Самостійна робота 1 з дисципліни **Data mining**

Студент - Сирота Сергій ТТП-42 (Serhii Syrota TTP-42)

Викладач - Криволап Андрій (PhD) (Kryvolap Andriy)

Task description:

```
Для одного з варіантів побудувати класифікатор використовуючи методи 1-Rule, Naive-Bayes, Decission Tree, kNN. Можливий варіант програмної реалізації з докладними поясненнями.
```

Task A

Q1	Q2	Q3	Q4	S
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	1	1	?

Rule 1

True/all - table:

Q1	Q2	Q3	Q4
3/10	3/10	8/10	5/10

Q3 - winner, S(10) = 1

Naive-Bayes

Q1	Q2	Q3	Q4	S
0	0	0	0	1

0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	1	1	?

P(S=1) = 0.5; P(S=0) = 0.5; Вважаємо, що усі атрибути(Q(1-5) features) незалежні один від одного та рівні між собою за вагою(значимістю).

Множина класів: {S1, S0}

```
P(f) = P(f(d1),f(d2),f(d3),f(d4),f(d5))

P(c|f) = ( P(f | c) * P(c) )/P(f)

f = (Q1,Q2,Q3,Q4)
c = 0 | 1

Independency between features => P(Q1, Q2) = P(Q1) * P(Q2)

P(c|f) = ( P(c) * P(Q1 | c) * P(Q2 | c) * P(Q3 | c) * P(Q4 | c)) / (P(Q1) * P(Q2) *
P(Q3) * P(Q4))

As the denominator remains the constant - we can remove that term
P(c|f) ~=  P(c) * P(Q1 | c) * P(Q2 | c) * P(Q3 | c) * P(Q4 | c)
```

Q1	S(0)	S(1)	P(S(0))	P(S(1))
0	3	5	3/8	5/8
1	0	2	0	1

Q2	S(0)	S(1)	P(S(0))	P(S(1))
0	3	3	3/6	3/6
1	2	2	2/4	2/4

Q3	S(0)	S(1)	P(S(0))	P(S(1))

0	4	1	4/5	1/5
1	1	4	1/5	4/5

Q4	S(0)	S(1)	P(S(0))	P(S(1))
0	3	3	3/6	3/6
1	2	2	2/4	2/4

Finally

Q1	Q2	Q3	Q4	S
1	1	1	1	?

```
P(S0 | (1,1,1,1)) = 0 * 0.5 * 0.2 * 0.5 = 0.05*0

P(S1 | (1,1,1,1)) = 1 * 0.5 * 0.8 * 0.5 = 0.2
```

Even without k-additive smoothing S1 probability is obviously higher

S(10) = 1

Decision Tree

Q1	Q2	Q3	Q4	S
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	1	1	?

Lets use CART algorithm and Gini impurity to split the dataset into a decision tree.

Stopping criteria: to utilize a minimum amount of the training data allocated to every leaf node. If the count is smaller than the specified threshold, the split is rejected and also the node is considered the last leaf

node.

Stop splitting count: 3.

Original Gini impurity = 0.5

```
Gini = 1 - Sum((p(i)^2), i=1 to C)
```

Gini Impurity for each feature:

