

# Cơ sở trí tuệ nhân tạo

## Bài tập về học máy

### 4. MÁY HỌC

1/

ID3 cần tính entropy trung bình của các thuộc tính tại mỗi nút để lựa chọn thuộc tính thích hợp nhất điền vào nút đó trên decision tree.

Đầu tiên ta liệt kê số lượng các output khác nhau trên từng thuộc tính để tính entropy

Thuộc tính	Giá trị thuộc tính	Số lượng output (yes:no)	Entropy trung bình
Bố hoặc mẹ đỗ đại học	Có	4:2	0.6887
	Không	0:2	
Điều kiện kinh tế	Cao	0:1	0.4056
	Thấp	1:3	
	Trung bình	3:0	
Học lực	Giỏi	2:0	0.3444
	Khá	2:1	
	Trung bình	0:3	
Có học thêm	Có	3:2	0.9512
	Không	1:2	

Học lực có entropy thấp nhất nên ta chọn học lực làm node gốc của decision tree

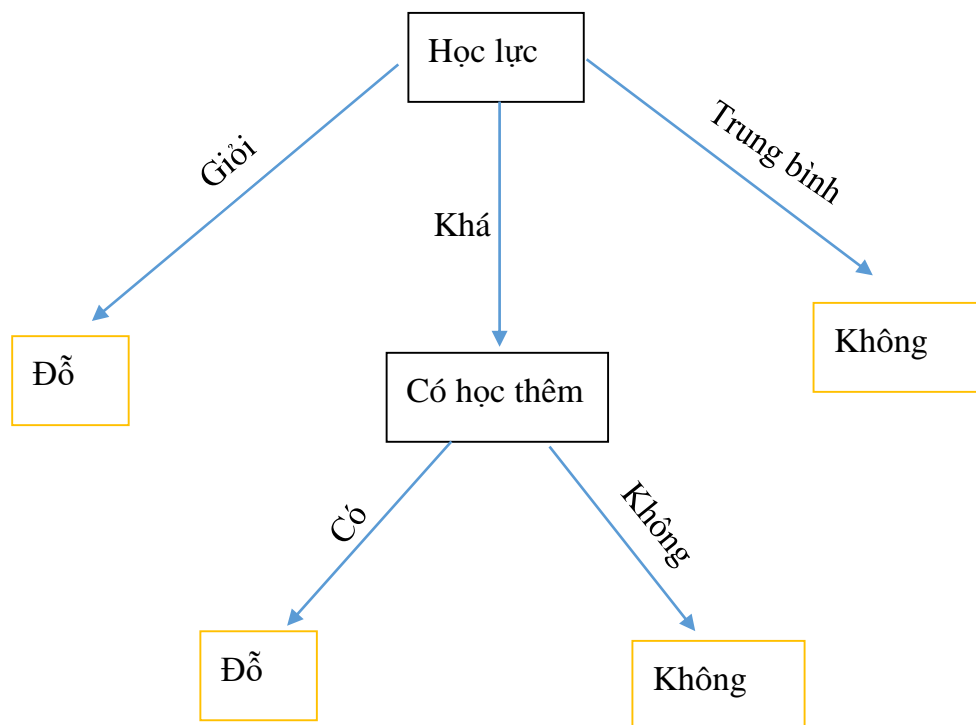
Hai nhánh Giỏi và Trung bình đều có sự phân bố tuyệt đối của output nên ta chỉ cần phân nhánh thêm ở nhánh khá. Tương tự như bảng trên ta lập bảng với các output nằm trên Học lực Khá

Thuộc tính	Giá trị thuộc tính	Số lượng output (yes:no)	Entropy trung bình
Bố hoặc mẹ đỗ đại học	Có	2:0	0.0
	Không	0:1	

Điều kiện kinh tế	Cao	0:0	0.0
	Thấp	0:1	
	Trung bình	2:0	
Có học thêm	Có	2:0	0.0
	Không	0:1	

Các thuộc tính đều có entropy trung bình bằng 0 nên ta có thể chọn bất kỳ thuộc tính nào để phân loại. Ở đây em chọn Có học thêm làm node kế tiếp

Từ đó ta có decision tree như sau



Từ đó ta có thể rút ra các quy luật

IF	Học lực = Giỏi	THEN	Kết quả = Đỗ
ELIF	Học lực = Trung bình	THEN	Kết quả = Không
ELIF	(Học lực = Khá) $\cap$ (Có học thêm = Có)	THEN	Kết quả = Đỗ
ELIF	(Học lực = Khá) $\cap$ (Có học thêm = Không)	THEN	Kết quả = Không
ELSE		THEN	Failure

