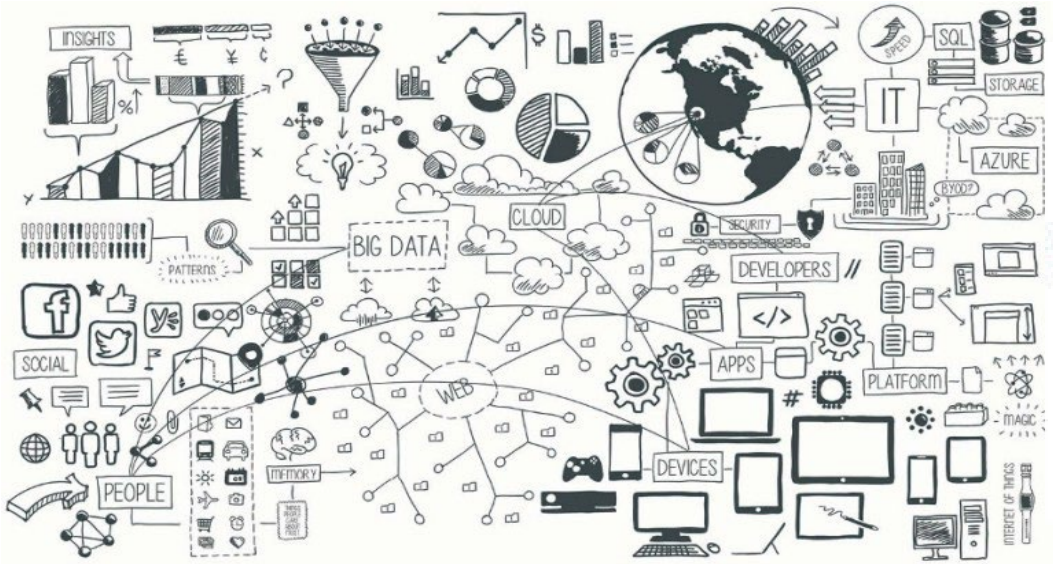
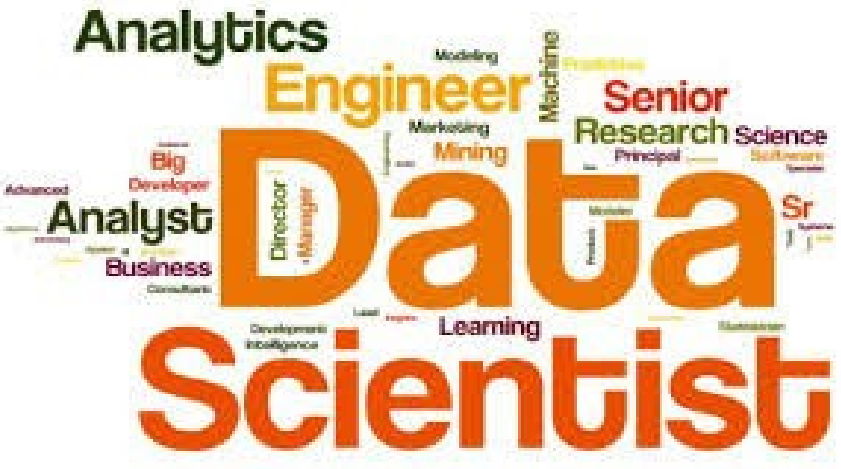
Data Mining (Minería de Datos)

Association Rules





Sixto Herrera

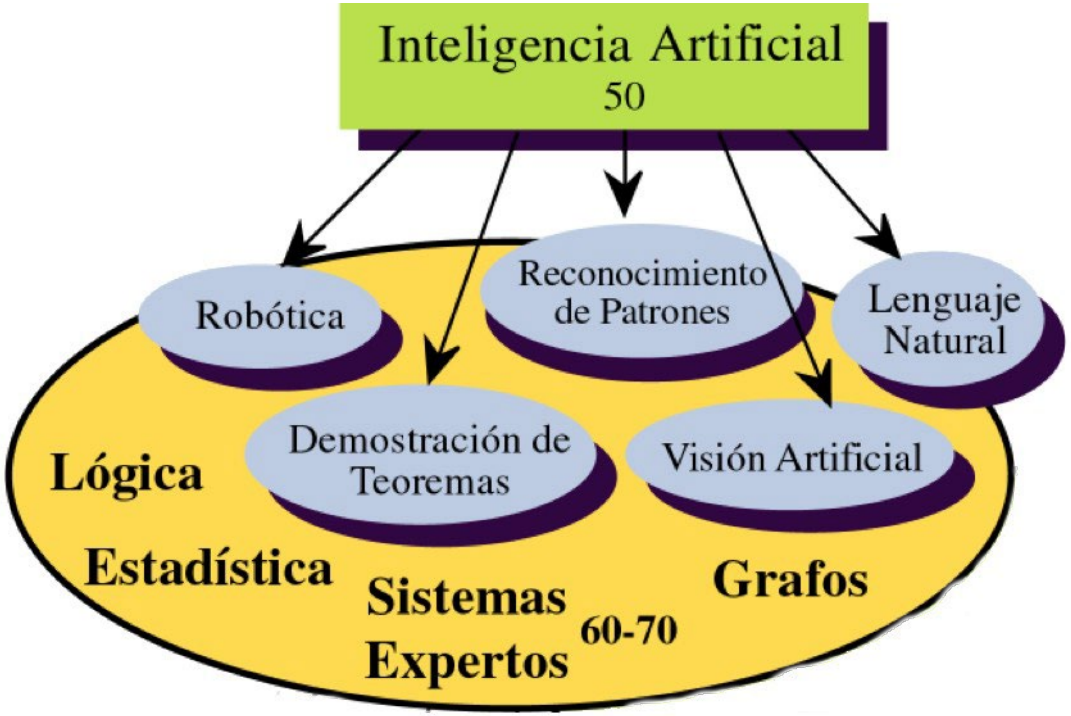
Joaquín Bedia

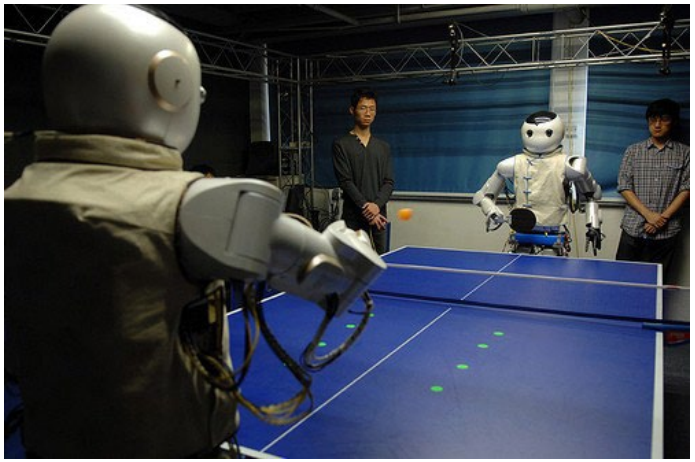
Grupo de Meteorología

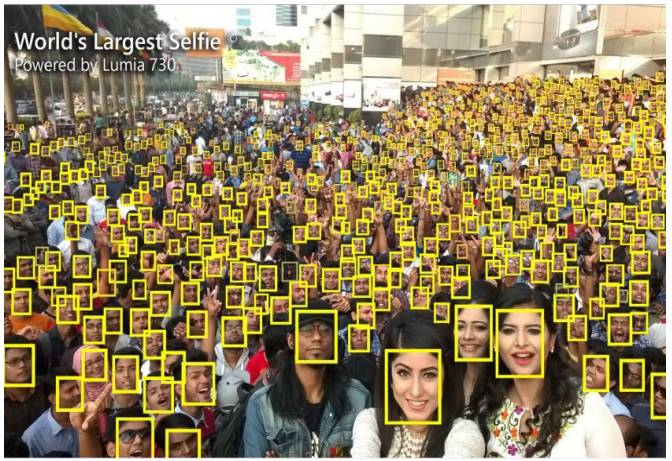
Univ. de Cantabria – CSIC
MACC / IFCA

Oct	29	Presentación, introducción y perspectiva histórica
	30	Paradigmas, problemas canonicos y data challenges
	31	**Reglas de asociación**
Nov	**4**	**Practica: Reglas de asociación**
	6	Evaluación, sobrejuste y crossvalidacion
	11	Practica: Crossvalidacion
	13	Árboles de clasificacion y decision
	18	Practica: Árboles de clasificación
	20	Técnicas de vecinos cercano (k-NN)
	25	Práctica: Vecinos cercanos
	27	Comparación de Técnicas de Clasificación.
Dic	2	Árboles de clasificación y regresion (CART)
	4	Práctica: Árboles de clasificación y regresion (CART)
	9	Practica: El paquete CARET
	11	Ensembles: Bagging and Boosting
	13	Random Forests
	16	Gradient boosting
	18	Practica: XAI-Explainable Artificial Intelligence
Ene	8	Reducción de dimensión no lineal
	13	Reducción de dimensión no lineal
	15	Técnicas de agrupamiento
	20	Técnicas de agrupamiento
	22	Predicción Condicionada
	24	Sesión de refuerzo/repaso.
	29	Examen

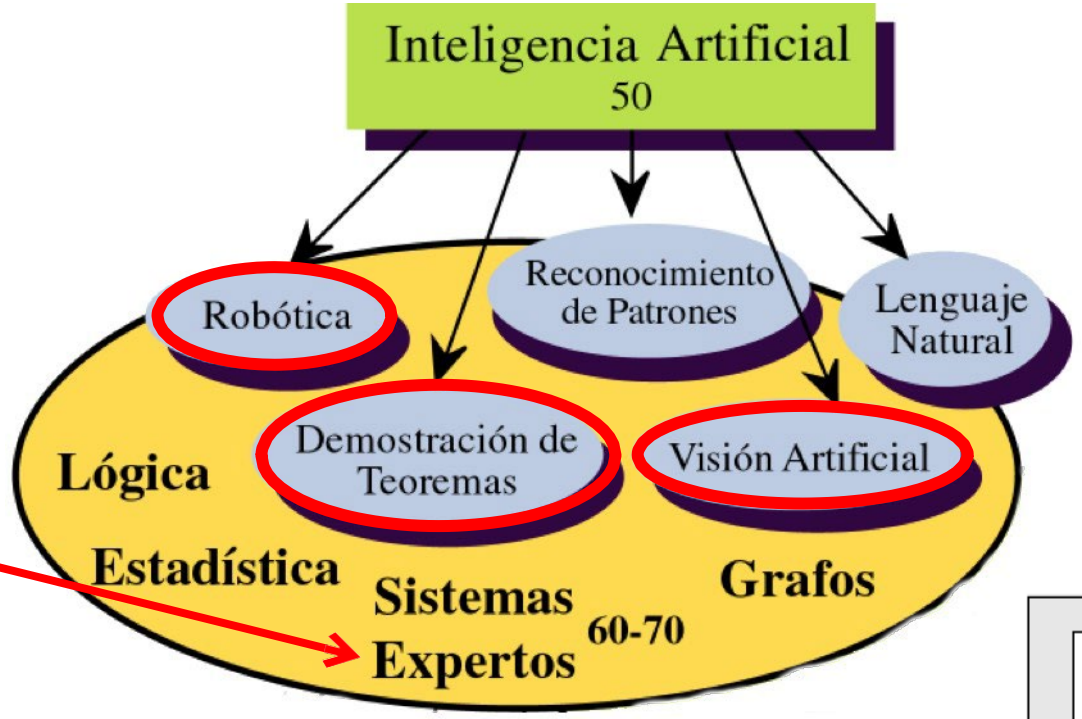
NOTA: Las líneas de código de R en esta presentación se muestran sobre un fondo gris.


Inteligencia Artificial
50
Robótica
Reconocimiento de Patrones
Lenguaje Natural
Demostración de Teoremas
Visión Artificial
Lógica
Estadística
Sistemas Expertos
60-70
Grafos




World's Largest Selfie
Powered by Lumia 730

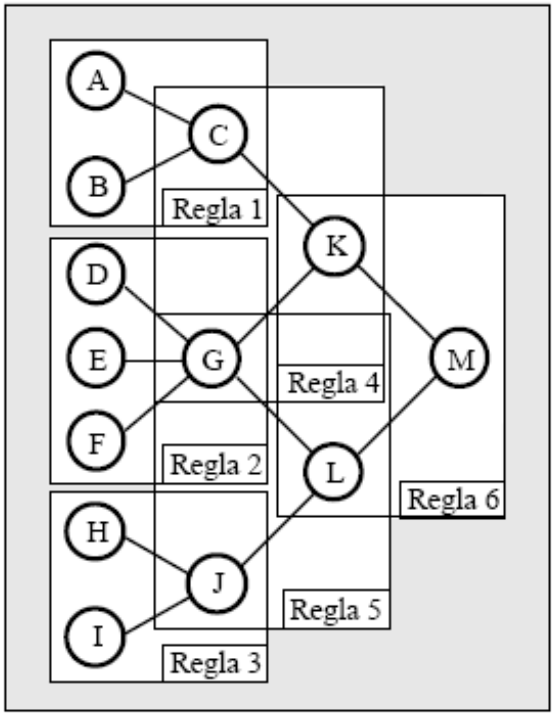

1996
Games
Kasparov
Deep Blue
Subscribe
8 7 6 5 4 3 2 1
a b c d e f g h



DATA MINING
1990

H, I cierto
L falso

G falso



Explicit representation of knowledge

Rules, graphs, etc.

- Modus ponens
  If P is true and P => Q is true then Q is true
- Modus tolens
  if P => Q is true and Q is false then ~P is true

Human-like reasoning

Logical inference, look for relations on graphs, etc.

Serial processing

+

Si un cliente compra pan y mantequilla ...

... entonces compra leche el 90%

NOTA: Las líneas de código de R en esta presentación se muestran sobre un fondo gris.

Master Universitario Oficial **Data Science**
con el apoyo del
UC UNIVERSIDAD DE CANTABRIA · UIMP Universidad Internacional Menéndez Pelayo · CSIC

PROBLEMS:

Oct	29	Presentación, introducción y perspectiva histórica
	30	Paradigmas, problemas canonicos y data challenges
	31	**Reglas de asociación**
Nov	**4**	**Practica: Reglas de asociación**
	6	Evaluación, sobrejuste y crossvalidacion
	11	Practica: Crossvalidacion
	13	Árboles de clasificacion y decision
	18	Practica: Árboles de clasificación
	20	Técnicas de vecinos cercano (k-NN)
	25	Práctica: Vecinos cercanos
	27	Comparación de Técnicas de Clasificación.
Dic	2	Árboles de clasificación y regresion (CART)
	4	Práctica: Árboles de clasificación y regresion (CART)
	9	Practica: El paquete CARET
	11	Ensembles: Bagging and Boosting
	13	Random Forests
	16	Gradient boosting
	18	Practica: XAI-Explainable Artificial Intelligence
Ene	8	Reducción de dimensión no lineal
	13	Reducción de dimensión no lineal
	15	Técnicas de agrupamiento
	20	Técnicas de agrupamiento
	22	Predicción Condicionada
	24	Sesión de refuerzo/repaso.
	29	Examen

¿Los registros de las ventas realizadas en el pasado son útiles para la gestión de un comercio?

1) Colocación/proposición de productos:
Si detecto productos que se compran de forma conjunta puedo planificar su colocación en el comercio.

2) Promociones y Ofertas:
Si detecto productos correlacionados puedo definir promociones que engloben a ambos (p.e. portátil + mochila, refrescos + snacks, juguetes + pilas, etc...).

3) Optimización y gestión de recursos:
Si detecto relaciones entre operaciones a realizar, puedo optimizar su planificación (p.e. en un servicio de atención al cliente si detectas problemas relacionados, puedes optimizar el servicio equipando al operario con material para resolver ambos problemas).
...

¿Los registros de las ventas realizadas en el pasado son útiles para la gestión de un comercio?

1) Colocación/proposición de productos:
Si detecto productos que se compran de forma conjunta puedo planificar su colocación en el comercio.

2) Promociones y Ofertas:
Si detecto productos correlacionados puedo definir promociones que engloben a ambos (p.e. portátil + mochila, refrescos + snacks, juguetes + pilas, etc...).

3) Optimización y gestión de recursos:
Si detecto relaciones entre operaciones a realizar, puedo optimizar su planificación (p.e. en un servicio de atención al cliente si detectas problemas relacionados, puedes optimizar el servicio equipando al operario con material para resolver ambos problemas).
...

Los algoritmos que buscan reglas de asociación analizan los datos registrados buscando relaciones entre los diferentes productos, obteniendo proposiciones del tipo: "El 90% de las compras que incluyen pan y mantequilla, compran también leche."

¿Los registros de las ventas realizadas en el pasado son útiles para la gestión de un comercio?

1) Colocación/proposición de productos:
Si detecto productos que se compran de forma conjunta puedo planificar su colocación en el comercio.

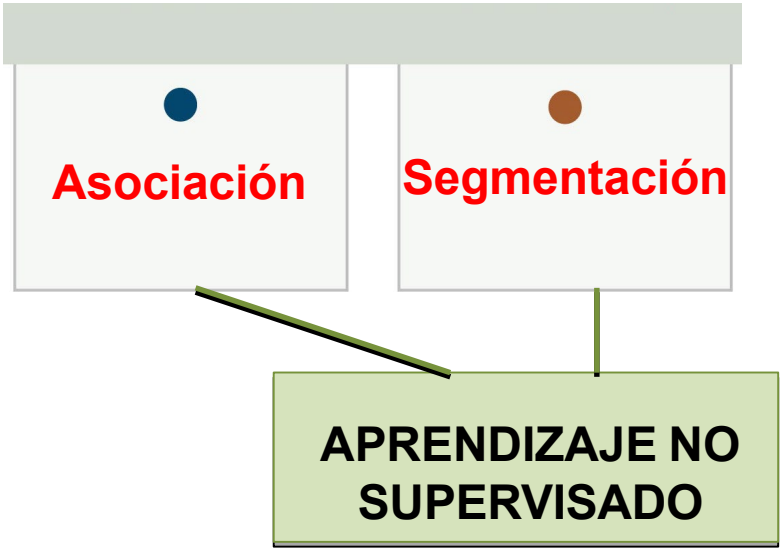
2) Promociones y Ofertas:
Si detecto productos correlacionados puedo definir promociones que engloben a ambos (p.e. portátil + mochila, refrescos + snacks, juguetes + pilas, etc...).

