**PHẦN 1: CÂY QUYẾT ĐỊNH**

1. **Xây dựng cây quyết định dựa vào bảng dữ liệu trong bảng trên bằng cách tính độ hỗn loạn trung bình cho từng nút.**

**CT: *Entropy = - p.log 2 p – (1- p).log 2 (1-p)***

**P**: là tỷ lệ các mẫu dương trong tập huấn luyện.

***Độ hỗn loạn trung bình của SNOW:***

**H (sticky)** = - (2/3) log2 (2/3) - (1/3) log2 (1/3) = **0.918**

***H ( fresh )=*** - (7/8) log2 (7/8) - (1/8) log2 (1/8) = **0.543**

***H ( frosted) =*** - (0/3) log2 (0/3) - (3/3) log2 (3/3) = **0**

***Độ hỗn loạn trung bình (AE) snow:***

**🡺AE(Snow)** = p(sticky) \* H (sticky) + p(fresh) \* *H ( fresh )* + p(frosted) \* *H ( frosted)* = (3/14)\*0,918 + (8/14)\*0,543 + (3/14)\*0 = **0,507**

**Độ hỗn loạn trung bình của *WEATHER***

**H (Foggy)** = - (3/5) log2 (3/5) - (2/5) log2 (2/5) = **0,970**

**H( Sunny)** = - (5/6) log2 (5/6) - (1/6) log2 (1/6) = **0,650**

**H(Windy)** = - (2/3) log2 (2/3) - (1/3) log2 (1/3***)***  = **0,918**

***Độ hỗn loạn trung bình (AE) weather:***

**🡺AE( Weather )** = p(foggy) \* H(Foggy) + p(sunny) \* H( Sunny ) + p(windy) \* H (Windy)

= (5/14)x0,970 + (6/14)x0,650+ (3/14)x0,918 = **0,821**

**Độ hỗn loạn trung bình của *SEASON***

**H( Low)** = -p.log2(p) – (1-p).log2(1-p) = **0,918** (với p =6/9)

**H( Mid)** = -p.log2(p) – (1-p).log2(1-p) = **0,918** (với p =2/3)

**H( High)** = -p.log2(p) – (1-p).log2(1-p) = **1** (với p =1/2)

***Độ hỗn loạn trung bình (AE) season:***

**🡺AE( Season)** = p(low) \* H( Low ) + p(mid) \* H (Mid) + p(high) \* H ( High )

= (9/14)x0,918 + (3/14)x0,918+ (2/14)x1= **0,929**

**Độ hỗn loạn trung bình của *Physical condition***

H (Rested) = -p.log2(p) – (1-p).log2(1-p) = **0,684** (với **p =9/11**)

H (Injured) = -p.log2(p) – (1-p).log2(1-p) = **0** (với **p =0/2**)

H (Tired) = -p.log2(p) – (1-p).log2(1-p) = **0** (với **p =0/1**)

**Độ hỗn loạn trung bình (AE) Physical condition**

**🡺AE( Physical Condition**) = p(rested) \* H (Rested) + p(injured) \* H(Injured) + p(tired) \* H (Tired )

= (11/14) \* 0,684 + (2/14) \* 0 + (1/14) \* 0= **0,537**

***Vì Độ hỗn loạn trung bình của snow nhỏ nhất nên ta chọn snow là thuộc tính quyết định:***

* ***Ta được bảng sau:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Snow** | **Weather** | **Season** | **Physical Condition** | **Go skiing** |
| **frosted** | Windy | High | Tired | **No** |
| Foggy | Mid | Rested | **No** |
| Foggy | Low | Injured | **No** |
| **sticky** | Foggy | Low | Rested | No |
| Sunny | Low | Rested | Yes |
| Sunny | Mid | Rested | Yes |
| **fresh** | Sunny | Low | Injured | No |
| Sunny | Low | Rested | Yes |
| Sunny | High | Rested | Yes |
| Sunny | Mid | Rested | Yes |
| Windy | Low | Rested | Yes |
| Windy | Low | Rested | Yes |
| Foggy | Low | Rested | Yes |
| Foggy | Low | Rested | Yes |

**SNOW**

sticky

**!!!!!**

**!!!!!**

**NO**

Frosted

Fresh

Sau khi chọn snow là thuộc tính quyết định , thì ta tiếp tục chọn thuộc tính tiếp theo, tính độ hỗn loạn trung bình ở từng nhánh: sticky, fresh.

