)
	NEGATIVO	False Positives (FP)	True Negatives (TN)

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Exactitud (accuracy):

Mide el porcentaje de predicciones correctas realizadas por el modelo frente al total.

Es decir es el cociente entre los casos bien clasificados por el modelo y la suma de todos los casos.

En conjuntos de datos poco equilibrados, no es una métrica muy útil. Por ejemplo, si lo que intentamos predecir es una enfermedad rara, y nuestro algoritmo clasifica a todos los individuos como sanos, podría ser muy preciso (incluso un 99%), pero también, totalmente inútil.

Exactitud = (TP + TN) / (TP+FN+FP+TN)

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- True Positive Rate (TPR), Recall o Sensitivity o Cobertura:

Predicciones positivas correctas entre el nro total de positivos. En la matriz de confusión, es el valor del elemento de la diagonal dividido por la suma de la fila relevante.

Lo podemos ver como el porcentaje de predicciones positivas bien clasificadas por el modelo respecto al total de positivos o dicho de otra manera **es la capacidad del modelo de detectar casos positivos o relevantes.**

$$TPR = TP / (TP + FN)$$

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- True Negative Rate (TNR), Specify o Especificidad:

Es el porcentaje dado por la proporción entre los casos negativos bien clasificados por el modelo respecto del total de negativos.

Podemos verlo como la capacidad del modelo de discriminar correctamente los casos negativos.

$$TNR = TN / (TN + FP)$$

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- False Positive Rate (FPR): Predicciones positivas incorrectas entre nro total de negativos.
- **Podemos entenderlo como el indicador que nos dice cual es la probabilidad de que sea una falsa alarma.**

- **FPR = FP / (FP + TN)**

En la matriz de confusión, de más de dos dimensiones, es la suma de la columna menos el valor del elemento de la diagonal dividido por la suma de las filas de las otras clases.

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- False Omission Rate (FOR): Predicciones negativas incorrectas entre nro total de predicciones negativas.
- Podemos verlo como el porcentaje que indica que tan fuerte es la presencia de predicciones negativas sobre el total de predicciones negativas.

  - **FOR = FN / (FN + TN)**

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Precisión o Precision:

Mide el porcentaje de predicciones positivas correctas.

Podemos entenderlo como el indicador que nos dice que tan cerca esta una predicción del valor verdadero.

$$TP = TP / (TP + FP)$$

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- F-measure:

Permite caracterizar con único valor la bondad de un clasificador o algoritmo. La formula de esta medida está establecida como:

F-measure=2*Precision*Cobertura/(Precision+ Cobertura)

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Receiver Operating characteristic Curve (Curva ROC) (Curva de Características Operativas del Receptor):

En problemas complejos un clasificador aumentará el número de True Positives (TP) a costa de incrementar también el de False Positives (FP).

Se busca entonces un clasificador que sea capaz de incrementar TP a un ritmo (mucho) mayor que FP.

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Receiver Operating characteristic Curve (Curva ROC) :

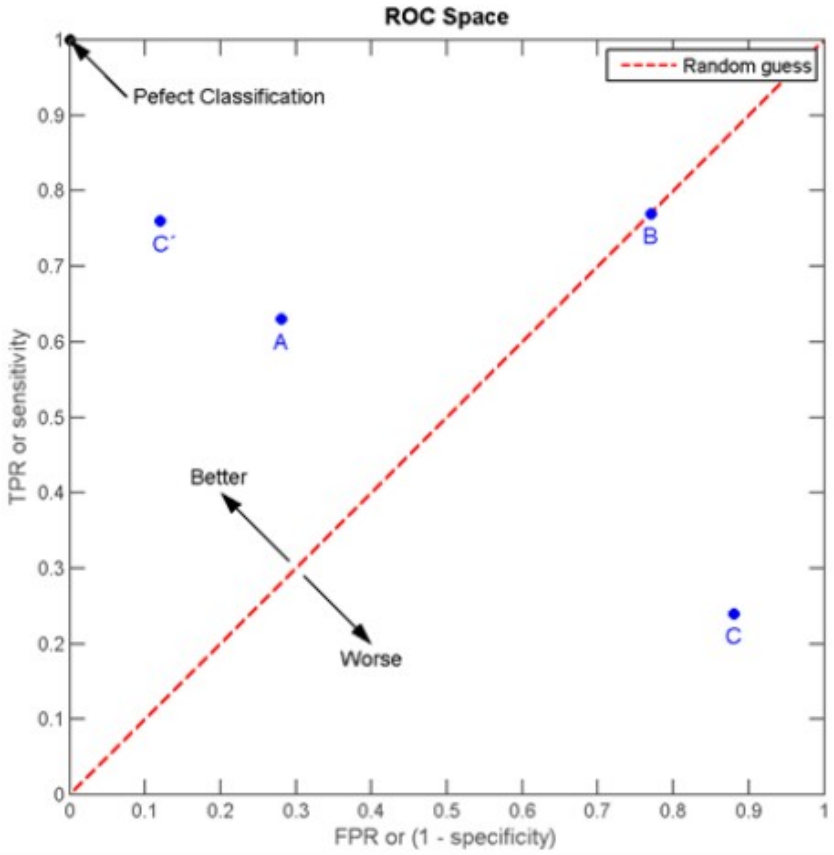
Los gráficos ROC son gráficos bidimensionales en los cuales se representa FPR (False Positive Rate) en el eje X y TPR (True Positive Rate) en el eje Y.

Un gráfico ROC muestra el compromiso entre beneficio (True Positives) y coste (False Positives).

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Receiver Operating characteristic Curve (Curva ROC) :

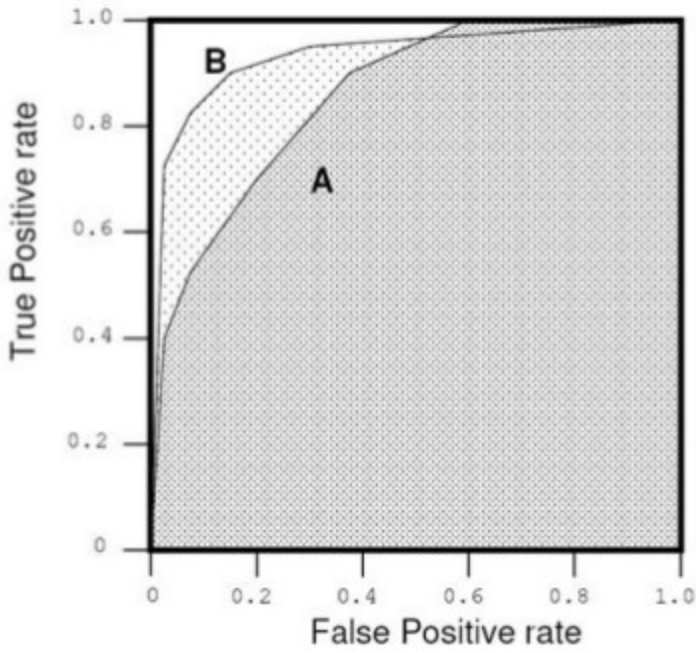


APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Area Under Curve (AUC) o ROC Area:

El área bajo la curva ROC (AUC) permite representar en un único valor el rendimiento del clasificador. Esto puede resultar útil para realizar comparativas entre clasificadores



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- MCC (Coeficiente de correlacion de Matthews):

Informa la calidad de clasificación de las clases. Se tiene en cuenta los valores TP, FP y FN, es considerada como una medida equilibrada.

$$MCC = \frac{(TP * TN) + (FP * FN)}{\sqrt{(FP + TP)(TP + FN)(TN + FP)(TN + FN)}}$$

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Evaluación de clasificadores

- Area PRC (Precision Vs Recall):

Este valor dice como es el comportamiento de cada clase y para esto utiliza precisión vs recall, de igual manera que la curva ROC entre más cercano a 1 mejor el comportamiento. Indica en que porcentaje los clasificadores se comportan de manera correcta.

