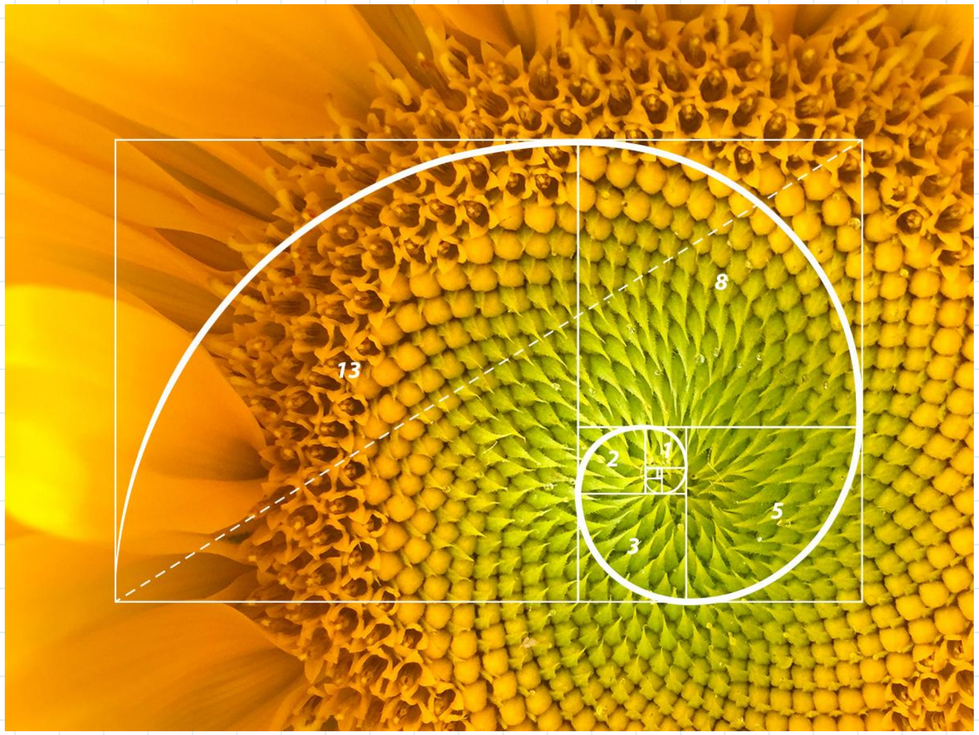
TM-03

MACHINE LEARNING

Decision Tree & Random Forest

Dr. Sirojul Munir, S.Si., M.Kom.
rojulman@nurulfikri.ac.id

ARTIFICIAL INTELLIGENCE – INFORMATICS STTNF



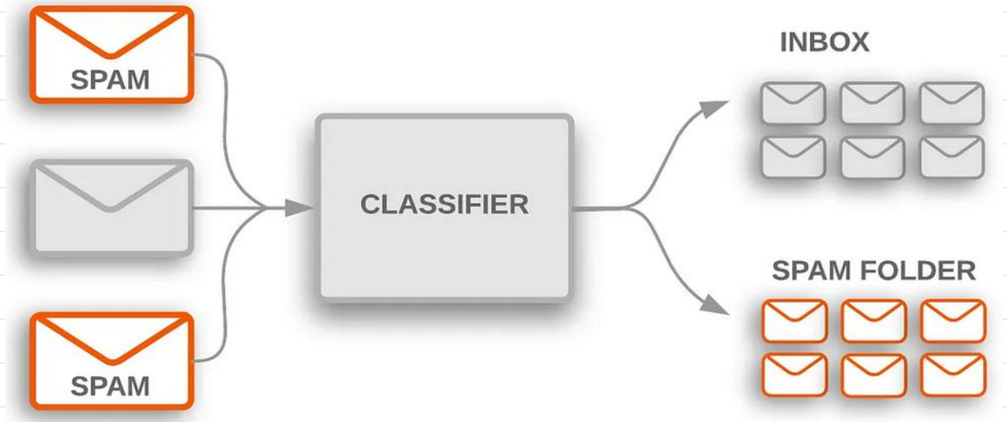
Daur ulang Project Data Science

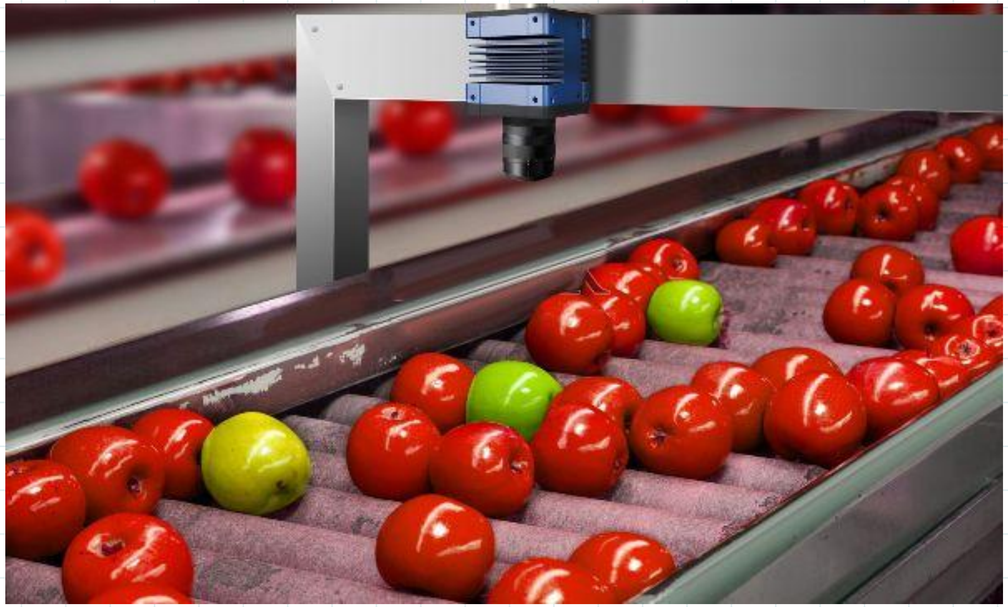
Algoritma Klasifikasi

- Algoritma klasifikasi mengacu pada **pemodelan prediktif** untuk **memprediksi label kelas** berdasarkan data **inputan**

- Algoritma **mengelompokan data** atau objek ke **dalam kategori** atau **kelas** tertentu **berdasarkan fitur** dan **atribut** data

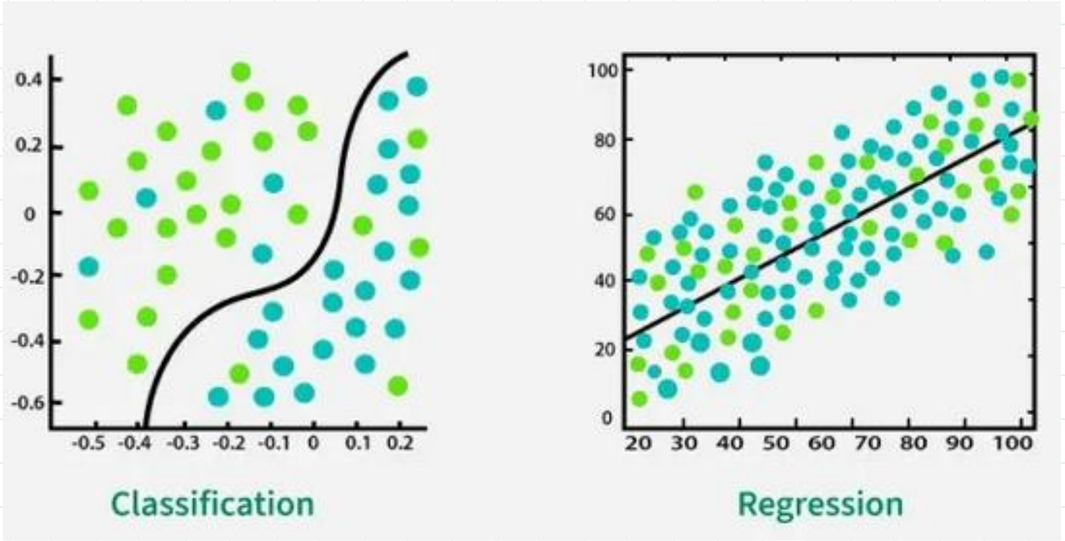
- Tujuan: **memprediksi** kategori atau kelas yang tepat untuk setiap data **berdasarkan model**, **pola** atau aturan yang telah dilakukan dalam proses training data