***Tính độ hỗn loạn trung bình AE weather(Sticky) :***

**H (Sticky, Foggy)** = -p.log2(p) – (1-p).log2(1-p) = 0 (**với p =0/1**)

**H (Sticky, Sunny)** = -p.log2(p) – (1-p).log2(1-p) = 0 (**với p =2/2**)

**🡺AE ( Weather(Sticky))** = p(foggy) \* H (Sticky, Foggy) + p(sunny) \* H (Sticky, Sunny)

= (1/3) \* 0 + (2/3) \* 0 = **0**

***Tính độ hỗn loạn trung bình AE Season(Stick):***

**H(Sticky, Low)** = -p.log2(p) – (1-p).log2(1-p) = 1 (**với p =1/2**)

**H (Sticky, Mid)** = -p.log2(p) – (1-p).log2(1-p) = 0 (**với p =1/1**)

**🡺AE (Season(Sticky) )** = p(low) \* H (Sticky, Low) + p(mid) \* H(Sticky, Mid)

= (2/3) \* 1 + (1/3) \* 0 = **0,666**

***Tính độ hỗn loạn trung bình AE Physical condition (stick)***

**H (Sticky, Rested**) = -p.log2(p) – (1-p).log2(1-p) = **0,918** (**với p =2/3**)

**🡺AE ( Physical Condition(Sticky) )** = p(rested) \* H (Sticky, rested)

= (2/3) \* 1 + (3/3) \* 0,918 = **0,918**

***Vì Độ hỗn loạn (AE) của sticky-weather nhỏ nhất nên sẽ chọn thuôc tính weather làm nút tiếp theo ở nhánh sticky:***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **sticky** | Foggy | Low | Rested | No |
| Sunny | Low | Rested | Yes |
| Mid | Rested | Yes |

***Trường hợp nhánh fresh:***

**Tính Độ hỗn loạn trung bình AE của (Fresh) :**

**H (Fresh,Sunny)** = -p.log2(p) – (1-p).log2(1-p) = **0,811** (với **p =3/4**)

**H (Fresh,Windy)** = -p.log2(p) – (1-p).log2(1-p) = **0** (với **p =2/2**)

**H (Fresh,Foggy)** = -p.log2(p) – (1-p).log2(1-p) = **0** (với **p =2/2**)

**🡺AE ( Weather(Fresh) )** = p(sunny) \* H(Fresh,Sunny) + p(windy) \* H (Fresh,Windy) + p(foggy) \* H (Fresh,Foggy)

= (4/8)\*0,811 +(2/8)\*0+(2/8)\*0 = **0,405**

H(Fresh,Low) = -p.log2(p) – (1-p).log2(1-p) = 0,650 (với p =5/6)

H(Fresh,Mid) = -p.log2(p) – (1-p).log2(1-p) = 0 (với p =1/1)

H(Fresh,High) = -p.log2(p) – (1-p).log2(1-p) = 0 (với p =1/1)

🡺AE ( Season(Fresh)) = p(low)\*H(Fresh,Low) + p(mid)\*H (Fresh,Mid) + p(high) \* H (Fresh,High)

= (6/8)\*0,650 +(1/8)\*0+(1/8)\*0 = **0,487**

H(Fresh,Injured) = -p.log2(p) – (1-p).log2(1-p) = 0 (với p =0/1)

