TEKNIK INFORMATIKA
FAKULTAS TEKNIK UNIVERSITAS MATARAM

Learning

Ramaditia D

v1.1 update 8 Nov 20

Outline

- Pembelajaran Mesin
- Supervised Learning
- Data Pembelajaran
- Klasifikasi
- Evaluasi Pembelajaran Klasifikasi
- Klasifikasi dengan kNN
- Klasifikasi Naïve Bayes Classifier

Pembelajaran Mesin

- Kajian mengenai pembuatan program komputer yang secara otomatis meningkatkan atau menyesuaikan kinerjanya melalui pengalaman.
- “programming by example”
- Tujuan: menghasilkan algoritme pembelajaran yang belajar secara otomatis tanpa bantuan manusia
- Tidak menyelesaikan masalah secara langsung, program mencari metode terbaik dengan mempelajari contoh yang disediakan.

Kesulitan Membangun Expert System

- Tidak ada pakar (manusia) untuk masalah yang harus diselesaikan
- Ada pakar, tetapi tidak mampu menjelaskan kepakarannya.
- Masalah bersifat dinamis dimana fenomena berubah dengan cepat
- Aplikasi perlu disesuaikan/dipersonalkan untuk setiap pengguna komputer secara terpisah.

Terapan Pembelajaran Mesin

- Pengenalan huruf dan angka (*character*)
- Pengenalan tulisan tangan (*handwriting*)
- Deteksi wajah
- Penyaringan spam
- Pengenalan suara
- Pemahaman bahasa pembicaraan
- Prediksi pasar saham
- Prediksi cuaca
- Diagnosa medis
- Deteksi penipuan
- Pencocokan sidik jari (*fingerprint*)

Jenis Pembelajaran Mesin

- **Supervised Learning**
  - Model Pembelajaran Terbimbing
- **Unsupervised Learning**
  - Model Pembelajaran Tidak Terbimbing
- **Reinforcement Learning**
  - Model Pembelajaran berdasarkan reward dan punishment.

Supervised Learning

- Model Pembelajaran Terawasi yaitu “mengajarkan” model dengan pengetahuan sehingga dapat memprediksikannya di masa mendatang

- **Terdapat pemberian label** dalam proses pembelajaran. Mengajarkan model dan melatihnya dengan beberapa data dari **dataset yang berlabel**.

- Diberikan kumpulan data pasangan *input-output*, kemudian mempelajari fungsi untuk melakukan pemetaan *input* ke *output*.

Data Pembelajaran (Dataset)

Input/Atribut/fitur

Date	Humidity	Pressure	Temperature	Rain
January 1	93%	999.7	20	Rain
January 2	49%	1015.5	27	No Rain
January 3	79%	1031.1	28	No Rain
January 4	65%	984.9	21	Rain
January 5	90%	975.2	22	Rain

Output/Target/Class/Label

***f*(*inputs*) = Outputs**

f(*humidity*, *pressure*, *temp*) = Rain/No Rain?

f(93, 999.7, 20) = Rain

f(49, 1015.5, 27) = No Rain

f(79, 1031.1, 28) = No Rain

Data Pembelajaran (Dataset)

Output

Input/Atribut/fitur

Date	Humidity	Pressure	Temperature	Rain
January 1	93%	999.7	20	Rain
January 2	49%	1015.5	27	No Rain
January 3	79%	1031.1	28	No Rain
January 4	65%	984.9	21	Rain
January 5	90%	975.2	22	Rain