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Salida WEKA:

```
=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               1,000    0,000    1,000      1,000    1,000      1,000    1,000     1,000     drugA
               1,000    0,005    0,941      1,000    0,970      0,968    1,000     0,996     drugB
               1,000    0,005    0,941      1,000    0,970      0,968    0,998     0,952     drugC
               0,963    0,014    0,963      0,963    0,963      0,949    0,994     0,980     drugX
               0,956    0,018    0,978      0,956    0,967      0,940    0,993     0,987     drugY
Weighted Avg.  0,970    0,013    0,970      0,970    0,970      0,954    0,995     0,985

=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
 23  0  0  0  0 |  a = drugA
  0 16  0  0  0 |  b = drugB
  0  0 16  0  0 |  c = drugC
  0  0  0 52  2 |  d = drugX
  0  1  1  2 87 |  e = drugY
```

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

Algoritmo ID3 devuelve un Árbol de decisión

- R = conjunto de atributos no clasificadores
- C = atributo clasificador
- S = conjunto de entrenamiento
- Inicio
  - Si S esta vacío:
    - devolver un único nodo con valor falla;
  - Si todos los registros de S tienen el mismo valor para el atributo clasificador:
    - devolver un único nodo con dicho valor;
  - Si R esta vacío:
    - devolver un único nodo con el valor mas frecuente del atributo clasificador en los registros de S;
  - Si R no esta vacío:
    - D ← atributo con mayor ganancia entre los atributos de R;
    - Sean {dj} (j=1,2,…,m) los valores del atributo D
    - Sean {Sj} (j=1,2,…,m) los subconjuntos correspondientes a los valores de dj respectivamente;
    - devolver un árbol con la raíz nombrada como D y los arcos nombrados como d1,d2,…,dm
    - que van respectivamente a los arboles ID3(R-{D}, C, S1), ID3(R-{D}, C, S2),…, ID3(R-{D}, C, Sm);

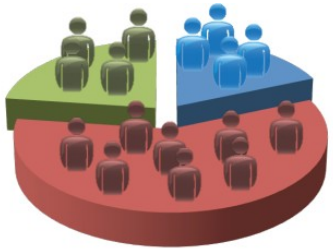
  Fin

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

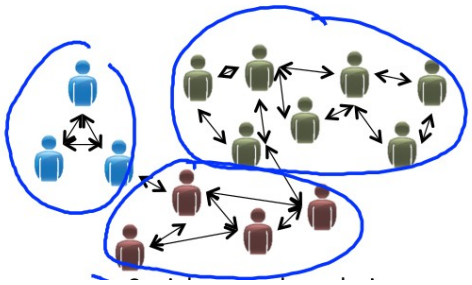
- Como se calcula la ganancia de la información de la tabla para construir
- nuestro árbol:
  - Ver documento adjunto: Calculo de la informacion.pdf

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

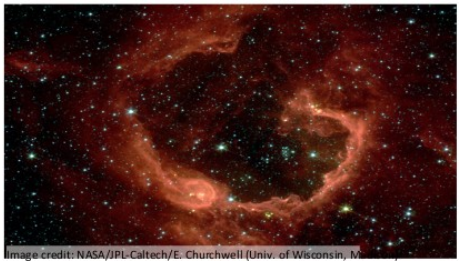
- Clustering con WEKA: (Aprendizaje no supervisado)
  - Aplicaciones:
    - Segmentación de mercado:



    - Análisis de redes sociales:



    - Análisis de datos astronómicos:



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

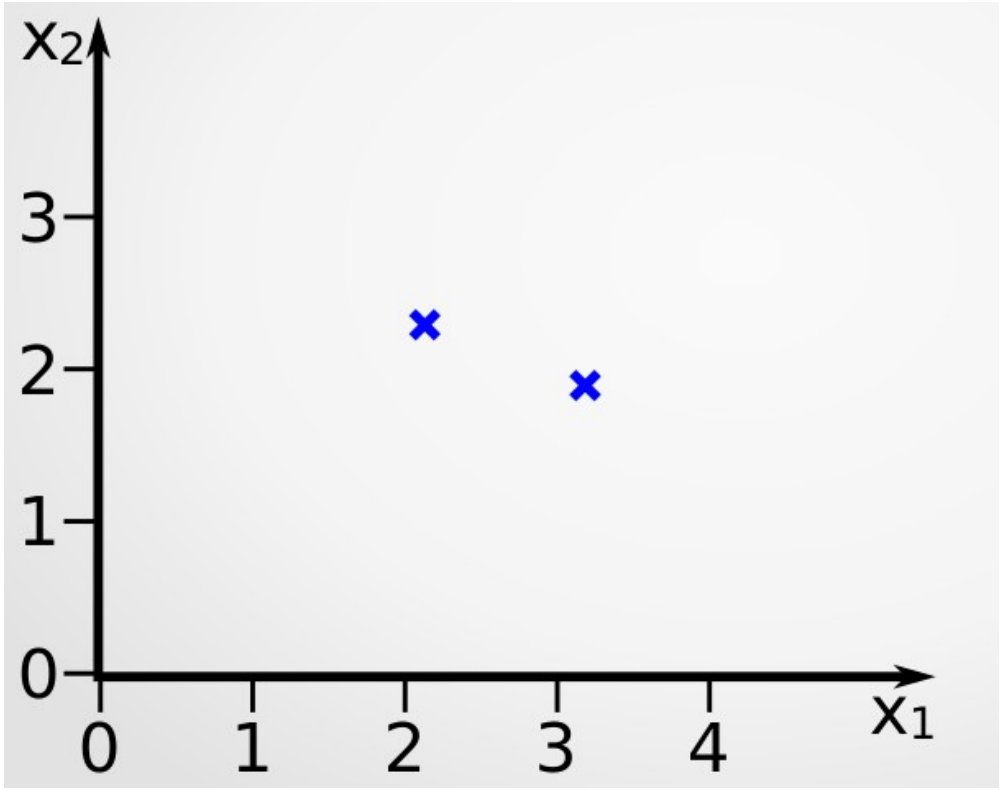
- Clustering con WEKA: (Aprendizaje no supervisado)
  - En el Explorer lo encontramos en la pestaña “Cluster”.
  - El proceso es similar al ya visto, debemos abrir un dataset y luego ingresar a la pestaña mencionada.
  - Dentro de la pestaña “Cluster” encontraremos los diferentes algoritmos de clasificación, los mas relevantes para nosotros seran:
    - Kmeans
    - Coweb
    - EM

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - Agrupa objetos en *k* grupos basándose en sus características.
  - El agrupamiento se realiza minimizando la suma de distancias entre cada objeto y el centroide de su grupo o cluster.
  - Solo datos numéricos.
  - Se suele usar la distancia cuadrática.
  - Requiere de normalización.
  - Admite ruido.
  - Agrupaciones fijas y disjuntas.
  - **Depende mucho de las semillas.**

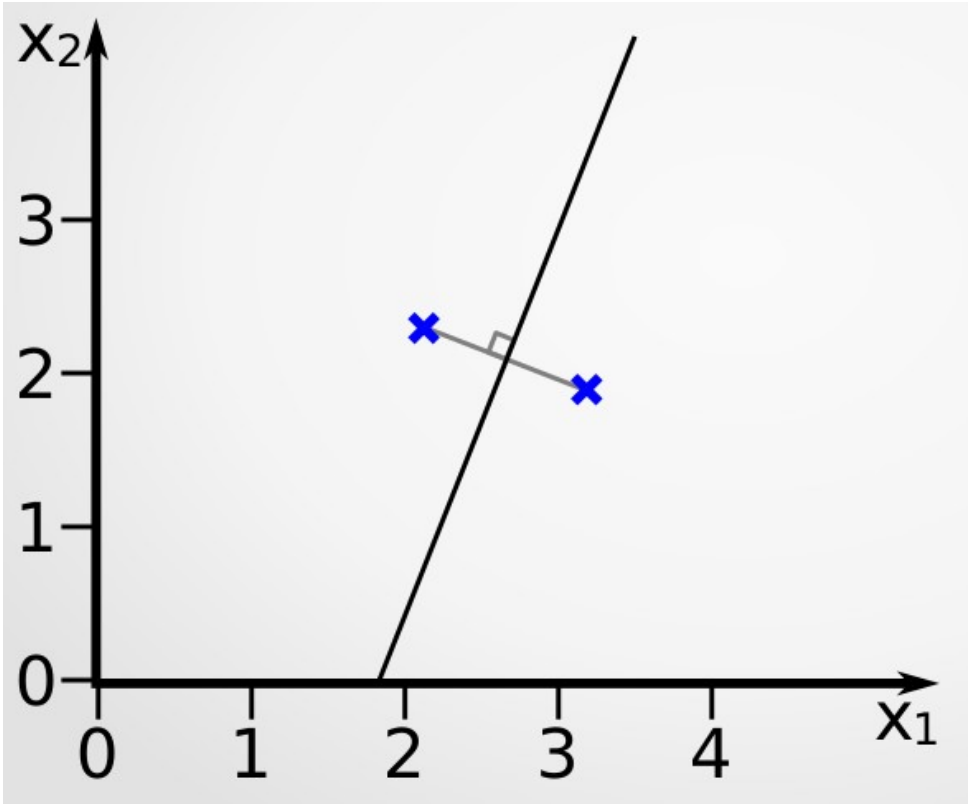
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - Basado en los grafos de VORONOI (2 semillas)



