Artificial Intelligence

Classification: Decision Tree

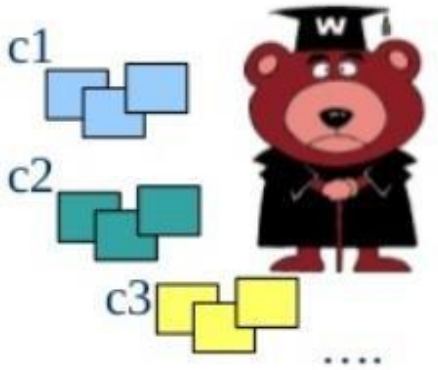
tita@pens.ac.id

Prerequisite

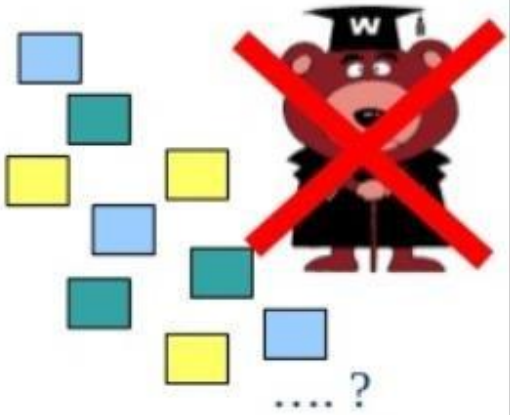
- Data structure (Tree)
- Searching algorithms (greedy algorithm, heuristic search, hill climbing, alpha-beta pruning)
- Logic (OR, AND rules)
- Probability (Dependent and Independent)
- Information Theory (Entropy)

Supervised Vs. Unsupervised

- **Supervised**
  - **knowledge of output -** learning with the presence of an "expert" / teacher
    - data is **labelled** with a class or value
    - **Goal:** predict class or value label
      - e.g. Neural Network, Support Vector Machines, Decision Trees, Bayesian Classifiers ....



- **Unsupervised**
  - **no knowledge of output** class or value
    - data is **unlabelled** or value un-known
    - **Goal:** determine data patterns/groupings
  - Self-guided learning algorithm
    - (internal self-evaluation against some criteria)
    - e.g. k-means, genetic algorithms, clustering approaches ...

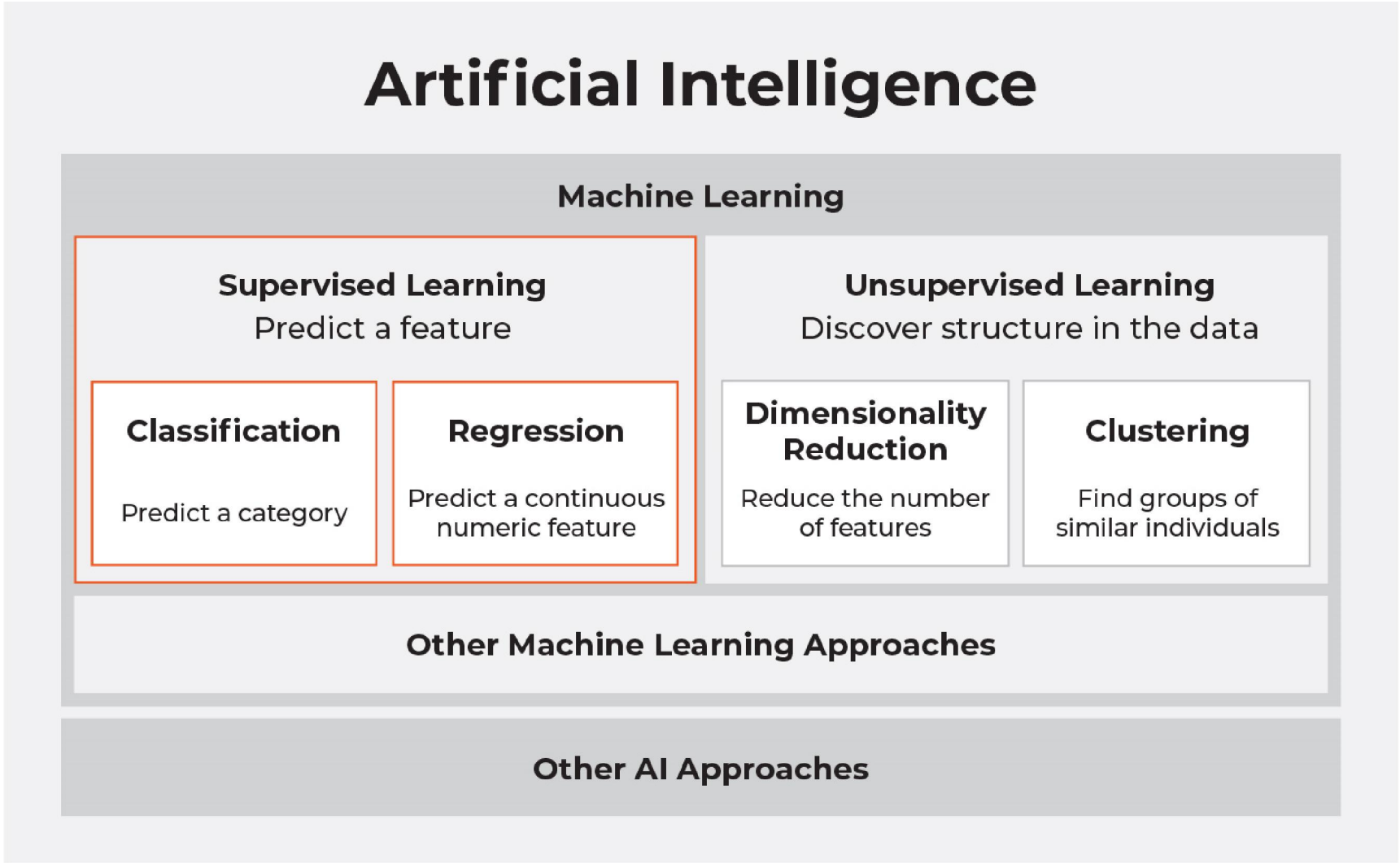


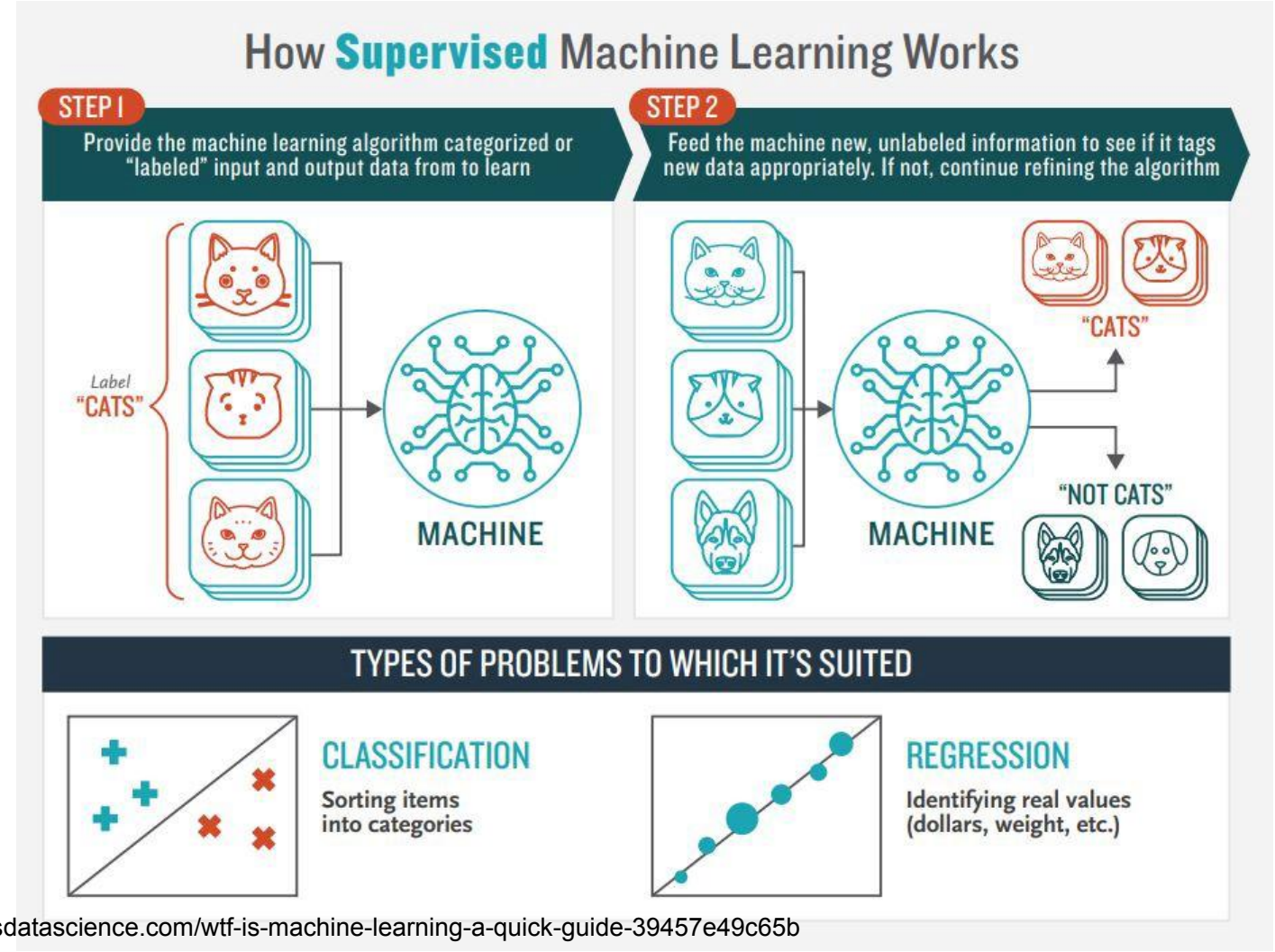
https://www.researchgate.net/figure/Supervised-VS-Unsupervised-Learning_fig6_323108787