2/

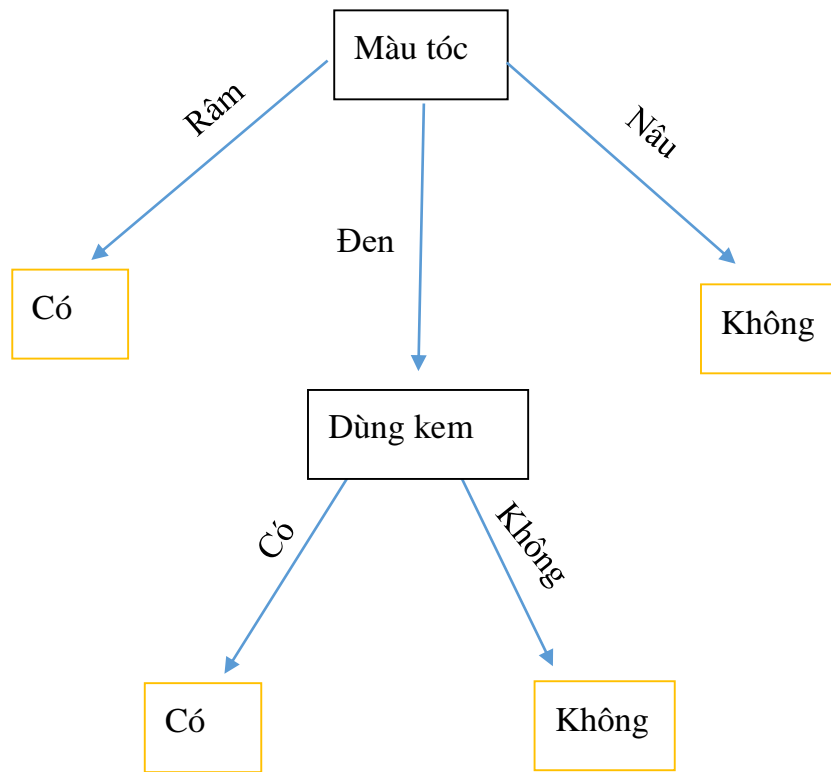
Tương tự câu 1. Trước tiên ta tính entropy của từng thuộc tính dựa vào bảng tổng các output trên từng giá trị thuộc tính

Thuộc tính	Giá trị thuộc tính	Số lượng output (yes:no)	Entropy trung bình
Màu tóc	Đen	1:2	0.3444
	Râm	0:3	
	Nâu	2:0	
Chiều cao	Cao	0:1	0.3444
	Trung bình	2:1	
	Thấp	0:4	
Cân nặng	Nặng	1:3	0.75
	Trung bình	1:2	
	Nhẹ	1:0	
Dùng kem	Có	0:4	0.4056
	Không	3:1	

Từ đây ta chọn Màu tóc làm thuộc tính cho node gốc. Khi đó 2 nhánh Râm và Nâu đã thuần nhất một lớp output nên tiếp theo ta chỉ cần phân nhánh trên giá trị Đen

Thuộc tính	Giá trị thuộc tính	Số lượng output (yes:no)	Entropy trung bình
Chiều cao	Cao	0:1	0.0
	Trung bình	1:0	
	Thấp	0:1	
Cân nặng	Nặng	0:0	0.6667
	Trung bình	0:1	
	Nhẹ	1:1	
Dùng kem	Có	0:2	0.0
	Không	1:0	

Từ đây em chọn Dùng kem làm thuộc tính tiếp theo (có thể chọn chiều cao vì có cùng entropy) thì được decision tree như sau:



Các quy luật tương ứng

IF	Màu tóc = Râm	THEN Kết quả = Có
ELIF	Màu tóc = Nâu	THEN Kết quả = Không
ELIF	(Màu tóc = Đen) $\cap$ (Dùng kem = Có)	THEN Kết quả = Có
ELIF	(Màu tóc = Đen) $\cap$ (Dùng kem = Không)	THEN Kết quả = Không
ELSE		THEN Failure

3/

Tương tự, trước tiên ta lập bảng và tính entropy các thuộc tính ở node gốc

Thuộc tính	Giá trị thuộc tính	Số lượng output (Âu:Á)	Entropy trung bình
Dáng	Nhỏ	4:1	0.7956
	To	2:1	
Cao	TB	2:2	0.5

