Instituto Politécnico Nacional
Escuela Superior de Cómputo
Secretaría Académica
Departamento de Ingeniería en Sistemas Computacionales

Minería de datos (*Data Mining*)
Árboles de decisión

1

Profesora: *Dra. Fabiola Ocampo Botello*

Clasificación

La clasificación se refiere a la tarea de clasificar objetos a una de varias categorías predefinidas.

La entrada de datos para la clasificación se compone de una serie de registros, donde cada registro representa una instancia y se caracteriza por ser una tupla (x, *y*) donde x es el conjunto de atributos y *y* es un atributo especial, la etiqueta de la clase.

Por ejemplo, suponga que se tiene la clase persona* de la Escuela: profesor, alumnos, pae y administradores. El conjunto de atributos (x) contiene los datos identificados de las personas y la variable *y* es de tipo discreta que representa las diversas clases o categorías que puede tener x.

2

* Imagine una clase disjunta, total. En el que todos las subclases tienen los mismos atributos.

3

Tan, Steinbach, Karpatne, & Kumar (2005) establecen la clasificación como sigue:

Definición de clasificación: Es la tarea de aprendizaje que considera una función *f* que asocia cada conjunto de atributos x a una de las clases predefinidas y etiquetas en *y*.

Esta función f es conocida de manera informa como modelo de clasificación. Un modelo de clasificación es útil por las siguientes razones:

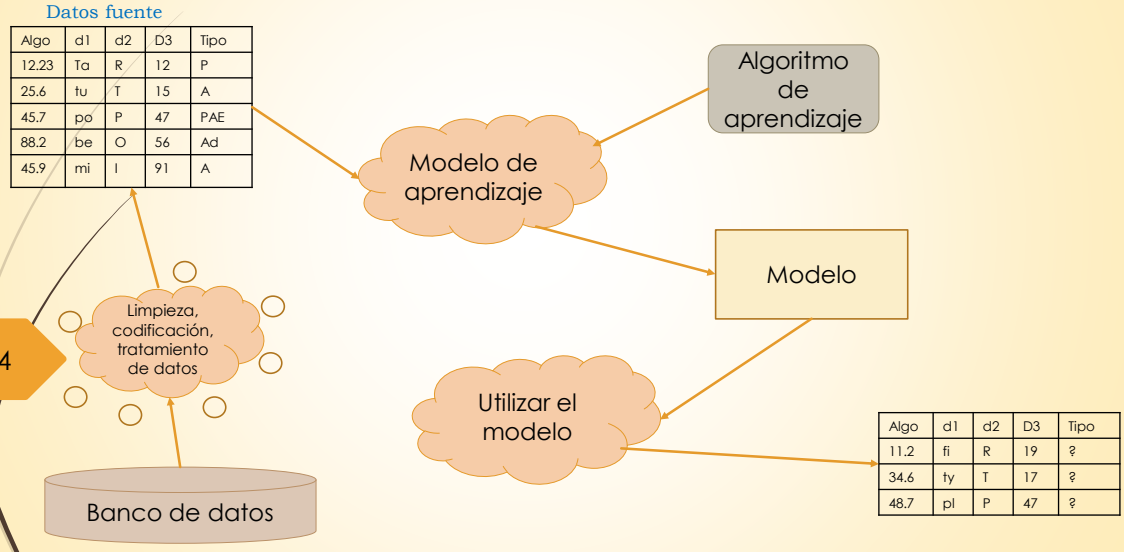
- Modelo descriptivo. Un modelo de clasificación puede servir como una herramienta para distinguir objetos de diferentes clases.
- Modelo de predicción. Un modelo de clasificación puede servir para para predecir la etiqueta de clase de un registro desconocido.

Las técnicas de clasificación son más adecuadas para predecir o describir conjuntos de datos de categorías binarias o nominales. Son menos efectivas en categorías ordinales porque no consideran el orden jerárquico de los grupos.

4

Los árboles de decisión son una de las técnicas de clasificación.

