1. Soporte y Cobertura

El archivo **Globos.xlsx** contiene datos referidos a un experimento psicológico. De cada instancia se sabe el color del globo, el tamaño, si fue inflado por un adulto o un niño y si se estira o no. El objetivo es construir un modelo predictivo para determinar si un globo permanecerá inflado o no (atributo **Inflado?**).

a) Calcular el soporte (para reglas, equivalente a la tasa de acierto o accuracy) y la cobertura de cada una de las siguientes reglas, por separado, para **Globos.xlsx**:

R1: SeEstira? = Si y Edad=Adulto -> Inflado? = Si
R2: SeEstira? = Si y Edad=Niño -> Inflado? = No
R3: SeEstira? = No -> Inflado? = No

Regla	Soporte	Cobertura
R1		
R2		
R3		

b) Las tres reglas en conjunto ¿cubren todo el conjunto de entrenamiento? ¿Por qué es importante eso?

c) Calcular la tasa de acierto del conjunto de las 3 reglas para el conjunto de datos

a)

Regla	Soporte	Cobertura
R1	4/16	4/16
R2	2/16	4/16
R3	8/16	8/16

b) Si, lo cubren, esto es importante por las siguientes razones:
Para que el algoritmo que crea las reglas termine se necesita que todo el conjunto se encuentre cubierto, en caso de que no suceda esto no tendríamos un modelo como tal porque el algoritmo no se puede dar por terminado.
Si fuera del algoritmo el modelo no está cubierto, hay situaciones donde la entrada no se puede clasificar con las reglas existentes y debemos elegir otro algoritmo que se encargue de cubrir esto, como podría ser el ZeroR.

c) Tiene una tasa de 14/16, con error en la entrada de la fila 3 y 7.

2. Algoritmos de generación de reglas de clasificación: ZeroR, OneR, Prism

a) A partir de los datos del archivo **Globos.xlsx** determine, utilizando los métodos **ZeroR y OneR,** el conjunto de reglas de clasificación que permita predecir si un globo dado permanecerá inflado o no (atributo ***Inflado?***).

	ZeroR	OneR	PRISM
Reglas			

La resolución de este ejercicio debe realizarse de forma manual. En ambos casos deberá detallar los cálculos realizados para determinar las reglas indicadas.

En el caso de OneR, realizar una tabla que incluya la tasa de acierto para cada regla posible de cada atributo

Regla	Error

Además, hacer una tabla con el error total de cada atributo.

Atributo	Error

Se revisaron los datos del archivo Globos.xlsx y llegamos a lo siguiente:

En el caso de ZeroR se queda con los atributos que tengan “Inflado = No” por ser clase mayoritaria.

En el caso de OneR creamos la siguiente tabla revisando quienes no cumplian las reglas para calcular el error.

Atributo	Reglas	Error	Total
Color	Si (Color = Amarillo) entonces Inflado = Si	4/8	6/16
	Si (Color = Rojo) entonces Inflado = No	2/8	
Tamaño	Si (Tamaño = Chico) entonces Inflado = Si	3/6	6/16
	Si (Tamaño = Mediano) entonces Inflado = No	0/4	
	Si (Tamaño = Grande) entonces Inflado = Si	3/6	
Se_estira?	Si (Se_estira? = Si) entonces Inflado = Si	2/8	2/16
	Si (Se_estira? = No) entonces Inflado = No	0/8	
Edad	Si (Edad = Niño) entonces Inflado = No	2/8	6/16
	SI (Edad = Adulto) entonces Inflado = Si	4/8	

Atributo	Reglas	Acierto	Total
Color	Si (Color = Amarillo) entonces Inflado = Si	4/8	10/16
	Si (Color = Rojo) entonces Inflado = No	6/8	
Tamaño	Si (Tamaño = Chico) entonces Inflado = Si	3/6	10/16
	Si (Tamaño = Mediano) entonces Inflado = No	4/4	
	Si (Tamaño = Grande) entonces Inflado = Si	3/6	
Se_estira?	Si (Se_estira? = Si) entonces Inflado = Si	6/8	14/16
	Si (Se_estira? = No) entonces Inflado = No	8/8	
Edad	Si (Edad = Niño) entonces Inflado = No	6/8	10/16
	SI (Edad = Adulto) entonces Inflado = Si	4/8	

b) Repetir el punto a) utilizando PRISM.