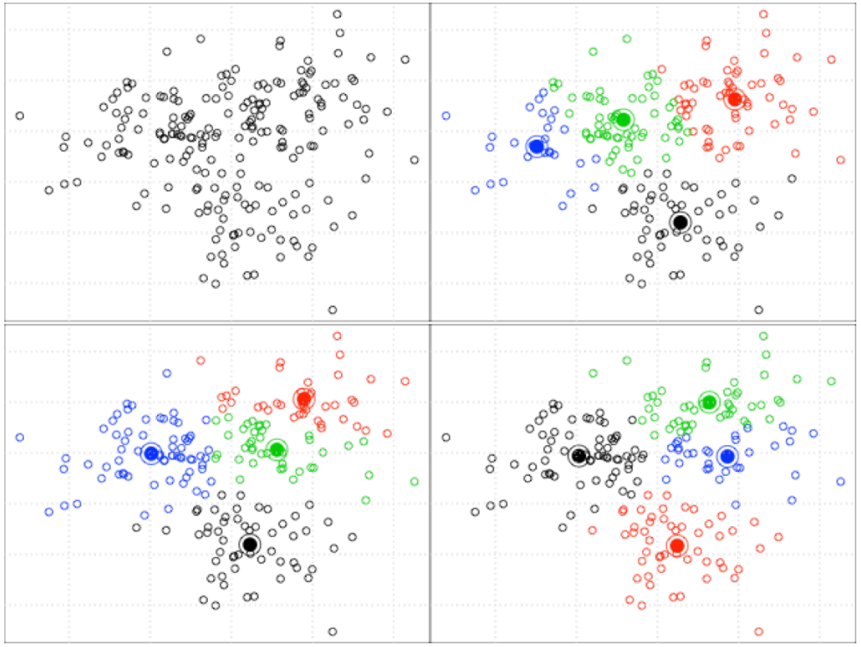
Los algoritmos que buscan reglas de asociación analizan los datos registrados buscando relaciones entre los diferentes productos, obteniendo proposiciones del tipo: "El 90% de las compras que incluyen pan y mantequilla, compran también leche."

A partir de las reglas encontradas, se resuelven consultas de diferentes tipos:

- Encontrar todas las reglas relacionando cualquier producto con uno dado (p.e. leche).
- Encontrar todas las reglas que dependen de un producto dado (p.e. pan).
- Encontrar todas las reglas que cumplan los dos criterios anteriores para ciertos productos dados.
- Encontrar el conjunto de reglas con mayor confianza cumpliendo alguno de los criterios anteriores.
- ...

https://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Frequent_Pattern_Mining



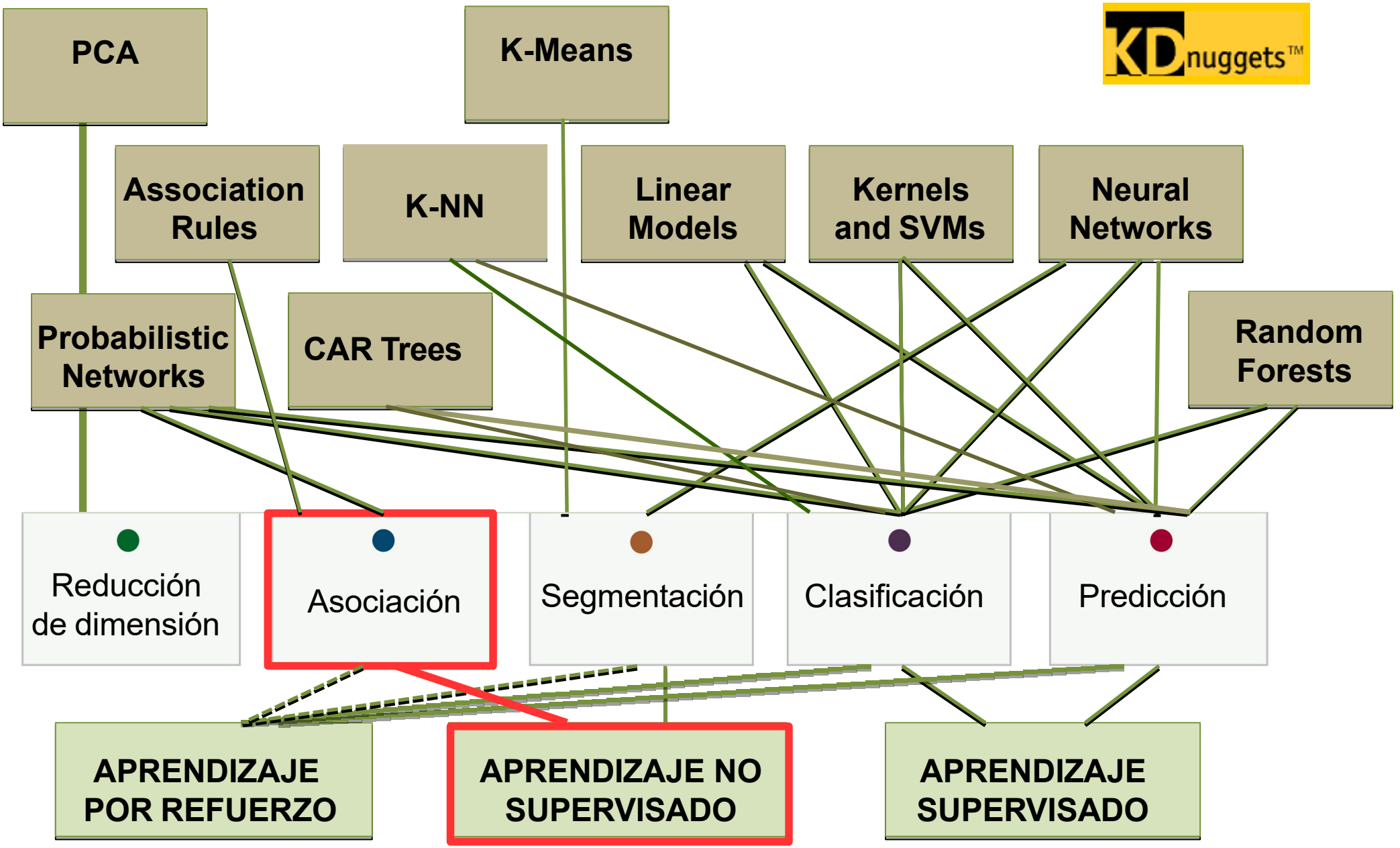


- Target Variable: *There is no target variable (association)*

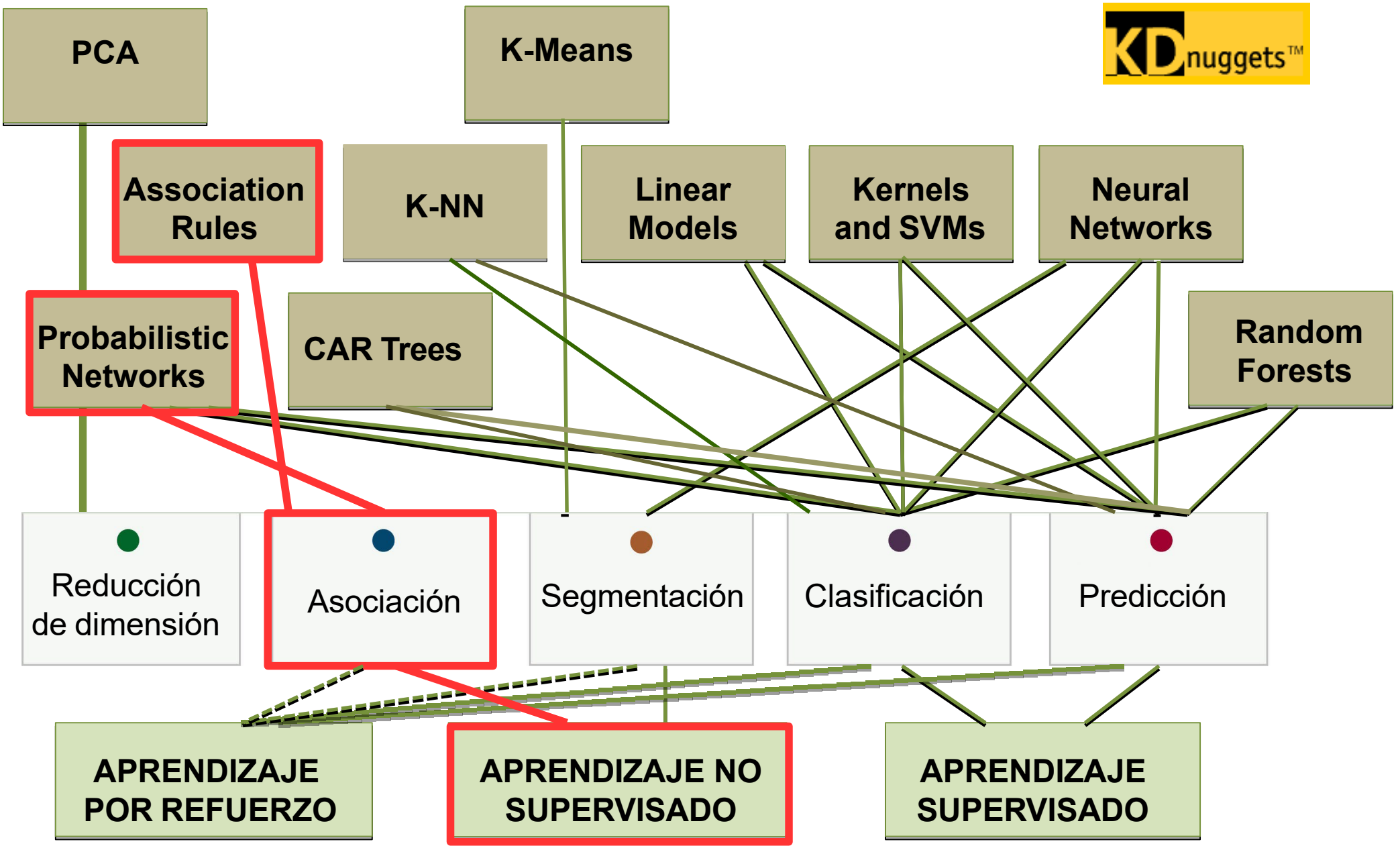
  K (cluster), discrete: #clusters (segmentation)

- Predictive Variables: $\{X_1, X_2, \ldots, X_N\}$: *continuous or discrete*
  - "Covariates" used to make predictions.
- Predictive Model: Algorithmic, based on $(X_1, X_2, \ldots, X_N)$.
  - Ad-hoc "learning" and "prediction" engine.

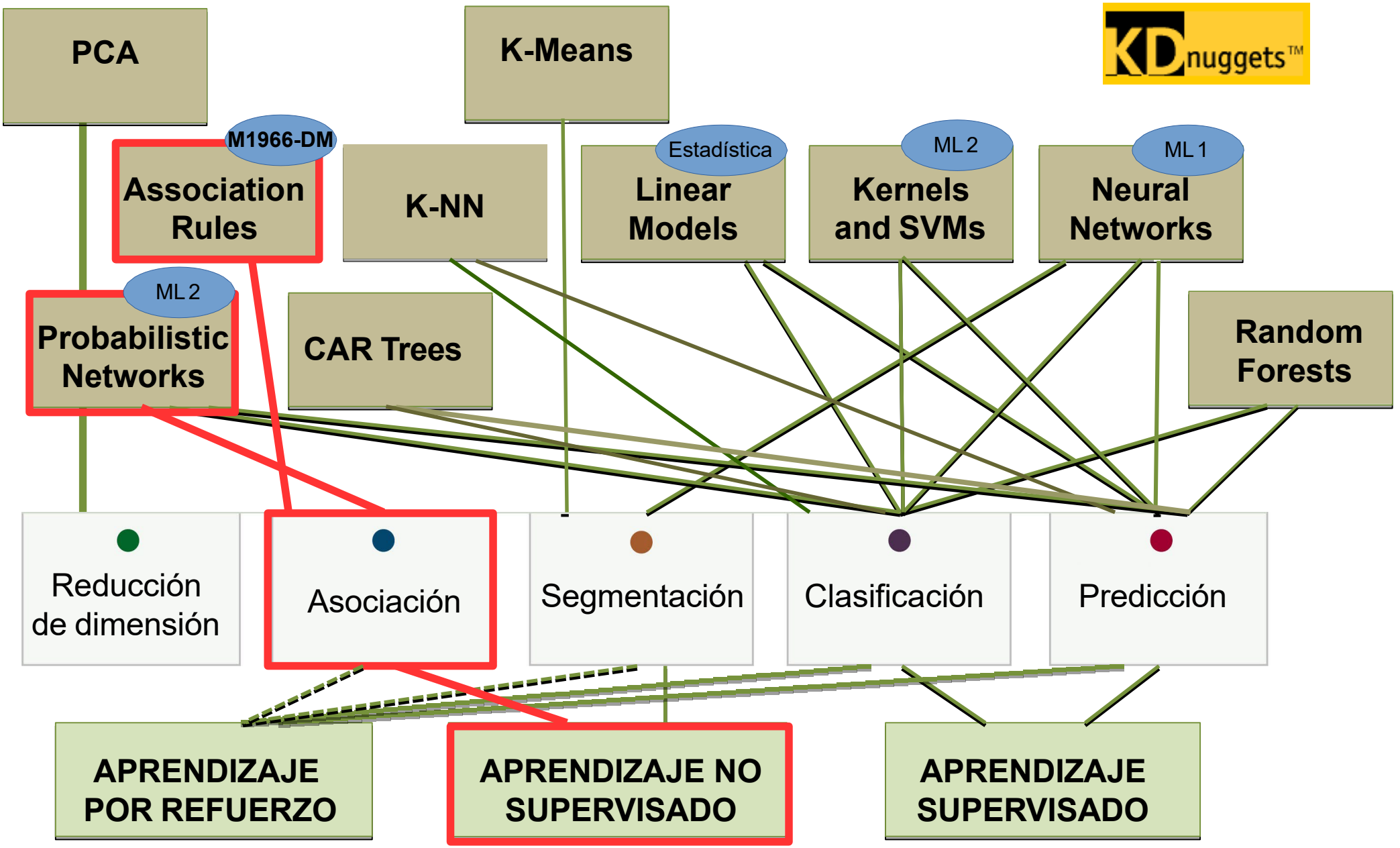
https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html



https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html



https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html



Listado de datasets utilizados en el curso

EN FUNCIÓN DE LA NATURALEZA DE LOS DATOS PODEMOS CLASIFICARLAS COMO

SÓLO CATEGÓRICAS (FACTORES)

- **Groceries.** Disponible en kaggle y en el paquete {arulesViz} de R.

-**Mushroom.** Disponible en kaggle y UCI.

MIXTOS (CONTINUOS Y FACTORES)

- **Iris.** Disponible en kaggle, UCI y el paquete {datasets} de R.

- **MNIST.** Disponible en

….

CONTINUOS

- **Meteo.** Basados en mediciones de parámetros físicos, químicos, etc...

Listado de datasets utilizados en el curso

EN FUNCIÓN DE LA NATURALEZA DE LOS DATOS PODEMOS CLASIFICARLAS COMO

SÓLO CATEGÓRICAS (FACTORES) ← Reglas de Asociación

- **Groceries.** Disponible en kaggle y en el paquete {arulesViz} de R.

- **Mushroom.** Disponible en kaggle y UCI.

MIXTOS (CONTINUOS Y FACTORES)

- **Iris.** Disponible en kaggle, UCI y el paquete {datasets} de R.

- **MNIST.** Disponible en

….

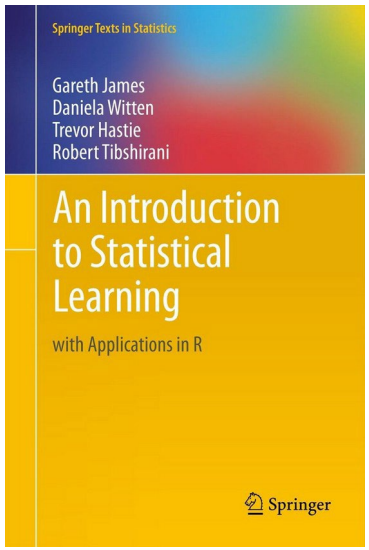
CONTINUOS

- **Meteo.** Basados en mediciones de parámetros físicos, químicos, etc...

Tareas

1 30-60mins

Echa un vistazo a los datasets que hay en el paquete ISLR.




```
install.packages("ISLR")
library("ISLR")
library(help = "ISLR")
```

Analiza la estructura de los datasets: ¿de qué tipo son? ¿para qué tipo de problemas serían adecuados? e.g.

```
data("Hitters")
str(Hitters)
```

¿Hay algún dataset en este paquete que entronque en este problema? En caso de existir, ¿cual(es) es(son)?

2 60-90mins

Lee con calma el siguiente notebook de kaggle sobre las duraciones de los trayectos de taxi en Nueva York:

https://www.kaggle.com/headsortails/nyc-taxi-eda-update-the-fast-the-curious/notebook

Reglas de Asociación

- Introducción – Objetivo
- Modelo Formal - Conceptos Básicos
- Reglas de Asociación
- Algoritmo Apriori
- Librería Arules/ArulesViz
- Redes Probabilísticas → Introducción
- Algoritmo Eclat

Reglas de Asociación

- Introducción – Objetivo
- **Modelo Formal - Conceptos Básicos**
- Reglas de Asociación
- Algoritmo Apriori
- Librería Arules/ArulesViz
- Redes Probabilísticas → Introducción
- Algoritmo Eclat

Sean **X**=(X1, X2,..., Xm) un conjunto de variables binarias (***items***). Sea ***T*** un conjunto de transacciones, representadas por vectores binarios ***t*** en los que ***t(k)=1*** si se da la variable ***Xk***. De este modo, decimos que la transición ***t*** cumple ***X*** si para todas los items ***k*** de ***X*** se da que ***t(k)=1***.

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Sean **X**=(X1, X2,..., Xm) un conjunto de variables binarias (***items***). Sea ***T*** un conjunto de transacciones, representadas por vectores binarios ***t*** en los que ***t(k)=1*** si se da la variable ***Xk***. De este modo, decimos que la transición ***t*** cumple ***X*** si para todas los items ***k*** de ***X*** se da que ***t(k)=1***.

Definimos una ***regla de asociación*** como una implicación ***X -> Y***, donde ***X*** es un conjunto de items e ***Y*** es un conjunto de items no incluidos en ***X***.