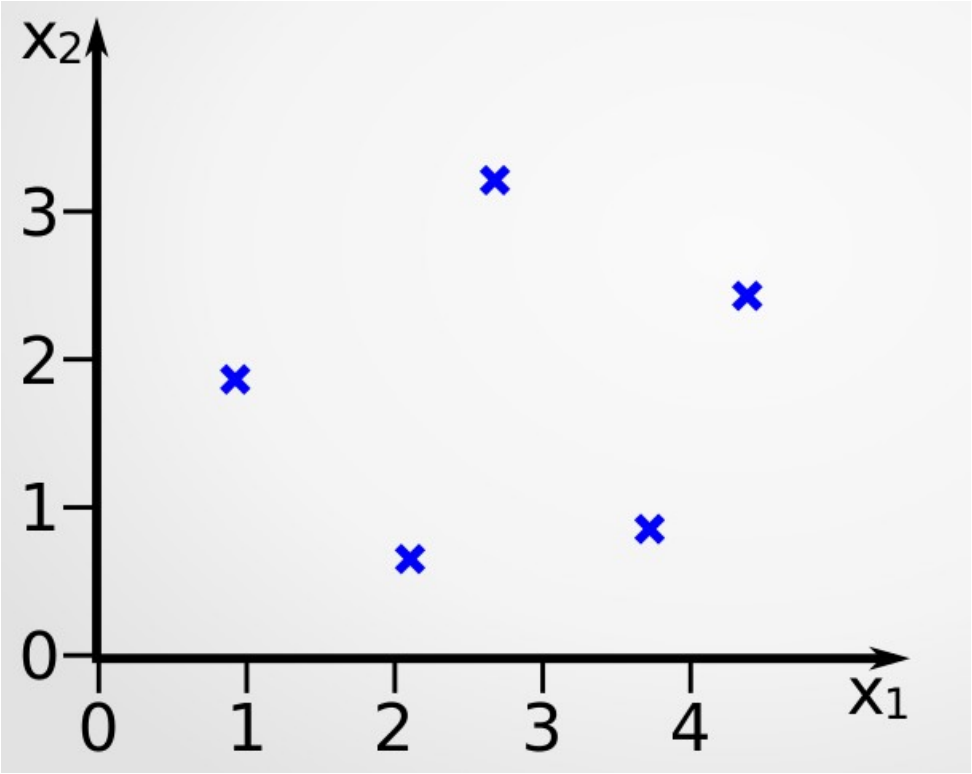
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - Basado en los grafos de VORONOI (2 semillas)



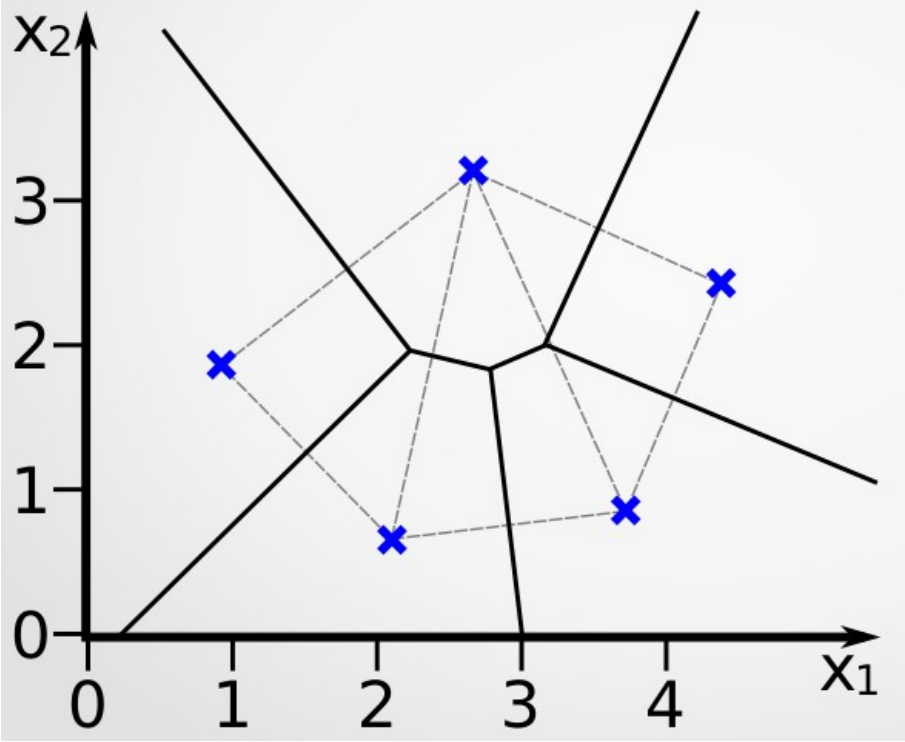
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - Basado en los grafos de VORONOI (mas semillas)



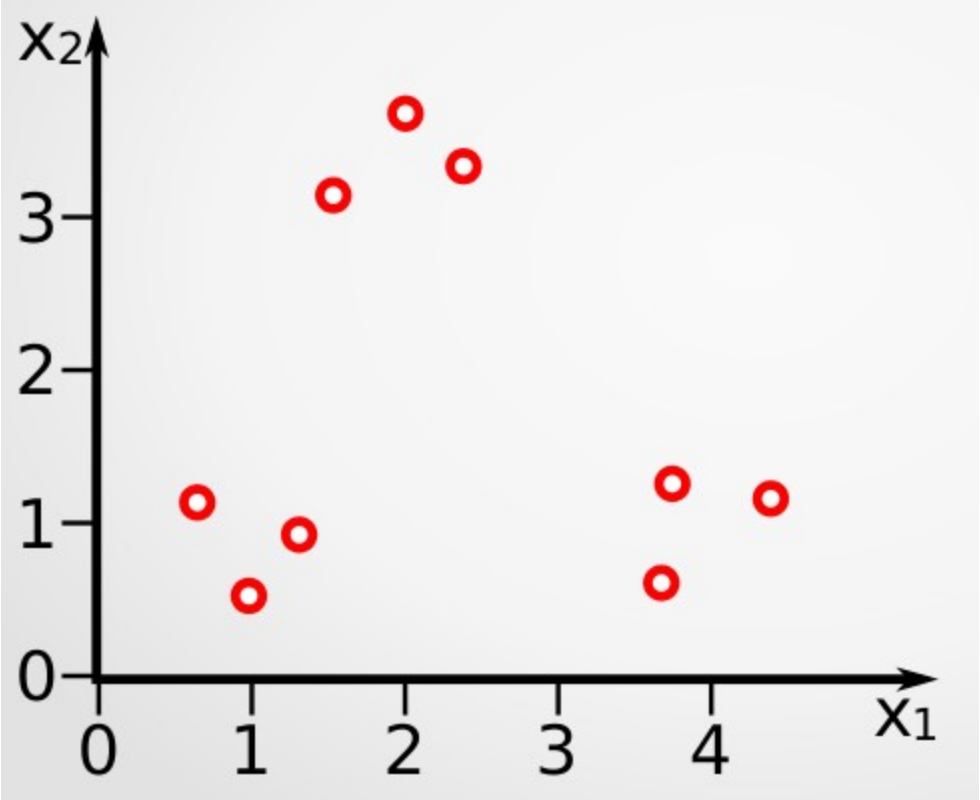
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - Basado en los grafos de VORONOI (mas semillas)



