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

Minería de datos (*Data Mining*)
Medidas de particionamiento en árboles de decisión

1

Profesora: *Dra. Fabiola Ocampo Botello*

¿Cómo construir el árbol de decisión?

Tan, Steinbach & Kumar Vipin (2014) mencionan que la construcción de un árbol de decisión en dos etapas:

Encontrar el árbol óptimo es computacionalmente inviable debido al tamaño exponencial del espacio de búsqueda.

2

Uno de esos algoritmos es el **algoritmo de Hunt**, que es la base de muchos algoritmos de inducción de árboles de decisión existentes, incluidos ID3, C4.5 y CART.

Algoritmo de Hunt

Tan, Steinbach & Kumar Vipin (2014)

En el algoritmo de Hunt, un árbol de decisiones crece de forma recursiva dividiendo los registros de entrenamiento en subconjuntos sucesivamente más puros.

Sea Dt el conjunto de registros de entrenamiento que están asociados con el nodo t y y = {y1, y2, … , yc} sean las etiquetas de clase

Páso 1: si todos los registros de Dt pertenecen a la misma clase yt, entonces t es un nodo hoja etiquetado como yt.

3

Paso 2: Si Dt contiene registros que pertenecen a más de una clase, se selecciona una condición de prueba de atributo para dividir los registros en subconjuntos más pequeños. Se crea un nodo hijo para cada resultado de la condición de prueba y los registros en Dt se distribuyen a los hijos según los resultados.

Métodos para expresar condiciones de prueba de atributos

Tan, Steinbach & Kumar Vipin (2014)

Los algoritmos de inducción de árboles de decisión deben proporcionar un método para expresar una condición de prueba de atributo y sus resultados correspondientes para diferentes tipos de atributo.

Atributos binarios. La condición de prueba para un atributo binario genera dos resultados potenciales.

4

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

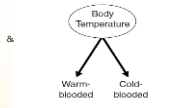

Figure 4.8. Test condition for binary attributes.

Atributos nominales. Dado que un atributo nominal puede tener muchos valores, su condición de prueba se puede expresar de dos formas,

Para una división de múltiples vías , donde el número de resultados depende del número de valores distintos para el atributo correspondiente.

Por otro lado, algunos algoritmos de árbol de decisión, como CART, producen solo divisiones binarias al considerar todas las formas $2^{k-1} - 1$ de crear una partición binaria de k valores de atributo.

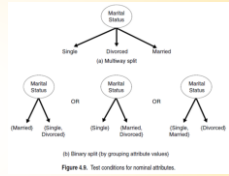

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Atributos ordinales Los atributos ordinales también pueden producir divisiones binarias o de múltiples vías. Los valores de atributo ordinales se pueden agrupar siempre que la agrupación no viole la propiedad de orden de los valores de atributo.

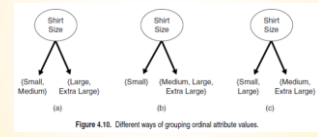

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Atributos continuos Para los atributos continuos, la condición de prueba se puede expresar:

a) Como una prueba de comparación (A <v) o (A ≥ v) con resultados binarios. Para el caso binario, el algoritmo del árbol de decisión debe considerar todas las posibles posiciones de división v, y selecciona la que produce la mejor partición.

b) Una consulta de rango con resultados de la forma vi ≤ A <vi + 1, para i = 1, … , k. Para la división de múltiples vías, el algoritmo debe considerar todos los rangos posibles de valores continuos.

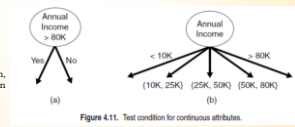

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Mediciones de nodos considerando las medidas de partición

Tan, Steinbach & Kumar Vipin (2005)

División de atributos binarios

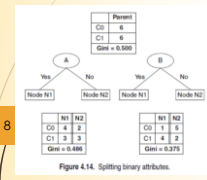

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Suponga que inicialmente hay dos formas de dividir el nodo de partida (Gini = 0.5), hay igual cantidad de registros en ese nodo.

Si se elige el atributo A para dividir los datos, el índice de Gini para el nodo N1 es 0,4898 y para el nodo N2 es 0,480. El promedio ponderado del índice de Gini para los nodos descendientes es (7/12) × 0,4898 + (5/12) × 0,480 = 0,486.

El promedio ponderado del índice de Gini para el atributo B es 0.375.