Proceso para construir un modelo de clasificación



La evaluación del desempeño de un modelo de clasificación considera dos aspectos:

1. La cantidad de registros previstos por el modelo de forma adecuada.
2. La cantidad de registros previstos por el modelo de forma inadecuada.

Lo anterior se presenta en una **matriz de confusión**.

Ejemplo de una matriz de confusión:

		Clase prevista	
		Clase 1	Clase 0
Clase actual	Clase 1	f11	f10
	Clase 0	f01	f00

5

$$\text{Accuracy} = \frac{\text{Número de predicciones correctas}}{\text{Número total de predicciones}}$$

$$\text{Accuracy} = \frac{f11+f00}{f11+f10+f01+f00}$$

$$\text{Error rate} = \frac{\text{Número de predicciones incorrectas}}{\text{Número total de predicciones}}$$

$$\text{Error rate} = \frac{f10+f01}{f11+f10+f01+f00}$$

Árbol de decisión

6

Rokach & Maimon (2010) establecen que un árbol de decisión es un algoritmo de clasificación expresado como un particionamiento recursivo de las instancias presentes en un espacio.

Es un árbol dirigido con diversos nodos:

- Un nodo raíz. No tiene arcos de entrada.
- Nodos internos. Tiene exactamente un arco de entrada y puede tener uno o varios arcos de salida.
- Nodos hoja. También llamados nodos terminales. Tiene un sólo arco de entrada y ningún arco de salida.

Los nodos internos parten el conjunto de instancias en dos o más sub conjuntos conforme a una función discreta. En el caso más simple, cada pregunta considera un solo atributo, de este modo las instancias son organizadas considerando ese atributo.
Cada hoja representa una clase.

7

Dunham (2002) establece la siguiente definición de árbol de decisión.

Definición: Dada una base de datos D = {t1, t2, …, tn}, donde ti = {ti1, ti2, …, tih} y el esquema de la base de datos contiene los siguientes atributos {A1, A2, …, Ah}. También se considera un conjunto de clases C = {C1, C2, …, Cm}. Un árbol de decisión (*decision tree, DT*) o árbol de clasificación es un árbol asociado con D que tiene las propiedades siguientes:

1. Cada nodo interno se etiqueta con un atributo Ai.
2. Cada arco es etiquetado con un predicado que puede ser aplicado al atributo asociado con el padre.
3. Cada nodo hoja es etiquetado con una clase Cj.

Una vez resuelto el problema de clasificación se usa los árboles de decisión en dos procesos:

1. Árbol de decisión de inducción. Construye un DT usando un conjunto de datos de entrenamiento.
2. Para cada ti Є D, se aplica el DT para determinar su clase.

8

Dunham (2002) establece las siguientes ventajas desventajas de los árboles de decisión.

Ventajas de la utilización de árboles de decisión para clasificación

- Las reglas generadas son fáciles de interpretar y entender.
- Cada tupla de la base de datos es filtrada por el árbol.
- Puede ser construido para datos con muchos atributos.

Desventajas de la utilización de árboles de decisión para clasificación

- No manejan adecuadamente los datos continuos.
- Los dominios de los atributos deben ser divididos en categorías para tener un manejo adecuado.
- No todos los problemas de clasificación se solucionan con DT.
- Puede presentar problemas con los datos faltantes.

9

Otras características de los árboles de decisión (Dunham, 2002):

- Debido a que el DT es construido a partir de un conjunto de datos de entrenamiento, podría ser necesario realizar un ajuste.
- El proceso de construcción del árbol ignora las correlaciones existentes entre los datos.
- Los atributos de particionamiento se refieren a los atributos de la base de datos que serán usados para etiquetar los nodos en el árbol y que permitirán realizar las divisiones.
- Los predicados de particionamiento se refieren a las etiquetas que tendrán los arcos en el árbol.
- El árbol se construye de arriba debajo de forma recursiva partiendo del conjunto de datos de entrenamiento.
- Considerando el conjunto de datos de entrenamiento, se elige primero el atributo que "mejor" particiona.
- Los algoritmos difieren en la forma en que determinan el "mejor" atributo y el "mejor" predicado que utilizan para realizar el particionamiento.

