Minería de Datos

Preprocesamiento: Normalizacion - Discretización

Dr. Edgar Acuña
Departamento de Ciencias Matemáticas
Universidad de Puerto Rico-Mayaguez

E-mail: edgar.acuna@upr.edu, eacunaf@gmail.com

Website: academic.uprm.edu/eacuna

Preprocesamiento-Normalización

- La normalización de datos consiste en re-escalar los valores de los datos dentro de un rango especificado, tal como -1 a 1 o 0 a 1.
- También es conocido como “normalizacion del rango”.

Razones para normalizar

- Normalizar los datos de entrada ayudará a acelerar la fase de aprendizaje.
- Los atributos con rangos grandes de valores tendran mas peso que los atributos con rangos de valores mas pequenos, y entonces dominarán la medida de distancia. Por ejemplo, el clasificador K-nn sando la medida de distancia euclideana depende de que todas las dimensiones de los valores de entrada estén en la misma escala.
- También puede ser necesario aplicar algún tipo de normalización de datos para evitar problemas numéricos tales como pérdida de precisión y desbordamientos aritméticos (overflows).

Normalización Z-score

Los valores V son normalizados en base a la media y desviación estándar.

V' = (V - mean) / std

Este método trabaja bien en los casos en que no se conoce el máximo y mínimo de los datos de entrada pero no cuando existen outliers que tienen un gran efecto en el rango de los datos. Usando la libreria dprep de R se tiene:

zbupa=rangenorm(bupa,'znorm',superv=T)

Normalización Min-Max

Este método realiza una transformación lineal de los datos originales V en el intervalo especificado [newmin,newmax]

V' = (V - min) * (newmax - newmin) / (max - min) + newmin

La ventaja de este método es que preserva exactamente todas las relaciones entre los datos. No introduce ningún potencial sesgo en los datos. La desventaja es que se encontrará un error “fuera del límite” ("out of bounds“) si un futuro ingreso de datos cae fuera del rango original.

Usando dprep se tiene:

mmbupa=rangenorm(bupa,mmnorm,superv=TRUE)

Discretización

Proceso que transforma datos cuantitativos en datos cualitativos.

- Algunos algoritmos de clasificación aceptan solo atributos categóricos (Naïve Bayes, Bayesian Networks, Rough Sets).
- El proceso de aprendizaje frecuentemente es menos eficiente y menos efectivo cuando los datos tienen solamente variables cuantitativas.