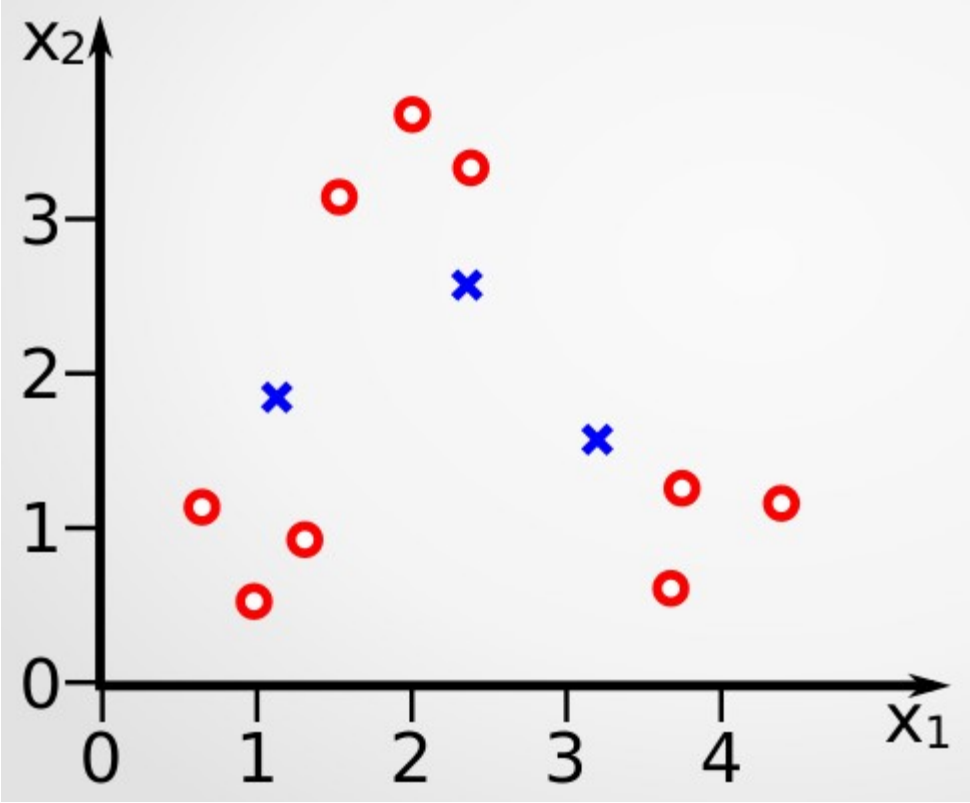
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
- Comenzando con los datos



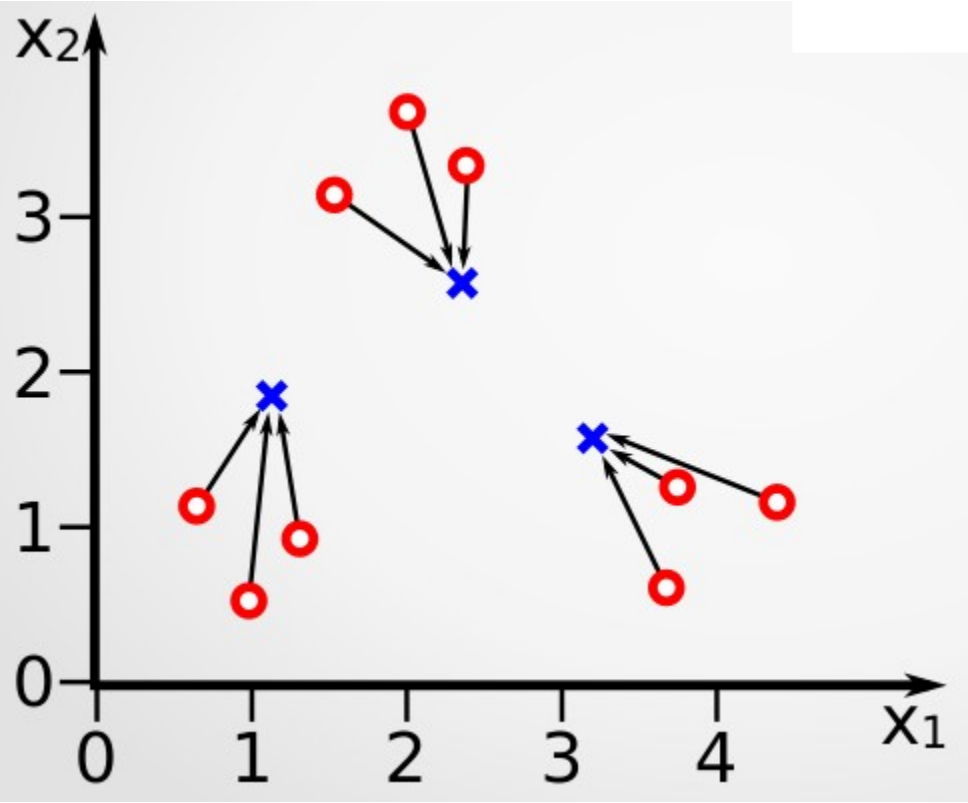
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
- Plantamos las semillas



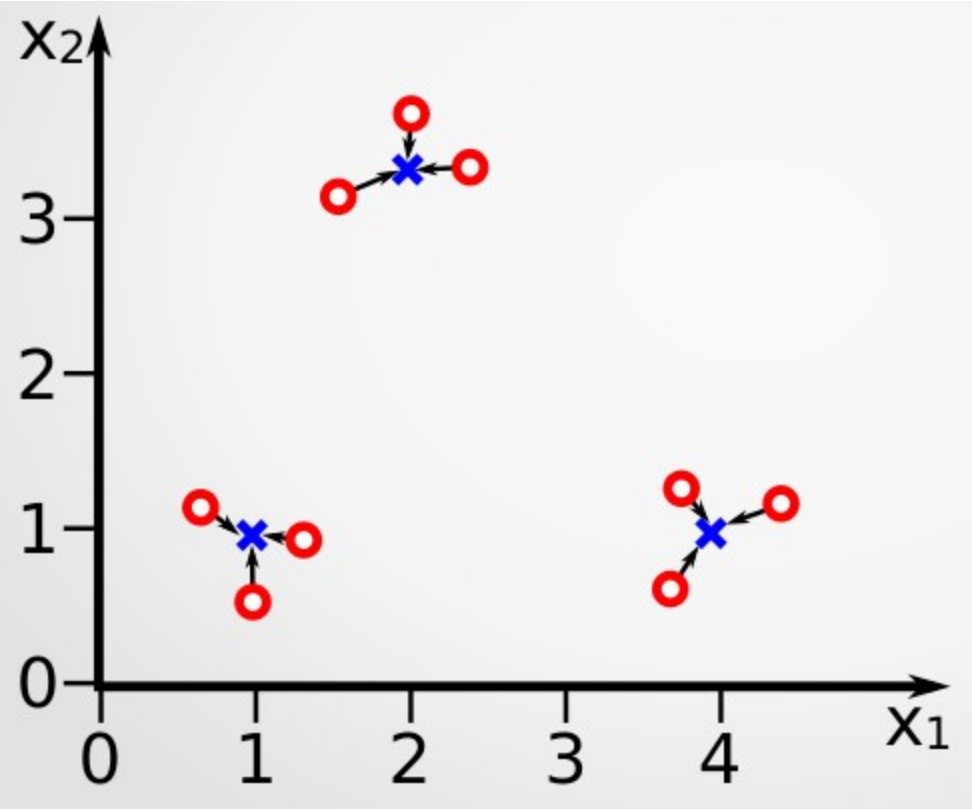
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
- Asignamos cada ejemplo a la semilla mas cercana



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

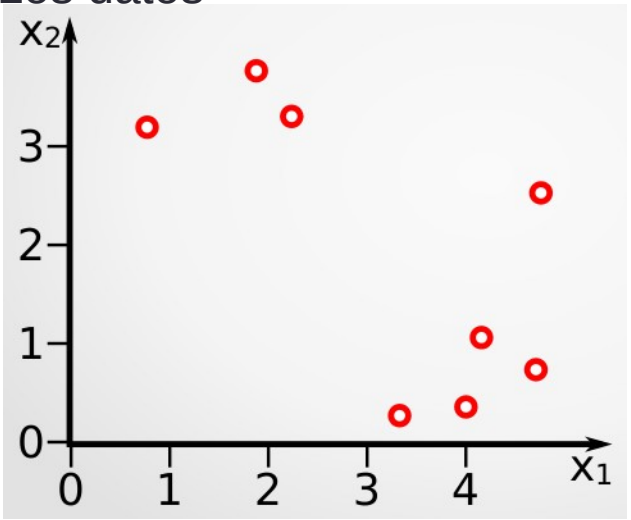
- Kmeans:
- Desplazamos las semillas a los centros de cada grupo



