EJERCICIO 1 -- Técnicas de Preprocesamiento

Los **"filtros" o técnicas de preprocesado** que os propongo aplicar sobre la base de datos comentada, son los siguientes:

1. Discretizar en 3 intervalos por igual anchura e igual frecuencia. Descrubre por ti mismo cómo acceder a ese "filtro" en WEKA. Entiende, documenta y explica sus parámetros "bins" y useEqualFrequency", así como el efecto del filtro en las variables numéricas. **Se trata que documentes, comentes y analices el efecto de la aplicación de este filtro (y los siguientes).**

   El filtro "Discretize" se encuentra dentro de las herramientas de preprocesamiento non-sobrevisadas. Es un filtro que agrupa los valores de cada variable según dos opciones: 1) equal bin width o 2) equal frequency distribution. El algoritmo equal bin widths agrupa los valores de una variable en bins de misma anchura w. Esta anchura es determinado por w = (max-min)/k, donde k corresponde al numero total de bins, y max y min corresponden al mínimo y el máximo de los valores dentro de cada variable. El numero de bins k puede ser especificado por el usuario a través del parámetro "bins", pero por lo que entiendo Weka puede buscar cual es el numero de bins optimo. Por otro lado, el algoritmo equal frequency agrupa los datos de tal manera que para cada variable resulta una distribución de los valores en cuela cada bin tiene un numero parecido de valores. No se pueden usar los dos filtros al mismo tiempo, hay que elegir entre la opción "equal bind width" o "equal frequency distribution". Entiendo que aunque se puede especificar el numero de bins conjunto con la opción equal frequency distribución, esta configuración del filtro no garantiza que los bins tienen la misma anchura. Cuando se aplica la este filtro con la opción equal frequency y un numero de 3 bins, el filtro agrupa todos los valores de las variables numéricas en 3 bins en cuales hay un numero de casos similares. Se puede observar que este filtro no afecta la distribución de las clases dentro de cada variable. Por esto, la distribución de los valores dentro de la variable clase no cambia cuando se aplica el filtro.

2. La técnica de discretización supervisada o "class-dependent" explicada en las transparencias. Entiende, documenta y explica el efecto del filtro en las variables numéricas.

   Esta técnica consiste en agrupar las valores de tal manera que la entropia de la variable clase esta minimizada en cada intervalo. Como cada valor tiene un label correspondiente, se trata de agrupar los valores numéricos en intervalos en cuales hay muchos casos que pertenecen a una clase, y pocos casos que pertenecen a la otra clase. Mas la distribución de las clases dentro de un interval es homogénea, mas bajo sera la información o entropia que tiene este bin con respecto a la variable clase. Aplicando este filtro a los datos se puede observar que el numero de intervalos no es constante a través de las variables, y que cada variable puede tener un numero de bins diferente. También se puede observar que en los intervalos la mayoría de casos pertenecen a una clase, y que hay muy pocos casos de la otra clase.

3. Localiza un dataset en formato *.arff de WEKA que contenga valores perdidos ("diabetes.arff" no los tiene): recuerda que éstos se representan en WEKA mediante el símbolo '?' (sin comillas). En el siguiente subdirectorio tienes un conjunto de datasets en formato WEKA: la elegida pueder partir de aquí u otra ubicación que consideres. Fíjate en el "filtro" "ReplaceMissingValues". ¿En qué consiste? ¿Qué valores imputa en los valores perdidos de variables numéricas? ¿Qué valores imputa en los valores perdidos de las variables nominales-discretizadas (e.g. color)? Entiende, documenta y explica el efecto del filtro.

El dataset que he utilizado es “titanic.arff”. El filtro “Replace Missing Values” consiste en remplazar valores perdidos por la mediana o el modo. Si la variable es numérica, los valores perdidos serán substituidos por la media de las valores en esa variable. En el caso de una variable nominal, los valores perdido serán substituidos por el modo de los valores en esa variable. Por ejemplo en la seria de valores numéricas 5.0,6.0,8.0,9.0,?.1.0 el valor perdido ? seria substituido por el valor (5.0+6.0+8.0+9.0+1.0)/5.0=5.8. En el caso de una variable nominal con valores naranja, rojo, naranja, verde, azul, azul, naranja, ?, negro el valor ? seria substituido por el valor “naranja”. En el dataset “titanic.arff” por ejemplo, los valores perdidos ocurren exclusivamente en la variable “cabina” cuela refleja donde se alojaran los pasajeros en el barco. Cuando se aplica el filtro “Replace.Missing Values” los valores perdidos dentro de la variable “cabina” son substituidos por el valor “C23,C25,C27” porque este valor es el valor mas frecuente dentro de esta variable.

4. Normalizar una variable: clave cuando se va a trabajar por ejemplo, con un clasificador como el del vecino más próximo y se deben calcular distancias Euclideas entre valores de una misma variable. Con el objetivo de que todas las variables tengan el mismo “peso”-”relevancia” en el cálculo de la distancia total entre dos casos. Entiende, documenta y explica el efecto del filtro en las variables numéricas.