La resolución de este ejercicio debe realizarse de forma manual. Detallar los cálculos realizados para determinar las reglas indicadas. Comenzar con la clase Inflado=Si. Para cada iteración de PRISM en una clase particular, armar una tabla indicando cada regla posible (incluyendo todos los atributos en la misma tabla) y el error asociado

Regla (A1=V1, A2=V2, ... → Clase)	Error

Como se nos indica trabajamos con “Inflado = Si”

Reglas	Error
Si (Color = Amarillo) entonces Inflado = Si	4/8
Si (Color = Rojo) entonces Inflado = Si	6/8
Si (Tamaño = Chico) entonces Inflado = Si	3/6
Si (Tamaño = Mediano) entonces Inflado = Si	4/4
Si (Tamaño = Grande) entonces Inflado = Si	3/6
Si (Se_estira? = Si) entonces Inflado = Si	2/8
Si (Se_estira? = No) entonces Inflado = Si	8/8
Si (Edad = Niño) entonces Inflado = Si	6/8
SI (Edad = Adulto) entonces Inflado = Si	4/8

Se elige la regla “(Se_estira? = Si) entonces Inflado = Si” y como la tasa de error todavía es 2/8 necesitamos agregar atributos al antecedente para llegar a una regla perfecta

Reglas	Error
Si (Color = Amarillo) entonces Inflado = Si	4/8
Si (Color = Rojo) entonces Inflado = Si	6/8
Si (Tamaño = Chico) entonces Inflado = Si	3/6
Si (Tamaño = Mediano) entonces Inflado = Si	4/4
Si (Tamaño = Grande) entonces Inflado = Si	3/6
Si (Se_estira? = Si) entonces Inflado = Si	2/8
Si (Se_estira? = No) entonces Inflado = Si	8/8
Si (Edad = Niño) entonces Inflado = Si	6/8
SI (Edad = Adulto) entonces Inflado = Si	4/8

Se elige la regla “(Se_estira? = Si y Color = Amarillo) entonces Inflado = Si” que es perfecta y reducimos el dataset.

Reglas	Error
Si (Se_estira? = Si y Color = Amarillo) entonces Inflado = Si	0/4
Si (Se_estira? = Si y Color = Rojo) entonces Inflado = Si	2/4
Si (Se_estira? = Si y Tamaño = Chico) entonces Inflado = Si	1/4
Si (Se_estira? = Si y Tamaño = Grande) entonces Inflado = Si	1/4
Si (Se_estira? = Si y Edad = Niño) entonces Inflado = Si	2/4
Si (Se_estira? = Si y Edad = Adulto) entonces Inflado = Si	0/4

Tenemos que encontrar la siguiente regla:

Reglas	Error
Si (Color = Amarillo) entonces Inflado = Si	4/4
Si (Color = Rojo) entonces Inflado = Si	6/8
Si (Tamaño = Chico) entonces Inflado = Si	3/4
Si (Tamaño = Mediano) entonces Inflado = Si	4/4
Si (Tamaño = Grande) entonces Inflado = Si	3/4
Si (Se_estira? = Si) entonces Inflado = Si	2/4
Si (Se_estira? = No) entonces Inflado = Si	8/8
Si (Edad = Niño) entonces Inflado = Si	6/6
SI (Edad = Adulto) entonces Inflado = Si	4/6