Dado que los subconjuntos del atributo B tienen un índice de Gini más pequeño, se prefiere al atributo A.

División de atributos nominales

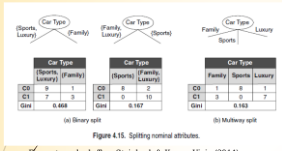

Figure 4.15. Splitting nominal attributes.

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Para la primera agrupación binaria del atributo Tipo de automóvil, el índice Gini de {Deportes, lujo} es 0.4922 y el índice Gini de {Familia} es 0.3750. El índice de Gini promedio ponderado para la agrupación es igual a 16/20 × 0,4922 + 4/20 × 0,3750 = 0,468

De manera similar, para la segunda agrupación binaria de {Deportes} y {Familia, lujo}, el índice de Gini promedio ponderado es 0.167. El segundo grupo tiene un índice de Gini más bajo porque sus subconjuntos correspondientes son mucho más puros.

Para la división de múltiples vías, el índice de Gini se calcula para cada valor de atributo. Dado que Gini ({Familia}) = 0.375, Gini ({Deportes}) = 0 y Gini ({Lujo}) = 0.219, el índice de Gini general para la división de múltiples vías es igual a 4/20 × 0,375 + 8/20 × 0 + 8/20 × 0,219 = 0,163.

9

La división de múltiples vías tiene un índice de Gini más pequeño en comparación con ambas divisiones de dos vías. Este resultado no es sorprendente porque la división de dos vías en realidad fusiona algunos de los resultados de una división de múltiples vías y, por lo tanto, da como resultado subconjuntos menos puros.

División de atributos continuos

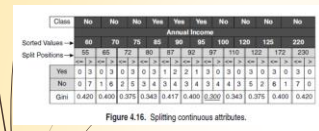

Figure 4.16. Splitting continuous attributes.

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Considere que la condición de prueba Ingreso anual ≤ **v** se usa para dividir los registros de capacitación para el problema de clasificación.
Un método de fuerza bruta para encontrar v es considerar cada valor del atributo en los N registros como una posición de división candidata.
Para cada candidato v, el conjunto de datos se escanea una vez para contar el número de registros con ingresos anuales menores o mayores que v.
Luego calculamos el índice de Gini para cada candidato y elegimos el que da el valor más bajo.
Este enfoque es computacionalmente costoso porque requiere operaciones O(N) para calcular el índice de Gini en cada posición de división candidata.

10

Para el primer candidato, v = 55, ninguno de los registros tiene ingresos anuales inferiores a $55K.
Como resultado, el índice de Gini para el nodo descendiente con Ingreso anual <$ 55K es cero.
Por otro lado, el número de registros con ingresos anuales mayores o iguales a $55K es 3 (para la clase Sí) y 7 (para la clase No), respectivamente.
Por tanto, el índice de Gini para este nodo es 0,420.
El índice de Gini general para esta posición dividida candidata es igual a 0 × 0 + 1 × 0,420 = 0,420.

Características de la inducción del árbol de decisión

Tan, Steinbach & Kumar Vipin (2014. Sección 4.3.7)

Características importantes de los algoritmos de inducción de árboles de decisión:

1) La inducción del árbol de decisiones es un enfoque no paramétrico para construir modelos de clasificación. En otras palabras, no requiere ningún supuesto previo con respecto al tipo de distribuciones de probabilidad satisfechas por la clase y otros atributos.

2) Encontrar un árbol de decisión óptimo es un problema NP-completo. Muchos algoritmos de árboles de decisión emplean un enfoque basado en la heurística para guiar su búsqueda en el vasto espacio de hipótesis.

11

3) Las técnicas desarrolladas para construir árboles de decisión hacen posible construir modelos rápidamente incluso cuando el tamaño del conjunto de entrenamiento es muy grande. Una vez que se ha construido un árbol de decisión, clasificar un registro de prueba es extremadamente rápido, en el peor de los casos de O(w), donde w es la profundidad máxima del árbol.

4) Los árboles de decisión, especialmente los árboles de menor tamaño, son relativamente fáciles de interpretar. Las precisiones de los árboles también son comparables a otras técnicas de clasificación para muchos conjuntos de datos simples.

5) Los árboles de decisión proporcionan una representación expresiva para aprender funciones valoradas discretamente. Sin embargo, no se generalizan bien a ciertos tipos de problemas booleanos. Un ejemplo notable es la función de paridad, cuyo valor es 0(1) cuando hay un número impar (par) de atributos booleanos con el valor Verdadero.