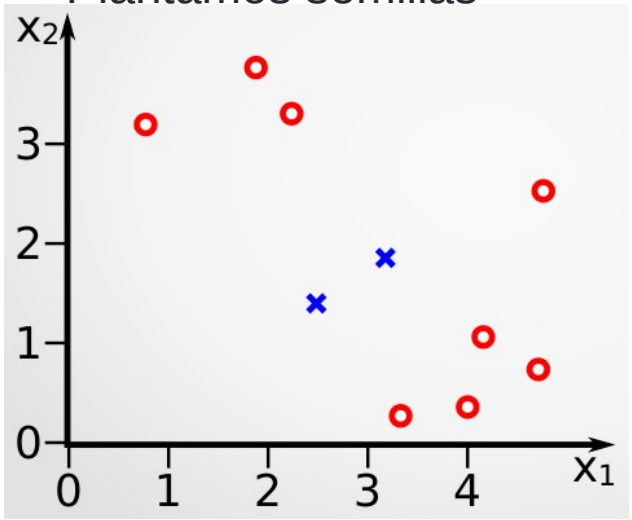
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
- Ejemplo paso a paso

Los datos



Plantamos semillas



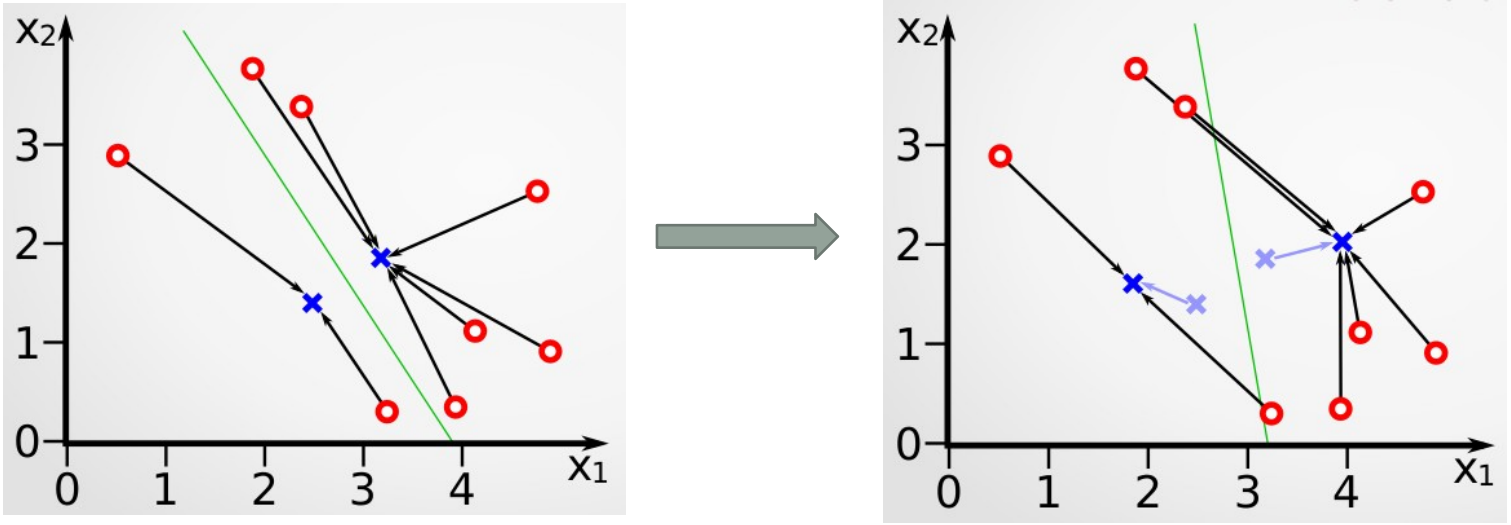
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
- Ejemplo paso a paso

  Asignamos cada ejemplo a

  Cada semilla

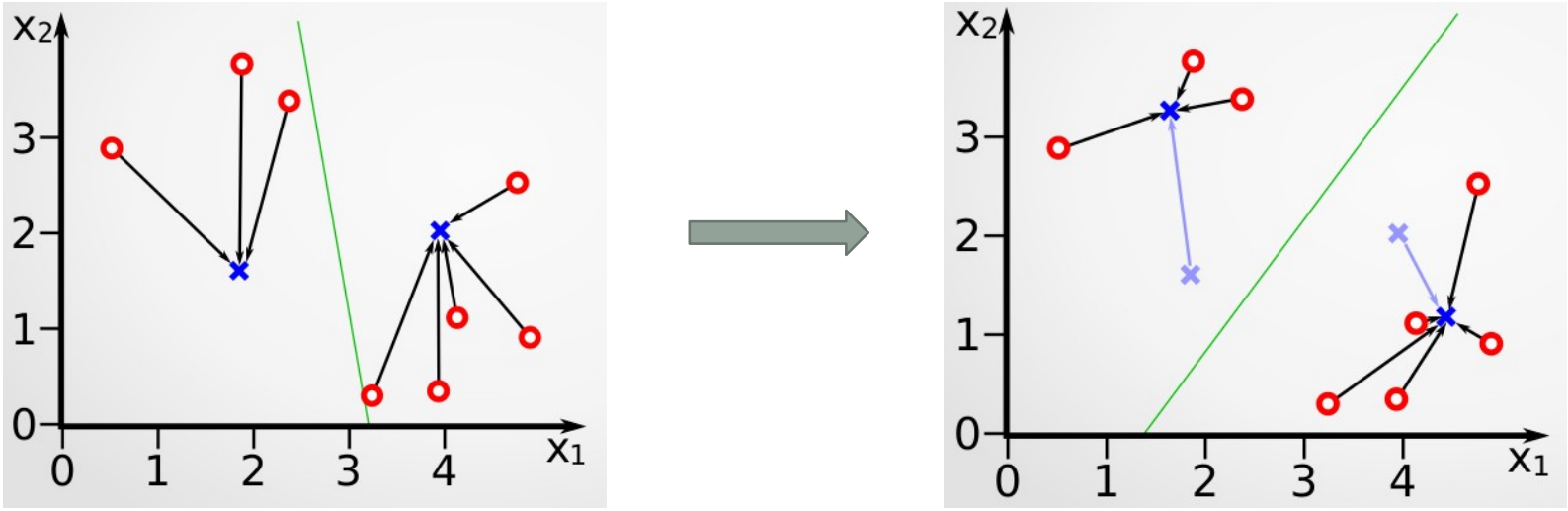
Desplazamos las semillas
al centro de cada grupo



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
- Ejemplo paso a paso
  - Asignamos cada ejemplo a
  - Cada semilla

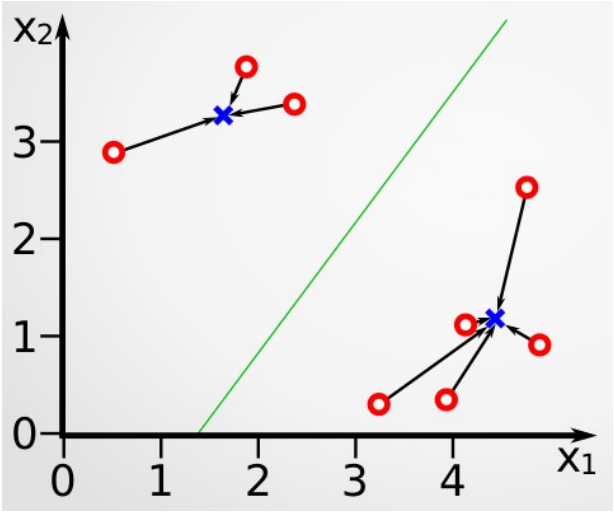
Desplazamos las semillas
al centro de cada grupo