Supervised vs. Unsupervised Learning

- Supervised learning (classification)
  - Supervision: The training data (observations, measurements, etc.) are accompanied by **labels** indicating the class of the observations
  - New data is classified based on the training set

- Unsupervised learning (clustering)
  - The class labels of training data is unknown
  - Given a set of measurements, observations, etc. with the aim of establishing the existence of classes or clusters in the data


Artificial Intelligence
Machine Learning
Supervised Learning
Predict a feature
Classification
Predict a category
Regression
Predict a continuous numeric feature
Unsupervised Learning
Discover structure in the data
Dimensionality Reduction
Reduce the number of features
Clustering
Find groups of similar individuals
Other Machine Learning Approaches
Other AI Approaches



https://towardsdatascience.com/wtf-is-machine-learning-a-quick-guide-39457e49c65b

Illustration of classification

https://towardsdatascience.com/machine-learning-classifiers-a5cc4e1b0623

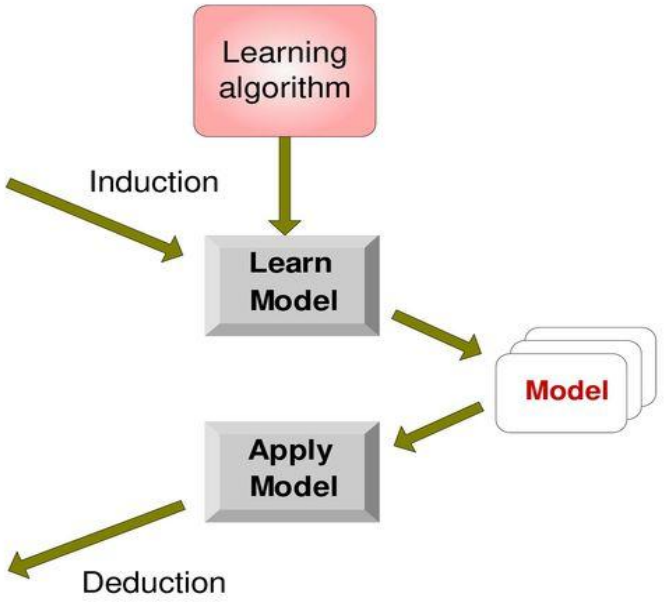
Illustrating Classification Task

Tid	Attrib1	Attrib2	Attrib3	Class
1	Yes	Large	125K	No
2	No	Medium	100K	No
3	No	Small	70K	No
4	Yes	Medium	120K	No
5	No	Large	95K	Yes
6	No	Medium	60K	No
7	Yes	Large	220K	No
8	No	Small	85K	Yes
9	No	Medium	75K	No
10	No	Small	90K	Yes

Training Set

Tid	Attrib1	Attrib2	Attrib3	Class
11	No	Small	55K	?
12	Yes	Medium	80K	?
13	Yes	Large	110K	?
14	No	Small	95K	?
15	No	Large	67K	?

Test Set



Klasifikasi

- Klasifikasi adalah proses untuk menemukan model atau fungsi yang membedakan kelas data.
- Tujuannya untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui.
- Model bisa berupa aturan “jika-maka”, berupa decision tree, formula matematis atau neural network.
- Proses klasifikasi biasanya dibagi menjadi dua fase : learning dan test.
  - Pada fase learning, sebagian data yang telah diketahui kelas datanya diumpankan untuk membentuk model perkiraan.
  - Pada fase test model yang sudah terbentuk diuji dengan sebagian data lainnya untuk mengetahui akurasi dari model tsb.
  - Bila akurasinya mencukupi model ini dapat dipakai untuk prediksi kelas data yang belum diketahui.
- Klasifikasi dicirikan dengan data training mempunyai **label**, berdasarkan label ini proses klasifikasi memperoleh pola attribut dari suatu data.

Klasifikasi

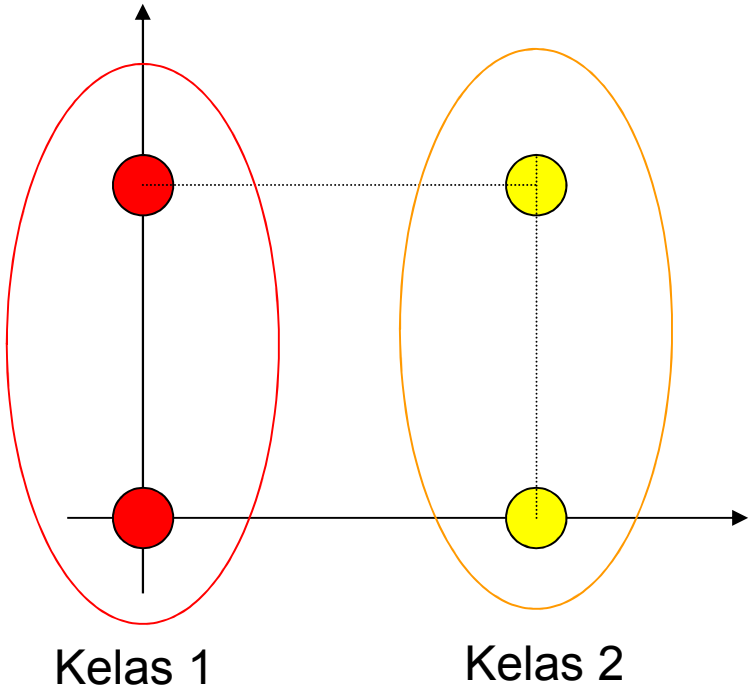
- Typical applications:
  - Credit/loan approval
  - Medical diagnosis: if a tumor is cancerous or benign
  - Fraud detection: if a transaction is fraudulent
  - Web page categorization: which category it is

Decision Tree

- Decision Tree adalah salah satu metode klasifikasi yang paling popular karena sederhana dan mudah diinterpretasikan oleh manusia.
- It is enhanced greedy search algorithm that implement heuristic function using probability as comparison values, but does not implement backtracking (because it is greedy!).

Proses Klasifikasi Dalam Data Mining

X	Y	Kelas
0	0	1
0	1	1
1	0	2
1	1	2

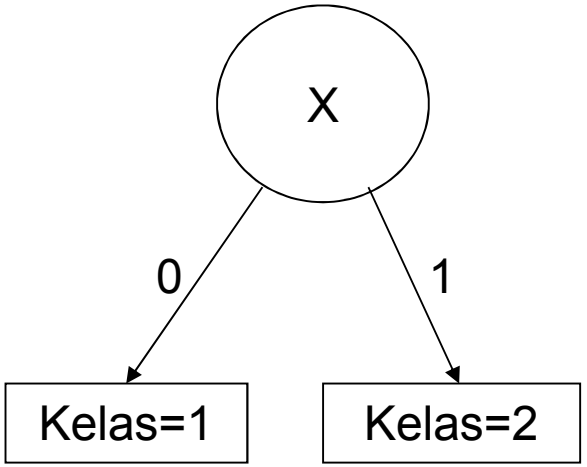


Klasifikasi Dapat Direpresentasikan Dalam Bentuk Tree

Konsep Decision Tree

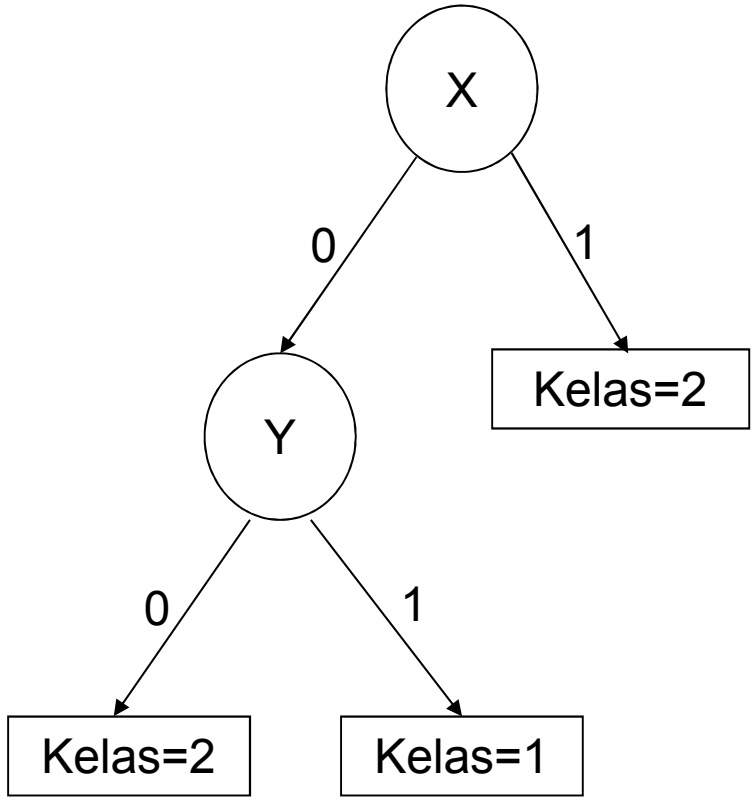
Mengubah data menjadi pohon keputusan (*decision tree*) dan aturan-aturan keputusan (*rule*)

