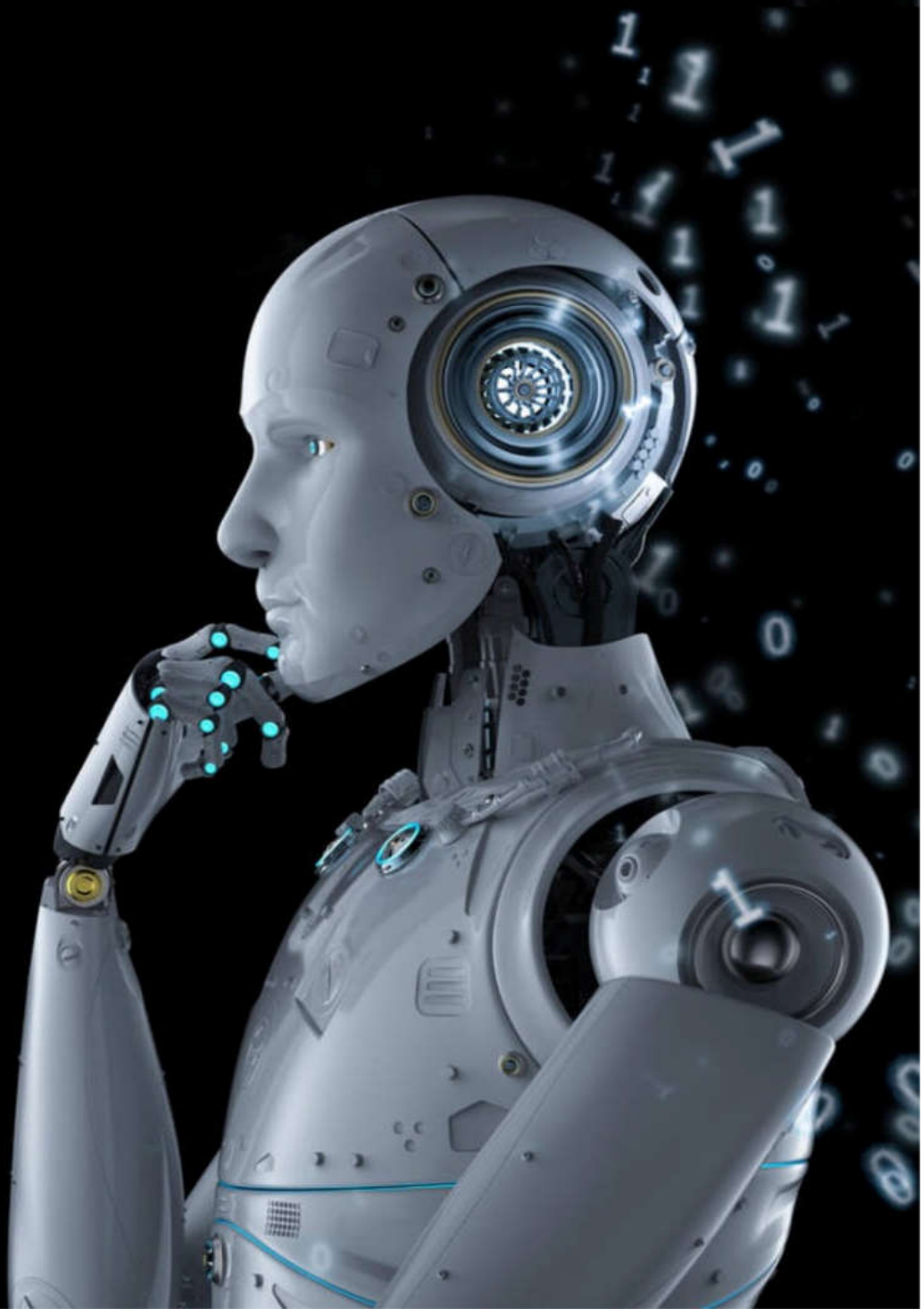
MODUL PEMBELAJARAN
MACHINE LEARNING



Hairani, M.Eng

LEMBAR PENGESAHAN

BAHAN AJAR

Judul	: Modul Pembelajaran Machine Learning
Program Studi	: S1 Ilmu Komputer
Disusun Oleh	: Hairani, S.Kom., M.Eng.
NIDN	: 0810069201
Jabatan	: Dosen Ilmu Komputer

Disahkan Oleh :

Mataram, 1 Maret 2023

Penyusun	**Kaprodi**
Hairani, S.Kom., M.Eng.	**Dr. Dadang Priyanto., M.Kom**
NIDN. 0810069201	NIDN. 0825117401

KATA PENGANTAR

Segala Puji bagi allah SWT yang telah melimpahkan nikmatnya sehingga modul pembelajaran machine learning ini bisa diselesaikan engan baik. Diharapkan Modul ini dapat membantu mahasiswa dalam memahami mata kuliah machine learning.

Pada kesempatan ini, penulis menyampaikan banyak terima kasih kepada berbagai pihak yang ikut berkontribusi dalam penyelesaian modul ini.

Mataram, 1 Maret 2023

Penyusun

Hairani, S.Kom., M.Eng.

DAFTAR ISI

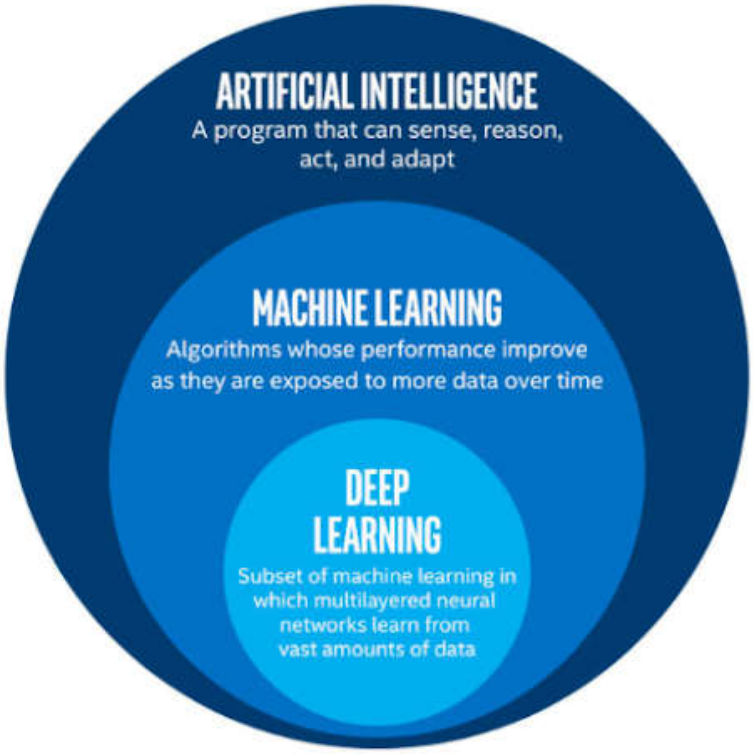
LEMBAR PENGESAHAN BAHAN AJAR ii
KATA PENGANTAR iii
DAFTAR ISI iv
BAB I PENGANTAR MACHINE LEARNING 1
BAB II REGRESI LINEAR 6
BAB III NAÏVE BAYES 11
BAB IV METODE K-NEAREST NEIGHBOAR 18
BAB V METODE C4.5 22
BAB VI METODE ARTIFICIAL NEURAL NETWORK 30
BAB VII CLUSTERING METHOD 46
BAB VIII ASSOCIATION RULE 59
BAB IX PRINCIPAL COMPONENT ANALISYS 66
BAB X RECOMMENDATION SYSTEM 72
BAB XI METRIK EVALUASI KINERJA 78

BAB I
PENGANTAR MACHINE LEARNING

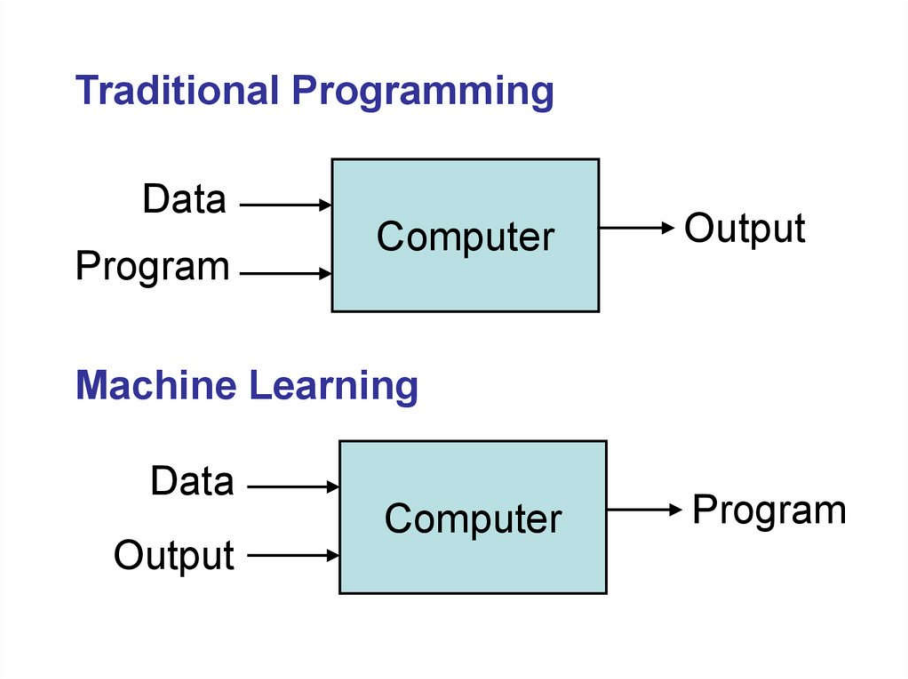
1.1 Pengertian Machine Learning

Pembelajaran mesin adalah subbidang kecerdasan buatan (AI). Tujuan pembelajaran mesin secara umum adalah untuk memahami struktur data dan memasukkan data tersebut ke dalam model yang dapat dipahami dan dimanfaatkan oleh orang-orang. Pembelajaran mesin adalah aplikasi AI yang memberikan sistem kemampuan untuk belajar sendiri dan meningkatkan dari pengalaman tanpa diprogram secara eksternal. Jika komputer Anda memiliki pembelajaran mesin, komputer tersebut mungkin dapat memainkan bagian yang sulit dari suatu permainan atau memecahkan persamaan matematika yang rumit untuk Anda. Meskipun pembelajaran mesin adalah bidang dalam ilmu komputer, ini berbeda dari pendekatan komputasi tradisional. Dalam komputasi tradisional, algoritma adalah sekumpulan instruksi yang diprogram secara eksplisit yang digunakan oleh komputer untuk menghitung atau memecahkan masalah. Alih-alih, algoritme pembelajaran mesin memungkinkan komputer melatih input data dan menggunakan analisis statistik untuk menghasilkan nilai yang termasuk dalam rentang tertentu. Oleh karena itu, pembelajaran mesin memfasilitasi komputer dalam membangun model dari data sampel untuk mengotomatiskan proses pengambilan keputusan berdasarkan input data.

Dalam pembelajaran mesin, tugas umumnya diklasifikasikan ke dalam kategori luas. Kategori ini didasarkan pada bagaimana pembelajaran diterima atau bagaimana umpan balik pada pembelajaran diberikan kepada sistem yang dikembangkan. Dua dari metode pembelajaran mesin yang paling banyak diadopsi adalah pembelajaran terawasi (*supervised learning*) yang melatih algoritme berdasarkan contoh data input dan output yang diberi label oleh manusia, dan pembelajaran tanpa pengawasan (*unsupervised learning*)yang menyediakan algoritme tanpa data berlabel untuk memungkinkannya menemukan struktur di dalam inputnya.



Gambar 1.1 Machine Learning vs AI and Deep Learning



Gambar 1.2 Perbedaan Traditional Programming dan Machine Learning

Aplikasi machine learning dalam dunia nyata:

1. Keuangan komputasi, untuk penilaian kredit dan perdagangan algoritmik
2. Pemrosesan gambar dan visi komputer, untuk pengenalan wajah, deteksi gerakan, dan deteksi objek
3. Biologi komputasi, untuk deteksi tumor, penemuan obat, dan pengurutan DNA
4. Produksi energi, untuk peramalan harga dan beban
5. Otomotif, kedirgantaraan, dan manufaktur, untuk pemeliharaan prediktif
6. Pemrosesan bahasa alami

1.2 Cara Machine Learning Bekerja

Proses pembelajaran mesin mencakup langkah-langkah berikut:

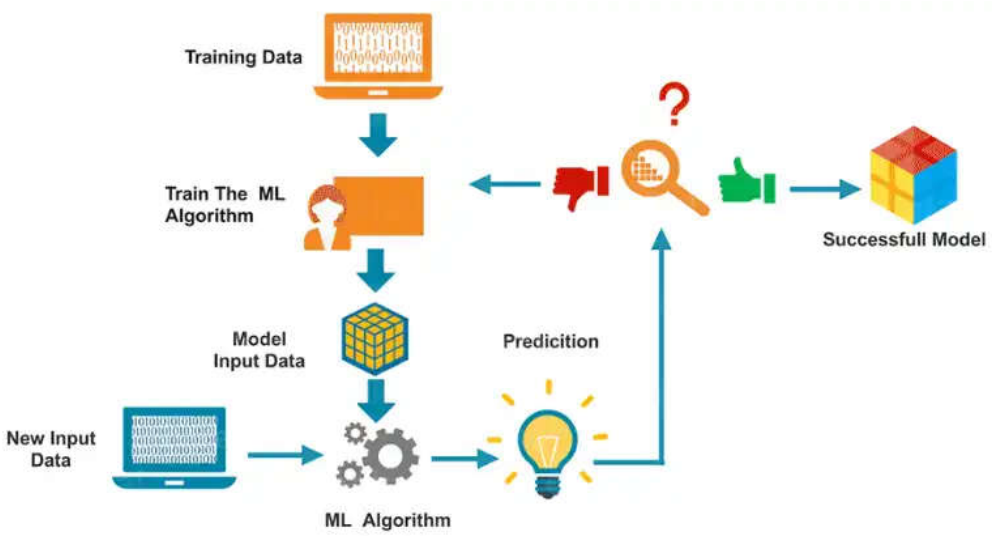
Pengumpulan Data – Langkah utama dari proses Pembelajaran Mesin adalah mengumpulkan informasi yang relevan dari berbagai sumber

Persiapan Data – Setelah semua data dikumpulkan, perlu diidentifikasi, disortir, dan diklasifikasikan sebelum dianalisis. Teknik penyiapan data bergantung pada jenis tugas yang akan dilakukan oleh aplikasi Machine Learning.

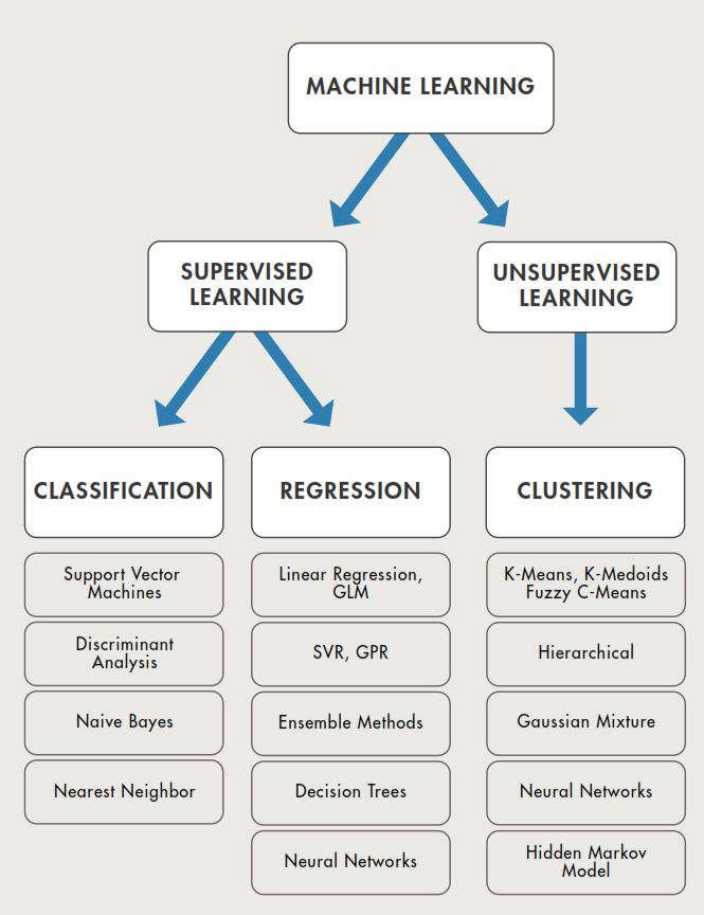
Pelatihan – Tahap ini melibatkan pelatihan mesin untuk belajar mandiri dari data yang dianalisis. Algoritma pembelajaran dibuat berdasarkan berbagai parameter dan hasil yang diharapkan dari aplikasi.

Evaluasi – Pada langkah ini, aplikasi Machine Learning diuji untuk mengevaluasi kinerjanya dan juga mengidentifikasi bug serta menemukan area perbaikan

Fine Tuning – Membuat aplikasi Machine Learning adalah proses yang berkelanjutan. Seiring berkembangnya teknik persiapan dan analisis data, algoritme dan model aplikasi Pembelajaran Mesin perlu disesuaikan.



Gambar 1.3 Cara Bekerja Machine Learning



Gambar 1.4 Teknik Machine Leraning

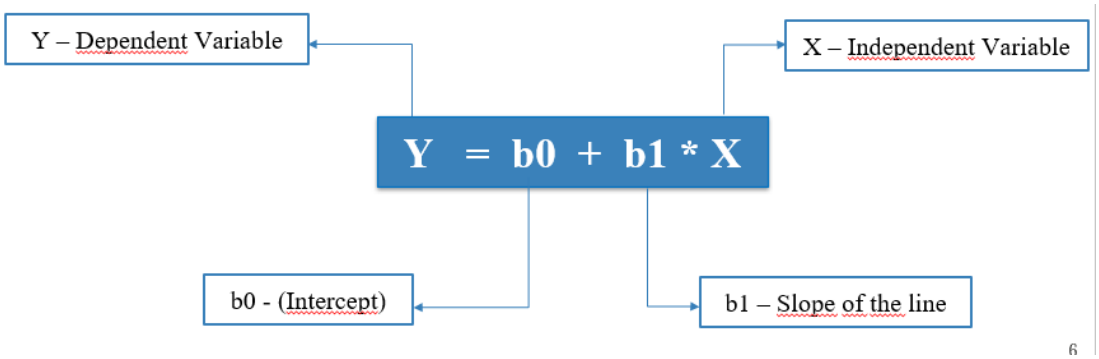
1.3 Tugas

1. Sebutkan pengertian machine learning dengan Bahasa sendri!
2. Sebutkan dan jelaskan perbedaan supervised learning dan unsupervised learning!
3. Sebutkan Contoh machine learning dalam bidang pendidikan!

BAB II
REGRESI LINEAR

2.1 Pengertian Regresi Linear

Regresi linier sederhana berguna untuk menemukan hubungan antara dua variabel kontinu. Salah satunya adalah prediktor atau variabel independen dan lainnya adalah respon atau variabel dependen. Itu mencari hubungan statistik tetapi bukan hubungan deterministik. Adapun rumus regresi linear sederhana ditunjukkan seperti berikut:



2.2 Studi Kasus

Carilah b0, b1 menggunaan metode linier regresi sederhana menggunakan dataset pada Tabel 2.1.