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

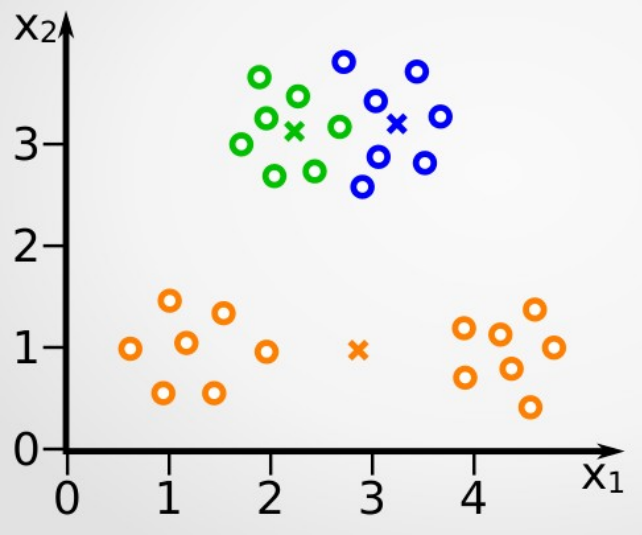
- Kmeans:
- Ejemplo paso a paso

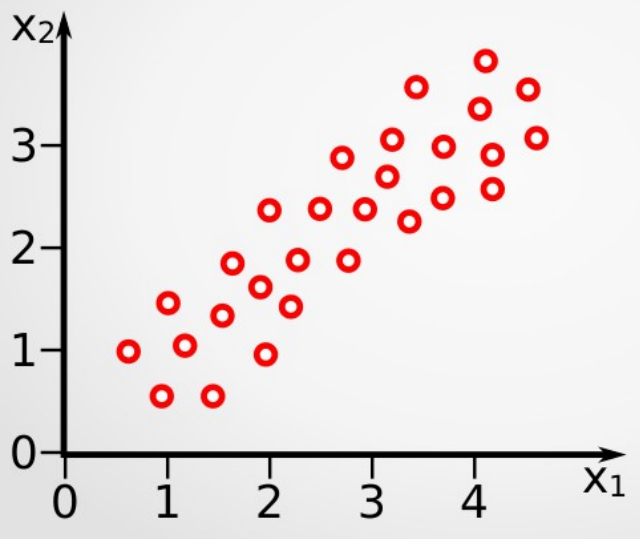
  Nos detenemos cuando ya no se mueven los centroides o cuando ya no hay cambios en la asignación de cada ejemplo a cada centroide (semilla)



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
- Problemas





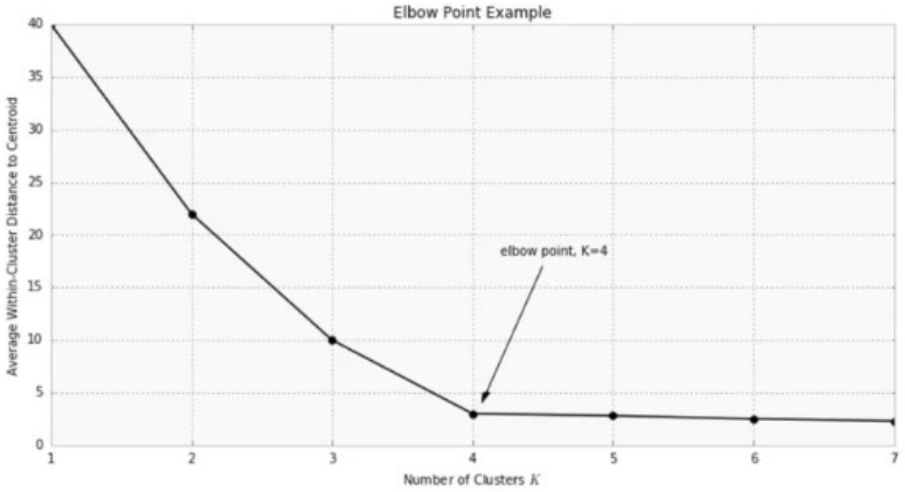
APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - Los objetos se representan con vectores reales de **d** dimensiones ($\mathbf{x}_1$,$\mathbf{x}_2$, …,$\mathbf{x}_n$) y el algoritmo *k-means* construye **k** grupos donde se **minimiza la suma de distancias de los objetos**, dentro de cada grupo **S**={S_1,S_2, …,S_k}, a su centroide.
  - El problema se puede formular de la siguiente forma:

$$\min_{\mathbf{S}} E(\boldsymbol{\mu}_i) = \min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|^2$$

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - La cantidad de cluster (k) se determina usando el método del codo:
    - Utiliza la distancia media de las observaciones a su centroide.
    - Cuanto **más grande** es el número de clusters **k**, la **varianza intra-cluster tiende a disminuir**.
    - Cuanto menor es la distancia intra-cluster mejor, ya que significa que los clústers son más compactos.
    - **Busca el valor k que satisfaga que un incremento de k, no mejore sustancialmente la distancia media intra-cluster.**



APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Kmeans:
  - Ejemplos:
    - Weka:
      - https://colab.research.google.com/drive/1VLlvLkvkoU0M94u2xShppazUKmZY3hB9?usp=sharing
    - Python:
      - https://colab.research.google.com/drive/1N_oOfKsR6UU8zUfarfJ9fJSvgnKgLyvC?usp=sharing
    - Octave:
      - https://drive.google.com/file/d/1L2tc6xUF4TWMqzPEcui4HyydKdu8SI4s/view?usp=sharing

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

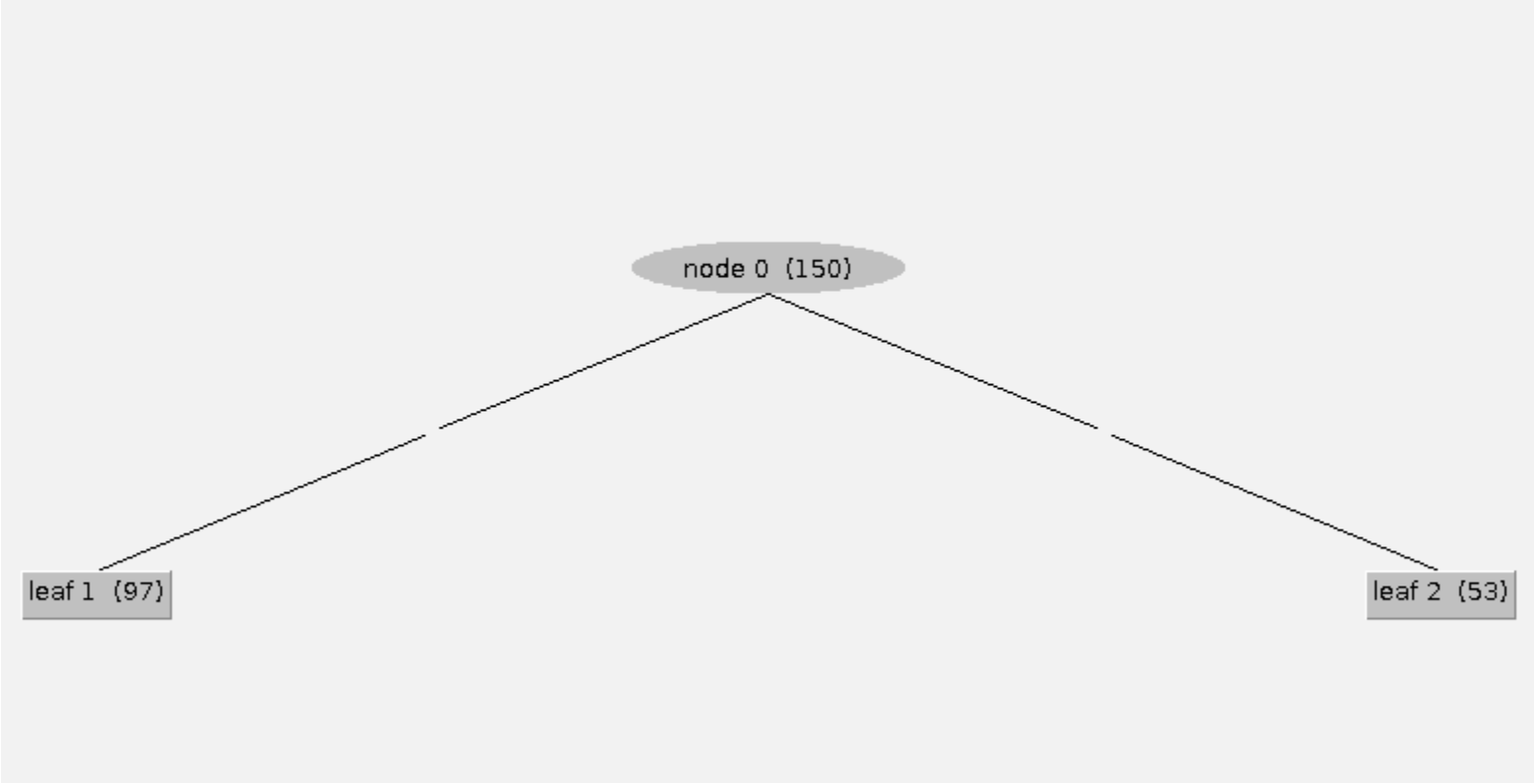
- Coweb:
  - Es un algoritmo de clustering jerárquico.
  - Se caracteriza porque utiliza aprendizaje incremental, esto es, realiza las agrupaciones instancia a instancia.
  - Utiliza la medida llamada **category utility** para guiar el proceso de aprendizaje y determinar la pertenencia de cada instancia a un grupo. **(Esta medida obtiene valores altos para aquellas agrupaciones que presentan una alta similitud entre los elementos de un mismo grupo y una baja similitud entre objetos de grupos diferentes)**
  - Durante la ejecución del algoritmo se forma un árbol (árbol de clasificación) donde las hojas representan los clusters y el nodo raíz engloba por completo el conjunto de datos de entrada:
    - Al principio, el árbol consiste en un único nodo raíz.
    - Las instancias se van añadiendo una a una y el árbol se va actualizando en cada paso.