Ejemplo de discretizacion aplicado a una parte de Bupa

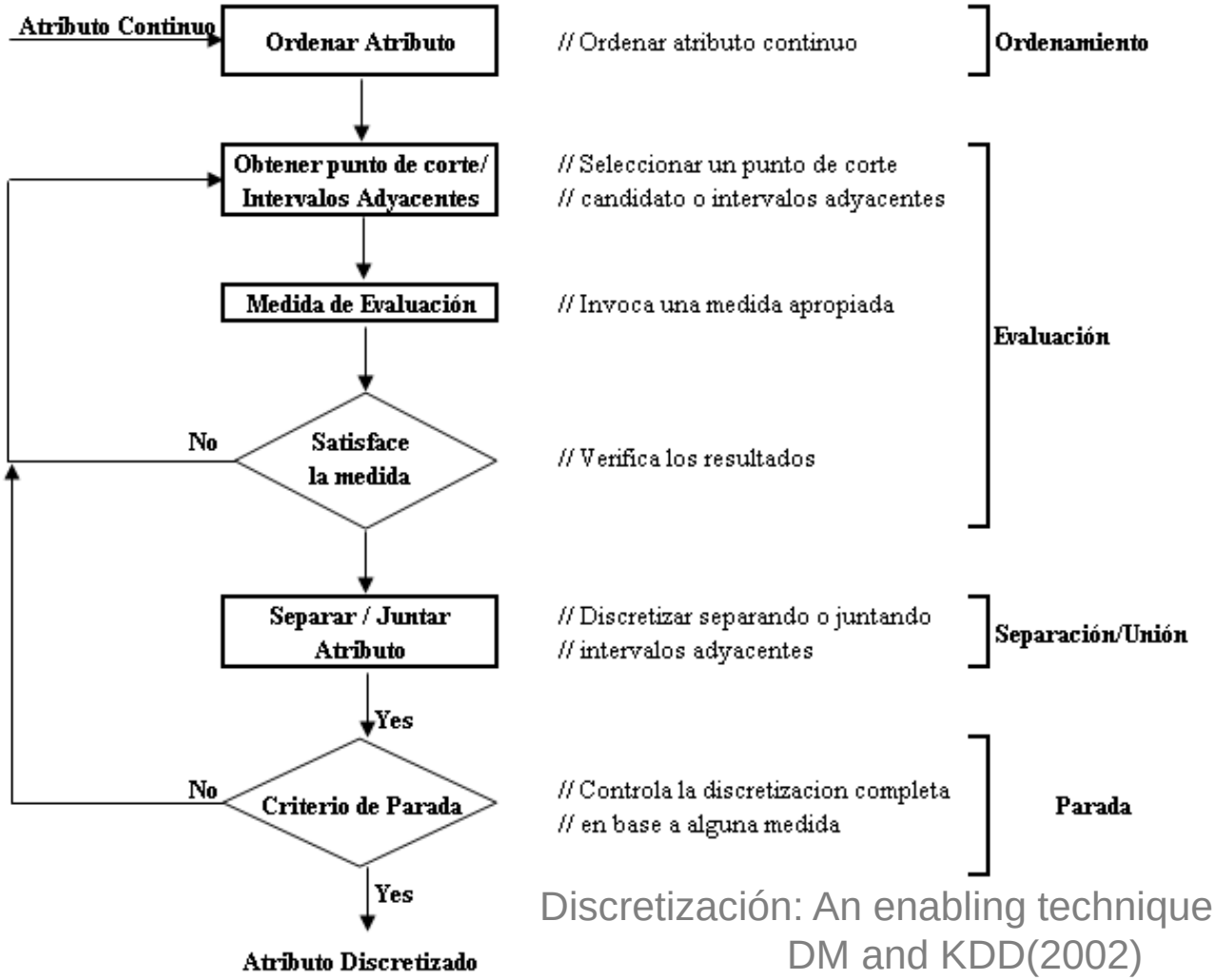
```
> m
   V1  V2  V3  V4 V5
45 5.1 3.8 1.9 0.4  1
46 4.8 3.0 1.4 0.3  1
47 5.1 3.8 1.6 0.2  1
48 4.6 3.2 1.4 0.2  1
49 5.3 3.7 1.5 0.2  1
50 5.0 3.3 1.4 0.2  1
51 7.0 3.2 4.7 1.4  2
52 6.4 3.2 4.5 1.5  2
53 6.9 3.1 4.9 1.5  2
54 5.5 2.3 4.0 1.3  2
55 6.5 2.8 4.6 1.5  2
```

```
> disc.ew(m,1:4)
   V1 V2 V3 V4 V5
45  1  3  1  1  1
46  1  2  1  1  1
47  1  3  1  1  1
48  1  2  1  1  1
49  1  3  1  1  1
50  1  2  1  1  1
51  2  2  2  2  2
52  2  2  2  2  2
53  2  2  2  2  2
54  1  1  2  2  2
55  2  2  2  2  2
```

Top-down (Separar) versus Bottom-up (Juntar)