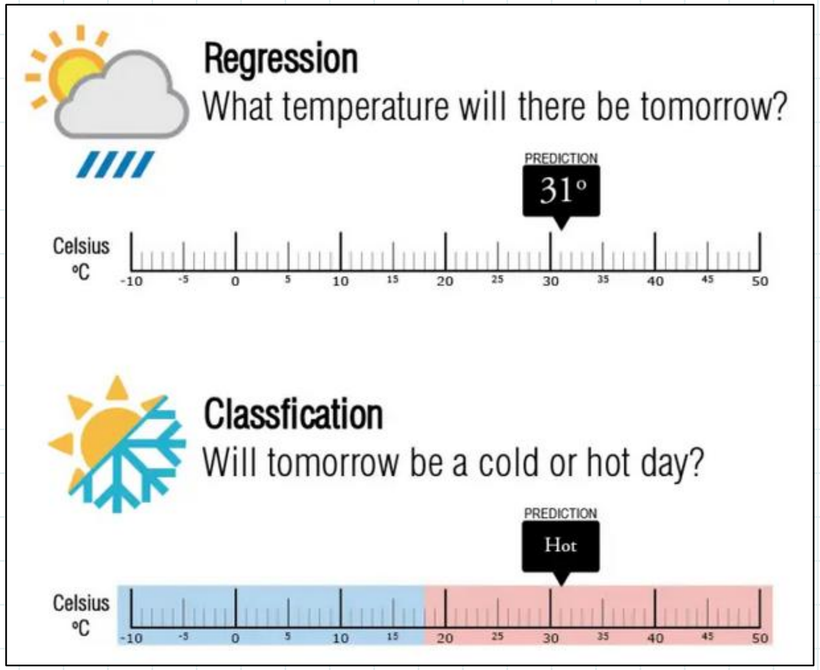


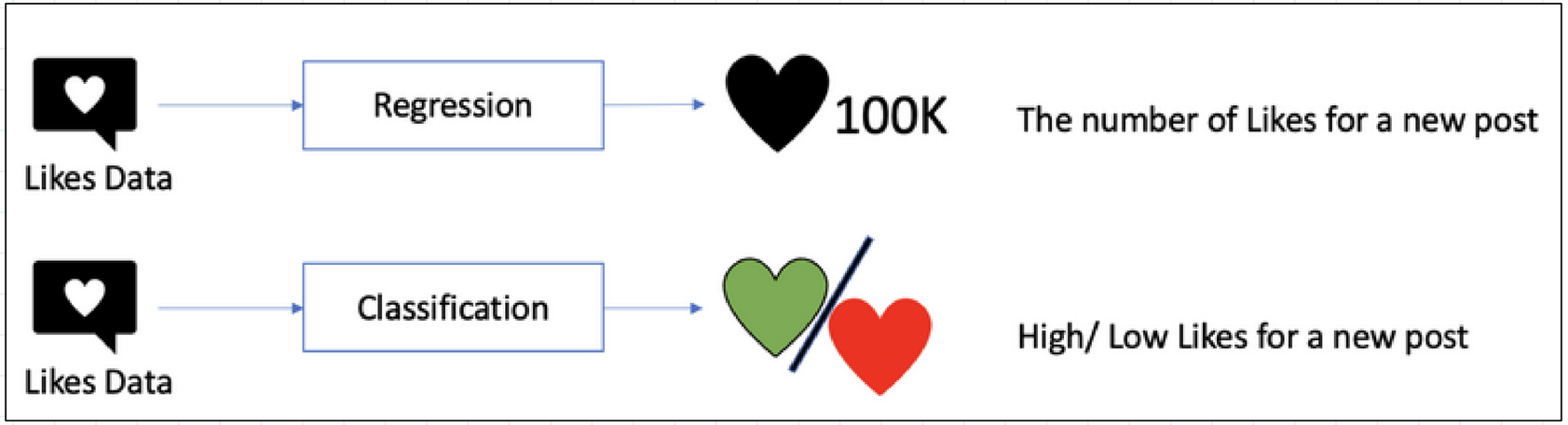


Regresi vs Klasifikasi

- **Regresi**: Model memprediksi label atau kelas yang bersifat kontinu / numerik
- **Klasifikasi:** Model memprediksi label atau kelas yang bersifat diskret / kategorikal