***f*(*inputs*) = Outputs**

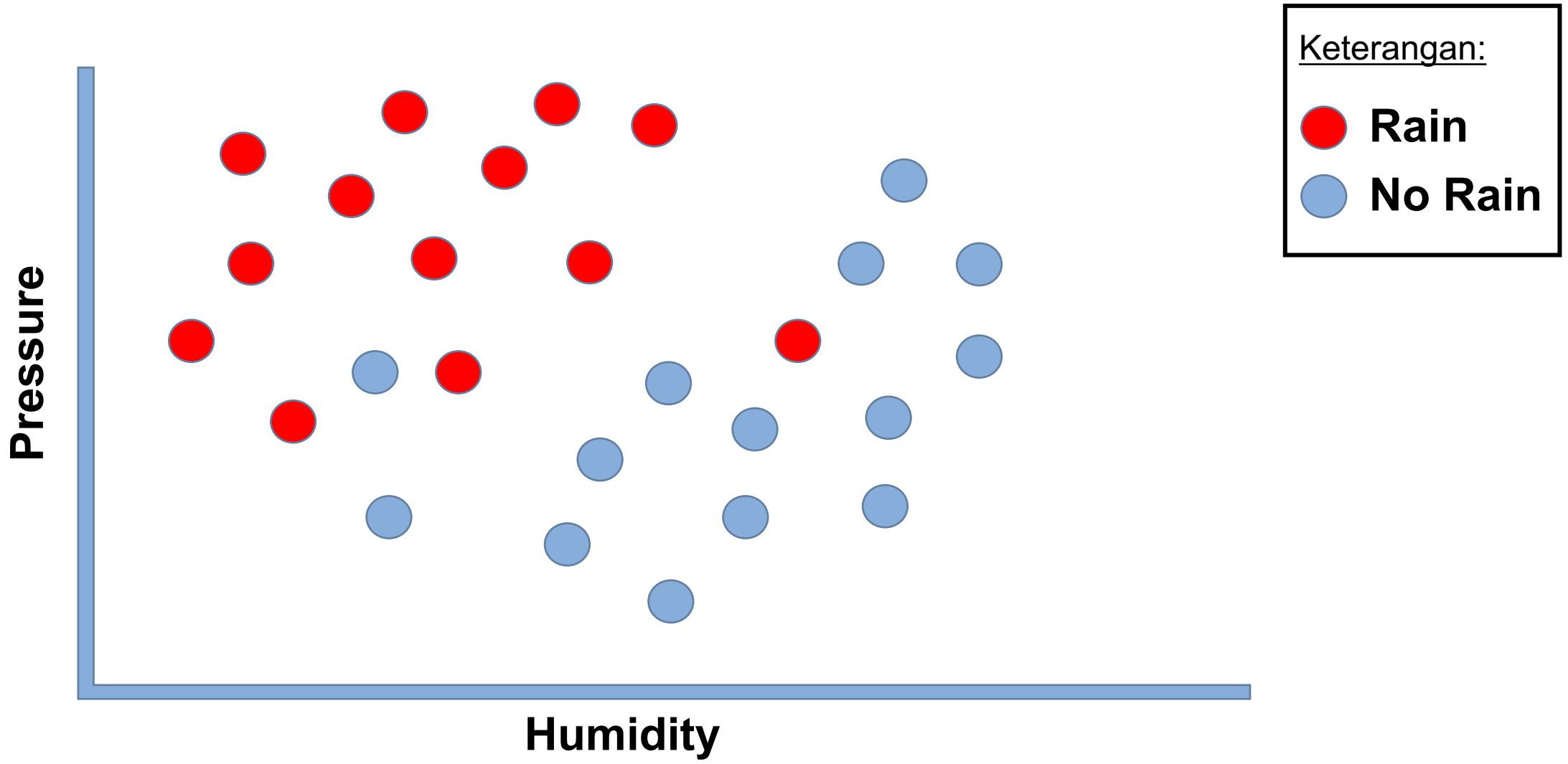
$f(pressure, temp) = \text{Humidity?}$

$f(999.7, 20) = 93$

$f(1015.5, 27) = 49$

$f(1031.1, 28) = 79$

Visualisasi Data



Jenis Nilai Data

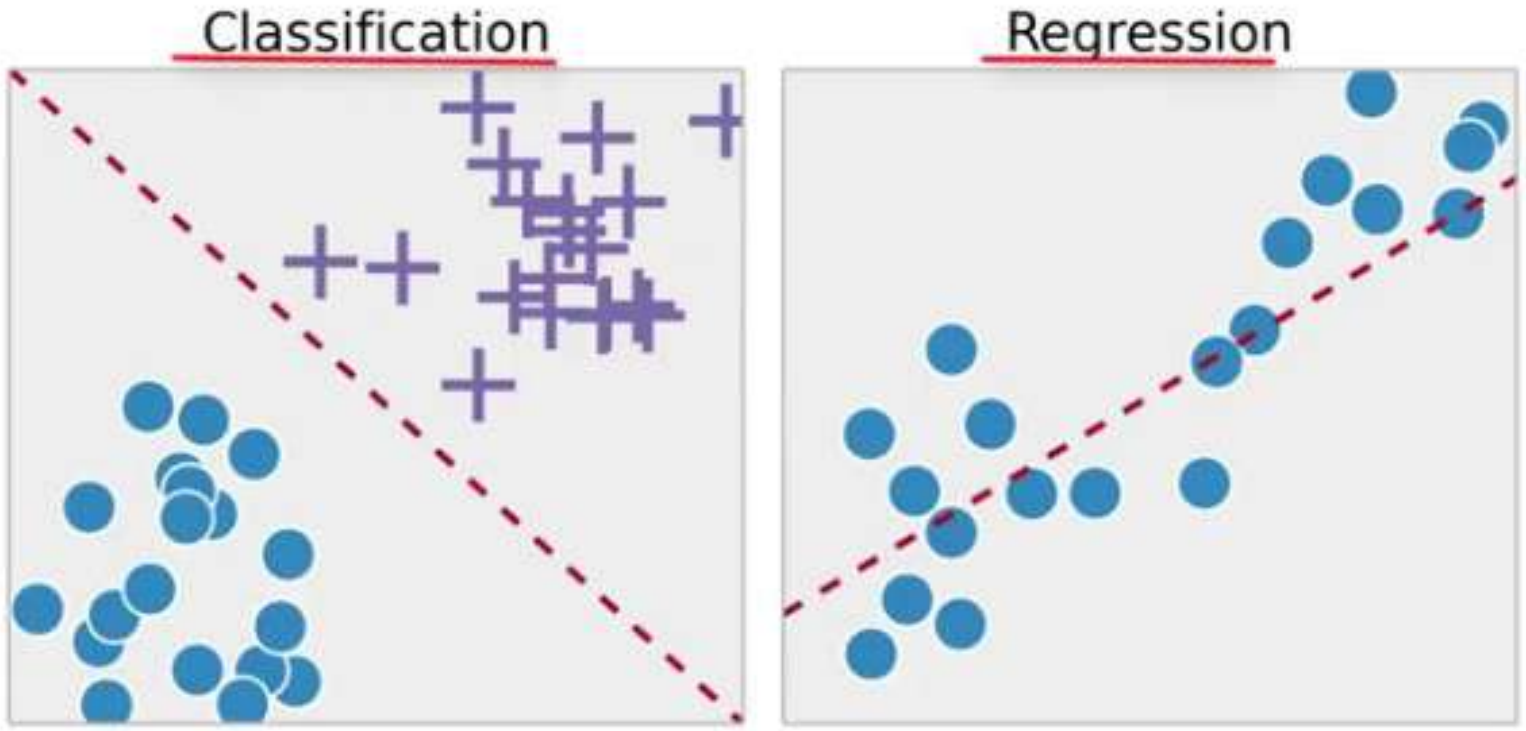
Date	Humidity	Pressure	Temperature	Rain
January 1	93%	999.7	20	Rain
January 2	49%	1015.5	27	No Rain
January 3	79%	1031.1	28	No Rain
January 4	65%	984.9	21	Rain
January 5	90%	975.2	22	Rain

Numerik

Non Numerik/Categorical

Pada data tersebut terlihat nilai data memiliki dua jenis, Yang pertama adalah angka.
Data yang paling umum digunakan adalah angka. Yang kedua adalah huruf yaitu non-numerik, karena berisi karakter bukan angka.

Tipe Supervised Learning

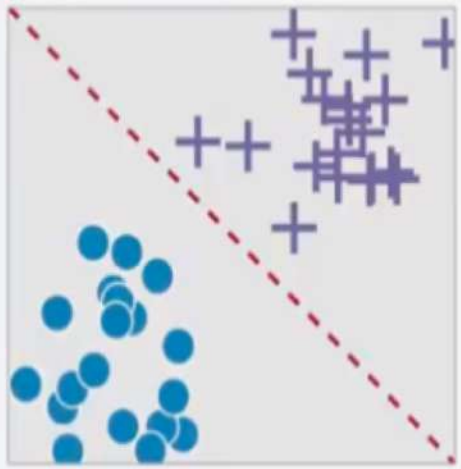


Klasifikasi Vs Regresi

Klasifikasi adalah proses memprediksi label atau kategori kelas diskrit.

ID	Clump	UnifSize	UnifShape	MargAdh	SingEpiSize	BareNuc	BlandChrom	NormNucl	Mit	Class
1000025	5	1	1	1	2	1	3	1	1	benign
1002945	5	4	4	5	7	10	3	2	1	benign
1015425	3	1	1	1	2	2	3	1	1	malignant
1016277	6	8	8	1	3	4	3	7	1	benign
1017023	4	1	1	3	2	1	3	1	1	benign
1017122	8	10	10	8	7	10		7	1	malignant
1018099	1	1	1	1	2	10	3	1	1	benign
1018561	2	1	2	H	2	1	3	1	1	benign
1033078	2	1	1	1	2	1	1	1	5	benign
1033078	4	2	1	1	2	1	2	1	1	benign

Categorical Values

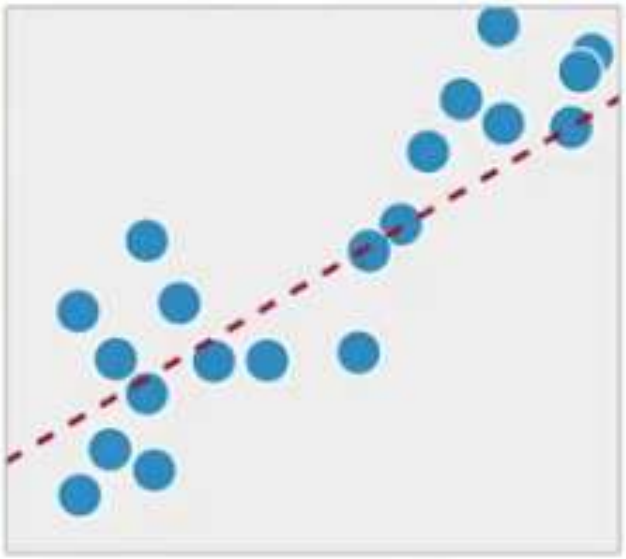


Klasifikasi Vs Regresi

Regresi adalah proses memprediksi nilai kontinu sebagai lawan dari prediksi nilai kategorikal dalam Klasifikasi.

	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_COMB	CO2EMISSIONS
0	2.0	4	8.5	196
1	2.4	4	9.6	221
2	1.5	4	5.9	136
3	3.5	6	11.1	255
4	3.5	6	10.6	244
5	3.5	6	10.0	230
6	3.5	6	10.1	232
7	3.7	6	11.1	255
8	3.7	6	11.6	267
9	2.4	4	9.2	?

Continuous Values



Klasifikasi

- Klasifikasi adalah proses memprediksi label atau kategori kelas diskrit.
- Diberikan sebuah himpunan observasi berupa data tabel, lengkap dengan label *class*-nya

age	ed	employ	address	income	debtinc	creddebt	othdebt	default
41	3	17	12	176	9.3	11.359	5.009	1
27	1	10	6	31	17.3	1.362	4.001	0
40	1	15	14	55	5.5	0.856	2.169	0
41	1	15	14	120	2.9	2.659	0.821	0
24	2	2	0	28	17.3	1.787	3.057	1
41	2	5	5	25	10.2	0.393	2.157	0
39	1	20	9	67	30.6	3.834	16.668	0
43	1	12	11	38	3.6	0.129	1.239	0
24	1	3	4	19	24.4	1.358	3.278	1
36	1	0	13	25	19.7	2.778	2.147	0

Data berupa Kategori

- Model Klasifikasi harus menentukkan *class* dari observasi baru yang belum diberikan label class.

age	ed	employ	address	income	debtinc	creddebt	othdebt	default
37	2	16	10	130	9.3	10.23	3.21	

?

