

Supervised Learning, Intro to Decision Trees Algorithm

Dr. Mohamed Mostafa

AI 511 Foundations of Artificial Intelligence

Lecture 8





Outline

- Machine Learning Concept
- Machine Learning Approaches
- What is Decision Tree
 - **Basic Decision Tree Algorithm**
 - Computing purity: Information Measure
 - Information Entropy
 - Information Gain
 - Examples
 - Advantage and Disadvantages of Decision Tree Algorithm

Resources for this lecture

- Data Mining And Knowledge Discovery

Handboook, Published at

[https://www.researchgate.net/publication/22523](https://www.researchgate.net/publication/225237661_Decision_Trees)

[7661_Decision_Trees](https://www.researchgate.net/publication/225237661_Decision_Trees).

.. AND ..

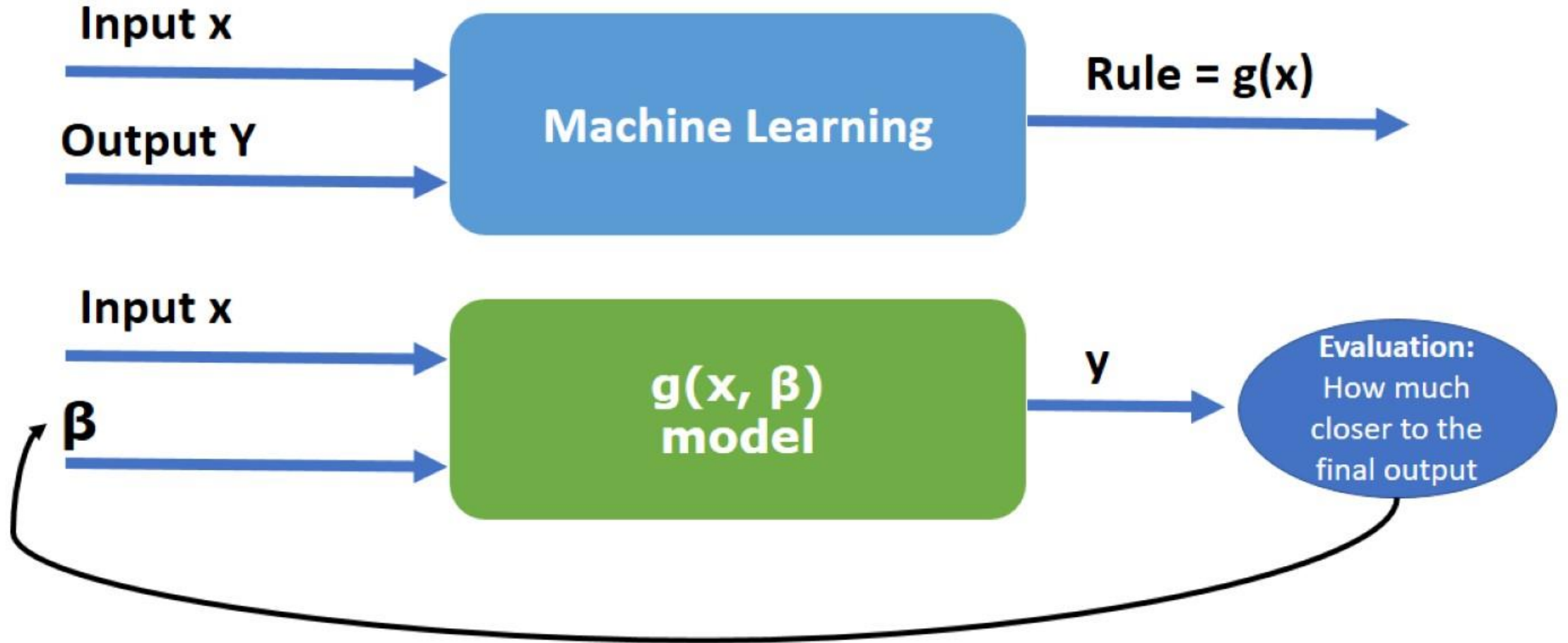
- Introduction to Artificial Intelligence Course,. Dr. Amr S. Ghoneim