X	Y	Kelas
0	0	1
0	1	1
1	0	2
1	1	2



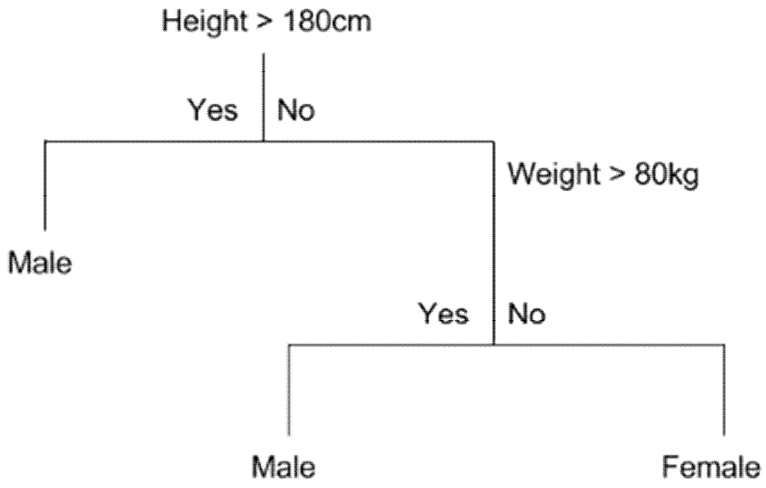
Klasifikasi Dapat Direpresentasikan Dalam Bentuk Tree

X	Y	Z	Kelas
0	0	0	2
0	0	1	2
0	1	0	1
0	1	1	1
1	0	0	2
1	0	1	2
1	1	0	2
1	1	1	2



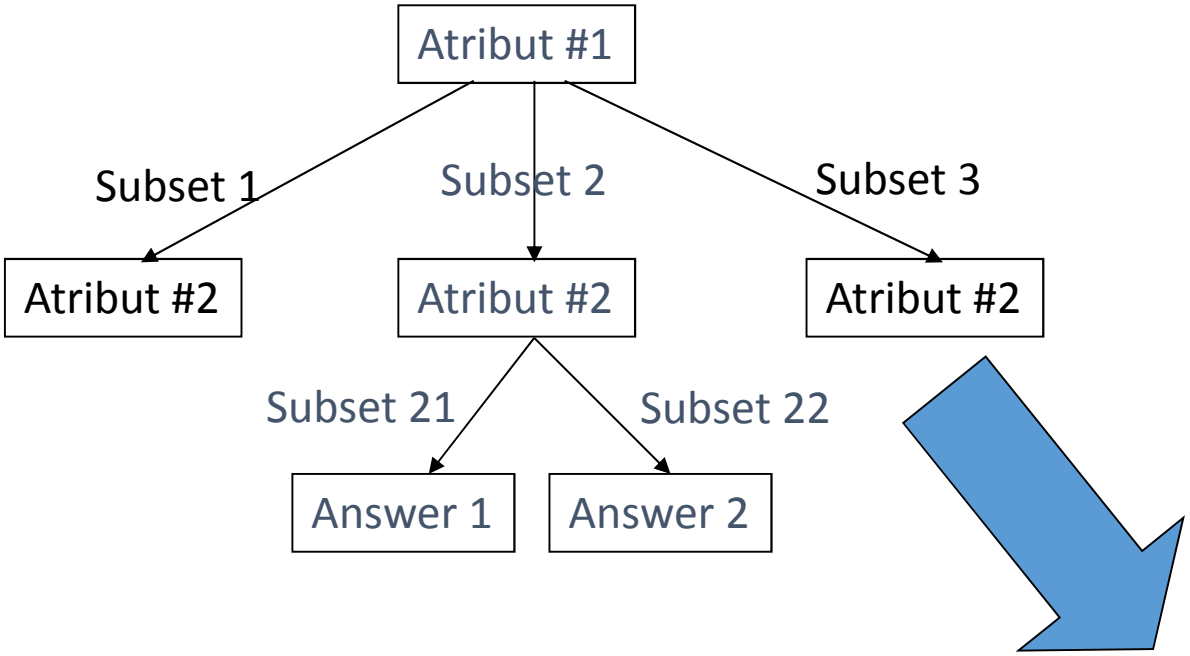
Struktur Decision Tree

- A decision tree is a classification and prediction tool having a tree-like structure.
- Decision Tree dibentuk dari 3 tipe simpul:
  - Root Node
  - Internal Node
  - Leaf Node
- Root Node adalah titik awal dari suatu decision tree



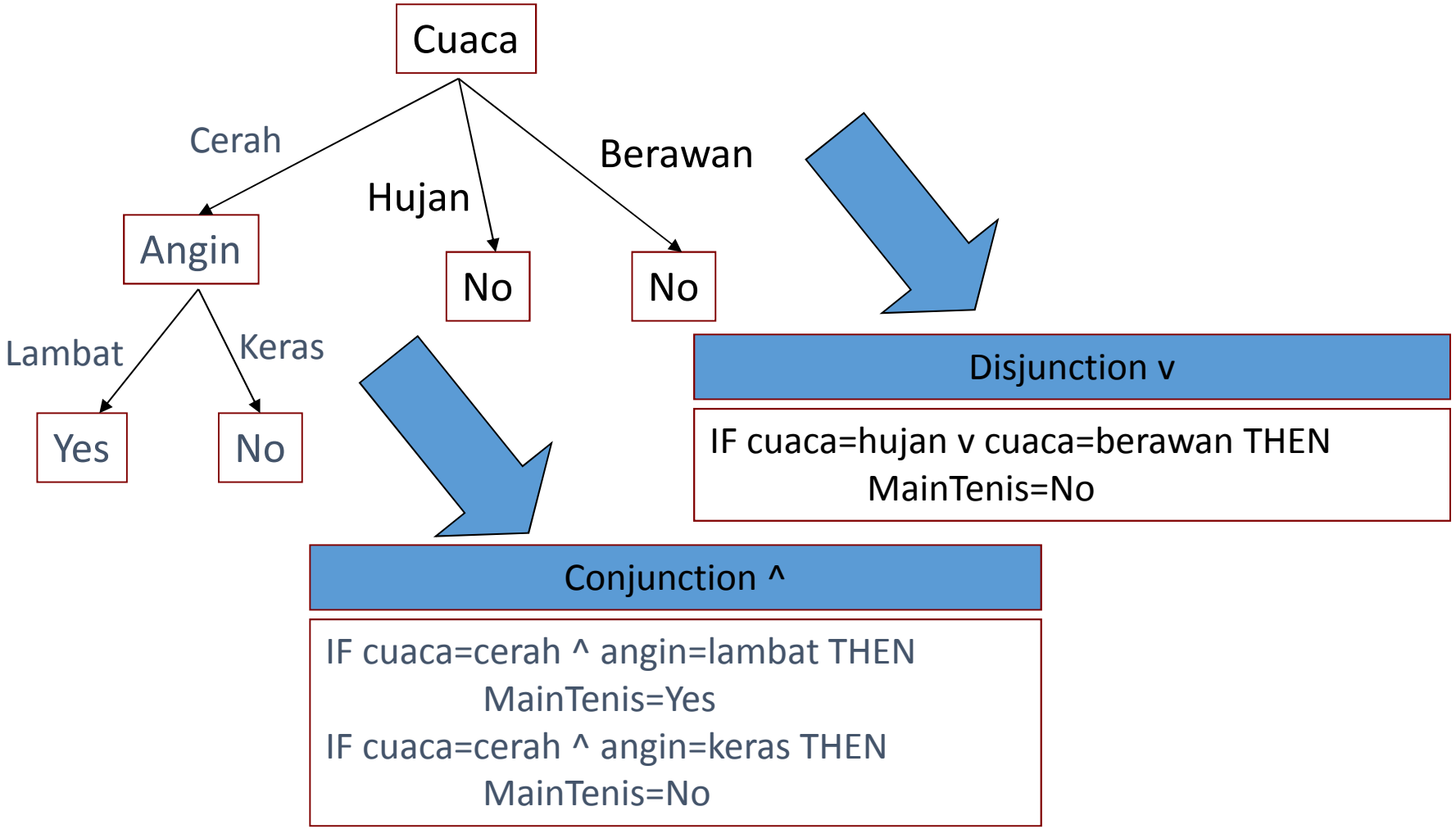
  - Setiap Internal Node berhubungan dengan suatu pertanyaan atau pengujian.
    - each internal node denotes a test on an attribute
    - each branch represents an outcome of the test
  - Leaf Node memuat suatu keputusan akhir atau kelas target untuk suatu pohon keputusan.
    - each leaf node (terminal node) holds a class label.