   Este filtro va transformar las valores de una variable de tal manera que el mínimo de las valores dentro de una variable corresponde al valor 0, y el máximo de las valores corresponde al valor 1. La transformación aplicada a los datos es: (x-min)/(max-min), donde x corresponde al valor de un caso dentro una variable y min y max corresponden al mínimo y al máximo de los valores de esta variable. Aplicando el filtro a las variables numéricas se puede ver que el mínimo y máximo cambien en la pantalla de Weka. Por ejemplo en la variable edad del dataset “titanic.arff” antes de la normalización el mínimo y máximo son de 0.42 y 89 años, y después de normalizar estos valores son transformados a 0 y 1.

5. El filtro “unsupervised—Instance—Resample”. Entiende, documenta y explica el efecto del filtro.

   Este filtro permite seleccionar una sub-muestra que tiene propiedades similares a la muestra original. Por ejemplo se puede seleccionar una sub-muestra que corresponde a 10% del numero de casos manteniendo dentro de estos 10% las frecuencias de cada variable. Por ejemplo, en el dataset “titanic.arff”, el ratio de hombres y mujeres fue de 577/314=1.83 y la edad media de 29.7 años (desviación estándar: 13). Después de aplicar el filtro “Resample” se puede observar un ratio de hombres y mujeres parecido de 59/30=1.96 y una edad media de 30 años (desviación estándar: 13). Los parametros importantes de este filtro en Weka son el “randomSeed” o la semilla del generador de números aleatorios, “no Replacement” determina si un caso puede ser parte de otra sub-muestra o no, y “sampleSizePercent” determina el tamaño de la sub-muestra en porcentaje de la muestra total.

6. El filtro “unsupervised--Attribute--AddExpression”. Entiende, documenta y explica el efecto del filtro.

   Este filtro permite aplicar transformaciones o ecuaciones a las variables. Por ejemplo se pueden transformar las variables aplicándoles una transformación logarítmica. Aplicando este filtro se crea una nueva variable que tiene el nombre asignado por el usuario. Por ejemplo volviendo al dataset “Diabetes” se pueden cuadrar los valores de la edad de los sujetos. Al aplicar el filtro se crea una nueva variable con los valors cuadrados de la edad. Se pueden aplicar muchas transformaciones lineales y non-lineales como la adición, la substracción, la multiplicación, el logaritmo, etc

7. El filtro “unsupervised-Attribute-InterQuartileRange”. WEKA exige que se aplique sobre un dataset donde todas las predictoras sean numéricas: tenlo en cuenta. Explica qué función realiza y expón su efecto sobre los datos. Entiende sus parámetros clave. Entiende, documenta y explica el efecto del filtro.

   La función de este filtro es de detectar valores extremes y los “outliers”. Dentro de este filtro se definen unos intervalos basados en el IQR, y los valores que quedan debajo o encima de unos sellos predefinidos son categorizados como valor extremo o outlier. Aplicando el filtro genera dos nuevas variables en la pantalla de Weka: “Outlier” y “ExtremeValues”, en cuales se puede visualizar el numero de valores extremos y outliers. De manera general y añadiendo o substrayendo unos márgenes definidos por los parámetros EVF, OF y IQR, se puede decir que los outliers son valores que quedan o encima del tercero cuartillo mas o debajo del primer cuartillo, y los valores extremos son valores aun mas pequeños o mas largos que los outliers.

8. El “MultiFilter” que cuelga como un “filtro” principal. Entiende, documenta y explica el efecto del filtro.

   El “MultiFilter” permite de aplicar varios filtros sucessivamente. Por ejemplo, se pueden añadir varios filtros y luego el programa ejecuta cada filtro de manera sucesiva. Se podrían por ejemplo añadir los filtros “Discretize” y “Standardize” y “InterQuartileRange” para 1) discretear, estandardizar los datos y luego detectar outliers.y valores extremos. En resumen, la funcionalidad de este filtro es de implementar una “pipeline” de pre-per-procesamiento de datos para la aplicación de algoritmos de aprendizaje automático Una vez añadido, se pueden configurar los parámetros de cada filtro haciendo un doble click en cada filtro.