Machine Learning Concept

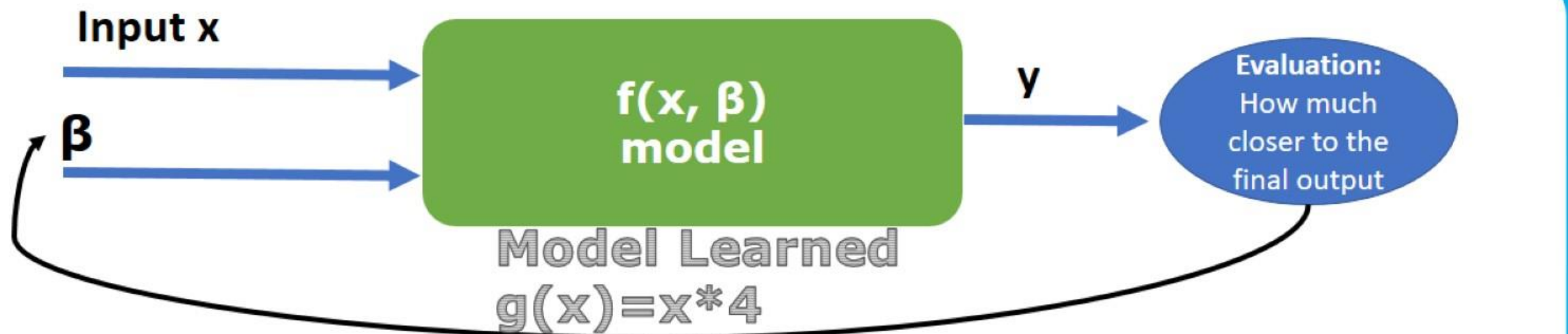


Input $x = 2$;
 Rule = $x * x$;
 $y = f(x) = 2 * 2 = 4$

Machine Learning Concept



Machine Learning Concept



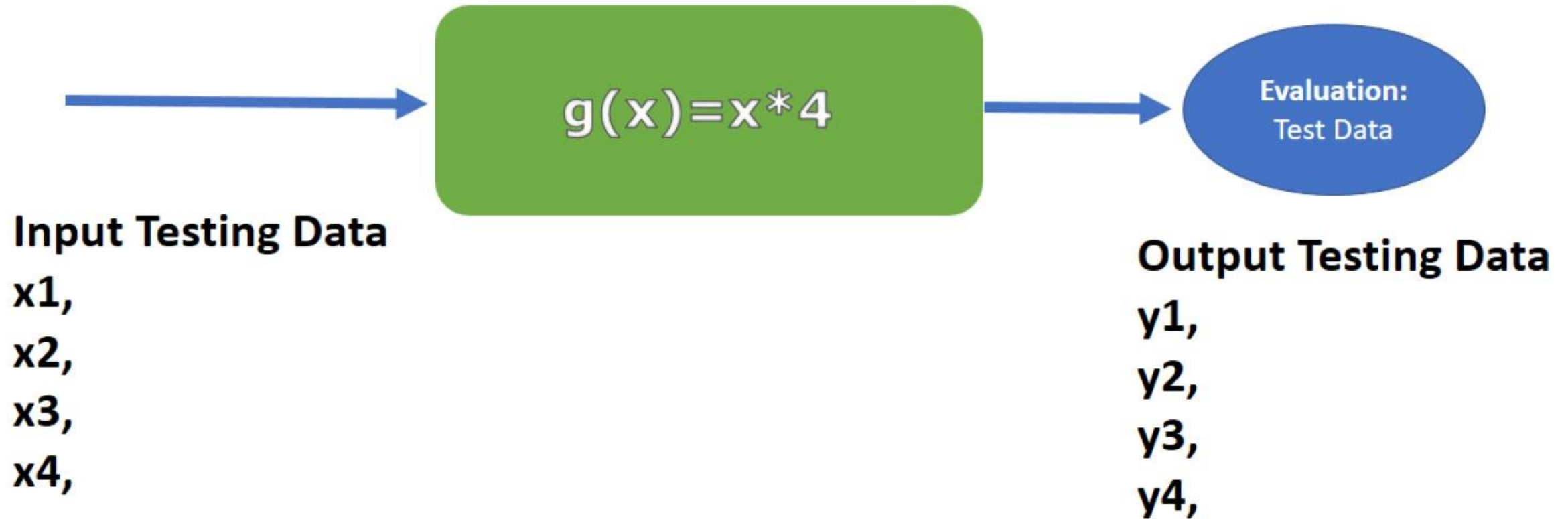
Input $x = 2$;
Output $y = 8$
 $y = 2 * \beta = 8$

$y = 2 * 1 = 2$
 $\text{eval} = 8 - 2 = 6$ (difference is 6)

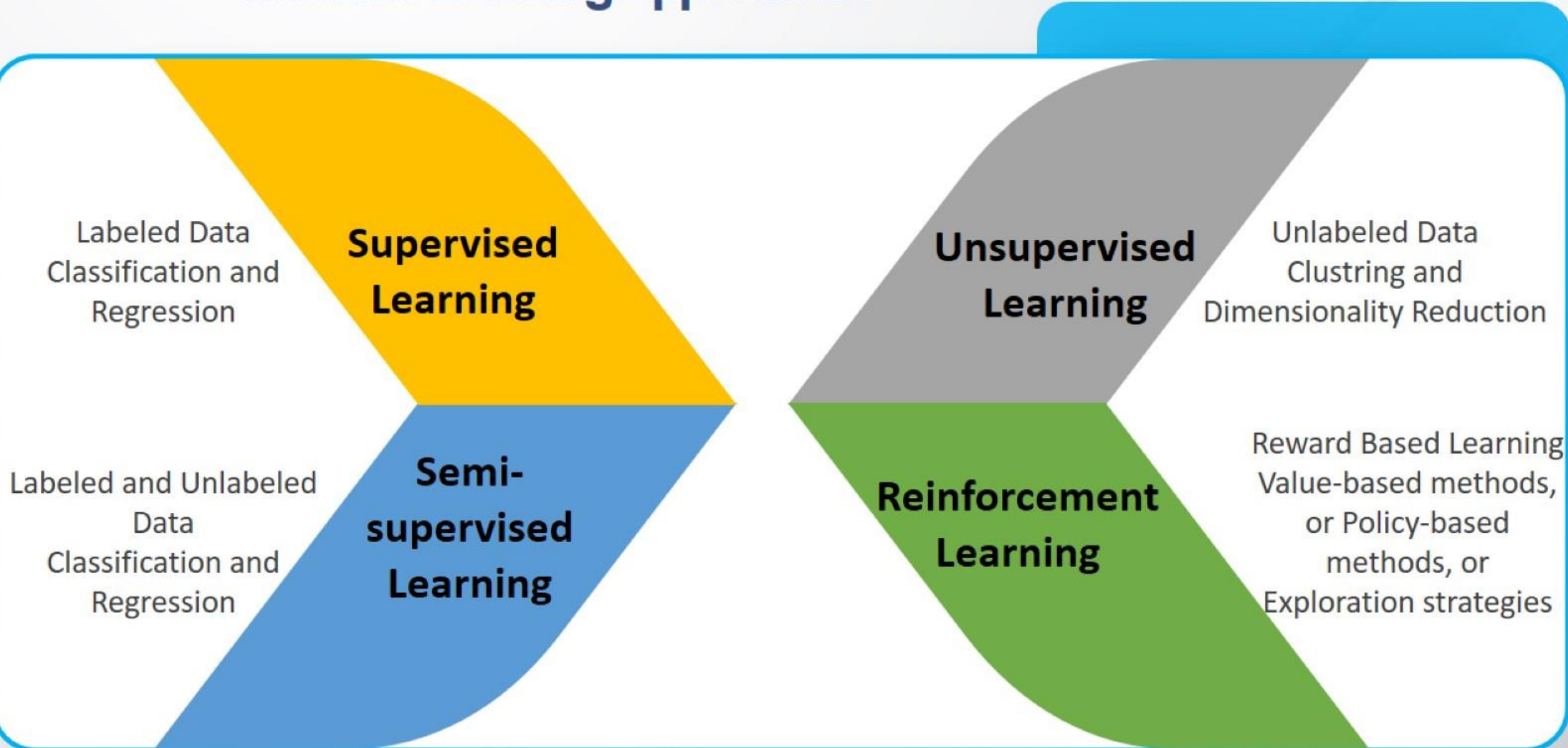
$y = 2 * 2 = 4$
 $\text{eval} = 8 - 4 = 4$ (difference is 4)

$y = 2 * 3 = 6$
 $\text{eval} = 8 - 6 = 2$ (difference is 2)

