UNIVERSIDAD DE CÓRDOBA

Escuela Politécnica Superior

INTRODUCCIÓN AL APRENDIZAJE AUTOMÁTICO

Departamento de Informática y Análisis Numérico

Documentación de Prácticas

Juan Jesús Carmona Tejero

Gregorio Corpas Prieto

Índice general

Índice de Figuras

Índice de Tablas

Capítulo 1

Introducción a WEKA

1.1. Filtros no supervisados

1.1.1. Normalize

Realiza una normalización de todos los valores numéricos en el conjunto de datos.

Uso

Los valores son modificados al rango [0,1], tomando el valor 0 el dato más pequeño del conjunto y tomando el valor 1 el dato mayor del mismo, quedando el resto de valores en valores continuos dentro del rango.

Ejemplo

Como se observa en la figura 1.1, se tiene un conjunto de atributos originales a los cuales se les aplicará el filtro. Una vez filtrados, se puede observar en la figura 1.2 cómo estos han quedado en un rango 0-1.

```
@data
13.71,5.65,2.45,20.5,95,1.68,0.61,0.43,1.3,4,0.6,1.68,746.423729,y2
12.96,3.45,2.35,18.5,106,1.39,0.7,0.6,0.96,5.58,0.87,2.11,570,y2
12.77,2.39,2.28,19.5,86,1.39,0.51,0.26,1.56,7.1,0.61,1.33,425,y2
12.85,3.27,2.58,22,106,1.65,0.6,0.43,1.41,7.3,0.7,1.56,750,y2
```

Figura 1.1: Base de datos 'wine'

```
@data
0.705263,0.970356,0.582888,0.510309,0.271739,0.241379,0.056962,0.566038,0.280757,0.232082,0.097561,0.150183,0.334111,y2
0.507895,0.535573,0.529412,0.407216,0.391304,0.141379,0.075949,0.886792,0.173502,0.366894,0.317073,0.307692,0.208274,y2
0.457895,0.326087,0.491979,0.458763,0.173913,0.141379,0.035865,0.245283,0.362776,0.496587,0.105691,0.021978,0.10485,y2
0.478947,0.5,0.652406,0.587629,0.391304,0.231034,0.054852,0.566038,0.315457,0.513652,0.178862,0.106227,0.336662,y2
0.463158,0.381423,0.59893,0.587629,0.456522,0.172414,0.21519,0.660377,0.072555,0.735495,0.073171,0.131868,0.136947,y2
```

Figura 1.2: Base de datos 'wine' tras aplicar Normalize

1.1.2. ReplaceMissingValues

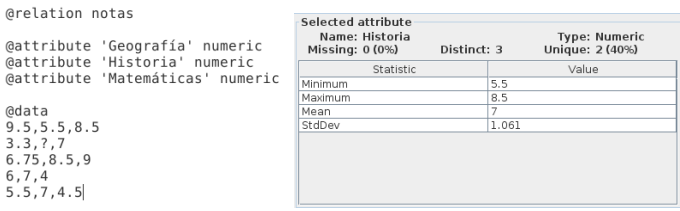
Reemplaza todo valor perdido, los cuales son representados con el signo '?', de los atributos nominales y numéricos.

Uso

Busca aquellos valores perdidos y los reemplaza con los valores de las modas y medias para dicho atributo.

Para ilustrar el uso de este filtro se ha usado una pequeña base de datos que contiene notas de 3 asignaturas para varias instancias que son los alumnos, y se observa como los datos perdidos son reemplazados.

Ejemplo



```
@relation notas

@attribute 'Geografía' numeric
@attribute 'Historia' numeric
@attribute 'Matemáticas' numeric

@data
9.5,5.5,8.5
3.3,?,7
6.75,8.5,9
6,7,4
5.5,7,4.5
```

(a) Base de datos 'notas' con valores perdidos

Selected attribute

Name: Historia | Missing: 0 (0%) | Distinct: 3 | Type: Numeric | Unique: 2 (40%)

Statistic	Value
Minimum	5.5
Maximum	8.5
Mean	7
StdDev	1.061

(b) Media tras aplicar el filtro

Figura 1.3: Base de datos y media calculada tras aplicar ReplaceMissingValues

```
@relation notas-weka.filters.unsupervised.attribute.ReplaceMissingValues

@attribute Geografía numeric
@attribute Historia numeric
@attribute Matemáticas numeric

@data
9.5,5.5,8.5
3.3,7,7
6.75,8.5,9
6,7,4
5.5,7,4.5
```

Figura 1.4: Fichero final con el valor perdido reemplazado

1.1.3. RemoveUseless

Elimina atributos nominales donde la varianza es muy grande o muy pequeña y que por tanto, no tienen utilidad.

