

# REGLAS DE ASOCIACIÓN.

## EQUIPO 7

TELLO ESPARZA JESUS OMAR

1798181

MENDOZA SÁNCHEZ EVELIN YAMILE

1795417

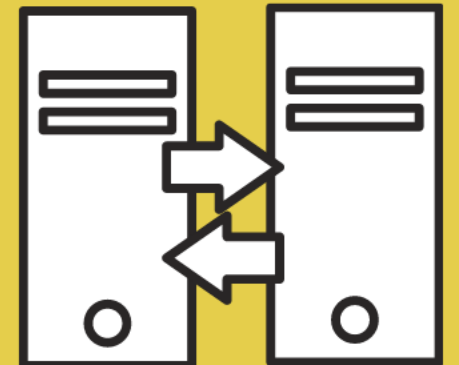
RAMIREZ ARELLANO NATHANAEL

1745134

TÉCNICAS DE MINERÍA DE DATOS.

# TÉCNICA DESCRIPTIVA.

El objetivo es encontrar patrones que den un resumen de las relaciones ocultas dentro de los datos. Descubre las características más importante de las bases de datos.





---

## REGLAS DE ASOCIACIÓN

Descubrir hechos que ocurren en común dentro de un conjunto de datos. Relación relevante entre los eventos.

Describe el análisis y la presentación de reglas 'fuertes' descubiertas en bases de datos utilizando diferentes medidas de interés. Se presentó un trabajo en el que indicaban las reglas de asociación que descubrían las relaciones entre los datos recopilados a gran escala en los sistemas de terminales de punto de venta de unos supermercados. Por ejemplo, la siguiente regla:

**{cebolla,vegetales}=>{carne}**

Encontrada en los datos de ventas de un supermercado, indicaría que un consumidor que compra cebollas y verdura a la vez, es probable que compre también carne. Esta información se puede utilizar como base para tomar decisiones sobre marketing como precios promocionales para ciertos productos o dónde ubicar estos dentro del supermercado

# Ejemplo práctico.

Para ilustrar estos conceptos véase el siguiente ejemplo sobre ventas en un supermercado. El conjunto de items es:

**{Leche,Pan,Mantequilla,Cerveza}**

Un ejemplo de regla para el supermercado podría ser:

**{Leche,Pan} => {Mantequilla}**

Base de datos con 4 items y 5 transacciones

ID	Leche	Pan	Mantequilla	Cerveza
1	1	1	0	0
2	0	1	1	0
3	0	0	0	1
4	1	1	1	0
5	0	1	0	0

Significaría que si el cliente compró 'leche' y 'pan' también compró 'mantequilla', es decir, según la especificación formal anterior se tendría que:

**X={Leche,Pan}**

**Y={Mantequilla}**

Los conjuntos de items X y Y se denominan respectivamente "antecedente" (o parte izquierda) y "consecuente" (o parte derecha) de la regla.

## **Reglas significativas, 'soporte' y 'confianza'**

Para seleccionar reglas interesantes del conjunto de todas las reglas posibles que se pueden derivar de un conjunto de datos se pueden utilizar restricciones sobre diversas medidas de "significancia" e "interés". Las restricciones más conocidas son los umbrales mínimos de "soporte" y "confianza".

El 'soporte' de un conjunto de items X en una base de datos D se define como la proporción de transacciones en la base de datos que contiene dicho conjunto de items:

$$\text{sop}(X) = \frac{|X|}{|D|}$$

En el ejemplo anterior el conjunto {Leche,Pan} tiene un soporte de:

$$\text{sop}(X) = \frac{2}{5} = 0.4$$

Es decir, el soporte es del 40% (2 de cada 5 transacciones). La 'confianza' de una regla se define como:

$$\text{conf}(X \Rightarrow Y) = \frac{\text{sop}(X \cup Y)}{\text{sop}(X)} = \frac{|X \cup Y|}{|X|}$$

