



### 1. Objetivo del laboratorio

Desarrollar de forma autónoma **dos Notebook** que permitan implementar algoritmos para obtener reglas de asociación y patrones secuenciales.

### 2. Elementos a utilizar:

- Lenguaje Python
- Librería numérica *NumPy*, *pandas*, *scikit-learn* y gráfica *Matplotlib*
- Entorno Anaconda
- Editor Jupyter

### 3. Práctica 1 (reglas de Asociación)

#### Objetivo (5 puntos)

Crear una “librería” que nos permita solucionar todos los problemas relacionados con las reglas de asociación. Para ello tendremos que implementar el algoritmo Apriori y una serie de métodos para obtener la distinta información que este genera.

- 1) Empezaremos generando los itemsets frecuentes para  $k=1$ . En este punto necesitaremos obtener el soporte de un itemset candidato. Por lo tanto, se tendrá que implementar un método que dado un itemset devuelva su soporte.
- 2) A partir de  $k=2$ .
  - a. Generar los itemset candidatos a partir de los datasets frecuentes del punto anterior.
  - b. Podar candidatos que contengan subconjuntos de longitud  $k$  que no sean frecuentes.
  - c. Calcular el soporte de cada candidato (usando el método del punto 1)
  - d. Elimina los candidatos que no sean frecuentes, dejando sólo los frecuentes.
- 3) Repetir el proceso 2 hasta que no se generen nuevos itemsets frecuentes.
- 4) Con los itemsets frecuentes. Generar todas las posibles reglas con la confianza de cada una de ellas.
- 5) Evaluar las reglas que son de alta confianza. Para ello crearemos un método que dado el antecesor y sucesor nos dé la confianza de la regla.
- 6) Listar todas las reglas que sean de alta confianza.
- 7) Crear los siguientes métodos. Dado un antecesor, devolver todas las reglas que contengan a dicho antecesor. Dado un umbral mínimo devolver todas las reglas que cumplan con dicha confianza.

Para probar la librería usaremos el Dataset `movies_dataset.csv` que se ha proporcionado y usaremos las listas de géneros asociadas a cada película. Prueba al menos tres configuraciones de soporte y frecuencia. Para una de ellas, interpreta algunas de las reglas obtenidas que te hayan resultado curiosas.

### 4. Práctica 2 (patrones secuenciales)

#### Objetivo (5 puntos)

La biblioteca de la Universidad está pensando en reordenarla de manera que los usuarios pierdan el menos tiempo posible buscando lo que necesitan. Basándose en el orden de consulta de varias fuentes, pretende que la información esté lo más próxima posible entre ella. Para ello disponemos del Dataset `library.csv` que tiene información del orden de consulta que ha realizado cada usuario. En cada registro tendremos el id del usuario, el momento en que empezó a consultar la información (en milisegundos) y las fuentes consultadas. Implementar el algoritmo GSP para obtener patrones secuenciales.

- 1) Partiendo del csv obtener las secuencias para cada usuario.
- 2) Generar las secuencias frecuentes para  $k=1$ . Para ello necesitaremos implementar un método que dada una secuencia nos diga su soporte.



- 3) Generar las posibles combinaciones (secuencias candidatas) en las que los elementos pertenecen a la misma transacción y aquellos que son de diferentes transacciones.
- 4) Calcular el soporte de dichas secuencias candidatas con respecto a las secuencias de nuestro csv.
- 5) Eliminar aquellas que no cumplan con el soporte mínimo.
- 6) Repetir hasta que no se generen más secuencias candidatas.

Una vez comprobado su correcto funcionamiento. Prueba al menos dos configuraciones de soporte. Para una de ellas, interpreta algunos de los patrones secuenciales que te hayan resultado curiosos.

### 5. Forma de entrega del laboratorio:

La entrega consistirá en un fichero comprimido RAR con nombre **LAB02-GRUPOxx.RAR** subido a la tarea **LAB1** que **contenga únicamente**

1. **Por cada práctica** un notebook de Jupyter (archivos con extensión **.ipynb**).
2. La **memoria del laboratorio** que se irá construyendo en el Notebook de manera que se explique todo lo que se hace.

**Las entregas que no se ajusten exactamente a esta norma NO SERÁN EVALUADAS.**

### 6. Rúbrica de la Práctica:

#### 1. IMPLEMENTACIÓN: Multiplica la nota del trabajo por 0/1

Siendo una práctica de Data Mining, todos los aspectos de programación se dan por supuesto. La implementación será:

- Original: Código fuente no copiado de internet. Grupos con igual código fuente serán suspendidos
- Correcta: El programa funciona y ejecuta correctamente todo lo planteado en los apartados de cada práctica.
- Comentada: Inclusión (**obligatoria**) de comentarios.
- En las gráficas que se realicen proporciona todos los datos que creas necesarios.

#### 2. MEMORIA DEL LABORATORIO

Obligatorio redacción clara y correcta ortográfica/gramaticalmente. Cada paso que se haga tiene que estar justificado.