{Limón, Naranja} -> Galletas

Livello di confidenta di 50% perché solo una delle due persone che ha comprato limone e arancia ha poi comprato i biscotti: casi favorevoli / casi tot

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Sean **X**=(X1, X2,..., Xm) un conjunto de variables binarias (***items***). Sea ***T*** un conjunto de transacciones, representadas por vectores binarios ***t*** en los que ***t(k)=1*** si se da la variable ***Xk***. De este modo, decimos que la transición ***t*** cumple ***X*** si para todas los items ***k*** de ***X*** se da que ***t(k)=1***.

Definimos una ***regla de asociación*** como una implicación ***X -> Y***, donde ***X*** es un conjunto de items e ***Y*** es un conjunto de items no incluidos en ***X***.

{Limón, Naranja} -> Galletas

Dada una regla de asociación, se define el ***factor de confianza 0<c<1*** como el ratio entre las transacciones en ***T*** que cumplen ***X*** e ***Y*** y las transacciones que cumplen ***X***.

¿Cuál es la confianza de la regla: {Limón, Naranja} -> Galletas?

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Sean **X**=(X1, X2,..., Xm) un conjunto de variables binarias (***items***). Sea ***T*** un conjunto de transacciones, representadas por vectores binarios ***t*** en los que ***t(k)=1*** si se da la variable ***Xk***. De este modo, decimos que la transición ***t*** cumple ***X*** si para todas los items ***k*** de ***X*** se da que ***t(k)=1***.

Por lo tanto, dado un conjunto de transacciones ***T***, estamos interesados en generar todas las reglas que cumplan ciertas condiciones que podemos clasificar en dos tipos:

1.Sintácticas: se refieren a buscar reglas que involucren determinados items específicos, bien sea en el conjunto ***X***, en el ***Y*** o en ambos.

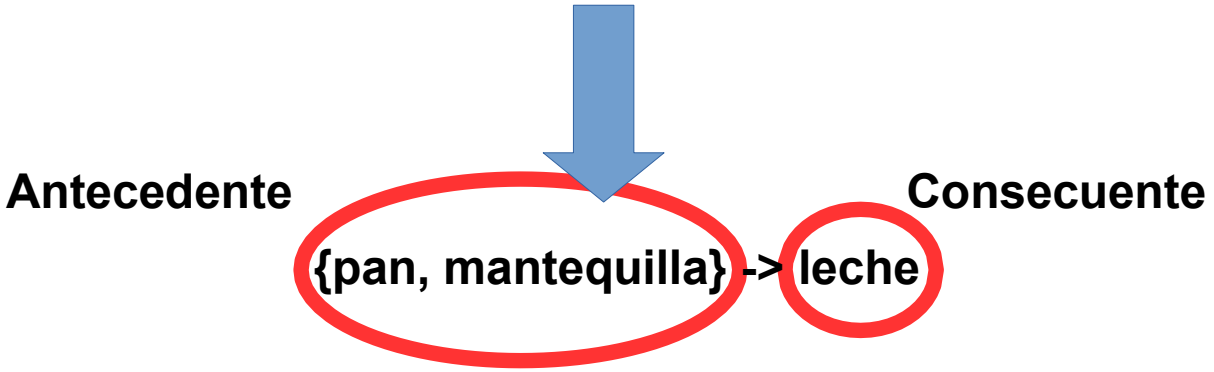
2.Frecuencia/relevancia: se refieren a buscar reglas que se den en un número significativo de casos dentro del conjunto total. Habitualmente, estaremos interesados en reglas cuya relevancia sea superior a un umbral ya que serán aquellas con mayor impacto. De este modo, una relevancia baja implica que la regla no es muy importante y que puede, de ser necesario, ser considerada en etapas posteriores.

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

“El 90% de las compras que incluyen pan y mantequilla, compran también leche.”

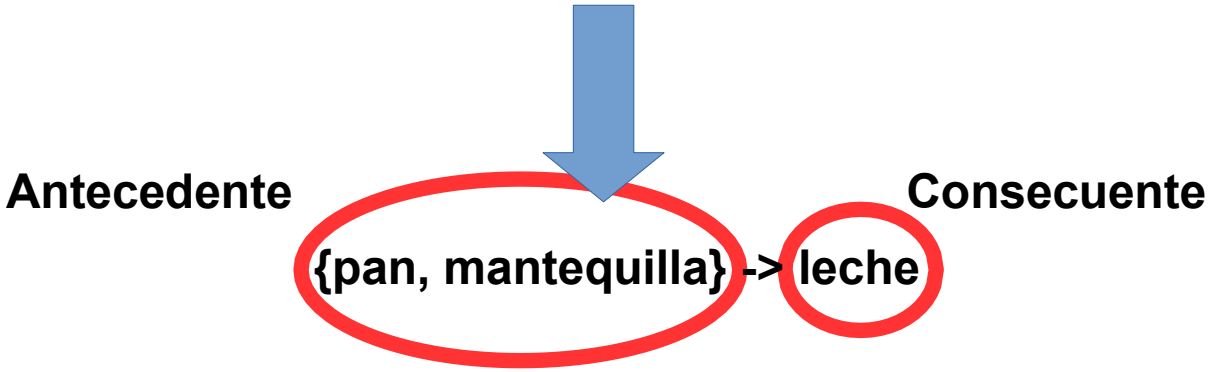
{pan, mantequilla} -> leche

“El 90% de las compras que incluyen pan y mantequilla, compran también leche.”



“El 90% de las compras que incluyen pan y mantequilla, compran también leche.”

Confianza (C)

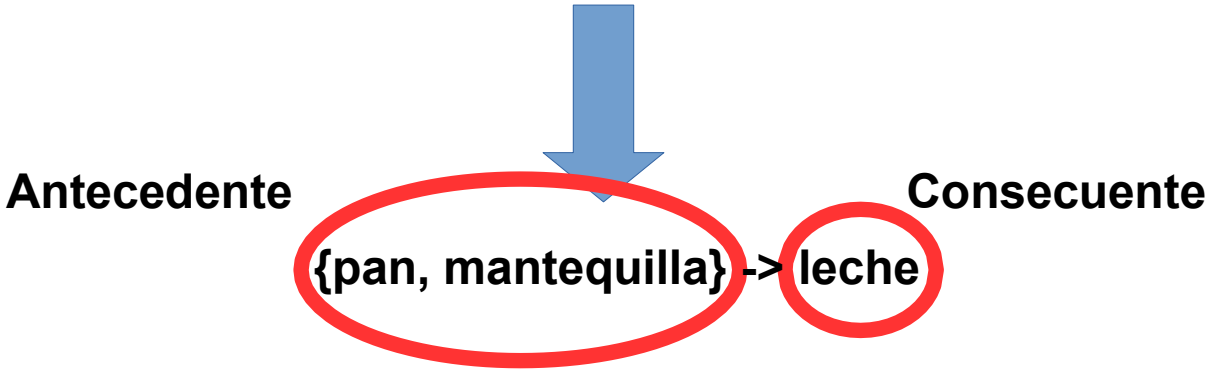


C({pan, mantequilla} -> leche) = S({pan, mantequilla,leche})/S({pan, mantequilla})

El soporte del conjunto X, S(X), es el número de observaciones en las que se da dicho evento

“El 90% de las compras que incluyen pan y mantequilla, compran también leche.”

Confianza (C)



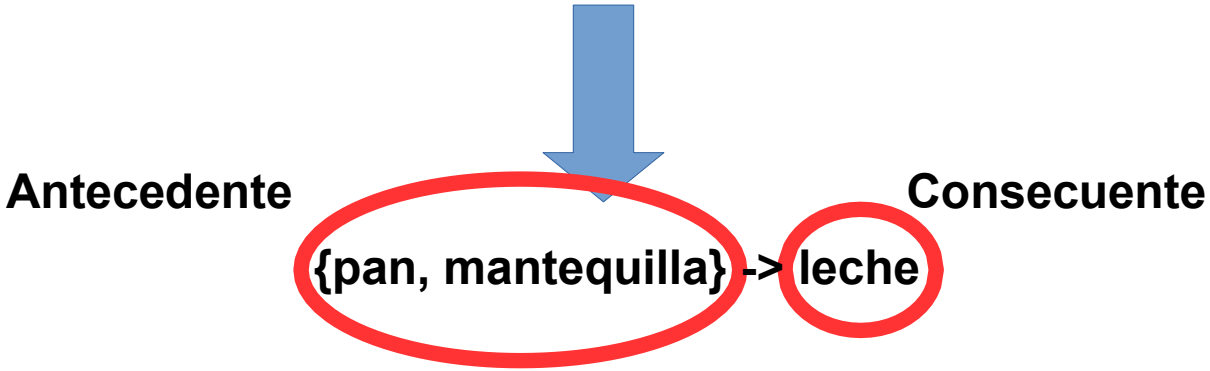
C({pan, mantequilla} -> leche) = S({pan, mantequilla, leche})/S({pan, mantequilla})

El soporte del conjunto X, S(X), es el número de observaciones en las que se da dicho evento. El soporte de una regla viene dada por el soporte del conjunto {X,Y}, S(X,Y).

De cada 10 veces que se ha comprado pan y mantequilla en la tienda, 9 se ha comprado también leche.

“El 90% de las compras que incluyen pan y mantequilla, compran también leche.”

Confianza (C)



C({pan, mantequilla} -> leche) = S({pan, mantequilla, leche})/S({pan, mantequilla})

El soporte del conjunto X, S(X), es el número de observaciones en las que se da dicho evento. El soporte de una regla viene dada por el soporte del conjunto {X,Y}, S(X,Y).

De cada 10 veces que se ha comprado pan y mantequilla en la tienda, 9 se ha comprado también leche.

Nota: El factor de confianza es una medida del peso de la regla mientras que el soporte/relevancia se corresponde con la significancia estadística.

Reglas de Asociación

- Introducción – Objetivo
- Modelo Formal - Conceptos Básicos
- **Reglas de Asociación**
- Algoritmo Apriori
- Librería Arules/ArulesViz
- Redes Probabilísticas → Introducción
- Algoritmo Eclat

A partir de las definiciones de soporte y confianza de una regla surge, de forma natural, el siguiente algoritmo:

1) Definir umbrales ***MinSupp*** y ***MinConf*** para el soporte y la confianza de la regla.

2) Enumerar todas las reglas de asociación posibles.

3) Calcular el soporte y la confianza de cada regla.

4) Eliminar las reglas que no superen los umbrales establecidos. Es decir, ***C(X → Y)<MinConf*** ó ***S(X,Y)<MinSupp***.

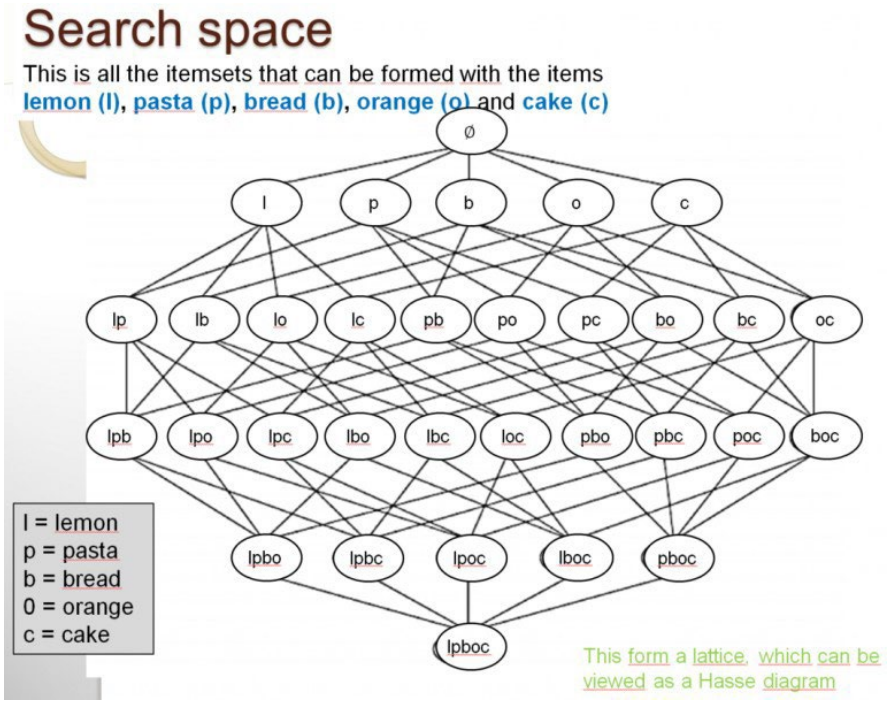
A partir de las definiciones de soporte y confianza de una regla surge, de forma natural, el siguiente algoritmo:

1) Definir umbrales ***MinSupp*** y ***MinConf*** para el soporte y la confianza de la regla.

2) Enumerar todas las reglas de asociación posibles.

3) Calcular el soporte y la confianza de cada regla.

4) Eliminar las reglas que no superen los umbrales establecidos. Es decir, ***C(X → Y)<MinConf*** ó ***S(X,Y)<MinSupp***.



5 items → 32=2^5 itemsets → 3^5-2^6+1 reglas.

!!!!Computacionalmente Inviable¡¡¡¡

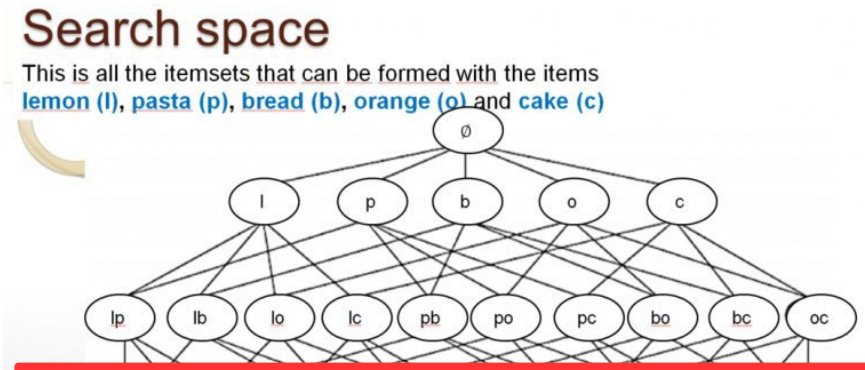
A partir de las definiciones de soporte y confianza de una regla surge, de forma natural, el siguiente algoritmo:

1) Definir umbrales ***MinSupp*** y ***MinConf*** para el soporte y la confianza de la regla.

2) Enumerar todas las reglas de asociación posibles.

3) Calcular el soporte y la confianza de cada regla.

4) Eliminar las reglas que no superen los umbrales establecidos. Es decir, ***C(X → Y)<MinConf*** ó ***S(X,Y)<MinSupp***.



5 items → 32=2^5 itemsets → 3^5-2^6+1 reglas.

!!!!Computacionalmente Inviable¡¡¡¡

1) Reducir el **número de candidatos** → Técnicas de **poda**.

2) Reducir el **número de transacciones** conforme aumenta el tamaño del itemset.

3)Estructurar los datos de forma eficiente para almacenar los candidatos o las transacciones, de forma que se **reduzcan las comparaciones**.

Reglas de Asociación

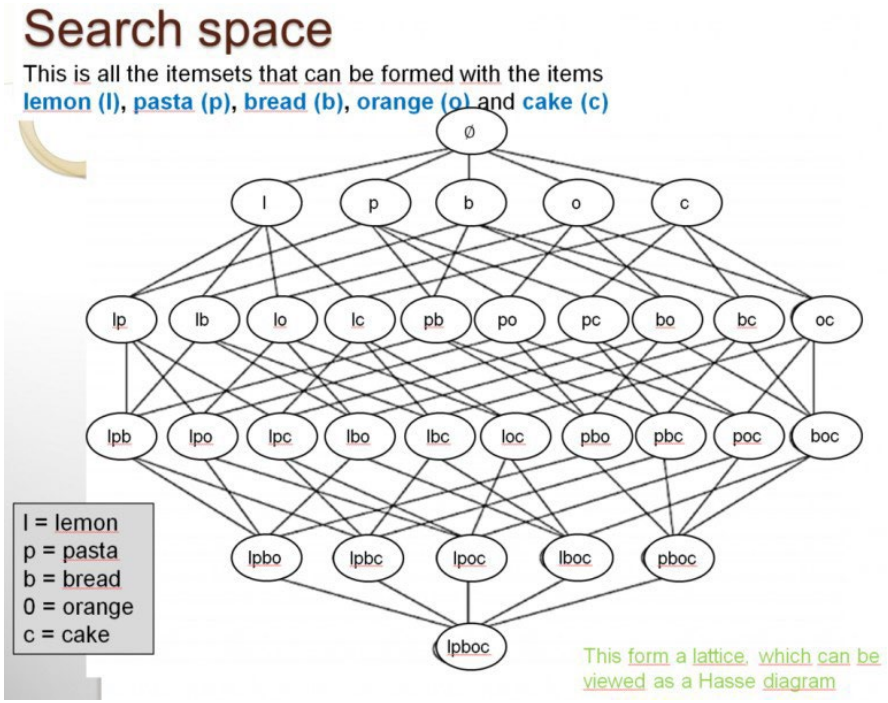
- Introducción – Objetivo
- Modelo Formal - Conceptos Básicos
- Reglas de Asociación
- **Algoritmo Apriori**
- Librería Arules/ArulesViz
- Redes Probabilísticas → Introducción
- Algoritmo Eclat

El algoritmo **APRIORI** se enmarca dentro de las técnicas que buscar la reducción del número de candidatos a través de la ***"poda"*** de algunas de las ramas del ***"espacio de búsqueda"***.
Se fundamenta en la siguiente propiedad (***anti-monotonía del soporte***):

$$\forall X, Y : (X \subseteq Y) \Rightarrow S(X) \geq S(Y)$$

Numero di persone che comprano due elementi = frequenza di quell'insieme di elementi
es. {P,M} insieme di persone che comprano pane e burro, {P,M,L} insieme di persone che comprano pane, burro e limone, possiamo dire che il primo insieme è contenuto nel secondo e che la frequnza è maggiore

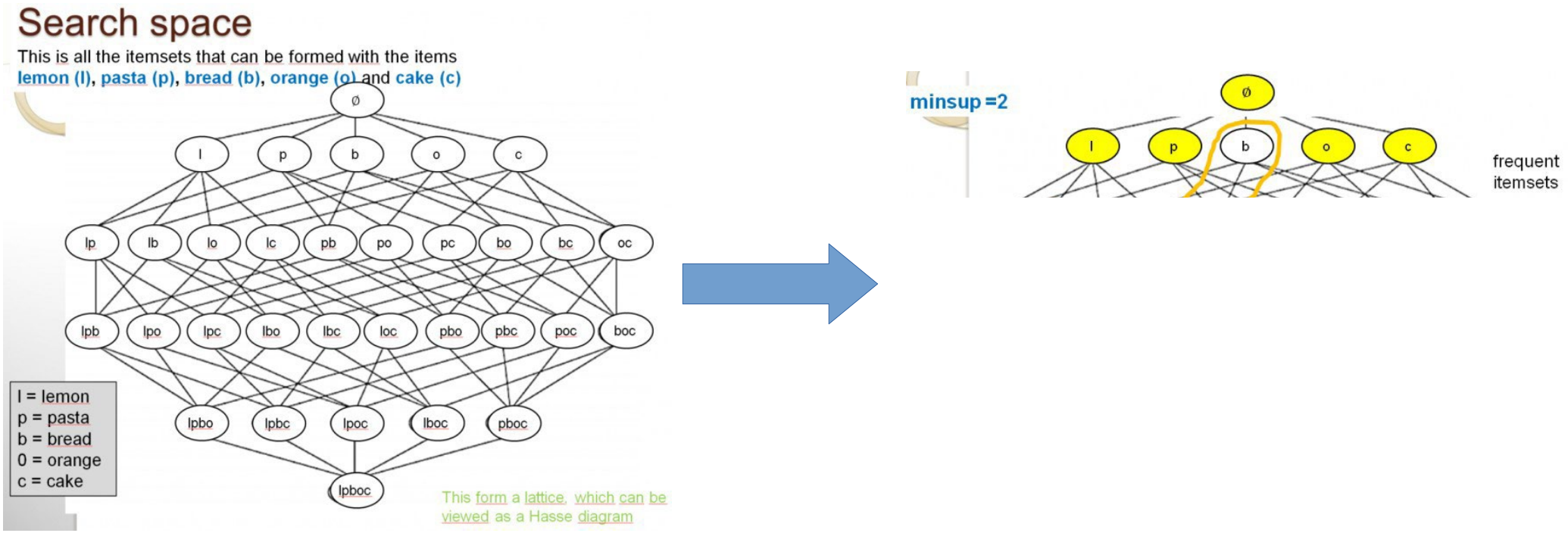
Es decir, si un itemset es frecuente, también lo son todos sus subconjuntos ya que el soporte de un itemset nunca puede ser superior al de cualquier subconjunto.