H(Fresh,Rested) = -p.log2(p) – (1-p).log2(1-p) = 0 (với p =7/7)

🡺AE (Physical Condition(Fresh) ) = p(injured)\*H (Fresh,Injured) + p(rested)\* H (Fresh,Rested)

= (1/8)x0+(7/8)x0 = **0**

**🡺 *Vì Độ hỗn loạn (AE) của fresh-* Physical Condition nhỏ nhất nên sẽ chọn thuộc tính Physical Condition là nút tiếp theo của Fresh**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **fresh** | Sunny | Low | **Injured** | No |
| Sunny | Low | **Rested** | Yes |
| Sunny | High | Yes |
| Sunny | Mid | Yes |
| Windy | Low | Yes |
| Windy | Low | Yes |
| Foggy | Low | Yes |
| Foggy | Low | Yes |

***Cây Quyết Định***

**SNOW**

**NO**

**Physical**

**Weather**

sticky

Frosted

Fresh

**YES**

**NO**

sunny

foggy

**Yes**

**No**

Rested

Injured

**PHẦN 2: THỰC HÀNH VỚI WEKA**

1. **Nội dung file \*.afrr:**

**@relation** TruotTuyet

**@attribute** Snow {Sticky,Fresh,Frosted}

**@attribute** Weather {Foggy,Sunny,Windy}

**@attribute** Seasion {Low,Mid,High}

**@attribute** Physical\_Condition {Rested,Injured,Tired}

**@attribute** Go\_skiing {Yes,No}

**@data**

Sticky,Foggy,Low,Rested,No

Fresh,Sunny,Low,Injured,No

Fresh,Sunny,Low,Rested,Yes

Fresh,Sunny,High,Rested,Yes

Fresh,Sunny,Mid,Rested,Yes

Frosted,Windy,High,Tired,No

Sticky,Sunny,Low,Rested,Yes

Frosted,Foggy,Mid,Rested,No

Fresh,Windy,Low,Rested,Yes

Fresh,Windy,Low,Rested,Yes

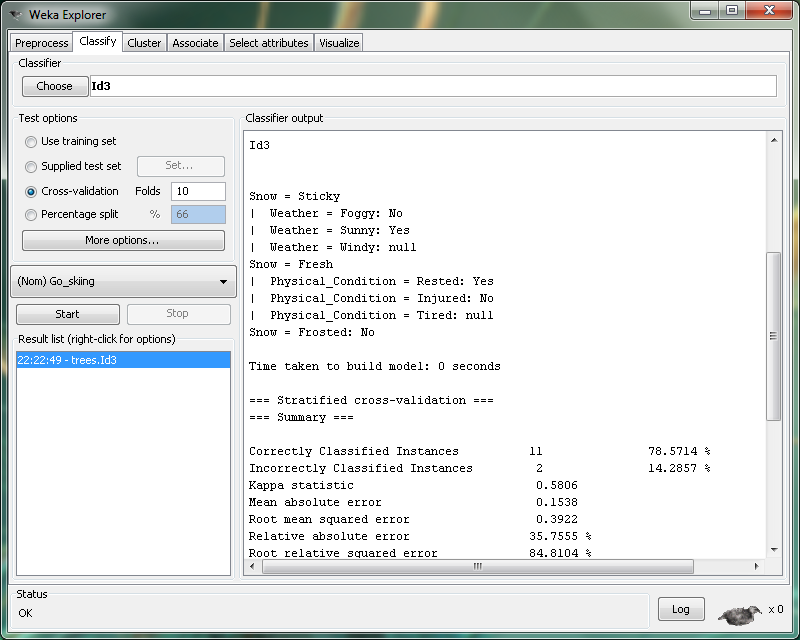
Fresh,Foggy,Low,Rested,Yes

Fresh,Foggy,Low,Rested,Yes

Sticky,Sunny,Mid,Rested,Yes

Frosted,Foggy,Low,Injured,No

1. **Thực nghiệm cây quyết định: chọn thuật toán ID3, vẽ lại cây quyết định**

****

**Vẽ lại cây quyết định: *Id3***

Cây bắt đầu từ snow và có 3 nhánh

***Snow = Sticky***//là nhánh 1

*| Weather = Foggy: No //*khi weather = foggy thì không đi trượt tuyết

*| Weather = Sunny: Yes //*khi weather = sunny thì đi trượt tuyết

*| Weather = Windy: null* // không có nhánh weather = windy

***Snow = Fresh*** // là nhánh thứ 2 trong cây

*| Physical\_Condition = Rested: Yes //*khi Physical\_Condition = Rested thì đi trượt tuyết

*| Physical\_Condition = Injured: No* // khi Physical\_ Condition = Injured thì không đi trượt tuyết.

*| Physical\_Condition = Tired: null //*không có nhánh Physical\_Condition. = Tired