10

Muchos de los algoritmos de DT consideran los siguientes aspectos (Dunham, 2002):

- Elección del atributo de particionamiento
- El orden de los atributos de particionamiento
- Particionamiento. Dominio, número de particiones
- Estructura del árbol
- Criterio de detención o paro
- Datos de entrenamiento
- Poda. Podría ser necesario realizar algunas modificaciones para mejorar el desempeño del árbol. La fase de poda podría eliminar comparaciones redundantes o eliminar subárboles para mejorar el desempeño.

El conjunto de datos de entrenamiento y el algoritmo de árbol utilizado determinan la forma del árbol.

Algoritmo ID3

Sancho Capparini, Fernando (2009) establece que en 1979, J. Ross Quinlan presentó un método para construir árboles de decisión con muy buenas características: un buen balanceo y un tamaño pequeño, basándose en el menor número de preguntas posibles para poder encontrar las respuestas en todos los casos, si es posible. Para lo cual se basó en la teoría de la información desarrollada por Shannon en 1948. Este algoritmo se llama ID3 y es muy utilizado actualmente.

11

Sancho Capparini, Fernando (2009) indica que este árbol usa el concepto de **Ganancia de Información** para seleccionar el atributo más útil en cada paso. Utiliza un método voraz para decidir la pregunta que mayor ganancia proporcione en cada paso, esto es, aquella que permite separar mejor los ejemplos respecto a la clasificación final.

La estrategia básica del ID3 es elegir los atributos de particionamiento con la mayor información.

El concepto usado para cuantificar la información se llama **entropía**. La entropía es usada para medir la cantidad de incertidumbre en un conjunto de datos.

12

Sancho Capparini, Fernando (2009) presenta dos ejemplos para comprender la incertidumbre:

1. En una muestra totalmente homogénea, en la que todos los elementos se clasifican por igual tiene una incertidumbre mínima, esto es, no se tienen dudas de cuál es la clasificación de cualquiera de sus elementos. En este caso la incertidumbre (entropía) es cero.
2. En una muestra igualmente distribuida en el que se tienen el mismo número de casos en cada posible clasificación tiene una incertidumbre máxima. En este caso, la incertidumbre (entropía) es 1.

La fórmula para calcular la entropía es (Sancho, 2009):

$$E(S) = \sum_{i=1}^{C} -p_i \log_2(p_i)$$

Dunham (2002) define la Ganancia de información como la diferencia entre la entropía del conjunto de datos original y la suma de las entropías de cada uno de las divisiones del conjunto de datos.

13

La fórmula de Ganancia es:

Gain(T,X) = E(T) − E(T,X)

Donde:
S es el conjunto de datos analizados
C es el número de clases
pi es la proporción de casos que hay de la clase i en la muestra analizada

Ejemplo: Calcular la entropía de los datos presentados en la tabla número 1.

Tabla 1. *Datos de ejemplo para calcular la entropía.* Elaborada por la autora de este material educativo. Considerando lo expuesto en Sancho (2009).

JuegaGolf	Panorama	Temperatura	Humedad	Viento
No	Lluvioso	Caliente	Alta	FALSO
No	Lluvioso	Caliente	Alta	VERDADERO
Si	Nublado	Caliente	Alta	FALSO
Si	Soleado	Templado	Alta	FALSO
Si	Soleado	Frío	Normal	FALSO
No	Soleado	Frío	Normal	VERDADERO
Si	Nublado	Frío	Normal	VERDADERO
No	Lluvioso	Templado	Alta	FALSO
Si	Lluvioso	Frío	Normal	FALSO
Si	Soleado	Templado	Normal	FALSO
Si	Lluvioso	Templado	Normal	VERDADERO
Si	Nublado	Templado	Alta	VERDADERO
Si	Nublado	Caliente	Normal	FALSO
No	Soleado	Templado	Alta	VERDADERO