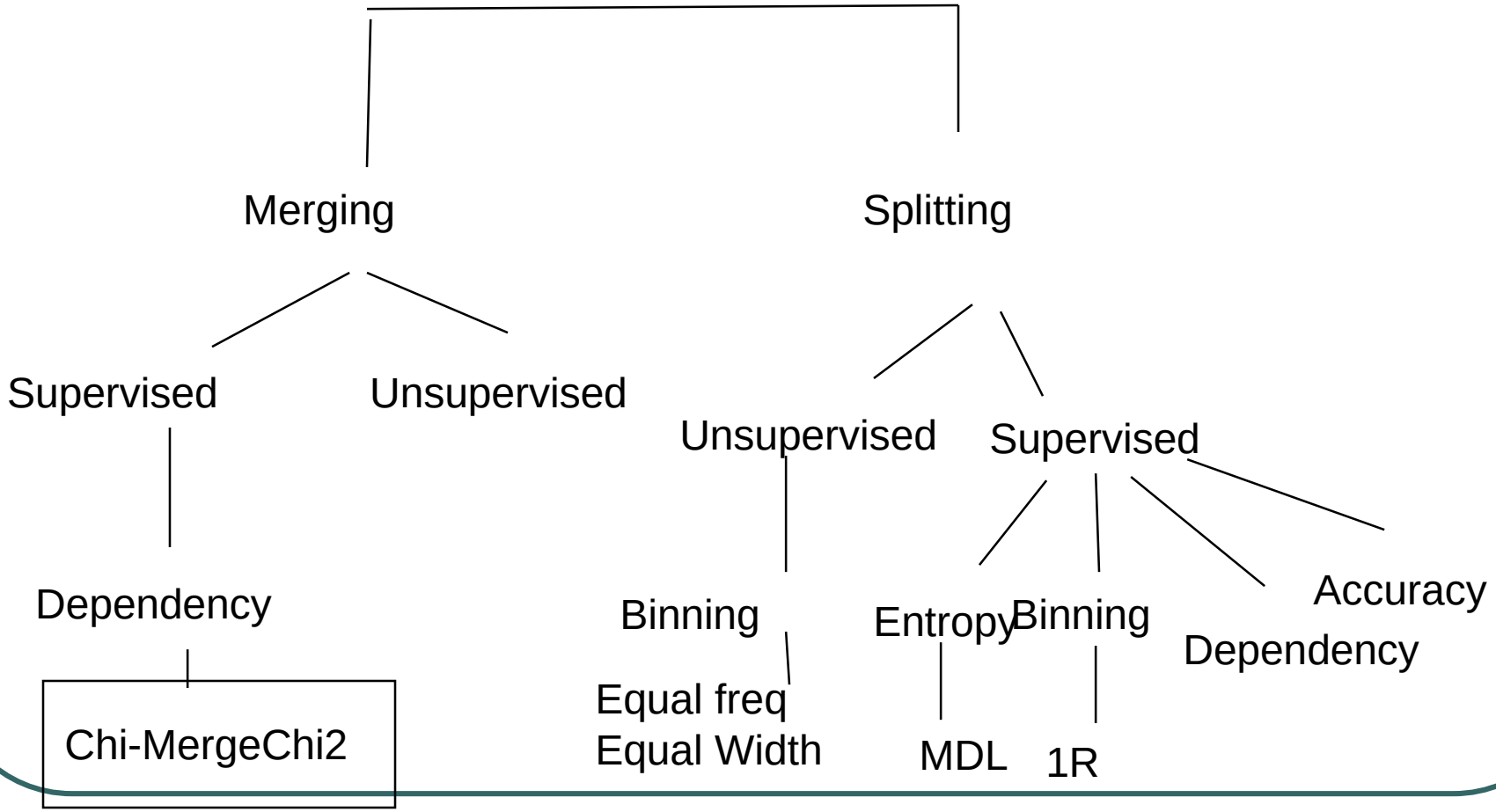
- Los métodos Top-down inician con una lista vacía de puntos de corte (o split-points) y continúan agregando nuevos puntos a la lista ”separando” los intervalos mientras la discretización progresa.
- Los métodos Bottom-up inician con la lista completa de todos los valores continuos de la variable como puntos de corte y eliminan algunos de ellos “juntando” los intervalos mientras la discretización progresa.

Discretización



Discretización: An enabling technique. Liu et al. DM and KDD(2002)

Clasificacion de los metodos de discretization



Evaluación de los métodos de discretización

- El número total de intervalos generados. Un número pequeño de intervalos generados es bueno hasta cierto punto.
- El número de inconsistencias en el conjunto de datos discretizado. La inconsistencia debe disminuir.
- La precision de prediccion. El proceso de discretización no debe tener un gran efecto en el tasa de error de mala clasificación.

Intervalos de igual amplitud (binning)

- Dividir el rango de cada variable en k intervalos de igual tamaño.
- si *A* es el menor valor y *B* el mayor valor del atributo, el ancho de los intervalos será:

$$W = (B-A) / k$$

- Los límites de los intervalos ademas de A y B seran:

$$A+W, A+2W, \ldots, A + (k-1)W$$

- Formas de determinar k:
  - Fórmula de Sturges: $k=\log_2(n+1)$, n: número de observaciones.
  - Fórmula de Friedman-Diaconis: $W=2*IQR*n^{-1/3}$, donde IQR = Q3 - Q1. Luego k=(B-A)/W
  - Fórmula de Scott: $W=3.5*s*n^{-1/3}$, donde s es la desviación estándar. Luego k=(B-A)/W.

  Este metodo es considerado como no supervisado, global y estatico.
- Problemas
  - (a) No supervisado
  - (b) De donde proviene k?
  - (c) Sensitivo a outliers.

Ejemplo: Discretización usando intervalos de igual amplitud (libreria Dprep)

```
> dbupa=disc.ew(bupa,1:6,out="num")
> table(dbupa[,1])
 1  7  8  9 10 11 12 13 14 15 16 17 18
 1  3  2 22 39 52 60 79 35 23 19  7  3
> table(dbupa[,2])
 1  2  3  4  5  6  7  8  9 10 11 12 13
 1  6 30 62 80 58 43 27 21  8  4  3  2
> table(dbupa[,3])
  1   2   3   4   5   6   7   8   9  11  12  16
 24 105 114  48  19  14   8   3   5   1   1   3
> table(dbupa[,4])
  1   2   3   4   5   6   7   8   9  10  11  13  14  15  16
  3  21  76 108  71  23  16  10   6   3   3   1   1   1   2
> table(dbupa[,5])
  1   2   3   4   5   6   7   8   9  11  12  15
172  83  39  17   9  11   3   3   1   5   1   1
> table(dbupa[,6])
  1   2   3   4   5   6   7   8  10  11  13
134  56  50  57   6  23  10   4   1   2   2
```

Intervalos de igual frecuencia

- Dividir el rango en *k* intervalos
- Cada intervalo contendrá aproximadamente el mismo número de instancias.
- El proceso de discretización ignora la información de la clase.

