Artificial Intelligence

Learning

Chapter 18 “Artificial Intelligence A Modern Approach”

Chapter 5 “Inteligência Artificial Fundamentos e Aplicações”

POLITÉCNICO DE LEIRIA

Slides by Carlos Grilo

Learning, what is it?

- An agent is learning if it improves its performance on future tasks after making observations about the world
- Acquisition of new knowledge
- Restructuring knowledge previously acquired

Why would we want an agent to learn?

- Sometimes human programmers have no idea how to program a solution themselves
- Designers cannot anticipate all possible situations that the agent might find itself in
- Designers cannot anticipate all changes in the world over time

Learning strategies

- What distinguishes them:
  - The inferential apparatus used by the agent to learn
  - The previous knowledge of the agent
  - How much the agent depends on a teacher
  - How the agent interacts with the environment in order to learn

Learning strategies

- Direct implantation of knowledge
- By instruction
- By deduction
- Through analogy
- By induction

Direct implantation of knowledge

- The system inferential capacities are basically null
- Total dependence from a teacher
- No need of previous knowledge
- In the limit, we can view the implementation of a data base as one form of this type of learning

Learning by instruction

- The learning agent assimilates and integrates the knowledge transmitted by an external source (which can be an inteligent tutor system)
- The existence of previous knowledge can facilitate (or complicate, in the case of wrong believes) the learning process

Learning by deduction

- The learning agent is equipped with a (formal) theory describing the domain where the learning process is taking place
- Thanks to inference rules, the agent is able to analyze and understand the facts and questions posed about the domain
- The formal system, made up of the theory (the axioms) and the inference rules, constitute the previous knowledge
- The system learns through deductive inference, increasing its knowledge through the application of inference rules to the initial knowledge

Learning by analogy

- Consists in the transformation and increase of the knowledge and methods in one domain, so that they can be used in similar tasks in another domain or simply to new problems
- A previous task consists in discovering the analogy, which presupposes an inferential capacity superior to the one of previous types of learning
- It also presupposes the existence of previous knowledge about problems and their solutions, possibly, of different domains
- This knowledge may be provided by a teacher

Case based reasoning

- CBR can be viewed as a form of learning by analogy. It consists in using previous experiences directly as a way of solving new problems
- In CBR there is a case base and, to solve a new problem, we look that case base for other similar problem(s), which are then used as a basis to solve the problem
- There must be a similarity measure that allows us to measure the similarity between the new problem and previous ones

Inductive learning

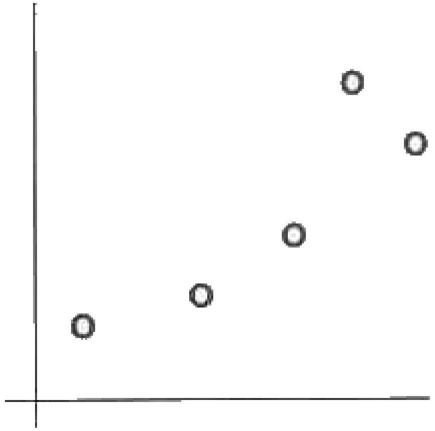
- Goal: given a set of examples, the system tries to infer general concepts or laws
- The system has no previous knowledge about the domain
- The inferences are not necessarily valid, unlike what happens with deduction. Putting it another way, the inferred concept may not be correct
- The lack of a previous theory forces the learning system into a considerable inferencial work

Inductive learning

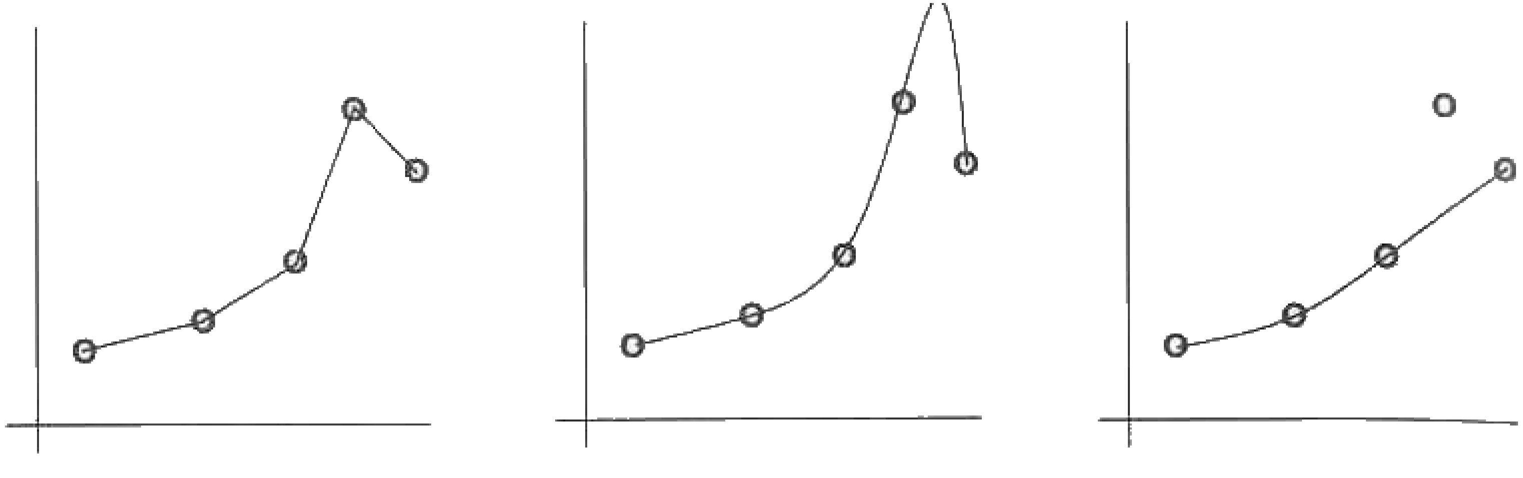
- It can be supervised or not
- When it is supervised, the system depends totally on the teacher, that presents it and classifies the examples of the concept or rule that is supposed to be learned
- When it is not supervised, the system learns by observation and/or by discovery using the environment passively (observation) or actively (experimentation) to acquire the data

Hypothesis generation

- Which function better describes these points?



Hypotheses



Inductive learning

- Given:
  - Instances (X): set of objects from the domain
  - Target concept (C): the concept supposed to be learned
  - Training examples (T): positive and negative examples (instances) of the target concept C
  - Hypotheses (H): set of hypotheses that describe the target concept
- Goal: find a hypothesis h from H such that $h(x) = c(x)$ for all x in X

Example: do sports?