Tabel 2.1 Dataset Harga Rumah

Luas rumah (x)	Harga (y)
2104	400
1600	330
2400	369
1416	232
3000	540

Adapun rumus yang bisa digunakan untuk menyelesaikan kasus diatas menggunakan persamaan (2.1), (2.2), dan (2.3).

$$b_0 = \overline{Y} - (b_1 * \overline{X}) \quad (2.1)$$

$$b_1 = \frac{\sum (X - \bar{X})(Y - \bar{Y})}{\sum (X - \bar{X})^2} \quad (2.2)$$

$$Y = b_0 + b_1 X \quad (2.3)$$

Penyelesaian

a. Gunakan tabel bantu seperti pada Tabel 2.2

Tabel 2.1 Tabel Bantu

	X	Y	$(X-\bar{X})$	$(Y-\bar{Y})$	$(X-\bar{X})*(Y-\bar{Y})$	$(X-\bar{X})^2$
	2104	400	0	25.8	0	0
	1600	330	-504	-44.2	22276.8	254016
	2400	369	296	-5.2	-1539.2	87616
	1416	232	-688	-142.2	97833.6	473344
	3000	540	896	165.8	148556.8	802816
Total	**10520**	**1871**			**267118**	**1617792**
Average	**2104**	**374.2**				

b. Mencari nilai b0 dan b1

$$b_1 = \frac{\sum (X - \bar{X})(Y - \bar{Y})}{\sum (X - \bar{X})^2} = \frac{267118}{1617792} = 0{,}165$$

$$b_0 = \bar{Y} - (b_1 * \bar{X})$$
$$= 374{,}2 - (0{,}165 * 2104)$$
$$= 374{,}2 - 347{,}16$$
$$= 27{,}04$$

$$Y = b_0 + b_1 X$$
$$= 27.04 + 0{,}165X$$

Carilah b0, b1, b1 menggunaan metode linier regresi berganda menggunakan dataset pada Tabel 2.3.

Tabel 2.3 Dataset Waktu Pengantaran

Lampu **X1**	Jarak **X2**	Waktu **Y**
2	0.5	1
2	1	1.5
3	2	2
3	1.5	1.5
3	2.5	2.5

Adapun rumus yang bisa digunakan untuk menyelesaikan kasus diatas menggunakan persamaan (2.4), (2.5), (2.6) dan (2.7).

$$Y = b_0 + b_1 X_1 + b_2 X_2 \tag{2.4}$$

$$nb_0 + b_1 \sum X_1 + \ b_2 \sum X_2 = \sum Y \tag{2.5}$$

$$b_0 \sum X_1 + b_1 \sum X_1^{\,2} + b_2 \sum (X_1 X_2) = \sum X_1 Y \tag{2.6}$$

$$b_0 \sum X_2 + b_1 \sum X_1 X_2 + b_2 \sum X_2^{\,2} = \sum X_2 Y \tag{2.7}$$

Penyelesaian

a. Gunakan tabel bantu seperti pada Tabel 2.4

Tabel 2.4 Tabel Bantu

Lampu **X1**	Jarak **X2**	Waktu **Y**	X_1^2	$X_2^{\,2}$	$(X_1\ X_2\ )$	$(X_1\ Y)$	$(X_2\ Y)$
2	0.5	1	4	0.25	1	2	0.5
2	1	1.5	4	1	2	3	1.5
3	2	2	9	4	6	6	4
3	1.5	1.5	9	2.25	4.5	4.5	2.25
3	2.5	2.5	9	6.25	7.5	7.5	6.25
$\sum X_1$	$\sum X_2$	$\sum Y$	$\sum X_1^2$	$\sum X_2^{\,2}$	$\sum (X_1^2 X_2^{\,2})$	$\sum (X_1^2 Y)$	$\sum (X_2^{\,2}\ Y)$
13	**7.5**	**8.5**	35	13.75	21	23	14.5

b.Mencari nilai b0, b1, dan b2

$$nb_0 + b_1 \sum X_1 + \ b_2 \sum X_2 = \sum Y$$

$$(5)b_0 + b_1(13) + \ b_2(7.5) = 8.5 \qquad (1)$$

$$b_0\sum X_1 + b_1\sum X_1^2 + b_2\sum (X_1X_2) = \sum X_1Y$$
$$(13)b_0 + (35)b_1 + (21)b_2 = 23 \qquad (2)$$

$$b_0\sum X_2 + b_1\sum X_1X_2 + b_2\sum X_2^2 = \sum X_2Y$$
$$(7.5)b_0 + (21)b_1 + (13.75)b_2 = 14.5 \qquad (3)$$

$$(5)b_0 + b_1(13) + b_2(7.5) = 8.5 \qquad (1)$$
$$(13)b_0 + (35)b_1 + (21)b_2 = 23 \qquad (2)$$
$$(7.5)b_0 + (21)b_1 + (13.75)b_2 = 14.5 \qquad (3)$$

$$(5)b_0 + b_1(13) + b_2(7.5) = 8.5 \quad |13| \quad 65b_0 + 169b_1 + 97.5b_2 = 110.5$$
$$(13)b_0 + (35)b_1 + (21)b_2 = 23 \quad |5| \quad 65b_0 + 175b_1 + 105b_2 = 115$$
$$6b_1 + 7.5b_2 = 4.5$$

$$(13)b_0 + (35)b_1 + (21)b_2 = 23 \quad |7.5| \quad 97.5b_0 + 262b_1 + 157.5b_2 = 172.5$$
$$(7.5)b_0 + (21)b_1 + (13.75)b_2 = 14.5 \quad |13| \quad 97.5b_0 + 273b_1 + 178.75b_2 = 188.5$$
$$10.5b_1 + 21.25b_2 = 16$$

$$6b_1 + 7.5b_2 = 4.5 \qquad (4)$$
$$10.5b_1 + 21.25b_2 = 16 \qquad (5)$$

$$(5)b_0 + b_1(13) + b_2(7.5) = 8.5 \qquad (1)$$
$$(13)b_0 + (35)b_1 + (21)b_2 = 23 \qquad (2)$$
$$(7.5)b_0 + (21)b_1 + (13.75)b_2 = 14.5 \qquad (3)$$
$$6b_1 + 7.5b_2 = 4.5 \qquad (4)$$
$$10.5b_1 + 21.25b_2 = 16 \qquad (5)$$

$$6b_1 + 7.5b_2 = 4.5 \quad |10| \quad 63b_1 + 78.75b_2 = 47.25$$
$$10.5b_1 + 21.25b_2 = 16 \quad |6| \quad 63b_1 + 127.5b_2 = 96$$
$$48.75b_2 = 48.75$$
$$b_2 = 1$$

$6b_1 + 7.5b_2 = 4.5 \quad (4)$

$6b_1 = 4.5 - 7.5$

$b_1 = -0.5$

$5b_0 + 13b_1 + 7.5b_2 = 8.5$

$5b_0 + 13(-0.5) + 7.5(1) = 8.5$

$b_0 = 1.7$

Sehingga persamaan yang didapatkan adalah

$$Y = b_0 + b_1 X_1 + b_2 X_2$$

$$Y = 1.7 + (-0.5) X_1 + (1) X_2$$

$$Y = 1.7 - 0.5 X_1 + X_2$$

2.3 Tugas

Carilah b0, b1, dan b2 menggunaan metode linier regresi berganda, serta tulis persamaan menggunakan dataset pada Tabel 2.5?

Tabel 2.5 Dataset Keputusan Konsumen

Promosi (x_1)	Harga (x_2)	Keputusan Konsumen (y)
10	7	23
2	3	7
4	2	15
6	4	17
8	6	23
7	5	22
4	3	10
6	3	14
7	4	20
6	3	19

BAB III
NAÏVE BAYES

3.1 Konsep Naïve Bayes

Algoritma Naive Bayes merupaka metode klasifikasi berdasarkan konsep probabilitas yang dikemukakan oleh Thomas Bayes. Algoritma Naive Bayes memprediksi peluang masa depan berdasarkan pengalaman sebelumya sehigga dikenal sebagai Teorema Bayes Ciri utama dari metode NB adalah asumsi yang sangat kuat (naif) akan indepedensi dari masing-masing kondisi atau kejadian. Adapun tahapan-tahapan dalam perhitungan metode naive bayes ditunjukkan pada Gambar 3.1.

Gambar 3.1 Tahapan Perhitungan Naïve Bayes

Formula naive bayes secara umum yang digunakan seperti pada persamaan 3.1.

$$P(H|E) = \frac{P(E|H) * P(H)}{P(E)} \quad (3.1)$$

Likelihood, Class Prior Probability, Posterior Probability, Predictor Prior Probability

Keterangan:

P(H|E) = Probabilitas hipotesis H terjadi jika evidence E terjadi.

P(E|H) = Probabilitas munculnya evidence E, jika hipotesis H terjadi.

P(H) = Probabilitas hipotesis H tanpa memandang evidance apa pun.

P(E) = Probabilitas evidance tanpa memandang apa pun.

Sedangkan formulasi yang digunakan untuk perhitungan naive bayes pada modul ini seperti persamaan 3.2.

P(H|E) = P (E1 | H) * P(E2 | H) * P (E3 | H) * P(H) (3.2)

Untuk atribut yang bernilai kontinu umumnya diasumsikan memiliki distribusi Gaussian, P(Xk|Ci) didefinisikan pada persamaan 3.3.

$$P(x_k|C_i) = \frac{1}{\sigma_{i,k}\sqrt{2\pi}} e^{-\frac{(x_k-\mu_{i,k})^2}{2\sigma_{i,k}^2}} \quad (3.3)$$

di mana μ = rata-rata dan $^{\delta}$ = deviasi standar dari nilai-nilai pada atribut Ak untuk kelas Ci.

3.2 Studi Kasus

Diberikan sebuah dataset pada Tabel 3.1 dengan jumlah 10 instance dan 4 atribut.

Tabel 3.1 Data Training Dataset Kredit

No	Refund	Marital Status	Income	Evade
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90	Yes

Berdasarkan data training pada Tabel 3.1 , apabila diketahui seorang calon kreditur dengan kriteria seperti Tabel 3.2

Tabel 3.2 Data Testing Kredit

Refund	Marital Status	Income	Evade
No	Divorce	100K	?

Penyelesaian

1. **Menghitung jumlah class (class prior probability)**

Evade			Probability	
	Yes	No	**Yes**	**No**
Total	3	7	**3/10**	**7/10**

2. Menghitung jumlah kasus yang sama dengan class yang sama Probabilitas kemunculan setiap nilai untuk atribut **Refund**

Refund	**Evade**		**Probability**	
	Yes	No	**P(Yes)**	**P(No)**
Yes	0	3	0	**3/7**
No	3	4	**3/3**	**4/7**
Total	3	7	**1**	**1**

Probabilitas kemunculan setiap nilai untuk atribut **Marital Status**

Marital Status	**Evade**		**Probability**	
	Yes	No	**P(Yes)**	**P(No)**
Single	2	2	**2/3**	**2/7**
Married	0	4	**0**	**4/7**
Divorce	1	1	**1/3**	**1/7**

Total	3	7	**1**	**1**

Probabilitas kemunculan setiap nilai untuk atribut **Income**

Karena atrubut **Income** adalah atribut kontinue (numerik) maka penyelsaiannya menggunakan metode Distribusi Gaussian dengan formula 5.2.

$$P(x_k|C_i) = \frac{1}{\sigma_{i,k}\sqrt{2\pi}} e^{-\frac{(x_k-\mu_{i,k})^2}{2\sigma_{i,k}^2}}$$

Untuk mencari rata-rata dan standar deviasinya menggunakan rumus berikut ini:

Rumus Rata-rata :

$$\mu = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Rumus Standar Deviasi:

$$\sigma = \sqrt{\frac{\sum(X-\mu)^2}{n-1}}$$

Income	Yes	No
1	95K	125K
2	85K	100K
3	90K	70K
4		120K
5		60K
6		220K
7		75K
Rata-rata (μ)	**90**	**110**

Standar Deviasi ($^{\delta}$)	**5**	**54,54**

Rata-rata:

$$\mu_{yes} = \frac{(95+85+90)}{3} = 90$$

$$\mu_{No} = \frac{(125+100+70+120+60+220+75)}{7} = 110$$

Standar Deviasi:

$$\delta_{Yes} = \sqrt{\frac{(95-90)^2+(85-90)^2+(90-90)^2}{(3-1)}}$$
$$= \sqrt{\frac{(5)^2+(-5)^2+(0)^2}{2}}$$
$$= \sqrt{\frac{50}{2}} = 5$$

$$\delta_{No} = \sqrt{\frac{(125-110)^2+(100-110)^2+(70-110)^2+(120-110)^2 + (60-110)^2+(220-110)^2+(75-110)^2}{(7-1)}}$$
$$= \sqrt{\frac{(15)^2+(-10)^2+(-40)^2+(10)^2+(-50)^2+(-110)^2+(-35)^2)}{(6)}}$$
$$= \sqrt{\frac{17850}{6}} = 54,54$$

Distribusi Gaussian:

$$P(Income = 100 \mid Yes) = \frac{1}{(5)\sqrt{(2(3,14)}} e^{-\frac{(100-90)^2}{2(5)^2}}$$

$$P(Income = 100 \mid Yes) = \frac{1}{(5)\sqrt{(6,28}} e^{-\frac{(10)^2}{50}} = 0,011$$

$$P(Income = 100 \mid No) = \frac{1}{(54,54)\sqrt{(2(3,14)}} e^{-\frac{(100-110)^2}{2(54,54)^2}}$$

$$P(Income = 100 \mid No) = \frac{1}{(54,54)\sqrt{6,28}} e^{-\frac{(-10)^2}{2(2975)^2}} = 0,007$$

3. Mengalikan Semua Variabel Kelas

$$P(X \mid Evade = Yes) = P(refund = No \mid Yes) * P(marital = Divorces \mid Yes) * P(Income = 100K \mid Yes)$$

$$P(X \mid Evade = Yes) = \frac{3}{3} * \frac{1}{3} * 0,011 * \frac{3}{10} = 0,0011$$

$$P(X \mid Evade = No) = P(refund = No \mid No) * P(marital = Divorces \mid No) * P(Income = 100K \mid No)$$

$$P(X \mid Evade = No) = \frac{4}{7} * \frac{1}{7} * 0,007 * \frac{7}{10} = 0,5714 * 0,1429 * 0,007 * 0,7 = 0,0004$$

4. Bandingkan Hasil Perkelas

Karena **0,0011** > **0**, **0004** sehingga kesimpulannya **diberikan kredit**

3.3 Tugas

1. Di berikan Data *Training* dan Data Testing yang masing-masing ditunjukkan pada Tabel 3.3 dan Tabel 3.4.

Tabel 5.3. Data *Training*

No	Age	Income	Student	Buy Computer (Class)
1.	35	Medium	Yes	Yes
2.	30	High	No	No
3.	40	Low	Yes	No
4.	35	Medium	No	Yes
5.	45	Low	No	Yes
6.	35	High	No	Yes
7.	35	Medium	No	No
8.	25	Low	No	No
9.	28	High	No	No
10.	35	Medium	Yes	Yes

Tabel 5.4. Data *Testing*

No	Age	Income	Student	Buy Computer

				(Class)
1.	35	Low	Yes	?

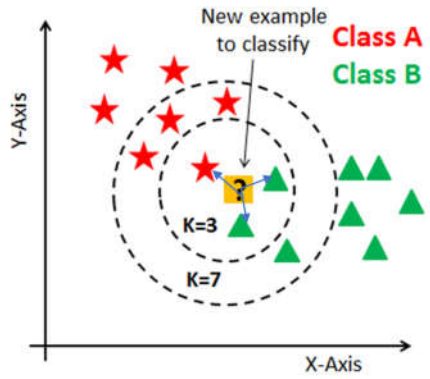
Anda diminta untuk mengklasifikasikan data *testing* pada Tabel 3.4 berdasarkan data *training* menggunakan Metode Naive Bayes. Apakah keputusanya pada Tabel 3.4 termasuk Beli Komputer atau Tidak?

BAB IV
METODE K-NEAREST NEIGHBOAR

4.1 Konsep K-NN

Metode k-NN adalah algoritme sederhana yang mengklasifikasikan kasus baru berdasarkan ukuran kesamaan (misalnya, fungsi jarak). KNN telah digunakan dalam estimasi statistik dan pengenalan pola pada awal tahun 1970-an sebagai teknik non-parametrik. Dalam KNN, K adalah jumlah tetangga terdekat. Jumlah tetangga adalah faktor penentu dalam klasifikasi kelasnya. K umumnya bilangan ganjil jika jumlah kelasnya 2.

Metode KNN bekerja berdasarkan prinsip bahwa setiap titik data yang berdekatan satu sama lain akan berada di kelas yang sama. Dengan kata lain, KNN mengklasifikasikan titik data baru berdasarkan kemiripan seeprti ilustrasi Gambar 4.1



Gambar 4.1 Cara Kerja Metode k-NN

Adapun rumus jarak yang bisa digunakan pada metode k-NN seperti persamaan (4.1), (4.2), dan (4.3).

Euclidean $$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \quad (4.1)$$

Manhattan $$\sum_{i=1}^{k}|x_i - y_i| \quad (4.2)$$

Minkowski $$\left(\sum_{i=1}^{k} \left(|x_i - y_i| \right)^q \right)^{1/q} \quad (4.3)$$

4.2 Studi Kasus

Diberikan Data Traning seperti pada Tabel 4.1.

Tabel 4.1 Data Training

Height	Weight	Label
158 cm	64 kg	male
170 cm	66 kg	male
183 cm	84 kg	male
191 cm	80 kg	male
155 cm	49 kg	female
163 cm	59 kg	female
180 cm	67 kg	female
158 cm	54 kg	female
178 cm	77 kg	female

Ujilah data testing (Tabel 4.2) berdasarkan data training yang diberikan menggunakan **K=3**

Tabel 4.2 Data Testing

Height	Weight	Label
155	70	?

Penyelesaian:

Height	Weight	Label	Distance from test instance
158 cm	64 kg	male	$\sqrt{(158-155)^2+(64-70)^2}=6.71$
170 cm	66 kg	male	$\sqrt{(170-155)^2+(64-70)^2}=21.93$
183 cm	84 kg	male	$\sqrt{(183-155)^2+(84-70)^2}=31.30$
191 cm	80 kg	male	$\sqrt{(191-155)^2+(80-70)^2}=37.36$
155 cm	49 kg	female	$\sqrt{(155-155)^2+(49-70)^2}=21.00$
163 cm	59 kg	female	$\sqrt{(163-155)^2+(59-70)^2}=13.60$
180 cm	67 kg	female	$\sqrt{(180-155)^2+(67-70)^2}=25.18$
158 cm	54 kg	female	$\sqrt{(158-155)^2+(54-70)^2}=16.28$
178 cm	77 kg	female	$\sqrt{(178-155)^2+(77-70)^2}=24.04$

Height	Weight	Label	Distance from data testing
158 cm	64 kg	male	6,71
163 cm	59 kg	female	13,60
158 cm	54 kg	female	16,28
155 cm	49 kg	female	21
170 cm	66 kg	male	21,93
178 cm	77 kg	female	24,04
180 cm	67 kg	female	25,18
183 cm	84 kg	male	31,30
191 cm	80 kg	male	37,36

Dengan menggunakan **K= 3**, dapat disimpulkan bahwa data testing yang digunakan termasuka **Label = Female.**

4.3 Tugas

1. Sebuah perusahaan makanan ringan ingin mengklasifikasikan kualitas produknya ke dalam 2 kategori, yaitu kualitas BAIK dan BURUK. Untuk menilai kualitas tersebut, digunakan 3 variabel, yaitu: kenaikan derajat keasaman (%) dan penyusutan volume. Ada 10 sampel yang digunakan untuk keperluan pengujian seperti terlihat pada Tabel 4.3.