Uso

Realiza el análisis y transformación de los componentes principales de los datos. Se usa junto con una búsqueda de Ranker. La reducción de la dimensionalidad se logra eligiendo suficientes vectores propios para tener en cuenta algún porcentaje de la varianza en los datos originales, por defecto 0.95.

Ejemplo

Parar ilustrar este filtro se ha usado una pequeña base de datos que contiene un atributo nominal que representa un color, y 2 atributos numéricos que representan otros datos asociados. Se puede observar como el atributo nominal tiene un valor distinto para cada instancia, y por tanto, la varianza es la máxima, así pues el filtro actúa descartando dicho atributo nominal.

```
@attribute color {blanco,rojo,azul}
@attribute 'x1' numeric
@attribute 'x2' numeric

@data
blanco, 20, 30
rojo, 25, 31
azul, 21, 29
```

(a) Base de datos 'colores' con atributo nominal

```
@attribute x1 numeric
@attribute x2 numeric

@data
20,30
25,31
21,29
```

(b) Fichero tras aplicar filtro

Figura 1.5: Antes y después de aplicar RemoveUseless

1.1.4. PrincipalComponents

Realiza el análisis y transformación de las componentes principales de los datos.

Uso

La reducción de la dimensionalidad se logra eligiendo suficientes vectores propios para tener en cuenta algún porcentaje de la varianza en los datos originales (por defecto 0.95). El ruido de los atributos puede filtrarse transformándolo en el espacio de la Componente Principal, eliminando algunos de los vectores propios peores, y luego transformando de nuevo al espacio original.

Ejemplo

```
@attribute color {blanco,rojo,azul}
@attribute 'x1' numeric
@attribute 'x2' numeric

@data
blanco, 20, 30
rojo, 25, 31
azul, 21, 29
```

Figura 1.6: Base de datos 'colores'

```
@attribute -0.632x1-0.621color=rojo+0.414color=blanco+0.207color=azul numeric
@attribute '0.772color=azul-0.617color=blanco-0.154color=rojo+0    x1' numeric
@attribute x2 numeric

@data
1.195229,-1.069045,30
-1.792843,-0.267261,31
0.597614,1.336306,29
```

Figura 1.7: Fichero final tras aplicar PrincipalComponents

1.1.5. RandomProjection

Reduce la dimensionalidad de los datos proyectándolos en un subespacio de menor dimensión usando una matriz aleatoria con columnas de longitud unitaria.

Uso

Primero aplica el filtro NominalToBinary para convertir todos los atributos a numérico antes de reducir la dimensión. Conserva el atributo de marca de clase.

Ejemplo

Como se muestra en la Figura 1.8 partimos de una base de datos con un atributo nominal de 3 posibles opciones 'Blanco', 'Rojo' y 'Azul' junto con un atributo numérico x1. Una vez aplicado el filtro, el primer atributo queda establecido en una matriz numérica de un subespacio menor.

```
@attribute color {blanco,rojo,azul}
@attribute 'x1' numeric

@data
blanco, 20
rojo, 25
azul, 21
```

Figura 1.8: Base de datos 'colores' con un atributo numérico

```
@data
0,0,0,0,0,0,0,0,0,0,20
0,1.732051,1.732051,0,0,1.732051,1.732051,0,-1.732051,0,25
0,1.732051,0,0,0,0,1.732051,0,-1.732051,0,21
```

Figura 1.9: Fichero final tras aplicar RandomProjection

1.1.6. NominalToBinary

Convierte todos los atributos nominales en atributos binarios numéricos.

Uso

Un atributo con k posibles valores se transforma en k atributos binarios (0-1) si la clase es nominal. Los atributos binarios se dejan binarios. Si la clase es numérica, es posible que desee utilizar la versión supervisada de este filtro.

Ejemplo

1.1.7. RemoveMissclassified

Elimina aquellas instancias que han sido incorrectamente clasificadas, de modo que no existan valores atípicos.

- **Uso**

  Permite escoger la marca de clase en la que se basan las clasificaciones erróneas, el clasificador sobre el que se basarán las clasificaciones erróneas, si el resultado será descartartado o aceptado, número de iteraciones, pliegues y umbral de error permisible.

- **Ejemplo**

1.1.8. RemovePercentage

Permite eliminar un porcentaje de la información de la base de datos.