Mengubah Tree Menjadi Rules



If atribut#1=subset2 ^ atribut#2=subset21
then answer=answer1
If atribut#1=subset2 ^ atribut#2=subset22
then answer=answer2

Conjunction & Disjunction

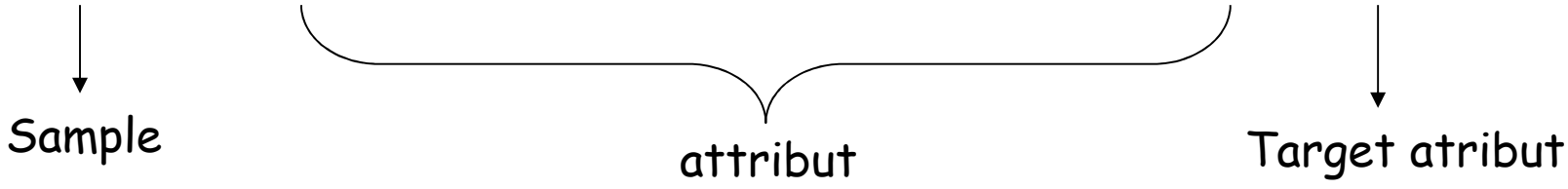


Konsep Data Dalam Decision Tree

- Data dinyatakan dalam bentuk tabel dengan **atribut** dan **record**.
- **Atribut** menyatakan suatu parameter yang dibuat sebagai kriteria dalam pembentukan tree.
  - Misalkan untuk menentukan bermain tenis, kriteria yang diperhatikan adalah cuaca, angin dan temperatur.
  - Salah satu atribut merupakan atribut yang menyatakan data solusi per-item data yang disebut dengan **target atribut**.
- Atribut memiliki nilai-nilai. Misalkan atribut cuaca mempunyai nilai-nilai berupa cerah, berawan dan hujan.

Konsep Data Dalam Decision Tree

Nama	Cuaca	Angin	Temperatur	Main
Ali	cerah	keras	panas	tidak
Budi	cerah	lambat	panas	ya
Heri	berawan	keras	sedang	tidak
Irma	hujan	keras	dingin	tidak
Diman	cerah	lambat	dingin	ya



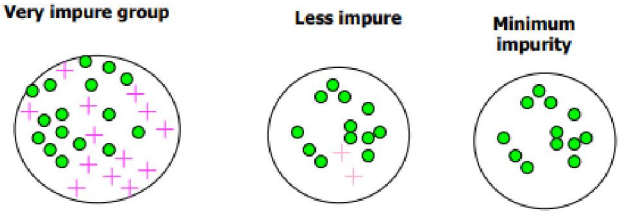
Terms related to decision trees.

Entropy

- A common way to measure impurity
- In machine learning, entropy is a measure of the randomness in the information being processed.
- Measures the level of impurity in a group of examples
- The higher the entropy, the harder it is to draw any conclusions from that information.

$$H(x) = -\ p_+\ log_2\ p_+ - p_-\ log_2\ p_-$$

$$H(X) = -\sum_{i=1}^{n} P(X = i)\log_2 P(X = i)$$



- **Information Gain**
- It is the amount of information gained about a random variable or signal from observing another random variable.
- It can be considered as the difference between the entropy of parent node and weighted average entropy of child nodes.

$$G(x, y) = H(x) - \sum_{i \in value(y)} \frac{|\Delta y_i|}{|\Delta y|}\ H(y_i)$$

Gini Impurity

- It is a measure of how often a randomly chosen element from the set would be incorrectly labeled if it was randomly labeled according to the distribution of labels in the subset.
- Gini impurity is **lower bounded by 0**, with 0 occurring if the data set contains only one class

$$Gini(E) = 1 - \sum_{j=1}^{c} p_j^2$$

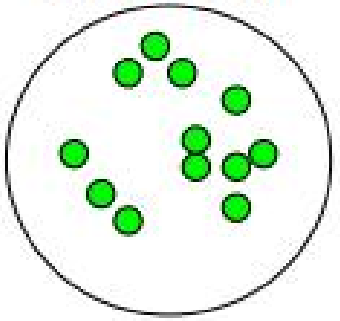
2-Class Cases:

Entropy $H(x) = -\sum_{i=1}^{n} P(x=i)\log_2 P(x=i)$

- What is the entropy of a group in which all examples belong to the same class?
  - entropy = $-1 \log_2 1 = 0$

not a good training set for learning

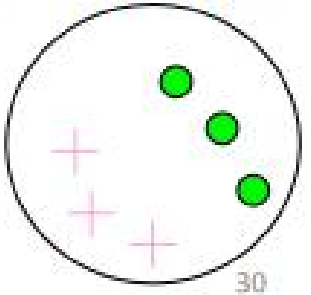
Minimum impurity



- What is the entropy of a group with 50% in either class?
  - entropy = $-0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$

good training set for learning

Maximum impurity



Algorithms to build a decision tree:

- **CART**
  - Classification and Regression Trees.
  - This makes use of Gini impurity as the metric.

- **ID3**
  - Iterative Dichotomiser 3.
  - This uses entropy and information gain as metric.

Classification using the ID3 algorithm

Algoritma untuk induksi Decision Tree

- Basic algorithm (a greedy algorithm):
  - Tree dibangun dengan metode rekursif top down divide and conquer.
  - Dimulai dari menganalisa seluruh data pelatihan, dimulai dari membetuk simpul root Tree.
  - Atribut-atribut ini dipartisi secara rekursif berdasarkan atribut yang terpilih.
  - Atribut-atribut uji dipilih berdasarkan heuristik atau pengukuran statistik (missal: information gain).
  - Partisi berakhir jika semua atribut sudah menjadi node dalam Tree.

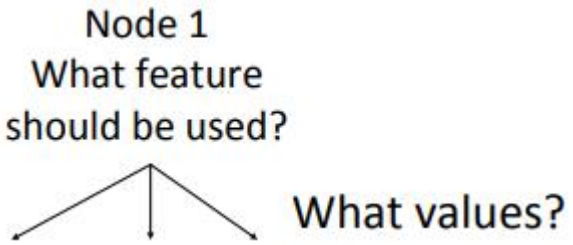
- Catatan:
  - Atribut-atribut (berupa kolom dalam tabel) berada dalam suatu kategori (jika bernilai continue, maka harus dikonversi ke nilai diskrit).

How do you pick the starting test condition?

Entropy-Based Automatic Decision Tree Construction

- The answer to this question lies in the values of **Entropy** and **Information Gain**.
- **Entropy:** Entropy in Decision Tree stands for homogeneity.
  - If the data is completely homogenous, the entropy is 0;
  - Otherwise, if the data is divided (50-50%) entropy is 1.
- **Information Gain:** Information Gain is the decrease/increase in Entropy value when the node is split.
- Based on the computed values of Entropy and Information Gain, we choose the best attribute at any particular step.

Quinlan suggested information gain in his ID3 system and later the gain ratio, both based on entropy.



Entropy

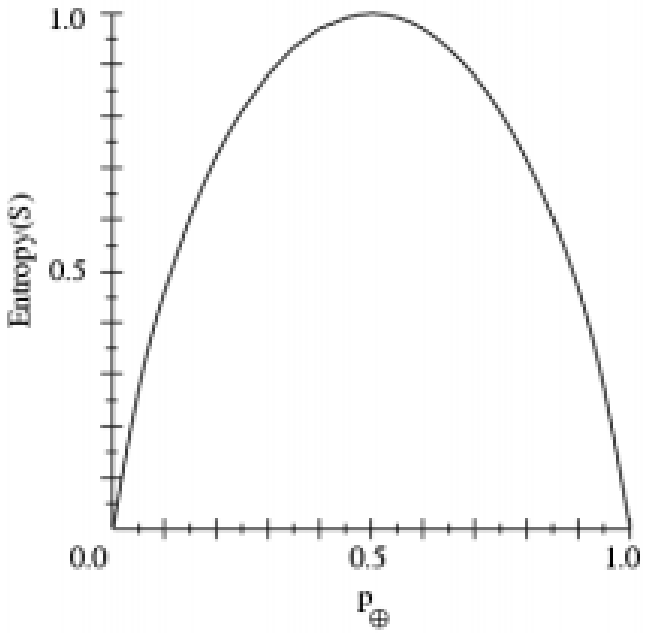
See page 15 & 16, for entropy

- S adalah ruang (data) sample yang digunakan untuk training.
- P+ adalah jumlah yang bersolusi positif (mendukung) pada data sample untuk kriteria tertentu.
- P- adalah jumlah yang bersolusi negatif (tidak mendukung) pada data sample untuk kriteria tertentu.
- Besarnya Entropy pada ruang sample S didefinisikan dengan:

$$\textbf{Entropy(S)=H(x)= } -p_+\log_2(p_+) - p_-\log_2(p_-)$$

Where:

- **S** is a sample of training examples
- **P+** is the porpotion of positive examples in S
- **P-** is the porpotion of negative examples in S
- Centropy measures the impurity of S



Semakin kecil nilai Entropy maka semakin baik untuk digunakan dalam mengekstraksi suatu kelas.