Se elige la regla “(Se_estira? = Si) entonces Inflado = Si” y como la tasa de error todavía es 2/4 necesitamos agregar atributos al antecedente para llegar a una regla perfecta

Reglas	Error
Si (Se_estira? = Si y Color = Rojo) entonces Inflado = Si	2/4
Si (Se_estira? = Si y Tamaño = Chico) entonces Inflado = Si	1/2
Si (Se_estira? = Si y Tamaño = Grande) entonces Inflado = Si	1/3
Si (Se_estira? = Si y Edad = Niño) entonces Inflado = Si	2/2
Si (Se_estira? = Si y Edad = Adulto) entonces Inflado = Si	0/2

Se elige la regla “(Se_estira? = Si y Edad = Adulto) entonces Inflado = Si” que es perfecta.

La clase queda cubierta con las reglas:

- “(Se_estira? = Si y Color = Amarillo) entonces Inflado = Si”
- “(Se_estira? = Si y Edad = Adulto) entonces Inflado = Si”

c) Si en la generación del modelo PRISM se comenzará por la otra clase (Inflado=No), ¿cómo quedaría el modelo?

Nos resta trabajar con “Inflado = No”

Reglas	Error
Si (Color = Amarillo) entonces Inflado = No	4/8
Si (Color = Rojo) entonces Inflado = No	2/8
Si (Tamaño = Chico) entonces Inflado = No	3/6
Si (Tamaño = Mediano) entonces Inflado = No	0/4
Si (Tamaño = Grande) entonces Inflado = No	3/6
Si (Se_estira? = Si) entonces Inflado = No	6/8
Si (Se_estira? = No) entonces Inflado = No	0/8
Si (Edad = Niño) entonces Inflado = No	2/8
SI (Edad = Adulto) entonces Inflado = No	4/8

Se elige la regla “(Se_estira? = No) entonces Inflado = No” que es perfecta y reducimos el dataset.

Reglas	Error
Si (Color = Amarillo) entonces Inflado = No	4/4
Si (Color = Rojo) entonces Inflado = No	2/4
Si (Tamaño = Chico) entonces Inflado = No	3/4
Si (Tamaño = Grande) entonces Inflado = No	3/4
Si (Se_estira? = Si) entonces Inflado = No	6/8
Si (Edad = Niño) entonces Inflado = No	2/4
SI (Edad = Adulto) entonces Inflado = No	4/4

Se elige la regla “(Edad = niño) entonces Inflado = No” y como la tasa de error todavía es 2/4 necesitamos agregar atributos al antecedente para llegar a una regla perfecta.

Reglas	Error
Si (Edad = Niño y Color = Rojo) entonces Inflado = No	0/2
Si (Edad = Niño y Color = Amarillo) entonces Inflado = No	2/2
Si (Edad = Niño y Tamaño = Chico) entonces Inflado = No	1/2
Si (Edad = Niño y Tamaño = Grande) entonces Inflado = No	1/2
Si (Edad = Niño y Se_estira? = Si) entonces Inflado = No	2/4

Se elige la regla “Si (Edad = Niño y Color = Rojo) entonces Inflado = No” que es perfecta.

La clase queda cubierta con las reglas:

- “(Se_estira? = No) entonces Inflado = No”
- “(Edad = Niño y Color = Rojo) entonces Inflado = No”

3. Métricas de evaluación de reglas

b) En base a los datos de la tabla **enfermedad.xslx**, calcule manualmente el soporte, interés, y confianza de las siguientes:

Regla	Soporte	Cobertura	Confianza	Interés
Enfermedad → Antecedentes				
Enfermedad → Adulto				
Adulto → Enfermedad				
Trabajo Pesado → Adulto				

Regla	Soporte	Cobertura	Confianza	Interés
Enfermedad -> Antecedentes	2/10	5/10	(2/10)/(5/10)= 0.4	(2/10)/(5/10)*(2/10)= 2
Enfermedad -> Adulto	4/10	5/10	(4/10)/(5/10)= 0.8	(4/10)/(5/10)*(7/10)= 1.14
Adulto -> Enfermedad	4/10	7/10	(4/10)/(7/10)= 0.57	(4/10)/(7/10)*(5/10)= 1.14
Trabajo Pesado -> Adulto	3/10	5/10	(3/10)/(5/10)= 0.6	(3/10)/(5/10)*(7/10)= 0.86

1. ¿Por qué es interesante la regla **Enfermedad → Antecedentes**, si bien su soporte y confianza son bajos?

