**Explicaciones:**

Implementación de Itemsets Frecuentes

Definición del Problema:

* Creamos una lista de todas las transacciones, donde cada transacción es una lista de ítems.
* Usamos la biblioteca python-constraint para definir un problema de constraint programming.
* Definimos las variables del problema, que son los ítems, con posibles valores 0 (no incluido) o 1 (incluido).

Restricción de Soporte:

* Definimos una función de restricción que verifica si un conjunto de ítems aparece en un número suficiente de transacciones para cumplir con el umbral de soporte mínimo.

Solución del Problema:

* Usamos el método getSolutions de python-constraint para encontrar todos los conjuntos de ítems que cumplen con la restricción de soporte.
* Filtramos y guardamos los itemsets frecuentes.

Implementación de Reglas de Asociación

Definición del Problema:

* Utilizamos los itemsets frecuentes encontrados en el paso anterior.
* Para cada itemset, generamos posibles reglas de la forma A -> B usando combinaciones de ítems.

Cálculo de Confianza:

* Calculamos la confianza de cada regla como el soporte del itemset completo dividido por el soporte del lado izquierdo de la regla (lhs).
* Filtramos las reglas que cumplen con el umbral de confianza mínima.

Generación de Reglas:

* Las reglas que cumplen con el umbral de confianza mínima se guardan y se muestran como resultado.

**Conclusión**

La implementación de la búsqueda de itemsets frecuentes y reglas de asociación utilizando constraint programming permite explorar patrones interesantes en los datos de transacciones de mercado. Este enfoque puede ser adaptado y extendido para conjuntos de datos más grandes y complejos, ajustando los umbrales de soporte y confianza según sea necesario.