HỆ TRỢ GIÚP QUYẾT ĐỊNH

Tuần 7 (Bài 1) Hai V. Pham

Cây quyết định (Decision Tree)

- Cây quyết định là một kiểu mô hình dự báo
- Kỹ thuật học máy dùng trong cây quyết định được gọi là học bằng cây quyết định
- Phương tiện có tính mô tả dành cho việc tính toán các xác suất có điều kiện
- Sự kết hợp của các kỹ thuật toán học và tính toán nhằm hỗ trợ việc mô tả, phân loại và tổng quát hóa một tập dữ liệu cho trước

Khái niệm cây quyết định

- Cây quyết định là một cấu trúc phân cấp của các nút và các nhánh
  - 3 loại nút trên cây:
    - Nút gốc
    - Nút nội bộ: mang tên thuộc tính của CSDL
    - Nút lá: mang tên lớp C_i
  - Nhánh: mang giá trị có thể của thuộc tính
- Cây quyết định được sử dụng trong phân lớp bằng cách duyệt từ nút gốc của cây cho đến khi đụng đến nút lá.

Ví dụ

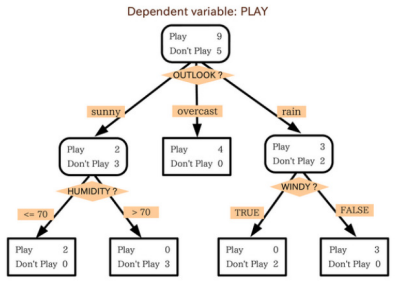
David là quản lý của một câu lạc bộ đánh golf nổi tiếng. Anh ta đang có rắc rối chuyện các thành viên đến hay không đến. Có ngày ai cũng muốn chơi golf nhưng số nhân viên câu lạc bộ lại không đủ phục vụ. Có hôm, không hiểu vì lý do gì mà chẳng ai đến chơi, và câu lạc bộ lại thừa nhân viên.

Mục tiêu của David là tối ưu hóa số nhân viên phục vụ mỗi ngày bằng cách dựa theo thông tin dự báo thời tiết để đoán xem khi nào người ta sẽ đến chơi golf. Để thực hiện điều đó, anh cần hiểu được tại sao khách hàng quyết định chơi và tìm hiểu xem có cách giải thích nào cho việc đó hay không.

Vậy là trong hai tuần, anh ta thu thập thông tin về: Trời (outlook) (nắng (sunny), nhiều mây (overcast) hoặc mưa (raining)). Nhiệt độ (temperature) bằng độ F. Độ ẩm (humidity). Có gió mạnh (wind) hay không.

Và tất nhiên là số người đến chơi golf vào hôm đó. David thu được một bộ dữ liệu gồm 14 dòng và 5 cột.

Ví dụ:



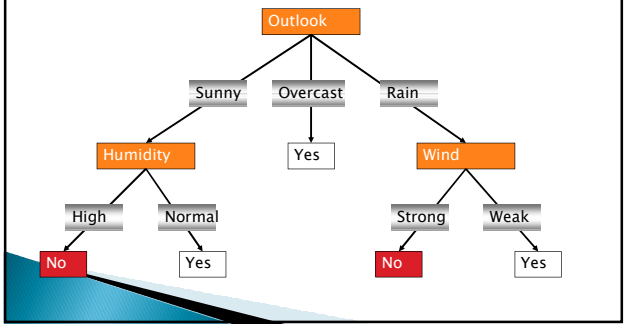
5

Ví dụ

Day	Outlook	Temp.	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	No
6	Rain	Cool	Normal	Strong	Yes
7	Overcast	Cool	Normal	Weak	No
8	Sunny	Mild	High	Weak	Yes
9	Sunny	Cold	Normal	Weak	Yes
10	Rain	Mild	Normal	Strong	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