- Instances (X): days described by attributes sky, temperature, humidity, wind, etc.
- Target concept (C): Do sports?
- Training examples (T):

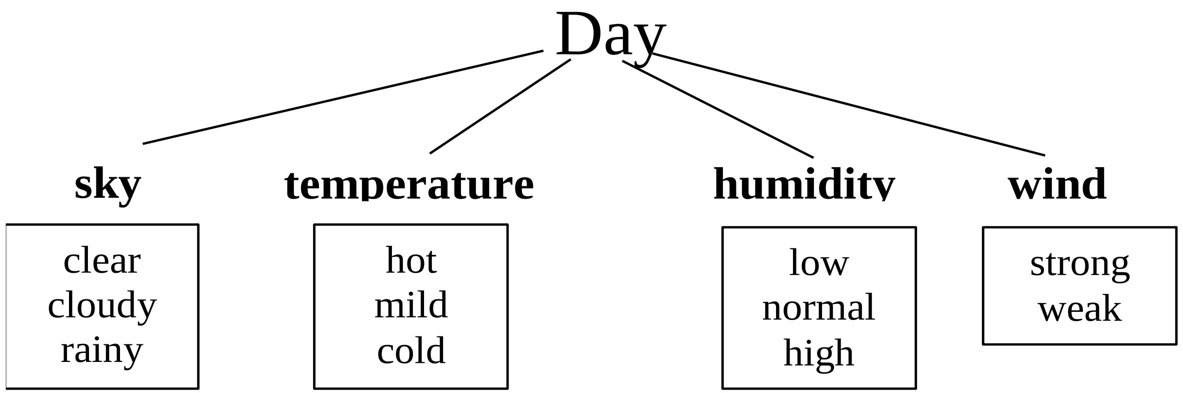
  < [sky = clear, temperature = mild, humidity = low, wind=strong], +>

  < [sky = cloudy, temperature = low, humidity = high, wind=weak], ->

  etc.
- Hypotheses (H):
  - [sky = clear and temperature = mild]
  - [sky = clear], etc.

Instances

- Instances are represented as a set of attribute/value pairs. For example:



- Instance example: x = [clear, hot, normal, strong]
- Notice: depending on the problem and the learning algorithm, the values may be discret or continuous

Training examples and hypoteses

- Training examples: <instance, c(instance)>

  Examples:

  Example 1 = <[clear, hot, normal, strong], **+**>

  Example 2 = <[rainy, cold, high, strong], **-**>

- Hypoteses are sets of restrictions over instances attributes:

  h1 = [sky = clear and temperature = mild]

  h2 = [sky = clear or humidity = low]

  h3 = [wind = strong]

Inductive learning

Problem: Characterize the clients from the following table that abandon the company

Name	Age	Consumption	Abandones
António	65	12	Y
Sílvia	86	27	N
Luís	45	29	N
Ana	55	36	Y
Teresa	35	32	Y
Rui	82	47	N
Jorge	65	43	N
Daniel	42	49	Y

Inductive learning

Easier solution: a solution for each name

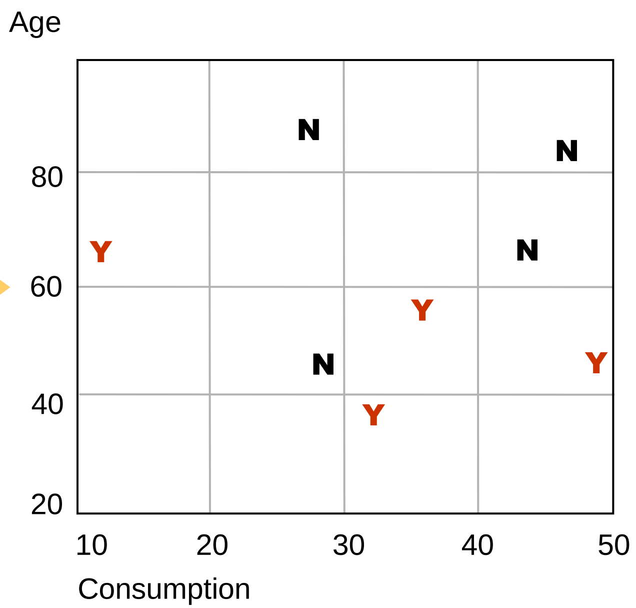
Problem: it is completely useless

Name	Abandones
António	Y
Sílvia	N
Luís	N
Ana	Y
Teresa	Y
Rui	N
Jorge	N
Daniel	Y

Let us try visual inspection

Inductive learning

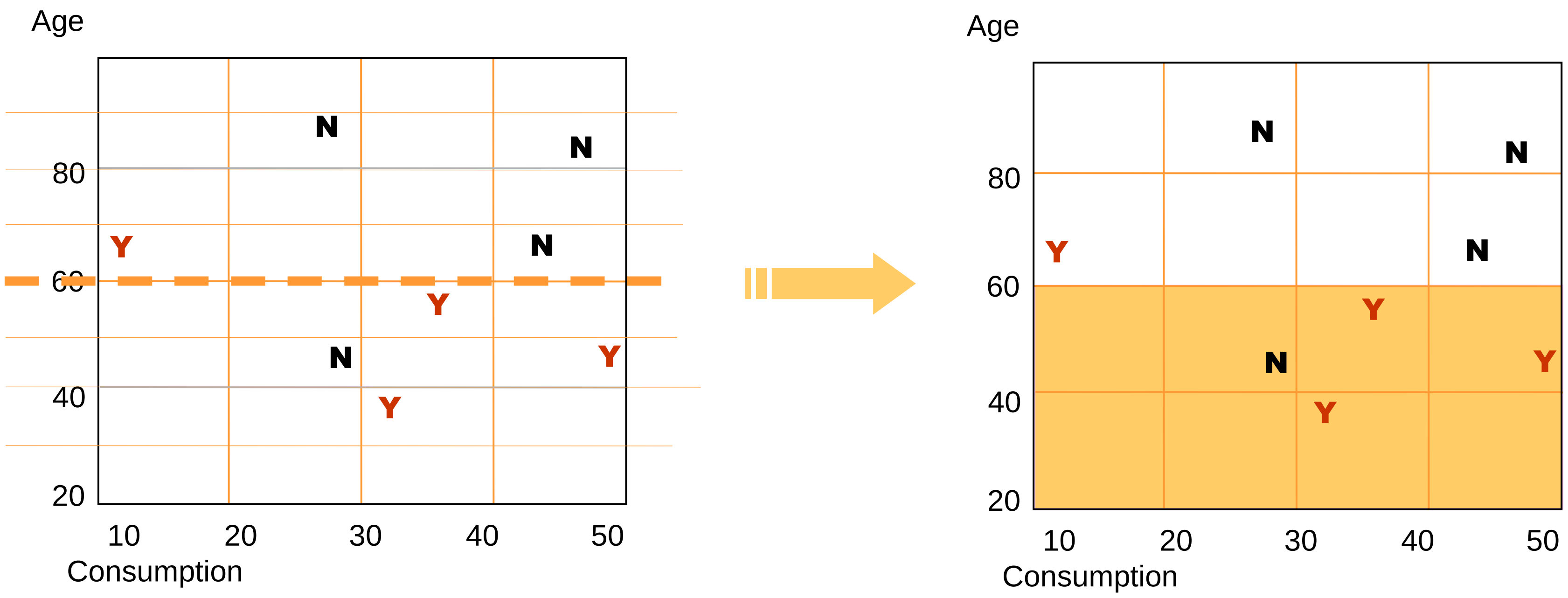
Name	Age	Consumption	Abandones
António	65	12	S
Sílvia	86	27	N
Luís	45	29	N
Ana	55	36	S
Teresa	35	32	S
Rui	82	47	N
Jorge	65	43	N
Daniel	42	49	S