- **Uso**

- **Ejemplo**

1.1.9. Resample

Produce una submuestra aleatoria de un conjunto de datos utilizando el muestreo con reemplazo o sin reemplazo.

- **Uso**

  Se puede especificar el número de instancias en el conjunto de datos generado. Cuando se utilizan en modo por lotes, los lotes posteriores no son remuestrados.

- **Ejemplo**

1.2. Filtros supervisados

1.2.1. AttributeSelection

- **Uso**

- **Ejemplo**

1.2.2. Discretize

- **Uso**

- **Ejemplo**

1.2.3. NominalToBinary

- **Uso**

- **Ejemplo**

1.2.4. SpreadToBinary

- **Uso**

- **Ejemplo**

1.2.5. ClassBalancer

- **Uso**

- **Ejemplo**

1.2.6. Resampler

- **Uso**

- **Ejemplo**

1.3. Base de datos Wine

1.3.1. Detalles de la base de datos

1.3.2. Modificación del fichero con nombres de atributo descriptivos

1.3.3. Descripción de atributos y clases

1.3.4. Tratamiento de elementos perdidos

1.3.5. Diferencia entre Distinct y Unique

1.3.6. Eliminar atributos identificadores

1.3.7. Relaciones visualmente significativas en entorno Visualice

1.4. Aplicación de 6 filtros a la base de datos

1.4.1. Filtro de selección de características

- **Uso**
- Resultados:
- **Ejemplo**

1.4.2. Filtro de selección de patrones

- **Uso**
- Resultados:
- **Ejemplo**

1.4.3. Filtro de filters/supervised/attribute/*

- **Uso**
- Resultados:
- **Ejemplo**

1.4.4. Filtro de filters/supervised/instance/*

- **Uso**
- Resultados:
- **Ejemplo**

1.4.5. Filtro de filters/unsupervised/attribute/*

- **Uso**

- Resultados:
- **Ejemplo**

1.4.6. Filtro de filters/unsupervised/instance/*

- **Uso**
- Resultados:
- **Ejemplo**

1.5. Conversión de todo atributo nominal a codificación binaria

1.6. División de la base de datos en particiones

1.6.1. División en 10-Holdout 75-25

- **Uso**
- Resultados:
- **Ejemplo**

1.6.2. División en 10-Fold

- **Uso**
- Resultados:
- **Ejemplo**

Capítulo 2

Clasificación y Regresión en WEKA

2.1. Algoritmo IB1 con 10-fold crossvalidation

2.1.1. Visualización de la clasificación

2.1.2. Interpretación de los resultados

2.2. Algoritmo IBK (k=1, k=3, k=5) con 10-fold crossvalidation a 1

2.2.1. Cálculo de media y desviación típica de las medidas

- Accuracy:
- Kappa:
- RMSE:
- F-Measure:
- Media ponderada AUC:

2.2.2. Visualización de la clasificación

2.2.3. Interpretación de los resultados

2.3. Base de datos house.arff

2.3.1. Carga de la base de datos

2.3.2. Variable Granite

- Influencia en el modelo
- Conclusiones

2.3.3. Variable binaria Bathroom

- Influencia en el modelo
- Aplicación de algoritmo de regresión lineal
- Conclusiones

2.3.4. Variable Bedrooms

- Influencia en el modelo
- Conclusiones

2.3.5. Variable HouseSize

- Influencia en el modelo
- Aportación al modelo de regresión lineal
- Conclusiones

2.4. Base de datos autoMpg.arff

2.4.1. Aplicación del algoritmo LinearRegression con un 80-20

2.4.2. Resultados obtenidos

- Conclusiones
- Tablas comparativas

2.4.3. Atributo que aporta más información de la variable dependiente

2.4.4. Atributo que no aporta información al modelo

2.4.5. Errores cometidos

- Visualización
- Representación de las diferentes cruces

2.4.6. Modificación del parametro attributteSelectionMethod

- Conclusión en el nuevo modelo
- Análisis de menor peso de "weight" frente "acceleration"

2.5. Método SimpleLinearRegression en base de datos autoMpg.arff

- Respuesta
- Solución
- Visualización
- Comentario de resultados

2.6. Algoritmos Logistic y SimpleLogistic

2.6.1. Análisis

- Conclusión
- Métricas
- Variables más influyentes (beta)
- Variables no usadas
- Visualización
- Asociación de fórmulas con modelos obtenidos según el algoritmo

2.6.2. Visualización gráfica de errores cometidos

2.6.3. Análisis de parámetros

- maxBoostingIterations
  - Análisis
  - Modificación
- heuristicStop
  - Análisis
  - Modificación