Algoritma Klasifikasi

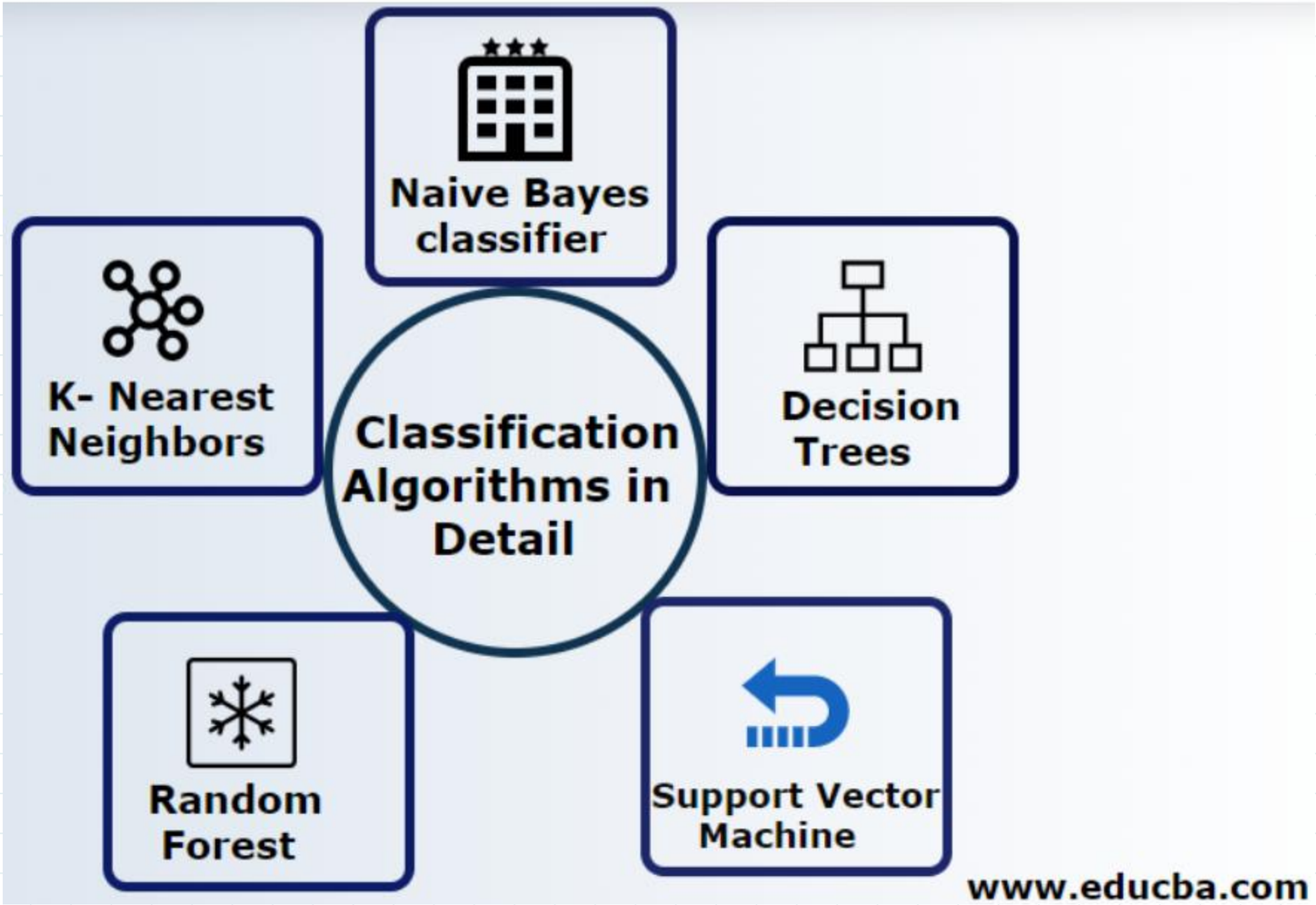
1.K- Nearest Neighbors

2.Naive Bayes classifier

3.Decision Trees

4.Support Vector Machine

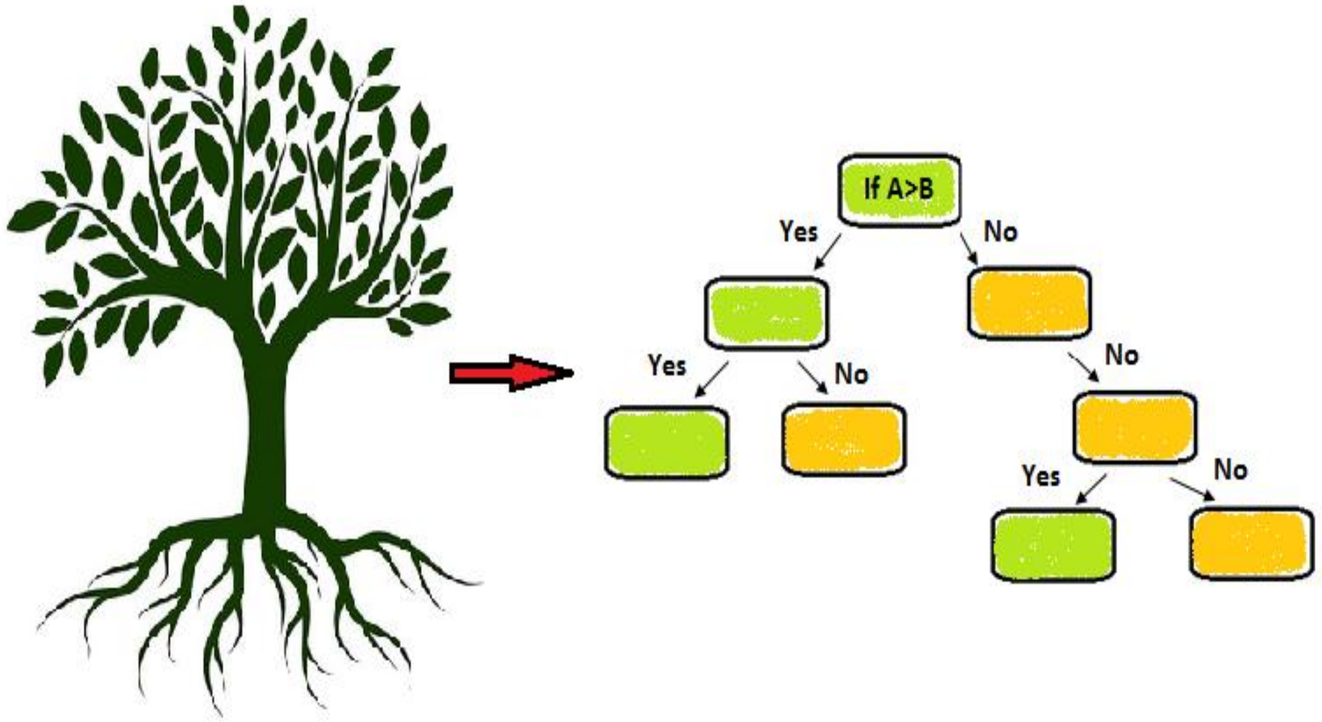
5.Random Forest



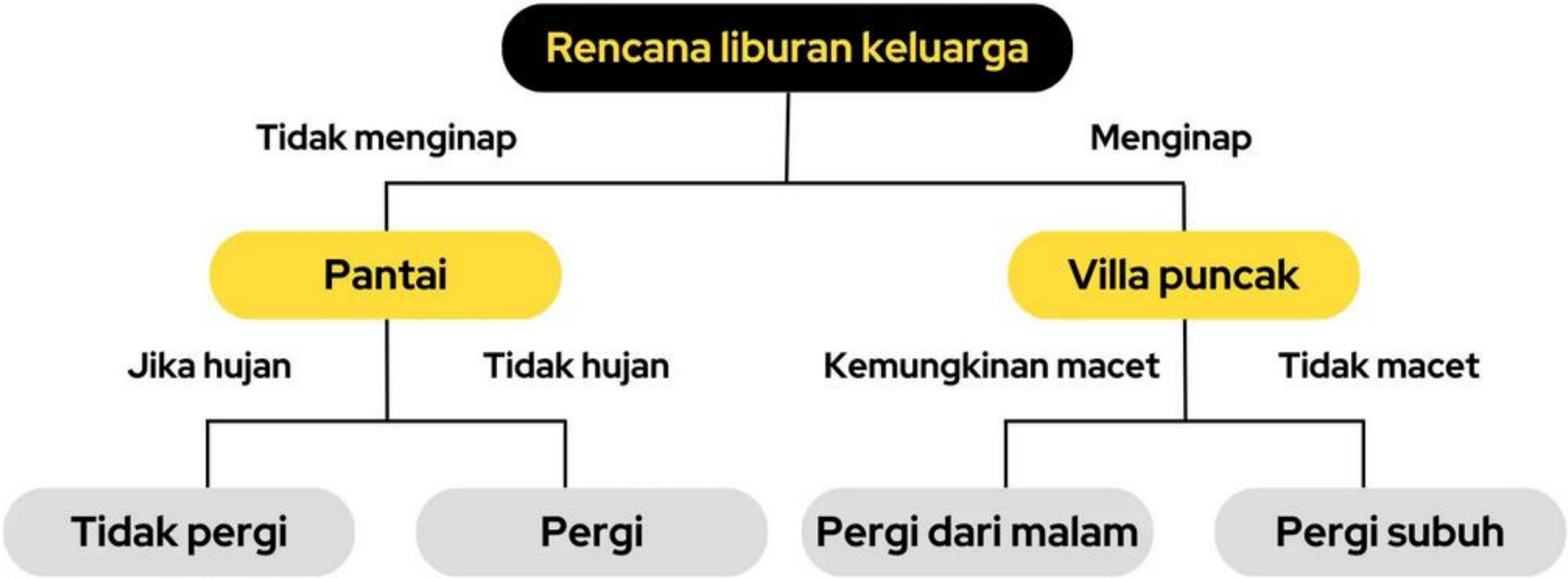
Pohon Keputusan (*Decision Tree*)

- ***Decision Tree*:** Pendekatan analisis prediktif berdasarkan struktur hirarki pohon keputusan