14

15

				Count
Panorama	Lluvioso	JuegaGolf	No	3
			Si	2
	Nublado	JuegaGolf	Si	4
	Soleado	JuegaGolf	No	2
			Si	3

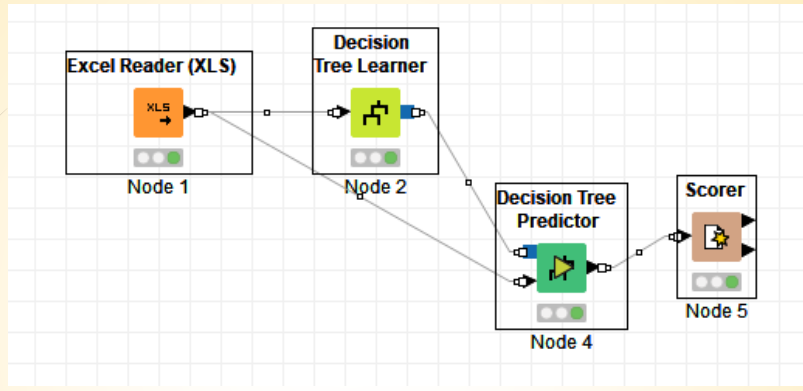
				Count
Viento	Falso	JuegaGolf	No	2
			Si	6
	Verdadero	JuegaGolf	No	3
			Si	3

				Count
Humedad	Alta	JuegaGolf	No	4
			Si	3
	Normal	JuegaGolf	No	1
			Si	6

				Count
Temp	Caliente	JuegaGolf	No	2
			Si	2
	Frio	JuegaGolf	No	1
			Si	3
	Templado	JuegaGolf	No	2
			Si	4

Las tablas de datos presentadas en esta diapositiva fueron desarrollados por la autora de este material educativo.

16



Confusion Matrix - 2:5 - Scorer

File Hilite

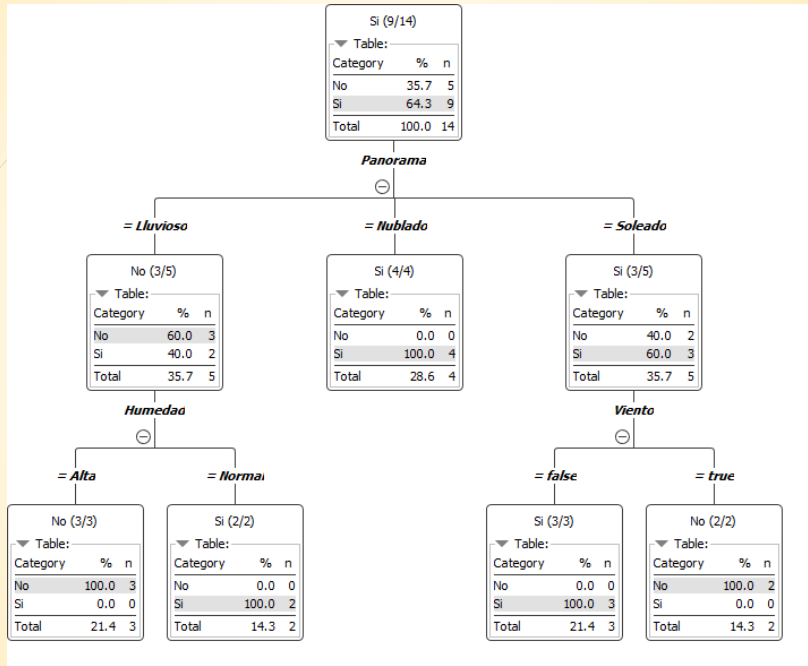
JuegaGolf ...	No	Si
No	5	0
Si	0	9

Correct classified: 14 Wrong classified: 0

Accuracy: 100 % Error: 0 %

Cohen's kappa (κ) 1

17



18

Tan et al., (2005) señalan lo siguiente:

El modelo de clasificación se crea utilizando un conjunto de instancias conocidas, a las cuales se les llama conjunto de entrenamiento. Este conjunto tiene datos en los atributos y etiquetas de clases para su identificación en cada tupla.