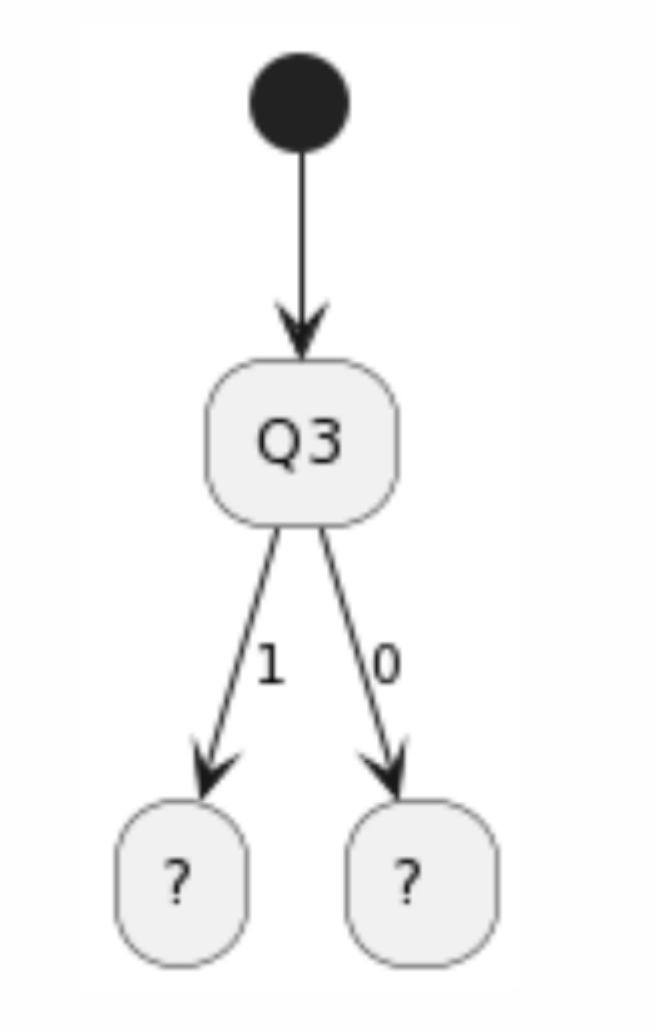
Q1	S(0)	S(1)	GI(Q1 = 0)	GI(Q1 = 1)	AVG GI	Gini Gain
0	3	5	1 - (3/8)^2 - (5/8)^2 = 0.46875	1 - (0/2)^2 - (2/2)^2 = 0	(8/10)*(0.46875) + 0 = 0.375	0.125
1	0	2				

Q2	S(0)	S(1)	GI(Q2 = 0)	GI(Q2 = 1)	AVG GI	Gini Gain
0	3	3	1 - (3/6)^2 - (3/6)^2 = 0.5	1 - (2/4)^2 - (2/4)^2 = 0.5	(6/10 + 4/10)*0.5 = 0.5	0
1	2	2				

Q3	S(0)	S(1)	GI(Q3 = 0)	GI(Q3 = 1)	AVG GI	Gini Gain
0	4	1	1 - (4/5)^2 - (1/5)^2 = 0.32	1 - (1/5)^2 - (4/5)^2 = 0.32	0.32	0.18
1	1	4				

Q4	S(0)	S(1)	GI(Q4 = 0)	GI(Q4 = 1)	AVG GI	Gini Gain
0	3	3	1 - (3/6)^2 - (3/6)^2 = 0.5	1 - (2/4)^2 - (2/4)^2 = 0.5	(6/10 + 4/10)*0.5 = 0.5	0
1	2	2				

By maximizing the Gini Gain, we've get the first node - Q3



To find other leafs, let's recalculate Gini Gain for Q3 = 0 and Q3 = 1 for the other features.