Conclusion: still, it is not easy to identify the models ruling the data

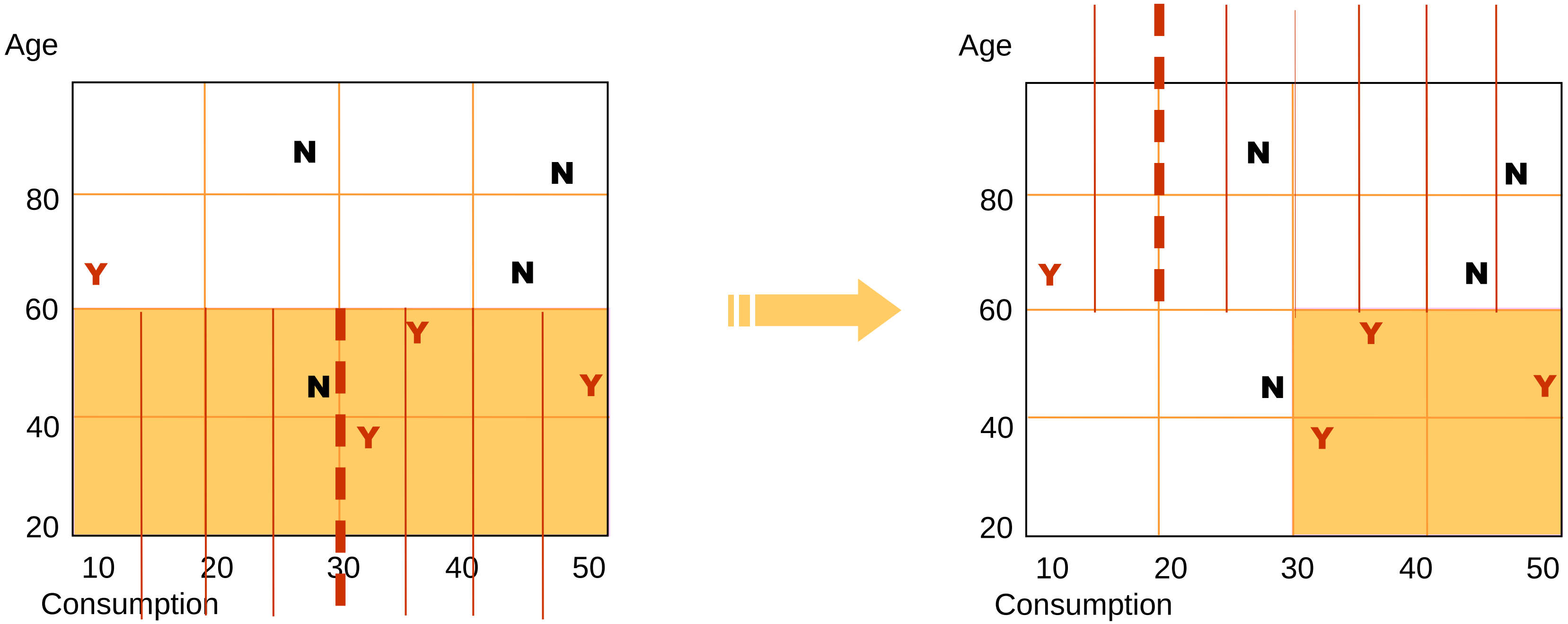
Inductive learning

Let us try to linearly separate de Ns from the Ys (best effort)

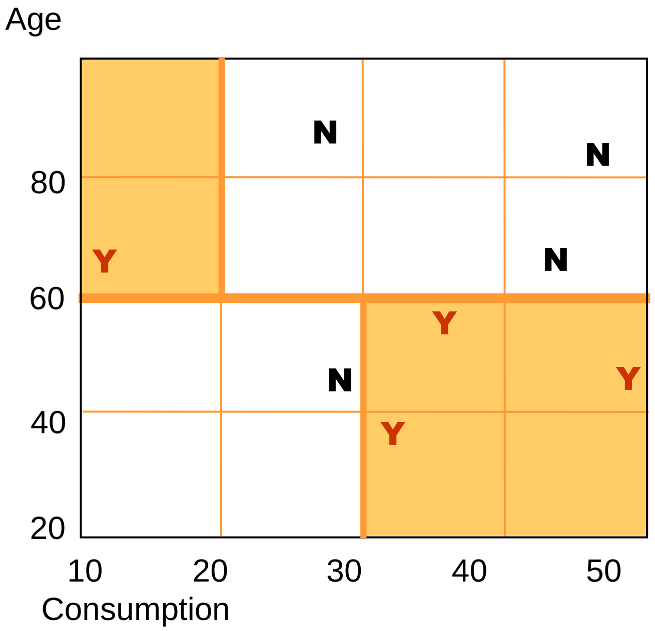


Inductive learning

Now, we do the same for each of the two resulting areas



Inductive learning



Rule set

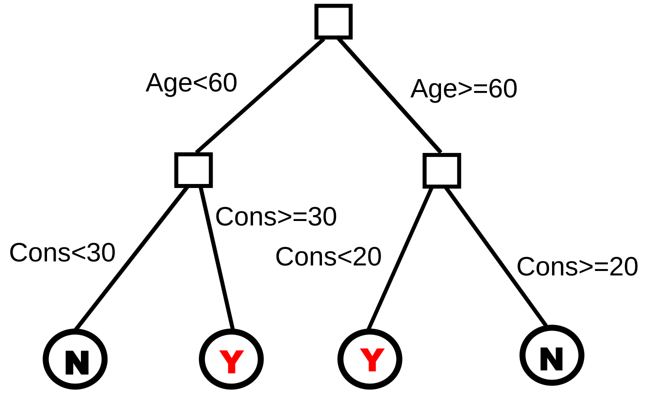
If Age<60 **and** Consumption<30 **then** does not abandon

If Age<60 **and** Consumption>=30 **then** abandones

If Age>=60 **and** Consumption<20 **then** abandones

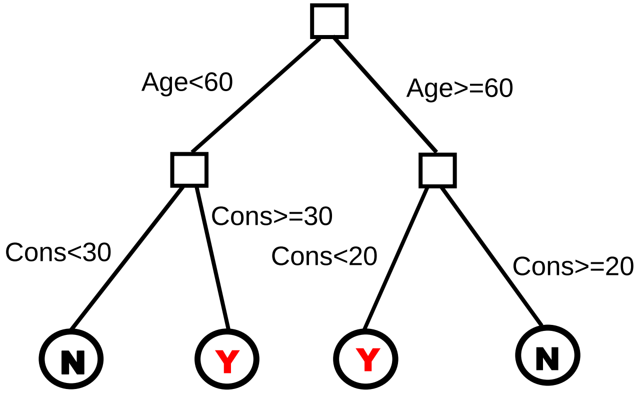
If Age>=60 **and** Consumption>=20 **then** does not abandon

Decision tree



Inductive learning

Name	Age	Consumption	Abandones
José	25	28	?
Maria	54	11	?
Ana	87	47	?
Filipa	32	13	?
Rosa	43	14	?
Carlos	34	38	?
António	21	49	?
Zara	72	14	?