El proceso para crear un modelo de clasificación dado un conjunto de entrenamiento se conoce como aprendizaje algorítmico.

El proceso de usar un algoritmo de aprendizaje para construir un modelo de clasificación a partir de los datos de entrenamiento se conoce como inducción. Este proceso también se conoce como modelo de aprendizaje o construcción del modelo. El proceso de aplicar un modelo de clasificación en instancias de prueba desconocidas para predecir sus etiquetas de clase se conoce como deducción.

El proceso de clasificación implica dos pasos:

1. Aplicar un algoritmo de aprendizaje a los datos de entrenamiento para aprender un modelo, y luego
2. Aplicar el modelo para asignar etiquetas a instancias no etiquetadas

Rokach & Maimon (2015) indican que construir un árbol de decisión óptimo es una tarea difícil.

En términos generales, estos métodos pueden ser divididos en dos grupos: de arriba hacia abajo (*top-down*) y de abajo hacia arriba (*bottom-up*) con clara preferencia en la literatura al primer grupo.

Ejemplos de árboles de tipo arriba hacia abajo son:
- ID3
- C4.5
- CART

Algunos de ellos incluyen dos fases conceptuales: crecimiento y poda (C4.5 y CART). Otros algoritmos de entrenamiento sólo consideran la fase de crecimiento.

19

Rokach & Maimon (2015) establecen los siguientes criterios de detención del crecimiento:

- Todas las instancias en el conjunto de entrenamiento pertenecen a un solo valor de *y* (clase).
- Se ha alcanzado la profundidad máxima del árbol.
- El número de casos en el nodo terminal es menor que el número mínimo de casos para los nodos principales (padres).

Los mismos autores señalan que evaluar el desempeño de un árbol de clasificación es una tarea fundamental en el aprendizaje automático.
Algunos de los criterios que presentan son:
- La matriz de confusión, la cual presenta la cantidad de elementos que han sido clasificados correcta e incorrectamente.
- El coeficiente de correlación.

20

21

Tan et al., (2005) establecen que hay medidas que pueden usarse para determinar la bondad de una condición de prueba de un atributo. Estas medidas intentan dar preferencia a las condiciones de prueba de atributos que dividen las instancias de entrenamiento en subconjuntos más puros en los nodos secundarios.

La impureza de un nodo mide qué tan diferentes son las etiquetas de clase para las instancias de datos que pertenecen a un nodo común.

Las medidas para evaluar la impureza de un nodo:

$$\text{Entropy} = -\sum_{i=0}^{c-1} p_i(t)\log_2 p_i(t),$$

$$\text{Gini index} = 1-\sum_{i=0}^{c-1} p_i(t)^2,$$

$$\text{Classification error} = 1-\max_i[p_i(t)],$$

22

Calcular la impureza de los siguientes nodos:

Node N_1	Count
Class=0	0
Class=1	6

Node N_2	Count
Class=0	1
Class=1	5

Node N_3	Count
Class=0	3
Class=1	3

$$\text{Entropy} = -\sum_{i=0}^{c-1} p_i(t)\log_2 p_i(t),$$

$$\text{Gini index} = 1-\sum_{i=0}^{c-1} p_i(t)^2,$$

$$\text{Classification error} = 1-\max_i[p_i(t)],$$

Ejemplo tomado de Tan et al., (2005)

Ejemplos de cálculo de impureza:

Node N_1	Count
Class=0	0
Class=1	6

$\text{Gini} = 1 - (0/6)^2 - (6/6)^2 = 0$
$\text{Entropy} = -(0/6)\log_2(0/6) - (6/6)\log_2(6/6) = 0$
$\text{Error} = 1 - \max[0/6, 6/6] = 0$