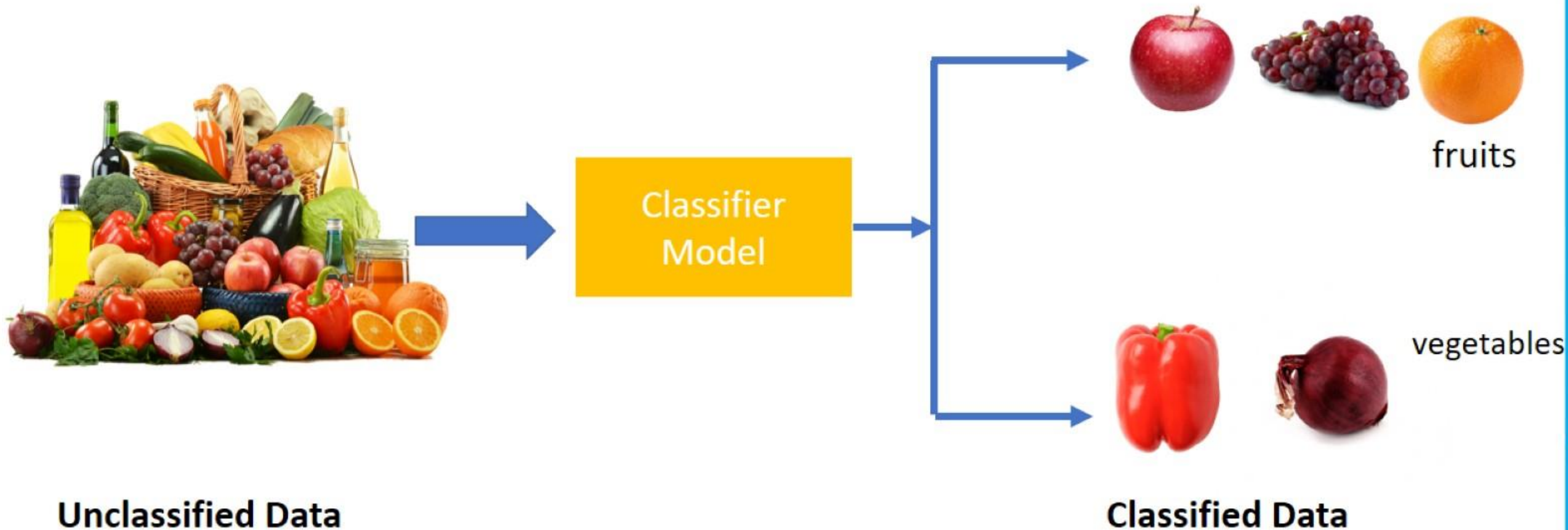
$y = 2 * 4 = 8$
 $\text{eval} = 8 - 8 = 0$ (difference is 0)