APRENDIZAJE AUTOMATICO PUESTO EN PRACTICA

- Coweb:
  - La **actualización** consiste en encontrar el mejor sitio donde incluir la nueva instancia, **operación que puede necesitar de la reestructuración de todo el árbol** (incluyendo la generación de un nuevo nodo anfitrión para la instancia y/o la fusión / partición de nodos existentes) o simplemente la inclusión de la instancia en un nodo que ya existía.
  - El algoritmo es muy sensible a dos parámetros:
    - **Acuity:** representa la medida de error de un nodo. En consecuencia, permite controlar el factor de ramificación.
    - **Cut-off:** Este valor se utiliza para evitar el crecimiento desmesurado del numero de clusters.

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- Coweb:
  - Ejemplo usando Dataset iris.arff:

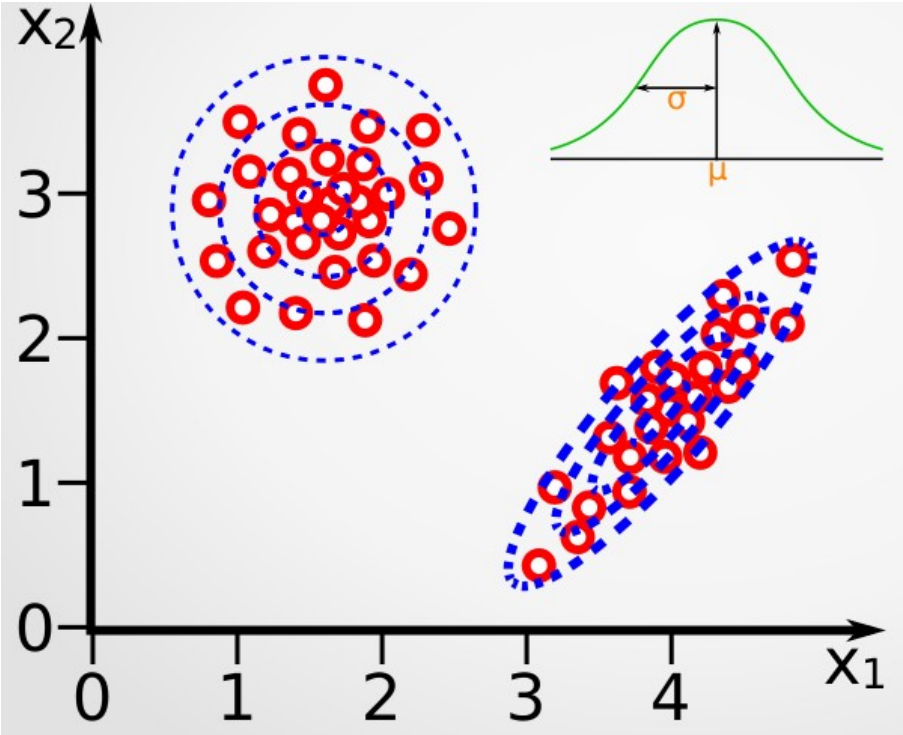


APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- EM:
  - Mejora k-Medias con gaussianas.
  - Permite un nro. de clases NO fija.
  - Agrupaciones NO disjuntas.
  - Calcula semillas con estadística.
  - Sólo valores numéricos.
  - Requiere normalización

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

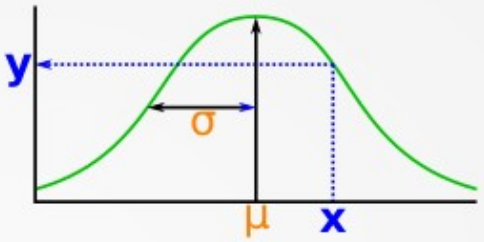
- EM: Distribución Gaussiana



APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- EM: Distribución Gaussiana (Cálculos)

$$f(x;\mu,\sigma^2)=\frac{1}{\sqrt{2\pi\sigma^2}}\cdot\exp\left(\frac{-1}{2}\frac{(x-\mu)^2}{\sigma^2}\right)$$



$$\mu=\frac{1}{N}\sum_{i=1}^{N}x_i \quad , \quad \sigma^2=\frac{1}{N}\sum_{i=1}^{N}(x_i-\mu)^2$$

$$f(\bar{x};\bar{\mu},\bar{\Sigma})=(2\pi)^{\frac{-N}{2}}|\bar{\Sigma}|^{\frac{-1}{2}}\exp\left(\frac{-1}{2}(\bar{x}-\bar{\mu})^T\bar{\Sigma}^{-1}(\bar{x}-\bar{\mu})\right)$$

$\mu \equiv$ media

$\sigma \equiv$ desviación típica/estándar

$\sigma^2 \equiv$ varianza

$\bar{\Sigma} \equiv$ matriz de covarianza

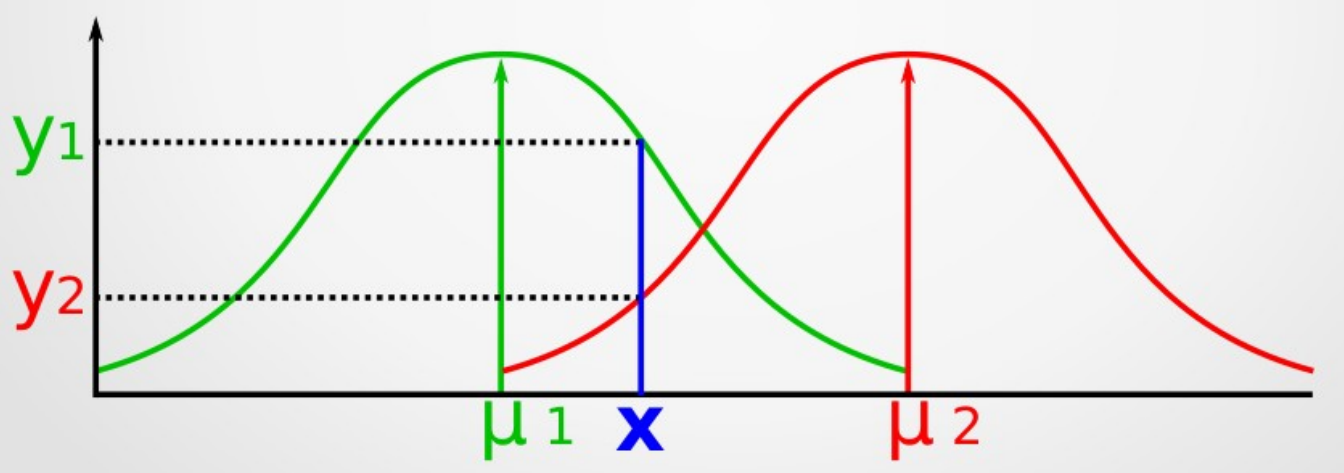
$$\bar{x}=\begin{pmatrix}1.0 & 1.9\\ 2.0 & 3.1\\ 3.0 & 4.0\\ 4.0 & 5.1\\ 5.0 & 5.9\end{pmatrix} \qquad \bar{\mu}=(3.0 \quad 4.0)$$

$$\bar{\Sigma}=\begin{pmatrix}2.50 & 2.50\\ 2.50 & 2.51\end{pmatrix}$$

$$\bar{\Sigma}^{-1}=\begin{pmatrix}100.4 & -100.0\\ -100.0 & 100.0\end{pmatrix}$$

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- EM:
  - Por cada ejemplo, se calcula la probabilidad de pertenecer a cada una de las gaussianas:
    - Donde μ es el centroide de la gaussiana.



APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- EM: Pasos
  - El algoritmo termina cuando las medias o varianzas son muy similares al paso anterior.

- Paso E: recalcula pesos de ejemplos:

$$P(\bar{x}_i)=\sum_{k=1}^{K} P(K=k,\bar{x}_i)=\sum_{k=0}^{K} P(K=k)\cdot P(\bar{x}_i|K=k)$$

$$P(K=k)=w_k \quad (\text{peso}\equiv\text{probabilidad})$$

$$P(\bar{x}_i|K=k)=(2\pi)^{\frac{-N}{2}}|\bar{\Sigma}|^{\frac{-1}{2}}\exp\left(\frac{-1}{2}(\bar{x}_i-\bar{\mu}_k)^T\Sigma^{-1}(\bar{x}_i-\bar{\mu}_k)\right)$$

$$p_{ik}=w_k\cdot(2\pi)^{\frac{-N}{2}}|\bar{\Sigma}|^{\frac{-1}{2}}\exp\left(\frac{-1}{2}(\bar{x}_i-\bar{\mu}_k)^T\Sigma^{-1}(\bar{x}_i-\bar{\mu}_k)\right)$$

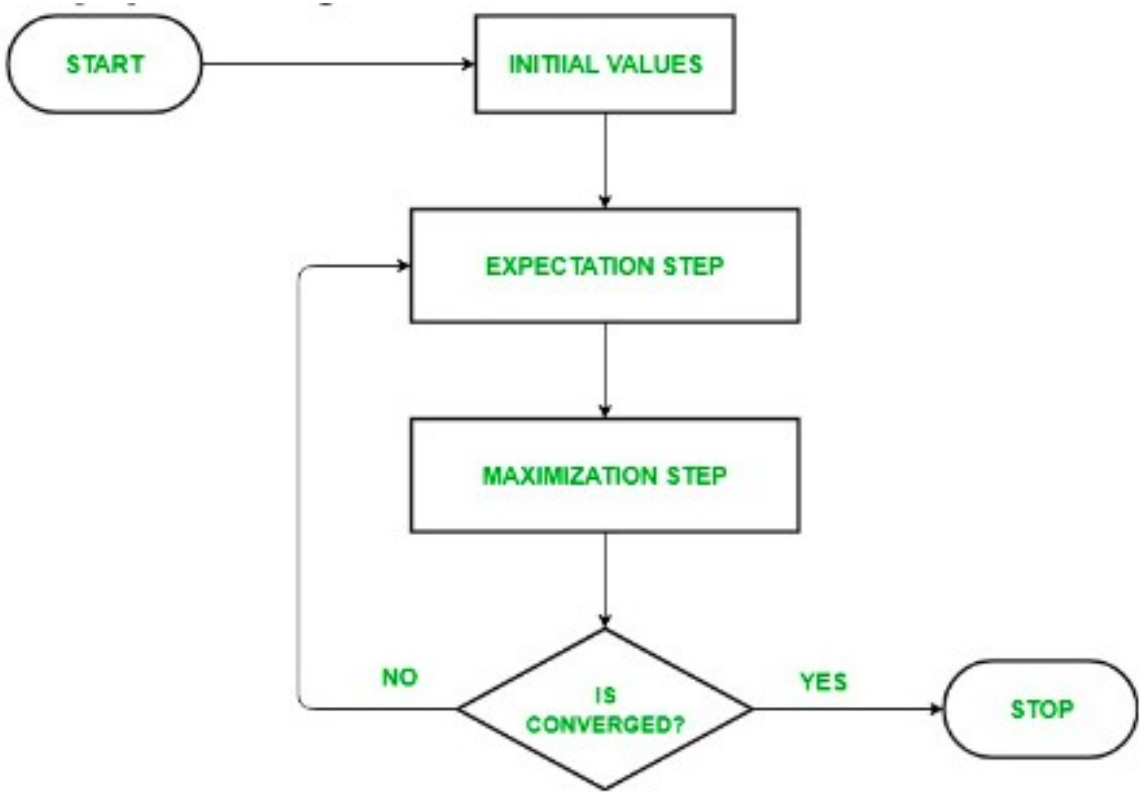
- Paso M: recalcula pesos:

$$n_k=\sum_{i=1}^{N} p_{ik} \rightarrow w_j\leftarrow\frac{n_k}{N}\ ,\ \bar{\mu}_k\leftarrow\frac{1}{n_k}\cdot\sum_{i=1}^{N} p_{ik}\cdot\bar{x}_i$$

$$\bar{\Sigma}_k\leftarrow\frac{1}{n_k}\cdot\sum_{i=1}^{N} p_{ik}\cdot(\bar{x}_i-\bar{\mu}_k)^T(\bar{x}_i-\bar{\mu}_k)$$

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- EM: Pasos



APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- EM:
  - ¿Cómo se ajusta el numero de clases?
    - Primera opción:
      - Se crean k clases iniciales.
      - Se lanza algoritmo EM.
      - Se eliminan clases según: $-\sum_{i=1}^{N} \log P(\bar{x}_i) + coste \cdot k$
      - Si el resultado de la formula da un valor alto (comparado con las demás clases) entonces me quedo con la clase.
      - Si no es necesario quitar ninguna, se termina.

    - Segunda opción:
      - Se crea no aleatorio de nuevas clases.
      - Si varianza cerca de nula: se quita clase.
      - Si medias y varianzas muy similares: se quita una de ellas.

El coste es un parámetro mas que depende de los datos (del problema analizado)

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- EM:
  - https://drive.google.com/file/d/1E8Fl_jhXa4-8KMwAVvhcwgELTn0Brm_m/view?usp=drive_link

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- Recomendaciones: Normalización y Escalado
  - Normalización: A la hora de usar cualquier algoritmo de clustering, es buena idea normalizar nuestros datos.
    - Normalizar: se refiere a que los valores de cada atributo estén en escalas similares.
  - Normalizar ayuda al clustering porque, recordemos, los grupos se forman a partir de distancias.
  - Si hay atributos con escalas muy diferentes, los atributos de escala mayor dominarán las distancias.
  - El **escalado** va a **transformar los valores de las características de forma que estén confinados en un rango [a, b]**, típicamente [0, 1] o [-1, 1].
  - La **normalización** va a **transformar las características de forma que todas compartan un mismo valor medio y una misma desviación media**.

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- Recomendaciones: Normalización y Escalado

Escalado de variables (Feature Scaling o MinMax Scaler)

En este caso, cada entrada se normaliza entre unos límites definidos:

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

- No es un método adecuado para datos estables.

APRENDIZAJE AUTOMÁTICO PUESTO EN PRACTICA

- Recomendaciones: Normalización y Escalado

Escalado estándar (Standard Scaler)

Una alternativa al escalado de variables es usar otra técnica conocida como *escalado estándar* (a cada dato se le resta la media de la variable y se le divide por la desviación típica).

$$X_{normalized} = \frac{X - X_{mean}}{X_{stddev}}$$

- Metodo no adecuado para cuando los datos presentas variaciones como picos

APRENDIZAJE AUTOMÁTICO: Utilidades

- Links útiles:
  - https://mvnrepository.com/artifact/nz.ac.waikato.cms.weka/weka-stable
  - https://archive.ics.uci.edu/ml/datasets.php
  - https://datahub.io/sports-data/spanish-la-liga#resource-season-1819
  - https://sourceforge.net/projects/meka/files/Datasets/
  - https://www.cs.waikato.ac.nz/~ml/weka/downloading.html
  - https://secondport.notion.site/Programmatic-SEO-Datasets-a95a2bffd60d4956b2e8e03187f43266?pvs=4