Node N_2	Count
Class=0	1
Class=1	5

$\text{Gini} = 1 - (1/6)^2 - (5/6)^2 = 0.278$
$\text{Entropy} = -(1/6)\log_2(1/6) - (5/6)\log_2(5/6) = 0.650$
$\text{Error} = 1 - \max[1/6, 5/6] = 0.167$

Node N_3	Count
Class=0	3
Class=1	3

$\text{Gini} = 1 - (3/6)^2 - (3/6)^2 = 0.5$
$\text{Entropy} = -(3/6)\log_2(3/6) - (3/6)\log_2(3/6) = 1$
$\text{Error} = 1 - \max[3/6, 3/6] = 0.5$

Ejemplo tomado de Tan et al., (2005)

23

Tan et al., (2005) establecen que para determinar la bondad de una condición de prueba de atributo, necesitamos comparar el grado de impureza del nodo primario (antes de dividir) con el grado ponderado de impureza de los nodos secundarios (después de dividir).

Cuanto mayor sea su diferencia, mejor será la condición de la prueba. Esta diferencia, Δ, también denominada ganancia de pureza de una condición de prueba de atributo, se puede definir de la siguiente manera:

$$\Delta = I(\text{parent}) - I(\text{children})$$

El algoritmo de aprendizaje del árbol de decisión selecciona la condición de prueba de atributo que muestra la máxima ganancia.

24

25

Gain Ratio

Un problema que se presenta con las medidas de impureza como la entropía y el Gini Index es que tienden a favorecer atributos cualitativos con un gran número de valores diferentes.

Considere el siguiente ejemplo tomado de Tan et al., (2005).

Table 3.3. A sample data for the loan borrower classification problem.

ID	Home Owner	Marital Status	Annual Income	Defaulted?
1	Yes	Single	125000	No
2	No	Married	100000	No
3	No	Single	70000	No
4	Yes	Married	120000	No
5	No	Divorced	95000	Yes
6	No	Single	60000	No
7	Yes	Divorced	220000	No
8	No	Single	85000	Yes
9	No	Married	75000	No
10	No	Single	90000	Yes

26

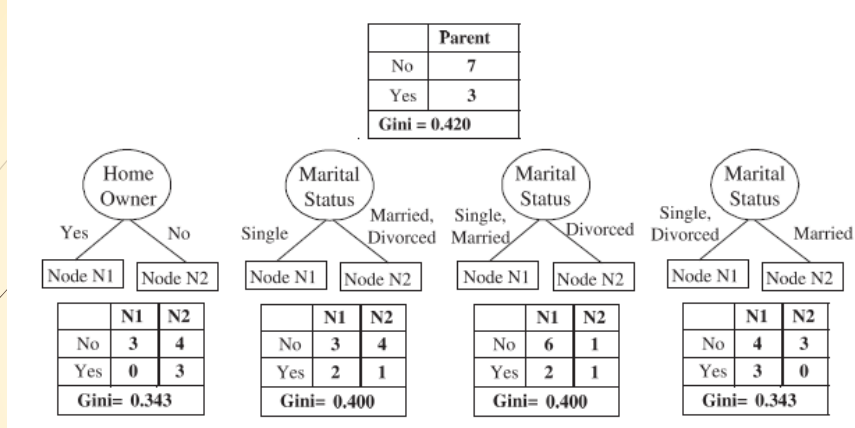


Figure 3.13. Splitting criteria for the loan borrower classification problem using Gini index.

Si se considerara el atributo ID como parte del conjunto de entrenamiento ¿Qué sucedería?

Tan et al., (2005) establecen dos formas de tratar con este problema:

- Una forma de solucionar el problema es crear árboles binarios, lo cual evita manejar atributos con un número muy variable de particiones. Esta estrategia es empleada por el algoritmo CART.
- Otra forma es modificar el criterio de particionamiento para tomar en cuenta el número de particiones que genera el atributo, por ejemplo en el algoritmo C4.5 se utiliza una medida llamada Gain Ratio, la cual es usada para atributos que producen un gran número de nodos hijo.

