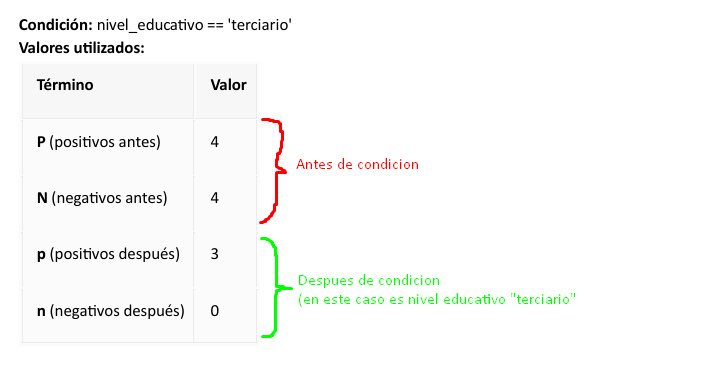
TP 3 ALGORITMO FOIL

**(Todo el Actividad 1 y parte de la Actividad 2 está en el código)**

EXPLICACION DEL CODIGO:

* \*Utiliza el dataframe dado y con un bucle for itero las veces que sale “persona” en la lista.
* \*“persona” es una nueva variable que almacena lo que hay en la lista “datos”
* \*Separa Positivos y Negativos con un for más compacto, luego cuenta estos



1.Fracción después de aplicar la condición:

(p/p+n) = (3/3+0) = 3/3 = 1000

2. Fracción antes de aplicar la condición:

(P/P+N) = (4/4+4) = 4/8 = 0,500

3. Logaritmos:

Log2(p/p+n) = log2 1,00 = 0,000

Log2(P/P+N) = log2 0,500 = 1,000

4. FOIL Gain:

p x ( log2 (p/p+n) – log2(P/P+N))

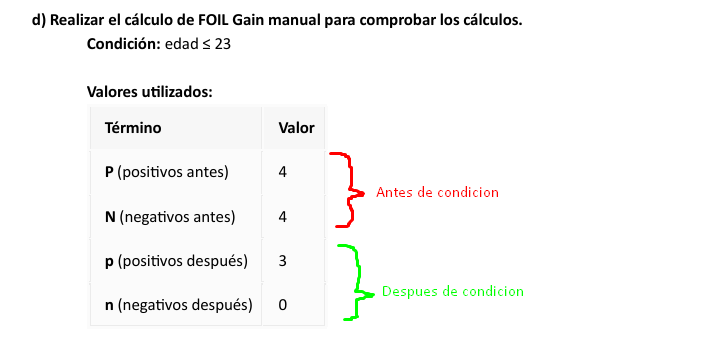
FG = 3 x (0,00 – (-1,000)) = 3 x 1 = 3 = FG

5. Interpretación

Esta condición tiene 3 positivos y excluye a los negativos. La ganancia es significativa y por eso es efectiva

**(el código es prácticamente el mismo resultado, lo único que cambia es la condición, asiqué no lo vi necesario hacer)**

Actividad 2:



1.Fracción después de aplicar la condición:

(p/p+n) = (3/3+0) = 3/3 = 1000

2. Fracción antes de aplicar la condición:

(P/P+N) = (4/4+4) = 4/8 = 0,500

3. Logaritmos:

Log2(p/p+n) = log2 1,00 = 0,00

Log2(P/P+N) = log2 0,500 = 1,00

4. FOIL Gain:

p x ( log2 (p/p+n) – log2(P/P+N)) = 3 x 0,00 +1,00 = 3 = FG

5. Interpretacion

Esta condicion tiene 3 positivos y excluye a los negativos. La ganancia es significativa y por eso es efectiva