Dasar menghitung nilai log

- $Log_2(1) = 0$
- $Log_2(2) = 1$
- $Log_2(4) = 2$
- $Log_2(\frac{1}{2}) =$ -1
- $Log_2\left(\frac{1}{4}\right) = -2$
- $\left(\frac{1}{2}\right) Log_2\left(\frac{1}{2}\right) = \left(\frac{1}{2}\right)(-1) = -\frac{1}{2}$

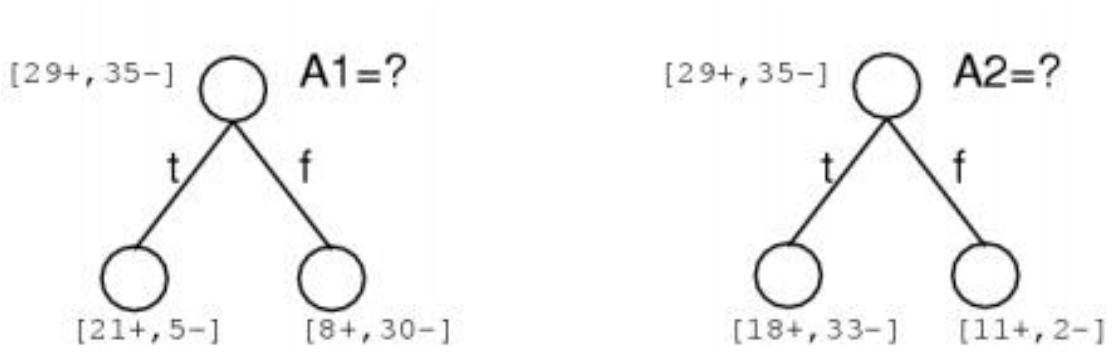
Information Gain

- We want to determine which attribute in a given set of training feature vectors is most useful for discriminating between the classes to be learned.
- Information gain tells us how important a given attribute of the feature vectors is.
- We will use it to decide the ordering of attributes in the nodes of a decision tree.

Information Gain

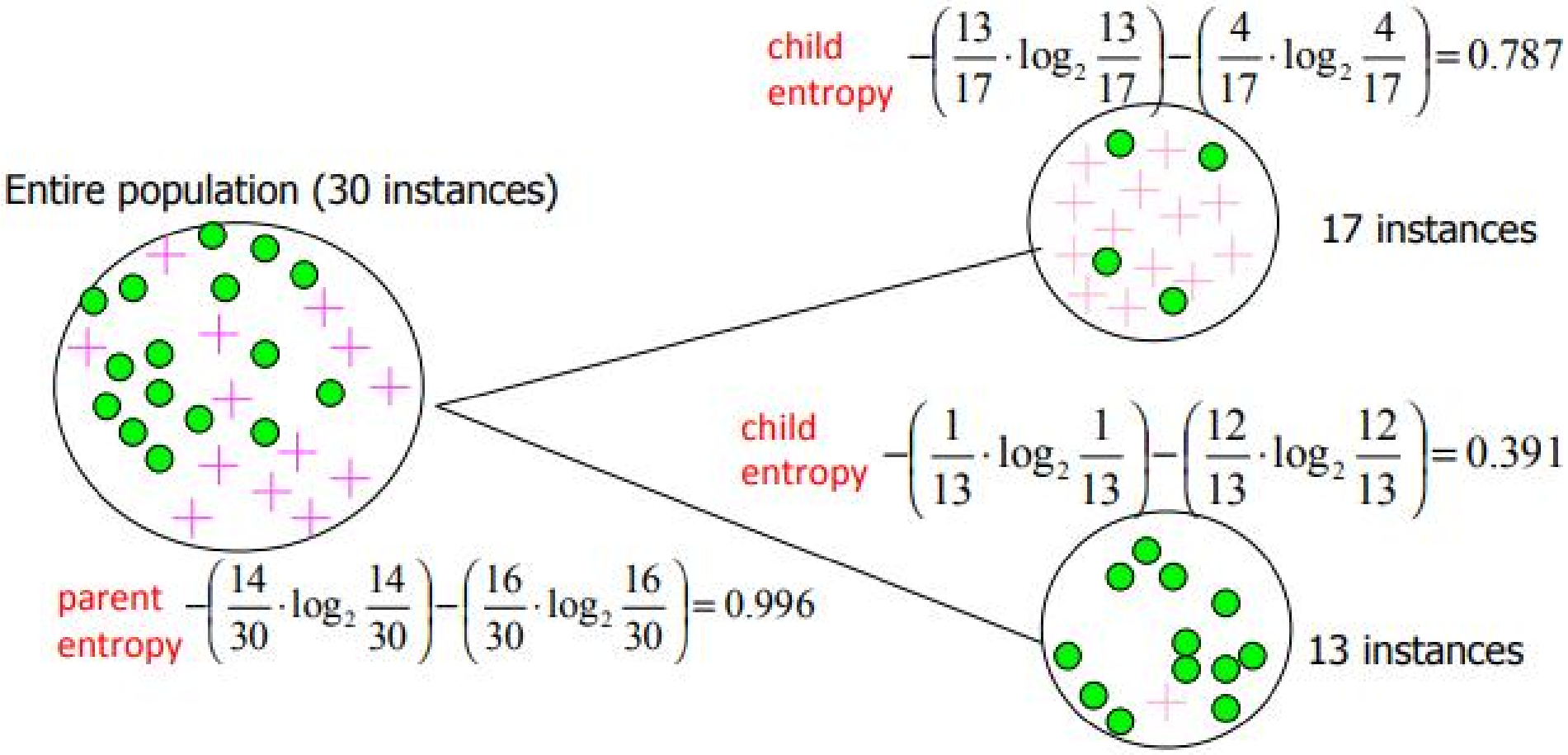
- Information Gain is the mutual information between input attribute A and target variable Y.
- Information Gain is the expected reduction in entropy of target variable Y for data sample S, due to sorting on variable A

$$Gain(S, A) = I_S(A, Y) = H_S(Y) - H_S(Y|A)$$



Calculating Information Gain

- Information Gain = entropy(parent) – [average entropy(children)]



(Weighted) Average Entropy of Children = $\left(\frac{17}{30}\cdot 0.787\right)+\left(\frac{13}{30}\cdot 0.391\right)=0.615$

Information Gain= 0.996 - 0.615 = 0.38

Contoh Pembentukan Tree (1)

Pembentukan decision tree untuk kasus yang berhubungan dengan data cuaca untuk penentuan apakah seseorang bermain tennis atau tidak

Predictors/features/attributes | Response

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

- There are four independent variables to determine the dependent variable: Outlook, Temperature, Humidity, and Wind.
- The dependent variable is whether to play tennis or not.
- What prediction would we make for <Outlook=Sunny, Temperature=Hot, Humidity=High, Wind=Weak> ?

Mengubah bentuk data (tabel) menjadi model tree.

Dalam Modul ini menggunakan algoritma ID3.