La fórmula del Gain Ratio es la siguiente:

27

$$\text{Gain ratio} = \frac{\Delta_{\text{info}}}{\text{Split Info}} = \frac{\text{Entropy(Parent)} - \sum_{i=1}^{k} \frac{N(v_i)}{N}\text{Entropy}(v_i)}{-\sum_{i=1}^{k} \frac{N(v_i)}{N}\log_2 \frac{N(v_i)}{N}}$$

Para ilustrar lo anterior considere el siguiente ejemplo:

Table 3.5. Data set for Exercise 2.

Customer ID	Gender	Car Type	Shirt Size	Class
1	M	Family	Small	C0
2	M	Sports	Medium	C0
3	M	Sports	Medium	C0
4	M	Sports	Large	C0
5	M	Sports	Extra Large	C0
6	M	Sports	Extra Large	C0
7	F	Sports	Small	C0
8	F	Sports	Small	C0
9	F	Sports	Medium	C0
10	F	Luxury	Large	C0
11	M	Family	Large	C1
12	M	Family	Extra Large	C1
13	M	Family	Medium	C1
14	M	Luxury	Extra Large	C1
15	F	Luxury	Small	C1
16	F	Luxury	Small	C1
17	F	Luxury	Medium	C1
18	F	Luxury	Medium	C1
19	F	Luxury	Medium	C1
20	F	Luxury	Large	C1

$$\text{Entropy(parent)} = -\frac{10}{20}\log_2\frac{10}{20} - \frac{10}{20}\log_2\frac{10}{20} = 1.$$

Cálculos para Gender

$$\begin{aligned}\text{Entropy(children)} &= \frac{10}{20}\left[-\frac{6}{10}\log_2\frac{6}{10} - \frac{4}{10}\log_2\frac{4}{10}\right] \times 2 = 0.971\\ \text{Gain Ratio} &= \frac{1-0.971}{-\frac{10}{20}\log_2\frac{10}{20} - \frac{10}{20}\log_2\frac{10}{20}} = \frac{0.029}{1} = 0.029\end{aligned}$$

Cálculos para Car Type

$$\begin{aligned}\text{Entropy(children)} &= \frac{4}{20}\left[-\frac{1}{4}\log_2\frac{1}{4} - \frac{3}{4}\log_2\frac{3}{4}\right] + \frac{8}{20}\times 0\\ &\quad + \frac{8}{20}\left[-\frac{1}{8}\log_2\frac{1}{8} - \frac{7}{8}\log_2\frac{7}{8}\right] = 0.380\\ \text{Gain Ratio} &= \frac{1-0.380}{-\frac{4}{20}\log_2\frac{4}{20} - \frac{8}{20}\log_2\frac{8}{20} - \frac{8}{20}\log_2\frac{8}{20}} = \frac{0.620}{1.52} = 0.41\end{aligned}$$

Cálculos para Customer ID

$$\begin{aligned}\text{Entropy(children)} &= \frac{1}{20}\left[-\frac{1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}\right] \times 20 = 0\\ \text{Gain Ratio} &= \frac{1-0}{-\frac{1}{20}\log_2\frac{1}{20}\times 20} = \frac{1}{4.32} = 0.23\end{aligned}$$

Figuras tomadas de Tan et al., (2005).

Poda de árboles

Rokach & Maimon (2015) establecen que el empleo de criterios estrictos de detención tiende a crear árboles de decisión pequeños y mal balanceados. Por otro lado, el uso de otros criterios de detención tiende a generar grandes árboles de decisión que están sobreajustados para el conjunto de entrenamiento.