***Snow = Frosted: No*** //là nhánh thứ 3. khi snow = frosted thì không đi trượt tuyết.

**căn cứ vào thông tin từ thuật toán ID3 ta có cây sau:**

**No**

**No**

**Yes**

**Yes**

**No**

Sticky

Frosted

Fresh

Foggy

Sunny

Rested

Injured

**Null**

**Null**

Windy

Tired

**các bộ phân lớp do weka xây dựng có giống kết quả tính được từ lý thuyết không? Nếu không tại sao?**

Trong thuật toán ID3 thì các bộ phận phân lớp trong weka đưa ra kết luận đúng với thực tế bên ngoài chẳng hạn ta có kết quả sau:

Cây quyết định :

*=== Classifier model (full training set) ===*

*Id3*

*snow = sticky*

*| weather = sunny: yes*

*| weather = windy: null*

*| weather = foggy: no*

*snow = fresh*

*| physical\_condition = rested: yes*

*| physical\_condition = tired: null*

*| physical\_condition = injured: no*

*snow = frosted: no*

đưa ra kết luận đi hay không đi:

***=== Confusion Matrix ===***

***a b <-- classified as***

*9 0 | a = yes*

*2 2 | b = no*

***Thực nghiệm, bằng cách chọn kiểu test là cross-validation với số fols mặc định là 10, hãy cho biết tỉ lệ chính xác của bộ phân lớp, số mẫu phân lớp đúng/ sai/ không phân* *lớp được. diễn giải lại phân bố dữ liệu thông tin từ confusion matrix.***

**ĐỐI VỚI THUẬT TOÁN ID3**

=== Stratified cross-validation ===

=== Summary ===

Correctly Classified Instances 11 78.5714 % // là tỷ lệ % độ chính xác của Phân lớp

Incorrectly Classified Instances 2 14.2857 % // là tỷ lệ % độ không chính xác của Phân lớp

Kappa statistic 0.5806

Mean absolute error 0.1538 // là tỷ lệ % lội tuyệt đối của Phân lớp

Root mean squared error 0.3922

Relative absolute error 35.7555 % //*Lỗi tương đối tuyệt đối*

Root relative squared error 84.8104 %

UnClassified Instances 1 7.1429 % // là tỷ lệ % không Phân lớp được

Total Number of Instances 14

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure ROC Area Class

1 0.5 0.818 1 0.9 0.8 Yes

0.5 0 1 0.5 0.667 0.7 No

Weighted Avg. 0.846 0.346 0.874 0.846 0.828 0.769

=== Confusion Matrix ===

a b <-- classified as

9 0 | a = Yes

2 2 | b = No

//ý nghĩa: tổng cộng có 13 trường hợp phân lớp được trong đó có 11 phân lớp đúng và 2 phân lớp sai. Trong 11 trường hợp phân lớp đúng thì có 9 trường là đi trượt tuyết 2 trường hợp là không đi. Trong 2 trường hợp sai, bên ngoài là đi trượt tuyết mà Weka trả lời là không đi.