El algoritmo **APRIORI** se enmarca dentro de las técnicas que buscar la reducción del número de candidatos a través de la ***"poda"*** de algunas de las ramas del ***"espacio de búsqueda"***.
Se fundamenta en la siguiente propiedad (***anti-monotonía del soporte***):

$$\forall X, Y : (X \subseteq Y) \Rightarrow S(X) \geq S(Y)$$

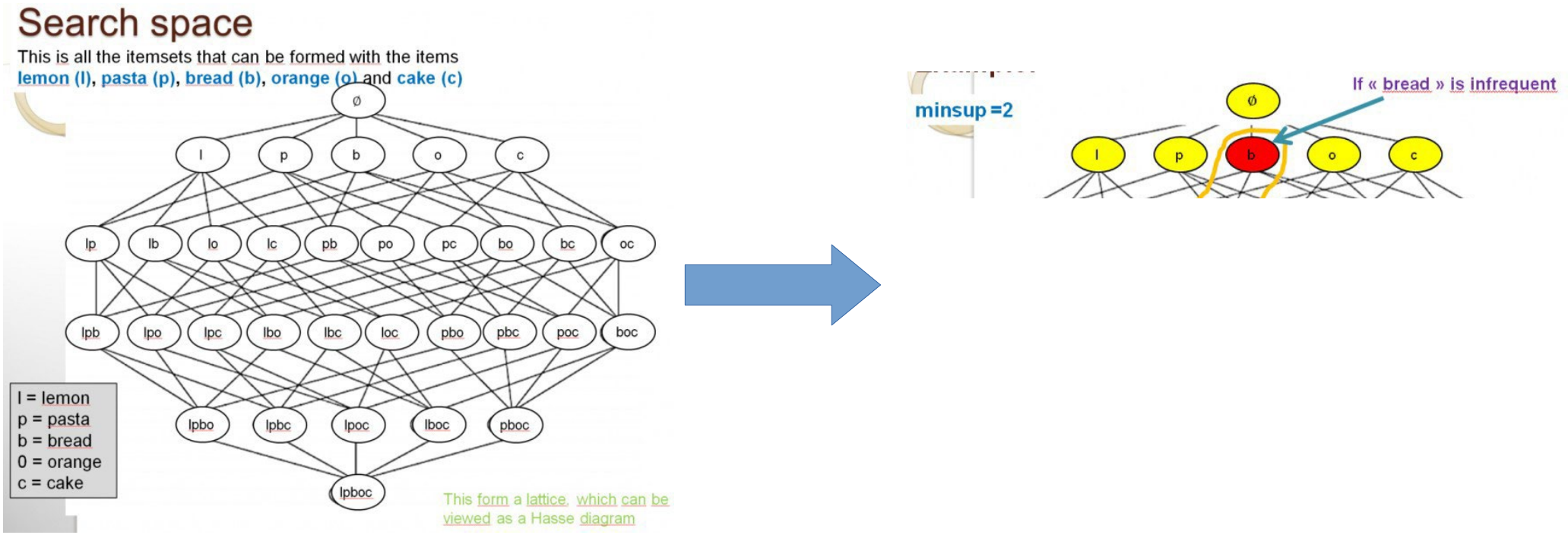
Es decir, si un itemset es frecuente, también lo son todos sus subconjuntos ya que el soporte de un itemset nunca puede ser superior al de cualquier subconjunto.



El algoritmo **APRIORI** se enmarca dentro de las técnicas que buscar la reducción del número de candidatos a través de la ***"poda"*** de algunas de las ramas del ***"espacio de búsqueda"***.
Se fundamenta en la siguiente propiedad (***anti-monotonía del soporte***):

$$\forall X, Y : (X \subseteq Y) \Rightarrow S(X) \geq S(Y)$$

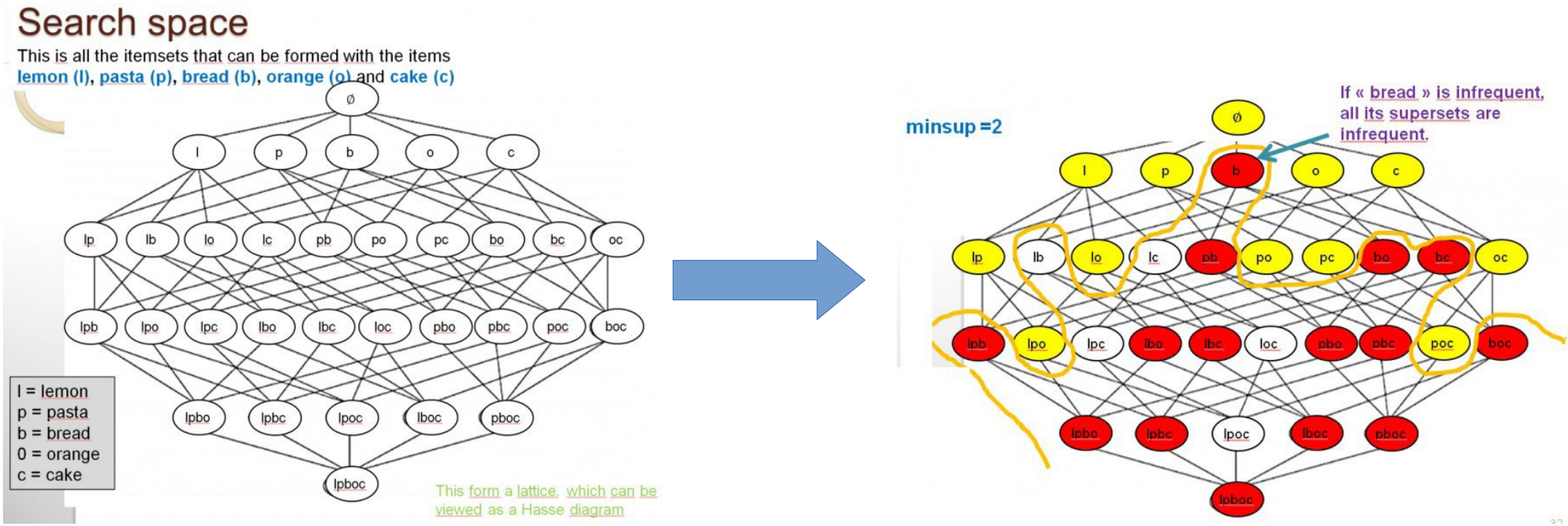
Es decir, si un itemset es frecuente, también lo son todos sus subconjuntos ya que el soporte de un itemset nunca puede ser superior al de cualquier subconjunto.



El algoritmo **APRIORI** se enmarca dentro de las técnicas que buscar la reducción del número de candidatos a través de la ***"poda"*** de algunas de las ramas del ***"espacio de búsqueda"***.
Se fundamenta en la siguiente propiedad (***anti-monotonía del soporte***):

$$\forall X, Y : (X \subseteq Y) \Rightarrow S(X) \geq S(Y)$$

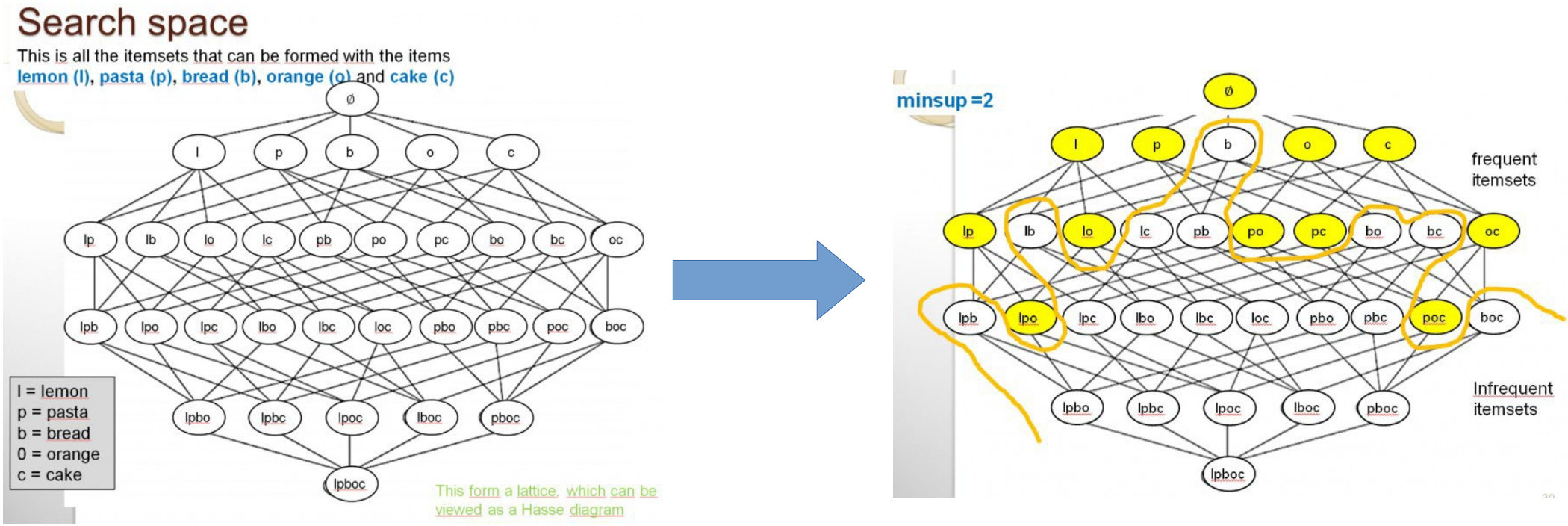
Es decir, si un itemset es frecuente, también lo son todos sus subconjuntos ya que el soporte de un itemset nunca puede ser superior al de cualquier subconjunto.
Los itemsets derivados de itemsets poco frecuentes no se incluyen en el siguiente paso.



El algoritmo **APRIORI** se enmarca dentro de las técnicas que buscar la reducción del número de candidatos a través de la ***"poda"*** de algunas de las ramas del ***"espacio de búsqueda"***.
Se fundamenta en la siguiente propiedad (***anti-monotonía del soporte***):

$$\forall X, Y : (X \subseteq Y) \Rightarrow S(X) \geq S(Y)$$

Es decir, si un itemset es frecuente, también lo son todos sus subconjuntos ya que el soporte de un itemset nunca puede ser superior al de cualquier subconjunto.
Los itemsets derivados de itemsets poco frecuentes no se incluyen en el siguiente paso.

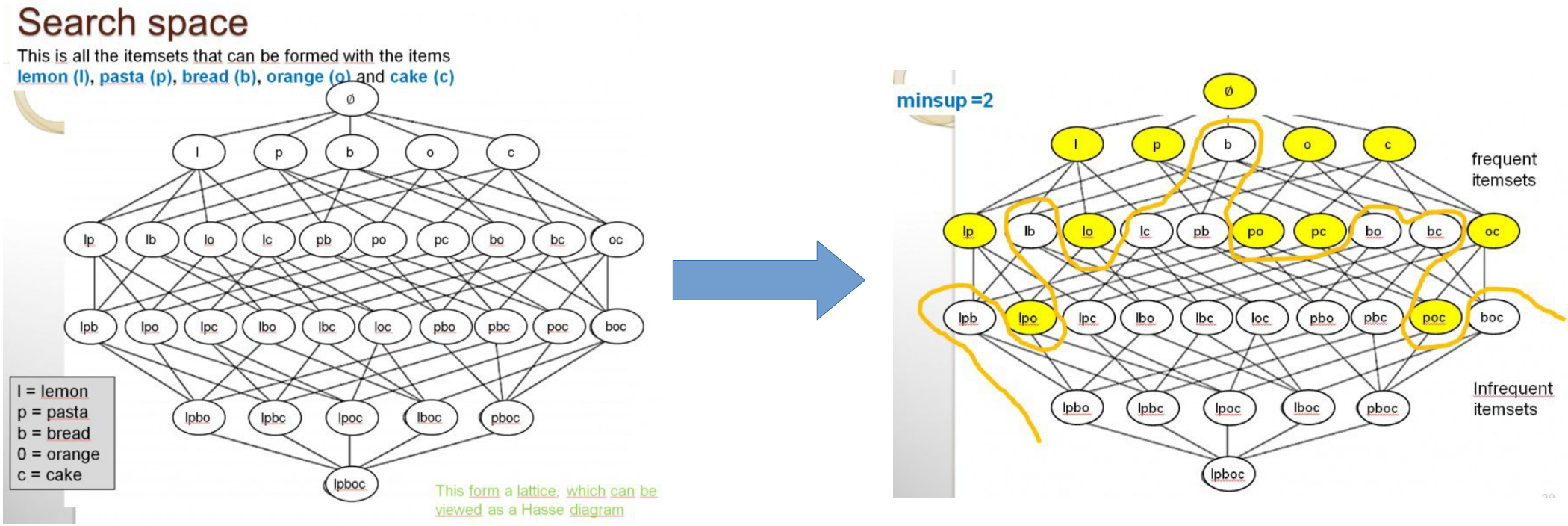


El algoritmo **APRIORI** se enmarca dentro de las técnicas que buscar la reducción del número de candidatos a través de la ***"poda"*** de algunas de las ramas del ***"espacio de búsqueda"***.
Se fundamenta en la siguiente propiedad (***anti-monotonía del soporte***):

$$\forall X, Y : (X \subseteq Y) \Rightarrow S(X) \geq S(Y)$$

Es decir, si un itemset es frecuente, también lo son todos sus subconjuntos ya que el soporte de un itemset nunca puede ser superior al de cualquier subconjunto.
Del mismo modo,

$$si \; \exists X \subseteq Y : X \; no \; frecuente \Rightarrow Y \; no \; frecuente$$



El algoritmo **APRIORI** se enmarca dentro de las técnicas que buscar la reducción del número de candidatos a través de la ***"poda"*** de algunas de las ramas del ***"espacio de búsqueda"***.
Se fundamenta en la siguiente propiedad (***anti-monotonía del soporte***):

$$\forall X, Y : (X \subseteq Y) \Rightarrow S(X) \geq S(Y)$$

Es decir, si un itemset es frecuente, también lo son todos sus subconjuntos ya que el soporte de un itemset nunca puede ser superior al de cualquier subconjunto.
Del mismo modo,

$$si \; \exists X \subseteq Y : X \; no \; frecuente \Rightarrow Y \; no \; frecuente$$

Si bien, en cierto modo es una reformulación de la propiedad anterior, da lugar a uno de los pasos del algoritmo **APRIORI** y, por lo tanto, nos permitirá entender éste mejor. Dada una relevancia mínima ***MinSupp***:

1. i = 1 (tamaño de los conjuntos)

2. Generar un conjunto unitario en ***S1*** para cada atributo.

3. Comprobar la relevancia de todos los conjuntos en ***Si***, descartando aquellos tales que:

S(X) < MinSupp

4. Combinar los conjuntos en ***Si*** creando conjuntos de tamaño ***i+1*** en ***Si+1***.

5. Si ***Si*** no es vacío **entonces** ***i = i+ 1***. Ir a 3.

6. Si no, devolver ***{S1, S2,.., Si}***

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Definimos la cota para el soporte/relevancia: ***MinSupp = 2***

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Definimos la cota para el soporte/relevancia: ***MinSupp = 2***

1.- Subconjuntos de tamaño **i=1**: **{I1, I2, I3, I4, I5}**

2.- Soporte/relevancia: **{S(I1) = 4; S(I2) = 3; S(I3) = 3; S(I4) = 1; S(I5) = 2}**

¿Todos los subconjuntos cumplen la condición del soporte?

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Definimos la cota para el soporte/relevancia: ***MinSupp = 2***

1.- Subconjuntos de tamaño **i=1**: **{I1, I2, I3, I4, I5}**

2.- Soporte/relevancia: **{S(I1) = 4; S(I2) = 3; S(I3) = 3; S(I4) = 1; S(I5) = 2}**

¿Todos los subconjuntos cumplen la condición del soporte? NO

3.- Aplicamos el ***MinSupp***: **S1 = {I1; I2; I3; I5}** → Se excluye el I4-Pan y derivados.

4.- Construimos subconjuntos de tamaño **i=2** a partir de **S1**: **{I1I2; I1I3; I1I5; I2I3; I2I5; I3I5}**

5.- Aplicamos la segunda propiedad, ¿Se excluye algún subconjunto de S1?

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Definimos la cota para el soporte/relevancia: ***MinSupp = 2***

1.- Subconjuntos de tamaño **i=1**: **{I1, I2, I3, I4, I5}**

2.- Soporte/relevancia: **{S(I1) = 4; S(I2) = 3; S(I3) = 3; S(I4) = 1; S(I5) = 2}**

¿Todos los subconjuntos cumplen la condición del soporte? **NO**

3.- Aplicamos el ***MinSupp***: **S1 = {I1; I2; I3; I5}** → Se excluye el I4-Pan y derivados.