Q1	Q3	S(0)	S(1)
0	0	3	1
1	1	1	0
0	1	0	4
1	0	1	0

`GI(Q3 = 0 && Q1 = 0) = 1 - (3/4)^2 - (1/4)^2 = 0.375`

`GI(Q3 = 0 && Q1 = 1) = 1 - 1 = 0`

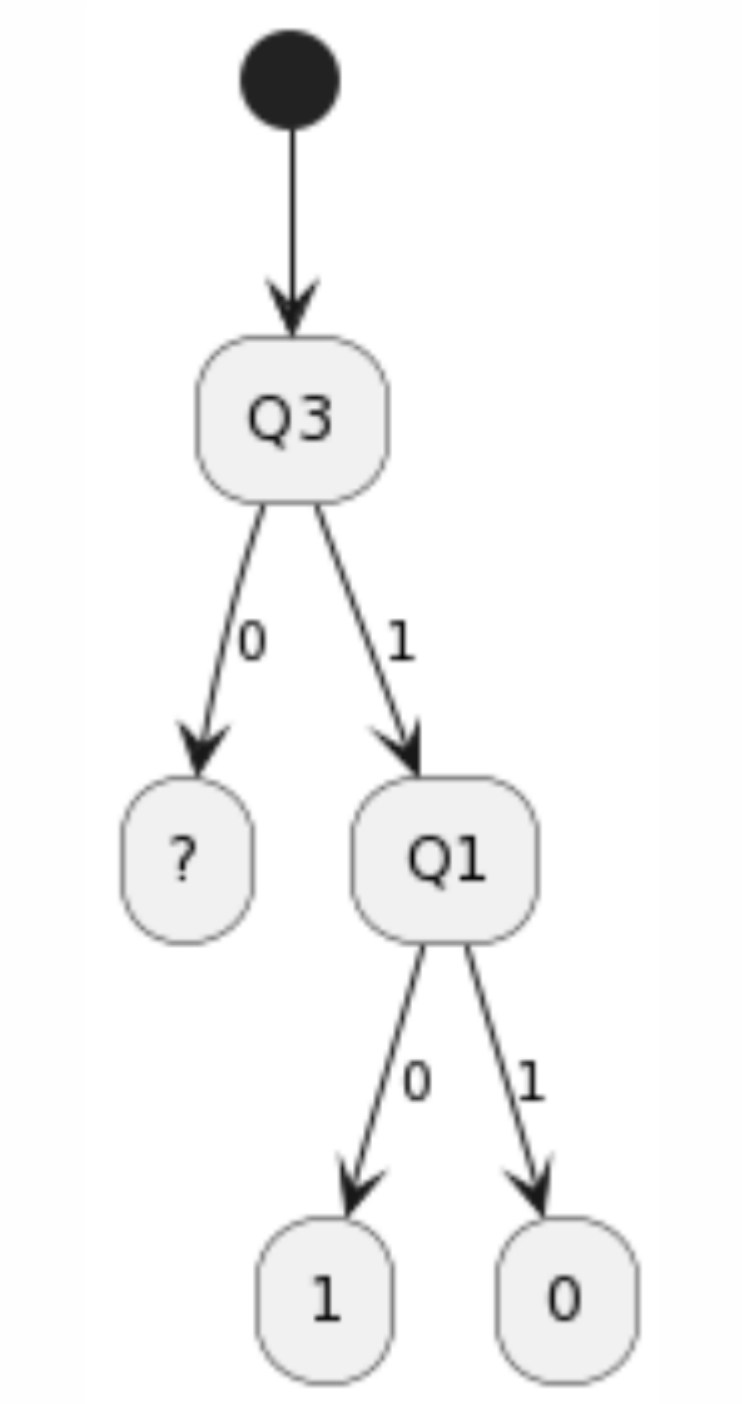
`GI(Q3 = 1 && Q1 = 0) = 1 - 1 = 0`

`GI(Q3 = 1 && Q1 = 1) = 1 - 1 = 0`

`AVG GI(Q3=0 && Q1)= 4/5 x 0.375 + 0 = 0.3`

`AVG GI(Q3=1 && Q1)= 0`

Let's stop splitting Q1=1 add the new node (Q3=1 && Q1)