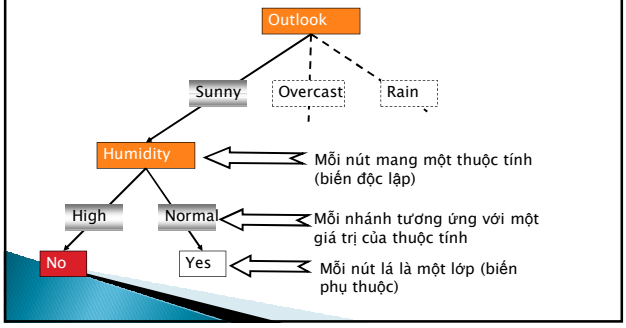
Ví dụ

Kiểm tra khi nào chơi golf, khi nào không chơi



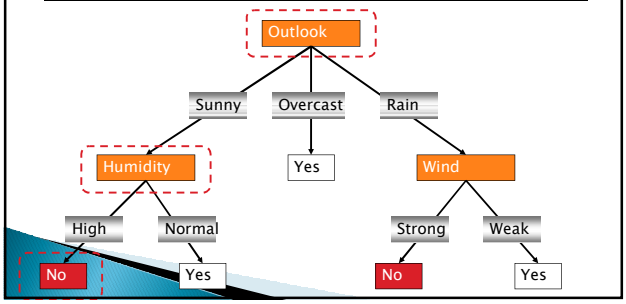
Ví dụ

Kiểm tra khi nào chơi golf, khi nào không chơi



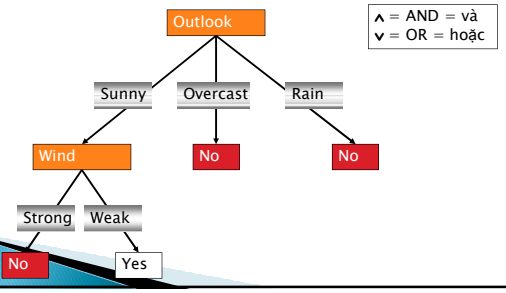
Duyệt cây quyết định

Day	Outlook	Temp.	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No



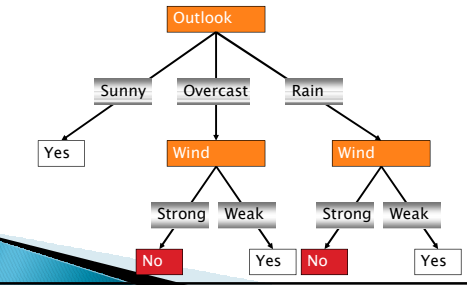
Biểu thức luận lý

Outlook=Sunny ∧ Wind=Weak



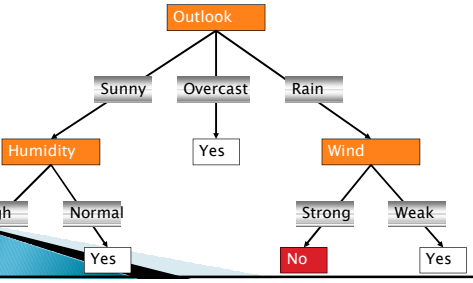
Biểu thức luận lý

Outlook=Sunny ∨ Wind=Weak



Biểu thức luận lý

(Outlook=Sunny ∧ Humidity=Normal)
∨ Outlook=Overcast
∨ (Outlook=Rain ∧ Wind=Weak)



Xây dựng cây quyết định

- Cây được thiết lập từ trên xuống dưới
- Rời rạc hóa các thuộc tính dạng phi số
- Các mẫu huấn luyện nằm ở gốc của cây
- Chọn một thuộc tính để phân chia thành các nhánh. Thuộc tính được chọn dựa trên độ đo thống kê hoặc độ đo heuristic
- Tiếp tục lặp lại việc xây dựng cây quyết định cho các nhánh

Xây dựng cây quyết định

- Điều kiện dừng
  - Tất cả các mẫu rơi vào một nút thuộc về cùng một lớp (nút lá)
  - Không còn thuộc tính nào có thể dùng để phân chia mẫu nữa
  - Không còn lại mẫu nào tại nút

Các công thức

Gini Impurity (sự hỗn tạp): $I_G(i) = 1 - \Sigma^m_{j=1} f(i,j)^2$, với $f(i,j)$ là tần suất giá trị j tại nút i, $I_G(i)$ đạt min (=0), nếu tất cả các trường hợp của nút đều chỉ nhận một giá trị

Information Gain (độ đo mang tin):

$I_E(i) = - \Sigma^m_{j=1} f(i,j) \log_2 f(i,j)$, entropy

Misclassification Measure (độ đo phân lớp sai):

$I_M(i) = 1 - \max_j f(i,j)$

15

Lựa chọn thuộc tính

- Độ đo để lựa chọn thuộc tính: Thuộc tính được chọn là thuộc tính có lợi nhất cho quá trình phân lớp (tạo ra cây nhỏ nhất)
- Có 2 độ đo thường dùng
  - 1. Độ lợi thông tin (Information gain)
    - Giả sử tất cả các thuộc tính dạng phi số
    - Có thể biến đổi để áp dụng cho thuộc tính số
  - 2. Chỉ số Gini (Gini index)
    - Giả sử tất cả các thuộc tính dạng số
    - Giả sử tồn tại một vài giá trị có thể phân chia giá trị của từng thuộc tính
    - Có thể biến đổi để áp dụng cho thuộc tính phi số