4.- Construimos subconjuntos de tamaño **i=2** a partir de **S1**: **{I1I2; I1I3; I1I5; I2I3; I2I5; I3I5}**

5.- Calculamos su soporte/relevancia: **{I1I2 = 3; I1I3 = 3; I1I5 = 2; I2I3 = 2; I2I5 = 1; I3I5 = 2}**

6.- Aplicamos la segunda propiedad, ¿Se excluye algún subconjunto? **SI**

7.- Aplicamos el ***MinSupp***: **S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}** → Se excluye el I2I5 y derivados.

8.- Construimos subconjuntos de tamaño **i=3** a partir de **S2**: **{I1I2I3; I1I2I5; I1I3I5; I2I3I5}**

1.- Subconjuntos de tamaño **i=1**: **{I1, I2, I3, I4, I5}**

2.- Soporte/relevancia: **{S(I1) = 4; S(I2) = 3; S(I3) = 3; S(I4) = 1; S(I5) = 2}**

3.- Aplicamos el ***MinSupp***: **S1 = {I1; I2; I3; I5}** → Se excluye el I4-Pan y derivados.

4.- Construimos subconjuntos de tamaño **i=2** a partir de **S1**: **{I1I2; I1I3; I1I5; I2I3; I2I5; I3I5}**

5.- Aplicamos la segunda propiedad. Para **i=2** no se excluye ningún subconjunto.

6.- Calculamos su soporte/relevancia: **{I1I2 = 3; I1I3 = 3; I1I5 = 2; I2I3 = 2; I2I5 = 1; I3I5 = 2}**

7.- Aplicamos el ***MinSupp***: **S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}** → Se excluye el I2I5 y derivados.

8.- Construimos subconjuntos de tamaño **i=3** a partir de **S2** **{I1I2I3; I1I2I5; I1I3I5; I2I3I5}**

Se considera un orden de los items para realizar la unión sin incurrir en la repetición de conjuntos. De este modo, únicamente se combinan items que en la ordenación estén en posiciones superiores a la de los items de los itemsets considerados.

1.- Subconjuntos de tamaño **i=1**: **{I1, I2, I3, I4, I5}**

2.- Soporte/relevancia: **{S(I1) = 4; S(I2) = 3; S(I3) = 3; S(I4) = 1; S(I5) = 2}**

3.- Aplicamos el ***MinSupp***: **S1 = {I1; I2; I3; I5}** → Se excluye el I4-Pan y derivados.

4.- Construimos subconjuntos de tamaño **i=2** a partir de **S1**: **{I1I2; I1I3; I1I5; I2I3; I2I5; I3I5}**

5.- Aplicamos la segunda propiedad. Para **i=2** no se excluye ningún subconjunto.

6.- Calculamos su soporte/relevancia: **{I1I2 = 3; I1I3 = 3; I1I5 = 2; I2I3 = 2; I2I5 = 1; I3I5 = 2}**

7.- Aplicamos el ***MinSupp***: **S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}** → Se excluye el I2I5 y derivados.

8.- Construimos subconjuntos de tamaño **i=3** a partir de **S2**: **{I1I2I3; I1I2I5; I1I3I5; I2I3I5}**

9.- Aplicamos la segunda propiedad, eliminando **{I1I2I5; I2I3I5}**.

10.- Calculamos su soporte/relevancia: **{I1I2I3 = 2; I1I3I5 = 2}**

11.- Aplicamos el ***MinSupp***: **S3 = {I1I2I3; I1I3I5}**

12.- Construimos subconjuntos de tamaño **i=4** a partir de **S3**: **{I1I2I3I5}**

13.- Aplicamos la segunda propiedad, eliminando **{I1I2I3I5}** al contener un subconjunto de orden 3 no frecuente (**I1I2I5** ó **I2I3I5**).

1.- Subconjuntos de tamaño **i=1**: **{I1, I2, I3, I4, I5}**

2.- Soporte/relevancia: **{S(I1) = 4; S(I2) = 3; S(I3) = 3; S(I4) = 1; S(I5) = 2}**

3.- Aplicamos el ***MinSupp***: **S1 = {I1; I2; I3; I5}** → Se excluye el I4-Pan y derivados.

4.- Construimos subconjuntos de tamaño **i=2** a partir de **S1**: **{I1I2; I1I3; I1I5; I2I3; I2I5; I3I5}**

5.- Aplicamos la segunda propiedad. Para **i=2** no se excluye ningún subconjunto.

6.- Calculamos su soporte/relevancia: **{I1I2 = 3; I1I3 = 3; I1I5 = 2; I2I3 = 2; I2I5 = 1; I3I5 = 2}**

7.- Aplicamos el ***MinSupp***: **S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}** → Se excluye el I2I5 y derivados.

8.- Construimos subconjuntos de tamaño **i=3** a partir de **S2**: **{I1I2I3; I1I2I5; I1I3I5; I2I3I5}**

9.- Aplicamos la segunda propiedad, eliminando **{I1I2I5; I2I3I5}**.

10.- Calculamos su soporte/relevancia: **{I1I2I3 = 2; I1I3I5 = 2}**

11.- Aplicamos el ***MinSupp***: **S3 = {I1I2I3; I1I3I5}**

12.- Construimos subconjuntos de tamaño **i=4** a partir de **S3**: **{I1I2I3I5}**

13.- Aplicamos la segunda propiedad, eliminando **{I1I2I3I5}** al contener un subconjunto de orden 3 no frecuente (**I1I2I5** ó **I2I3I5**).

¡¡¡Se evalúan 13 soportes en lugar de los 32 posibles!!!

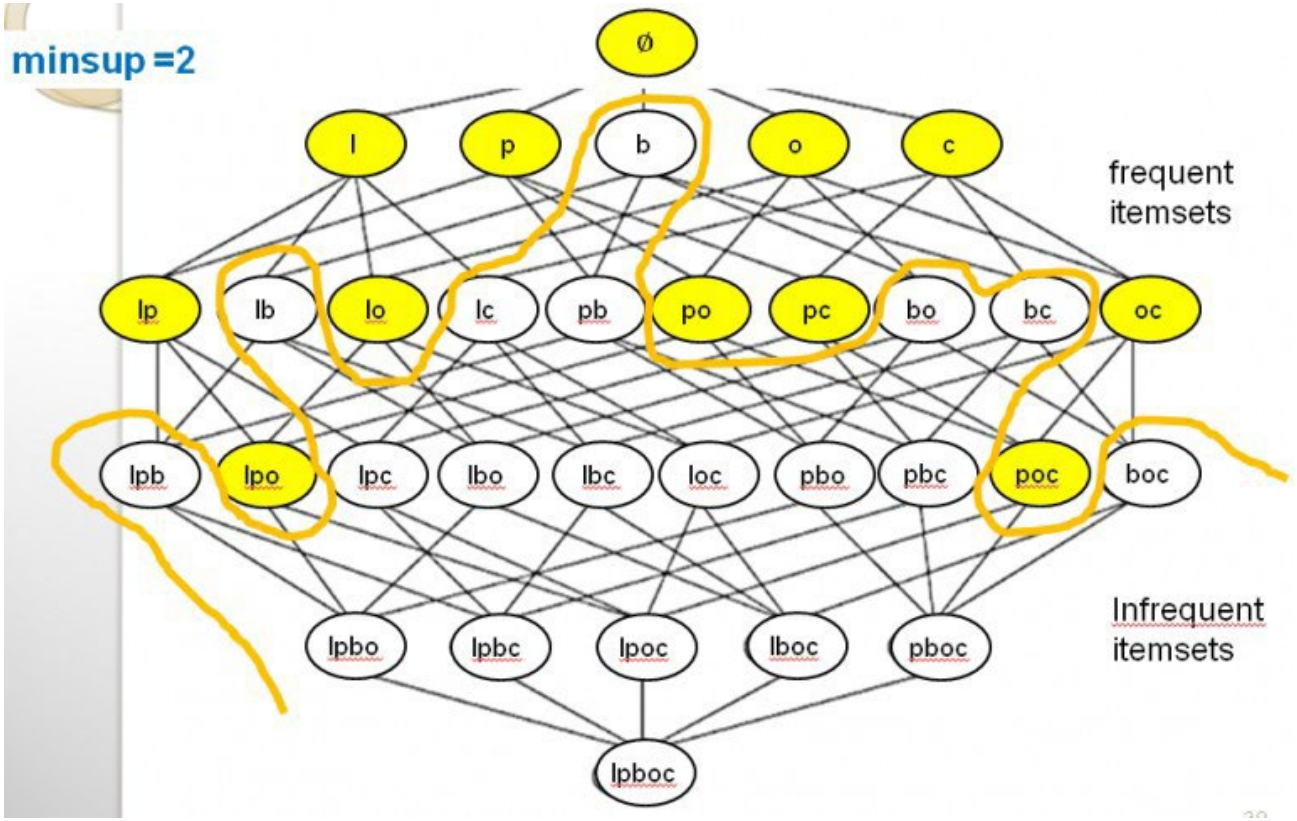
Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Definimos la cota para el soporte/relevancia: ***MinSupp = 2***

Item
Pasta = **p**
Limón = **l**
Naranja = **O**
Pan = **b**
Galletas = **c**

11/32 Itemsets

13/32 Candidatos



El algoritmo **APRIORI** obtiene el conjunto de itemsets frecuentes a partir de los cuales debemos obtener las reglas de asociación. Por ejemplo, para el itemset ***X={I1I2I3I4}*** tenemos las siguientes reglas posibles:

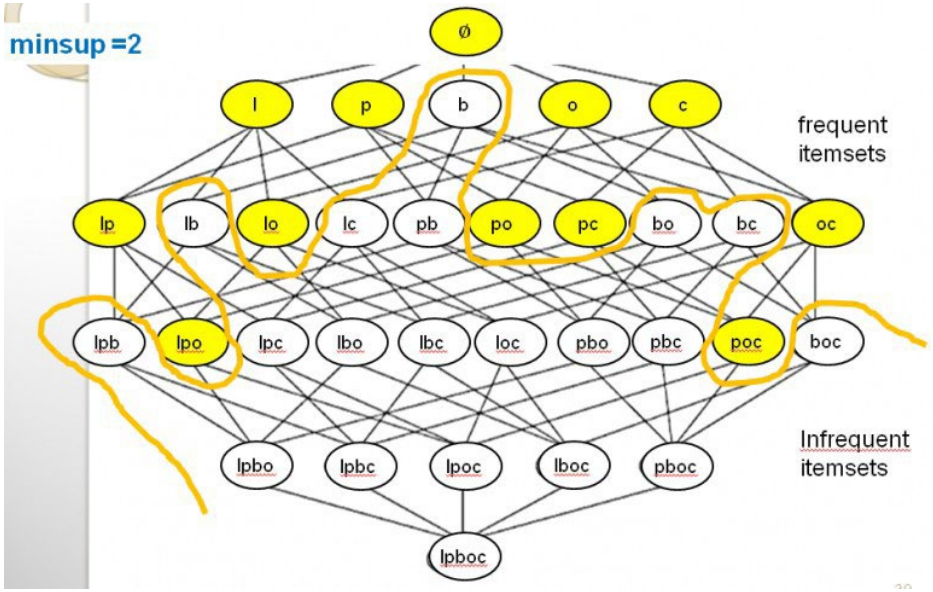
I1I2I3 → I4, I1I2I4 → I3, I1I3I4 → I2, I2I3I4 → I1 (4 reglas)

I1 → I2I3I4, I2 → I1I3I4, I3 → I1I2I4, I4 → I1I2I3 (4 reglas)

I1I2 → I3I4, I1I3 → I2I4, I1I4 → I2I3, I2I3 → I1I4, I2I4 → I1I3, I3I4 → I1I2 (6 reglas)

En general, para un itemset ***X*** con ***n*** elementos existen ***2^n-2*** reglas de asociación. Por tanto,

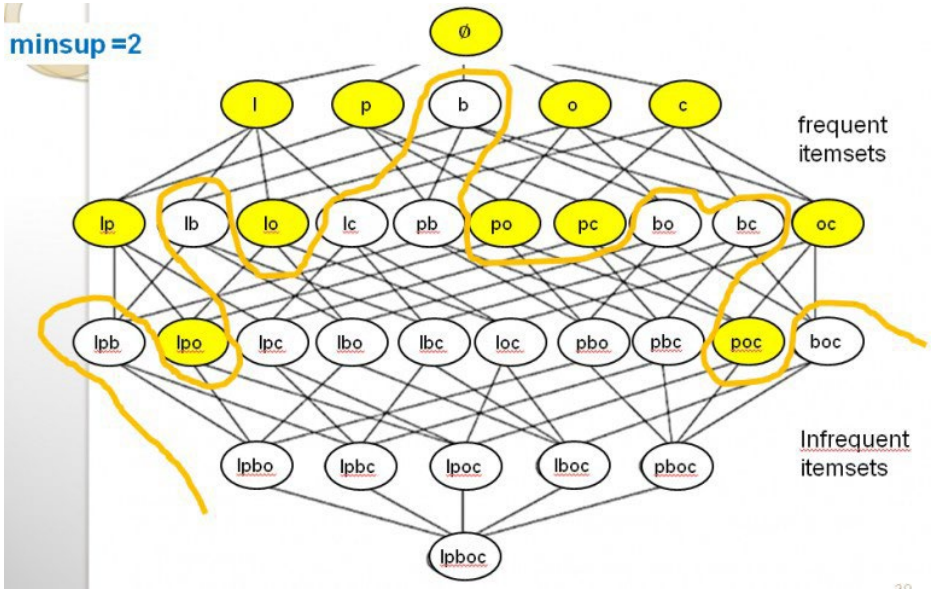
¿Cómo generar reglas de forma eficiente?



El algoritmo **APRIORI** obtiene el conjunto de itemsets frecuentes a partir de los cuales debemos obtener las reglas de asociación. Por ejemplo, para el itemset ***X={I1I2I3I4}*** tenemos las siguientes reglas posibles:
I1I2I3 → I4, I1I2I4 → I3, I1I3I4 → I2, I2I3I4 → I1 (4 reglas)
I1 → I2I3I4, I2 → I1I3I4, I3 → I1I2I4, I4 → I1I2I3 (4 reglas)
I1I2 → I3I4, I1I3 → I2I4, I1I4 → I2I3, I2I3 → I1I4, I2I4 → I1I3, I3I4 → I1I2 (6 reglas)
En general, para un itemset ***X*** con ***n*** elementos existen ***2^n-2*** reglas de asociación. Por tanto,

¿Cómo generar reglas de forma eficiente?

Uno de los criterios para filtrar las reglas es establecer una confianza mínima, ***MinConf***, siendo la confianza: ***C(X → Y) = S({X,Y})/S(X).***
Sin embargo, al igual que en el caso de los itemsets y el soporte, **¿la confianza cumple alguna propiedad que permita "podar" el árbol de posibles reglas?**



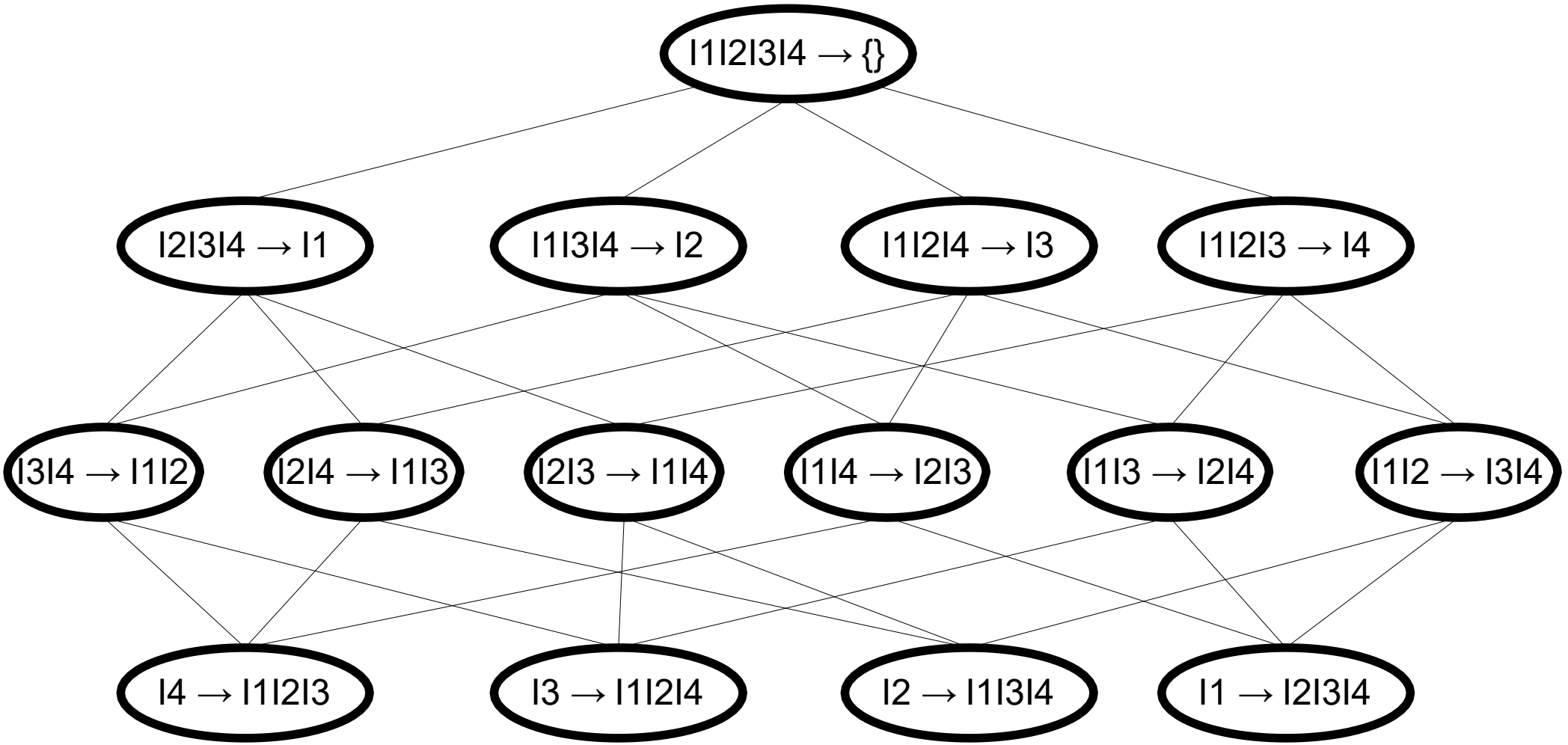
El algoritmo **APRIORI** obtiene el conjunto de itemsets frecuentes a partir de los cuales debemos obtener las reglas de asociación. Por ejemplo, para el itemset ***X={I1I2I3I4}*** tenemos las siguientes reglas posibles:
I1I2I3 → I4, I1I2I4 → I3, I1I3I4 → I2, I2I3I4 → I1 (4 reglas)
I1 → I2I3I4, I2 → I1I3I4, I3 → I1I2I4, I4 → I1I2I3 (4 reglas)
I1I2 → I3I4, I1I3 → I2I4, I1I4 → I2I3, I2I3 → I1I4, I2I4 → I1I3, I3I4 → I1I2 (6 reglas)
En general, para un itemset ***X*** con ***n*** elementos existen ***2^n-2*** reglas de asociación. Por tanto,

¿Cómo generar reglas de forma eficiente?