Contoh Pembentukan Tree (2)

- Berdasarkan data tersebut (disebut sebagai data training):
  - atribut kategori adalah atribut yang berisi penentuan apakah seseorang bermain tenis (Yes) atau tidak (No).
  - atribut non kategori nya adalah:

Atribut	Nilai yang dimungkinkan
Outlook	Sunny, Overcast, Rain
Temperature	Hot, Mild, Cool
Humidity	High, Normal
Windy	True, False

Contoh Pembentukan Tree (3)

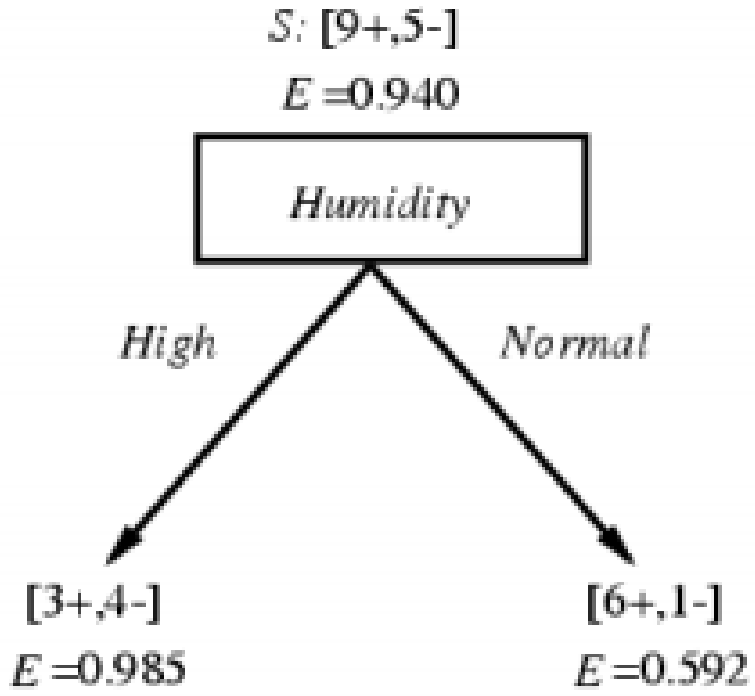
Langkah pertama adalah menentukan atribut yang terpilih menjadi Root Node:

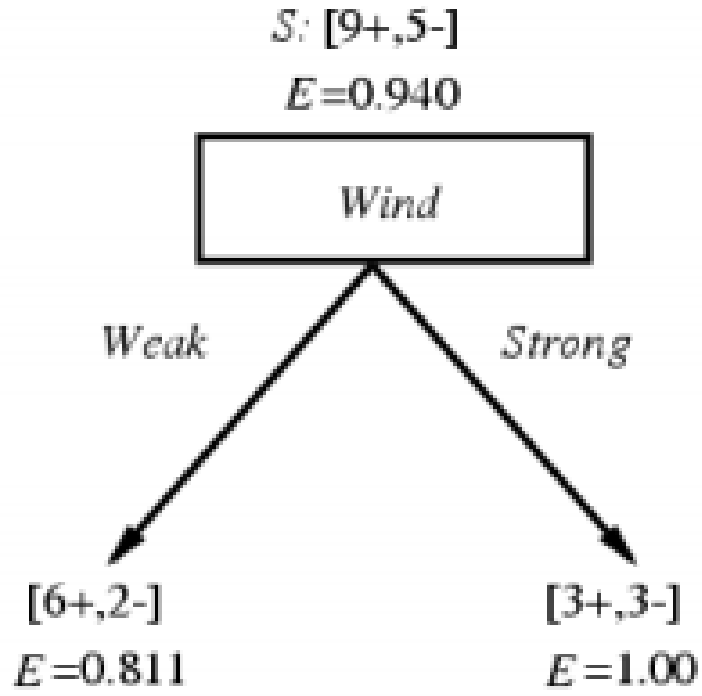
- Menentukan node terpilih:
  - Gunakan nilai entropy dari setiap kriteria dengan data sample yang ditentukan.
  - Yang menjadi node terpilih adalah kriteria dengan nilai entropy **yang paling kecil**. Yaitu yang memiliki Information gain yang paling besar.

- Keterangan:
  - Pernyataan YES (+)
  - Pernyataan NO (-)

Selecting an attribute as a node (Trial)

- Which attribute is the best classifier?





Gain (S, Humidity)

= .940 - (7/14).985 - (7/14).592
= .151

Gain (S, Wind)

= .940 - (8/14).811 - (6/14)1.0
= .048

Contoh Pembentukan Tree (4)

Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$

Outlook

- B1: Sunny
  - 2 play (+)
  - 3 not play (-)
- B2: Overcast
  - 4 play (+)
- B3: Rain
  - 2 play (+)
  - 3 not play (-)
- Average entropy untuk Outlook

$$= \frac{5}{14}\left[-\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right)\right] + \frac{4}{14}\left[-\frac{4}{4}\log_2\left(\frac{4}{4}\right)\right] + \frac{5}{14}\left[-\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right)\right]$$

$$= 0.686$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

		play		
		yes	no	total
Outlook	sunny	3	2	5
	overcast	4	0	4
	rainy	2	3	5
				14

Contoh Pembentukan Tree (5)

Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$

Temperature

- B1: Hot
  - 2 play (+)
  - 2 not play (-)
- B2: Mild
  - 4 play (+)
  - 2 not play (-)
- B3: Cool
  - 3 play (+)
  - 1 not play (-)
- Average entropy untuk Temperature

$$= \frac{4}{14}\left[-\frac{2}{4}\log_2\left(\frac{2}{4}\right) - \frac{3}{4}\log_2\left(\frac{2}{4}\right)\right] + \frac{6}{14}\left[-\frac{4}{6}\log_2\left(\frac{4}{6}\right) - \frac{2}{6}\log_2\left(\frac{2}{6}\right)\right] + \frac{4}{14}\left[-\frac{3}{4}\log_2\left(\frac{3}{4}\right) - \frac{1}{4}\log_2\left(\frac{1}{4}\right)\right]$$

$$= 0.82$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

Contoh Pembentukan Tree (6)

Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$

Humidity

- B1: High
  - 3 play (+)
  - 4 not play (-)
- B2: Normal
  - 6 play (+)
  - 1 not play (-)
- Average entropy untuk Humidity

$$= \frac{7}{14}\left[-\frac{3}{7}\log_2\left(\frac{3}{7}\right) - \frac{4}{7}\log_2\left(\frac{4}{7}\right)\right] + \frac{7}{14}\left[-\frac{6}{7}\log_2\left(\frac{6}{7}\right) - \frac{1}{7}\log_2\left(\frac{1}{7}\right)\right]$$

$$= 0.785$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

Contoh Pembentukan Tree (7)

Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$

Wind

- B1: Weak
  - 6 play (+)
  - 2 not play (-)
- B2: Strong
  - 3 play (+)
  - 3 not play (-)
- Average entropy untuk Wind

$$= \frac{8}{14}\left[-\frac{6}{8}\log_2\left(\frac{6}{8}\right) - \frac{2}{8}\log_2\left(\frac{2}{8}\right)\right] + \frac{6}{14}\left[-\frac{3}{6}\log_2\left(\frac{3}{6}\right) - \frac{3}{6}\log_2\left(\frac{3}{6}\right)\right]$$

$$= 0.8922$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

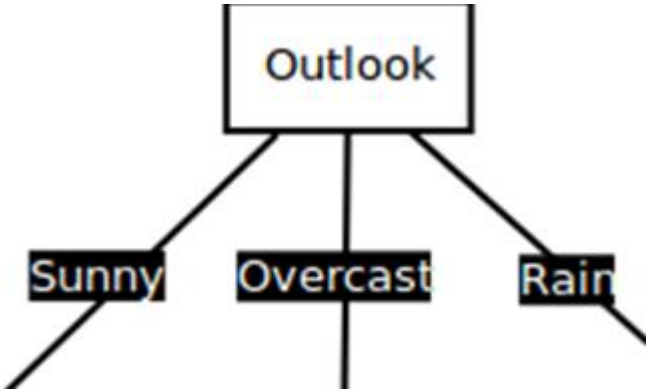
TABLE 3.2
Training examples for the target concept *PlayTennis*.

Contoh Pembentukan Tree (8)

Hasil perhitungan semua average entropy.

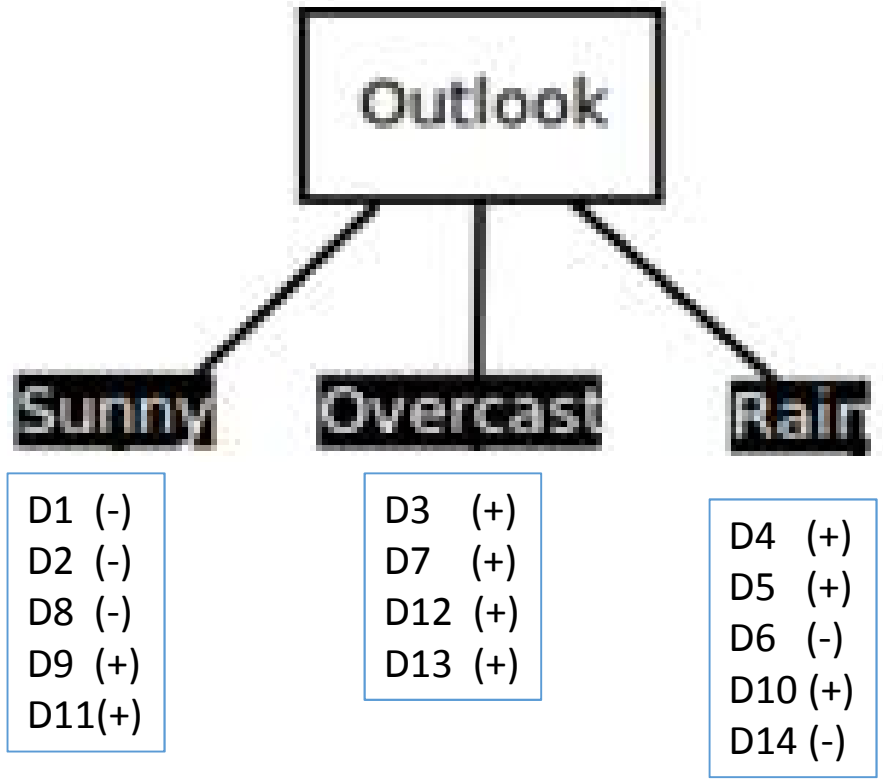
Atribut	Average Entropy
Outlook	**0.686**
Temperature	0.820
Humidity	0.785
Windy	0.892