Độ lợi thông tin (Information gain)

- S: số lượng tập huấn luyện
- S_i: số các mẫu của S nằm trong lớp C_i với $i = \{1, \ldots, m\}$
- Thông tin cần biết để phân lớp một mẫu

$$I(s_1, s_2, \ldots, s_m) = -\sum_{i=1}^{m} \frac{s_i}{s} \log_2 \frac{s_i}{s}$$

Độ lợi thông tin

- Thuộc tính A có các giá trị $\{a_1, a_2, \ldots, a_n\}$
- Dùng thuộc tính A để phân chia tập huấn luyện thành n tập con $\{S_1, S_2, \ldots, S_n\}$
- S_{ij} : số mẫu của lớp C_i thuộc tập con S_j $(A=a_j)$
- Entropy của thuộc tính A:

$$E(A) = \sum_{j=1}^{n} \frac{s_{1j} + \ldots + s_{mj}}{s} I(s_{1j}, \ldots, s_{mj})$$

- Độ lợi thông tin dựa trên phân nhánh bằng thuộc tính A:

$$G(A) = I(s_1, s_2, \ldots, s_m) - E(A)$$

- Tại mỗi cấp, chúng ta chọn thuộc tính có độ lợi lớn nhất để phân nhánh cây hiện tại

Ví dụ

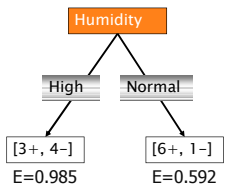
Day	Outlook	Temp.	Humidity	Wind	Play?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	No
6	Rain	Cool	Normal	Strong	Yes
7	Overcast	Cool	Normal	Weak	No
8	Sunny	Mild	High	Weak	Yes
9	Sunny	Cold	Normal	Weak	Yes
10	Rain	Mild	Normal	Strong	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Độ lợi thông tin, ví dụ

- Ta có
  - S = 14
  - m = 2
  - C_1 = "Yes", C_2 = "No"
  - S_1 = 9, S_2 = 5

$$I(S_1,S_2) = I(9,5) = -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.940$$

Độ lợi thông tin, ví dụ



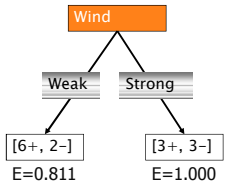
$$-\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7} = 0.985$$

$$-\frac{6}{7}\log_2\frac{6}{7} - \frac{1}{7}\log_2\frac{1}{7} = 0.592$$

Gain(S,Humidity)
=0.940 - (7/14)*0.985 – (7/14)*0.592
=0.151

Ghi chú: Để tính $\log_2 5$ bằng máy tính điện tử, nhấn: 5 log / 2 log =

Độ lợi thông tin, ví dụ

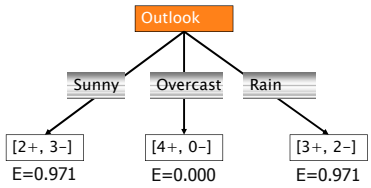


$$-\frac{6}{8}\log_2\frac{6}{8}-\frac{2}{8}\log_2\frac{2}{8}=0.811$$

$$-\frac{3}{6}\log_2\frac{3}{6}-\frac{3}{6}\log_2\frac{3}{6}=1.000$$

Gain(S,Wind)
=0.940 - (8/14)*0.811 – (6/14)*1.000
=0.048

Độ lợi thông tin, ví dụ



Gain(S,Wind)=0.048
Gain(S,Humidity)=0.151

Gain(S,Outlook)
=0.940 - (5/14)*0.971
- (4/14)*0.0 – (5/14)*0.0971
=0.247

Chỉ số Gini

- Chỉ số Gini của nút t:

$$\text{GINI}(t)=1-\sum_j p(j|t)^2$$

Trong đó $p(j|t)$ là tần suất của lớp j trong nút t
- Lớn nhất là $1-1/n_c$ khi các mẫu phân bố đều trên các lớp
- Thấp nhất là 0 khi các mẫu chỉ thuộc về một lớp