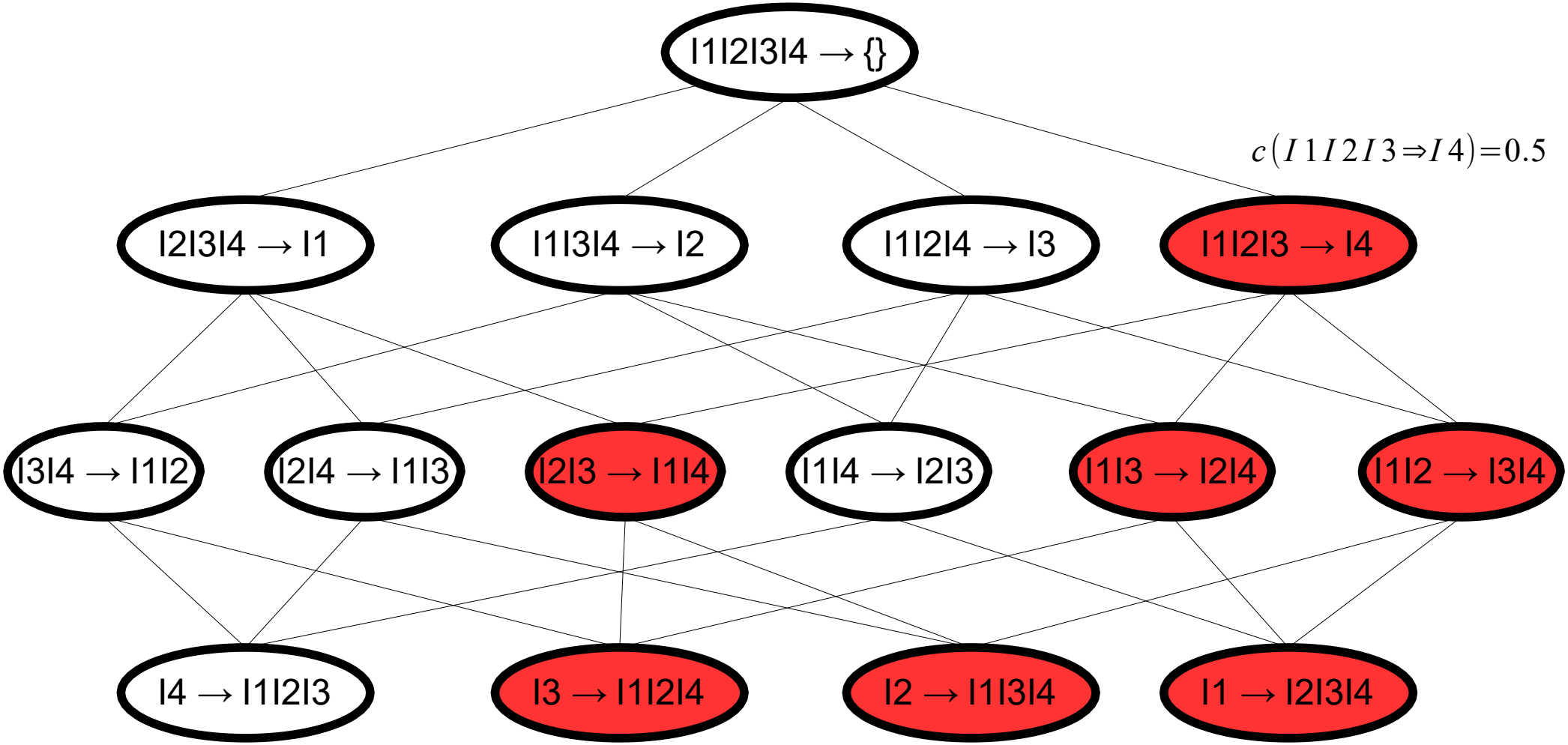
Uno de los criterios para filtrar las reglas es establecer una confianza mínima, ***MinConf***, siendo la confianza: ***C(X → Y) = S({X,Y})/S(X).***
Sin embargo, al igual que en el caso de los itemsets y el soporte, **¿la confianza cumple alguna propiedad que permita "podar" el árbol de posibles reglas?**

La confianza es antimonótona con respecto al número de items en el consecuente de la regla:

$$c(I1\,I2\,I3 \Rightarrow I4) \geq c(I1\,I2 \Rightarrow I3\,I4) \geq c(I1 \Rightarrow I2\,I3\,I4)$$



$$c(I1\,I2\,I3 \Rightarrow I4) \geq c(I1\,I2 \Rightarrow I3\,I4) \geq c(I1 \Rightarrow I2\,I3\,I4)$$



Por la propiedad de la confianza, al no cumplir el criterio de la confianza la regla I1I2I3 → I4, ninguna regla del subárbol inferior la cumplirá de modo que no es necesario evaluarlas.

$$c(I1 I2 I3 \Rightarrow I4) \geq c(I1 I2 \Rightarrow I3 I4) \geq c(I1 \Rightarrow I2 I3 I4)$$

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Definimos la cota para la confianza de la regla: ***MinConf = 0.75***

S3 = {I1I2I3; I1I3I5}

C(I1I2 → I3) = 2/3; C(I1I3 → I2) = 2/3; C(I2I3 → I1) = 2/2;

C(I1 → I2I3) = 2/4; C(I2 → I1I3) = 2/3; C(I3 → I1I2) = 2/3;

C(I1I3 → I5) = 2/3; C(I1I5 → I3) = 2/2; C(I3I5 → I1) = 2/2;

C(I1 → I3I5) = 2/4; C(I3 → I1I5) = 2/3; C(I5 → I1I3) = 2/2;

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Definimos la cota para la confianza de la regla: ***MinConf = 0.75***

S3 = {I1I2I3; I1I3I5}

~~**C(I1I2 → I3) = 2/3;**~~ ~~**C(I1I3 → I2) = 2/3;**~~ **C(I2I3 → I1) = 2/2;**

C(I1 → I2I3) = 2/4; C(I2 → I1I3) = 2/3; C(I3 → I1I2) = 2/3;

No era necesario estimar esta confianza ya que, por la propiedad de la confianza, ya podía concluirse que esas reglas no iban a cumplir el criterio de la confianza para el umbral dado.

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Definimos la cota para la confianza de la regla: ***MinConf = 0.75***

S3 = {I1I2I3; I1I3I5}

C(I2I3 → I1) = 2/2; C(I1I5 → I3) = 2/2; C(I3I5 → I1) = 2/2;

C(I5 → I1I3) = 2/2;

S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}

C(I1 → I2) = 3/4; C(I2 → I1) = 1; C(I1 → I3) = 3/4; C(I3 → I1) = 1; C(I5 → I1) = 1;

C(I2 → I3) = 2/3; C(I3 → I2) = 2/3; C(I3 → I5) = 2/3; C(I5 → I3) = 1;

Notar que, dado que siempre se da el item 1, las reglas con este item como consecuente tienen confianza 1 aun cuando no aportan información relevante, ¿podemos filtrar reglas en base a ese criterio?

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Definimos la cota para la confianza de la regla: ***MinConf = 0.75***

S3 = {I1I2I3; I1I3I5}

C(I2I3 → I1) = 2/2; C(I1I5 → I3) = 2/2; C(I3I5 → I1) = 2/2;

C(I5 → I1I3) = 2/2;

S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}

C(I1 → I2) = 3/4; C(I2 → I1) = 1; C(I1 → I3) = 3/4; C(I3 → I1) = 1; C(I5 → I1) = 1;

C(I2 → I3) = 2/3; C(I3 → I2) = 2/3; C(I3 → I5) = 2/3; C(I5 → I3) = 1;

Notar que, dado que siempre se da el item 1, las reglas con este item como consecuente tienen confianza 1 aun cuando no aportan información relevante, ¿podemos filtrar reglas en base a ese criterio? Medidas de interés de la regla: ***lift(X→Y) = P(Y|X)/P(Y)=P(X,Y)/P(X)P(Y)***

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Definimos la cota para la confianza de la regla: ***MinConf = 0.75***

S3 = {I1I2I3; I1I3I5}

C(I2I3 → I1) = 2/2 (1); C(I1I5 → I3) = 2/2 (1.33); C(I3I5 → I1) = 2/2 (1);

C(I5 → I1I3) = 2/2 (1.33);

S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}

C(I1 → I2) = 3/4 (1); C(I1 → I3) = 3/4 (1);

C(I2 → I1) = 1 (1); C(I2 → I3) = 2/3 (0.88);

C(I3 → I1) = 1 (1); C(I3 → I2) = 2/3 (0.88); C(I3 → I5) = 2/3 (1.33);

C(I5 → I1) = 1 (1); C(I5 → I3) = 1 (1.33);

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Definimos la cota para la confianza de la regla: ***MinConf = 0.75***

S3 = {I1I2I3; I1I3I5}

C(I2I3 → I1) = 2/2 (1); C(I1I5 → I3) = 2/2 (1.33); C(I3I5 → I1) = 2/2 (1);

Independientes

C(I5 → I1I3) = 2/2 (1.33);

S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}

C(I1 → I2) = 3/4 (1); C(I1 → I3) = 3/4 (1);

C(I2 → I1) = 1 (1); C(I2 → I3) = 2/3 (0.88)

Relación negativa

C(I3 → I1) = 1 (1); C(I3 → I2) = 2/3 (0.88); C(I3 → I5) = 2/3 (1.33);

Relación positiva

C(I5 → I1) = 1 (1); C(I5 → I3) = 1 (1.33);

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Otra medida es la ***convicción de la regla***: ***conv(X → Y)=(1-S(Y))/(1-C(X → Y))***

S3 = {I1I2I3; I1I3I5}

C(I2I3 → I1) = 2/2 (1, 0); C(I1I5 → I3) = 2/2 (1.33, Inf); C(I3I5 → I1) = 2/2 (1, 0);

C(I5 → I1I3) = 2/2 (1.33, Inf);

S2 = {I1I2; I1I3; I1I5; I2I3; I3I5}

C(I1 → I2) = 3/4 (1, 1); C(I2 → I1) = 1 (1, 0); C(I2 → I3) = 2/3 (0.88, 0.375);

Notar que la convicción nula se corresponde con reglas sin interés, valores **superiores/inferiores** a 1 indica que la regla predecirá correctamente **más/menos** casos que si la relación entre ambos itemsets fuera aleatoria.

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Filtramos las reglas en base a su confianza (***C(X → Y) = S({X,Y})/S(X)***)

Definimos la cota para la confianza de la regla: ***MinConf = 0.75***

C(I1I5 → I3) = 2/2 (1.33); C(I5 → I1I3) = 2/2 (1.33);

C(I1 → I2) = 3/4 (1); C(I1 → I3) = 3/4 (1);

C(I2 → I3) = 2/3 (0.88); C(I3 → I2) = 2/3 (0.88); C(I3 → I5) = 2/3 (1.33);

C(I5 → I3) = 1 (1.33);

Transacción	I1-Pasta	I2-Limo
T1	1	1
T2	1	1
T3	1	0
T4	1	1

Filtramos las reglas en base a su confianz

Definimos la cota para la confianza de la r

C(I1I5 → I3) = 2/2 (1.33); C(I5 → I1I3) = 2

C(I1 → I2) = 3/4 (1); C(I1 → I3) = 3/4 (1);

C(I2 → I3) = 2/3 (0.88); C(I3 → I2) = 2/3 (

C(I5 → I3) = 1 (1.33);

Measure	Formula
ϕ-coefficient	$\frac{P(A,B)-P(A)P(B)}{\sqrt{P(A)P(B)(1-P(A))(1-P(B))}}$
Goodman-Kruskal's (λ)	$\frac{\sum_j \max_k P(A_j,B_k)+\sum_k \max_j P(A_j,B_k)-\max_j P(A_j)-\max_k P(B_k)}{2-\max_j P(A_j)-\max_k P(B_k)}$
Odds ratio (α)	$\frac{P(A,B)P(\overline{A},\overline{B})}{P(A,\overline{B})P(\overline{A},B)}$
Yule's Q	$\frac{P(A,B)P(\overline{AB})-P(A,\overline{B})P(\overline{A},B)}{P(A,B)P(\overline{AB})+P(A,\overline{B})P(\overline{A},B)} = \frac{\alpha-1}{\alpha+1}$
Yule's Y	$\frac{\sqrt{P(A,B)P(\overline{AB})}-\sqrt{P(A,\overline{B})P(\overline{A},B)}}{\sqrt{P(A,B)P(\overline{AB})}+\sqrt{P(A,\overline{B})P(\overline{A},B)} = \frac{\sqrt{\alpha}-1}{\sqrt{\alpha}+1}$
Kappa (κ)	$\frac{P(A,B)+P(\overline{A},\overline{B})-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A)P(B)-P(\overline{A})P(\overline{B})}$
Mutual Information (M)	$\frac{\sum_i \sum_j P(A_i,B_j)\log\frac{P(A_i,B_j)}{P(A_i)P(B_j)}}{\min(-\sum_i P(A_i)\log P(A_i), -\sum_j P(B_j)\log P(B_j))}$
J-Measure (J)	$\max\left(P(A,B)\log(\frac{P(B\|A)}{P(B)}) + P(A\overline{B})\log(\frac{P(\overline{B}\|A)}{P(\overline{B})}),\right.$ $\left. P(A,B)\log(\frac{P(A\|B)}{P(A)}) + P(\overline{A}B)\log(\frac{P(\overline{A}\|B)}{P(\overline{A})})\right)$
Gini index (G)	$\max\Big(P(A)[P(B\|A)^2+P(\overline{B}\|A)^2]+P(\overline{A})[P(B\|\overline{A})^2+P(\overline{B}\|\overline{A})^2]$ $-P(B)^2-P(\overline{B})^2,$ $P(B)[P(A\|B)^2+P(\overline{A}\|B)^2]+P(\overline{B})[P(A\|\overline{B})^2+P(\overline{A}\|\overline{B})^2]$ $-P(A)^2-P(\overline{A})^2\Big)$
Support (s)	$P(A,B)$
Confidence (c)	$\max(P(B\|A), P(A\|B))$
Laplace (L)	$\max\left(\frac{NP(A,B)+1}{NP(A)+2}, \frac{NP(A,B)+1}{NP(B)+2}\right)$
Conviction (V)	$\max\left(\frac{P(A)P(\overline{B})}{P(A\overline{B})}, \frac{P(B)P(\overline{A})}{P(B\overline{A})}\right)$
Interest (I)	$\frac{P(A,B)}{P(A)P(B)}$
cosine (IS)	$\frac{P(A,B)}{\sqrt{P(A)P(B)}}$
Piatetsky-Shapiro's (PS)	$P(A,B)-P(A)P(B)$
Certainty factor (F)	$\max\left(\frac{P(B\|A)-P(B)}{1-P(B)}, \frac{P(A\|B)-P(A)}{1-P(A)}\right)$
Added Value (AV)	$\max(P(B\|A)-P(B), P(A\|B)-P(A))$
Collective strength (S)	$\frac{P(A,B)+P(\overline{AB})}{P(A)P(B)+P(\overline{A})P(\overline{B})} \times \frac{1-P(A)P(B)-P(\overline{A})P(\overline{B})}{1-P(A,B)-P(\overline{AB})}$
Jaccard (ζ)	$\frac{P(A,B)}{P(A)+P(B)-P(A,B)}$
Klosgen (K)	$\sqrt{P(A,B)}\max(P(B\|A)-P(B), P(A\|B)-P(A))$

Existen muchas más medidas de interés con diferentes propiedades y aplicaciones.

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Aún filtrando las reglas en base a los criterios anteriores, pueden darse reglas asociadas a una misma propiedad, o **redundantes**.

$$C\left(X_1 \Rightarrow Y_1\right) = \left\{X_2 \Rightarrow Y_2 \middle| X_1 \subseteq X_2 \wedge X_2\, Y_2 \subseteq X_1\, Y_1\right\}$$

Estas reglas deben también eliminarse para obtener un conjunto mínimo representativo de las relaciones relevantes.

C(I1I5 → I3) = 2/2 (1.33); C(I5 → I1I3) = 2/2 (1.33);

C(I1 → I2) = 3/4 (1); C(I1 → I3) = 3/4 (1);

C(I2 → I3) = 2/3 (0.88); C(I3 → I2) = 2/3 (0.88); C(I3 → I5) = 2/3 (1.33);

C(I5 → I3) = 1 (1.33);

Aún filtrando las reglas en base a los criterios anteriores, pueden darse reglas asociadas a una misma propiedad, o **redundantes**.

$$C\left(X_1 \Rightarrow Y_1\right) = \left\{X_2 \Rightarrow Y_2 \middle| X_1 \subseteq X_2 \wedge X_2\, Y_2 \subseteq X_1\, Y_1\right\}$$

Estas reglas deben también eliminarse para obtener un conjunto mínimo representativo de las relaciones relevantes.

C(I1I5 → I3) = 2/2 (1.

C(I1 → I2) = 3/4 (1); (

C(I2 → I3) = 2/3 (0.88

C(I5 → I3) = 1 (1.33);

is.redundant {arules} R Documentation

Find Redundant Rules

Description

Provides the generic functions and the S4 method `is.redundant` to find redundant rules.

Usage

```
is.redundant(x, ...)
## S4 method for signature 'rules'
is.redundant(x, measure = "confidence")
```

Arguments

`x`	a set of rules.
`measure`	measure used to check for redundancy.
`...`	additional arguments.

Details

A rule is redundant if a more general rules with the same or a higher confidence exists. That is, a more specific rule is redundant if it is only equally or even less predictive than a more general rule. A rule is more general if it has the same RHS but one or more items removed from the LHS. Formally, a rule *X -> Y* is redundant if

for some X' subset X, conf(X' -> Y) >= conf(X -> Y).