Tabel 4.3 Data *Training*

No	Variabel		Kategori
	Kenaikan derajat keasaman (V_1)	Penyusutan volume (V_2)	
1	3	2	Baik
2	4	1	Baik
3	4	3	Baik
4	5	1	Baik
5	5	4	Baik
6	6	5	Buruk
7	7	6	Buruk
8	8	4	Buruk
9	7	2	Buruk
10	9	1	Buruk

Berdasarkan data training pada Tabel 4.3, akan dilakukan testing terhadap satu data yang ditunjukkan pada Tabel 4.4 menggnakan metode K-NN dengan **K= 5.**

Tabel 4.4. Data *Testing*

No	**Variabel**		**Kategori**
	Kenaikan derajat keasaman (V_1)	Penyusutan volume (V_2)	
1.	6	3	?

Anda diminta untuk mengklasifikasikan data *testing* pada Tabel 2 berdasarkan data *training* menggunakan Metode KNN dengan K= 5. Apakah kategori pada Tabel 4.4 termasuk Baik atau Buruk?

BAB V
METODE C4.5

5.1 Konsep C4.5

Metode C4.5 merupakan salah satu metode klasifikasi berbasis pohon keputusan. Kelebihan algoritma C4.5 dapat menghasilkan pohon keputusan yang mudah diinterprestasikan, memiliki tingkat akurasi yang dapat diterima, efisien dalam menangani atribut bertipe diskret dan dapat menangani atribut bertipe diskret dan numerik. Adapun tahapan-tahapan metode C4.5 seperti berikut:

1. Siapkan data training
2. Pilih atribut sebagai akar
3. Buat cabang untuk tiap-tiap nilai
4. Ulangi proses untuk setiap cabang sampai semua Kasus pada cabang memiliki kelas yang sama

Untuk memilih atribut akar, didasarkan pada nilai Gain (Persamaan 5.2) tertinggi dari atribut-atribut yang ada. Untuk mendapatkan nilai Gain, harus ditentukan terlebih dahulu nilai Entropy (persamaan 5.1).

$$Entropy(S) = \sum_{i=1}^{n} -pi * \log_2 pi \tag{5.1}$$

Keterangan:

S = Himpunan Kasus

n = Jumlah Partisi S

pi = Proporsi dari Si terhadap S

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} * Entropy(S_i) \tag{5.2}$$

Keterangan:

S = Himpunan Kasus

A = Atribut

n = Jumlah Partisi Atribut A

| Si | = Jumlah Kasus pada partisi ke-i

| S | = Jumlah Kasus dalam S

5.2 Studi Kasus

1. Siapkan data training

No	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
1	Sunny	Hot	High	FALSE	No
2	Sunny	Hot	High	TRUE	No
3	Cloudy	Hot	High	FALSE	Yes
4	Rainy	Mild	High	FALSE	Yes
5	Rainy	Cool	Normal	FALSE	Yes
6	Rainy	Cool	Normal	TRUE	Yes
7	Cloudy	Cool	Normal	TRUE	Yes
8	Sunny	Mild	High	FALSE	No
9	Sunny	Cool	Normal	FALSE	Yes
10	Rainy	Mild	Normal	FALSE	Yes
11	Sunny	Mild	Normal	TRUE	Yes
12	Cloudy	Mild	High	TRUE	Yes
13	Cloudy	Hot	Normal	FALSE	Yes
14	Rainy	Mild	High	TRUE	No

2. Pilih atribut sebagai sebagai akar

Penentuan atribut sebagai root harus didasarkan pada Gain tertinggi dengan terlebih dahulu menghitung nilai entropy.

NODE	ATRIBUT		JML KASUS (S)	YA (Si)	TIDAK (Si)	ENTROPY	GAIN
1	TOTAL		14	10	4		
	OUTLOOK						
		CLOUDY	4	4	0		
		RAINY	5	4	1		
		SUNNY	5	2	3		
	TEMPERATURE						
		COOL	4	0	4		
		HOT	4	2	2		
		MILD	6	2	4		
	HUMADITY						
		HIGH	7	4	3		
		NORMAL	7	7	0		
	WINDY						
		FALSE	8	2	6		
		TRUE	6	4	2		

Perhitungan Entropy Akar

Entropy Total

$$Entropy(Total) = (-\frac{4}{14} * \log_2(\frac{4}{14})) + (-\frac{10}{14} * \log_2(\frac{10}{14}))$$

$$Entropy(Total) = 0.863120569$$

Entropy (Outlook

$$Entropy(Cloudy) = (-\frac{0}{4} * \log_2(\frac{0}{4})) + (-\frac{4}{4} * \log_2(\frac{4}{4})) = 0.000000000$$

$$Entropy(Rainy) = (-\frac{1}{5} * \log_2(\frac{1}{5})) + (-\frac{4}{5} * \log_2(\frac{4}{5})) = 0.721928095$$

$$Entropy(Sunny) = (-\frac{3}{5} * \log_2(\frac{3}{5})) + (-\frac{2}{5} * \log_2(\frac{2}{5})) = 0.970950594$$

Entropy (Temperature)

$$Entropy(Cool) = (-\frac{0}{4} * \log_2(\frac{0}{4})) + (-\frac{4}{4} * \log_2(\frac{4}{4})) = 0.000000000$$

$$Entropy(Hot) = (-\frac{2}{4} * \log_2(\frac{2}{4})) + (-\frac{2}{4} * \log_2(\frac{2}{4})) = 1.000000000$$

$$Entropy(Mild) = (-\frac{2}{6} * \log_2(\frac{2}{6})) + (-\frac{4}{6} * \log_2(\frac{4}{6})) = 0.918295834$$

Entropy (Humidity)

$$Entropy(High) = (-\frac{4}{7} * \log_2(\frac{4}{7})) + (-\frac{3}{7} * \log_2(\frac{3}{7})) = 0.985228136$$

$$Entropy(Normal) = (-\frac{0}{7} * \log_2(\frac{0}{7})) + (-\frac{7}{7} * \log_2(\frac{7}{7})) = 0.000000000$$

Entropy (Windy)

$$Entropy(False) = (-\frac{2}{8} * \log_2(\frac{2}{8})) + (-\frac{6}{8} * \log_2(\frac{6}{8})) = 0.811278124$$

$$Entropy(True) = (-\frac{4}{6} * \log_2(\frac{4}{6})) + (-\frac{2}{6} * \log_2(\frac{2}{6})) = 0.918295834$$

Perhitungan Gain Akar

$$Gain(Total, Outlook) = Entropy(Total) - \sum_{i=1}^{n} \frac{|Outlook_i|}{|Total|} * Entropy\ (Outlook_i)$$

$$Gain(Total, Outlook) = 0.863120569 - \left(\left(\frac{4}{14} * 0.000000000\right) + \left(\frac{5}{14} * 0.721928095\right) + \left(\frac{5}{14} * 0.970950594\right)\right)$$

$$Gain(Total, Outlook) = 0.258521037$$

$$Gain(Total, Temperature) = Entropy(Total) - \sum_{i=1}^{n} \frac{|Temperature_i|}{|Total|} * Entropy\ (Temperature_i)$$

$$Gain(Total, Temperature) = 0.863120569 - \left(\left(\frac{4}{14} * 0.000000000\right) + \left(\frac{4}{14} * 1.000000000\right) + \left(\frac{6}{14} * 0.918295834\right)\right)$$

$$Gain(Total, Temperature) = 0.183850925$$

$$Gain(Total, Humidity) = Entropy(Total) - \sum_{i=1}^{n} \frac{|Humidity_i|}{|Total|} * Entropy(Humidity_i)$$

$$Gain(Total, Humidity) = 0.863120569 - \left(\left(\frac{7}{14} * 0.985228136\right) + \left(\frac{7}{14} * 0.000000000\right)\right)$$

$$Gain(Total, Humidity) = 0.370506501$$

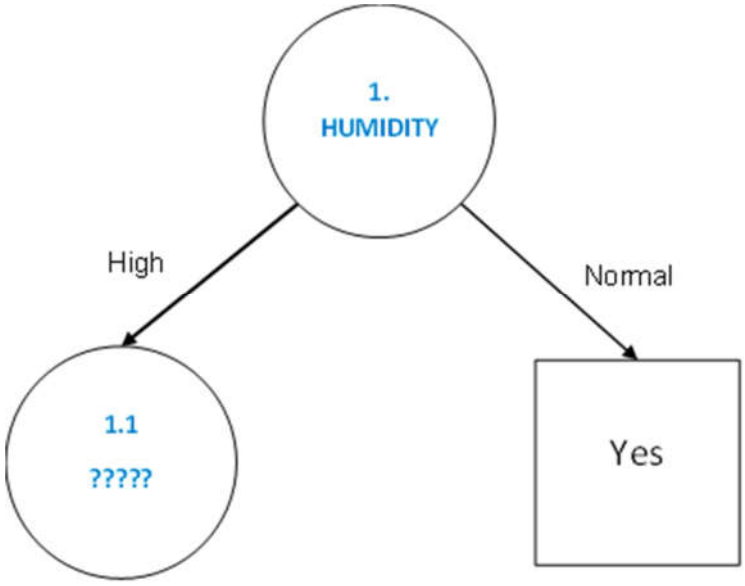
$$Gain(Total, Windy) = Entropy(Total) - \sum_{i=1}^{n} \frac{|Windy_i|}{|Total|} * Entropy(Windy_i)$$

$$Gain(Total, Windy) = 0.863120569 - \left(\left(\frac{8}{14} * 0.811278124\right) + \left(\frac{6}{14} * 0.918295834\right)\right)$$

$$Gain(Total, Windy) = 0.005977711$$

NODE	ATRIBUT		JML KASUS (S)	YA (Si)	TIDAK (Si)	ENTROPY	GAIN
1	TOTAL		14	10	4	0,86312	
	OUTLOOK						0,25852
		CLOUDY	4	4	0	0	
		RAINY	5	4	1	0,72193	
		SUNNY	5	2	3	0,97095	
	TEMPERATURE						0,18385
		COOL	4	0	4	0	
		HOT	4	2	2	1	
		MILD	6	2	4	0,91830	
	HUMADITY						**0,37051**
		HIGH	7	4	3	0,98523	
		NORMAL	7	7	0	0	
	WINDY						0,00598
		FALSE	8	2	6	0,81128	
		TRUE	6	4	2	0,91830	

Dari hasil pada Node 1, dapat diketahui bahwa atribut dengan **Gain tertinggi** adalah **HUMIDITY sebesar 0.37051**. Dengan demikian HUMIDITY dapat menjadi node akar.



Ada 2 nilai atribut dari **HUMIDITY yaitu HIGH dan NORMAL**. Dari kedua nilai atribut tersebut, nilai atribut NORMAL sudah mengklasifikasikan kasus menjadi 1 yaitu keputusan-nya Yes, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut **HIGH masih perlu dilakukan perhitungan lagi.**

3.Buat cabang untuk tiap-tiap nilai

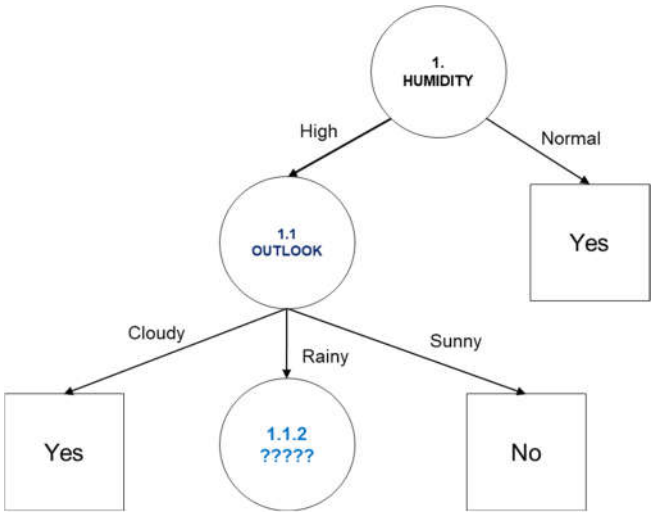
Untuk memudahkan, dataset di filter dengan mengambil data yang memiliki kelembaban **HUMADITY=HIGH** untuk membuat table Node 1.1

OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
Sunny	Hot	**High**	FALSE	No
Sunny	Hot	**High**	TRUE	No
Cloudy	Hot	**High**	FALSE	Yes
Rainy	Mild	**High**	FALSE	Yes
Sunny	Mild	**High**	FALSE	No
Cloudy	Mild	**High**	TRUE	Yes
Rainy	Mild	**High**	TRUE	No

Perhitungan Entropi Dan Gain Cabang

NODE	ATRIBUT		JML KASUS (S)	YA (Si)	TIDAK (Si)	ENTROPY	GAIN
1.1	HUMADITY		7	3	4	0,98523	
	OUTLOOK						0,69951
		CLOUDY	2	2	0	0	
		RAINY	2	1	1	1	
		SUNNY	3	0	3	0	
	TEMPERATURE						0,02024
		COOL	0	0	0	0	
		HOT	3	1	2	0,91830	
		MILD	4	2	2	1	
	WINDY						0,02024
		FALSE	4	2	2	1	
		TRUE	3	1	2	0,91830	

Dari hasil pada Tabel Node 1.1, dapat diketahui bahwa atribut dengan Gain tertinggi adalah OUTLOOK yaitu sebesar 0.69951. Dengan demikian **OUTLOOK dapat menjadi node kedua**.



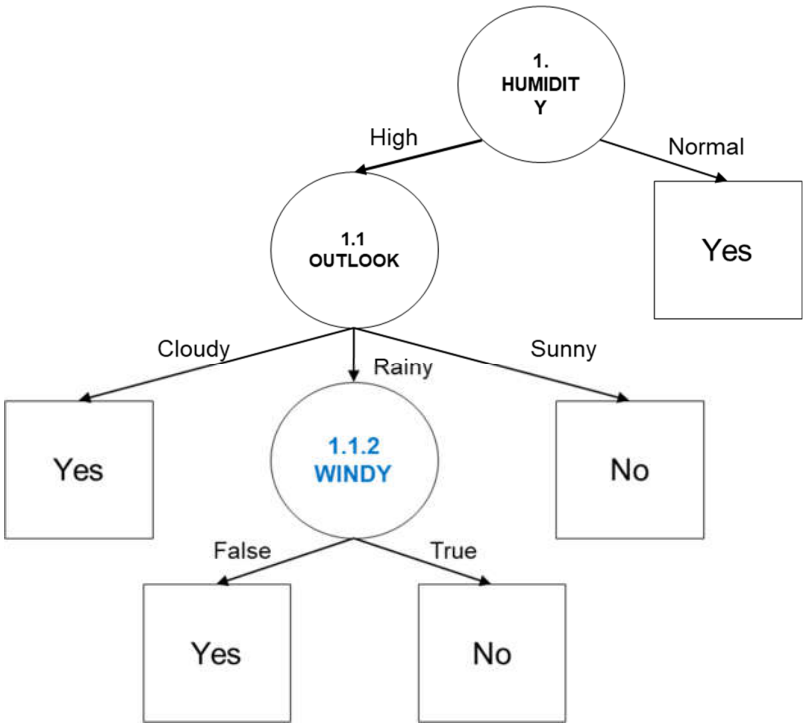
Atribut **CLOUDY = YES dan SUNNY= NO** sudah mengklasifikasikan kasus menjadi 1 keputusan, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut **RAINY masih perlu dilakukan perhitungan lagi**.

4.Ulangi proses untuk setiap cabang sampai semua kasus pada cabang memiliki kelas yang sama

OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
Rainy	Mild	High	FALSE	Yes
Rainy	Mild	High	TRUE	No

NODE	ATRIBUT		JML KASUS (S)	YA (Si)	TIDAK (Si)	ENTROPY	GAIN
1.2	HUMADITY HIGH & OUTLOOK RAINY		2	1	1	1	
	TEMPERATURE						0
		COOL	0	0	0	0	
		HOT	0	0	0	0	
		MILD	2	1	1	1	
	WINDY						1
		FALSE	1	1	0	0	
		TRUE	1	0	1	0	

Dari tabel diatas, **Gain Tertinggi adalah WINDY** dan menjadi node cabang dari atribut RAINY.



Gambar 5.1 Pohon Keputusan Terbentuk

Karena semua kasus sudah masuk dalam kelas. Jadi, pohon keputusan pada Gambar merupakan pohon keputusan terakhir yang terbentuk.

Kemudian ingin menguji satu baru yang ditjukkan pada Tabel 5.1 termasuk Play Golf Yes atau No berdasarkan pohon keputusan pada Gambar 5.1.

Tabel 5.1 Data Testing

Outlook	Temp.	Humidity	Windy	Play Golf
Rainy	Mild	High	False	?

Berdasarkan data testing Tabel 5.1 termasuk Play Golf **Yes**.

5.3 TUGAS

Buatlah pohon keputusan menggunakan metode C4.5 dengan data training pada Tabel 5.2.

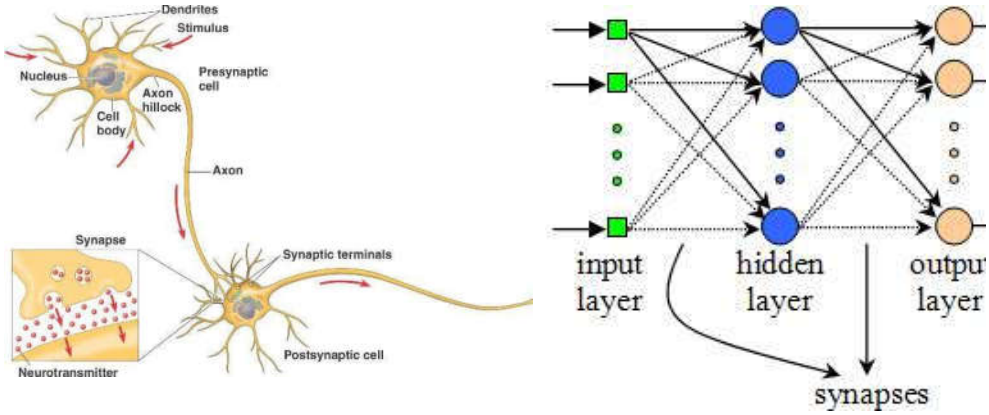
Tabel 5.2 Data Training

age	income	student	credit_rating	buys_computer (Class)
<=30	high	no	fair	no
<=30	high	no	excellent	no
31…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

BAB VI
METODE ARTIFICIAL NEURAL NETWORK

6.1 Konsep Artificial Neural Network

Jaringan saraf (Neural Network) adalah prosesor terdistribusi paralel secara masif yang terdiri dari unit pemrosesan sederhana, yang memiliki sifat alami kecenderungan untuk menyimpan pengetahuan pengalaman dan membuatnya tersedia untuk digunakan. Neural Network mengadopsi otak manusia untuk memproses tugas.