Target / Class dari Klasifikasi

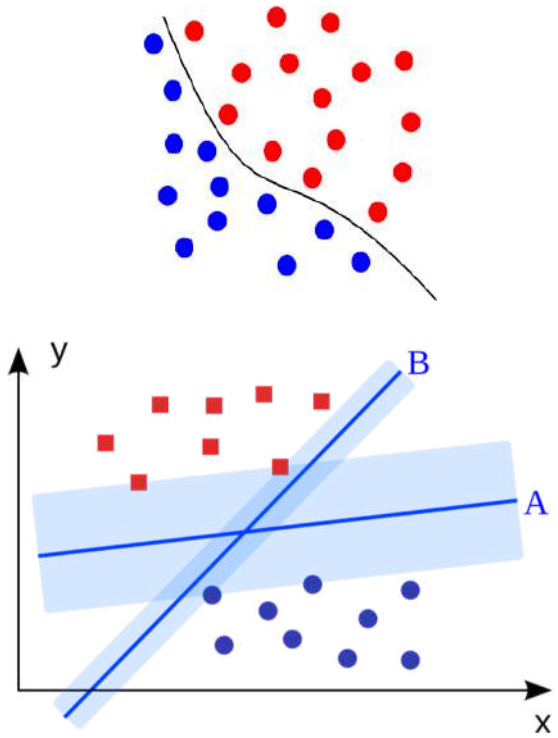
- Dari contoh yang sebelumnya, kita bias melihat bahwa target / class dari nasabah tersebut berupa binary
  - Memungkinkan melakukan pinjaman
  - Tidak memungkinkan melakukan pinjaman

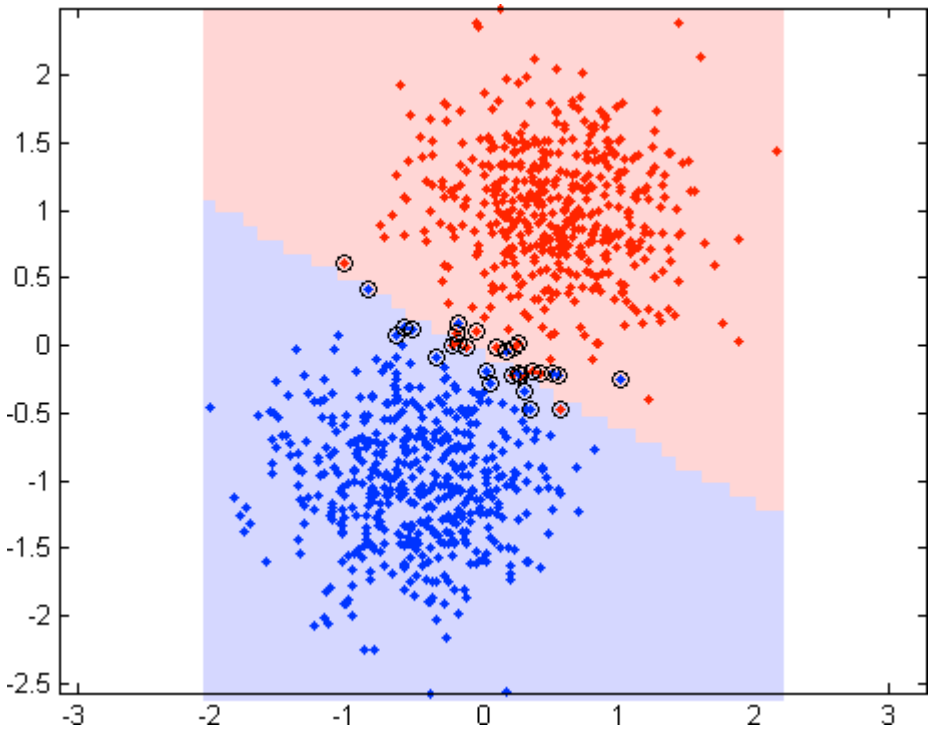
- Pada dasarnya, klasifikasi tidak hanya dapat melakukan *binary klasiifkasi* tetapi juga *multiclass klasifikasi.*

- Sebagai contoh:
  - Kelompok A, atau B, atau C.
  - Kucing, Harimau, atau Macan
  - Bunga anggrek, melati, atau bakung

Hal Penting dalam Klasifikasi

- Hal terpenting di dalam melakukan klasifikasi adalah menentukan fitur/aspek yang dapat dengan baik mengkategorikan suatu data.





Pengukuran Evaluasi dalam Klasifikasi

- Pengukuran Evaluasi (Evaluation Metrics) mendeskripsikan performa dari *model classifier* kita.
- Untuk membuat Evaluation Metrics, data training dibagi menjadi dua:

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0
5	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1

Training Data

Testing Data

Training data = Membuat *model classifier*.
Testing data = Memeriksa akurasi dari classifier

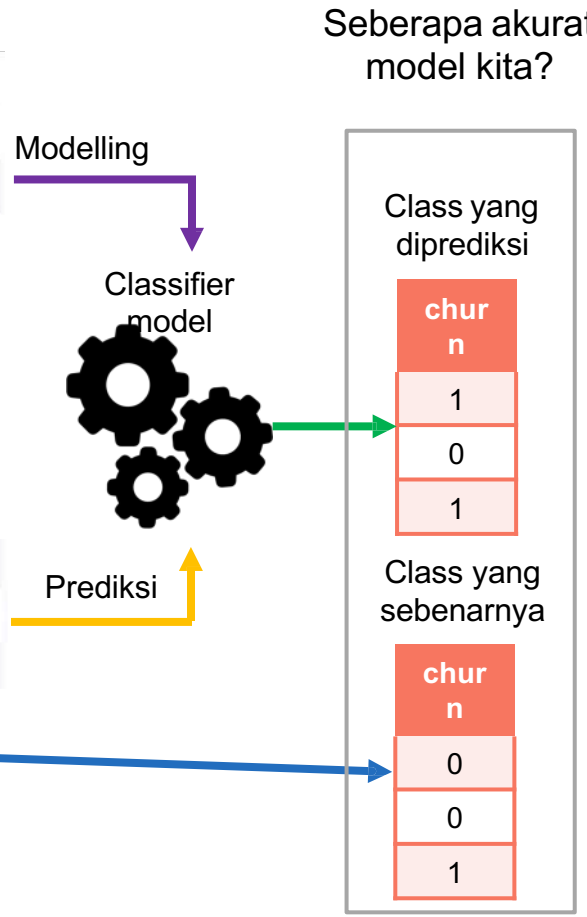
Training dan Testing Data

Training Data

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0

Testing Data

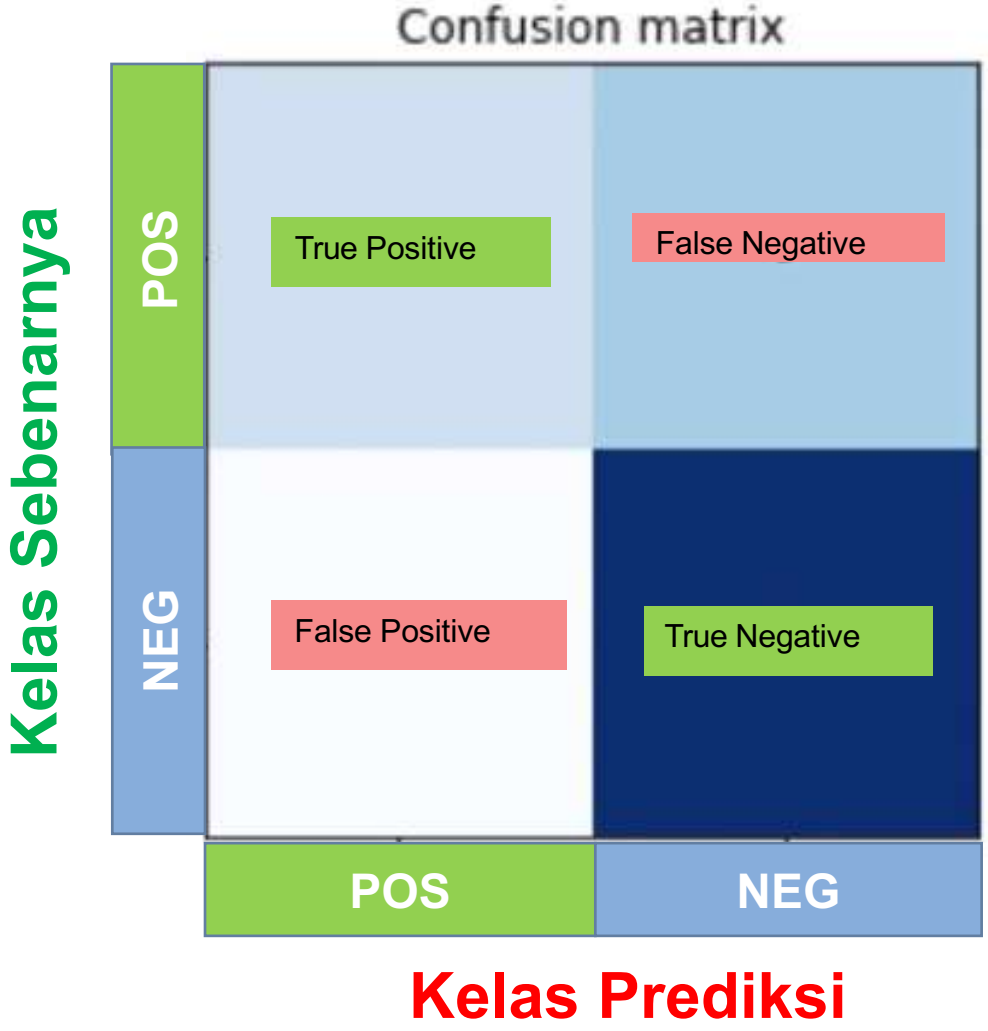
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0
5	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1