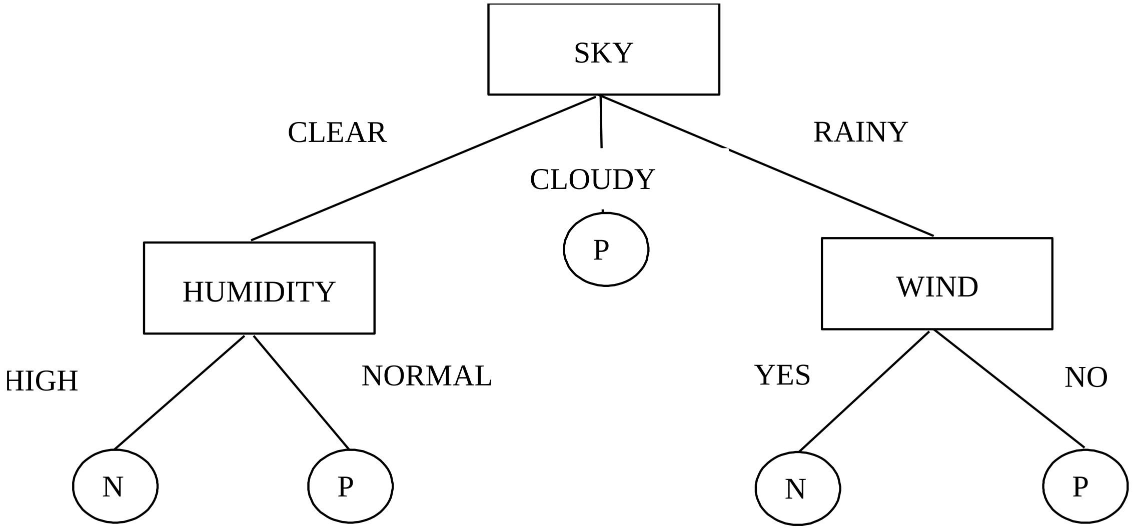
Name	Age	Consumption	Abandones
José	25	28	N
Maria	54	11	N
Ana	87	47	N
Filipa	32	13	N
Rosa	43	14	N
Carlos	34	38	Y
António	21	49	Y
Zara	72	14	Y

ID3 (decision trees)

- ID3 is a supervised algorithm which, from a set of positive and negative training examples of some class, builds a decision tree that defines that class
- Types of problems where ID3 is used:
  - Instances represented by attribute/value pairs
  - Target concept has nominal outputs
  - Target concept may have a disjunctive representation

    Example: sky = clear **or** temperature = mild

ID3 – representation of decision trees



- Each internal node tests an attribute
- Each branch corresponds to a possible value for that attribute
- Each leaf establishes a classification
- Each path root-leaf defines a rule

ID3 – instancies classification

- In order to classify an instance with a decision tree, we start by the root, testing the attribute specified by this node, and then following to the branch corresponding to the attribute value in that instance
- This process is repeated for the subtree of the next node
- In the end, a yes or no answer is given, that is, if the instance belongs, or not, to the class defined by the tree
- Therefore, decision trees represent boolean functions

ID3 – algorithm

```
ID3(Examples, Target-Attribute, Attributes) returns a decision tree
    Create a Root node for the tree
    If all the examples are positive then return a tree with only the Root node labeled with P
    If all the examples are negative then return a tree with only the Root node labeled with N
    If Attributes is empty then return a tree with only the root node labeled with the most common value of Target-Attribute in Examples
    Else
            $A$ = the attribute from Attributes that better classifies the Examples
            Label of root = $A$
            For each possible value, $v_i$, of $A$:
                Add a branch to the Root with test value $A = v_i$
                $Examples_{vi}$ = subset of Examples with $A = v_i$
                If $Examples_{vi}$ = empty then
                    Add a leaf node to the branch labeled with the most common value of Target-Attribute in Examples
              Else add to the branch the subtree returned by
                    ID3($Examples_{vi}$, Target-Attribute, Attributes – {A})
Return Root
```

ID3 – example

- **Target concept**: do sports?

Training examples

Nº	Sky	Temperature	Humidity	Wind	DoSports?
1	Clear	Hot	High	No	N
2	Clear	Hot	High	Yes	N
3	Cloudy	Hot	High	No	P
4	Rainy	Mild	High	No	P
5	Rainy	Cold	Normal	No	P
6	Rainy	Cold	Normal	Yes	N
7	Cloudy	Cold	Normal	Yes	P
8	Clear	Mild	High	No	N
9	Clear	Cold	Normal	No	P
10	Rainy	Mild	Normal	No	P
11	Clear	Mild	Normal	Yes	P
12	Cloudy	Mild	High	Yes	P
13	Cloudy	Hot	Normal	No	P
14	Rainy	Mild	High	Yes	N

ID3 – example (continued)

Let us consider the first step of the algorithm, where the root node is created

What attribute should be chosen?

Sky:

Clear	{1, 2, 8, 9, 11}	[2+, 3-]
Cloudy	{3, 7, 12, 13}	[4+, 0-]
Rainy	{4, 5, 6, 10, 14}	[3+, 2-]

Temperature:

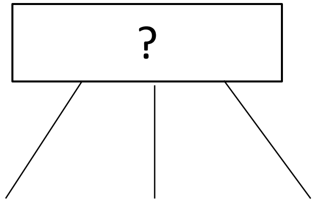
Hot	{1, 2, 3, 13}	[2+, 2-]
Mild	{4, 8, 10, 11, 12, 14}	[4+, 2-]
Cold	{5, 6, 7, 9}	[3+, 1-]

Humidity:

High:	{1, 2, 3, 4, 8, 12, 14}	[3+, 4-]
Normal	{5, 6, 7, 9, 10, 11, 13}	[6+, 1-]

Wind:

No	{1, 3, 4, 5, 8, 9, 10, 13}	[6+, 2-]
Yes	{2, 6, 7, 11, 12, 14}	[3+, 3-]

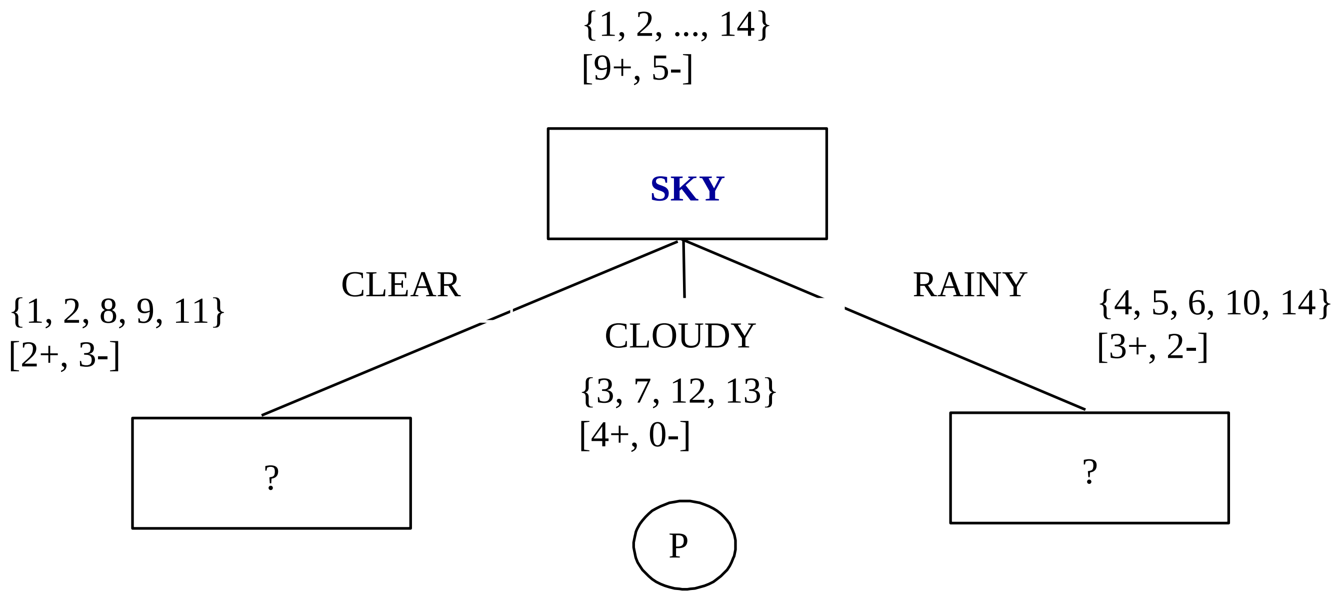


The idea is to choose the most discriminant attribute ---> the one that better separates the positive examples from the negative ones