- Untuk melakukan tugas, Neural Network menggunakan interkoneksi yang kuat dari sel komputasi yang dikenal sebagai "neuron"
- Pengetahuan diperoleh oleh jaringan saraf dari lingkungannya melalui proses pembelajaran
- Kekuatan koneksi interneuron, dikenal sebagai bobot sinaptik, digunakan untuk menyimpan pengetahuan yang diperoleh

Kemampuan Neural Network

- **Kemampuan Neural Network**
  - Nonlinier adalah properti yang sangat penting. terutama jika fisik yang mendasarinya mekanisme yang bertanggung jawab untuk menghasilkan sinyal input (mis.• sinyal suara)
- **Pemetaan Input-Output**
  - Neural network belajar dari contoh dengan membuat pemetaan input-output untuk masalah.
- **Adaptivity**

- Jaringan saraf memiliki kemampuan bawaan untuk mengadaptasi bobot sinaptiknya terhadap perubahan dalam lingkungan sekitar

- **Contextual Information**
  - Pengetahuan diwakili oleh struktur dan keadaan aktivasi jaringan saraf

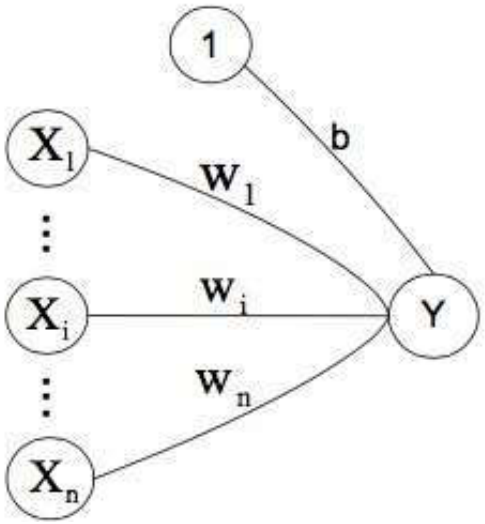
6.2 Arsitektur Neural Network

Secara umum arsitektur jaringan saraf , dapat dikelompokkan sebagai berikut:

1. Single-layer Feedforward Networks
2. Multilayer Feedforward Networks
3. Recurrent Networks

Single-layer Feedforward Networks

X : Input ; W : weight ; Y: output



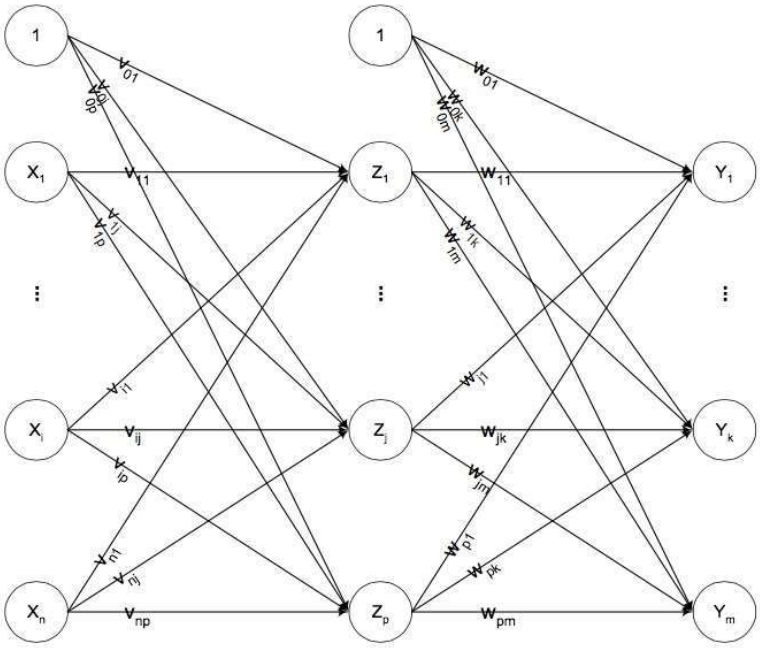
Multilayer Feedforward Networks

X : input ;

V, W : weight ;

Y: output ;

Z : hidden layer



6.3 Learning Process and Activation Function

a. Proses Pembelajaran

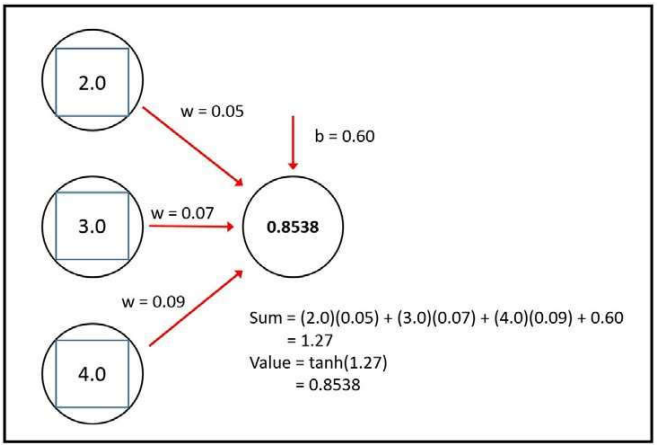
Fungsi Aktivasi

• Fungsi yang menentukan keadaan internal sebuah neuron dalam JST

• Keluarannya akan dikirim ke neuron lain sebagai input

• Identitas, tangga biner, tangga bipolar, sigmoid biner, sigmoid bipolar

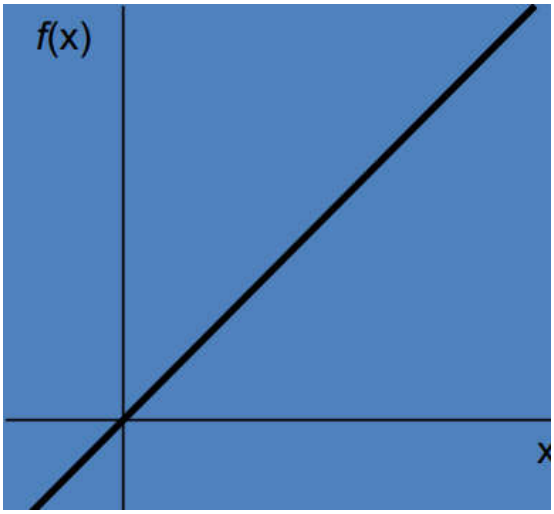
Contoh :



a. Fungsi-fungsi Aktivasi

Fungsi Identitas

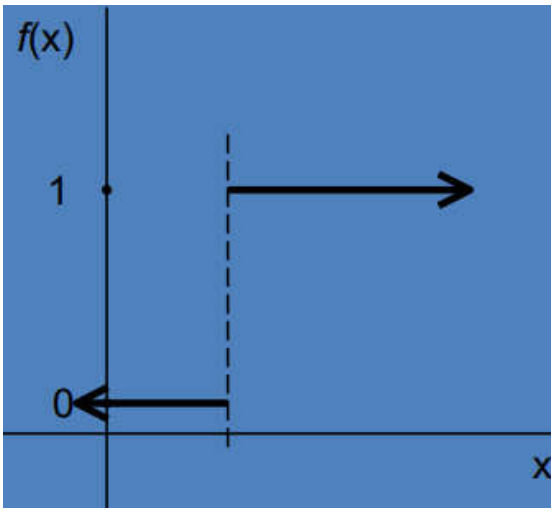
$f(x) = x,$ untuk semua



Fungsi Tangga Biner (Heaviside/threshold)

$$f(x) = \begin{cases} 1, & if \ x \geq \\ 0, & if \ x < \end{cases}$$

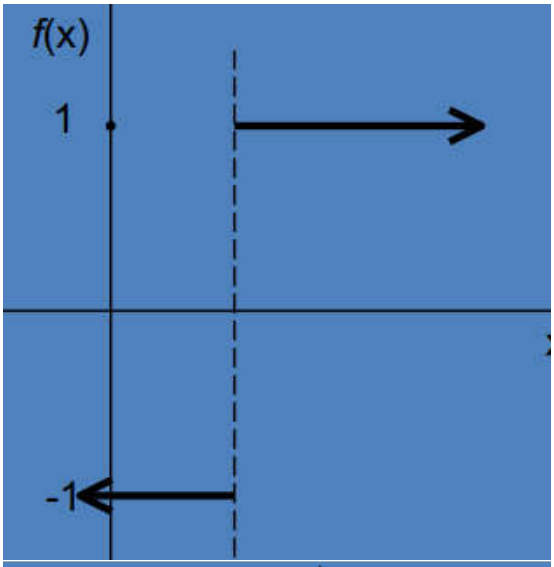
Biasa digunakan pada jaringan lapis tunggal Digunakan untuk mengubah input kontinyu menjadi output biner



Fungsi Tangga Bipolar

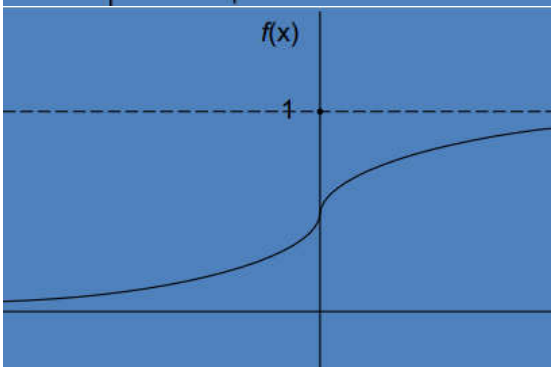
$$f(x)=\begin{cases}1, & if\ x\geq\theta\\ -1, & if\ x<\theta\end{cases}$$

Serupa dengan tangga biner Perbedaan pada range {-1,1}



Fungsi Sigmoid Biner

$$f(x)=\frac{1}{1-\exp(-\sigma x)}$$

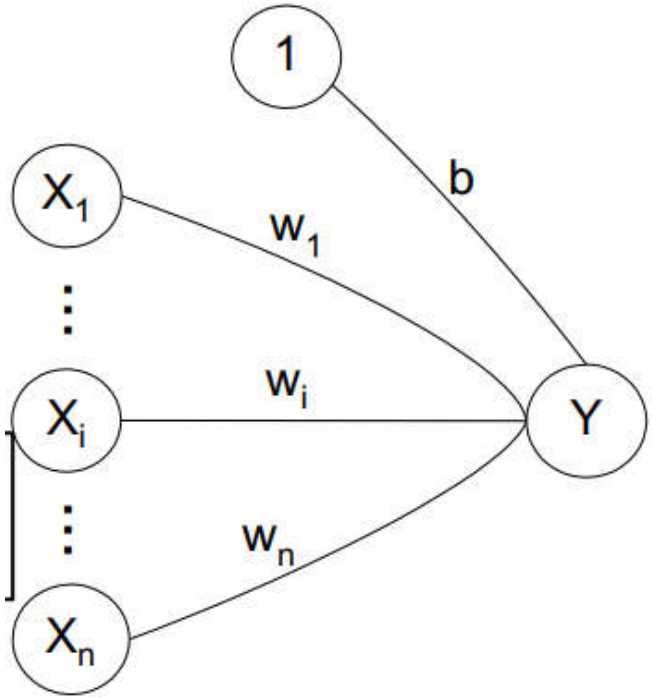


6.4 Algoritma Pembelajaran Perceptron

- Ditemukan oleh Rosenbalt (1962) dan Minsky– Papert (1969)
- Jaringan terdiri dari satu atau lebih unit masukan dan satu unit keluaran
- Mempunyai sebuah bias yang bernilai +1 dan mempunyai bobot b
- Fungsi aktivasi = fungsi tangga bipolar dengan nilai tetap

$$f(y_{in})=\begin{cases}1 & jika\ y_{in}>\theta\\ 0 & jika\ -\theta\leq y_{in}\leq\theta\\ -1 & jika\ y_{in}<-\theta\end{cases}$$

- n input 1 output 1 nilai bias



Algoritma Pembelajaran Perceptron

1. Inisialisasi bobot wi = 0 untuk i = 1, 2, 3, ….., n Set aktivasi untuk unit masukan xi = si i = 1, 2, …, n
2. Hitung total masukan ke unit keluaran

$$y = b + \sum_i x_i w_i$$

3. Masukkan ke fungsi aktivasi
4. Jika y ≠ t update bobot

   wi (new) = wi (old) + Δwi i = b, 1, 2, …, n

$$\Delta w_i = \alpha \cdot t \cdot x_i \qquad 0 \le \alpha \le 1$$

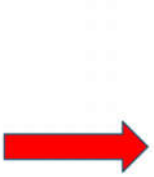
   Lakukan langkah 2-4

Studi Kasus Perhitungan Percepton

Melakukan pembelajaran perceptron untuk kasus logika AND dengan learning rate (α) = 1, Thresold (θ) = 0. Adapun bobot yang digunakan masing-masing $w_1 = 1,\ w_2 = 1,\ w_b = 1$.

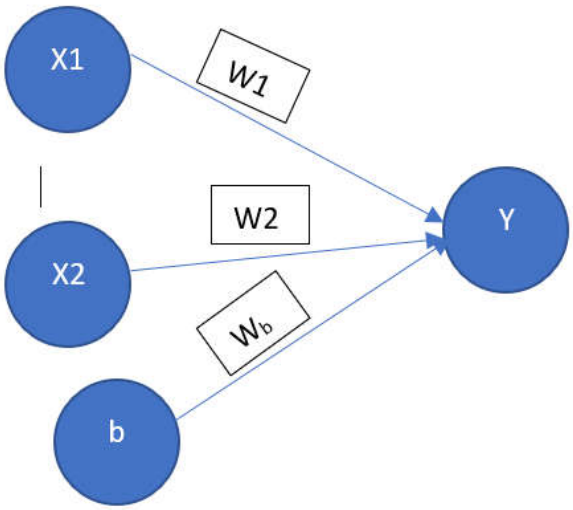
Gunakan **2 iterasi** saja.

s_1	s_2	t
1	1	1
1	0	0
0	1	0
0	0	0



x_1	x_2	t
1	1	1
1	-1	-1
-1	1	-1
-1	-1	-1

1. Gambar Arsitekturnya



2. Pembelajaran algoritma perceptron

Iterasi 1

Untuk data 1, $x_1 = 1,\ x_2 = 1,\ t = 1$

$$y = w_b + x_1 * w_1 + x_2 * w_2$$
$$y = 1 + 1*1 + 1*1 = 3$$

Karena y = t, sehingga bobotnya tidak berubah.

Untuk data 2, $x_1 = 1, x_2 = -1, t = -1$

$$y = w_b + x_1 * w_1 + x_2 * w_2$$
$$y = 1 + (1) * 1 + (-1) * 1 = 1$$

Karena y ≠ t, sehingga bobot di update.

$$w_i(new) = w_i(old) \; + \; a \; . \; t \; . \; x_i$$

$$w_1(new) = 1 + 1 * (-1) * 1 = 0$$
$$w_2(new) = 1 + 1 * (-1) * (-1) = 2$$
$$w_b(new) = 1 + 1 * (-1) = 0$$

Sehingga bobot baru yang digunakan :

$$w_1(new) = 0, \; w_2(new) = 2, \; w_b(new) = 0$$

Untuk data 3, $x_1 = -1, x_2 = 1, t = -1$

$$y = 0 + (-1) * 0 + 1 * 2 = 2$$

Karena y ≠ t, sehingga bobot di update.

$$w_i(new) = w_i(old) \; + \; a \; . \; t \; . \; x_i$$

$$w_1(new) = 0 + 1 * (-1) * (-1) = 1$$
$$w_2(new) = 2 + 1 * (-1) * (1) = 1$$
$$w_b(new) = 0 + 1 * (-1)) = -1$$

Sehingga bobot baru yang digunakan :

$$w_1(new) = 1, w_2(new) = 1, w_b(new) = -1,$$

Untuk data 4, $x_1 = -1, x_2 = -1, t = -1$

$$y = w_b + x_1 * w_1 + x_2 * w_1$$
$$y = (-1) + (-1) * 1 + (-1) * 1 = -3$$

Karena y = t, sehingga bobot tidak diupdate.

$$w_1(new) = 1, w_2(new) = 1, w_b(new) = -1,$$

Iterasi 2.

Untuk data 1, $x_1 = 1,\ x_2 = 1,\ t = 1$

$$w_1(new) = 1, w_2(new) = 1, w_b(new) = -1,$$

$$y = (-1) + 1 * 1 + 1 * 1 = 0$$

Karena y = t, sehingga bobotnya tidak berubah.

Untuk data 2, $x_1 = 1, x_2 = -1, t = -1$

$$w_1(new) = 1, w_2(new) = 1, w_b(new) = -1,$$

$$y = (-1) + 1 * 1 + (-1) * 1 = -2$$

Karena y = t, sehingga bobotnya tidak berubah.

Untuk data 3, $x_1 = -1, x_2 = 1, t = -1$

$$w_1(new) = 1, w_2(new) = 1, w_b(new) = -1,$$

$$y = (-1) + (-1) * 1 + 1 * 1 = -1$$

Karena y = t, sehingga bobotnya tidak berubah.

Untuk data 4, $x_1 = -1, x_2 = -1, t = -1$

$$w_1(new) = 1, w_2(new) = 1, w_b(new) = -1,$$

$$y = (-1) + (-1) * 1 + (-1) * 1 = -3$$

Karena y = t, sehingga bobotnya tidak berubah.

Iterasi dihentikan karena sudah sama dengan iterasi maksimum yang diterntukan yaitu 2 iterasi, sehingga bobot akhir yang didapatkan adalah

$$w_1(new) = 1, w_2(new) = 1, w_b(new) = -1, .$$

Pengujian metode perceptron.

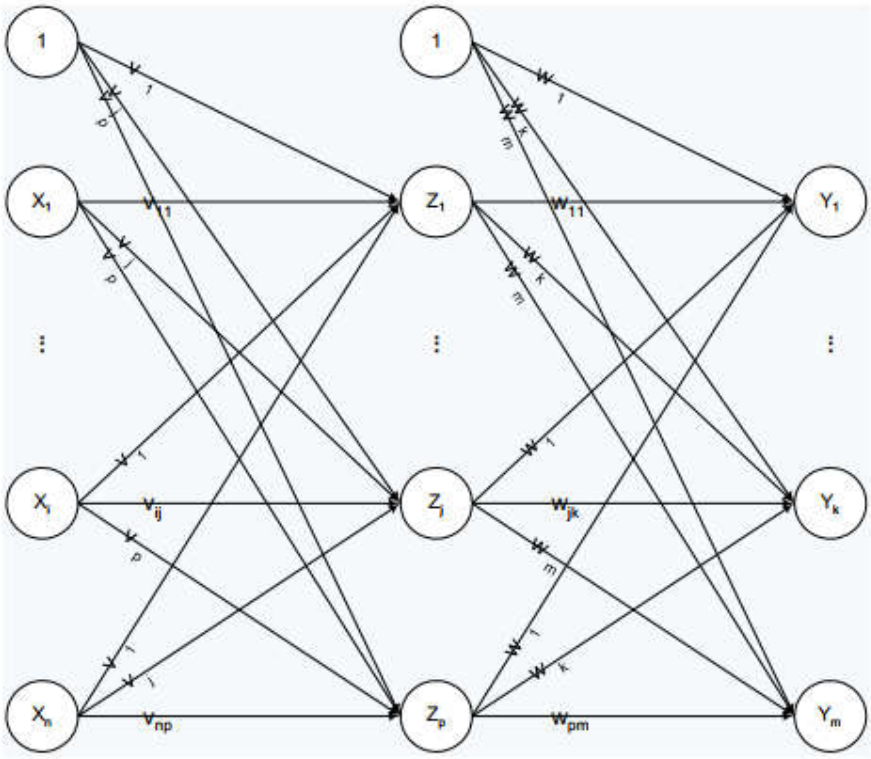
x1	x2	$net = \sum(x_i * w_i) + w_b$	$Y = f(net) = 1$, jika $net \geq 0$ $Y = f(net) = -1$, jika $net < 0$
1	1	1	1
1	-1	-1	-1
-1	1	-1	-1
-1	-1	-3	-1

Pola dapat dikenali.