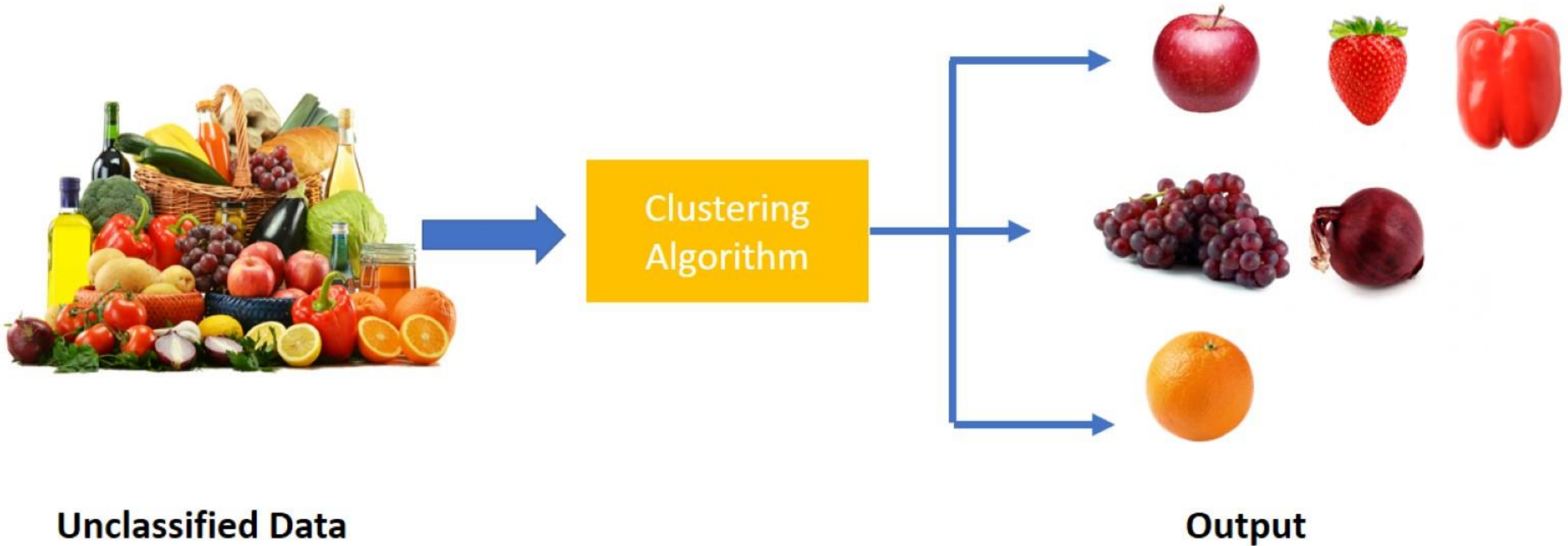
Machine Learning Approaches

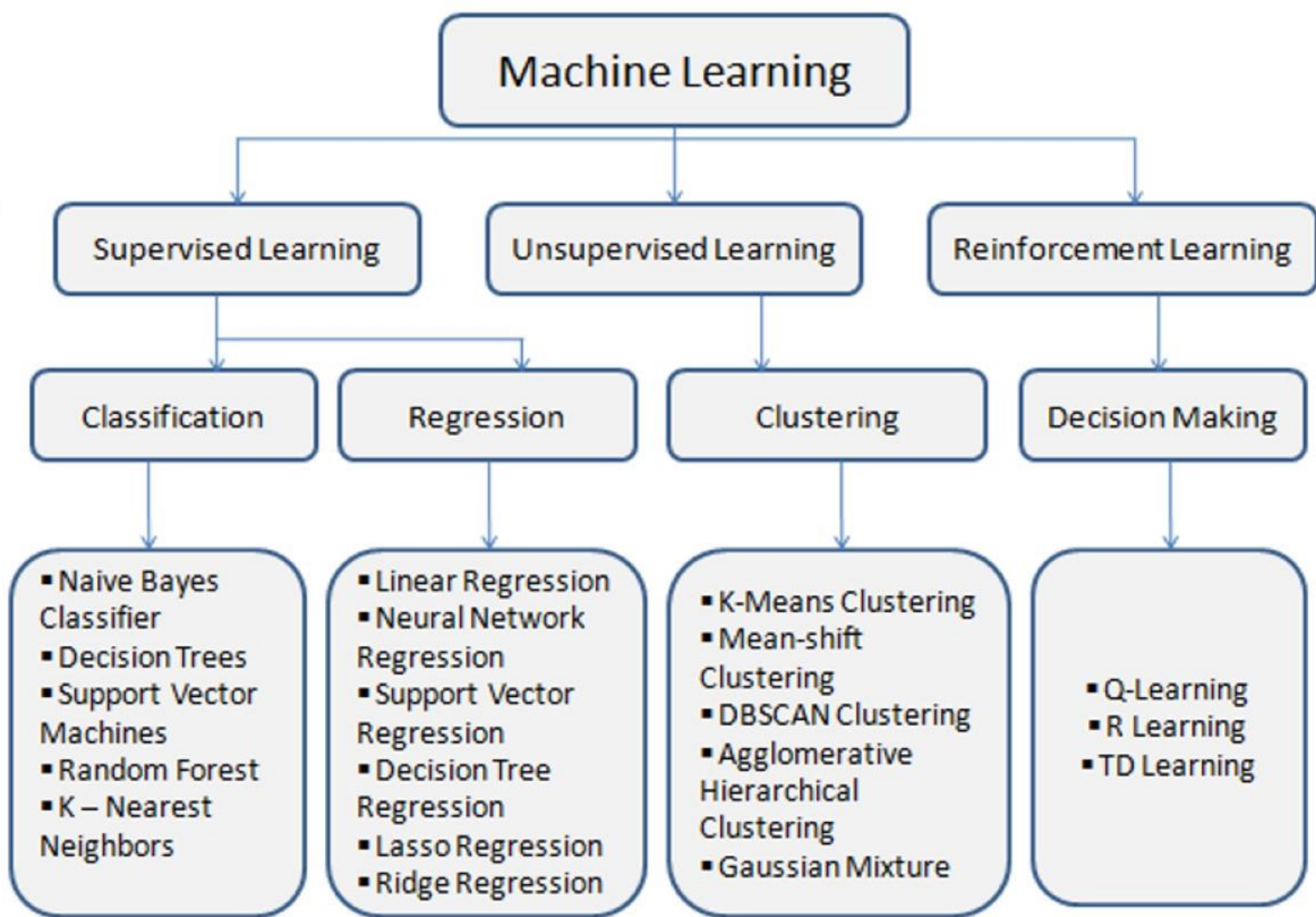


Classification Algorithms in Machine Learning



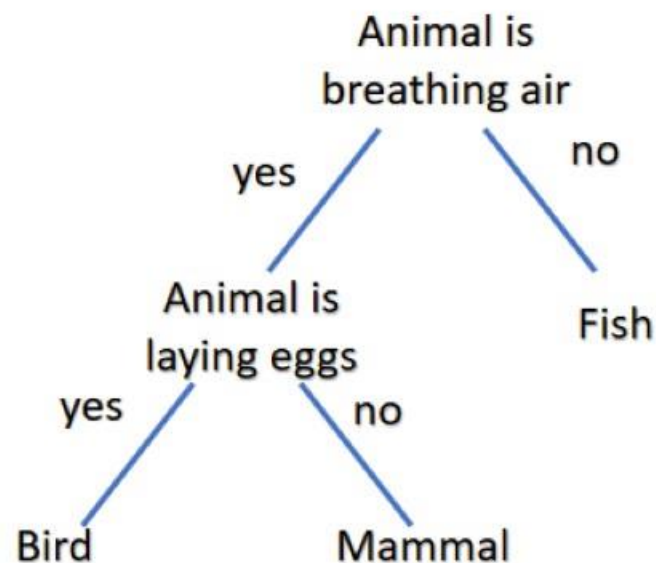
Clustering Algorithms in Machine Learning





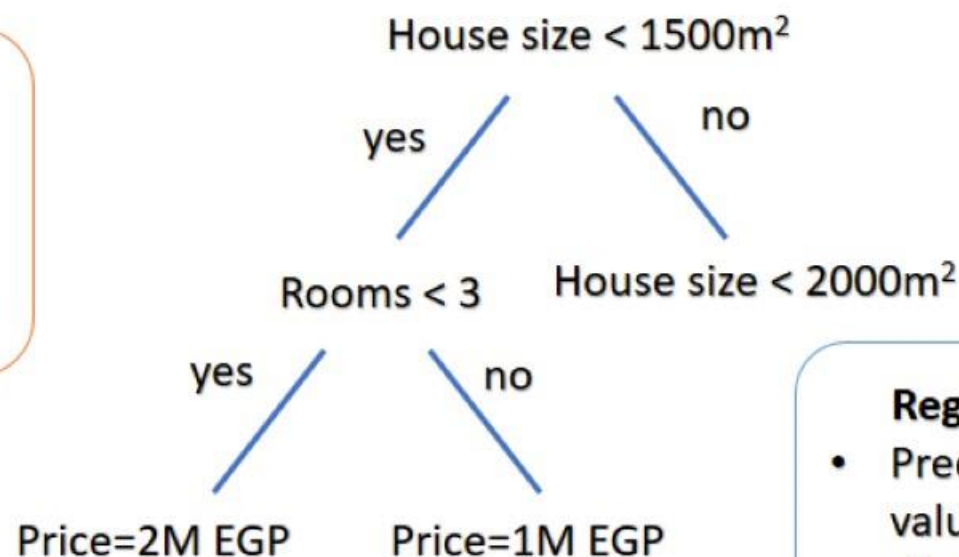
What is decision tree

- A decision tree is a classification approach in machine learning.
- Decision tree learning is a supervised learning method used in statistics, data mining, and machine learning to create a predictive model for classification or regression tasks.