Nilai entropy terkecil



Atribut Outlook terpilih sebagai node awal (root) karena memiliki average entropy terkecil

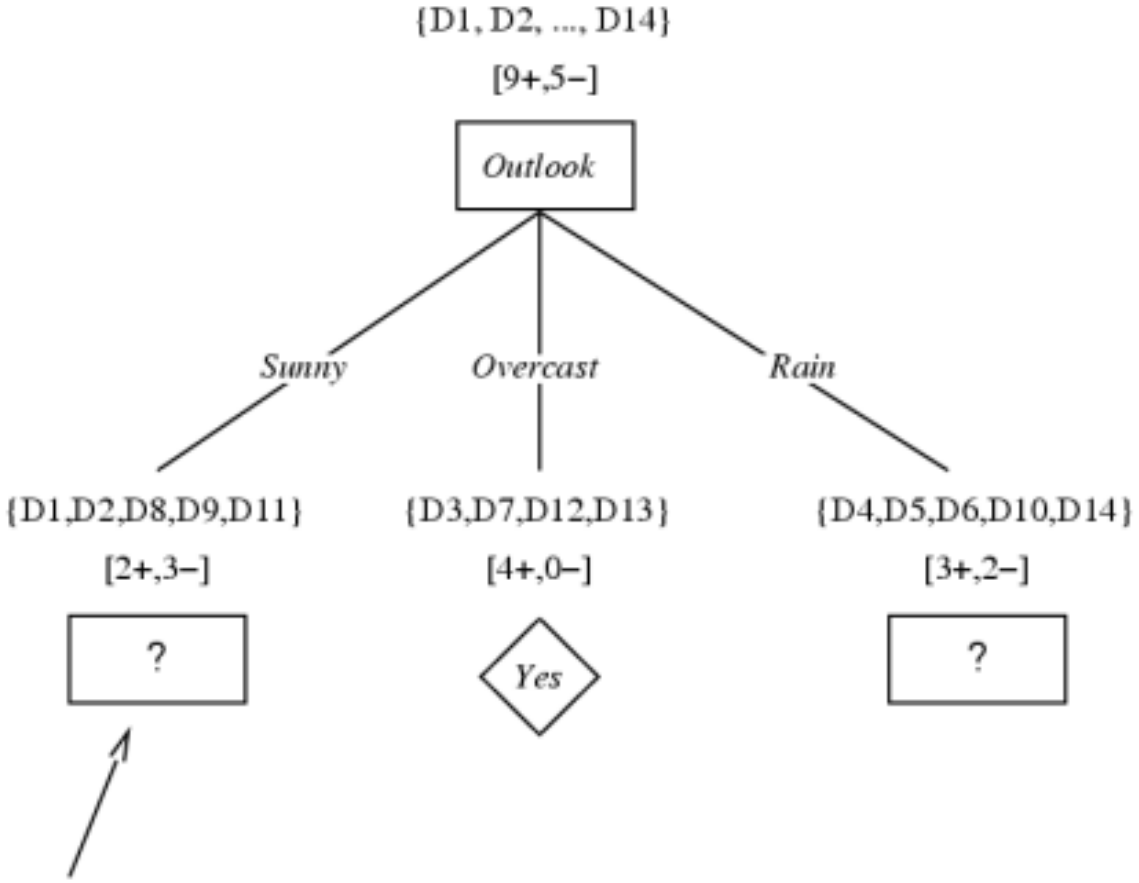
Contoh Pembentukan Tree (9)



Langkah selanjutnya:

- Penyusunan leaf node dari ROOT pada Tree dipilih pada bagian atribut yang memiliki nilai + dan -.
- Terdapat dua buah atribut yang memiliki nilai + dan -, yaitu Outlook=Sunny dan Outlook=Rain, maka kedua atribut tersebut pasti memiliki leaf node.
- Untuk menyusunan leaf node dilakukan satu-persatu.

Perhitungan Information Gain



S_{sunny} = {D1,D2,D8,D9,D11}

Gain (S_{sunny}, *Humidity*) = .970 − (3/5) 0.0 − (2/5) 0.0 = .970

Gain (S_{sunny}, *Temperature*) = .970 − (2/5) 0.0 − (2/5) 1.0 − (1/5) 0.0 = .570

Gain (S_{sunny}, *Wind*) = .970 − (2/5) 1.0 − (3/5) .918 = .019

Contoh Pembentukan Tree (10)

- Data training untuk Outlook = Sunny

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

Contoh Pembentukan Tree (11)

Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$

Temperature

- B1: Hot
  - 2 play (-)
- B2: Mild
  - 1 play (+)
  - 1 not play (-)
- B3: Cool
  - 1 play (+)
- Average entropy untuk Temperature

$$= \frac{2}{5}\left[-\frac{2}{2}\log_2\left(\frac{2}{2}\right)\right] + \frac{2}{5}\left[-\frac{1}{2}\log_2\left(\frac{1}{2}\right) - \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right] + \frac{1}{5}\left[-\frac{1}{1}\log_2\left(\frac{1}{1}\right)\right]$$

$$= 0.4$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

Contoh Pembentukan Tree (12)

Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$

Humidity

- B1: High
  - 3 not play (-)
- B2: Normal
  - 2 play (+)
- Average entropy untuk Humidity

$$= \frac{3}{5}\left[-\frac{3}{3}\log_2\left(\frac{3}{3}\right)\right] + \frac{2}{5}\left[-\frac{2}{2}\log_2\left(\frac{2}{2}\right)\right]$$

$$= 0$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

Contoh Pembentukan Tree (13)

Entropy(S) = $-p_+ \log_2 p_+ - p_- \log_2 p_-$

Wind

- B1: Weak
  - 1 play (+)
  - 2 not play (-)
- B2: Strong
  - 1 play (+)
  - 1 not play (-)
- Average entropy untuk Wind

$$= \frac{3}{5}\left[-\frac{2}{3}\log_2\left(\frac{2}{3}\right) - \frac{1}{3}\log_2\left(\frac{1}{3}\right)\right] + \frac{2}{5}\left[-\frac{1}{2}\log_2\left(\frac{1}{2}\right) - \frac{1}{2}\log_2\left(\frac{1}{2}\right)\right]$$

$$= 0.317$$

Day	*Outlook*	*Temperature*	*Humidity*	*Wind*	*PlayTennis*
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

TABLE 3.2
Training examples for the target concept *PlayTennis*.

Contoh Pembentukan Tree (14)

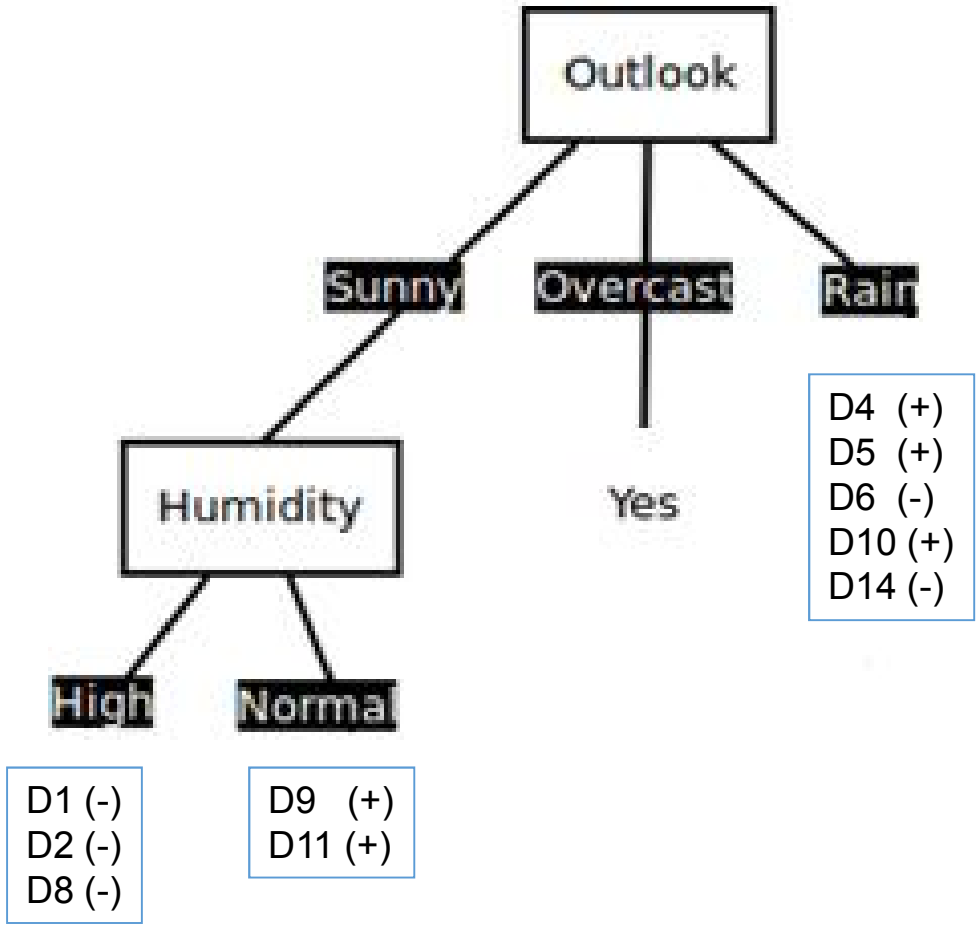
Hasil perhitungan average entropy untuk Outlook = Sunny.