6) Los algoritmos de árbol de decisión son bastante robustos a la presencia de ruido, especialmente cuando se emplean métodos para evitar el sobreajuste.

7) La presencia de atributos redundantes no afecta negativamente la precisión de los árboles de decisión. Un atributo es redundante si está fuertemente correlacionado con otro atributo en los datos.
Uno de los dos atributos redundantes no se utilizará para dividir una vez que se haya elegido el otro atributo.
Sin embargo, si el conjunto de datos contiene muchos atributos irrelevantes, es decir, atributos que no son útiles para la tarea de clasificación, entonces algunos de los atributos irrelevantes pueden elegirse accidentalmente durante el proceso de crecimiento del árbol, lo que da como resultado un árbol de decisión más grande que necesario.

12

13

8) Dado que la mayoría de los algoritmos de árboles de decisión emplean un enfoque de particionamiento recursivo de arriba hacia abajo, el número de registros se reduce a medida que recorremos el árbol.
En los nodos hoja, el número de registros puede ser demasiado pequeño para tomar una decisión estadísticamente significativa sobre la representación de clase de los nodos. Esto se conoce como el **problema de la fragmentación de datos**. Una posible solución es no permitir más divisiones cuando el número de registros cae por debajo de un cierto umbral.

9) Un subárbol se puede replicar varias veces en un árbol de decisión, como se ilustra en la Figura 4.19. Esto hace que el árbol de decisiones sea más complejo de lo necesario y quizás más difícil de interpretar.

Tal situación puede surgir de implementaciones de árboles de decisión que se basan en una condición de prueba de atributo único en cada nodo interno. Dado que la mayoría de los algoritmos del árbol de decisión utilizan una estrategia de partición de divide y vencerás, la misma condición de prueba se puede aplicar a diferentes partes del espacio de atributos, lo que conduce al problema de replicación del subárbol.

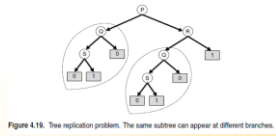

Figure 4.19. Tree replication problem. The same subtree can appear at different branches.

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

14

10) Cuando las condiciones de prueba usan un solo atributo a la vez. Como consecuencia, el procedimiento de crecimiento de árboles puede verse como el proceso de dividir el espacio de atributos en regiones disjuntas hasta que cada región contenga registros de la misma clase (ver Figura 4.20).
El límite entre dos regiones vecinas de diferentes clases se conoce como **límite de decisión**. Dado que la condición de prueba implica solo un atributo, los límites de decisión son rectilíneos; es decir, paralelo a los "ejes de coordenadas". Esto limita la expresividad de la representación del árbol de decisiones para modelar relaciones complejas entre atributos continuos. La figura 4.21 ilustra un conjunto de datos que no puede clasificarse eficazmente mediante un algoritmo de árbol de decisión que utiliza condiciones de prueba que involucran solo un atributo a la vez.

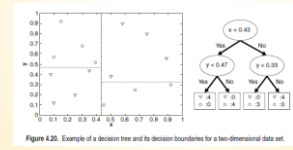

Figure 4.20. Example of a decision tree and its decision boundaries for a two-dimensional data set.

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

15

Continúa el número **10)** La figura 4.21 ilustra un conjunto de datos que no puede clasificarse eficazmente mediante un algoritmo de árbol de decisión que utiliza condiciones de prueba que involucran solo un atributo a la vez.

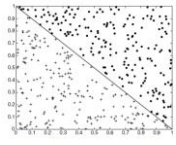

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Figure 4.21. Example of data set that cannot be partitioned optimally using test conditions involving single attributes.

11) Los estudios han demostrado que la elección de la medida de impurezas tiene poco efecto sobre el rendimiento de los algoritmos de inducción de árboles de decisión. Esto se debe a que muchas medidas de impurezas son bastante consistentes entre sí, como se muestra en la Figura 4.13. De hecho, la estrategia utilizada para podar el árbol tiene un mayor impacto en el árbol final que la elección de la medida de impurezas. (Página 172)

16

11) Los estudios han demostrado que la elección de la medida de impurezas tiene poco efecto sobre el rendimiento de los algoritmos de inducción de árboles de decisión. Esto se debe a que muchas medidas de impurezas son bastante consistentes entre sí, como se muestra en la Figura 4.13.
De hecho, la estrategia utilizada para podar el árbol tiene un mayor impacto en el árbol final que la elección de la medida de impurezas.