Classification Tree

- Discrete values
- Classify into classes (categories)



Regression Tree

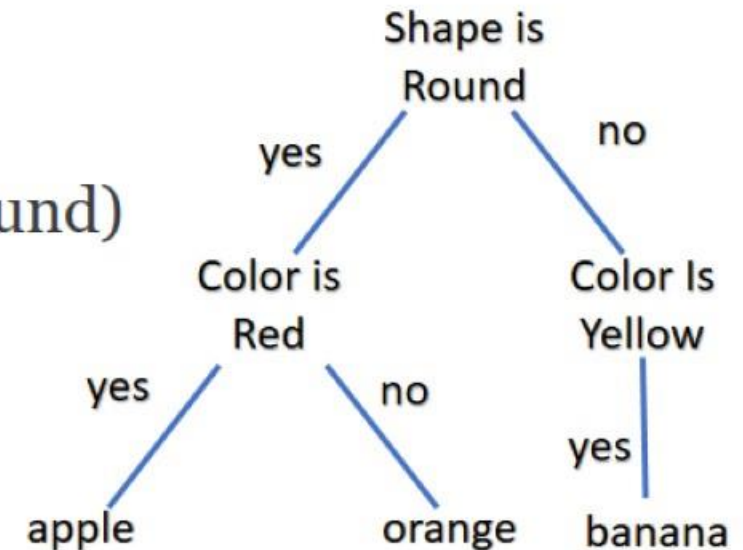
- Predict continuous values rather than discrete categories

Components of decision tree classification

- **Nodes:** test for the value of a certain attribute
- **Edges/ Branch:** correspond to the outcome of a test and connect to the next node or leaf
- **Leaf nodes:** terminal nodes that predict the outcome and represent class labels or class distribution

Consider a dataset of fruits where:

- An apple is represented as (color=red, shape=round)
- A banana is represented as (color=yellow, shape=elongated)
- An orange is represented as (color=orange, shape=round)



Basic Decision Tree Algorithm by Example

Given the following dataset:

No.	Outlook	Temperature	Humidity	Play Golf
1	sunny	hot	high	N
2	sunny	mild	high	N
3	overcast	hot	high	Y
4	rain	mild	high	Y
5	rain	cool	normal	N
6	rain	mild	normal	N
7	overcast	cool	normal	Y
8	sunny	cool	normal	Y
9	sunny	mild	high	?

Features

Outlook:

{ Sun, Overcast, Rain }

Temperature:

{ Hot, Mild, Cool }

Humidity:

{ High, Normal, Low }

Wind:

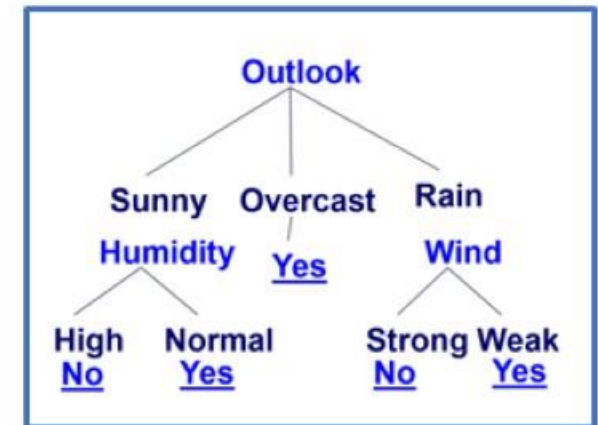
{ Strong, Weak }

Labels

Binary classification task = {Y, N}

Basic Decision Tree Algorithm by Example

- Let S be the set of Examples
 - **Label** is the target attribute (the prediction)
 - **Attributes** is the set of measured attributes
- Iterative Dichotomiser 3 Algorithm - $ID3(S, \text{Attributes}, \text{Label})$
- If all examples are labeled the same return a single node tree with Label
- Otherwise Begin
- A = attribute in **Attributes** that best classifies S (Create a Root node for tree)
- for each possible value v of A
- Add a new tree branch corresponding to $A = v$
- Let S_v be the subset of examples in S with $A = v$
- if S_v is empty: add leaf node with the common value of **Label** in S
- Else: below this branch add the subtree
- $ID3(S_v, \text{Attributes} - \{a\}, \text{Label})$
- End
- Return Root
-



Picking the Root Attribute

- The goal is to have the **resulting decision tree as small as possible**.
- The recursive algorithm is a **greedy heuristic search** for a simple tree, but **cannot guarantee optimality**.
- The **main decision** in the algorithm is the **selection of the next attribute to condition on**.

Picking the Root Attribute

- Consider data with two Boolean attributes (A,B).

$\langle (A = 0, B = 0), - \rangle$: 50 examples

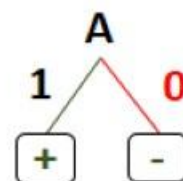
$\langle (A = 0, B = 1), - \rangle$: 50 examples

$\langle (A = 1, B = 0), - \rangle$: 0 examples

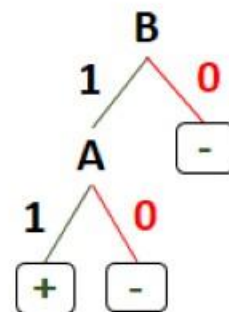
$\langle (A = 1, B = 1), + \rangle$: 100 examples

- What should be the first attribute we select?

- Splitting on A:** we get purely labeled nodes.
- Splitting on B:** we don't get purely labeled nodes.
- What if we have: $\langle (A = 1, B = 0), - \rangle$: 3 examples?



splitting on A



splitting on B

- (one way to think about it: # of queries required to label a random data point)*

Picking the Root Attribute

- Consider data with two Boolean attributes (A,B).