6.5 Algoritma Pembelajaran Back-propagation

- Back-propagation diperkenalkan oleh Rumelhart, Hinton dan Williams dan dipopulerkan pada buku Parallel Distributed Processing (Rumelhart and McLelland, 1986).
- Prinsip dasar algoritma propagasi-balik memiliki tiga fase:
    - Fase feedforward pola input pembelajaran
    - Fase kalkulasi dan back-propagation error yang didapat.
    - Fase penyesuaian bobot.
- Jaringan lapis banyak
- Terdiri dari satu lapisan unit-unit masukan, satu atau lebih lapisan tersembunyi dan satu lapisan unit keluaran
- Struktur dasar sama sepeti perceptron, sehingga disebut multilayer perceptron
- Setiap neuron pada suatu lapisan dalam jaringan Propagasi-Balik menapat sinyal
- masukan dari semua neuron pada lapisan sebelumnya beserta satu sinyal bias.

- **Back-propagation (Arsitektur)**
    - input layer ditunjukkan oleh unit-unit Xi
    - output layer ditunjukkan oleh unit-unit Yk
    - hidden layer ditunjukkan oleh unit-unit Zj



Algoritma Pembelajaran Back-propagation

- Pembelajaran back-propagation
    - Inisialisasi bobot

      wi = 0 atau angka acak untuk i = b, 1, 2, 3, ….., n Set laju pembelajaran α (0,1 ≤ nα ≤ 1)
- Selama syarat henti belum tercapai:

  a. Feedforward:

  2.Setiap unit masukan (Xi, i = 1, …, n) menerima sinyal masukan xi dan meneruskannya ke seluruh unit pada lapisan di atasnya (hidden units)

3.Setiap unit tersembunyi (Zj, j = 1, …, p) menghitung total sinyal masukan terbobot dan fungsi aktivasinya.

$$z\_in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij} \quad \Rightarrow \quad z_j = f(z\_in_j)$$

4. Setiap unit output (Yk, k = 1, …, m) menghitung total sinyal masukan terbobot dan fungsi aktivasinya

b. Backpropagation of error:

5. Setiap unit output (Yk, k = 1, …, m) menerima sebuah pola target yang sesuai dengan pola masukan pelatihannya. Unit tersebut menghitung informasi kesalahan dan mengoreksi bobot.

$$\delta_k = (t_k - y_k) f'(y\_in_k) \quad \Rightarrow \quad \Delta w_{jk} = \alpha \delta_k z_j$$

$$\Rightarrow \quad \Delta w_{0k} = \alpha \delta_k$$

6. Setiap unit tersembunyi (Zj, j = 1, …, p) menghitung selisih input (dari unit-unit pada layer di atasnya),

$$\delta\_in_j = \sum_{k=1}^{m} \delta_k w_{jk} \quad \Rightarrow \quad \delta_j = \delta\_in_j f'(z\_in_j)$$

$$\Rightarrow \quad \Delta v_{ij} = \alpha \delta_j x_i$$

$$\Rightarrow \quad \Delta v_{0j} = \alpha \delta_j$$

1. Setiap unit output (Yk, k = 1, …, m) dan unit tersembunyi (Zj, j = 1, …, p) mengubah bias dan bobot-bobotnya (j = 0, …, p) dan (i = 1, …, n):

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk} \qquad v_{ij}(new) = v w_{ij}(old) + \Delta v_{ij}$$

2. Uji syarat henti

$$\sum_{k=1}^{n} (t_k - y_k)^2$$

Pemilihan bobot awal dan bias pada back-propagation

1. Pemilihan bobot awal mempengaruhi apakah jaringan akan mencapai error
2. minimum global (atau lokal), dan jika tercapai, seberapa cepat konvergensinya
3. Update bobot tergantung pada fungsi aktivasi unit yang lebih dalam (pemberi sinyal input) dan turunan fungsi aktivasi unit yang lebih luar penerima sinyal input), sehingga perlu dihindari pemilihan bobot awal yang menyebabkan keduanya bernilai 0
4. Jika menggunakan fungsi sigmoid, nilai bobot awal tidak boleh terlalu besar karena dapat menyebabkan nilai turunannya menjadi sangat kecil (jatuh di daerah saturasi). Sebaliknya juga tidak boleh terlalu kecil, karena dapat menyebabkan net input ke unit tersembunyi atau unit output menjadi terlalu dekat dengan nol, yang membuat pembelajaran terlalu lambat.
5. Bobot dan bias diinisialisasi nilai acak antara -0.5 dan 0.5 (atau antara -1 dan 1, atau pada interval lain yang sesuai).

Contoh Backbropagation

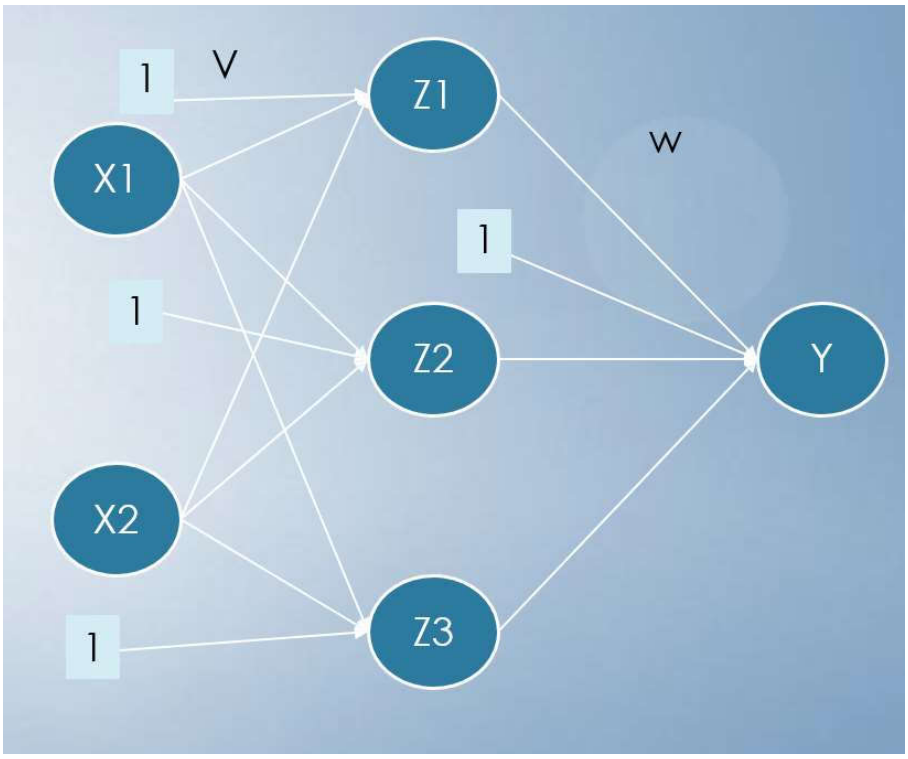
Back-propagation dengan sebuah layer tersembunyi (dengan 3 unit) untuk mengenali fungsi

logika XOR dengan laju pembelajaran $\alpha = 0{,}2$.

s_1	s_2	t
1	1	0
1	0	1
0	1	1
0	0	0

Penyelesaian:

Gambar Arsitekturnya



Inisialisasi bobot

	Z1	Z_2	Z_3
X_1	0,2	0,3	-0,1
X_2	0,3	0,1	-0,1
1	-0,3	0,3	0,3

	Y
Z_1	0,5
Z_2	-0,3
Z_3	-0,4
1	-0,1

Keluaran unit tersembunyi (Z_j)

z_net 1 = -0,3 + 1 (0,2) + 1 (0,3) = 0,2
z_net 2 = 0,3 + 1 (0,3) + 1 (0,1) = 0,7
z_net 3 = 0,3 + 1 (-0,1) + 1 (-0,1) = 0,1

Fungsi aktivasi (Z_j)

z_net 1 = 0,55
z_net 2 = 0,67
z_net 3 = 0,52

Keluaran unit (Y_{kj})

$y\_net\ k = -0,1 + 0,55\ (0,5) + 0,67\ (-0,3) + 0,52\ (-0,4) = -0,24$

$y = f(y_{\_net\ k}) = 0,44$

Faktor δ di unit keluaran Y_k

$\delta_k = (t_k - y_k)\ f'(y_{\_net\ k}) = (t - y)\ y\ (1 - y) = (0 - 0,44)\ (0,44)\ (1 - 0,44) = -0,11$

Perubahan bobot w_{jk} ($\Delta w_{jk} = \alpha\ \delta_k\ z_j$), dengan $\alpha = 0,2$

$\Delta w_{01} = 0,2\ (-0,11)\ (1) = -0,02$

$\Delta w_{11} = 0,2\ (-0,11)\ (0,55) = -0,01$ $\Delta w_{21} = 0,2\ (-0,11)\ (0,67) = -0,01$

$\Delta w_{21} = 0,2\ (-0,11)\ (0,67) = -0,01$

$\Delta w_{31} = 0,2\ (-0,11)\ (0,52) = -0,01$

Penjumlahan kesalahan dari unit tersembunyi (δ)

$\delta_{\_net\ j} = \sum \delta_k\ w_{jk}$

$\delta_{\_net\ j} = \delta\ w_{j1}\ \ (j = 1,2,3)$

$\delta_{\_net\ 1} = (-0,11)\ (0,5) = -0,05$

$\delta_{\_net\ 2} = (-0,11)\ (-0,3) = 0,03$

$\delta_{\_net\ 3} = (-0,11)\ (-0,4) = 0,04$

Faktor kesalahan δ di unit tersembunyi

$\delta_j = \delta_{\_net\ j}\ f'(z_{\_net\ j}) = \delta_{\_net\ j}\ z_j\ (1 - z_j)$

$\delta_1 = -0,05\ (0,55)\ (1 - 0,55) = -0,01$

$\delta_2 = 0,03\ (0,67)\ (1 - 0,67) = 0,01$

$\delta_3 = 0,04\ (0,52)\ (1 - 0,52) = 0,01$

Perubahan bobot ke unit tersembunyi $\Delta v_{ij} = \alpha\ \delta_j\ x_i$

		Z_1	Z_2	Z_3
X_1		$\Delta v_{11} = (0,2)\ (-0,01)$	$\Delta v_{12} = (0,2)\ (0,01)$	$\Delta v_{13} = (0,2)\ (0,01)$
		$(1) = -0,002 \approx 0$	$(1) = 0,002 \approx 0$	$(1) = 0,002 \approx 0$
X_2		$\Delta v_{21} = (0,2)\ (-0,01)$	$\Delta v_{22} = (0,2)\ (0,01)$	$\Delta v_{23} = (0,2)\ (0,01)$
		$(1) = -0,002 \approx 0$	$(1) = 0,002 \approx 0$	$(1) = 0,002 \approx 0$
1		$\Delta v_{01} = (0,2)\ (-0,01)$	$\Delta v_{02} = (0,2)\ (0,01)$	$\Delta v_{03} = (0,2)\ (0,01)$
		$(1) = -0,002 \approx 0$	$(1) = 0,002 \approx 0$	$(1) = 0,002 \approx 0$

Perubahan bobot unit keluaran (w_{jk} (baru) = w_{jk}(lama) + $\triangle w_{jk}$)

w_{01}(baru) = -0,1 – 0,02 = -0,12

w_{11}(baru) = 0,5 – 0,01 = 0,49

w_{21}(baru) = -0,3 – 0,01 = -0,31

w_{31}(baru) = -0,4 – 0,01 = -0,41

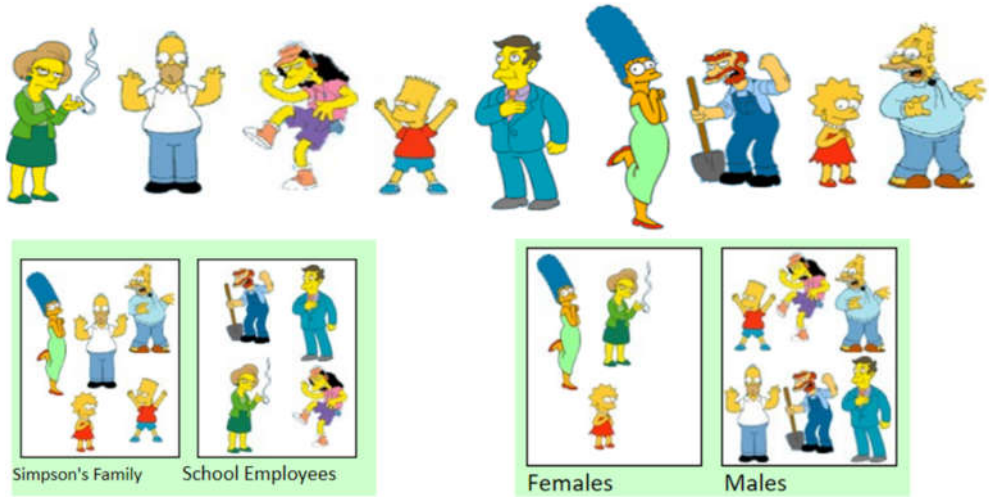
Perubahan bobot unit tersembunyi

	Z_1	Z_2	Z_3
X_1	v_{11} = 0,2 + 0 = 0,2	v_{12} = 0,3 + 0 = 0,3	v_{13} = -0,1 + 0 = -0,1
X_2	v_{21} = 0,3 + 0 = 0,3	v_{22} = 0,1 + 0 = 0,1	v_{23} = -0,1 + 0 = -0,1
1	v_{01} = -0,3 + 0 = -0,3	v_{02} = 0,3 + 0 = 0,3	v_{03} = 0,3 + 0 = 0,3

BAB VII
CLUSTERING METHOD

7.1 Konsep *Clustering*

Clustrering disebut juga sebagai *Unsupervised Learning*. *Clustering* merupakan mengelompokkan menjadi beberapa Cluster (kelompok) berdasarkan kesamaannya. Ilustrasi contoh pengelompokan seperti pada Gambar 7.1.



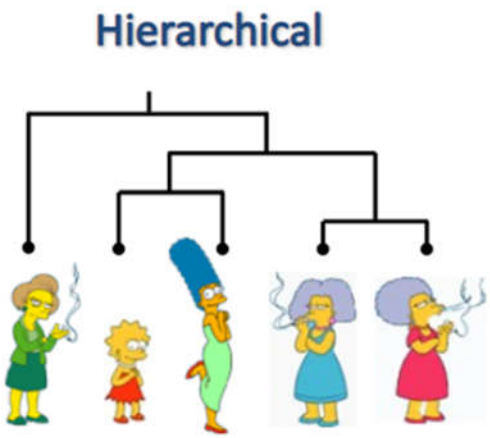
Gambar 7.1 Ilustrasi Pengelompokan

Jenis pengelompokan terbagi menjadi dua bagian yaitu :

Partitional algorithms yaitu membuat beberapa partisi dan mengelompokkan objek berdasarkan kriteria tertentu (Gambar 7.2). Salah satu contoh metode partitional adalah k-means.

Hierarchical algorithm yaitu Membuatk dekomposisi pengelompokan objek berdasarkan kriteria tertentu. Misal = tua-muda, merokok-tidak merokok) (Gambar 7.3).

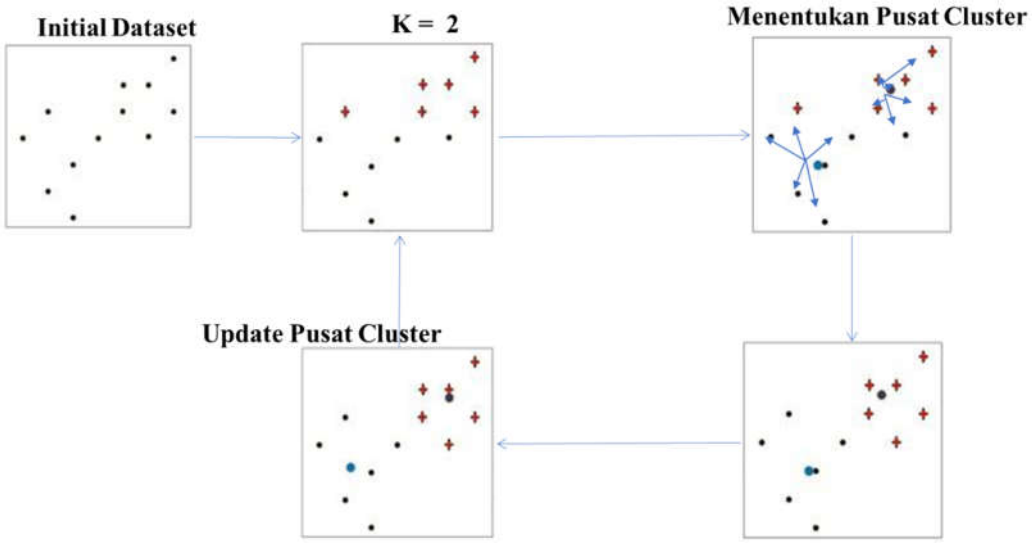
Gambar 7.2 Konsep Partitional Clustering



Gambar 7.3 Konsep Hierarchical Clustering

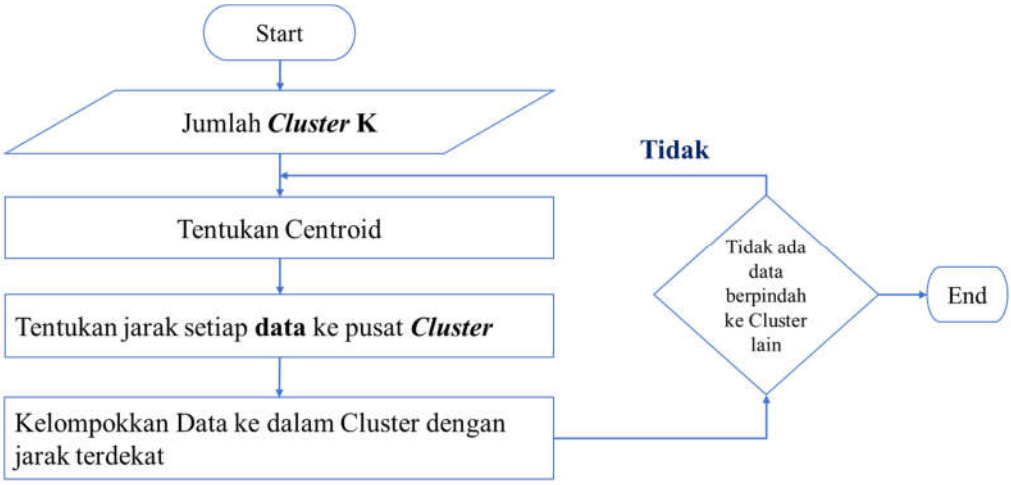
7.2 Metode K-Means

K-Means merupakan salah satu metode Clustering partitional yang digunakan untuk mempartisi N data ke dalam beberapa K kelompok Parameter K menunjukkan banyaknya cluster yang akan dibentuk. Setiap kelompok data memiliki jarak terdekat dengan centroidnya masing-masing. Cara Kerja metode k-means ditunjukkan pada Gambar 7.4.



Gambar 7.4 Cara Kerja Metode K-Means

Sedangkan flowchart metode k-means ditunjukkan pada Gambar 7.5.



Gambar 7.5 Flowchart Metode K-Means

Untuk melakukan perhitungan jarak data ke-i (***X_i***) pada pusat cluster ke-k (**C_k**), diberi nama (**d_ik**), dapat digunakan formula seperti berikut:

$$d_{ik} = \sqrt{\sum_{j=1}^{m} (x_{ij} - c_{kj})^2}$$

7.2 Studi Kasus

Diketahui angka kematian kasar (**CDR**) dan angka kelahiran kasar (**CBR**) 10 negara seperti terlihat pada Tabel 1. Negara-negara tersebut akan dikelompokkan berdasarkan **CBR dan CDRnya** menjadi tiga kelompok. Proses pengelompokkan menggunakan metode k-means.