ID3 – example (continued)

Apparently, Sky is the most discriminant attribute

Therefore, for now, we have the following tree:



ID3 – example (continued)

We must now do the same for each of the nodes that are not labeled yet, using only attributes Temperature, Humidity, Wind

Left node - examples: {1, 2, 8, 9, 11}

Temperature:

Hot	{1, 2}	[0+, 2-]
Mild	{8, 11}	[1+, 1-]
Cold	{9}	[1+, 0-]

Humidity:

High	{1, 2, 8}	[0+, 3-]
Normal	{9, 11}	[2+, 0-]

Wind:

Yes{2, 11}	[1+, 1-]	
No	{1,8,9}	[1+, 2-]

We can easily verify that attributes Temperature and Wind are not as discriminative as attribute Humidity

So, we label this node with label Humidity

ID3 – example (continued)

Let us now see the right node - examples: {4, 5, 6, 10, 14}:

Temperature:

Hot	{}	[0+, 0-]
Mild	{4, 10, 14}	[2+, 1-]
Cold	{5, 6}	[1+, 1-]

Humidity:

High	{4, 14}	[1+, 1-]
Normal	{5, 6, 10}	[2+, 1-]

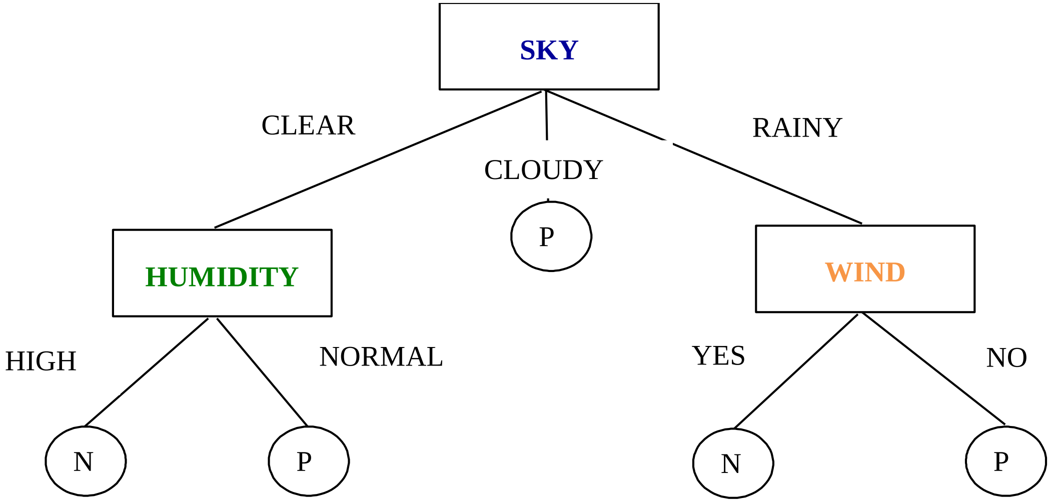
Wind:

Yes{6, 14}	[0+, 2-]	
No	{4,5,10}	[3+, 0-]

We can also see that attribute Wind is more discriminative than attributes Temperature and Humidity, therefore, we label the node with value Wind

ID3 – example (continued)

The final tree is as follows:



Note: not all attributes were used

The built tree represents the learned target concept and should be read as follows:

C = [Sky = cloudy] **or**

[Sky = clear **and** Humidity = normal] **or**

[Sky = rainy **and** Wind = no]

The most discriminant attribute

- In order to decide what is the most discriminant attribute, we need:

    1. To have a way of measuring the disorder or entropy of a set of already classified training examples

    2. To have a way of measuring how much we reduce the entropy when some attribute is chosen to separate the examples

Entropy

- If all the elements of a set T belong to the same class, the entropy is equal to 0

- If half the elements of the set belong to one class and the other half belong to another class, the entropy is equal to 1 (maximum entropy value)

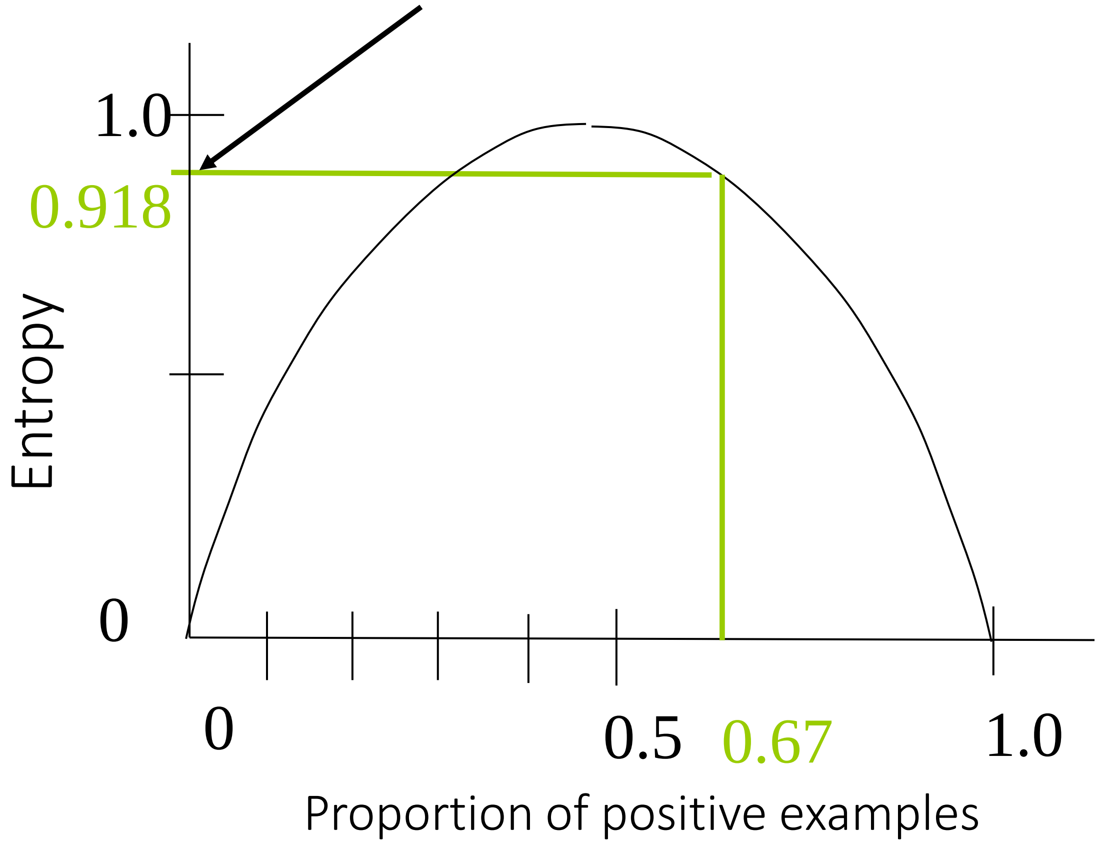
Entropy - example

Name	Hair	Height	Weight	Lotion	Burn
Sara	Blond	Medium	Light	No	Yes
Daniel	Blond	Tall	Medium	Yes	No
Xico	Brown	Short	Medium	Yes	No
Ana	Blond	Short	Medium	No	Yes
Emília	Ruivo	Medium	Heavy	No	Yes
Pedro	Brown	Tall	Heavy	No	No
João	Brown	Medium	Heavy	No	No
Carla	Blond	Short	Light	Yes	No