Confusion Matrix

- Cara membaca Confusion Matrix
- **True Positive:**
  - Diprediksi ***True***
  - Class sebenarnya ***True***
- **True Negative**
  - Diprediksi ***False***
  - Class sebenarnya ***False***

- **False Negative:**
  - Diprediksi ***False***
  - Class sebenarnya ***True***
- **False Positive:**
  - Diprediksi ***True***
  - Class sebenarnya ***False***

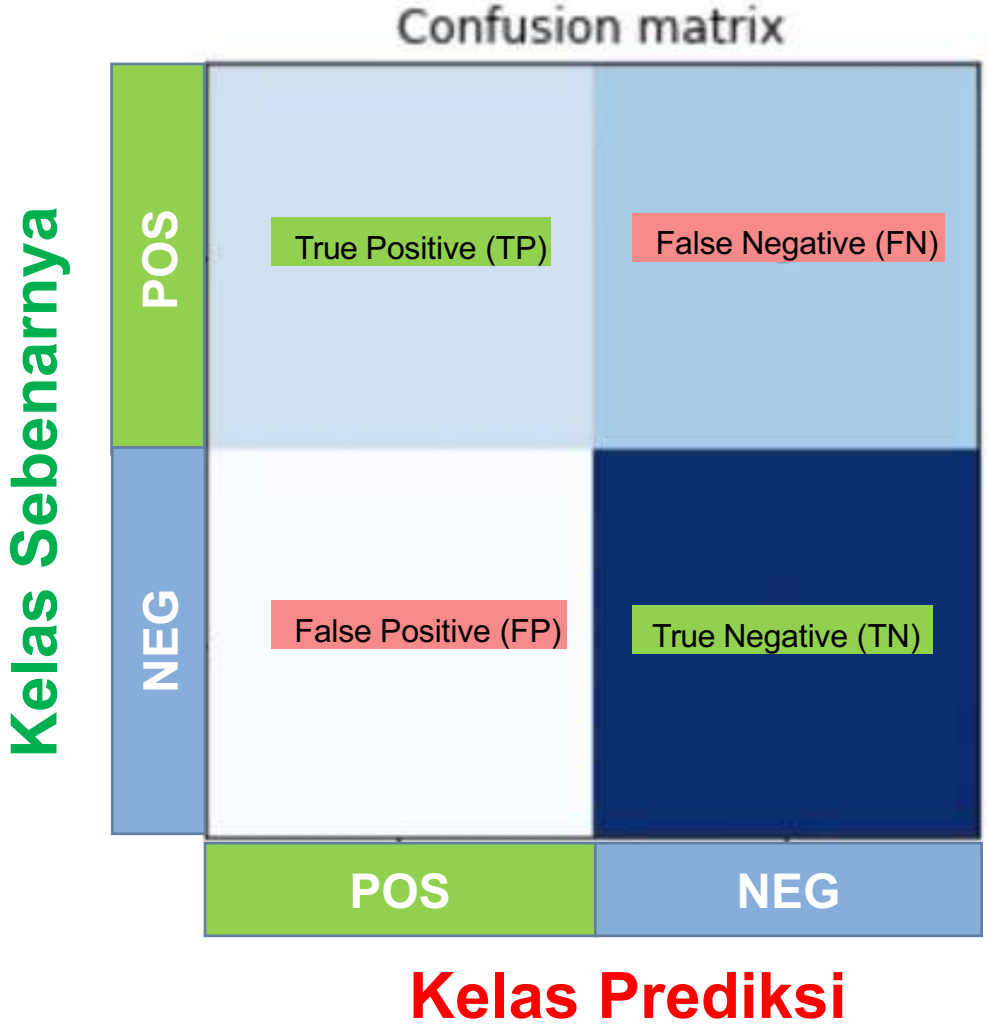


Confusion Matrix Evaluation Metric

Confusion Matrix Evaluation Metric:

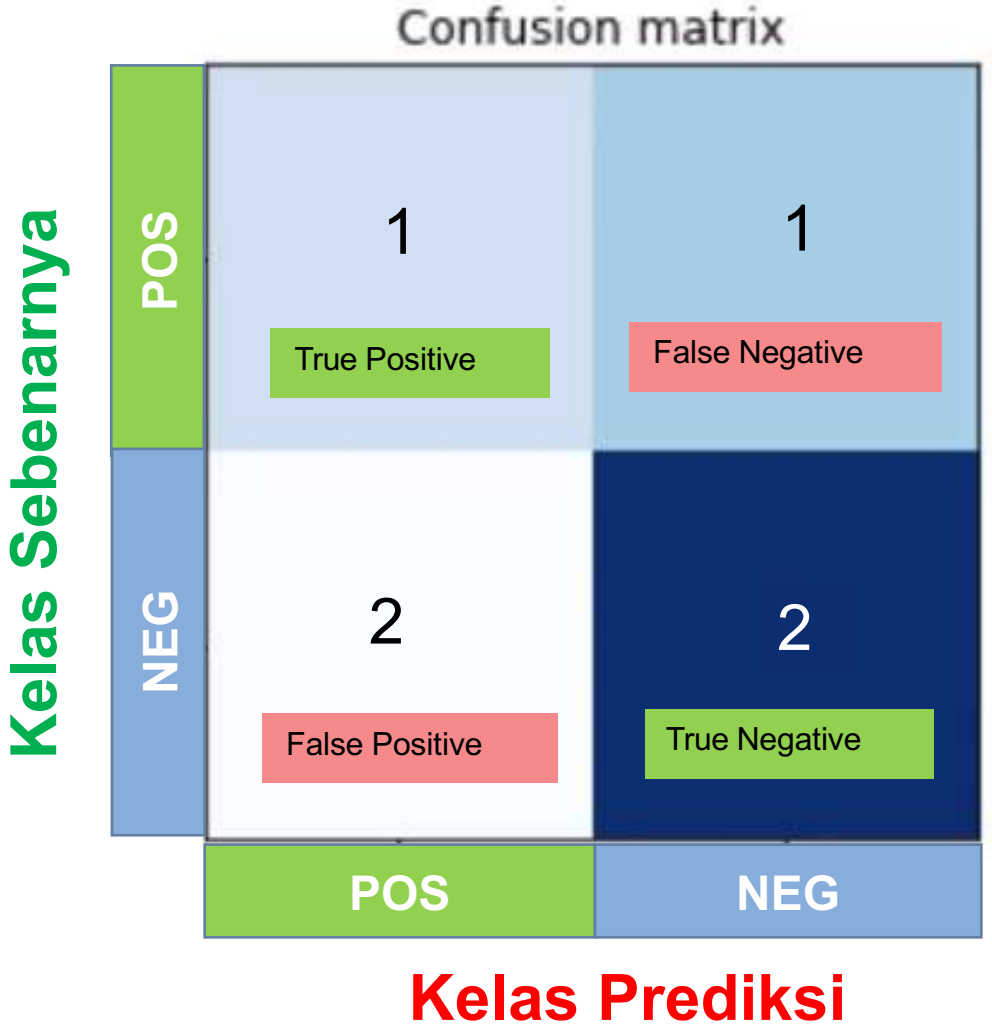
- Akurasi = $\frac{TP + TN}{TP + TN + FP + FN}$
- Precision = $\frac{TP}{TP + FP}$
- Recall = $\frac{TP}{TP + FN}$
- F1-Score = $2 \times \frac{Precision \times Recall}{(Precision + Recall)}$

Harmonic Average dari Prec. & Rec.