Tabel 7.1 CDR dan CBR tahun 1994

(sumber: ESCAP Population Data Sheet 1996)

No	Negara	CBR	CDR
1	Brunei Darusalam	27	3
2	Kamboja	38	14
3	Indonesia	24	8
4	Laos	43	15
5	Malaysia	28	5
6	Myanmar	32	11

7	Filipina	30	7
8	Singapura	17	5
9	Thailand	20	6
10	Vietnam	29	8

Penyelesaian

Iterasi 1

Step 1 : Tentukan jumlah cluster (K), serta tetapkan pusat cluster sembarang

Misalkan kita akan pengelompokkan data tersebut menjadi 3 cluster, **K = 3**. Misalkan pusat cluster kita tetapkan sembarang seperti pada Tabel 7.2.

Tabel 7.2. Penentuan Pusat Cluster Secara Random

No	Cluster	CBR	CDR
1	Brunei Darusalam **(C1)**	27	3
2	Indonesia **(C2)**	24	8
3	Singapura **(C3)**	17	5

Step 2 : Hitung jarak setiap data terhadap setiap pusat cluster. Misalkan untuk menghitung

Jarak data pertama (Brunei Darusalam) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{11}} = \sqrt{(27 - \mathbf{27})^2 + (3 - \mathbf{3})^2} = \mathbf{0}$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{12}} = \sqrt{(27 - \mathbf{24})^2 + (3 - \mathbf{8})^2} = \mathbf{5,8310}$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{13}} = \sqrt{(27 - \mathbf{17})^2 + (3 - \mathbf{5})^2} = \mathbf{10,1980}$$

Jarak data kedua (Kamboja) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{21}} = \sqrt{(38 - \mathbf{27})^2 + (14 - \mathbf{3})^2} = 15{,}5563$$

Jarak data kedua (Kamboja) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{22}} = \sqrt{(38 - \mathbf{24})^2 + (14 - \mathbf{8})^2} = 15{,}2315$$

Jarak data kedua (kamboja) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{23}} = \sqrt{(38 - \mathbf{17})^2 + (14 - \mathbf{5})^2} = 22{,}8473$$

Perhitungan data 3 – 10 sama dengan sebelumnya.

Step 3 : Suatu data akan menjadi anggota dari suatu cluster yang memiliki jarak terkecil dari pusat clusternya seperti pada Tabel 7.3.

Tabel 7.3 Hasil Pengelompokan pada Iterasi 1

No	Negara	Negara		Jarak			Anggota Cluster		
		CBR	CDR	C1	C2	C3	C1	C2	C3
1	Brunei Darusalam	27	3	**0**	5.8310	10.1980	*		
2	Kamboja	38	14	15.5563	**15.2315**	22.8473		*	
3	Indonesia	24	8	5.8310	**0**	7.6158		*	
4	Laos	43	15	**20**	20.2485	27.8568	*		
5	Malaysia	28	5	**2.2361**	5.0000	11.0000	*		
6	Myanmar	32	11	9.4340	**8.5440**	16.1555		*	
7	Filipina	30	7	**5**	6.0828	13.1529	*		
8	Singapura	17	5	10.1980	7.6158	**0**			*
9	Thailand	20	6	7.6158	4.4721	**3.1623**			*
10	Vietnam	29	8	5.3852	**5**	12.3693		*	

Step 4 : Hitung Cluster Baru

Hitung pusat cluster baru. Untuk cluster pertama (C1), ada 4 data yaitu data ke-1, 4, 5 dan data ke-7, sehingga:

$$\boldsymbol{C_{11}} = \frac{27 + 43 + 28 + 30}{4} = 32$$

$$C_{12} = \frac{3 + 15 + 5 + 7}{4} = 7{,}5$$

Sehingga ; **C1 = (32 , 7.5)**

Untuk cluster kedua, ada 4 data yaitu data ke-2, 3 , 6, data ke-10, sehingga:

$$C_{21} = \frac{38 + 24 + 32 + 29}{4} = 30{,}75$$

$$C_{22} = \frac{14 + 8 + 11 + 8}{4} = 10{,}25$$

Sehingga ; **C2 = (30.75 , 10.25)**

Untuk cluster ketiga, ada 2 data yaitu data ke-8 dan data ke-9, sehingga:

$$C_{31} = \frac{17 + 20}{2} = 18{,}5$$

$$C_{32} = \frac{5 + 6}{2} = 5{,}5$$

Sehingga ; **C3 = (18.5 , 5,5)**

Iterasi 2

Step 1 : Tentukan jumlah cluster (K), serta tetapkan pusat cluster sembarang

Ada 3 Pusat cluster baru didapatkan pada iterasi 1 yang ditunjukkan pada Tabel 7.4.

Tabel 7.4 Hasil Pusat Cluster pada Iterasi 1

No	Cluster	CBR	CDR
1	**(C1)**	**32**	**7.5**
2	**(C2)**	**30.75**	**10.25**
3	**(C3)**	**18.5**	**5.5**

Step 2 : Hitung jarak setiap data terhadap setiap pusat cluster.

Jarak data pertama (Brunei Darusalam) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{11}} = \sqrt{(27 - \mathbf{32})^2 + (3 - \mathbf{7.5})^2} = 6{,}7268$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{12}} = \sqrt{(27 - \mathbf{30.75})^2 + (3 - \mathbf{10.25})^2} = 8{,}1624$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{13}} = \sqrt{(27 - \mathbf{18.5})^2 + (3 - \mathbf{5.5})^2} = 8{,}8600$$

Jarak data kedua (Kamboja) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{21}} = \sqrt{(38 - \mathbf{32})^2 + (14 - \mathbf{7.5})^2} = 8{,}8459$$

Jarak data kedua (Kamboja) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{22}} = \sqrt{(38 - \mathbf{30.75})^2 + (14 - \mathbf{10.25})^2} = 8{,}1624$$

Jarak data kedua (kamboja) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{23}} = \sqrt{(38 - \mathbf{18.5})^2 + (14 - \mathbf{5.5})^2} = 21{,}2720$$

Perhitungan data 3 – 10 sama dengan sebelumnya.

Step 3 : Suatu data akan menjadi anggota dari suatu cluster yang memiliki jarak terkecil dari pusat clusternya seperti pada Tabel 7.6.

Gambar 7.6 Hasil Pengelompokan pada Iterasi 2

No	Negara	Negara		Jarak			Anggota Cluster Lama			Anggota Cluster Baru		
		CBR	CDR	C1	C2	C3	C1	C2	C3	C1	C2	C3
1	Brunei Darusalam	27	3	**6.7268**	8.1624	8.8600	*			*		
2	Kamboja	38	14	8.8459	**8.1624**	21.2720		*			*	
3	Indonesia	24	8	8.0156	7.1151	**6.0415**		*				**
4	Laos	43	15	13.3135	**13.1387**	26.2774	*				**	
5	Malaysia	28	5	**4.7170**	5.9266	9.5131	*			*		
6	Myanmar	32	11	3.5000	**1.4577**	14.5774		*			*	
7	Filipina	30	7	**2.0616**	3.3354	11.5974	*			*		
8	Singapura	17	5	15.2069	14.7182	**1.5811**			*			*
9	Thailand	20	6	12.0934	11.5596	**1.5811**			*			*
10	Vietnam	29	8	3.0414	**2.8504**	10.7935		*			*	

Terlihat masih ada 2 data yang berubah posisi dari kondisi semula, yaitu data ke-3 dan ke-4. Sehingga perlu dihitung pusat cluster baru.

Step 4 : Hitung Cluster Baru

Untuk cluster pertama, ada 3 data yaitu data ke-1, 5 dan data ke-7, sehingga:

$$C_{11} = \frac{27 + 28 + 30}{3} = \mathbf{28,3}$$

$$C_{12} = \frac{3 + 5 + 7}{3} = \mathbf{5}$$

Sehingga ; **C1 = (28.5 , 5)**

Untuk cluster kedua, ada 4 data yaitu data ke-2, 4 , 6, data ke-10 sehingga:

$$C_{21} = \frac{38 + 43 + 32 + 29}{4} = \mathbf{35,5}$$

$$C_{22} = \frac{14 + 15 + 11 + 8}{4} = \mathbf{12}$$

Sehingga ; **C2 = (35.5 , 12)**

Untuk cluster ketiga, ada 3 data yaitu data ke-3, 8 dan data ke-9, sehingga:

$$C_{31} = \frac{24 + 17 + 20}{3} = \mathbf{20,3}$$

$$C_{32} = \frac{8 + 5 + 6}{3} = \mathbf{6,3}$$

Sehingga ; **C3 = (20.3 , 6.3)**

Iterasi 3

Step 1 : Tentukan jumlah cluster (K), serta tetapkan pusat cluster sembarang

Ada 3 Pusat cluster baru didapatkan pada iterasi 2 yang ditunjukkan pada Tabel 7.5.

Tabel 7.5 Hasil Pusat Cluster pada Iterasi 2

No	Cluster	CBR	CDR
1	**(C1)**	**28.3**	**5**
2	**(C2)**	**35.5**	**12**
3	**(C3)**	**20.3**	**6.3**

Step 2 : Hitung jarak setiap data terhadap setiap pusat cluster.

Jarak data pertama (Brunei Darusalam) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{11} = \sqrt{(27 - 28.3)^2 + (3 - 5)^2 = 2,4037}}$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{12} = \sqrt{(27 - 35.5)^2 + (3 - 12)^2 = 12,3794}}$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{13} = \sqrt{(27 - 20.3)^2 + (3 - 6.3)^2 = 7,4536}}$$

Jarak data kedua (Kamboja) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{21} = \sqrt{(38 - 28.3)^2 + (14 - 5)^2 = 13,2077}}$$

Jarak data kedua (Kamboja) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{22} = \sqrt{(38 - 35.5)^2 + (14 - 12)^2 = 3,2016}}$$

Jarak data kedua (kamboja) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{23} = \sqrt{(38 - 20.3)^2 + (14 - 6.3)^2 = 19,2585}}$$

Perhitungan data 3 – 10 sama dengan sebelumnya.

Step 3 : Suatu data akan menjadi anggota dari suatu cluster yang memiliki jarak terkecil dari pusat clusternya seperti pada Tabel 7.7.

Gambar 7.7 Hasil Pengelompokan pada Iterasi 3

No	Negara	Negara		Jarak			Anggota Cluster Lama			Anggota Cluster Baru		
		CBR	CDR	C1	C2	C3	C1	C2	C3	C1	C2	C3
1	Brunei Darusalam	27	3	**2.4037**	12.3794	7.4536	*			*		
2	Kamboja	38	14	13.2077	**3.2016**	19.2585		*			*	
3	Indonesia	24	8	5.2705	12.1758	**4.0277**			*			*
4	Laos	43	15	17.7514	**8.0777**	24.2670		*			*	
5	Malaysia	28	5	**0.3333**	10.2591	7.7817	*			*		
6	Myanmar	32	11	7.0317	**3.6401**	12.5654		*			*	
7	Filipina	30	7	**2.6034**	7.4330	9.6896	*			*		
8	Singapura	17	5	11.3333	19.7800	**3.5901**			*			*
9	Thailand	20	6	8.3931	16.6208	**0.4714**			*			*
10	Vietnam	29	8	**3.0732**	7.6322	8.8255		*		**		

Terlihat masih ada 1 data yang berubah posisi dari kondisi semula, yaitu data ke-10. Sehingga perlu dihitung pusat cluster baru.

Step 4 : Hitung Cluster Baru

Untuk cluster pertama, ada 4 data yaitu data ke-1, 5, 7 dan data ke-10, sehingga:

$$C_{11} = \frac{27 + 28 + 30 + 29}{4} = \mathbf{28,5}$$

$$C_{12} = \frac{3 + 5 + 7 + 8}{4} = \mathbf{5,75}$$

Sehingga ; **C1 = (28.5 , 5.75)**

Untuk cluster kedua, ada 3 data yaitu data ke-2, 4 , dan data ke-6 sehingga:

$$C_{21} = \frac{38 + 43 + 32}{3} = \mathbf{37,67}$$

$$C_{22} = \frac{14 + 15 + 11}{3} = \mathbf{13.3}$$

Sehingga ; **C2 = (37.67 , 13.3)**

Untuk cluster ketiga, ada 3 data yaitu data ke-3, 8 dan data ke-9, sehingga:

$$C_{31} = \frac{24 + 17 + 20}{3} = \mathbf{20,3}$$

$$C_{32} = \frac{8 + 5 + 6}{3} = \mathbf{6,3}$$

Sehingga ; **C3 = (20.3 , 6.3)**

Iterasi 4

Step 1 : Tentukan jumlah cluster (K), serta tetapkan pusat cluster sembarang

Ada 3 Pusat cluster baru didapatkan pada iterasi 3 yang ditunjukkan pada Tabel 7.6.

Tabel 7.6 Hasil Pusat Cluster pada Iterasi 3

No	Cluster	CBR	CDR
1	**(C1)**	**28.5**	**5.75**
2	**(C2)**	**35.67**	**13.3**
3	**(C3)**	**20.3**	**6.3**

Step 2 : Hitung jarak setiap data terhadap setiap pusat cluster.

Jarak data pertama (Brunei Darusalam) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{11}} = \sqrt{(27 - \mathbf{28.5})^2 + (3 - \mathbf{5.75})^2} = \mathbf{3,1325}$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{12}} = \sqrt{(27 - \mathbf{37.67})^2 + (3 - \mathbf{12.3})^2} = \mathbf{14,8511}$$

Jarak data pertama (Brunei Darusalam) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{13}} = \sqrt{(27 - \mathbf{20.3})^2 + (3 - \mathbf{6.3})^2} = \mathbf{7,4536}$$

Jarak data kedua (Kamboja) dengan pusat cluster pertama adalah:

$$\boldsymbol{d_{21}} = \sqrt{(38 - \mathbf{28.5})^2 + (14 - \mathbf{5.75})^2} = \mathbf{12,5822}$$

Jarak data kedua (Kamboja) dengan pusat cluster kedua adalah:

$$\boldsymbol{d_{22}} = \sqrt{(38 - \mathbf{37.67})^2 + (14 - \mathbf{13.2})^2} = \mathbf{0,7454}$$

Jarak data kedua (kamboja) dengan pusat cluster ketiga adalah:

$$\boldsymbol{d_{23}} = \sqrt{(38 - \mathbf{20.3})^2 + (14 - \mathbf{6.3})^2} = \mathbf{19,2585}$$

Perhitungan data 3 – 10 sama dengan sebelumnya.

Step 3 : Suatu data akan menjadi anggota dari suatu cluster yang memiliki jarak terkecil dari pusat clusternya seperti pada Tabel 7.8.

Gambar 7.8 Hasil Pengelompokan pada Iterasi 4

No	Negara	Negara		Jarak			Anggota Cluster Lama			Anggota Cluster Baru		
		CBR	CDR	C1	C2	C3	C1	C2	C3	C1	C2	C3
1	Brunei Darusalam	27	3	**2.4037**	12.3794	7.4536	*			*		
2	Kamboja	38	14	13.2077	**3.2016**	19.2585		*			*	
3	Indonesia	24	8	5.2705	12.1758	**4.0277**			*			*
4	Laos	43	15	17.7514	**8.0777**	24.2670		*			*	
5	Malaysia	28	5	**0.3333**	10.2591	7.7817	*			*		
6	Myanmar	32	11	7.0317	**3.6401**	12.5654		*			*	
7	Filipina	30	7	**2.6034**	7.4330	9.6896	*			*		
8	Singapura	17	5	11.3333	19.7800	**3.5901**			*			*
9	Thailand	20	6	8.3931	16.6208	**0.4714**			*			*
10	Vietnam	29	8	**3.0732**	7.6322	8.8255	*			*		

Kesimpulan

Terlihat bahwa posisi data pada iterasi 4 sudah tidak mengalami perubahan, sehingga proses iterasi sudah dapat dihentikan. Hasil cluster datanya dapat dilihat pada Tabel 7.7.

Tabel 7.7 Hasil Akhir Pengelompokan Metode K-Means

No	Negara	Atribut		Klaster
		CBR	CDR	
1	Brunei Darusalam	27	3	**C1**
2	Kamboja	38	14	**C2**
3	Indonesia	24	8	**C3**
4	Laos	43	15	**C2**
5	Malaysia	28	5	**C1**
6	Myanmar	32	11	**C2**
7	Filipina	30	7	**C1**
8	Singapura	17	5	**C3**
9	Thailand	20	6	**C3**
10	Vietnam	29	8	**C1**

Kita bisa melakukan analisa hasil klaster yang terbentuk pada Tabel 7.7 berdasarkan karakteristik datanya seperti pada Cluster C1 terdapat 4 datanya yaitu Data 1, 5, 7 dan 10 termasuk negara

7.3 Tugas

Sebuah perusahaan melakukan penelitian terhadap data konsumen yang dimilikinya. Perusahaan akan melakukan pengelompokan data ke dalam 2 *Cluster* berdasarkan kriteria besaran gaji yang diterima dan pengeluaran perbulannya. Data-data sebanyak 20 konsumen dari perusahaan ditunjukkan pada Tabel 7.9.

Tabel 7.9 Data Konsumen

No	Gaji	Pengeluaran

1	**2500**	**1750**
2	**3800**	**4200**
3	**3900**	**3800**
4	**4350**	**5500**
5	**4400**	**3200**
6	**5500**	**5450**
7	**5600**	**5950**
8	**5750**	**4100**
9	**6850**	**6050**
10	**6900**	**8500**
11	**7250**	**9500**
12	**7350**	**6050**
13	**7500**	**8500**
14	**7800**	**9500**
15	**8200**	**6300**
16	**8500**	**6500**
17	**8550**	**8400**
18	**8750**	**6000**
19	**9100**	**10500**
20	**9100**	**8500**

BAB VIII
ASSOCIATION RULE

8.1 Konsep Analisis Asosiasi

Penambangan Aturan Asosiasi, seperti namanya, aturan asosiasi adalah pernyataan If/Then sederhana yang membantu menemukan hubungan antara database relasional yang tampaknya independen atau repositori data lainnya. Sebagian besar algoritma pembelajaran mesin bekerja dengan kumpulan data numerik dan karenanya cenderung bersifat matematis. Namun, penambangan aturan asosiasi cocok untuk data non-numerik, kategorikal dan hanya membutuhkan sedikit lebih banyak daripada penghitungan sederhana. Penambangan aturan asosiasi adalah prosedur yang bertujuan untuk mengamati pola, korelasi, atau asosiasi yang sering terjadi dari kumpulan data yang ditemukan di berbagai jenis basis data seperti basis data relasional, basis data transaksional, dan bentuk repositori lainnya.

Aturan Asosiasi adalah teknik pembelajaran yang membantu mengidentifikasi ketergantungan antara dua item data. Berdasarkan ketergantungan tersebut kemudian dipetakan sedemikian rupa sehingga dapat lebih menguntungkan. Aturan asosiasi selanjutnya mencari asosiasi yang menarik di antara variabel-variabel dari kumpulan data. Tidak diragukan lagi ini adalah salah satu konsep Pembelajaran Mesin yang paling penting dan telah digunakan dalam berbagai kasus seperti asosiasi dalam penambangan data dan produksi berkelanjutan, antara lain.

8.2 Studi Kasus

Diketahui 10 transaksi pembelian di sebuah mini market seperti pada Tabel 8.1.