This is equivalent to a negative or zero *improvement* as defined by Bayardo et al. (2000). In this implementation other measures than confidence, e.g. improvement of lift, can be used as well.

Transacción	I1-Pasta	I2-Limon	I3-Naranja	I4-Pan	I5-Galletas
T1	1	1	1	1	0
T2	1	1	0	0	0
T3	1	0	1	0	1
T4	1	1	1	0	1

Aún filtrando las reglas en base a los criterios anteriores, pueden darse reglas asociadas a una misma propiedad, o **redundantes**.

$$C\left(X_1 \Rightarrow Y_1\right) = \left\{X_2 \Rightarrow Y_2 \middle| X_1 \subseteq X_2 \wedge X_2\, Y_2 \subseteq X_1\, Y_1\right\}$$

Estas reglas deben también eliminarse para obtener un conjunto mínimo representativo de las relaciones relevantes.

~~C(I1I5 → I3)~~ = 2/2 (1.33); ~~C(I5 → I1I3)~~ = 2/2 (1.33); ← Redundantes y con C(I5 → I3).

C(I1 → I2) = 3/4 (1); C(I1 → I3) = 3/4 (1);

C(I5 → I3) = 1 (1.33);

Lift: Given two items, A and B, lift indicates whether there is a relationship between A and B, or whether the two items are occuring together in the same orders simply by chance (ie: at random). In summary, lift can take on the following values:

* **lift = 1** implies no relationship between A and B (ie: A and B occur together only by chance)

***lift > 1** implies that there is a positive relationship between A and B (ie: A and B occur together more often than random)

***lift < 1** implies that there is a negative relationship between A and B (ie: A and B occur together less often than random)

Confidence: Given two items, A and B, confidence measures the percentage of times that item B is purchased, given that item A was purchased. Confidence values range from 0 to 1, where 0 indicates that B is never purchased when A is purchased, and 1 indicates that B is always purchased whenever A is purchased.

Support: This is the percentage of orders that contains the item set.

Leverage: Leverage measures the difference of X and Y appearing together in the data set and what would be expected if X and Y where statistically dependent. The rational in a sales setting is to find out how many more units (items X and Y together) are sold than expected from the independent sells.

https://www.kaggle.com/xvivancos/market-basket-analysis

Reglas de Asociación

- Introducción – Objetivo
- Modelo Formal - Conceptos Básicos
- Reglas de Asociación
- Algoritmo Apriori
- **Librería aRules/arulesViz**
- Redes Probabilísticas → Introducción
- Algoritmo Eclat

Los paquetes ***aRules*** y ***arulesViz*** son dos librerías de R para el aprendizaje de Reglas de Asociación a partir de un conjunto de datos.

Introduction to arules – A computational environment for mining association rules and frequent item sets

Michael Hahsler
Southern Methodist University

Bettina Grün
Johannes Keppler University Linz

Kurt Hornik
Wirtschaftsuniversität Wien

Christian Buchta
Wirtschaftsuniversität Wien

Visualizing Association Rules: Introduction to the R-extension Package arulesViz

Michael Hahsler
Southern Methodist University

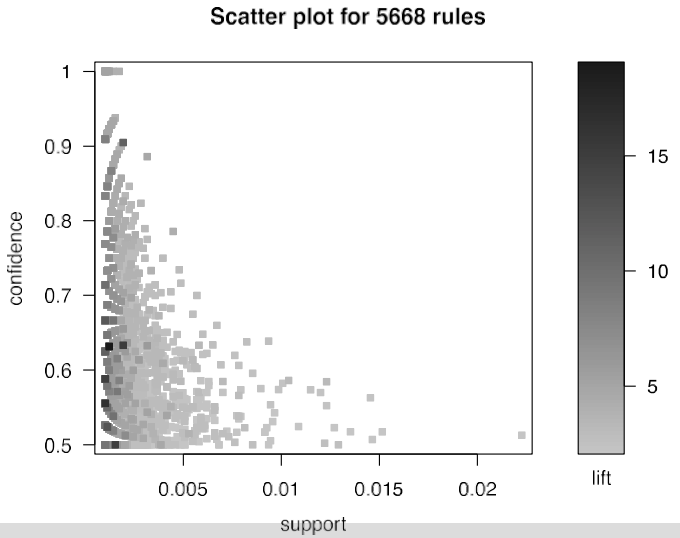
Sudheer Chelluboina
Southern Methodist University

A smaller dataset "Groceries" from **arulesViz** package will be used in the course.


```
transactions as itemMatrix in sparse format with
 9835 rows (elements/itemsets/transactions) and
 169 columns (items) and a density of 0.02609146

most frequent items:
      whole milk other vegetables       rolls/buns
            2513             1903             1809
```

```
library("arulesViz")
data("Groceries")
```

```
transactions as itemMatrix in sparse format with
 9835 rows (elements/itemsets/transactions) and
 169 columns (items) and a density of 0.02609146

most frequent items:
      whole milk other vegetables       rolls/buns
            2513             1903             1809
          yogurt          (Other)
            1372            34055
```




```
> rules <- apriori(Groceries, parameter=list(support=0.001, confidence=0.5))
> inspect(head(sort(rules, by ="lift"),3))
```

$$\text{lift}(X \Rightarrow Y) = \text{supp}(X \cup Y)/(\text{supp}(X)\text{supp}(Y))$$

```
  lhs                          rhs                  support confidence     lift
1 {Instant food products,
   soda}                    => {hamburger meat} 0.001220132  0.6315789 18.99565
2 {soda,
   popcorn}                 => {salty snack}    0.001220132  0.6315789 16.69779
3 {flour,
   baking powder}           => {sugar}          0.001016777  0.5555556 16.40807
```

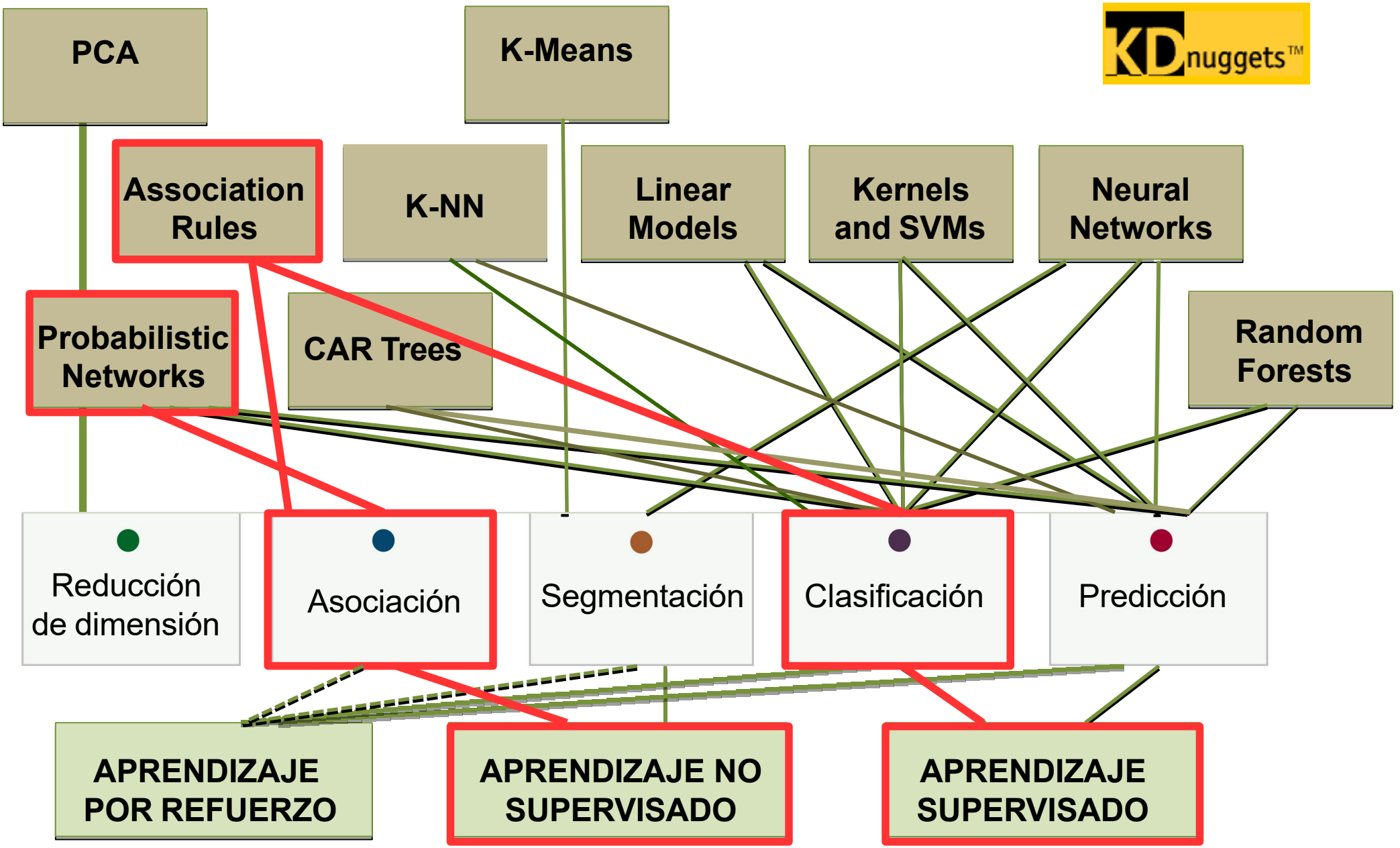
A smaller dataset “Groceries” from **arulesViz** package will be used in the course.

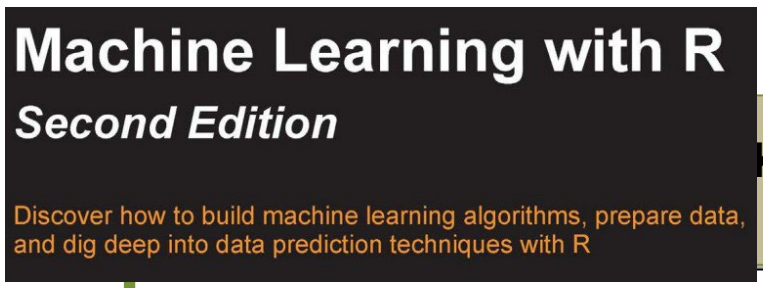

```
transactions as itemMatrix in sparse format with
 9835 rows (elements/itemsets/transactions) and
 169 columns (items) and a density of 0.02609146

most frequent items:
      whole milk other vegetables       rolls/buns
            2513             1903             1809
```

Reglas de Asociación

- Introducción – Objetivo
- Modelo Formal - Conceptos Básicos
- Reglas de Asociación
- Algoritmo Apriori
- Librería aRules/arulesViz
- **Reglas de Clasificación → Árboles de Clasificación**
- Algoritmo Eclat

https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html





Understanding classification rules

Classification rules represent knowledge in the form of logical if-else statements that assign a class to unlabeled examples. They are specified in terms of an **antecedent** and a **consequent**; these form a hypothesis stating that "if this happens, then that happens." A simple rule might state, "if the hard drive is making a clicking sound, then it is about to fail." The antecedent comprises certain combinations of feature values, while the consequent specifies the class value to assign when the rule's conditions are met.

Rule learners are often used in a manner similar to decision tree learners. Like decision trees, they can be used for applications that generate knowledge for future action, such as:

- Identifying conditions that lead to a hardware failure in mechanical devices
- Describing the key characteristics of groups of people for customer segmentation
- Finding conditions that precede large drops or increases in the prices of shares on the stock market

[149]

The 1R algorithm

The RIPPER algorithm

Decission Trees

Rules from decision trees

Association Rules

K-NN

Probabilistic Networks

CAR Trees

Random Forests

Reducción de dimensión

Asociación

Segmentación

Clasificación

Predicción

APRENDIZAJE POR REFUERZO

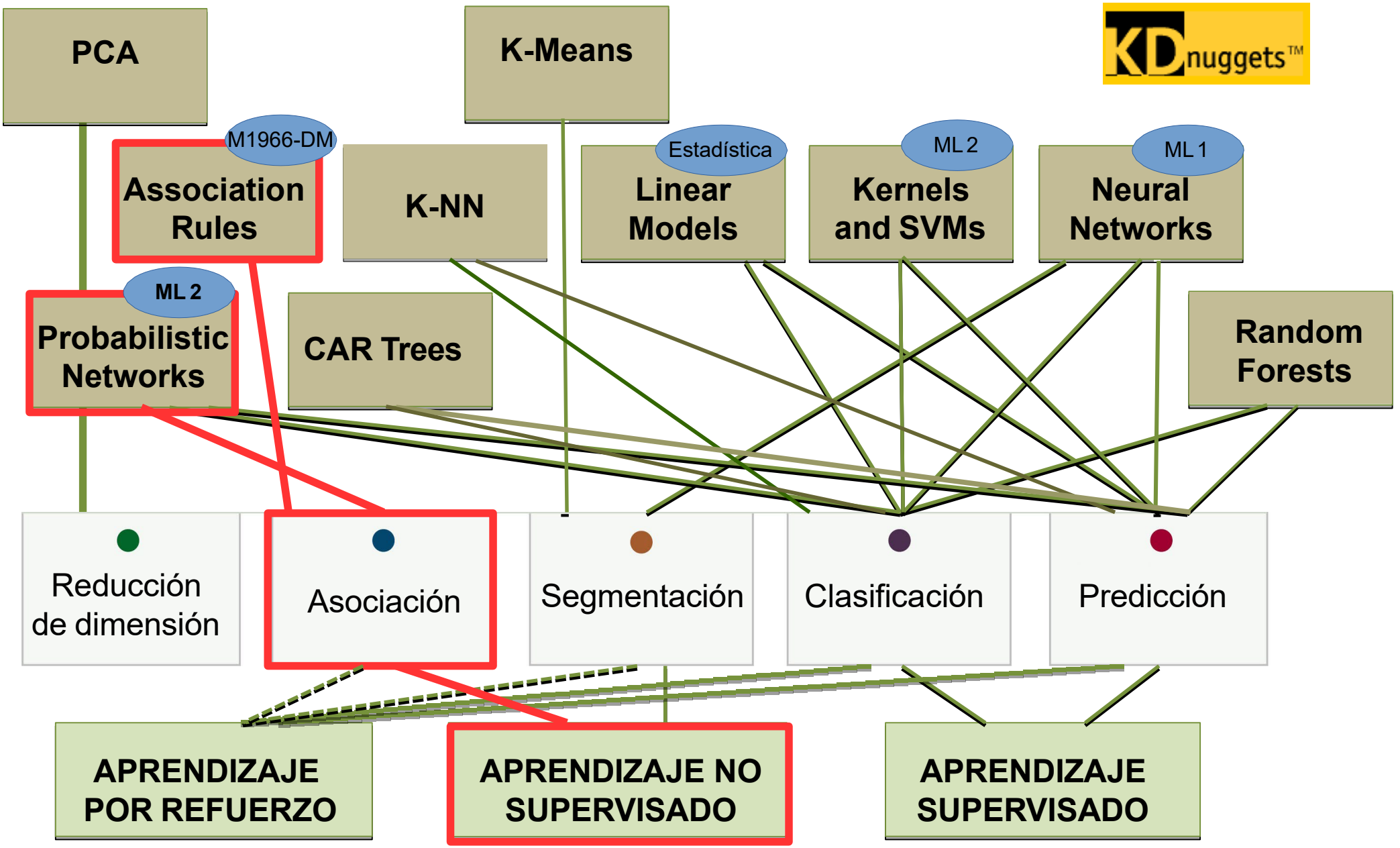
APRENDIZAJE NO SUPERVISADO

APRENDIZAJE SUPERVISADO

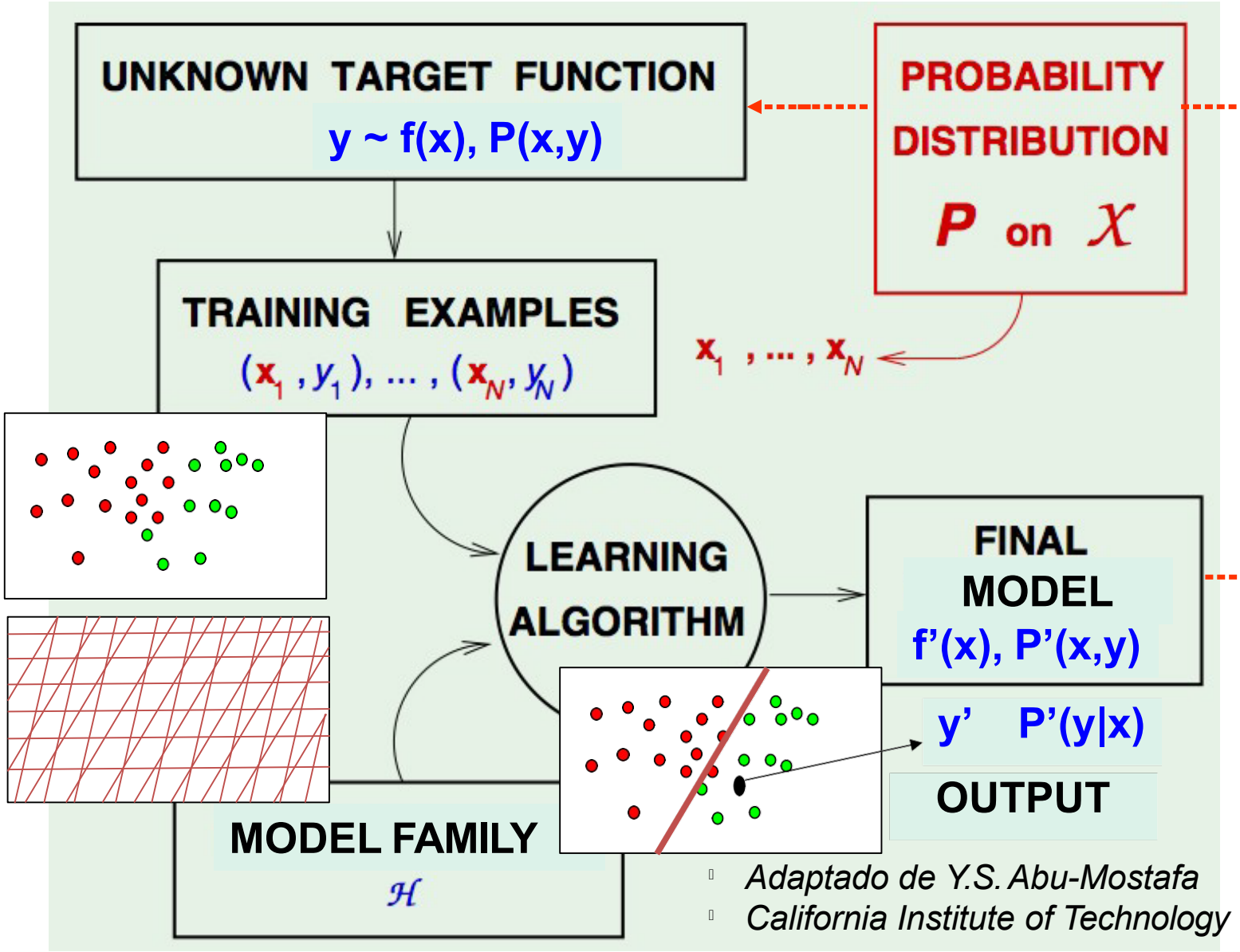
Reglas de Asociación

- Introducción – Objetivo
- Modelo Formal - Conceptos Básicos
- Reglas de Asociación
- Algoritmo Apriori
- Librería aRules/arulesViz
- **Redes Probabilísticas → Introducción**
- Algoritmo Eclat

https://www.kdnuggets.com/2017/10/top-10-machine-learning-algorithms-beginners.html



Generalization is the most important feature for data driven systems: They must perform "well" when applied to new data (**cross-validation**).