- Entropy({Daniel, Pedro}) = 0
- Entropy({Sara, Ana, Emília}) = 0
- Entropy({Sara, Emília, Xico, João}) = 1
- **Entropy({Sara, Emília, Xico}) = ?**

Entropy

Entropy({Sara, Emília, Xico }) = 0.918



Entropy

- Entropy allow us to measure the disorder in a set T of examples, where *p* are positive examples and *n* are negative examples (*p* + *n* = |T|)

$$Entropy(T) = I(p,n) = -\frac{p}{p+n}log_2\frac{p}{p+n} - \frac{n}{p+n}log_2\frac{n}{p+n}$$

Entropy example

- Entropy({Sara, Daniel, Xico, Ana, Emília, Pedro, João, Carla})

$$= I(3{,}5) = -\frac{3}{8}\log_2\frac{3}{8} - \frac{5}{8}\log_2\frac{5}{8} = 0.954$$

- Some properties of entropy:

- I(m, n) = I(n, m) I(m, 0) = 0 I(m, m) = 1

Information gain

- We now need to know how much we reduce the entropy if we use some specific attribute in order to separate the examples

- The Information gain measures how much one hopes to reduce the entropy if we divide T using attribute A

$$Gain(T,A) = Entropy(T) - \sum_{v \in values\ of\ A} \frac{|T_v|}{|T|} Entropy(T_v)$$

$$= Entropy(T) - \sum_{v \in values\ of\ A} \frac{p_v + n_v}{p + n} I(p_v, n_v)$$

ID3 – example

- **Target concept**: do sports?

Entropy of the original training set:

$$Entropy(T) = I(9,5)$$

$$= -\frac{9}{14}\log_2\frac{9}{14} - \frac{5}{14}\log_2\frac{5}{14} = 0.9402$$

Training examples

Nº	Sky	Temperature	Humidity	Wind	DoSports?
1	Clear	Hot	High	No	N
2	Clear	Hot	High	Yes	N
3	Cloudy	Hot	High	No	P
4	Rainy	Mild	High	No	P
5	Rainy	Cold	Normal	No	P
6	Rainy	Cold	Normal	Yes	N
7	Cloudy	Cold	Normal	Yes	P
8	Clear	Mild	High	No	N
9	Clear	Cold	Normal	No	P
10	Rainy	Mild	Normal	No	P
11	Clear	Mild	Normal	Yes	P
12	Cloudy	Mild	High	Yes	P
13	Cloudy	Hot	Normal	No	P
14	Rainy	Mild	High	Yes	N

ID3 – example (continued)

Let us consider the first step of the algorithm, where the root node is created

What attribute should be chosen?

Sky:

Clear	{1, 2, 8, 9, 11}	[2+, 3-]	$Entropy(T) = I(2,3) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5} = 0.9709$
Cloudy	{3, 7, 12, 13}	[4+, 0-]	$Entropy(T) = I(4,0) = 0$
Rainy	{4, 5, 6, 10, 14}	[3+, 2-]	$Entropy(T) = I(3,2) = -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.9709$

Temperature:

Hot	{1, 2, 3, 13}	[2+, 2-]	$Entropy(T) = I(2,2) = 1$
Mild	{4, 8, 10, 11, 12, 14}	[4+, 2-]	$Entropy(T) = I(4,2) = -\frac{4}{6}\log_2\frac{4}{6} - \frac{2}{6}\log_2\frac{2}{6} = 0.9183$
Cold	{5, 6, 7, 9}	[3+, 1-]	$Entropy(T) = I(3,1) = -\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4} = 0.8112$

Humidity:

High:	{1, 2, 3, 4, 8, 12, 14}	[3+, 4-]	$Entropy(T) = I(3,4) = -\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7} = 0.9852$
Normal	{5, 6, 7, 9, 10, 11, 13}	[6+, 1-]	$Entropy(T) = I(6,1) = -\frac{6}{7}\log_2\frac{6}{7} - \frac{1}{7}\log_2\frac{1}{7} = 0.5916$

Wind:

No	{1, 3, 4, 5, 8, 9, 10, 13}	[6+, 2-]	$Entropy(T) = I(6,2) = -\frac{6}{8}\log_2\frac{6}{8} - \frac{2}{8}\log_2\frac{2}{8} = 0.8112$
Yes	{2, 6, 7, 11, 12, 14}	[3+, 3-]	$Entropy(T) = I(3,3) = 1$

ID3 – example (continued)

Let us consider the first step of the algorithm, where the root node is created

What attribute should be chosen?

Sky:

Clear	{1, 2, 8, 9, 11}	[2+, 3-]
Cloudy	{3, 7, 12, 13}	[4+, 0-]
Rainy	{4, 5, 6, 10, 14}	[3+, 2-]

$$Gain(T, Sky) = 0.9402 - (\frac{5}{14} * 0.9709 + \frac{4}{14} * 0 + \frac{5}{14} * 0.9709) = 0.2467$$

Temperature:

Hot	{1, 2, 3, 13}	[2+, 2-]
Mild	{4, 8, 10, 11, 12, 14}	[4+, 2-]
Cold	{5, 6, 7, 9}	[3+, 1-]

$$Gain(T, Temperature) = 0.9402 - (\frac{4}{14} * 1 + \frac{6}{14} * 0.9183 + \frac{4}{14} * 0.8112) = 0.02915$$

Humidity:

High:	{1, 2, 3, 4, 8, 12, 14}	[3+, 4-]
Normal	{5, 6, 7, 9, 10, 11, 13}	[6+, 1-]

$$Gain(T, Humidity) = 0.9402 - (\frac{7}{14} * 0.9852 + \frac{7}{14} * 0.5916) = 0.1518$$

Wind:

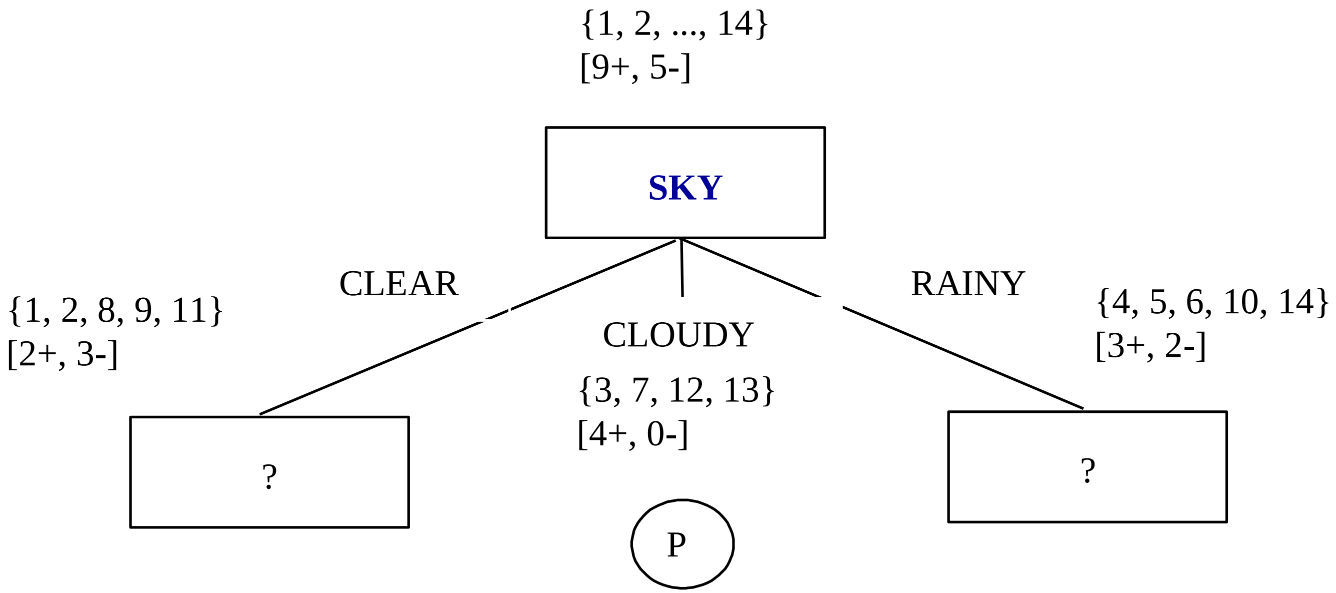
No	{1, 3, 4, 5, 8, 9, 10, 13}	[6+, 2-]
Yes	{2, 6, 7, 11, 12, 14}	[3+, 3-]