Tabel 8.1 Data Transaksi

Transaksi	Item Dibeli
1	**Susu, Teh, Gula**

2	**Teh, Gula, Roti**
3	**The, Gula**
4	**Susu, Roti**
5	**Susu, Gula, Roti**
6	**The, Gula**
7	**Gula, Kopi, Susu**
8	**Gula, Kopi, Susu**
9	**Susu, Roti, Kopi**
10	**Gula, The, Kopi**

Penyelesaian

Data transaksi pada Tabel 8.1 perlu dijadikan data tabular transaksi seperti pada Tabel 8.2

Tabel 8.2 Data Tabular Transaksi

Transaksi	Teh	Gula	Kopi	Susu	Roti
1	1	1	0	1	0
2	1	1	0	0	1
3	1	1	0	0	0
4	0	0	0	1	1
5	0	1	0	1	1
6	1	1	1	0	0
7	0	1	0	1	0
8	0	1	1	1	0
9	0	0	1	1	1
10	1	1	1	0	0

- **1 Itemset**

Barang	Jumlah
Gula	8
Susu	6

Teh	5
Roti	4
Kopi	4
Jumlah Transaksi	10

Menghitng nilai Support pada 1 Itemset seperti berikut:

$$Support\left(A\right)=\frac{Jumlah\ Transaksi\ Berisi\ A}{Total\ Transaksi}$$

$$Support\left(Gula\right)=\frac{Jumlah\ Transaksi\ Berisi\ \text{Gula}}{Total\ Transaksi}=\frac{8}{10}=0,8$$

$$Support\left(Susu\right)=\frac{Jumlah\ Transaksi\ Berisi\ \text{Susu}}{Total\ Transaksi}=\frac{6}{10}=0,6$$

Hasil masing-masing nilai support ditampilkan pada tabel berikut:

Barang	Support
Gula	8/10 = **0,8**
Susu	6/10 = **0,6**
The	5/10 = **0.5**
Roti	4/10 = **0,4**
Kopi	4/10 = **0,4**
Jumlah Transaksi	**10**

- **2-Item Set**

Kombinasi	Jumlah
Teh, Gula	5
Teh, Kopi	1
Teh, Susu	1
Teh, Roti	1
Gula, Kopi	3
Gula, Susu	4
Gula, Roti	2
Kopi, Susu	3
Kopi, Roti	1

Susu, Roti	3
Jumlah Transaksi	

Mencari Nilai Support

$$Support\left(A \cap B\right) = \frac{Jumlah\ Transaksi\ Berisi\ A\ dan\ B}{Total\ Transaksi}$$

$$Support\left(Teh \cap Gula\right) = \frac{Jumlah\ Transaksi\ Berisi\ \text{Teh}\ dan\ \text{Gula}}{Total\ Transaksi} = \frac{5}{10} = 0.5*100 = 50\%$$

$$Support\left(Teh \cap Kopi\right) = \frac{Jumlah\ Transaksi\ Berisi\ \text{Teh}\ dan\ \text{Kopi}}{Total\ Transaksi} = \frac{1}{10} = 0.1*100 = 10\%$$

Mencari Nilai Confidence

$$Confidence\left(A => B\right) = \frac{Jumlah\ Transaksi\ Berisi\ A\ dan\ B}{\text{Jumlah}\ Transaksi\ A}$$

$$Confidence\left(Teh => Gula\right) = \frac{Jumlah\ Transaksi\ Berisi\ \text{Teh}\ dan\ \text{Gula}}{\text{Jumlah}\ Transaksi\ \text{Teh}} = \frac{5}{5} = 1*100 = 100\%$$

$$Confidence\left(Teh => Kopi\right) = \frac{Jumlah\ Transaksi\ Berisi\ \text{Teh}\ dan\ \text{Kopi}}{\text{Jumlah}\ Transaksi\ \text{Teh}} = \frac{1}{5} = 0,2*100 = 20\%$$

Mencari Nilai Lift Ratio

$$Lift\ Ratio\ \left(A => B\right) = \frac{Confidence\left(A => B\right)}{\text{Support (B)}}$$

$$Lift\ Ratio\ \left(Teh => Gula\right) = \frac{Confidence\left(Teh => Gula\right)}{\text{Support (Gula)}} = \frac{1}{0.8} = 1,25$$

$$Lift\ Ratio\ \left(Teh => Kopi\right) = \frac{Confidence\left(Teh => Kopi\right)}{\text{Support (Gula)}} = \frac{0.2}{0.4} = 0,5$$

Kombinasi	Support	Confidence	Lift Ratio
Jika Beli Teh, Maka Beli Gula	50%	100%	1 / 0.8= **1,25**
Jika Beli Teh, Maka Beli Kopi	10%	20%	0.2 / 0.4 = 0,5
Jika Beli Teh, Maka Beli Susu	10%	20%	0,33
Jika Beli Teh, Maka Beli Roti	10%	20%	0,33

Jika Beli Gula, Maka Beli Kopi	30%	37,5%	0,94
Jika Beli Gula,Maka Beli Susu	40%	50%	0,83
Jika Beli Gula, Maka Beli Roti	20%	25%	0,63
Jika Beli Kopi, Maka Beli Susu	30%	75%	1,25
Jika Beli Kopi, Maka Beli Roti	20%	25%	0,63
Jika Beli Susu,Maka Beli Roti	30%	50%	1,25

Jika Menggunakan **Minimal Support >= 40** dan **Confidence >= 60%** maka aturan yang bisa dipakai adalah **nomor 1**.

- **3-Item Set**

Kombinasi	Jumlah
Teh, Gula, Kopi	1
Teh, Gula, Susu	1
Gula, Susu, Kopi	2
Gula, Susu, Roti	0
Gula, Kopi, Roti	0
Kopi, Susu, Roti	1

Mencari Nilai Support

$$Support\left(A \cap B \cap C\right) = \frac{Jumlah\ Transaksi\ Berisi\ A\ dan\ B\ dan\ C}{Total\ Transaksi}$$

$$Support\left(Teh \cap Gula \cap Kopi\right) = \frac{Jumlah\ Transaksi\ Berisi\ \text{Teh}\ dan\ \text{Gula dan Kopi}}{Total\ Transaksi}$$

$$= \frac{1}{10} = 0.1*100 = 10\%$$

$$Support\left(Teh \cap Gula \cap Susu\right) = \frac{Jumlah\ Transaksi\ Berisi\ \text{Teh}\ dan\ \text{Gula dan Susu}}{Total\ Transaksi}$$

$$= \frac{1}{10} = 0.1*100 = 10\%$$

Mencari Nilai Confidence

$$Confidence\left(A,B => C\right) = \frac{Jumlah\ Transaksi\ Berisi\ A\ dan\ B \text{ dan C}}{\text{Jumlah } Transaksi\ A \text{ dan B}}$$

$$Confidence\left(Teh,Gula => Kopi\right) = \frac{Jumlah\ Transaksi\ Berisi \text{ Teh } dan \text{ Gula dan Kopi}}{\text{Jumlah } Transaksi \text{ Teh dan Gula}}$$

$$= \frac{1}{5} = 0.2 * 100 = 20\%$$

$$Confidence\left(Teh,Gula => Susu\right) = \frac{Jumlah\ Transaksi\ Berisi \text{ Teh } dan \text{ Gula dan Susu}}{\text{Jumlah } Transaksi \text{ Teh dan Gula}}$$

$$= \frac{1}{5} = 0,2 * 100 = 20\%$$

Mencari Nilai Lift Ratio

$$Lift\ Ratio\ (A,B => C) = \frac{Confidence\left(A,B => C\right)}{\text{Support (C)}}$$

$$Lift\ Ratio\ (Teh,Gula => Kopi) = \frac{Confidence\left(Teh,Gula => Kopi\right)}{\text{Support (Kopi)}}$$

$$= \frac{0,2}{0.4} = 0,5$$

$$Lift\ Ratio\ (Teh,Gula => Susu) = \frac{Confidence\left(Teh => Gula\right)}{\text{Support (Susu)}}$$

$$= \frac{0.2}{0.6} = 0,33$$

sss	Support	Confidence	Lift Ratio
Jika Beli Teh dan Gula, Maka Beli Kopi	10%	20%	0.2 / 0.4= 0.5
Jika Beli Teh dan Gula, Maka Beli Susu	10%	20%	0.2 / 0.6 = 0,3
Jika Beli Gula dan Susu, Maka Beli Kopi	20%	50%	1.25
Jika Beli Kopi dan Susu Maka Beli Roti	10%	33%	0.83

Jika Menggunakan **Minimal Support >= 20** dan **Confidence >= 40%** maka aturan yang bisa dipakai adalah **nomor 3**.

8.3 Tugas

Diketahui transaksi seperti berikut:

Transaction	Item
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke

Tentukan aturan asosiasi yang terbentuk dengan ketentuan syarat minimum :

- **Support = 40%**
- **Confidence = 60%**

BAB IX
PRINCIPAL COMPONENT ANALISYS

9.1 Konse PCA

Principal Component Analysis (PCA) adalah salah satu metode reduksi dimensi pada machine learning. PCA akan memilihvariabel-variabel yang mampu menjelaskan sebagian besar variabilitas data. PCA mengurangi dimensi dengan membentuk variabel-variabel baru yang disebut Principal Components. Principal Components yang merupakan kombinasi linier dari variabel-variabel lama. Penghitungan Varians dan Principal Component ini dapat dilakukan dengan menggunakan konsep nilai eigen (eigenvalue) dan vektor eigen (eigenvector) dari ilmu Aljabar Linier.

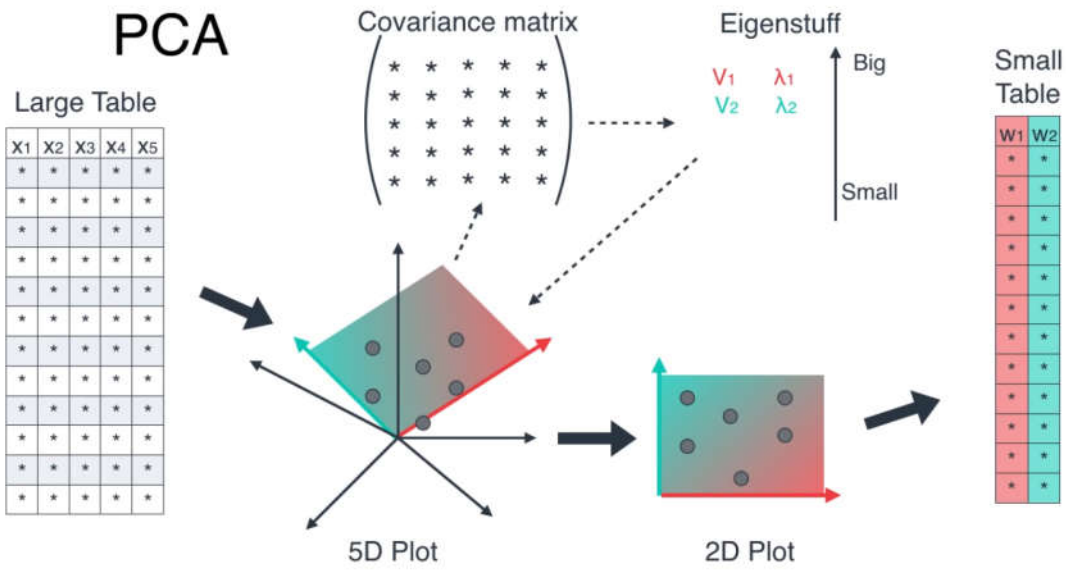
Principal Component Analysis (PCA) tentunya punya banyak manfaat dalam proses analisis data, misal:

a. Mengatasi multikolinieritas yang pada metode parametrik tertentu merupakan asumsi yang harus dipenuhi;
b. Mereduksi jumlah variabel yang akan dimasukkan ke model;
c. Jumlah variabel yang lebih sedikit tentu akan menyederhanakan model;
d. Juga mempercepat proses komputasi.

Adapun proses metode PCA dalam reduksi dimensi data sebagai berikut:

Langkah 1: Standarisasi dataset.
Langkah 2: Hitung matriks kovarians untuk fitur-fitur dalam kumpulan data.
Langkah 3: Hitung nilai eigen dan vektor eigen untuk matriks kovarians.
Langkah 4: Urutkan nilai eigen dan vektor eigen yang sesuai.
Langkah 5: Pilih k nilai eigen dan bentuk matriks vektor eigen.
Langkah 6: Transformasikan matriks asli.

Agar lebih mudah mengetahui proses kerja metode PCA dapat diilustrasikan pada Gambar 9.1.



Gambar 9.1 Proses Kerja Metode PCA

9.2 Studi Kasus

1. Standarisasi Dataset

Asumsikan kita memiliki dataset di bawah ini yang memiliki 4 fitur dan total 5 contoh pelatihan.

f1	f2	f3	f4
1	2	3	4
5	5	6	7
1	4	2	3
5	3	2	1
8	1	2	2

Pertama, kita perlu membakukan dataset dan untuk itu, kita perlu menghitung rata-rata dan standar deviasi untuk setiap fitur.

$$x_{new} = \frac{x - \mu}{\sigma}$$

	f1	f2	f3	f4
μ =	4	3	3	3.4
σ =	3	1.58114	1.73205	2.30217

Rata-rata dan standar deviasi sebelum standarisasi

Setelah menerapkan rumus untuk setiap fitur dalam dataset ditransformasikan seperti di bawah ini:

f1	f2	f3	f4
-1	-0.63246	0	0.26062
0.33333	1.26491	1.73205	1.56374
-1	0.63246	-0.57735	-0.17375
0.33333	0	-0.57735	-1.04249
1.33333	-1.26491	-0.57735	-0.60812

Standarisasi Dataset

2.Hitung matriks kovarian untuk seluruh dataset, rumus untuk menghitung matriks kovarians:

For Population

$$Cov(x,y) = \frac{\sum (x_i - \bar{x}) * (y_i - \bar{y})}{N}$$

For Sample

$$Cov(x,y) = \frac{\sum (x_i - \bar{x}) * (y_i - \bar{y})}{(N-1)}$$

The covariance matrix for the given dataset will be calculated as below

	f1	f2	f3	f4
f1	var(f1)	cov(f1,f2)	cov(f1,f3)	cov(f1,f4)
f2	cov(f2,f1)	var(f2)	cov(f2,f3)	cov(f2,f4)
f3	cov(f3,f1)	cov(f3,f2)	var(f3)	cov(f3,f4)
f4	cov(f4,f1)	cov(f4,f2)	cov(f4,f3)	var(f4)

Karena kami telah menstandarkan dataset, maka rata-rata untuk setiap fitur adalah 0 dan standar deviasinya adalah 1.

var(f1) = ((-1.0-0)² + (0.33-0)² + (-1.0-0)² +(0.33–0)² +(1.33–0)²)/5

var (f1) = 0.8

cov(f1,f2) =

((-1.0–0)*(-0.632456-0) +

(0.33–0)*(1.264911-0) +

(-1.0–0)* (0.632456-0)+

(0.33–0)*(0.000000 -0)+

(1.33–0)*(-1.264911–0))/5

cov(f1,f2 = -0.25298

Dengan cara yang sama dapat menghitung kovarians lain dan yang akan menghasilkan matriks kovarians di bawah ini

	f1	f2	f3	f4
f1	0.8	-0.25298	0.03849	-0.14479
f2	-0.25298	0.8	0.51121	0.4945
f3	0.03849	0.51121	0.8	0.75236
f4	-0.14479	0.4945	0.75236	0.8

matriks kovarians (rumus populasi)

3. Menghitung nilai eigen dan vektor eigen.

Vektor eigen adalah vektor bukan nol yang berubah paling banyak oleh faktor skalar ketika transformasi linier diterapkan padanya. Nilai eigen yang sesuai adalah faktor yang digunakan untuk menskalakan vektor eigen. Misalkan A adalah matriks bujur sangkar (dalam kasus kita matriks kovarians), ν vektor dan λ skalar yang memenuhi $A\nu = \lambda\nu$, maka λ disebut nilai eigen yang diasosiasikan dengan vektor eigen ν dari A.

Mengatur ulang persamaan di atas,

$$A\nu-\lambda\nu =0 \; ; \; (A-\lambda I)\nu = 0$$

Karena telah mengetahui ν adalah vektor bukan nol, satu-satunya cara persamaan ini bisa sama dengan nol, jika

det(A-λI) = 0

	f1	f2	f3	f4
f1	0.8 - λ	-0.25298	0.03849	-0.14479
f2	-0.25298	0.8- λ	0.51121	0.4945
f3	0.03849	0.51121	0.8 - λ	0.75236
f4	-0.14479	0.4945	0.75236	0.8 - λ

A-λI = 0

Selesaikan persamaan di atas = 0

λ = 2,51579324 , 1,0652885 , 0,39388704 , 0,02503121

Vektor eigen:

Selesaikan persamaan (A-λI)v = 0 untuk vektor v dengan nilai λ yang berbeda:

$$\begin{pmatrix} 0.800000-\lambda & -(0.252982) & 0.038490 & -(0.144791) \\ -(0.252982) & 0.800000-\lambda & 0.511208 & 0.494498 \\ 0.038490 & 0.511208 & 0.800000-\lambda & 0.752355 \\ -(0.144791) & 0.494498 & 0.752355 & 0.800000-\lambda \end{pmatrix} \times \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{pmatrix} = 0$$

Untuk λ = 2.51579324, selesaikan persamaan di atas menggunakan aturan Cramer, nilai vektor v adalah

v1 = 0,16195986

v2 = -0,52404813

v3 = -0,58589647

v4 = -0,59654663

Dengan pendekatan yang sama, kita dapat menghitung vektor eigen untuk nilai eigen lainnya, dapat dari matriks menggunakan vektor eigen.

```
        e1        e2        e3        e4
 0.161960 -0.917059 -0.307071  0.196162
-0.524048  0.206922 -0.817319  0.120610
-0.585896 -0.320539  0.188250 -0.720099
-0.596547 -0.115935  0.449733  0.654547
```

vektor eigen (matriks 4 * 4)

4. Urutkan nilai eigen dan vektor eigen yang sesuai.

Karena nilai eigen sudah diurutkan dalam hal ini, maka tidak perlu mengurutkannya lagi.