9. Todos los filtros vistos hasta ahora los podemos calificar de "generalistas" para matrices clásicas con variables numéricas y nominales. Son filtros que en un principio tienen posibilidad de aplicarse sobre cualquier dominio de datos. Dependiendo del dominio en cuestión y el tipo de dato, existen filtros específicos: imágenes, voz, texto... WEKA dispone de filtros de preproceso de "propósito general". Aún así, tiene un interesante filtro para el caso de texto. Os lanzo este ejercicio más como curiosidad que otra cosa. Para el tratamiento y análisis de textos ("text mining", "natural language processing"), el filtro "unsupervised--Attribute-StringToWordVector". Para practicar con él, carga la base de datos de este enlace, "Telecom_Tweets.arff". Una explicación sobre ella la doy en el siguiente fichero: explico varios datasets de un mismo subdirectorio. Localiza la explicación referida al nuestro. Al cargarlo en WEKA descubres que únicamente dos variables caracterizan los casos: un string (cadena de caracteres) con el comentario, y la etiqueta con el tipo de sentimiento-opinión. WEKA no puede trabajar con la variable de tipo string tal y como se encuentra. Se trata de transformar la cadena de caracteres en unigramas que reflejan la aparición (o ausencia) de las distintas palabras. Consulta la ayuda del filtro, intenta entender sus parámetros (no los entenderás todos, es lo de menos, pero al menos los que consideres esenciales), y date cuenta de su efecto al aplicarlo, qué tipos de variables se han creado, etc. Es un filtro complejo y muy completo. Entiende, documenta y explica el efecto del filtro.

   Entiendo que este filtro genera una variable binaria para cada palabra encontrada en los 130 tweets. El valor 0 indica la ausencia de cada palabra, y el valor 1 indica la presencia de una palabra. Los parámetros que me han llamado la atención son “minTermFreq”, “normalizeDocLength”,”outPutWordCounts”. Entiendo que

“minTermFreq” corresponde a la minima frecuencia con cuela
una palabra debe ocurrir para ser considerada como relevante,
“normalizeDocLength” es una opción que permite de normalizar las frecuencias de cada palabra relativo al numero de palabras en el dictionario (los tweets), y por fin entiendo que ”outPutWordCounts” permite substituir los valores binarios por frecuencias. .

10. Descubre algún “filtro” por tu cuenta que te llame tu atención y no hayamos visto. Entiende y aplícalo. Entiende, documenta y explica el efecto del filtro.
Un filtro que me ha llamado la atención es el filtro “numericalCleaning”. Este filtro permite de substituir todo los valores que están debajo en encima de un sello con unos valores predefinidos. Con este filtro se pueden “limpiar” los datos de los valores extremos que pueden afectar la estimación de parámetros por un modelo. Por ejemplo, con este filtro todos los valores < -1.97 pueden ser substituidos por el valor 1.97. El efecto principal es reducir la variabilidad de los valores dentro de lass variable y de homogeneizar los valores eliminando valores fuera de rangos predefinidos.

11.

EJERCICIO 2 -- Técnicas de Preprocesamiento

Date cuenta del tamaño de la base de datos y de los datos que recoge. A partir de este fichero imaginemos que queremos hacer lo siguiente: seleccionar los casos de los partidos jugados por un equipo concreto (lo haremos por ejemplo para el equipo "LAGUN_ARO_GBC", o para otro equipo que prefieras, es lo de menos): son datos de la temporada 2012-2013.

La base de datos tiene 612 casos (lineas) y 29 variables.

Busca dentro de los filtros que están en "unsupervised"--"instance"; es uno de estos el que debes aplicar para conseguir lo pedido: "RemoveWithValues". Parámetros clave con los que debes "jugar" (no siempre con todos a la vez): "attributeIndex", "invertSelection", "nominalIndices", "splitPoint".

Para poder seleccionar el equipo LAGUN_ARO_GBC hay que entrar el “28” en el campo attributeIndex (es el indice de la variable equipo), y el 123 en el campo nominalIndices (es el indice que corresponde al equipo LAGUN_ARO_GBC). También se debe activar el campo invertSelection cambiándole de false a true. De esa manera el filtro quita los otros equipos y solo guarda las estadísticas para el LAGUN_ARO_GBC.

El parámetro "splitPoint" lo debéis utilizar en caso de que prefiráis trabajar con este otro fichero que recoge todos los partidos jugados entre las temporadas 1990-1991 hasta 2012-2013. En caso de que queráis utilizar este fichero: primero filtrar por la temporada y escoger la 2012-2013, posteriormente escoger los partidos del equipo.

¿Con cuántos casos te has quedado finalmente (deben ser 34...)? Chequea que ni hay partidos de otras temporadas (fíjate en los valores que toma la variable "codigo_temporada") ni de otros equipos (variable “equipo").

¿Has llegado a buen fin? Comenta y documenta tus incidencias durante el ejercicio.

Me ha costado mucho entender como funciona el filtro “RemoveWithValues”, porque no entendía la función del campo “atrributeIndex”. Pero una vez que he entendido este campo, el resto del ejercicio ha funcionado bien. Me he quedado con 34 casos del equipo LAGUN_ARO_GBC. En el caso del secundo archivo, se puede usar el valor 56 como “split-point” para seleccionar solo la temporada 2012-2013. Es importante desactivar el campo “invertSelection“. Eso guarda solo los partidos de la temporada con el código 57 (2012-2013), y luego se puede aplicar el filtro con los valores attributeIndex=28, invertSlection=true y nominalIndex=123. Al final, solo quedan 34 casos.