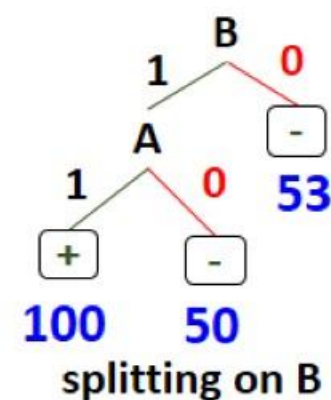
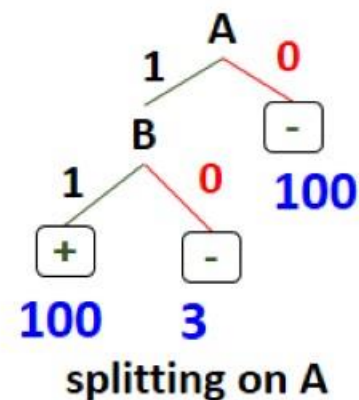
$\langle (A = 0, B = 0), - \rangle$: 50 examples

$\langle (A = 0, B = 1), - \rangle$: 50 examples

$\langle (A = 1, B = 0), - \rangle$: 3 examples

$\langle (A = 1, B = 1), + \rangle$: 100 examples

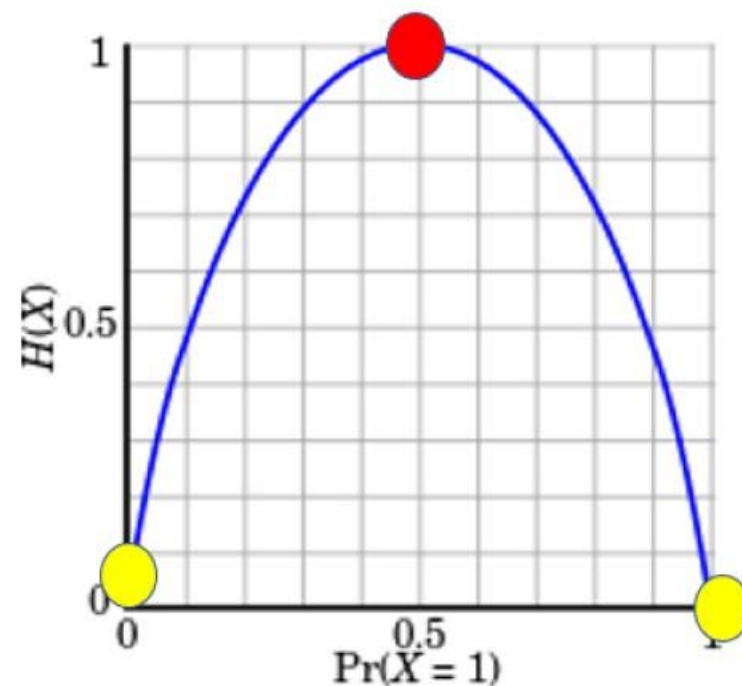
- What should be the first attribute we select?
- Trees look structurally similar;
which attribute should we choose?



- One way to think about it: # of queries required to label a random data point. If we choose A we have less uncertainty about the labels.

- In order to reduce uncertainty, features combinations needs to be measured.
- The term represents the amount of information required to place a new instance in the branch. In other words, which best attribute to select in root nodes and its successors in order to reduce the tree size.
- Methods include:
 - **Entropy**
 - **Gini impurity**

- Entropy (*Level of uncertainty*) of a set of examples, S , relative to a binary classification is:
- $Entropy(S) = -\sum_{s \in S} p(s) \log p(s)$
- $Entropy(S) = -p_+ \log(p_+) - p_- \log(p_-)$
- p_+ is the proportion of positive examples in S and
- p_- is the proportion of negative examples in S

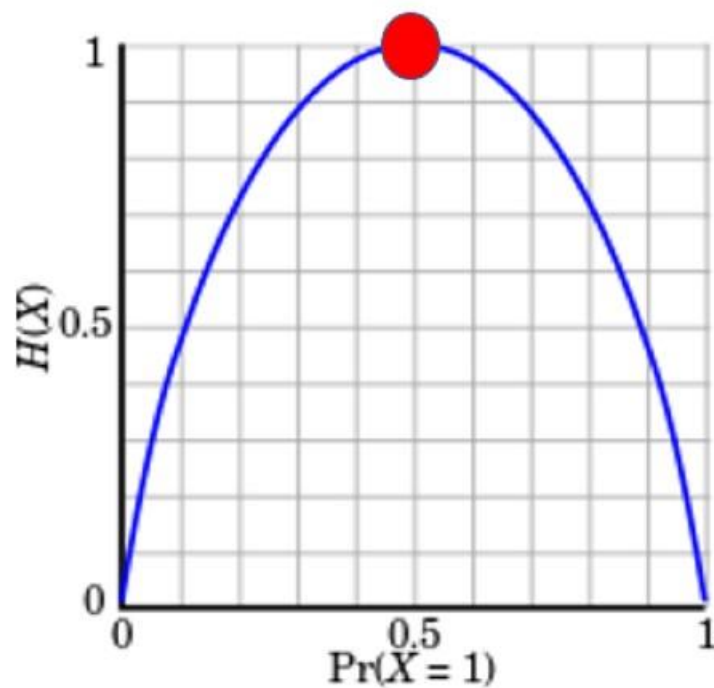


- Entropy (*Level of uncertainty*) of a set of examples, S , relative to a binary classification is:

- $Entropy(S) = -\sum_{s \in S} p(s) \log p(s)$

- Examples in S :

$$\begin{aligned} Entropy(S) &= -\sum_{i=1}^n p(x_i) \log_b p(x_i) \\ &= -\sum_{i=1}^2 \frac{1}{2} \log_2 \frac{1}{2} \\ &= -\sum_{i=1}^2 \frac{1}{2} \cdot (-1) = 1 \end{aligned}$$

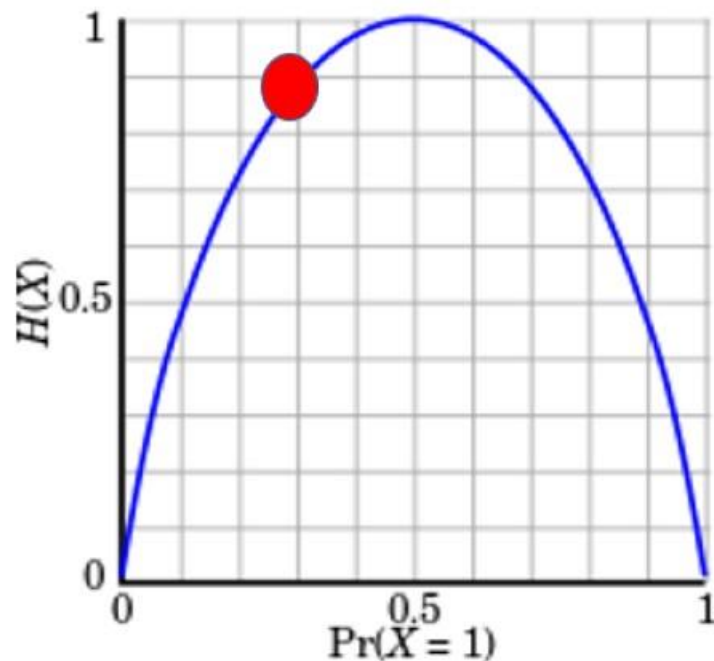


- Entropy (*Level of uncertainty*) of a set of examples, S , relative to a binary classification is:

- $Entropy(S) = -\sum_{s \in S} p(s) \log p(s)$

- Examples in S :

$$\begin{aligned} Entropy(S) &= -p \log_2(p) - q \log_2(q) \\ &= -0.7 \log_2(0.7) - 0.3 \log_2(0.3) \\ &\approx -0.7 \cdot (-0.515) - 0.3 \cdot (-1.737) \\ &= 0.8816 < 1 \end{aligned}$$



- If you use a rolling 6 sided die, with the received probability per each side

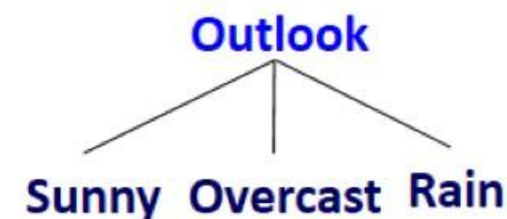
what is the information Entropy :

$$\text{Entropy}(S) = - \sum_{s \in S} p(s) \log p(s)$$

$$\text{Entropy}(S) = 2.58$$

- one = 1 = 8%
- two = 2 = 17%
- three = 3 = 25%
- four = 3 = 25%
- five = 2 = 17%
- six = 1 = 8%

- The information gain of an attribute **a** is the expected reduction in entropy caused by partitioning on this attribute
- $Gain(S, a) = Entropy(S) - \sum_{v \in values(S)} \frac{|S_v|}{|S|} Entropy(S_v)$
- Where:
- **S_v is the subset of S for which attribute a has value v , and the entropy of partitioning the data is calculated by weighing the entropy of each partition by its size relative to the original set.**
- Partitions of low entropy (*imbalanced splits*) lead to high gain
- Go back to check which of the A, B splits is better.



High Entropy – High level of Uncertainty

Low Entropy – No Uncertainty.

Will I play tennis today?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

Outlook: S(unny),
O(vercast),
R(ainy)

Temperature: H(ot),
M(edium),
C(ool)

Humidity: H(igh),
N(ormal),
L(ow)

Wind: S(trong),
W(eak)

Will I play tennis today?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

Calculate current entropy

- $p_+ = \frac{9}{14}$ $p_- = \frac{5}{14}$

- $Entropy(Play) =$

$$-p_+ \log_2(p_+) - p_- \log_2(p_-)$$

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

$$\approx 0.94$$

Information Gain: Outlook

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$Gain(S, a) = Entropy(S) - \sum_{v \in values(S)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Outlook = sunny:

$$p_+ = 2/5 \quad p_- = 3/5 \quad Entropy(O = S) = 0.971$$

Outlook = overcast:

$$p_+ = 4/4 \quad p_- = 0 \quad Entropy(O = O) = 0$$

Outlook = rainy:

$$p_+ = 3/5 \quad p_- = 2/5 \quad Entropy(O = R) = 0.971$$

Expected entropy

$$= \sum_{v \in values(S)} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$= (5/14) \times 0.971 + (4/14) \times 0 + (5/14) \times 0.971 = 0.694$$

$$Information\ gain = 0.940 - 0.694 = 0.246$$

Information Gain: Humidity

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

$$Gain(S, a) = Entropy(S) - \sum_{v \in values(S)} \frac{|S_v|}{|S|} Entropy(S_v)$$

Humidity = high:

$$p_+ = 3/7 \quad p_- = 4/7 \quad Entropy(H = H) = 0.985$$

Humidity = Normal:

$$p_+ = 6/7 \quad p_- = 1/7 \quad Entropy(H = N) = 0.592$$

Expected entropy

$$= \sum_{v \in values(S)} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$= (7/14) \times 0.985 + (7/14) \times 0.592 = 0.7785$$

$$Information\ gain = 0.940 - 0.7785 = 0.151$$

Which feature to split on?

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

Information gain:

Outlook: 0.246

Humidity: 0.151

Wind: 0.048

Temperature: 0.029

→ Split on Outlook

An Illustrative Example

Outlook



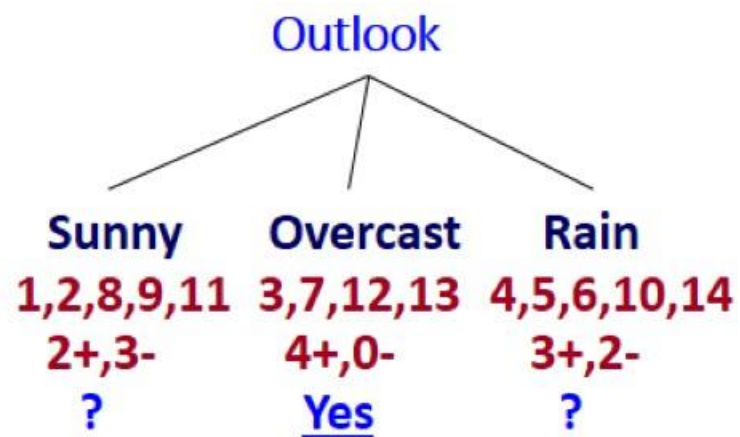
$$\text{Gain}(S, \text{Humidity}) = 0.151$$

$$\text{Gain}(S, \text{Wind}) = 0.048$$

$$\text{Gain}(S, \text{Temperature}) = 0.029$$

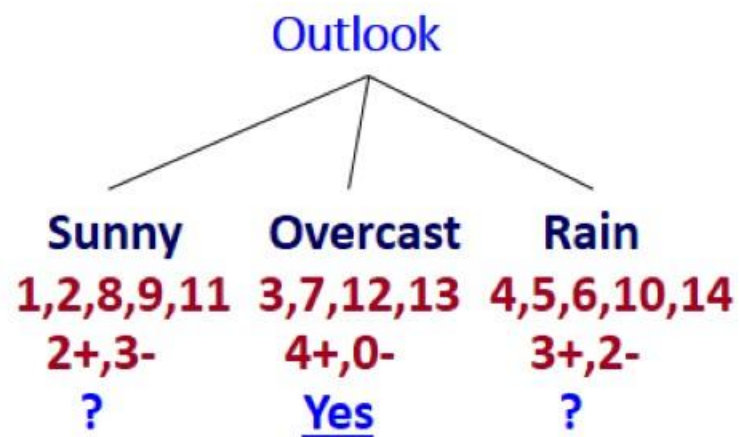
$$\text{Gain}(S, \text{Outlook}) = \mathbf{0.246}$$