$$Gain(T, Wind) = 0.9402 - (\frac{8}{14} * 0.8112 + \frac{6}{14} * 1) = 0.0480$$

ID3 – example (continued)

Sky is the most discriminant atribute because it has the larger information gain

Therefore, for now, we have the following tree:



ID3 – example (continued)

We must now do the same for each of the nodes that are not labeled yet, using only attributes Temperature, Humidity, Wind

Left node - examples: {1, 2, 8, 9, 11}

Temperature:

Hot	{1, 2}	[0+, 2-]	$Entropy(T) = I(0,2) = 0$
Mild	{8, 11}	[1+, 1-]	$Entropy(T) = I(1,1) = 1$
Cold	{9}	[1+, 0-]	$Entropy(T) = I(1,0) = 0$

Humidity:

High	{1, 2, 8}	[0+, 3-]	$Entropy(T) = I(0,3) = 0$
Normal	{9, 11}	[2+, 0-]	$Entropy(T) = I(2,0) = 0$

Wind:

Yes	{2, 11}	[1+, 1-]	$Entropy(T) = I(1,1) = 1$
No	{1,8,9}	[1+, 2-]	$Entropy(T) = I(1,2) = -\frac{1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3} = 0.9183$

ID3 – example (continued)

We must now do the same for each of the nodes that are not labeled yet, using only attributes Temperature, Humidity, Wind

Left node - examples: {1, 2, 8, 9, 11}

Temperature:

Hot	{1, 2}	[0+, 2-]
Mild	{8, 11}	[1+, 1-]
Cold	{9}	[1+, 0-]

$$Gain(T, Temperature) = 0.9709 - (\frac{2}{5} * 0 + \frac{2}{5} * 1 + \frac{1}{5} * 0) = 0.5709$$

Humidity:

High	{1, 2, 8}	[0+, 3-]
Normal	{9, 11}	[2+, 0-]

$$Gain(T, Humidity) = 0.9709 - (\frac{3}{5} * 0 + \frac{2}{5} * 0) = 0.9709$$

Wind:

Yes	{2, 11}	[1+, 1-]
No	{1,8,9}	[1+, 2-]

$$Gain(T, Wind) = 0.9709 - (\frac{2}{5} * 1 + \frac{3}{5} * 0.9183) = 0.0199$$

ID3 – example (continued)

Let us now see the right node - examples: {4, 5, 6, 10, 14}:

Temperature:

Hot	{}	[0+, 0-]	$Entropy(T) = I(0,0) = 0$
Mild	{4, 10, 14}	[2+, 1-]	
Cold	{5, 6}	[1+, 1-]	$Entropy(T) = I(1,1) = 1$

Humidity:

High	{4, 14}	[1+, 1-]	$Entropy(T) = I(1,1) = 1$
Normal	{5, 6, 10}	[2+, 1-]	$Entropy(T) = I(2,1) = -\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} = 0.9183$

Wind:

Yes	{6, 14}	[0+, 2-]	$Entropy(T) = I(0,2) = 0$
No	{4,5,10}	[3+, 0-]	$Entropy(T) = I(3,0) = 0$

ID3 – example (continued)

Let us now see the right node - examples: {4, 5, 6, 10, 14}:

Temperature:

Hot	{}	[0+, 0-]
Mild	{4, 10, 14}	[2+, 1-]
Cold	{5, 6}	[1+, 1-]

$$Gain(T, Temperature) = 0.9709 - (\frac{0}{5} * 0 + \frac{3}{5} * 0.9183 + \frac{2}{5} * 1) = 0.0199$$

Humidity:

High	{4, 14}	[1+, 1-]
Normal	{5, 6, 10}	[2+, 1-]

$$Gain(T, Humidity) = 0.9709 - (\frac{2}{5} * 1 + \frac{3}{5} * 0.9183) = 0.0199$$

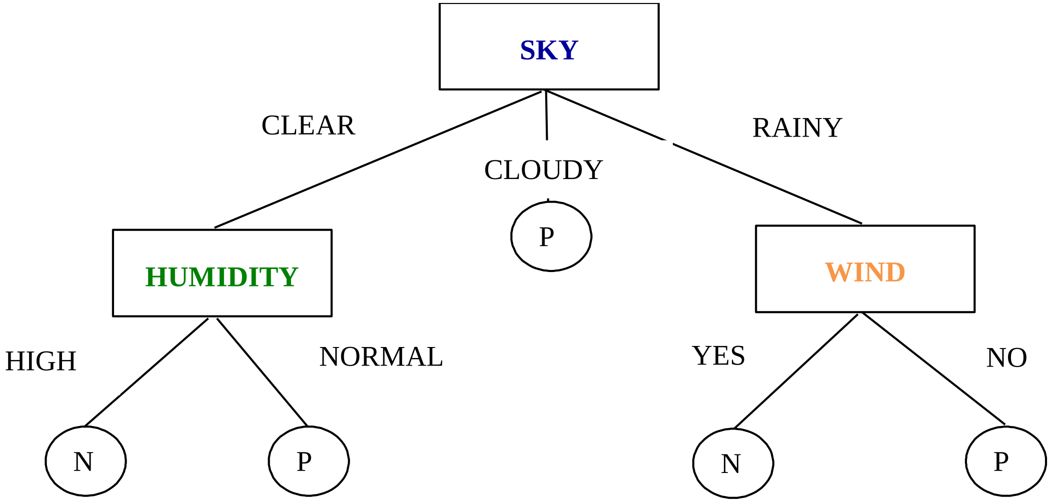
Wind:

Yes	{6, 14}	[0+, 2-]
No	{4,5,10}	[3+, 0-]

$$Gain(T, Humidity) = 0.9709 - (\frac{2}{5} * 0 + \frac{3}{5} * 0) = 0.9709$$

ID3 – example (continued)

The final tree is as follows:



Note: not all attributes were used

The built tree represents the learned target concept and should be read as follows:

C = [Sky = cloudy] **or**

[Sky = clear **and** Humidity = normal] **or**

[Sky = rainy **and** Wind = no]

Information gain – another example

- Let us consider a classification problem with the following attributes and their respective values:

Size = {Large, Medium, Small}

Weight = {Heavy, Medium, Light}

Shape = {Cube, Pyramid, Sphere, Parallelipiped}

- Suppose we have the following training examples:

Nº	Size	Weight	Shape	Classification
1	Medium	Heavy	Cube	**P**
2	Small	Medium	Pyramid	**N**
3	Small	Medium	Sphere	**P**
4	Large	Medium	Pyramid	**N**
5	Large	Light	Parallelipiped	**P**
6	Large	Medium	Parallelipiped	**N**
7	Large	Light	Sphere	**P**

Information gain – another example

- What is the most discriminant attribute?