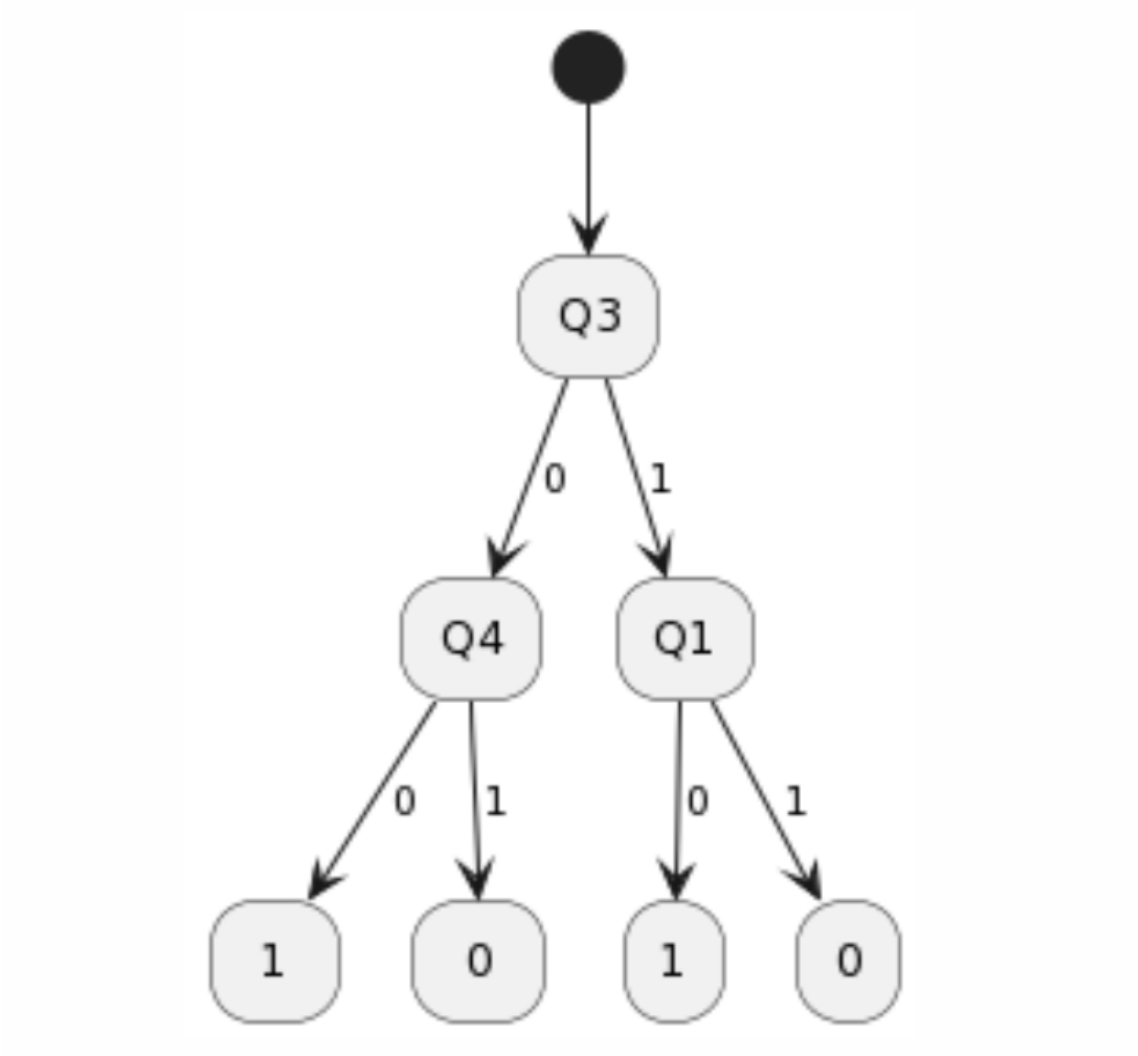
Q2	Q3	S(0)	S(1)	GI
1	0	2	0	0
0	0	2	1	1 - (1/3)^2 - (2/3)^2 = 0.44

`AVG GI(Q3=0 && Q2)= 3/5*0.44 = 0.264`

Q3	Q4	S(0)	S(1)	GI
0	0	2	1	1 - (1/3)^2 - (2/3)^2 = 0.44
0	1	2	0	0

`AVG GI(Q3=0 && Q4)= 3/5*0.44 = 0.264`

Stop splitting count in Q4, add the new node (Q3=0 && Q4)



S(10) = 1

KNN

Q1	Q2	Q3	Q4	S

0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	1	1	?

K=1, беремо k рядків з найменшими відстанями до (1,1,1,1).

За відстань беремо Евклідову відстань. Найменшою є (0,1,1,1) з відстанню 1.

При (0,1,1,1) S=1, тому при (1,1,1,1) S=1.

S(10) = 1

resources:

- https://medium.com/@jairiidriss/gini-gain-vs-gini-impurity-decision-tree-a-simple-explanation-a24ebfeebee9#:~:text=Gini%20Impurity(df)%20%3D%201,%2F14)%C2%B2%20%3D%200.459.

Plant UML decision tree:

```
@startuml
(*) --> "Q3"
 Q3 -->[0] "Q4"
 Q3 -->[1] "Q1"
 Q1 -->[0] "1"
 Q1 -->[1] "0"
 Q4 -->[0] " 1 "
 Q4 -->[1] " 0 "

@enduml
```