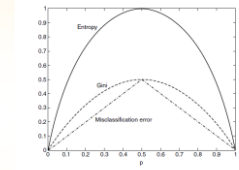

Figure 4.13. Comparison among the impurity measures for binary classification problems.

Figura tomada de Tan, Steinbach & Kumar Vipin (2014)

Sobreajuste del modelo

Tan, Steinbach & Kumar Vipin (2014. Sección 4.4)

Los errores cometidos por un modelo de clasificación generalmente se dividen en dos tipos:
a) errores de entrenamiento y
b) errores de generalización.

El **error de entrenamiento**, también conocido como error de resustitución o error aparente, es el número de errores de clasificación errónea cometidos en los registros de entrenamiento, mientras que el **error de generalización** es el error esperado del modelo en registros no vistos anteriormente.

17

Un buen modelo de clasificación no solo debe ajustarse bien a los datos de entrenamiento, sino que también debe clasificar con precisión los registros que nunca antes había visto.

Un modelo que se ajusta demasiado bien a los datos de entrenamiento puede tener un error de generalización más pobre que un modelo con un error de entrenamiento más alto. Esta situación se conoce como **sobreajuste del modelo (*overfitting model*).**

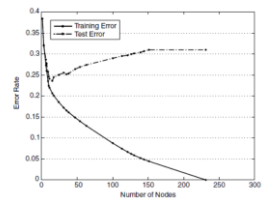

Figure 4.23. Training and test error rates.

Observe que las tasas de error de entrenamiento y prueba del modelo son grandes cuando el tamaño del árbol es muy pequeño. Esta situación se conoce como desajuste del modelo (*underfitting model*).

El desajuste se produce porque el modelo aún tiene que conocer la verdadera estructura de los datos. Como resultado, tiene un desempeño deficiente tanto en el entrenamiento como en los conjuntos de prueba.

A medida que aumenta el número de nodos en el árbol de decisión, el árbol tendrá menos errores de prueba y entrenamiento. Sin embargo, una vez que el árbol se vuelve demasiado grande, su tasa de error de prueba comienza a aumentar aunque su tasa de error de entrenamiento continúa disminuyendo. Este fenómeno se conoce como sobreajuste del modelo (*overfitting model*).

Para comprender el fenómeno de sobreajuste, tenga en cuenta lo siguiente:

- el error de entrenamiento de un modelo se puede reducir aumentando la complejidad del modelo
- Por ejemplo, los nodos hoja del árbol se pueden expandir hasta que se ajusten perfectamente a los datos de entrenamiento. Aunque el error de entrenamiento para un árbol tan complejo es cero, el error de prueba puede ser grande porque el árbol puede contener nodos que accidentalmente se ajustan a algunos de los puntos de ruido en los datos de entrenamiento.
- Dichos nodos pueden degradar el rendimiento del árbol porque no se generalizan bien en los ejemplos de prueba.

19

¿Por qué se presenta el sobreajuste del modelo (*overfitting model*)?

Sobreajuste por presencia de ruido

Table 4.3. An example training set for classifying mammals. Class labels with asterisk symbols represent mislabeled records.

Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Class Label
porcupine	warm-blooded	yes	yes	yes	yes
cat	warm-blooded	yes	yes	no	yes
bat	warm-blooded	yes	no	yes	no*
whale	warm-blooded	yes	no	no	no*
salamander	cold-blooded	no	yes	yes	no
komodo dragon	cold-blooded	no	yes	no	no
python	cold-blooded	no	no	yes	no
salmon	cold-blooded	no	no	no	no
eagle	warm-blooded	no	no	no	no
guppy	cold-blooded	yes	no	no	no

Table 4.4. An example test set for classifying mammals.

Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Class Label
human	warm-blooded	yes	no	no	yes
pigeon	warm-blooded	no	no	no	no
elephant	warm-blooded	yes	yes	no	yes
leopard shark	cold-blooded	yes	no	no	no
turtle	cold-blooded	no	yes	no	no
penguin	cold-blooded	no	no	no	no
eel	cold-blooded	no	no	no	no
dolphin	warm-blooded	yes	no	no	yes
spiny anteater	warm-blooded	no	yes	yes	yes
gila monster	cold-blooded	no	yes	yes	no

20

El problema de clasificación de mamíferos.
Dos de los diez registros de entrenamiento están mal etiquetados: los murciélagos (bat) y las ballenas (whale) se clasifican como no mamíferos en lugar de mamíferos.
En la figura 4.25 (a) se muestra un árbol de decisiones que se ajusta perfectamente a los datos de entrenamiento.
Aunque el error de entrenamiento para el árbol es cero, su tasa de error en el conjunto de prueba es del 30%.
Tanto los humanos como los delfines fueron clasificados erróneamente como no mamíferos porque sus valores de atributo para Temperatura corporal, Da a luz y Cuatro patas son idénticos a los registros mal etiquetados en el conjunto de entrenamiento.

Los osos hormigueros espinosos (*spiny anteater*), por otro lado, representan un caso excepcional en el que la etiqueta de clase de un registro de prueba contradice las etiquetas de clase de otros registros similares en el conjunto de entrenamiento.
Los errores debidos a casos excepcionales suelen ser inevitables y establecen la tasa de error mínima que puede alcanzar cualquier clasificador.

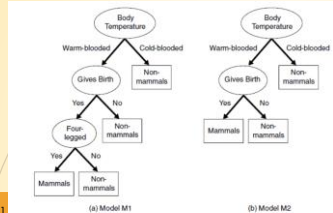

Figure 4.25. Decision tree induced from the data set shown in Table 4.3.

En la figura 4.25 (a) se muestra un árbol de decisiones que se ajusta perfectamente a los datos de entrenamiento. Aunque el error de entrenamiento para el árbol es cero, su tasa de error en el conjunto de prueba es del 30%.

En contraste, el árbol de decisión M2 que se muestra en la Figura 4.25 (b) tiene una tasa de error de prueba más baja (10%) aunque su tasa de error de entrenamiento es algo mayor (20%).

Es evidente que el primer árbol de decisión, M1, ha sobreajustado los datos de entrenamiento porque hay un modelo más simple con menor tasa de error en el conjunto de prueba.

La condición de prueba de atributo de cuatro patas en el modelo M1 es falsa porque se ajusta a los registros de entrenamiento mal etiquetados, lo que conduce a la clasificación errónea de los registros en el conjunto de prueba.

Sobreajuste debido a la falta de muestras representativas

Table 4.5. An example training set for classifying mammals.

Name	Body Temperature	Gives Birth	Four-legged	Hibernates	Class Label
salamander	cold-blooded	no	yes	yes	no
guppy	cold-blooded	yes	no	no	no
eagle	warm-blooded	no	no	no	no
poorwill	warm-blooded	no	no	yes	no
platypus	warm-blooded	no	yes	yes	yes

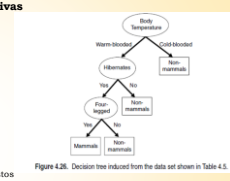

Figure 4.26. Decision tree induced from the data set shown in Table 4.5.

Los modelos que toman sus decisiones de clasificación basándose en una pequeña cantidad de registros de entrenamiento también son susceptibles de sobreajuste. Estos modelos se pueden generar debido a la falta de muestras representativas en los datos de entrenamiento y algoritmos de aprendizaje que continúan refinando sus modelos incluso cuando hay pocos registros de entrenamiento disponibles.

Este ejemplo demuestra claramente el peligro de realizar predicciones incorrectas cuando no hay suficientes ejemplos representativos en los nodos hoja de un árbol de decisión.

Todos estos registros de entrenamiento están etiquetados correctamente y el árbol de decisiones correspondiente. Aunque su error de entrenamiento es cero, su tasa de error en el conjunto de prueba es del 30%.

Referencias bibliográficas

Bhumika Gupta, Aditya Rawat, Akshay Jain, Arpit Arora, Naresh Dhami. (2017). Analysis of Various Decision Tree Algorithms for Classification in Data Mining. International Journal of Computer Applications (0975–8887). Volume 163 – No 8, April 2017.

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

Rokach, L. & Maimon, O. (2015). Data Mining with decision trees. Theory and Applications. Second Edition. World Scientific Publishing Co. Pte. Ltd.

Sancho Caparrini, Fernando (2009). Aprendizaje inductivo. Arboles de decisión. Portal Web. Disponible en: http://www.cs.us.es/~fsancho/?e=104

Tan Pang-Ning, Steinbach Michael, Kumar Vipin. (2014). Introduction to data mining. First Edition. Pearson New International Edition.