- *Adaptado de Y.S. Abu-Mostafa*
- *California Institute of Technology*

x	y	z	$p(x, y, z)$
0	0	0	0.12
0	0	1	0.18
0	1	0	0.04
0	1	1	0.16
1	0	0	0.09
1	0	1	0.21
1	1	0	0.02
1	1	1	0.18

La especificación directa de esta función de probabilidad no es posible en casos prácticos (10^{25} parámetros para 100 variables).

P(X,Y) = P(X) P(Y) passo da 4 combinazioni a 1, ovvero supponiamo di avere variabili binarie, a sx abbiamo {0,0}, {0,1}, {1,0}, {1,1} (a dx 1*1??)

Una solución es construir modelos con menos parámetros, limitando el número de posibles dependencias entre variables **Factorización:**

$$p(A \mid B) = \frac{p(A,\ B)}{p(B)} = \frac{p(B \mid A)\, p(A)}{p(B)}$$

¿Cómo se construye un modelo probabilístico que tenga un conjunto de independencias dado?

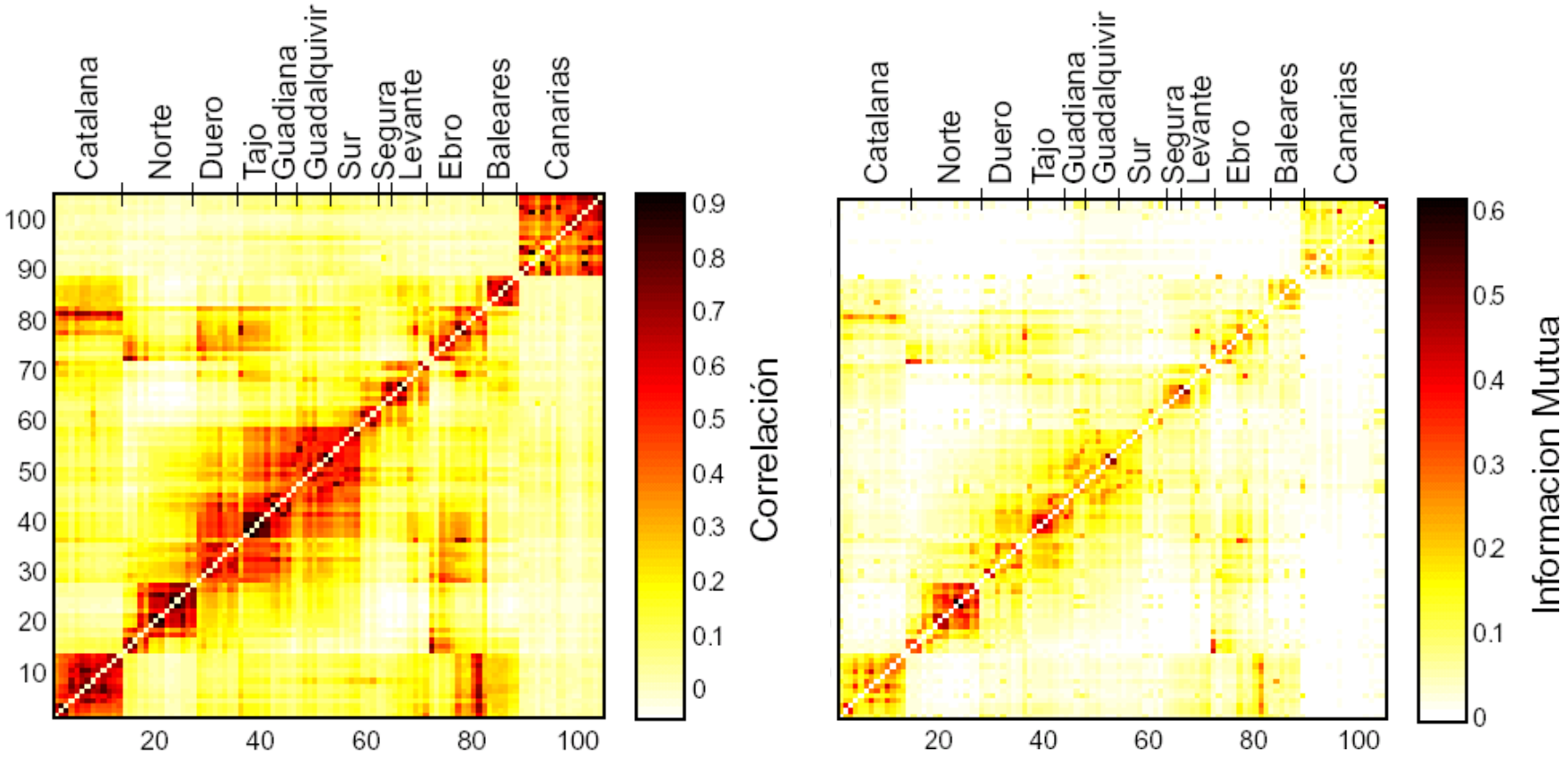
$$I(X_3, X_1|X_2) \text{ and } I(X_4, \{X_1, X_3\}|X_2).$$

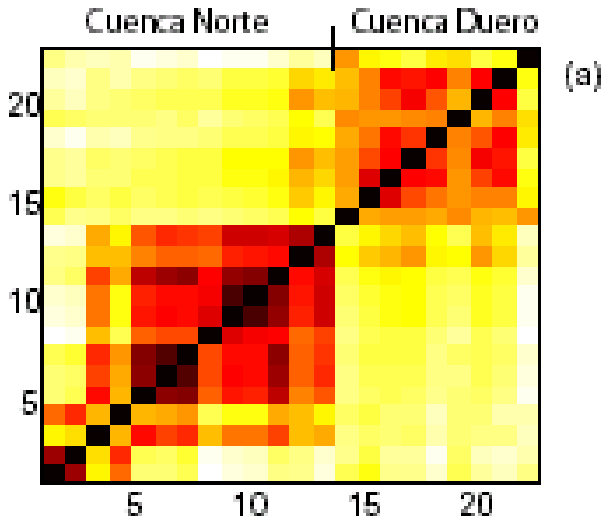
$$p(x_3|x_1, x_2) = p(x_3|x_2),$$

$$p(x_4|x_1, x_2, x_3) = p(x_4|x_2).$$

$$p(x_1, \ldots, x_4) = p(x_1)p(x_2|x_1)p(x_3|x_2)p(x_4|x_2).$$

La dependencia entre dos variables se puede calcular fácilmente con distintas medidas, como la correlación o al información mutua. Sin embargo, esta información es parcial.



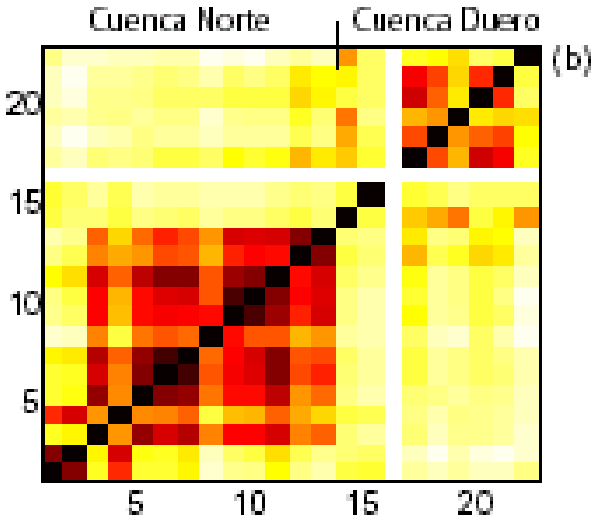


Es necesario el concepto de (in)dependencia condicional.

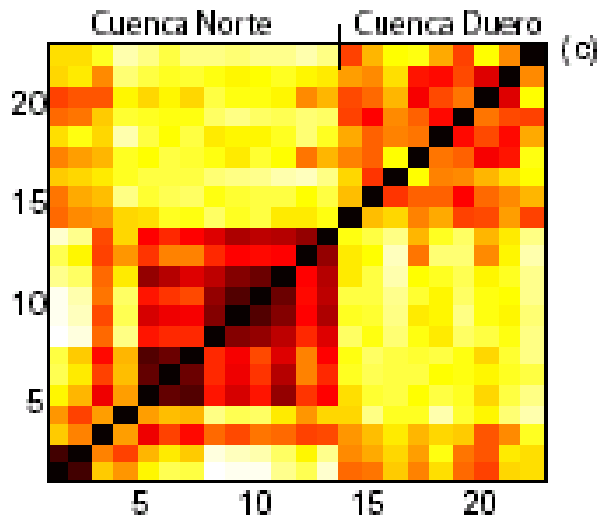
I(X,Y|Z) si

$$P_z(Y|X)=P_z(Y)$$

$$P(Y|X,Z)=P(Y|Z)$$



Precip(Palencia)=0 mm

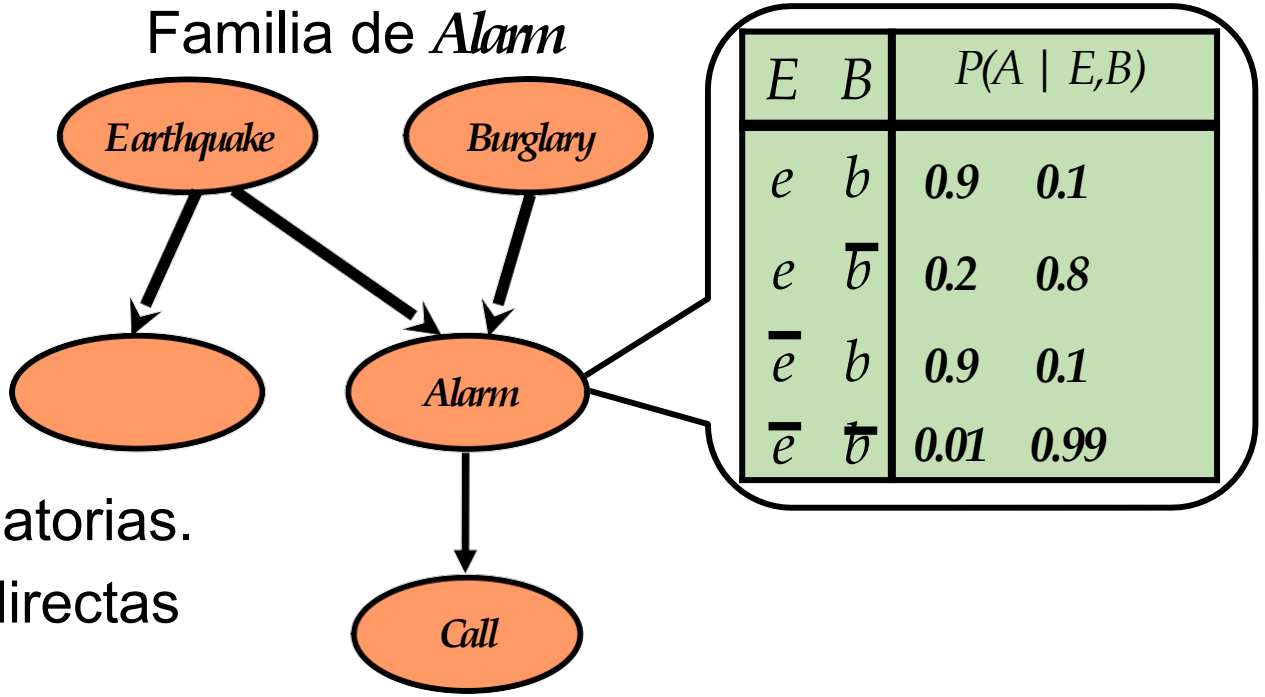


Precip(Palencia)>10 mm

La correlación parcial tiene en cuanta la relación entre dos variables, una vez que se ha eliminado la dependencia con una tercera.

Representación compacta de una función de probabilidad conjunta mediante independencia condicional.

Parte cualitativa:
Grafo dirigido acíclico (DAG), o grafos no dirigos (redes Markov)



Nodos – variables aleatorias.
Aristas – influencias directas

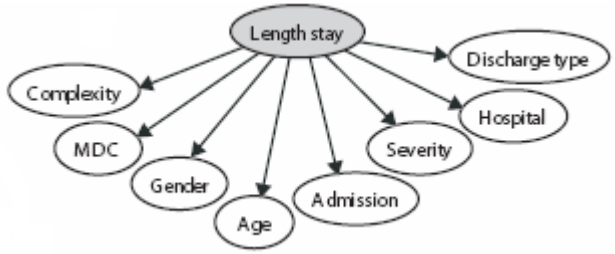
Junto con:
Factorización de una única función de probabilidad

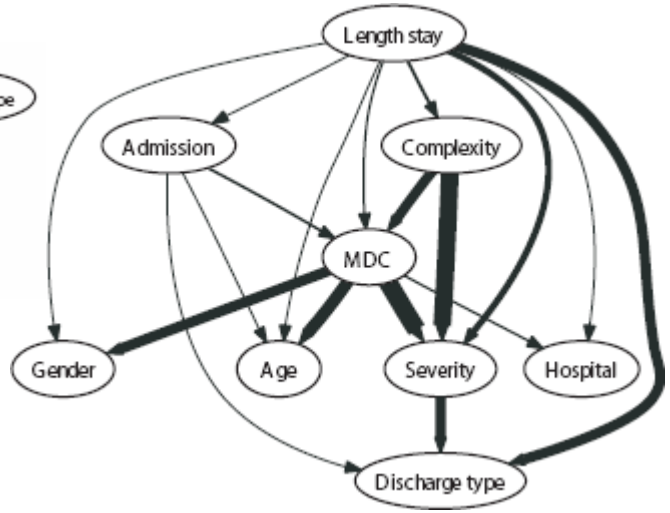
Parte cuantitativa:
Conjunto de funciones/tablas de probabilidad.

$$P(B,E,A,C,R) = P(B)P(E)P(A|B,E)P(R|E)P(C|A)$$

2009	ACTUALIZACIÓN Y MEJORA DEL DISEÑO DE LAS HERRAMIENTAS DE SOPORTE PARA EL ANÁLISIS DEL REGISTRO DE ALTAS DE HOSPITALIZACIÓN DE LOS HOSPITALES DEL SISTEMA NACIONAL DE SALUD	Director	MINISTERIO DE SANIDAD Y CONSUMO	MINISTERIO DE SANIDAD Y CONSUMO

1. Estancia Media
2. Tasa de Mortalidad
3. Tasa de Reingresos (a 30 días)
4. Tasa de Infección Nosocomial
5. Estancia Media Preoperatoria
6. Tasa de Cesáreas
Conjunto de indicadores

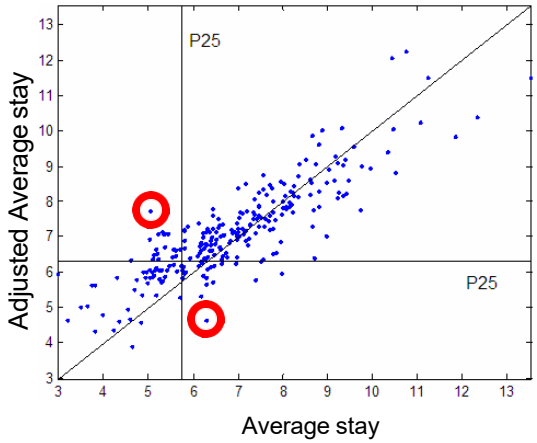




$$\frac{1}{\#(X)\#(\mathbf{Z})}\sum_{x,y,\mathbf{z}} P(y|x,\mathbf{z}) log_2\left(\frac{P(y|x,\mathbf{z})}{\frac{1}{\#(X)}\sum_x P(y|x,\mathbf{z})}\right)$$

RELACIONADAS CON LA ENFERMEDAD
1. Complejidad (medido a través del peso español de GRD-AP v 18)
2. Severidad (medido a través de GRD refinados)
3. Categoría Diagnostica Mayor de GRD-AP v 18
4. Tipo de GRD: medico, quirúrgico, indeterminado (Solo aplicado al indicador de infección nosocomial)
RELACIONADAS CON EL PACIENTE, O CON EL FUNCIONAMIENTO HOSPITALARIO
1. Edad
2. Sexo
3. Tipo de ingreso
4. Tipo de alta
5. Tipo de hospital
Variables de influencia propuestas para el análisis

Nuevo sistema genérico para el ajuste de indicadores sin recurrir a regresiones, etc.



1 30-60 mins

Revisar la práctica realizada en clase con el ejemplo teórico y ambos algoritmos, APRIORI y **ECLAT**.

Ejemplo realizado en clase:

Inicialmente consideraremos el ejemplo realizado en clase para familiarizarnos con los comandos y sus opciones para luego aplicarlo en un problema más general. Para ello, primero definiremos la tabla de transacciones del problema:

In [8]:
```
table <- list(c("p","l","O","b"), c("p","l"), c("p","O","c"), c("p","l","O","c"))
transactions <- as(table, "transactions")
inspect(transactions)
```

```
    items
[1] {b,l,O,p}
[2] {l,p}
[3] {c,O,p}
[4] {c,l,O,p}
```

Podemos ver el soporte, absoluto o relativo: de cada item:

In [9]:
```
itemFrequency(transactions, type="a")
itemFrequency(transactions, type="r")
itemFrequencyPlot(transactions)
```