Breiman *et al*. (1984, citado en Rokach & Maimon, 2015) desarrolló una metodología de poda basada en un criterio de detención que permite que el árbol de decisión sobreajuste el conjunto de entrenamiento, en donde el árbol sobreajustado se corta en un árbol más pequeño eliminando las subramas que no contribuyen a la precisión de la generalización.

29

Dunham (2012) establece que una vez que un árbol es construido, podría ser necesarios realizar algunos ajustes para mejorar el desempeño durante la fase de clasificación. La poda podría remover comparaciones redundantes o remover subárboles completos para mejorar el desempeño.
Algunas porciones del árbol podrían ser removidas o combinadas para reducir el tamaño total del árbol.

En el algoritmo C4.5 se proponen dos métodos de poda:

- Un subárbol se reemplaza por un nodo hoja si este reemplazo da como resultado una tasa de error cercana a la del árbol original. El reemplazo de subárbol funciona desde la parte inferior del árbol hasta la raíz.

30

- Otra estrategia de poda, llamada elevación de subárbol, reemplaza un subárbol por su subárbol más utilizado. Aquí un subárbol se eleva desde su ubicación actual a un nodo más arriba en el árbol. Nuevamente, se debe determinar el aumento en la tasa de error para este reemplazo.

Cierre del tema

- La técnica de árboles es fácil de entender, pero podrían generar sobreajustes.
- El algoritmo ID3 es sólo utilizado para datos categóricos.
- Los algoritmos C4.5 y C5, permiten el uso de datos continuos y técnicas mejoradas para el particionamiento.
- El algoritmo CART crea árboles binarios con grandes niveles de profundidad.

31

Referencias bibliográficas

Aguayo Canela, Mariano y Lora Monge, E. (2007). Cómo realizar "paso a paso" un contraste de hipótesis con SPSS para Windows: (III) Relación o asociación y análisis de la dependencia (o no) entre dos variables cuantitativas. Correlación y regresión lineal simple. *Documento de la Fundación Andaluza Beturia para la Investigación en Salud (fabis.org).* Dot. Núm. 0702005. Disponible en: http://www.fabis.org/html/archivos/docuweb/contraste_hipotesis_3r.pdf

Babbie R. Earl. (1988). *Métodos de investigación por encuesta.* Fondo de Cultura Económica. México.

Bennet, Briggs & Triola (2011). *Razonamiento estadístico.* Pearson. México.

Castillo M., A. (2013). *Estadística aplicada.* México, ed. Trillas.

Crear un gráfico de líneas. "s/f". *Fundación esplai. Ordenador práctico.* Disponible en: https://ordenadorpractico.es/mod/assign/view.php?id=274

Dunham, M. H. (2002). *Data mining: introductory and advanced topics.* Prentice Hall.

Ejemplos de tipos de representación gráfica. "s/f". *Material docente de la unidad de bioestadística clínica del Hospital Universitario Ramón y Cajal.* Disponible en: http://www.hrc.es/bioest/Ejemplos_histo.html

Han, Jiawei; Kamber, Micheline & Pei, Jian. (2012). *Data Mining: concepts and techniques.* Third edition. Morgan Kaufman Series.

Hernández Sampieri, R.; Fernández Collado, C; Baptista Lucio, P. (2003). *Metodología de la Investigación.* Tercera Edición. Editorial Mc. Graw Hill. D. F. México.

Mason, Lind & Marchal. (2000). *Estadística para administración y Economía.* Alfaomega. México.

Rodríguez Suárez, Yuniet, Díaz Amador, Anolandy, Herramientas de Minería de Datos. *Revista Cubana de Ciencias Informáticas* [en linea] 2009, 3 (Julio-Diciembre) : [Fecha de consulta: 24 de julio de 2019] Disponible en: <http://google.redalyc.org/articulo.oa?id=378343637009> ISSN 1994-1536

Sahu, Hemlata; Shrma, Shalini; Gondhalakar, Seema. (2011). A Brief Overview on Data Mining Survey. *International Journal of Computer Technology and Electronics Engineering (IJCTEE).* Vol.1.

32