Ví dụ chỉ số Gini

$$\text{GINI}(t) = 1 - \sum_j p(j|t)^2$$

C1	0
C2	6

P(C1) = 0/6 = 0 P(C2) = 6/6 = 1
GINI = 1 – (P(C1)²+P(C2)²) = 1 – (0+1) = 0

C1	1
C2	5

P(C1) = 1/6 P(C2) = 5/6
GINI = 1 – (1/6)² – (5/6)² = 0.278

C1	2
C2	4

P(C1) = 2/6 P(C2) = 4/6
GINI = 1 – (2/6)² – (4/6)² = 0.444

Phân nhánh bằng chỉ số Gini

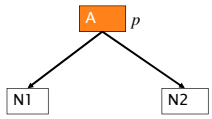
- Khi phân chia nút p thành k nhánh, chất lượng của phép chia được tính bằng:

$$\text{GINI}_{chia} = \sum_{i=1}^{k} \frac{n_i}{n} \text{GINI}(i)$$

trong đó
  - n_i là số mẫu trong nút i
  - n là số mẫu trong nút p
- Chọn thuộc tính có GINI_{chia} nhỏ nhất để phân nhánh

Phân nhánh thuộc tính nhị phân

- Chỉ phân thành 2 nhánh



	p
C1	6
C2	6

Gini=0.500

Gini(N1) =1–(5/6)²–(2/6)²
=0.194

Gini(N2) =1–(1/6)²–(4/6)²
=0.528

	N1	N2
C1	5	1
C2	2	4

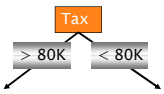
Gini=0.333

Gini_{chia} =7/12*0.194
+5/12*0.528
=0.333

Phân chia thuộc tính có giá trị liên tục

- Dựa trên một giá trị nếu muốn phân chia nhị phân
- Dựa trên vài giá trị nếu muốn có nhiều nhánh
- Với mỗi giá trị tính các mẫu thuộc một lớp theo dạng A<*v* và A>*v*
- Cách chọn giá trị *v* đơn giản: với mỗi giá trị *v* trong CSDL đều tính Gini của nó và lấy giá trị có Gini nhỏ nhất → kém hiệu quả

TID	Refund	Marital	Tax	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Phân chia thuộc tính có giá trị liên tục

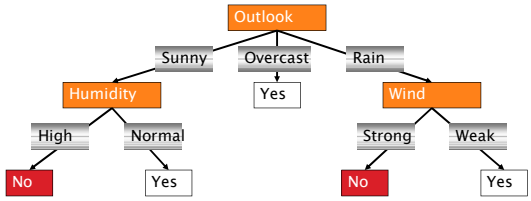
- Cách chọn giá trị *v* hiệu quả:
  - Sắp xếp các giá trị tăng dần
  - Chọn giá trị trung bình của từng giá trị của thuộc tính để phân chia và tính chỉ số gini
  - Chọn giá trị phân chia có chỉ số gini thấp nhất

Cheat	No	No	No	Yes	Yes	Yes	No	No	No	No
Taxable Income										
Sorted Values →	60	70	75	85	90	95	100	120	125	220
Split Positions →	55	65	72	80	87	92	97	110	122	172	230
	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >	<= >
Yes	0 3	0 3	0 3	0 3	1 2	2 1	3 0	3 0	3 0	3 0	3 0
No	0 7	1 6	2 5	3 4	3 4	3 4	3 4	4 3	5 2	6 1	7 0
Gini	0.420	0.400	0.375	0.343	0.417	0.400	0.300	0.343	0.375	0.400	0.420

Biến đổi cây quyết định thành luật

- Biểu diễn tri thức dưới dạng luật IF-THEN
- Mỗi luật tạo ra từ mỗi đường dẫn từ gốc đến lá
- Mỗi cặp giá trị thuộc tính dọc theo đường dẫn tạo nên phép kết (phép AND – và)
- Các nút lá mang tên của lớp

Biến đổi cây quyết định thành luật



R_1: If (Outlook=Sunny) ∧ (Humidity=High) Then Play=No
R_2: If (Outlook=Sunny) ∧ (Humidity=Normal) Then Play=Yes
R_3: If (Outlook=Overcast) Then Play=Yes
R_4: If (Outlook=Rain) ∧ (Wind=Strong) Then Play=No
R_5: If (Outlook=Rain) ∧ (Wind=Weak) Then Play=Yes

Ưu điểm của cây quyết định

- Cây quyết định dễ hiểu
- Việc chuẩn bị dữ liệu cho một cây quyết định là cơ bản hoặc không cần thiết
- Cây quyết định có thể xử lý cả dữ liệu có giá trị bằng số và dữ liệu có giá trị là tên thể loại
- Cây quyết định là một mô hình hộp trắng
- Có thể thẩm định một mô hình bằng các kiểm tra thống kê
- Cây quyết định có thể xử lý tốt một lượng dữ liệu lớn trong thời gian ngắn