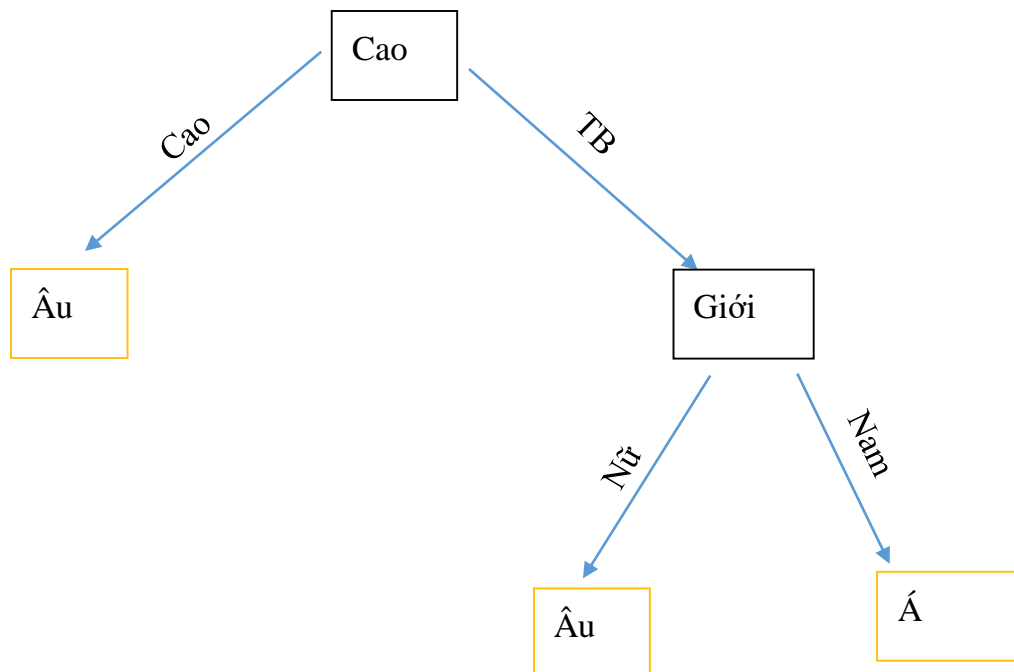
	Cao	4:0	
Giới	Nam	3:2	0.6068
	Nữ	3:0	

Từ đây ta chọn Cao làm thuộc tính cho node gốc và vì trên nhánh Cao chỉ có toàn kết quả Âu nên ta chỉ tiếp tục phân nhánh trên giá trị TB

Thuộc tính	Giá trị thuộc tính	Số lượng output (Âu:Á)	Entropy trung bình
Dáng	Nhỏ	1:1	1.0
	To	1:1	
Giới	Nam	0:2	0.0
	Nữ	2:0	

Chọn Giới làm node tiếp theo, trên các nhánh của Giới đã được phân nhánh tuyệt đối nên thuật toán kết thúc.

Ta có decision tree sau:



Các quy luật rút ra

IF	Cao = Cao	THEN	Châu = Âu
ELIF	(Cao = TB) $\cap$ (Giới = Nữ)	THEN	Châu = Âu
ELIF	(Cao = TB) $\cap$ (Giới = Nam)	THEN	Châu = Á

ELSE

THEN Failure

4/

Tương tự, trước tiên ta lập bảng và tính entropy các thuộc tính ở node gốc

Thuộc tính	Giá trị thuộc tính	Số lượng output (Có:Không)	Entropy trung bình
Quang cảnh	Mưa	0:2	0.3606
	Mây	3:0	
	Nắng	1:3	
Nhiệt độ	Nóng	1:4	0.6233
	Ấm	1:1	
	Mát	2:0	
Gió	Nhẹ	3:2	0.9000
	Mạnh	1:3	

Từ đây ta chọn thuộc tính có entropy nhỏ nhất là Quang cảnh làm node gốc

Tiếp theo ta chỉ cần phân nhánh trên giá trị Nắng (nhánh duy nhất có entropy khác 0).

Thuộc tính	Giá trị thuộc tính	Số lượng output (Có:Không)	Entropy trung bình
Nhiệt độ	Nóng	0:2	0.0
	Ấm	0:1	
	Mát	1:0	
Gió	Nhẹ	1:1	0.5
	Mạnh	0:2	

Chọn Nhiệt độ làm node kế tiếp, ta được decision tree sau:



4	Màu = Đỏ	0	2
5	Màu = Vàng	1	3
6	Màu = Tím	1	2
7	Vỏ = Nhẵn	0	4
8	Vỏ = Có gai	2	3

Ở description thứ 7 có số lượng Không lớn nhất và số lượng Có bằng 0 → Ta có rule sau:

IF Vỏ = Nhẵn THEN Độc tính = Không

Các example 1,2,6,7 được covered bởi rule trên, từ các example còn lại ta dựng tiếp description table như sau:

Thứ tự	Description	Có (độc tính)	Không (độc tính)
1	Vị = Ngọt	0	1
2	Vị = Cay	2	1
3	Vị = Chua	0	1
4	Màu = Vàng	1	2
5	Màu = Tím	1	1
6	Vỏ = Có gai	2	3

Description số 1 và 3 có cùng số example Không và không có example nào rơi vào class Có, ta có thể chọn 1 trong 2 description này để viết rule, ở đây em chọn description 1.

IF Vị = Ngọt THEN Độc tính = Không

Rule trên làm giảm example của Tím và Có gai đi 1. Làm description 5 có số example Không bằng 0, ta lại có rule sau

IF Màu = Tím THEN Độc tính = Có

Cứ tiếp tục ta viết thêm được 1 rule nữa:

IF Vị = Chua THEN Độc tính = Không

Đến đây chỉ còn 2 example chưa được cover là:

#	Vị	Màu	Vỏ	Độc tính
---	----	-----	----	----------



4	Cay	Vàng	Có gai	Có
9	Cay	Vàng	Có gai	Không

2 example này giống hệt nhau nhưng lại đều có mặt trong cả 2 class với số lượng như nhau nên không giúp ích gì cho việc phân loại.

Kết luận ta có luật suy diễn như sau:

IF Vỏ = Nhẵn THEN Độc tính = Không

ELSE IF Vị = Ngọt THEN Độc tính = Không

ELSE IF Màu = Tím THEN Độc tính = Có

ELSE IF Vị = Chua THEN Độc tính = Không

ELSE Failure