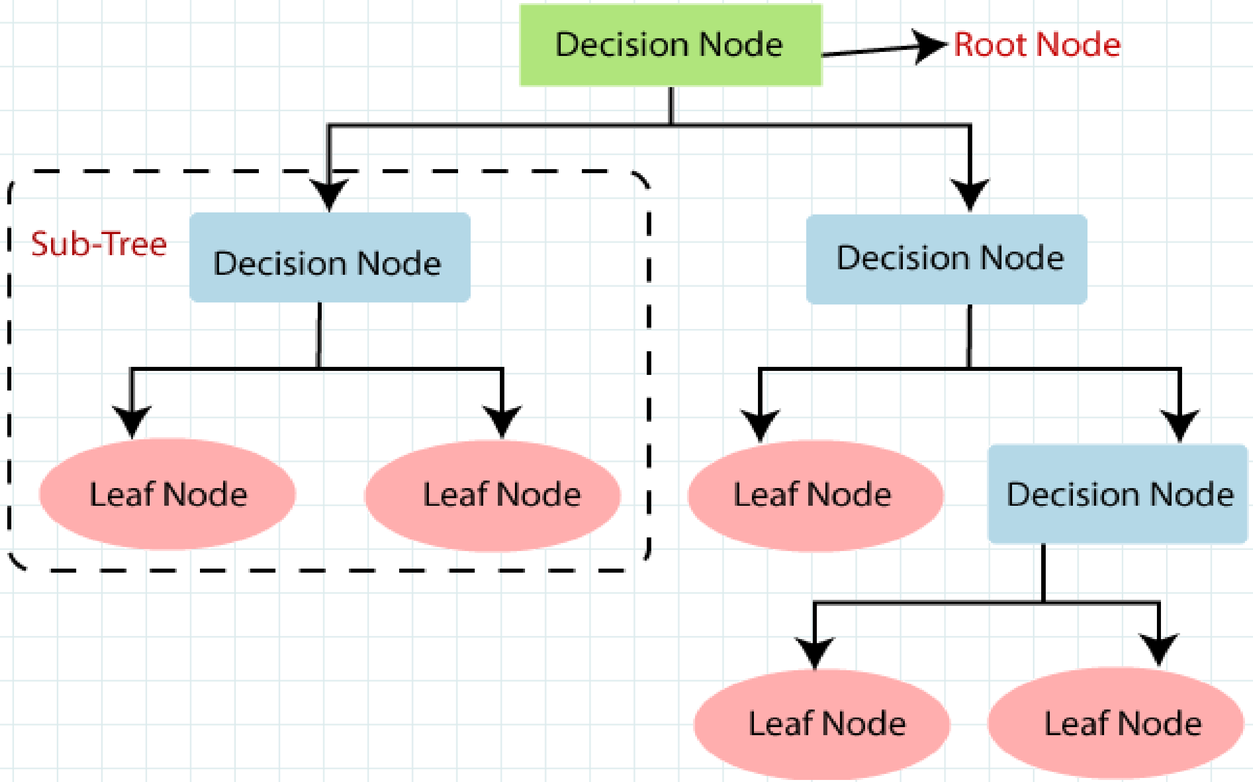
- Setiap pencabangan menyatakan kondisi yang harus dipenuhi
- Setiap ujung pohon menyatakan kelas data
- Model pohon (tree) menjadi aturan (rule)

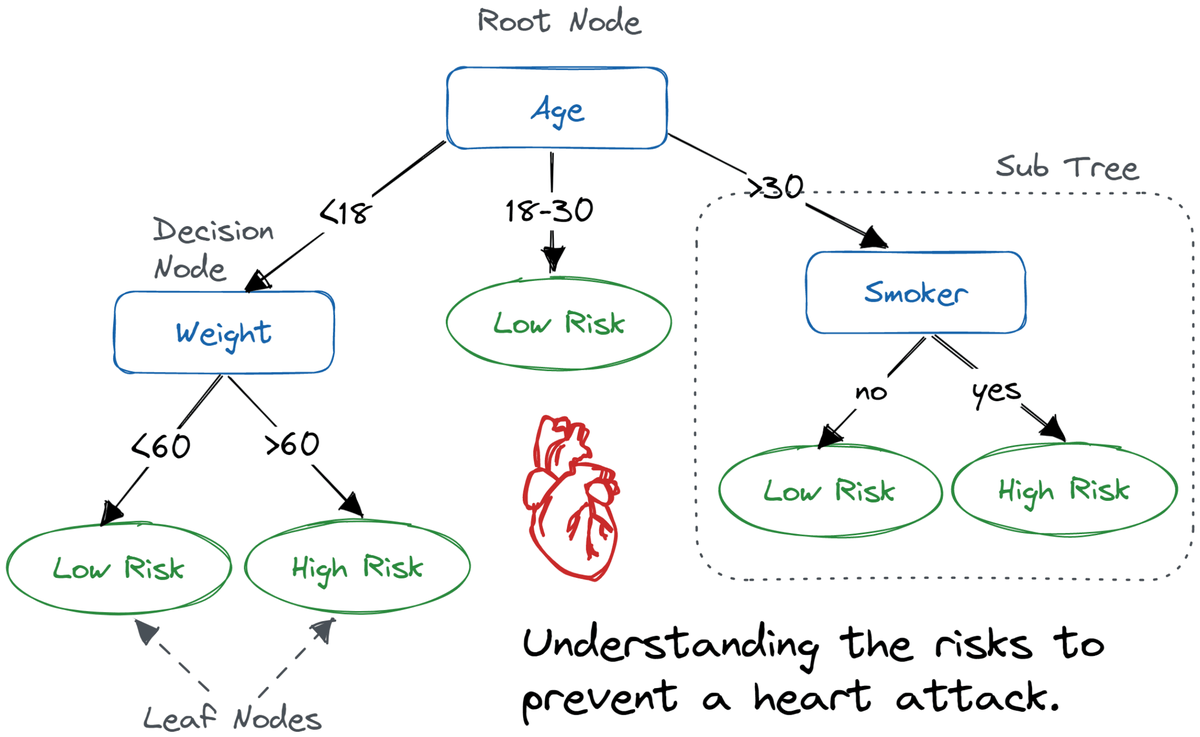


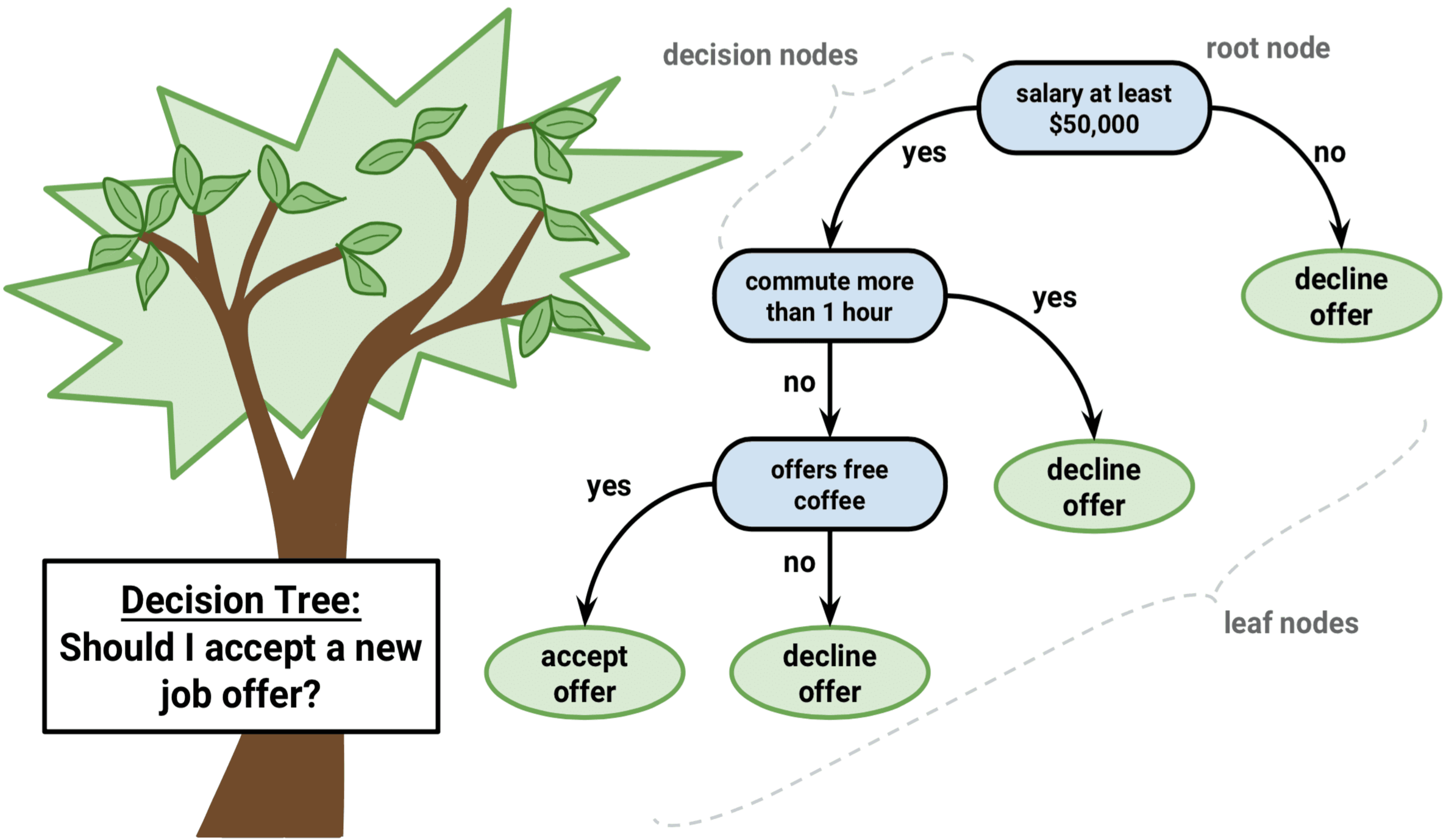
Contoh Kasus Pohon Keputusan:



Struktur Pohon Keputusan






Decision Tree:
Should I accept a new job offer?
decision nodes
root node
salary at least $50,000
yes
no
commute more than 1 hour
yes
decline offer
no
offers free coffee
decline offer
yes
no
accept offer
decline offer
leaf nodes

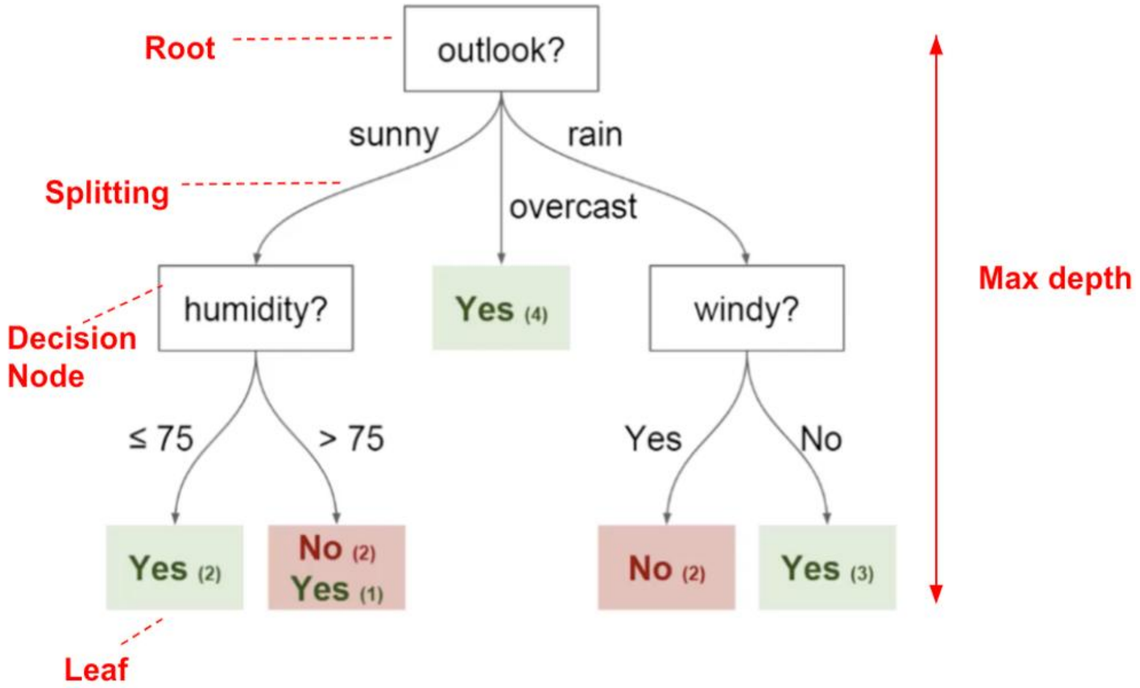
Terminilogi Decision Tree

1.Node Pohon:

- ✓Setiap node dalam pohon keputusan mewakili suatu keputusan atau pemisahan berdasarkan nilai fitur tertentu.
- ✓Node dibagi menjadi dua jenis:
  1.Node keputusan (Decision Node)
  2.Node daun (Leaf/Terminal Node).
- ✓Node keputusan memiliki dua atau lebih cabang yang mewakili nilai yang berbeda dari fitur yang dipilih.

2.Node Akar (Root Node):

- ✓Titik awal dari pemisahan data
- ✓Mewakili dataset yang akan dibagi berdasarkan nilai fitur (atribut)



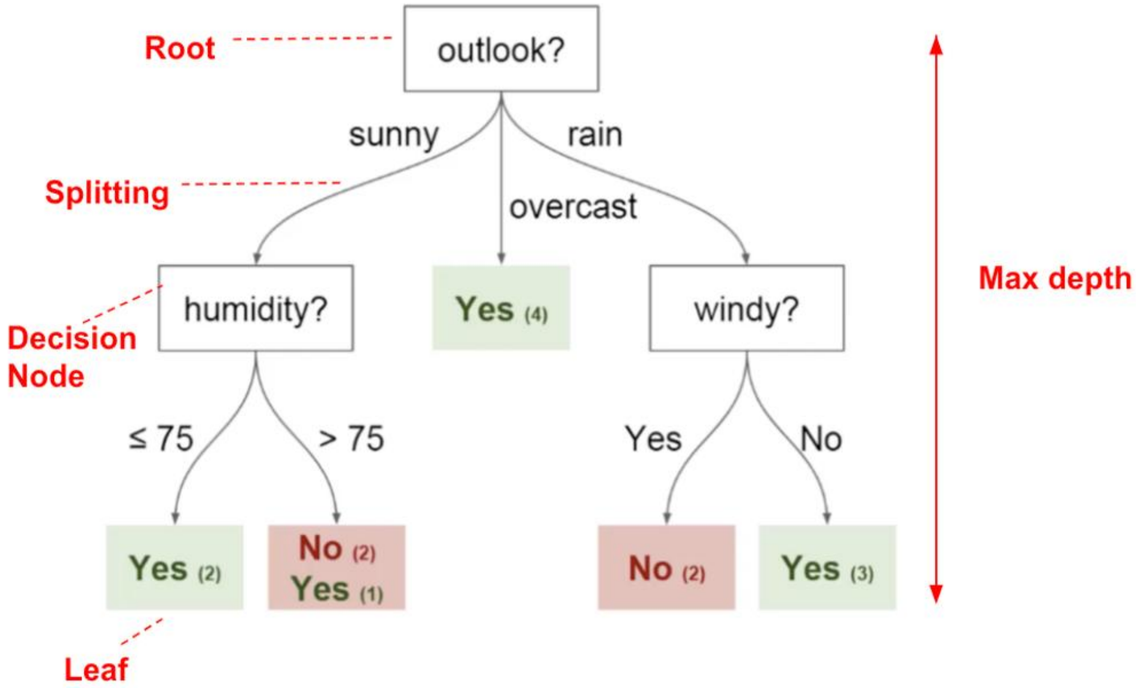
Terminilogi Decision Tree

3.Edge atau Cabang:

- ✓Edge: menghubungkan node keputusan dengan node lainnya dan menunjukan hasil dari pemilihan fitur pada node tersebut.
- ✓Setiap Edge/Cabang mewakili suatu nilai fitur yang dapat diambil oleh instan data

4.Node Daun:

- ✓Node yang tidak memiliki cabang lagi, dan mewakili hasil dari klasifikasi atau prediksi akhir
- ✓Setiap node daun berhubungan dengan satu kelas atau nilai target



