

(mini)Práctica 3 de *Minería de Datos*:

CLUSTERING

Pablo Bermejo, M. Julia Flores y José A. Gámez

1. Objetivo de la práctica

El objetivo de la práctica es realizar algunas pruebas más con alguna de las bases de datos ya preprocesadas de vuestro trabajo anterior y experimentar sobre ellas la técnica de clustering. Dividiremos esta práctica (que no debería ocuparos más de las dos horas de la sesión y otras dos horas fuera de ella) en dos apartados.

2. Apartado 1: Minería de datos descriptiva

Se trata aquí de segmentar vuestra base de datos de forma que podamos ver si hay subconjuntos compactos en la misma. Para ello podemos usar la siguiente secuencia de pasos:

- Cargar vuestra base de datos en WEKA.
- Nos vamos a la pestaña de **Cluster** en Weka.
- En **Cluster mode** encontramos marcado **Use training set**. Ya sabemos que en general esto no es una buena idea, así que marcamos **Percentage split** que por defecto usa 66 % de los datos para entrenar y 34 % como test.
- Como algoritmos de clustering podeis/debeis probar:
 - **EM**. Método basado en densidades y que podeis ver como el que se estudió de imputación de valores perdidos sólo que ahora la variable cluster esta *perdida* en todos los registros. El algoritmo no necesita el número de cluster como parámetro de entrada (lo deduce) pero podemos forzar a darle un valor.
 - **SimpleKMeans**. Estudiado en clase. Necesita k como parámetro de entrada.
 - **FarthestFirst**. Está basado en distancias. Necesita el valor (k) para el número de clusters.

Importante: antes de lanzar ningún algoritmo debeis usar el botón de **Ignore Attributes** para marcar que la clase no se use durante el clustering!!!

- Una vez hecho el clustering podeis intentar se puede intentar interpretar si los grupos que han salido son buenos, malos o regulares. P.e. si usamos la base de datos CPU que viene en el directorio data de WEKA, con Kmeans (k=2) y eliminando la variable clase se obtiene:

Cluster centroids:

Attribute	Cluster#		
	Full Data (137)	0 (112)	1 (25)
=====			
MYCT	207.2701	243.3482	45.64
MMIN	3118.2482	1785.2143	9090.24
MMAX	12202.9635	7685.7679	32440
CACH	26.4818	11.9821	91.44
CHMIN	5.2263	3.0536	14.96
CHMAX	19.8978	12.5179	52.96

Clustered Instances

```
0      63 ( 88%)
1       9 ( 13%)
```

de donde es fácil interpretar que el cluster 1 es el de los ordenadores buenos y el cero el de los malos, si observamos los valores de las variables para ambos clusters. Además, al final vemos que muy poquitos del test se asignan al cluster 1.

- El problema para nosotros es que no conocemos el significado de las variables de nuestro problema. Por tanto, una forma de evaluar nuestros algoritmos de clustering puede ser marcar **Classes to clusters evaluation**, obligar a que el número de clusters sea igual al de clases (2 en nuestro caso) y ver como de bien los clusters aprendidos se asocian a las clases reales de nuestro problema.

3. Apartado 2: Clustering como paso previo a la clasificación

Una vez hemos obtenido una serie de agrupamientos (sin usar la variable clase y validando con Percentage Split), podemos hacer el siguiente proceso:

- En la pestaña de [Preprocess](#) en donde se visualizan los atributos gráficamente marcamos No-class
- Elegimos el filtro `unsupervised.attribute.AddCluster` y configuramos los parámetros del cluster deseado (no olvideis ignorar la clase!!).
- Aplicamos el filtro y obtenemos un atributo más con el número de cluster.
- Partimos la base de datos en tantas bases de datos como clusters hayamos obtenido. Esto lo podeis hacer fuera de Weka o en Weka aplicando el filtro `unsupervised.instances.RemoveWithValue` lo ejecutais para cada valor del cluster, y vais grabando los ficheros resultantes.
- Para cada fichero resultante (tantos como clusters) eliminais la variable cluster (puesto que es inútil al tener el mismo valor para todos los registros) y aplicais un proceso de clasificación (con validación cruzada de 5 folds).

- El resultado final será una suma ponderada de todos los procesos de clasificación realizados, donde los pesos serán el % de ejemplos en cada cluster respecto al total. P.e. si una base de datos tiene 100 registros y la hemos dividido en tres clusters con ($c_0=20, c_1=50, c_2=30$) registros, entonces el resultado p.e. para AUC sería:

$$AUC(total) = \frac{20}{100}AUC(c_0) + \frac{50}{100}AUC(c_1) + \frac{30}{100}AUC(c_2)$$

Aviso: crear muchos clusters puede llevar a acertar más, pero también a sobreajustar!!! Además os obligará a trabajar más puesto que gran parte del trabajo aquí descrito es manual.

4. ¿Qué y cuándo hay que entregar?

Un breve informe con los resultados que hayas obtenido así como los ficheros resultantes de fraccionar el conjunto de datos total en función de los clusters. Debes remarcar las conclusiones obtenidas acerca del clustering (apartado 1) y la mejora obtenida en el proceso de clasificación respecto a no usar el clustering como preprocesamiento.

La fecha de entrega (Moodle) es el día 23 de Diciembre.