Contoh Confusion Matrix

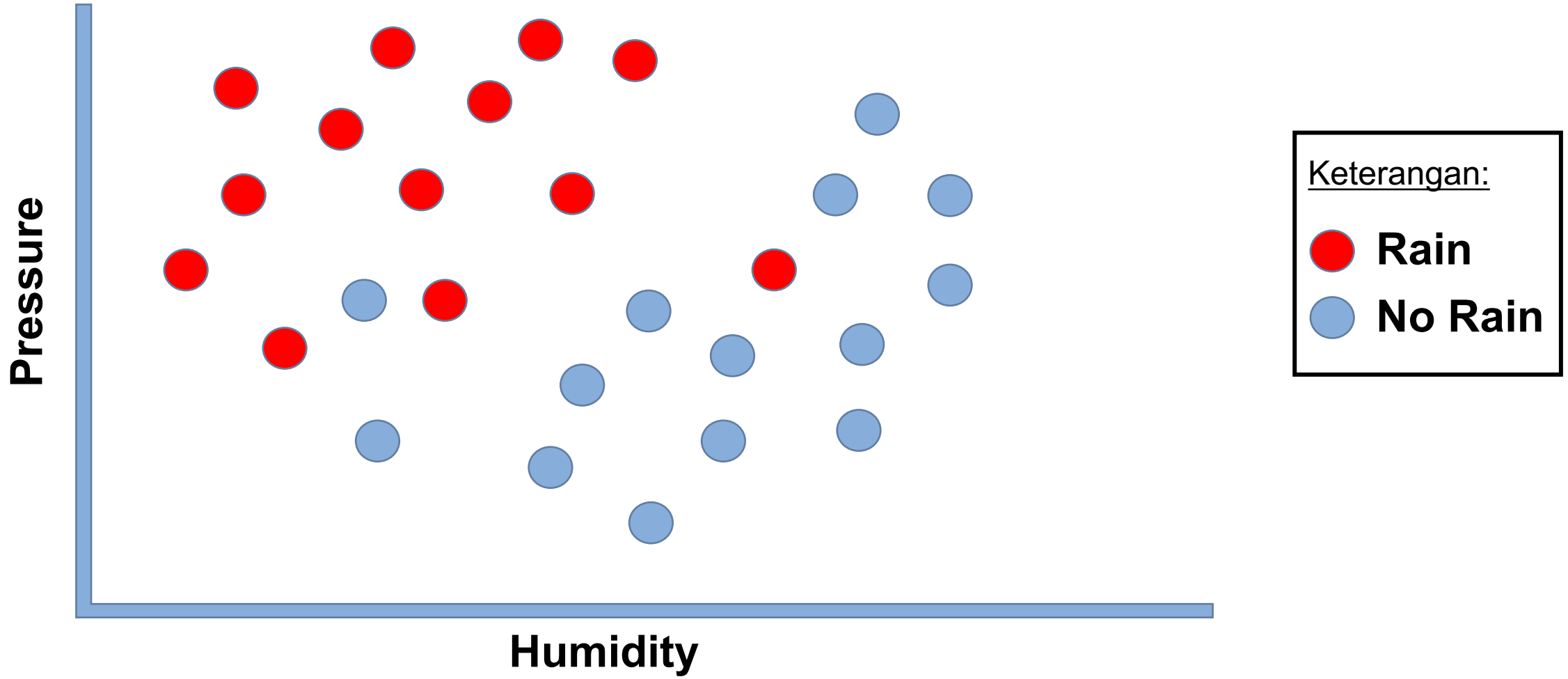
Data Test	Kelas Sebenarnya	Kelas Prediksi
1	Pos	Neg
2	Pos	Pos
3	Neg	Neg
4	Neg	Pos
5	Pos	Neg
5	Neg	Neg



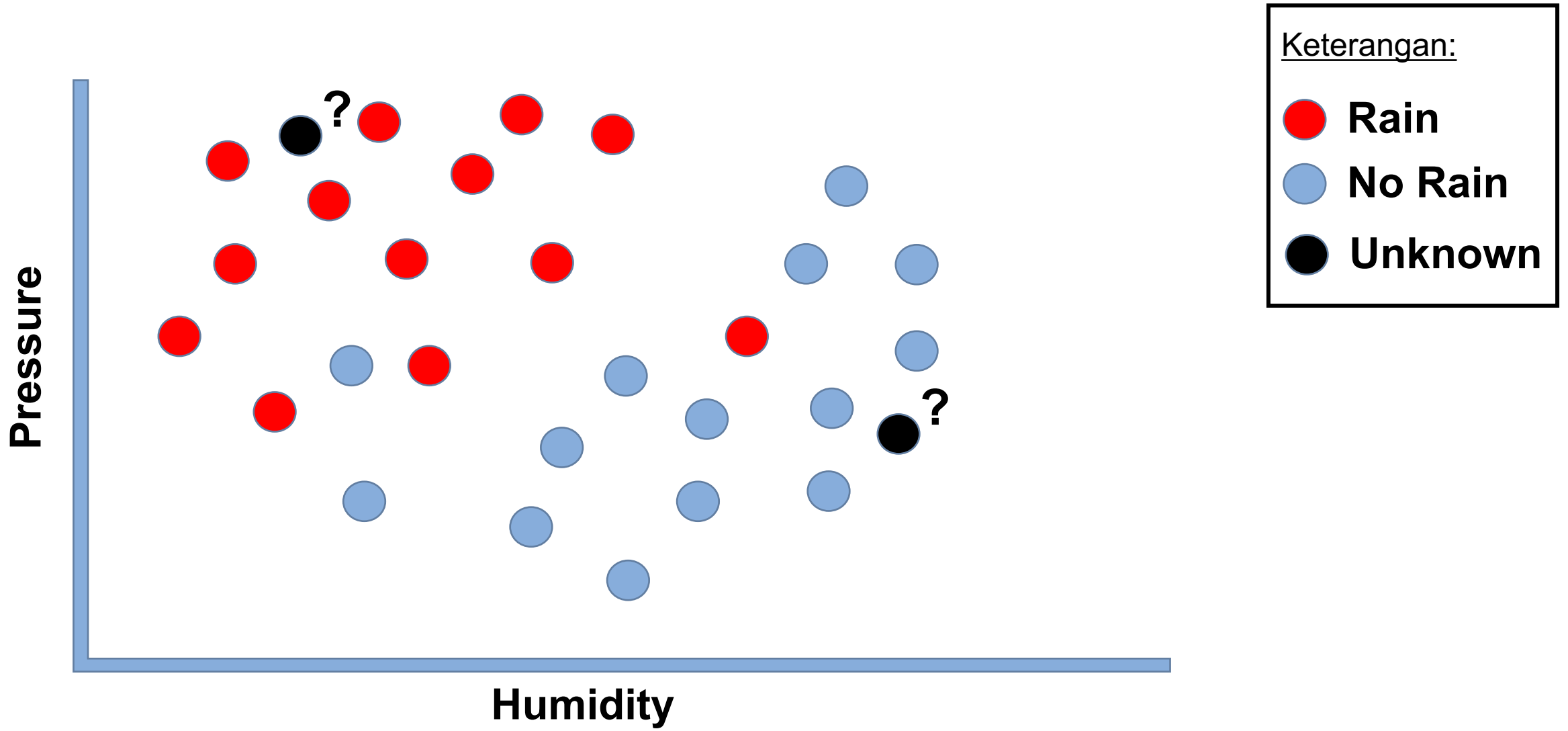
Metode/Algoritme Pembelajaran Untuk Klasifikasi

Contoh Proses Klasifikasi

Misalkan dimiliki dataset dengan visualisasi seperti berikut:

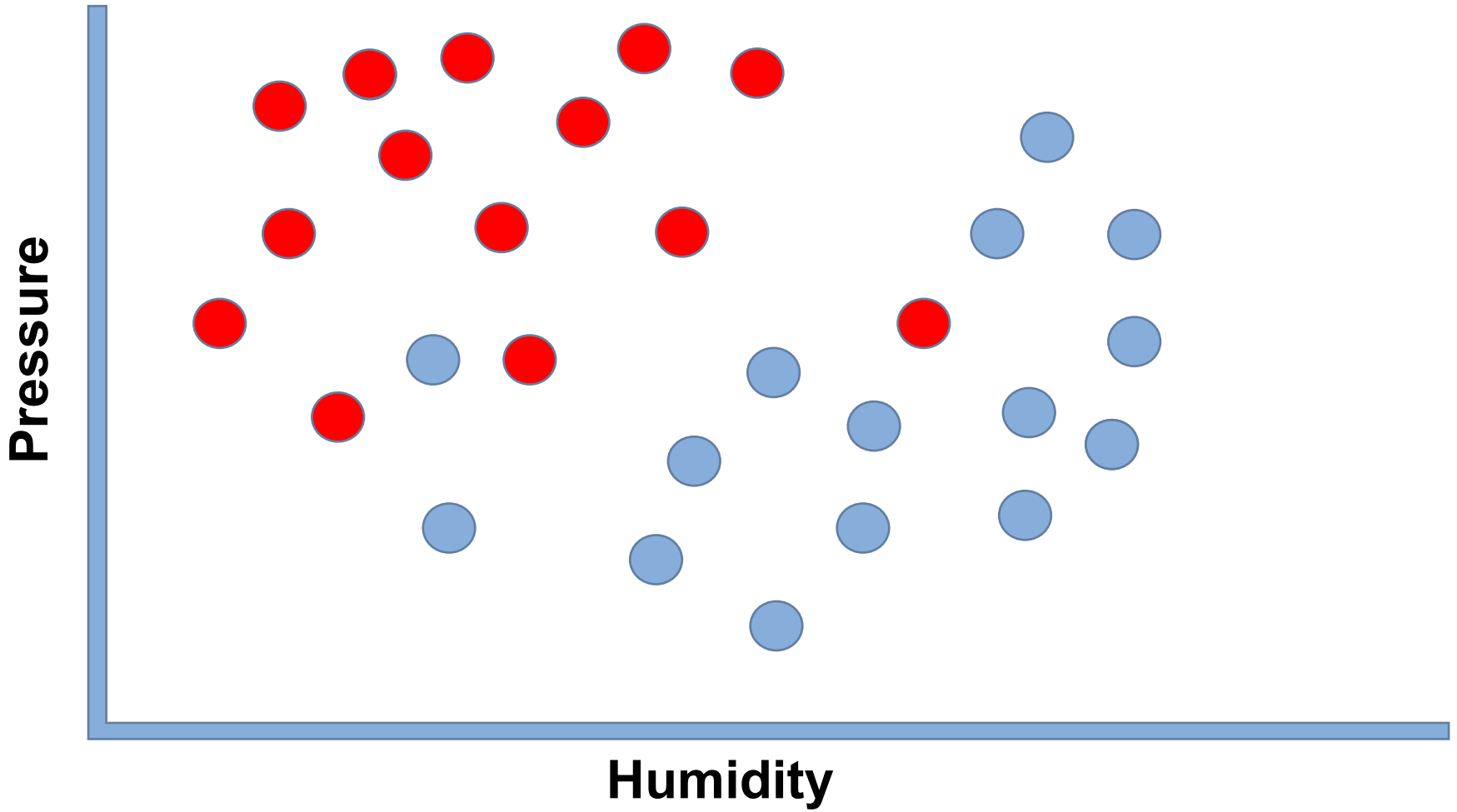


Klasifikasi Data Baru

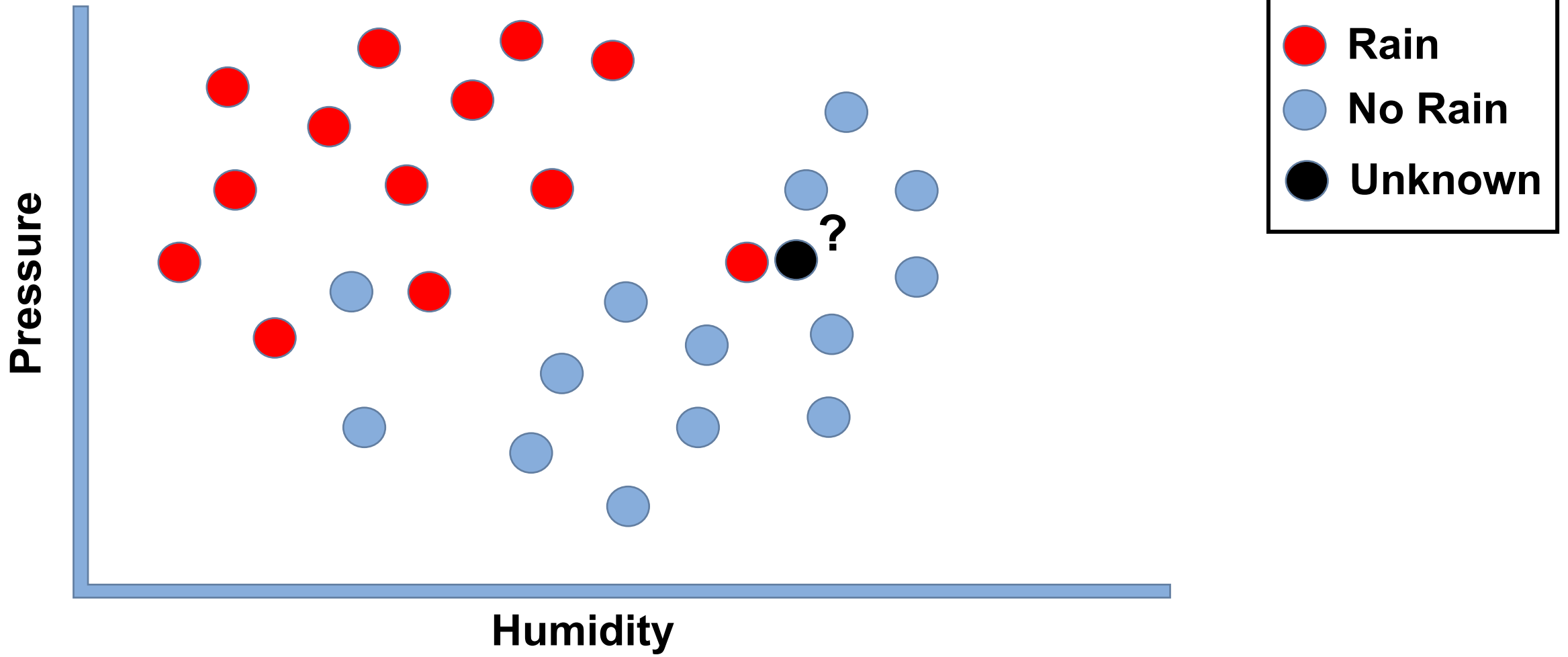


Nearest-Neighbor Classification

Nearest-Neighbor merupakan metode klasifikasi yang memetakan input ke dalam kelas dari data point terdekat dari input.