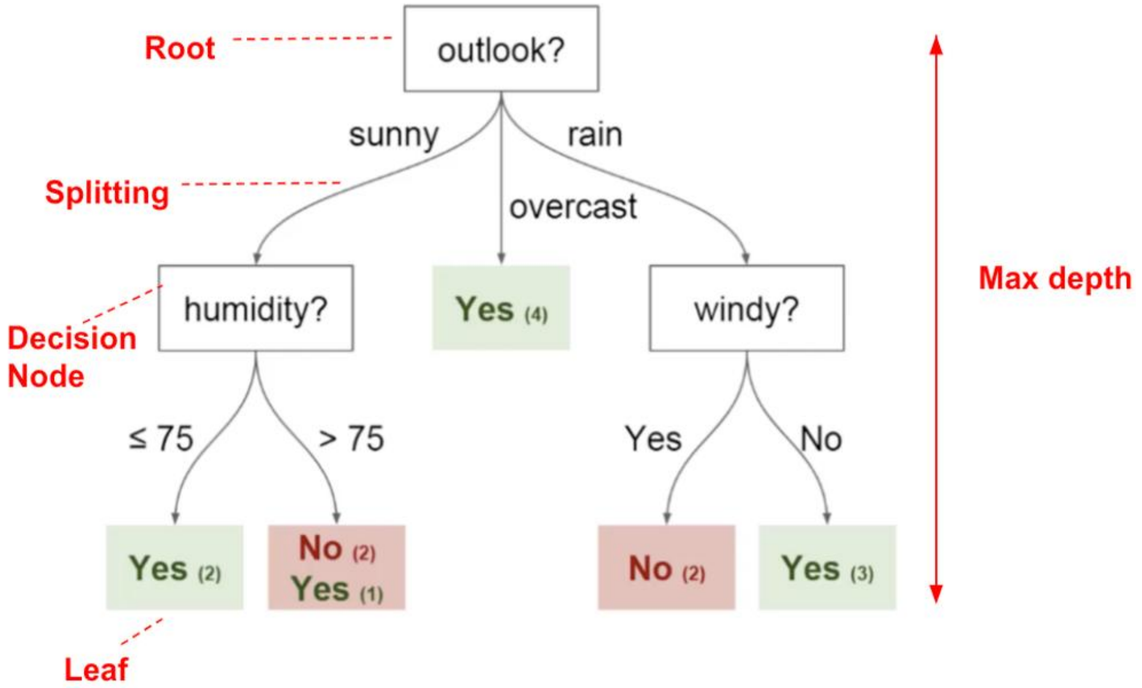
Terminilogi Decision Tree

5.Informasi Gain:

- ✓Informasi Gain adalah metrik yang digunakan untuk mengevaluasi kegunaan suatu atribut dalam memisahkan data
- ✓Informasi Gain mengukur sejauh mana pemilihan suatu atribut mengurangi ketidakpastian atau entropi tentang kelas targe

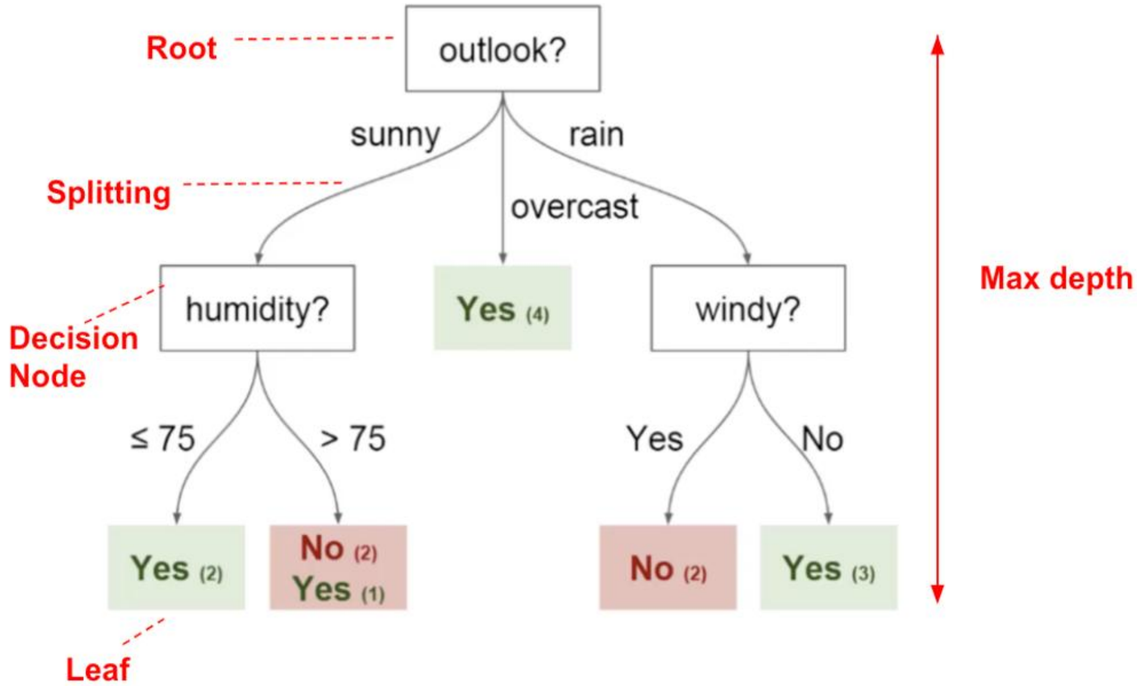
6.Entropi:

- ✓Entropi adalah ukuran ketidakpastian atau kekacauan dalam dataset.
- ✓Pemilihan atribut dengan Informasi Gain tinggi membantu mengurangi entropi dan meningkatkan kepastian tentang kelas target



Terminilogi Decision Tree

7. **Kriteria Berhenti:**
   - ✓ Kriteria berhenti adalah aturan yang menentukan kapan proses pembentukan pohon harus berhenti.
   - ✓ Contoh kriteria berhenti meliputi ketika semua instance dalam suatu subset memiliki label kelas yang sama atau ketika kedalaman maksimum pohon telah tercapai.
8. **Pruning (Pemangkasan):**
   - ✓ Pemangkasan adalah proses mengurangi kompleksitas pohon keputusan dengan menghapus cabang-cabang yang kurang penting atau dapat menyebabkan overfitting.
   - ✓ Pemangkasan dapat dilakukan setelah pembentukan pohon atau selama pembentukan pohon



Entropy & Gain

- Nilai perhitungan Entropy dan Gain Information digunakan untuk memilih fitur terbaik yang akan membagi dataset

1. Entropy (H):

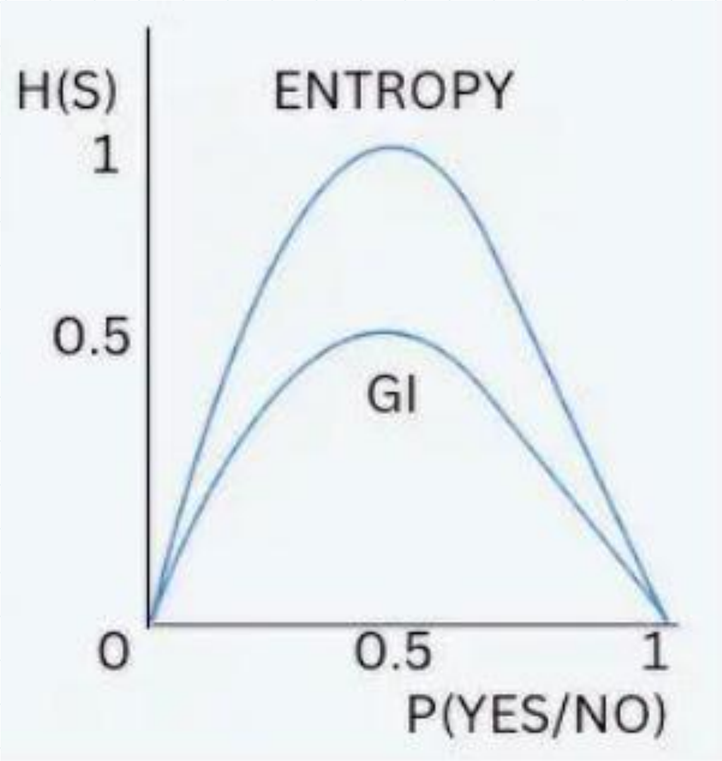
✓ Mengukur tingkat ketidakpastian atau kekacauan dalam suatu dataset

✓ Nilai entropi akan lebih tinggi jika dataset memiliki kecenderungan data beragam (campur aduk)

2. Gain Information (GI):

✓ mengukur seberapa baik suatu atribut membagi dataset.