Ejemplo: Intervalos de igual frecuencia

```
>args(disc.ef)
function (data, varcon, k, out = c("symb",
"num")) NULL
>dbupa=disc.ef(bupa,1:6,10,out="num")
> table(dbupa[,1])
 1  2  3  4  5  6  7  8  9 10
35 35 35 35 35 35 35 35 35 30
> table(dbupa[,2])
1  2  3  4  5  6  7  8  9 10
35 35 35 35 35 35 35 35 35 30
> table(dbupa[,3])
1  2  3  4  5  6  7  8  9 10
35 35 35 35 35 35 35 35 35 30
> table(dbupa[,4])
 1  2  3  4  5  6  7  8  9 10
35 35 35 35 35 35 35 35 35 30
```

Discretización basada en Entropía

- Fayyad and Irani (1993)
- Los métodos basados en entropía utilizan la información existente de la clase en los datos.
- La entropía (o contenido de información) es calculada en base a la clase. Intuitivamente, encuentra la mejor partición de cada atributo de tal forma que las divisiones sean las mas puras posible, i.e. la mayoría de los valores en una division corresponden a la misma clase. Formalmente, es caracterizado por encontrar la partición con la máxima ganancia de información.
- Es un metodo de discretizacion supervisado, global y estatico.

Discretización basada en Entropía (cont)

- Sea S el siguiente conjunto de 9 pares (atributo-valor, clase), S = {(0,Y), (4,Y), (12,Y), (16,N), (16,N), (18,Y), (24,N), (26,N), (28,N)}. Sea p_1 = 4/9 la fracción de pares con clase=Y, y p_2 = 5/9 la fracción de pares con clase=N.
- La entropía (o contenido de información) para S se define como:

  Entropy(S) = - p_1*$\log_2(p_1)$ – p_2*$\log_2(p_2)$.

En este caso, Entropy(S)=0.991076.

- Si la entropía es pequeña, entonces el conjunto es relativamente puro. El valor más pequeño posible es 0.
- Si la entropía es grande, entonces el conjunto esta muy mezclado. El valor más grande posible de entropia es 1, el cual es obtenido cuando p_1=p_2=0.5

Discretización basada en Entropía (cont)

- Si el conjunto de muestras S es particionado en dos intervalos S_1 y S_2 usando el punto de corte T, la entropía después de particionar es:

$$E(S,T) = \frac{|S_1|}{|S|} Ent(S_1) + \frac{|S_2|}{|S|} Ent(S_2)$$

donde | | denota cardinalidad. El punto de corte T se escoge de los puntos medios de los valores del atributo, i e: {2, 8, 14, 16, 17, 21, 25, 27}. Por ejemplo si T: valor de atributo=14

S_1= (0,Y), (4,Y), (12,Y) y S_2= (16,N), (16,N), (18,Y), (24,N), (26,N), (28,N)

E(S,T)=(3/9)*E(S_1)+(6/9)*E(S_2)=3/9*0+(6/9)* 0.6500224

E(S,T)=.4333

Ganancia de información de la partición, Gain(S,T) = Entropy(S) - E(S,T).

Gain=.9910-.4333=.5577

Discretización basada en Entropía (cont)

Igualmente, para T: v=21 se obtiene

Ganancia de Información=.9910-.6121=.2789.

Por lo que v=14 es una mejor partición.

- El objetivo de este algoritmo es encontrar la partición con la máxima ganancia de información. La ganancia máxima se obtiene cuando E(S,T) es mínima.
- La mejor partición (es) se encuentran examinando todas las posibles particiones y seleccionando la óptima. El punto de corte que minimiza la función de entropía sobre todos los posibles puntos de corte se selecciona como una discretización binaria.
- El proceso es aplicado recursivamente a particiones obtenidas hasta que se cumpla algún criterio de parada, e.g.,

$$Ent(S) - E(T,S) > \delta$$

Discretización basada en Entropía (cont)

Donde:

$$\partial > \frac{\log(N-1)}{N} + \frac{\Delta(T,S)}{N}$$

y,

$$\Delta(S,T) = \log_2(3^c - 2) - [cEnt(S) - c_1 Ent(S_1) - c_2 Ent(S_2)]$$

Aquí c es el número de clases en S, c_1 es el número de clases en S_1 y c_2 es el número de clases en S_2. Esto es llamado el Principio de Longitud de Descripción Mínima (MDLP)

Efectos de la Discretización

- Los resultados experimentales indican que después de la discretización
  - El tamaño del conjunto de datos puede ser reducido (Rough sets).
  - La exactitud de la clasificación puede mejorar