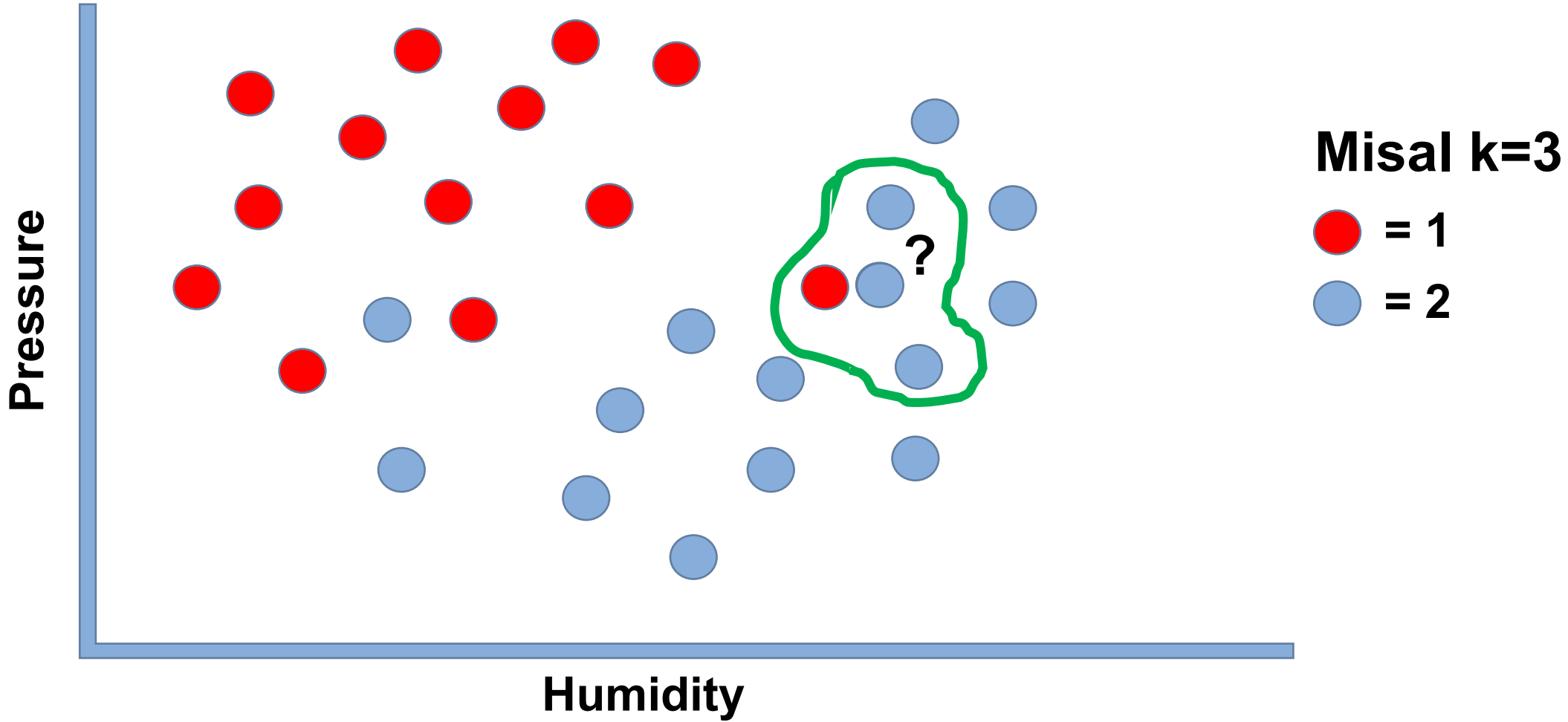


Klasifikasi Data Baru (2)



k-Nearest-Neighbor (*k*NN) Classification

k-Nearest-Neighbor merupakan metode klasifikasi yang memetakan input ke dalam kelas yang paling umum (mayoritas) dari "**k**" data point terdekat dari input.



Algoritme K-Nearest Neighbor

1. Diberikan sebuah data tidak terklasifikasi p, dan kumpulan data training P yang telah dilengkapi label classnya .
2. Pilih nilai dari jumlah ketetanggan K.
3. Hitung jarak antara p ke seluruh data yang ada dalam P.
4. Ambil K observasi yang merupakan data terdekat dengan p.
5. Klasifikasikan data tersebut dengan mayoritas class dari K-Tetangga terdekatnya.

KNN Lazy Learner

- K-NN tergolong sebagai Lazy Learner karena pembelajaran nya hanya membandingkan suatu data-baru dengan sekumpulan data latih yang sudah ada

- K-NN baru bekerja mengukur jarak ketika dilakukan klasifikasi.

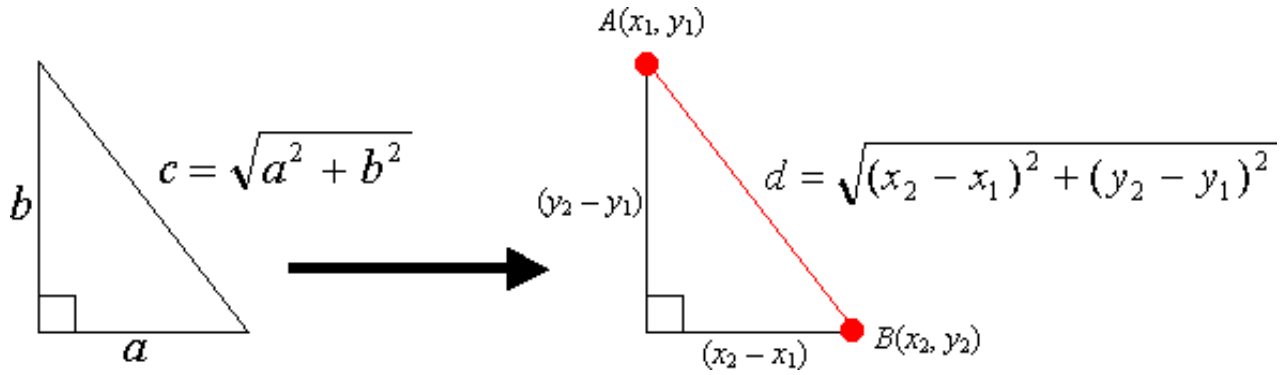
- Malas Belajar berarti tidak perlu untuk belajar atau melatih model dan semua titik data yang digunakan pada saat prediksi.

Jarak denga Tetangga

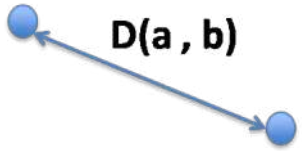
- Untuk menemukan titik yang terdekat yang terdekat, harus menemukan jarak antara titiknya.
- Jarak antara titik dapat menggunakan ukuran jarak:
  - Euclidean
  - Cosine
  - Hamming
  - Manhattan
  - Minkowski

K-NN dengan Euclidean Distance

Mencari Jarak Antara 2 Titik di 2D



$$D(a,b) = \sqrt{\sum_{i=1}^{n} (b_i - a_i)^2}$$



Contoh Kasus

- Terdapat beberapa data yang berasal dari survey questioner tentang klasifikasi kualitas kertas tissue apakah baik atau jelek, dengan objek training dibawah ini menggunakan dua attribute yaitu daya tahan terhadap asam dan kekuatan.

X1 = Daya tahan asam (detik)	X2 = Kekuatan (Kg/m^2)	Y = Klasifikasi
8	4	Baik
4	5	Jelek
4	6	Jelek
7	7	Baik
5	6	Jelek
6	5	Baik

Contoh Kasus

- Akan diproduksi kembali kertas tissue dengan attribute **X1=7** dan **X2=4**, tanpa harus mengeluarkan biaya untuk melakukan survey, maka dapat diklasifikasikan kertas tissue tersebut termasuk yang baik atau jelek.
-

X1 = Daya tahan asam (detik)	X2 = Kekuatan (kg/m^2)	Square distance to query distance (7,4)
8	4	$(8-7)^2 + (4-4)^2 = 1$
4	5	$(4-7)^2 + (5-4)^2 = 10$
4	6	$(4-7)^2 + (6-4)^2 = 13$
7	7	$(7-7)^2 + (7-4)^2 = 9$
5	6	$(5-7)^2 + (6-4)^2 = 8$
6	5	$(6-7)^2 + (5-4)^2 = 2$

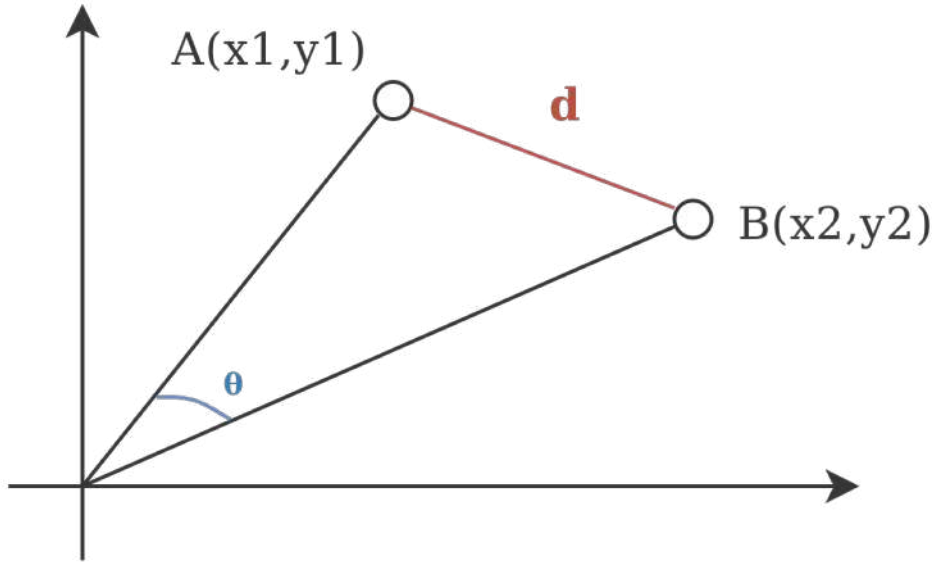
Contoh Kasus

X1= Daya tahan asam (detik)	X2= Kekuatan (Kg/m²)	Square distance to query distance (7,4)	Jarak terkecil	Apakah termasuk nearest neighbor (K)	Y= kategori nearest neighbor
8	4	$(8-7)^2 + (4-4)^2 = 1$	1	Ya	Baik
4	5	$(4-7)^2 + (5-4)^2 = 10$	5	Tidak	-
4	6	$(4-7)^2 + (6-4)^2 = 13$	6	Tidak	-
7	7	$(7-7)^2 + (7-4)^2 = 9$	4	Ya	Baik
5	6	$(5-7)^2 + (6-4)^2 = 8$	3	Ya	Jelek
6	5	$(6-7)^2 + (5-4)^2 = 2$	2	Ya	Baik