Atribut	Average Entropy
Temperature	0.400
Humidity	**0.000**
Windy	0.317

Nilai average entropy terkecil

Atribut Humidity terpilih sebagai leaf dari instance Sunny

Contoh Pembentukan Tree (15)

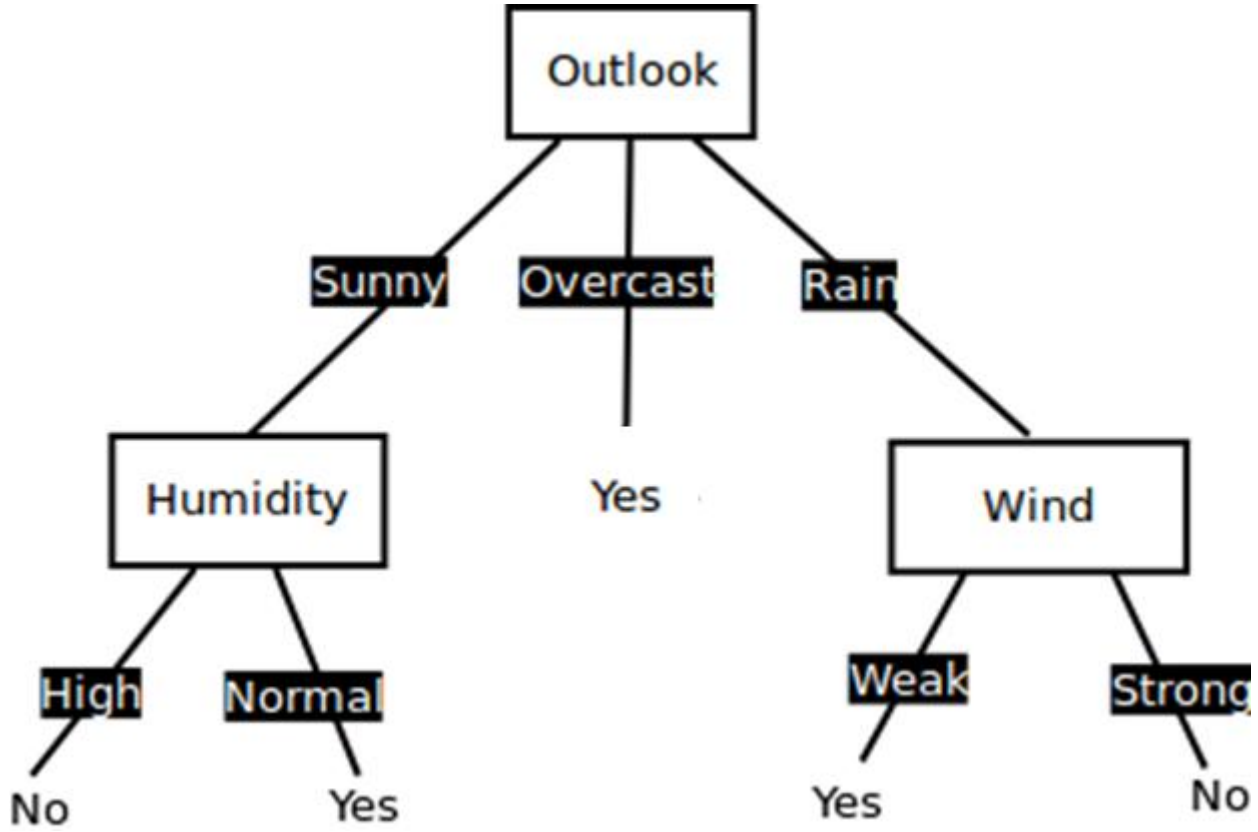


Langkah selanjutnya:

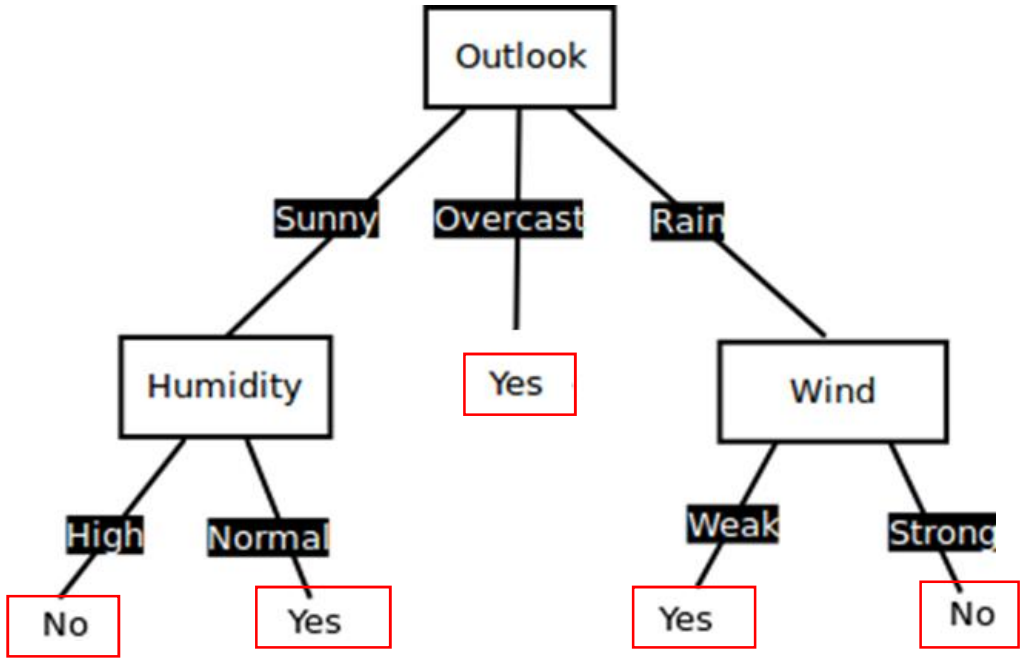
- Pada Tree, hanya Outlook=Rain yang memiliki nilai + dan –.
- Oleh karena itu dilakukan lagi perhitungan entropi lagi untuk menentukan leaf node.
- Caranya sama dengan cara sebelumnya, yaitu dengan menghitung nilai entropi.

Contoh Pembentukan Tree (16)

Hasil akhir Tree



Mengubah Tree menjadi Rule



R1: IF Outlooks=sunny ^ Humidity=high
THEN PlayTennis =NO
R2: IF Outlooks=rain ^ Windy=strong
THEN PlayTennis=NO
R3: IF Outlooks=sunny ^ Humidity=normal
THEN PlayTennis =YES
R4: IF Outlooks=rain ^ Windy=weak
THEN PlayTennis =YES
R5: IF Outlooks=overcast
THEN PlayTennis =YES

Advantages

- Decision trees generate understandable rules.
- Decision trees perform classification without requiring much computation.
- Decision trees are capable of handling both continuous and categorical variables.
- Decision trees provide a clear indication of which fields are most important for prediction or classification.

Disadvantages

- Decision trees are less appropriate for estimation tasks where the goal is to predict the value of a continuous attribute.
- Decision trees are prone to errors in classification problems with many class and a relatively small number of training examples.
- Decision trees can be computationally expensive to train. The process of growing a decision tree is computationally expensive. At each node, each candidate splitting field must be sorted before its best split can be found. In some algorithms, combinations of fields are used and a search must be made for optimal combining weights. Pruning algorithms can also be expensive since many candidate sub-trees must be formed and compared.

References

- Arun Mohan, Decision Tree Algorithm With Hands-On Example, 2019, https://medium.com/datadriveninvestor/decision-tree-algorithm-with-hands-on-example-e6c2afb40d38.
- Hafidz Jazuli, An Introduction to Decision Tree Learning: ID3 Algorithm, 2018. https://medium.com/machine-learning-guy/an-introduction-to-decision-tree-learning-id3-algorithm-54c74eb2ad55
- Eric Eaton, Decision Trees, https://www.seas.upenn.edu/~cis519/fall2017/lectures/02_DecisionTrees.pdf.
- Upasana Priyadarshiny, How to Create a Perfect Decision Tree, https://dzone.com/, 2019.
- Aliridho Barakbah, Modul ajar Kecerdasan Buatan, PENS.
- Tessy Badriyah, Modul ajar Kecerdasan Buatan, PENS.
- M. Rosyid Mubtada'i, Modul ajar Kecerdasan Buatan, PENS.