2. ¿Porqué varía la confianza entre **Adulto → Enfermedad** y **Enfermedad→Adulto**? ¿Varía el soporte en esos casos?

3. ¿Que significa que **Trabajo Pesado → Adulto** tenga una cobertura y confianza relativamente altas, pero un interés bajo?

1) El interés es alto no por Enfermedad -> Antecedentes, sino por Antecedentes -> Enfermedad ya que siempre que tienen antecedentes tienen una enfermedad, su soporte y confianza son bajos porque no representan un gran porcentaje de casos en lo que esto ocurre.

2) Porque la confianza se divide solo por una de las partes de la implicación, es decir, el soporte del antecedente de esa implicación.
El soporte como tiene en cuenta las dos partes de la implicaciones no varía, lo que lo hace simétrico.

3) Hay muchos ejemplos donde el trabajo pesado se realiza por un adulto, por eso su cobertura y confianza son relativamente altas, pero hay también varios ejemplos donde alguien es adulto y no necesariamente realiza trabajo pesado, y viceversa, es decir, que la dependencia entre ellos es baja por eso no tiene mucho interés.

4. Evaluación de compras de supermercado con RapidMiner

La tabla **Supermercado.xlsx** contiene información de las compras realizadas por 10 clientes distintos.

a) Aplique **manualmente** el algoritmo APriori para generar los **itemsets** frecuentes con soporte mayor o igual a 0.3. Ordénelos en base a su soporte.

NOTA: para entregar este ejercicio, deberá incluir en la resolución (1) una tabla con el conjunto de reglas finales y su soporte, y (2) la serie de tablas L_i y C_i como se indica en la resolución de la teoría, ordenadas de forma acorde. Para simplificar, puede utilizar la primer letra de cada item, por ejemplo V para Vino y y L para Leche.

El soporte mínimo es de 0,3. Con un conjunto de datos de 10 transacciones, significa que cada ítem tiene que aparecer al menos 3 veces para cumplir el soporte.

Tablas de L y C

C1

Itemset	**#**
{**B**izcocho}	6
{**G**alletas}	6
{**J**ugo}	5
{**M**iel}	4
{**V**ino}	4
{**L**eche}	2

L1

Itemset
{B}
{G}
{J}
{M}
{V}

C2

Itemset	**#**
{V,J}	4
{M,B}	4
{G,J}	4
{M,G}	3
{B,G}	3
{B,J}	3
{V,G}	3
{M,J}	2
{V,B}	2
{V,M}	1

L2

Itemset
{V,J}
{M,B}
{G,J}
{M,G}
{B,G}
{B,J}
{V,G}

C3

Itemset	#
{V,G,J}	3
{M,B,G}	3
{B,G,J}	2

L3

Itemset
{V,G,J}
{M,B,G}

Conjunto de reglas finales

Itemset	**#**
M -> B	4/10
B -> M	4/10
G -> J	4/10
J -> G	4/10
V -> J	4/10
J -> V	4/10
M -> G	3/10
G -> M	3/10
B -> J	3/10
J -> B	3/10
B -> G	3/10
G -> B	3/10
V -> G	3/10
G -> V	3/10
M^B -> G	3/10
M^G -> B	3/10
B^G -> M	3/10
V^G -> J	3/10
V^J -> G	3/10
J^G -> V	3/10