Dengan mengurutkan jarak terkecil, semisal diambil K=3, maka perbandingan nya adalah 2 (Baik) >1 (Jelek). Maka dapat disimpulkan kertas tissue dengan attribute **X1=7** dan **X2=4** masuk ke kelas **Baik.**

K-NN dengan Cosine Similarity

Mencari Kemiripan antara 2 Titik di 2D dengan Sudut yang terbentuk dari vektor 2 titik tersebut



$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}},$$

Naive Bayes

- Merupakan sebuah metode klasifikasi menggunakan metode probabilitas dan statistik yg dikemukakan oleh ilmuwan Inggris Thomas Bayes

- Merupakan metode pembelajaran klasifikasi untuk menentukan peluang terjadinya sesuatu apabila suatu kejadian lain telah terjadi

- Ciri utama dari Naïve Bayes Classifier ini adalah asumsi yg sangat kuat (naif) akan independensi dari masing-masing kondisi/kejadian.

Teorema Bayes

- Teorema bayes adalah sebuah metode untuk mencari sebuah kemungkinan kejadian baru dari kejadian-kejadian yang sudah diketahui sebelumnya.

- Antara teorema bayes dengan teori peluang terdapat hubungan yang sangat erat, karena untuk membuktikan Teorema Bayes tidak terlepas dari penggunaan teori peluang, dengan kata lain teori peluang adalah konsep dasar bagi teorema bayes.

Teorema Bayes

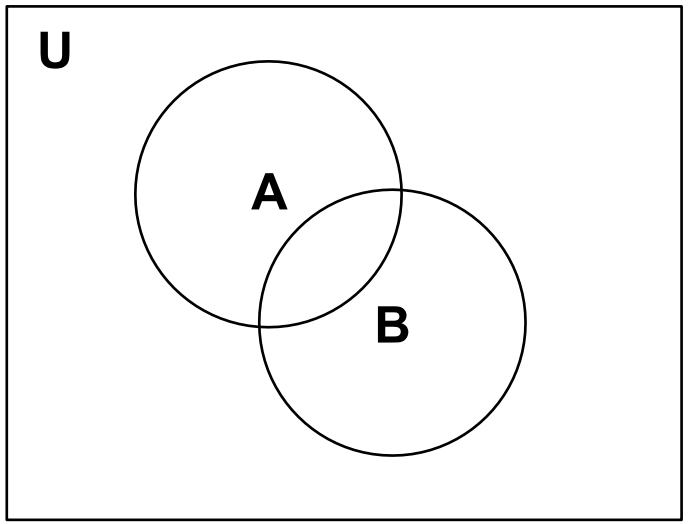


Diagram Venn dua event A dan B dalam U (semesta)

$$P(A|B) = \frac{P(A \cap B)}{P(B)}$$

$$P(B|A) = \frac{P(B \cap A)}{P(A)}$$

$$\mathbf{P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \quad ; P(B) \neq 0}$$

Teorema Bayes untuk klasifikasi

Keterangan:
V_{NB} = Kelas hasil klasifikasi Naïve Bayes
$P(v_j|a)$: probabilitas kelas v_j jika a telah terjadi

$$V_{NB} = \arg\max P(v_j|a)$$

Untuk banyak input/atribut

$$V_{NB} = \arg\max P(v_j|a_1, a_2, \ldots, a_n)$$

Asusmi Independen (naif) Naïve Bayes

Teorema Bayes untuk klasifikasi:

$$V_{NB} = \arg\max P(v_j | a_1, a_2, \ldots, a_n)$$

Keterangan:
$P(v_j)$: probabilitas kelas v_j
$P(a_i|v_j)$: probabilitas atribut a_i pada v_j

$$V_{NB} = \arg\max \frac{P(a_1, a_2, \ldots, a_n | v_j)\, P(v_j)}{P(a_1, a_2, \ldots, a_n)} \Rightarrow V_{NB} = \arg\max\ P(a_1, a_2, \ldots, a_n | v_j)\, P(v_j)$$

$$P(a_1, a_2, \ldots, a_n | v_j) = P(a_1 | v_j)\, P(a_2 | v_j, a_1)\, P(a_3 | v_j, a_1, a_2) \ldots P(a_n | v_j, a_1, a_2, \ldots, a_{n-1})$$

Asumsi Independen (naif):

$$P(a_1, a_1, \ldots, a_n | v_j) = \prod_i P(a_i | v_j)$$

Keputusan Naïve Bayes

- Keputusan Naïve Bayes ditentukan berdasarkan rumus berikut :

$$V_{NB} = \arg\max P(v_j) \prod_i P(a_i|v_j)$$

  - $P(v_j)$: probabilitas kelas v_j
  - $P(a_i|v_j)$: probabilitas atribut a_i pada v_j

- Yaitu dengan cara mengalikan semua peluang kejadian untuk setiap kelas, lalu membandingkannya untuk menentukan keputusan akhir

Contoh Kasus Naïve Bayes

- Diketahui data set berikut.
- Tentukan apakah keputusan permainan jika outlook=sunny, temp=cool, humidity=high, windy=true

outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes

outlook	temp.	humidity	windy	play
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

Contoh Pengerjaan Naïve Bayes

outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes

outlook	temp.	humidity	windy	play
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3	hot	2	2	high	3	4	false	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	true	3	3		
rainy	3	2	cool	3	1								

Contoh Model Probabilitas

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2	3	hot	2	2	high	3	4	false	6	2	9	5
overcast	4	0	mild	4	2	normal	6	1	true	3	3		
rainy	3	2	cool	3	1								

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

Proses Klasifikasi

outlook	temp.	humidity	windy	play
sunny	cool	high	true	?

$$v_{NB} = \arg\max_{v_j \in \{yes,no\}} P(v_j) \prod_i P(a_i|v_j)$$

$$= \arg\max_{v_j \in \{yes,no\}} P(v_j)P(\text{outlook} = \text{sunny}|v_j)P(\text{temp} = \text{cool}|v_j)$$
$$P(\text{humidity} = \text{high}|v_j)P(\text{windy} = \text{true}|v_j)$$

outlook			temperature			humidity			windy			play	
	yes	no		yes	no		yes	no		yes	no	yes	no
sunny	2/9	3/5	hot	2/9	2/5	high	3/9	4/5	false	6/9	2/5	9/14	5/14
overcast	4/9	0/5	mild	4/9	2/5	normal	6/9	1/5	true	3/9	3/5		
rainy	3/9	2/5	cool	3/9	1/5								

$P(\text{play} = \text{yes}) = 9/14 \quad P(\text{play} = \text{no}) = 5/14$

$P(\text{yes})P(\text{sunny}|\text{yes})P(\text{cool}|\text{yes})P(\text{high}|\text{yes})P(\text{true}|\text{yes})$
$= 9/14 \cdot 2/9 \cdot 3/9 \cdot 3/9 \cdot 3/9 = 0.0053$
$P(\text{no})P(\text{sunny}|\text{no})P(\text{cool}|\text{no})P(\text{high}|\text{no})P(\text{true}|\text{no})$
$= 5/14 \cdot 3/5 \cdot 1/5 \cdot 4/5 \cdot 3/5 = 0.0206$

$$v_{NB} = \arg\max_{v_j \in \{yes,no\}} P(v_j)P(\text{sunny}|v_j)P(\text{cool}|v_j)P(\text{high}|v_j)P(\text{true}|v_j)$$

$$= \text{no}$$