Por ejemplo, para la regla:

$$\{\text{Leche,Pan}\} \Rightarrow \{\text{Mantequilla}\}$$

La confianza sería:

$$\text{conf}(\{\text{Leche,Pan}\} \Rightarrow \{\text{Mantequilla}\}) = \frac{\text{sop}(\{\text{Leche,Pan}\} \cup \{\text{Mantequilla}\})}{\text{sop}(\{\text{Leche,Pan}\})} = \frac{0.2}{0.4} = 0.5$$

Este cálculo significa que el 50% de las reglas de la base de datos que contienen 'leche' y 'pan' en el antecedente también tienen 'mantequilla' en el consecuente;

Entonces, podemos definir la confianza como:

$$P(Y|X)$$

Las reglas de asociación deben satisfacer las especificaciones del usuario en cuanto a umbrales mínimos de soporte y confianza. Para conseguir esto el proceso de generación de reglas de asociación se realiza en dos pasos. Primero se aplica el soporte mínimo para encontrar los conjuntos de items más frecuentes en la base de datos. En segundo lugar se forman las reglas partiendo de estos conjuntos frecuentes de items y de la restricción de confianza mínima.



## EJEMPLO EN CÓDIGO

Transaction ID	<i>Milk</i>	<i>Eggs</i>	<i>Apples</i>	<i>Bread</i>
001	1	1	0	1
002	1	1	0	0
003	1	0	0	1
004	0	1	1	0

### Librerías

```
pip install pandas  
pip install mlxtend
```

# EJEMPLO EN CÓDIGO (APRIORI)

Paso 1: Crear una lista con los datos requeridos

```
dataset = [['Milk', 'Eggs', 'Bread'],  
['Milk', 'Eggs'],  
['Milk', 'Bread'],  
['Eggs', 'Apple']]
```

```
print(dataset)
```

Output:

```
[['Milk', 'Eggs', 'Bread'], ['Milk', 'Eggs'], ['Milk', 'Bread'], ['Eggs', 'Apple']]
```

# EJEMPLO EN CÓDIGO

Paso 2: Convertir la lista a un dataframe con valores booleanos

```
import pandas as pd
from mlxtend.preprocessing import TransactionEncoder

te = TransactionEncoder()
te_array = te.fit(dataset).transform(dataset)
df = pd.DataFrame(te_array, columns=te.columns_)
```

Output:

	Apple	Bread	Eggs	Milk
0	False	True	True	True
1	False	False	True	True
2	False	True	False	True
3	True	False	True	False

# EJEMPLO EN CÓDIGO

Paso 3: Encontrar la frecuencia de ocurrencias de cada item

```
from mlxtend.frequent_patterns import apriori

frequent_itemsets_ap = apriori(df, min_support=0.01, use_colnames=True)
```

Output:

	support	itemsets
0	0.25	(Apple)
1	0.50	(Bread)
2	0.75	(Eggs)
3	0.75	(Milk)
4	0.25	(Eggs, Apple)
5	0.25	(Eggs, Bread)
6	0.50	(Bread, Milk)
7	0.50	(Eggs, Milk)
8	0.25	(Eggs, Bread, Milk)

# EJEMPLO EN CÓDIGO

## Paso 4: Encontrar las reglas de asociación

## Interpretación

```
from mlxtend.frequent_patterns import association_rules

rules_ap = association_rules(frequent_itemsets_ap, metric="confidence", min_thresh
rules_fp = association_rules(frequent_itemsets_fp, metric="confidence", min_thresh
```

```
print(rules_ap)
```

Output:

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
0	(Apple)	(Eggs)	0.25	0.75	0.25	1.0	1.333333	0.0625	inf
1	(Bread)	(Milk)	0.50	0.75	0.50	1.0	1.333333	0.1250	inf
2	(Eggs, Bread)	(Milk)	0.25	0.75	0.25	1.0	1.333333	0.0625	inf