- The entropy of the training set [4+, 3-] is
  - $I(4, 3) = -(0.57 * \log_2 0.57) - (0.43 * \log_2 0.43) = 0.99$

- What happens if we choose attribute Shape?

- We will have 4 new sets
  - [1+, 0-] (with value Cube)
  - [0+, 2-] (with value Pyramid)
  - [1+, 1-] (with value Parallelipiped)
  - [2+, 0-] (with value Sphere)

Information gain – another example

- The entropy of the set of examples with value Cube is equal to

$$I(1, 0) = -(1 * \log_2 1) - (0 * \log_2 0) = -(1 * 0) - (0 * 1) = 0$$

- Likewise, the sets corresponding to the values Pyramid and Sphere also have entropy 0

- The entropy of the set of examples with value Parallelipiped is equal to

$$I(1, 1) = -(0.5 * \log_2 0.5) - (0.5 * \log_2 0.5) = 1$$

Information gain – another example

- The information gain of using the Shape attribute is equal to

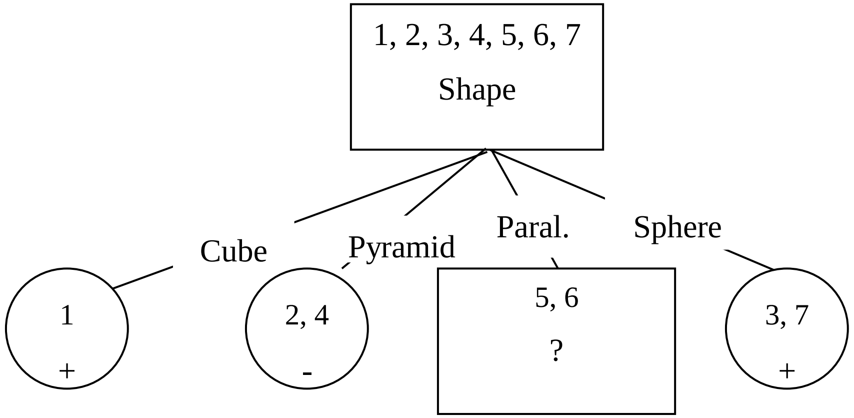
Gain(T, Shape) = 0.99 – ((1/7 * 0) + (2/7 * 0) + (2/7 * 1) + (2/7 * 0)) = 0.7

- Doing the same calculations for attributes Size and Weight, we get, respectively, the gains 0.13 and 0.52

- Therefore, we choose attribute Shape

Information gain – another example

- The resulting tree is as follows

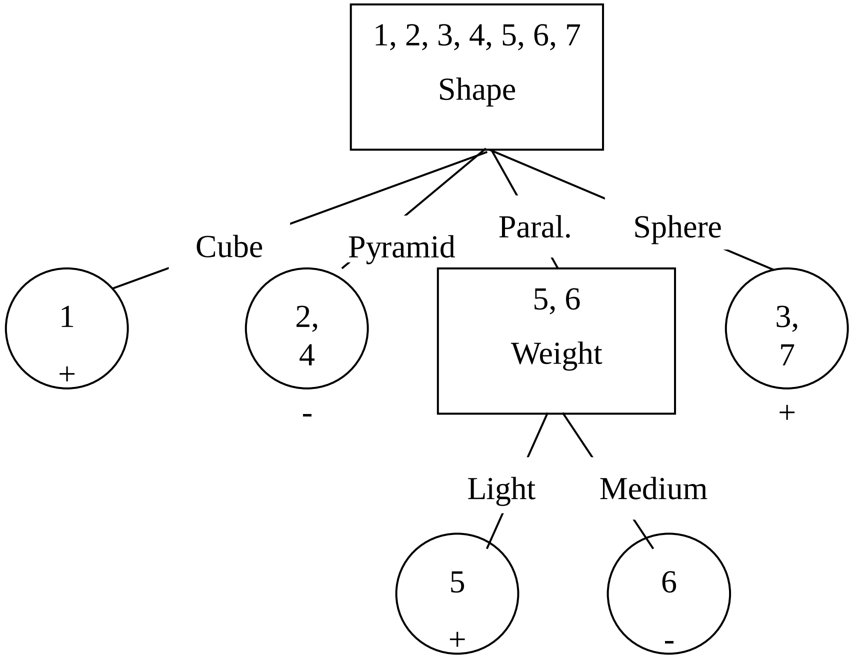


- In this tree, the node corresponding to the value Parallelipiped is the only one with examples of the two classes. Therefore, we need to repeat the process for this node

Information gain – another example

- Doing the calculation again, the chosen attribute will be attribute Weight (the gain of attribute Weight is equal to 1, while the gain of attribute Size is 0)

- The resulting tree is

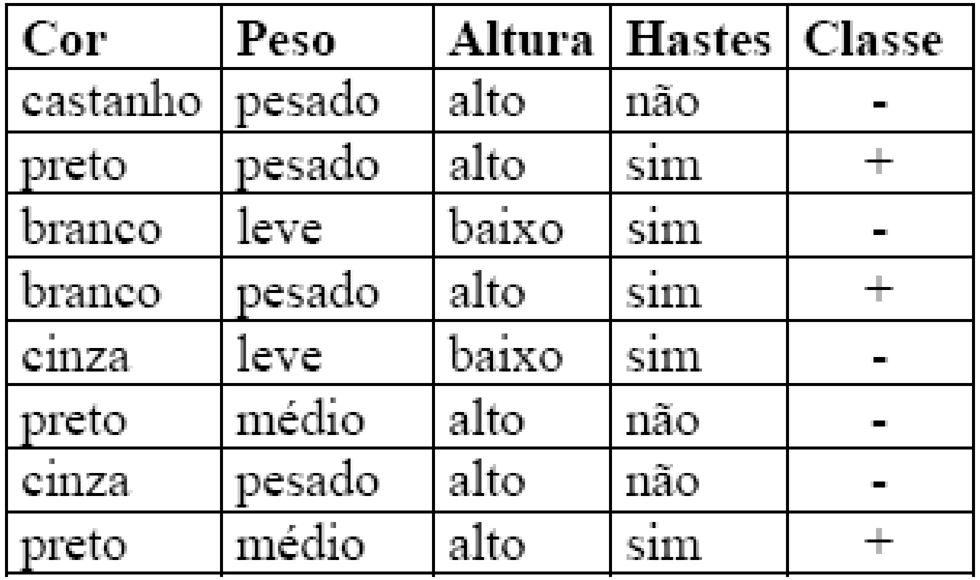


About overfitting

- The decision tree may not have a good performance when applied to other data
- As a limit, it is possible to build a tree with a branch for each training example
- We need to ajust the learning process in order to avoid overfitting

ID3 exercise

- Consider that we have the following training examples:



Cor	Peso	Altura	Hastes	Classe
castanho	pesado	alto	não	-
preto	pesado	alto	sim	+
branco	leve	baixo	sim	-
branco	pesado	alto	sim	+
cinza	leve	baixo	sim	-
preto	médio	alto	não	-
cinza	pesado	alto	não	-
preto	médio	alto	sim	+

- Please, use the ID3 algorithm to build a decision tree that is able to classify these examples