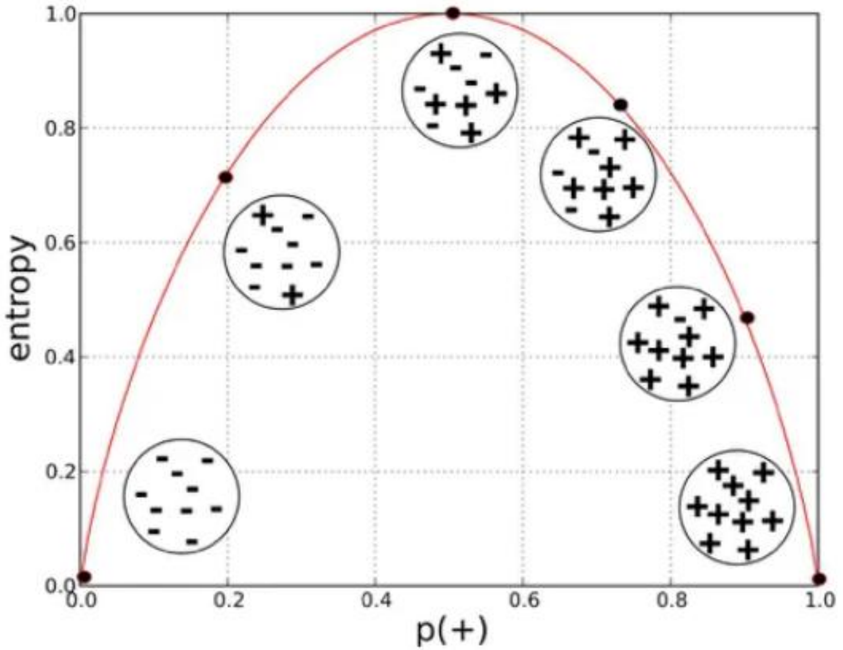
✓ Atribut dengan Information Gain yang lebih tinggi dianggap lebih baik untuk pembagian



Decision Tree :: Entropy (H)

$$H(S) = -\sum_{i=1}^{c} p_i \cdot \log_2(p_i)$$

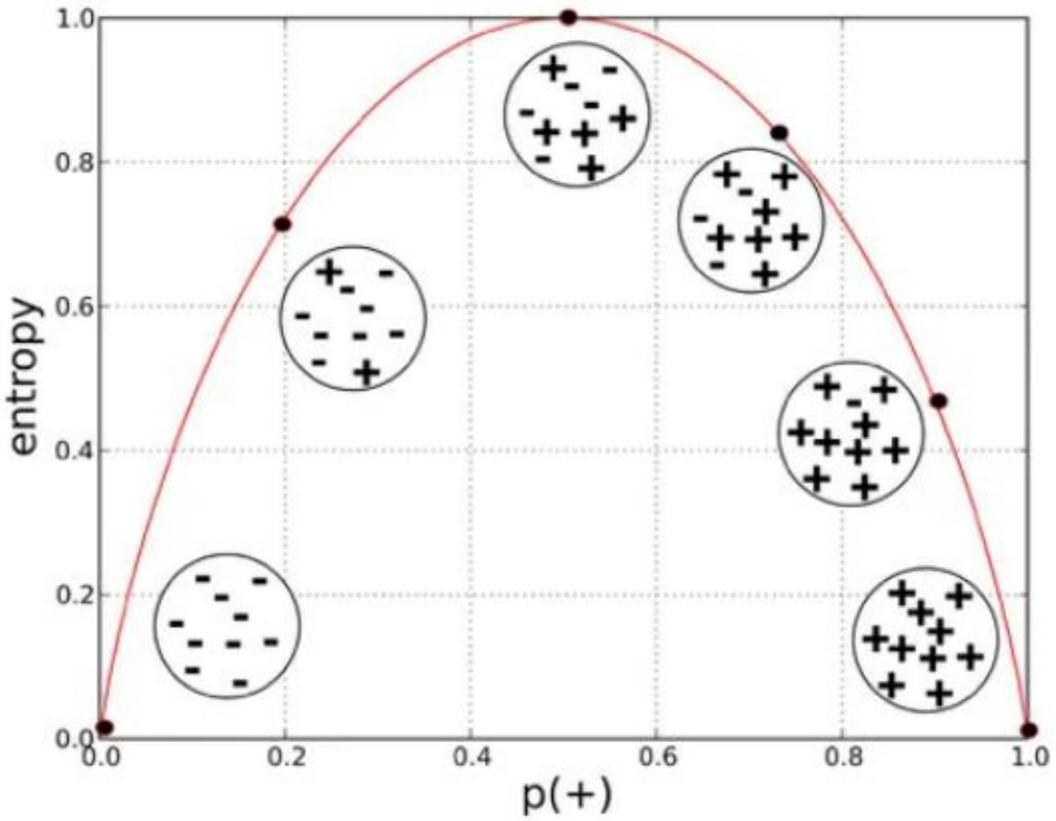
- $H(S)$: Entropi dari dataset S.
- c: Jumlah kelas dalam dataset.
- p_i: Proporsi dari dataset yang termasuk dalam kelas i.
- $\log_2$: Logaritma basis 2.



Entropy mengukur **seberapa murni (homogen)** atau **tidak murni (heterogen)** data dalam suatu node

- Jika semua data dalam node **memiliki kelas yang sama**, maka **entropy = 0** → node **sangat murni**.
- Jika data dalam node **tercampur rata antara kelas Yes dan No**, maka **entropy = 1** → node **paling tidak murni** (paling acak).

Decision Tree :: Entropy (H)



Entropy dapat diartikan seperti “**tingkat kebingungan**” **model** ketika melihat **data**:

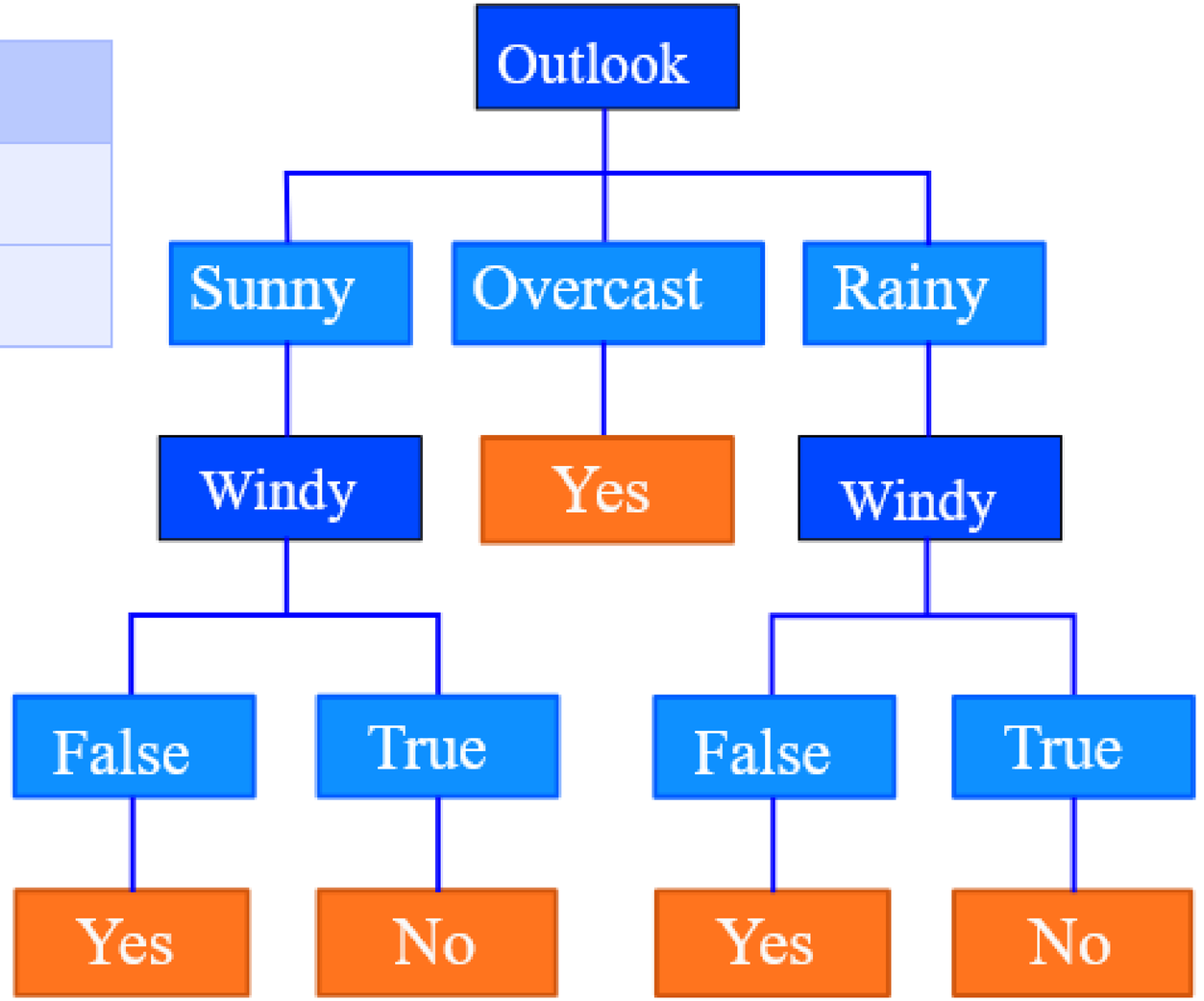
- Entropy 0 → tidak bingung sama sekali.
- Entropy 1 → benar-benar bingung (acak total).
- Entropy 0.5 – 0.9 → agak bingung; perlu informasi tambahan dari atribut lain.

Case Study :: Dataset bermain golf

Play Golf Dataset

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Play Golf	
Yes	No
9	5



Case Study :: Dataset bermain golf

Dataset mengandung **p** objects of class **P** and **n** of class **N, probabilitas P =** p/(p+n) , dan **N** = n/(p+n).

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

- a) Entropy of target class of the dataset(Whole entropy):

$$\begin{aligned} & Entropy(PlayGolf) \\ &= Entropy(5,9) = I(\frac{5}{5+9}, \frac{9}{5+9}) \\ &= I(0.36, 0.64) \\ &= -(0.36 log_2 0.36) - (0.64 log_2 0.64) \\ &= 0.94 \end{aligned}$$

(1)

Untuk dua kelas (misal: Yes / No):

$$Entropy(S) = -p(Yes)\log_2 p(Yes) - p(No)\log_2 p(No)$$

di mana:

- $p(Yes)$ = proporsi data dengan label "Yes"
- $p(No)$ = proporsi data dengan label "No"

Play Golf	
Yes	No
9	5

Case Study :: Dataset bermain golf

Dataset mengandung **p** objects of class **P** and **n** of class **N, probabilitas P =** p/(p+n) , dan **N** = n/(p+n).

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

- a) Entropy of target class of the dataset(Whole entropy):

$$\begin{aligned} & Entropy(PlayGolf) \\ &= Entropy(5,9) = I(\frac{5}{5+9}, \frac{9}{5+9}) \\ &= I(0.36, 0.64) \\ &= -(0.36 log_2 0.36) - (0.64 log_2 0.64) \\ &= 0.94 \end{aligned}$$

Untuk dua kelas (misal: Yes / No):

$$Entropy(S) = -p(Yes)\log_2 p(Yes) - p(No)\log_2 p(No)$$

di mana:

- $p(Yes)$ = proporsi data dengan label "Yes"
- $p(No)$ = proporsi data dengan label "No"

Play Golf	
Yes	No
9	5

Jadi **entropy** = **0.94**, artinya node ini cukup **tidak murni** (ada campuran antara Yes dan No).

Algoritma Decision Tree

Play Golf Dataset

Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Atribut Outlook (**Sunny / Overcast / Rainy**)

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

Algoritma Decision Tree : Entropy

"Seberapa besar ketidakpastian (entropy) pada label PlayGolf **jika** data dibagi berdasarkan atribut Outlook (**Sunny / Overcast / Rainy**) **baik digunakan untuk memecah data** di decision tree.?"

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

Rumus yang digunakan:

$$Entropy(A) = \sum_{i=1}^{k} \frac{p_i + n_i}{p + n} \times I(p_i, n_i)$$

- p_i: jumlah data positif (Yes) pada cabang ke-i
- n_i: jumlah data negatif (No) pada cabang ke-i
- $p + n$: total data
- $I(p_i, n_i)$: entropy di cabang ke-i

Setiap **nilai unik dari atribut** (Sunny, Overcast, Rainy) memiliki entropinya sendiri, lalu semuanya dirata-ratakan dengan bobot proporsi jumlah datanya.

Algoritma Decision Tree : Entropy

“Seberapa besar ketidakpastian (entropy) pada label PlayGolf **jika** data dibagi berdasarkan atribut Outlook (**Sunny / Overcast / Rainy**) **baik digunakan untuk memecah data** di decision tree.?”

		Play Golf		
		Yes	No	
Outlook	Sunny	3	2	5
	Overcast	4	0	4
	Rainy	2	3	5
				14

Kemudian dihitung rata-ratanya:

$$Entropy(PlayGolf, Outlook) = \frac{5}{14}(0.971) + \frac{4}{14}(0) + \frac{5}{14}(0.971)$$

$$= 0.693$$

(2)

Interpretasi:

- Nilai **entropy** = **0.693** menunjukkan bahwa **masih ada ketidakpastian sedang** ketika data dipisah berdasarkan `Outlook`.
- Ini artinya: `Outlook` **belum sepenuhnya memisahkan data** menjadi kelas yang murni (“Yes”/“No”), tapi **cukup baik** dibanding atribut lain (seperti `Temp` atau `Windy`).

- Entropy = 0.693 → ada **ketidakpastian sedang**; berarti masih ada **peluang bermain golf**, tapi belum bisa dipastikan hanya dari cuaca saja.

Decision Tree :: Information Gain (IG)

$$IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} \cdot H(S_v)$$

- $IG(S, A)$: Information Gain untuk atribut A dalam dataset S.
- $H(S)$: Entropi awal dataset S.
- $Values(A)$: Nilai-nilai yang dapat diambil oleh atribut A.
- S_v: Subset dari S yang memiliki nilai atribut A sama dengan v.
- $|S|$: Jumlah total instance dalam dataset S.

(1) (2)

$$Gain(Outlook) = Entropy(PlayGolf) - Entropy(PlayGolf, Outlook) = 0.940 - 0.693 = 0.247$$

Terminilogi Decision Tree :: Gain dan Entropi

- Dalam membangun Pohon Keputusan:
  1. Atribut dengan **Information Gain (IG) tertinggi akan dipilih** untuk pemisahan di setiap node
  2. Dataset dengan nilai **entropy (H) yang rendah** maka **nilai Information Gain (IG) akan cenderung tinggi** dan berlaku sebaliknya
  3. Tujuan dari Pohon Keputusan adalah **memilih atribut yang dapat mengurangi entropi maksimum** atau **mendapatkan information gain yang maksimum**

Case Study : Decision Tree

Terima Kasih

http://youtube.com/@rojulman