An Illustrative Example



	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

An Illustrative Example

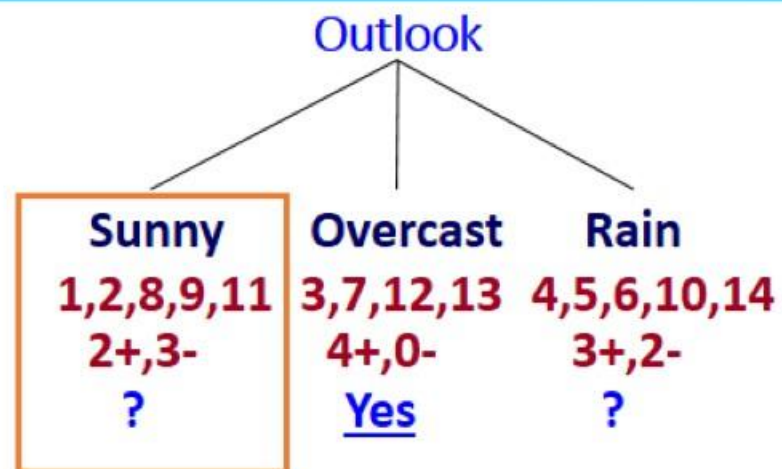


Continue until:

- Every attribute is included in **path**, or,
- All examples in the leaf have same label.

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
3	O	H	H	W	+
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

An Illustrative Example



$$\text{Gain}(S_{\text{sunny}}, \text{Humidity}) = .97 - (3/5) 0 - (2/5) 0 = .97$$

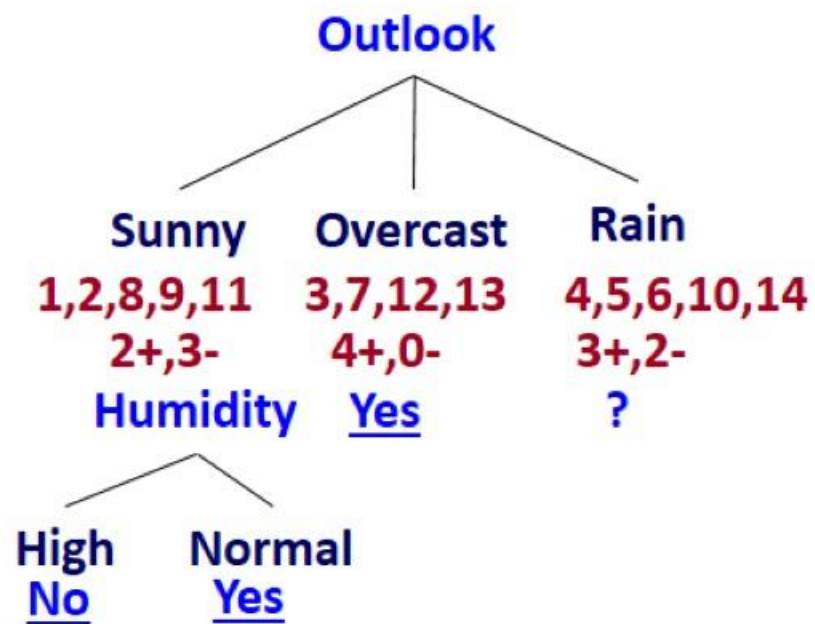
$$\text{Gain}(S_{\text{sunny}}, \text{Temp}) = .97 - 0 - (2/5) 1 = .57$$

$$\text{Gain}(S_{\text{sunny}}, \text{Wind}) = .97 - (2/5) 1 - (3/5) .92 = .02$$

Split on Humidity

	O	T	H	W	Play?
1	S	H	H	W	-
2	S	H	H	S	-
4	R	M	H	W	+
5	R	C	N	W	+
6	R	C	N	S	-
7	O	C	N	S	+
8	S	M	H	W	-
9	S	C	N	W	+
10	R	M	N	W	+
11	S	M	N	S	+
12	O	M	H	S	+
13	O	H	N	W	+
14	R	M	H	S	-

An Illustrative Example



Advantages of Decision trees in classification

1. Cost-effective construction:

- Decision trees offer a budget-friendly option for model construction, requiring minimal resources compared to alternative classification methods.

2. Rapid classification of unknown records:

- Decision trees excel in quickly classifying new, unseen data instances.

3. Simple interpretation, especially for small trees.

4. Comparable accuracy to other classification techniques.

5. Automatic exclusion of irrelevant features:

- Decision trees inherently identify and exclude unimportant features during the classification process.

Disadvantages of Decision trees in classification

1. Prone to overfitting:

- Decision trees are susceptible to overfitting, where the model captures noise or anomalies in the training data instead of learning the underlying patterns. This can lead to poor generalization performance on unseen data.

2. Restriction to parallel decision boundaries:

- Decision boundaries in decision trees are constrained to be parallel to attribute axes, limiting their ability to capture complex relationships between features.

3. Bias towards features with many levels:

- Decision tree models tend to favor splits on features with a large number of levels or categories, potentially leading to biased predictions if certain features dominate the decision-making process.

4. Sensitivity to small changes in training data.

5. Difficulty interpreting large trees.

Real-World Applications of Decision Trees

- **Medical Diagnosis**
- **Credit Risk Assessment and Fraud Detection**
- **Customer Relationship Management (CRM)**
- **Predictive Maintenance**
- **Crop Yield Prediction**
- **Churn Prediction**
- **Sentiment Analysis**

Additional Resources

- What is the C4.5 algorithm and how does it work?

[https://towardsdatascience.com/what-is-the-c4-5-algorithm-and-how-does-it-work-2b971a9e7db0#:~:text=3-,The%20C4.,\(univariate%20or%20multivariate%20predictors\).](https://towardsdatascience.com/what-is-the-c4-5-algorithm-and-how-does-it-work-2b971a9e7db0#:~:text=3-,The%20C4.,(univariate%20or%20multivariate%20predictors).)

- Decision trees : classification and regression trees (CART)

<https://www.niser.ac.in/~smishra/teach/cs460/23cs460/lectures/lec9.pdf>



Thank You