Tóm tắt cây quyết định

- Chuyển thành luật
- Phân lớp, khai phá dữ liệu
- Tỉa cây (tỉa cây trước–cùng với dựng cây, tỉa cây sau, sai số tỉa cây) , khử nhiễu
- Bảng quyết định – Cây quyết định – Mạng quyết định (có thêm nút HOẶC)

33

Cài đặt ứng dụng cây quyết định

```
'Lớp xây dựng cây quyết định của thuật toán DecisionTree*/
Public Class DecisionTree
Private mSamples As DataTable
Private mTotalPositives As Integer = 0
Private mTotal As Integer = 0
Private mTargetAttribute As String = "RESULT
Public mTrueValue As String = "True"
Private mFalseValue As String = "False"
Private mEntropySet As Double = 0.0
```

34

Cài đặt ứng dụng cây quyết định…

```
Private Function countTotalPositives(ByVal samples As DataTable) As Integer
Dim result As Integer = 0
For Each aRow As DataRow In samples.Rows
Dim s As String = "True"
If Not (aRow(mTargetAttribute).ToString().Trim().ToUpper() = mTrueValue.ToUpper()) Then s = "False"
If Boolean.Parse(s) = True Then result = result + 1
Next
Return result
End Function
```

35

Cài đặt ứng dụng cây quyết định…

```
' Duyệt qua bảng và kiểm tra thuộc tính có giá trị là value và trả về số phần tử True và số phần tử âm. */
Private Sub getValuesToAttribute(ByVal samples As DataTable, ByVal attribute As Attribute, ByVal value As String, ByRef positives As Integer, ByRef negatives As Integer)
positives = 0
negatives = 0
For Each aRow As DataRow In samples.Rows
If CType(aRow(attribute.AttributeName), String) = value Then
Dim s As String = "True"
If Not (aRow(mTargetAttribute).ToString().Trim().ToUpper() = mTrueValue.ToUpper()) Then s = "False"
If Boolean.Parse(s) = True Then
positives = positives + 1
Else
negatives = negatives + 1
End If
End If
Next
End Sub
```

36

Cài đặt …

- **Thủ tục tính toán Entropy:** $Entropy(U) = -\sum_{i \in I} p_i \log_2 p_i$

```
'Tính entropy -p+log(p+,2) + p-log(p-,2)
Private Function calcEntropy(ByVal positives As Integer,
ByVal negatives As Integer) As Double
    Dim total As Integer = positives + negatives
    Dim ratioPositive As Double = CType(positives /
    Dim ratioNegative As Double = CType(negatives / total,
Double)
        ' Cây sẽ ngưng làm việc khi phát hiện
  root.Attribute.value chứa giá trị  null
    If total = 0 Then Return 0
    If Not (ratioPositive = 0) Then ratioPositive = -
  (ratioPositive) * System.Math.Log(ratioPositive, 2)
    If Not (ratioNegative = 0) Then ratioNegative = -
  (ratioNegative) * System.Math.Log(ratioNegative, 2)
    Dim result As Double = ratioPositive + ratioNegative
    Return result
    End Function
```

37

Cài đặt …

- **Thủ tục tính lượng thông tin IG**

$IG(U, c) = Entropy(U) - \sum_{u \in U_c} \frac{|U_c|}{|U|} Entropy(U_c)$

```
'Tính    lượng    thông    tin    thu    thêm    (IG):
Private Function gain(ByVal samples As DataTable, ByVal
attribute As Attribute) As Double
    Dim values() As String = attribute.values
    Dim sum As Double = 0.0
    Dim _len As Integer = values.Length - 1
    For i As Integer = 0 To _len
    Dim positives, negatives As Integer
    positives = negatives = 0
    getValuesToAttribute(samples, attribute, values(i),
positives, negatives)
    Dim entropy As Double = calcEntropy(positives, negatives)
    sum += -CType((positives + negatives) / mTotal * entropy,
Double)
    Next i
    Return mEntropySet + sum
    End Function
```

38

Decision Tree (Software Demo)

- http://www.montefiore.ulg.ac.be/~geurts/dtapplet/dtexplication.html
- http://webdocs.cs.ualberta.ca/~aixplore/learning/DecisionTrees/Applet/DecisionTreeApplet.html

39