5. Pilih k nilai eigen dan bentuk matriks vektor eigen

Jika kita memilih 2 vektor eigen teratas, matriks akan terlihat seperti ini:

e1	e2
0.161960	-0.917059
-0.524048	0.206922
-0.585896	-0.320539
-0.596547	-0.115935

2 vektor eigen teratas (matriks 4*2)

6. Transformasikan matriks asli.

Matriks fitur * top k eigenvectors = Data yang Diubah

f1	f2	f3	f4
-1.000000	-0.632456	0.000000	0.260623
0.333333	1.264911	1.732051	1.563740
-1.000000	0.632456	-0.577350	-0.173749
0.333333	0.000000	-0.577350	-1.042493
1.333333	-1.264911	-0.577350	-0.608121

(5,4)

*

e1	e2
0.161960	-0.917059
-0.524048	0.206922
-0.585896	-0.320539
-0.596547	-0.115935

(4,2)

=

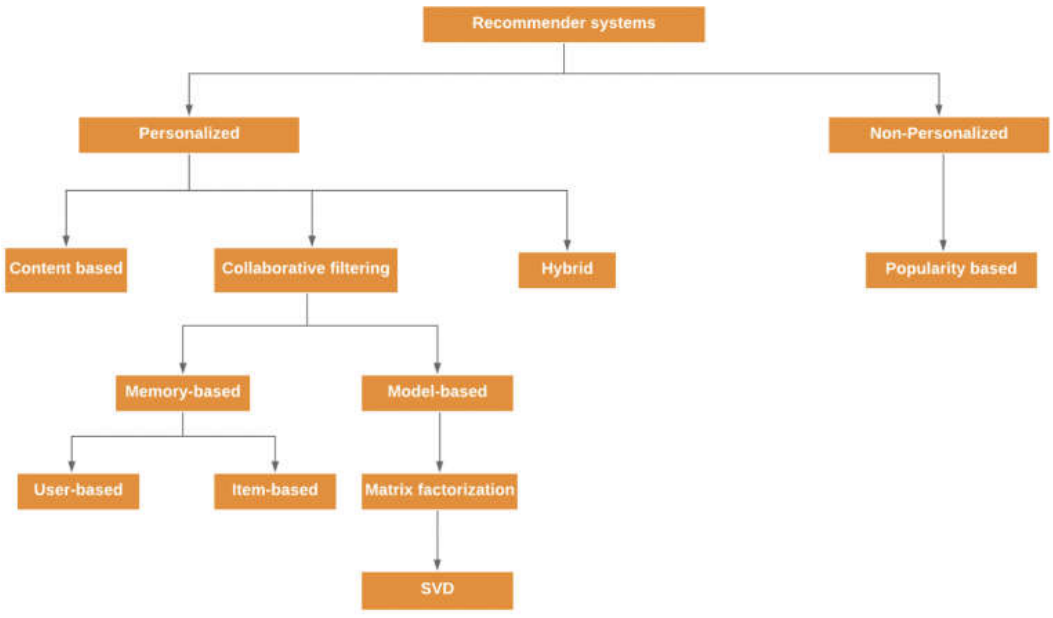
nf1	nf2
0.014003	0.755975
-2.556534	-0.780432
-0.051480	1.253135
1.014150	0.000239
1.579861	-1.228917

(5,2)

BAB X
RECOMMENDATION SYSTEM

10.1 Konsep *Recommendation System*

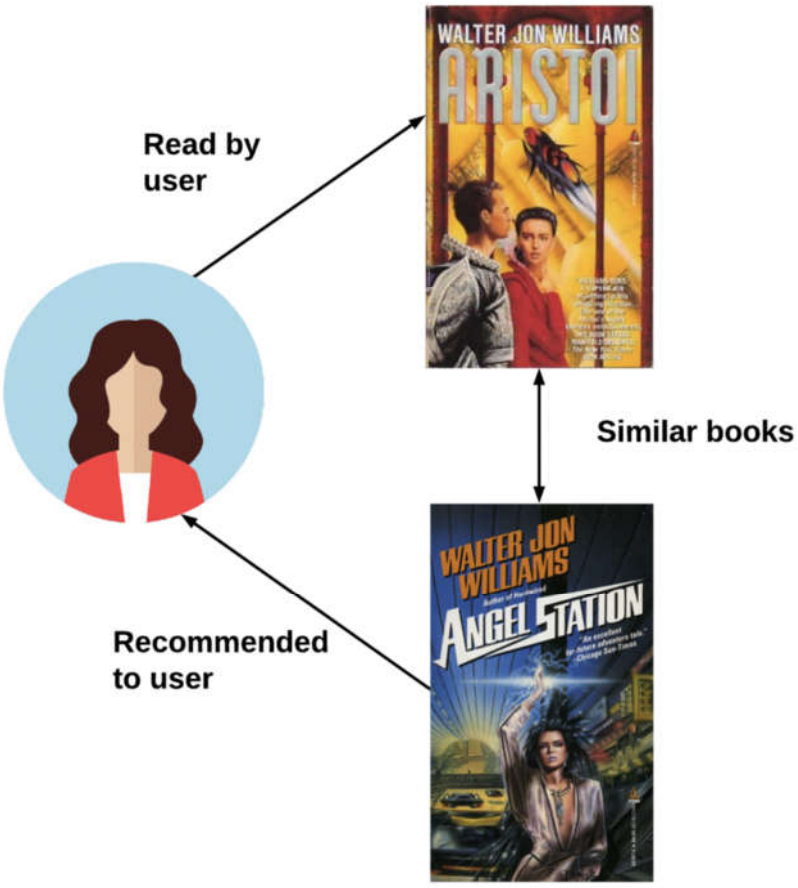
Sistem rekomendasi adalah sistem yang membantu pelanggan/pengguna menemukan produk atau layanan yang mungkin disukai. Ini seperti seorang penjual perusahaan yang tahu apa yang mungkin disukai pelanggan berdasarkan sejarah dan preferensi mereka. Pengumpulan data ada di mana-mana sekarang. Setiap aplikasi yangdigunakan di internet mengumpulkan data tentang aktivitas Anda, tentang bagaimana berinteraksi dengan sesuatu, apa yang dicari, dengan siapa Anda berinteraksi, dll. Aplikasi tersebut mengenal Anda lebih baik daripada diri Anda sendiri. Salah satu aplikasi paling umum dari data yang dikumpulkan ini adalah Recommender Systems. Sistem rekomendasi adalah aplikasi pembelajaran mesin yang memprediksi preferensi masa depan dari serangkaian produk untuk pengguna dan memberikan saran yang dipersonalisasi. Ada banyak jenis sistem rekomendasi yang tersedia saat ini seperti yang ditunjukkan pada Gambar 10.1.



Gambar 10.1 Jenis Sistem Rekomendasi

1) Content-Based Filtering

Gagasan utama Penyaringan Berbasis Konten adalah menyarankan item berdasarkan item tertentu. Misalnya, saat Anda membuat sistem rekomendasi film, sistem ini akan mempertimbangkan preferensi pengguna untuk sebuah film menggunakan metrik seperti peringkat, lalu menggunakan metadata item, seperti genre, sutradara, deskripsi film, pemeran, dan kru, dll untuk menemukan film yang mirip dengan yang disukai pengguna. Asumsikan Jenny menyukai buku fiksi ilmiah dan penulis favoritnya adalah Walter Jon Williams. Jika dia membaca buku Aristoi, maka buku rekomendasinya adalah Angel station, juga buku sci-fi yang ditulis oleh Walter Jon Williams (Gambar 10.2).



Gambar 10.2 Contoh Content-Based Filtering

2) Collaborative Filtering

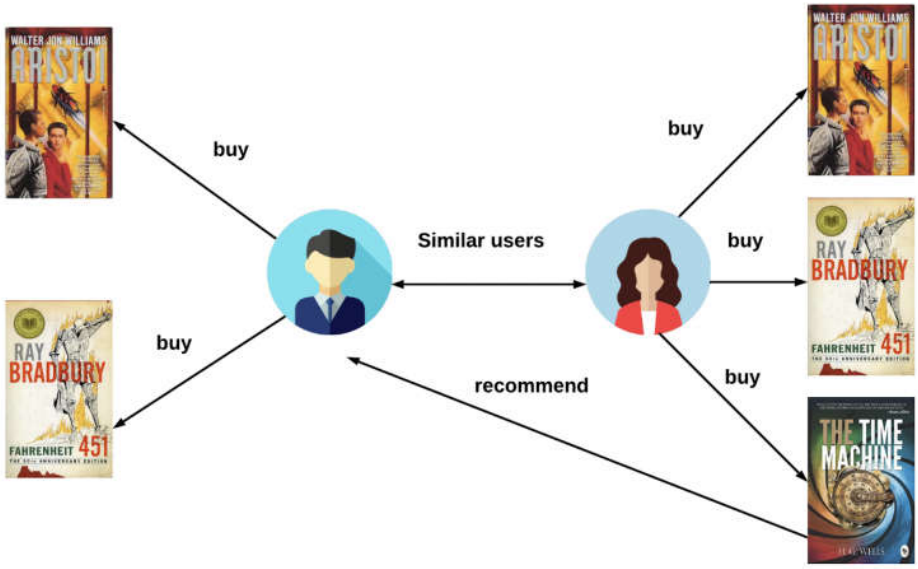
Teknik rekomendasi pemfilteran kolaboratif bergantung pada menemukan pengguna yang mirip dengan pengguna target untuk membuat rekomendasi yang

dipersonalisasi. Sistem rekomendasi penyaringan kolaboratif tidak memerlukan metadata item seperti sistem rekomendasi berbasis konten. Itu hanya bergantung pada interaksi pengguna-item sebelumnya untuk membuat rekomendasi baru.

Ada dua jenis pemfilteran kolaboratif yaitu berbasis pengguna (user) dan berbasis item.

Berbasis pengguna : "Pengguna yang mirip dengan Anda juga menyukai"

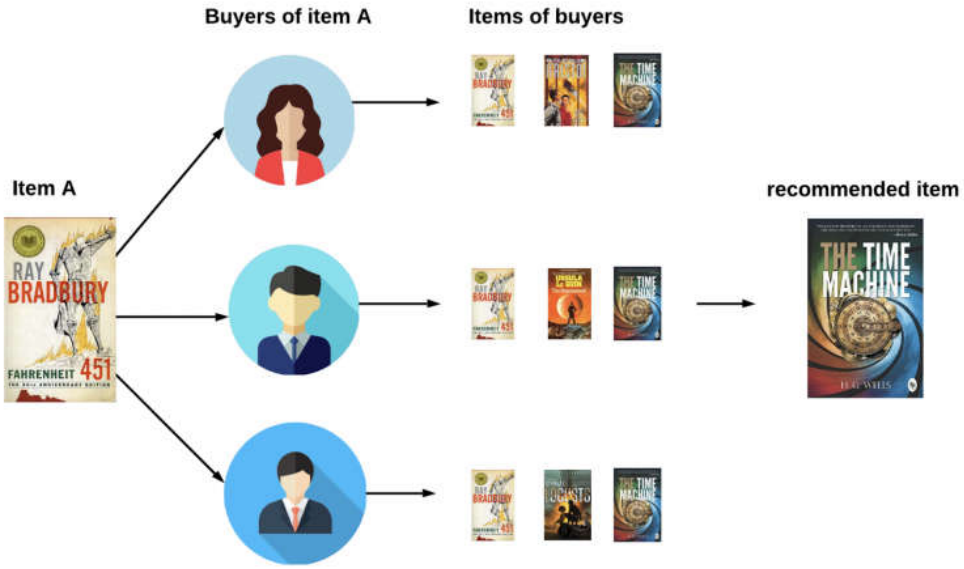
Produk direkomendasikan kepada pengguna berdasarkan fakta bahwa mereka dibeli / disukai oleh pengguna yang mirip dengan pengguna yang diamati. Jika kami mengatakan bahwa pengguna serupa, apa artinya? Misalnya, Jenny dan Tom menyukai buku fiksi ilmiah. Ketika buku sci-fi baru muncul dan Jenny membeli buku itu, karena Tom juga menyukai buku sci-fi maka kami dapat merekomendasikan buku yang dibeli Jenny.



Gambar 10.3 Contoh Rekomendasi Berbasis User

Berdasarkan item – "Pengguna yang menyukai item ini juga menyukai"

Jika John, Robert, dan Jenny menilai tinggi buku sci-fi Fahrenheit 451 dan Mesin waktu, misalnya memberi 5 bintang, maka ketika Tom membeli buku Fahrenheit 451 maka buku The mesin waktu juga direkomendasikan kepadanya karena sistem mengidentifikasi buku-buku serupa berdasarkan peringkat pengguna.



Gambar 10.4 Contoh Sistem Rekomendasi Colaborative Filtering Berbasis Item

10.2 Studi Kasus (Item-Based Collaborative Filtering.)

Misalkan sebuah dataset memiliki peringkat film yang diberikan oleh pengguna yang berbeda dalam format tabel seperti yang ditunjukkan pada Tabel dibawah ini.

User	Movie	Rating
Amy	Pulp Fiction	4
Amy	The GodFather	5
Calvin	Pulp Fiction	5
Robert	Pulp Fiction	3
Calvin	Forrest Gump	2
Robert	Forrest Gump	3
Robert	The GodFather	1
David	Forrest Gump	2
Bradley	Pulp Fiction	1
David	The GodFather	2
Bradley	Forrest Gump	3

Langkah 1: Membuat sparse matriks tempat kami menulis peringkat item pengguna dalam bentuk matriks

	Amy	Calvin	Robert	David	Bradley
Pulp Fiction	4	5	3	?	1
Forrest Gump	?	2	3	2	3
The GodFather	5	?	1	2	?

Di pengguna matriks ini, Amy sudah menilai dan menonton film Pulp Fiction dan The GodFather tetapi belum menonton film Forrest Gump. Kami akan menggunakan matriks di atas untuk contoh kami dan akan mencoba membuat matriks kesamaan item-item menggunakan **metode Cosine Similarity** untuk menentukan seberapa mirip film satu sama lain. Adapun rumus Cosine similiaritu sebagai berikut:

$$\text{Similarity}(p,q) = \cos\theta = \frac{p \cdot q}{\|p\|\|q\|} = \frac{\sum_{i=1}^{n} p_i q_i}{\sqrt{\sum_{i=1}^{n} p_i^{\,2}}\sqrt{\sum_{i=1}^{n} q_i^{\,2}}}$$

Cosinus sudut antara dua vektor diperoleh dari perkalian titik kedua vektor tersebut dibagi dengan hasil kali besar kedua vektor tersebut.

Langkah 2: Untuk menghitung kemiripan antara film Pulp Fiction (P) dan Forrest Gump (F), pertama-tama kita akan menemukan semua pengguna yang telah menilai kedua film tersebut. Dalam kasus kami, Calvin (C), Robert (R), dan Bradley (B) telah menilai film-film tersebut. Kami sekarang membuat dua vektor:

v1 = 5 C + 3 R + 1 B

v2 = 2 C + 3 R + 3 B

Maka Cosine Kemiripan antara film Pulp Fiction dan Forrest Gump adalah:

cos(v1,v2) = (5*2 + 3*3 + 1*3) / sqrt[(25+9+1) * (4+9+9)] = 0.792

Demikian pula, kita dapat menghitung kesamaan cosinus dari semua film dan matriks kesamaan terakhir kita adalah:

Langkah 3: Sekarang kita dapat memprediksi dan mengisi peringkat untuk pengguna untuk item yang belum diberi peringkat. Jadi untuk menghitung peringkat pengguna Amy untuk film Forrest Gump, kami akan menggunakan matriks kesamaan yang dihitung bersama dengan film yang sudah diberi peringkat oleh pengguna. Oleh karena itu, peringkatnya adalah:

(4*0.792 + 5*0.8) / (0.792+ 0.8) = 4.5

Oleh karena itu, matriks terakhir kita adalah:

	Pulp Fiction	Forrest Gump	The GodFather
Amy	4	**4.5**	5
Calvin	5	2	**4.01**
Robert	3	3	1
David	**2**	2	2
Bradley	1	3	**1.94**

10.3 Tugas

Seperti yang ditunjukkan pada Tabel dibawah, ada tiga peringkat pengguna pada lima film. Rentang peringkat adalah dari 1 hingga 5, di mana 1 menunjukkan bahwa pengguna sama sekali tidak menyukai film tersebut, dan 5 menunjukkan bahwa pengguna menyukainya; sedangkan tanda tanya berarti film tersebut belum diberi rating oleh pengguna.

	Alex	Bob	Tom
Avengers	4	5	3
Star wars	2	3	?
Thor	?	4	4
Spider-man	5	?	4
Iron Man	4	3	3

BAB XI
METRIK EVALUASI KINERJA

11.1 Konsep Evaluasi Kinerja

Dalam Pembelajaran Mesin, metrik evaluasi kinerja digunakan untuk menghitung kinerja model pembelajaran mesin pada data testing. Ini membantu dalam menemukan seberapa baik model pembelajaran mesin dapat bekerja pada kumpulan data yang belum pernah dilihatnya sebelumnya.

Jenis metrik pembelajaran mesin

Ada beberapa metrik di luar sana untuk mengevaluasi model ML di berbagai aplikasi. Sebagian besar dapat dimasukkan ke dalam dua kategori berdasarkan jenis prediksi dalam model ML. Klasifikasi adalah jenis prediksi yang digunakan untuk memberikan variabel keluaran dalam bentuk kategori dengan atribut yang mirip. Misalnya, model seperti itu dapat memberikan keluaran biner seperti menyortir pesan spam dan non-spam.

Beberapa metrik klasifikasi populer yang akan kami bahas adalah

Akurasi atau Ketepatan,

Presisi,

Ingat, dan

Skor F1, dll.

Regresi adalah sejenis prediksi di mana variabel keluarannya numerik, bukan kategoris (berlawanan dengan klasifikasi). Outputnya terus menerus. Misalnya, ini dapat membantu memprediksi lama tinggal pasien di rumah sakit.

Beberapa metrik regresi populer yang akan kami bahas adalah

MSE (Mean Squared Error),

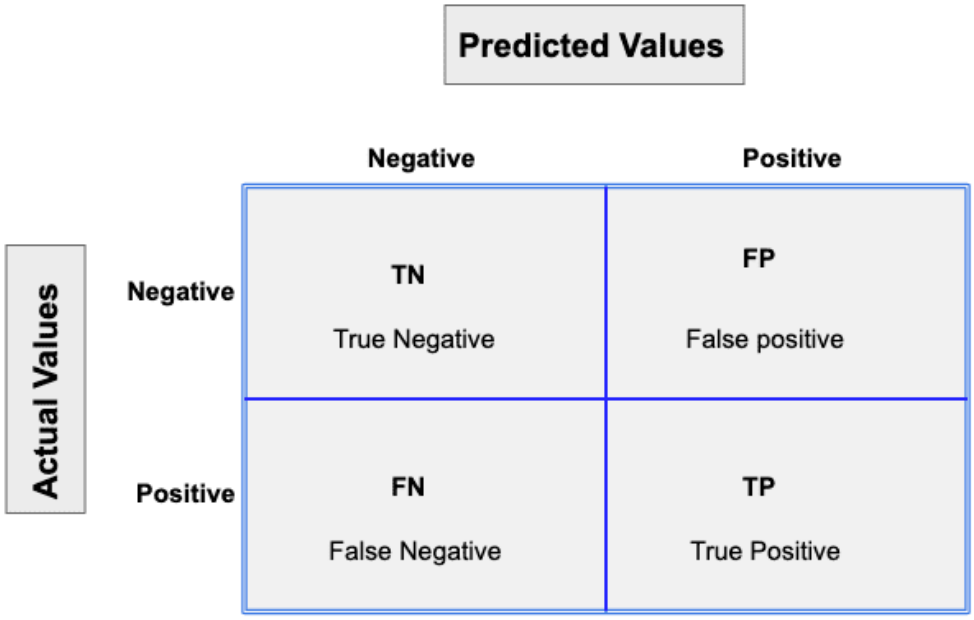
RMSE (Root Mean Squared Error), dan

MAE (Mean Absolute Error).

1. **Confusion Matriks**

Confusion matrix adalah representasi matriks dari hasil prediksi dari setiap pengujian biner yang sering digunakan untuk menggambarkan kinerja model klasifikasi (atau "pengklasifikasi") pada sekumpulan data uji yang nilai sebenarnya

diketahui. Matriks konfusi itu sendiri relatif sederhana untuk dipahami, tetapi terminologi terkait dapat membingungkan.



Gambar 11.1 Tabel Confusion Matriks

Setiap prediksi dapat menjadi salah satu dari empat hasil, berdasarkan kesesuaiannya dengan nilai sebenarnya:

True Positive (TP): Benar yang Diprediksi dan Benar dalam kenyataan.

True Negative (TN): Diprediksi Salah dan Salah pada kenyataannya.

False Positive (FP): Diprediksi Benar dan Salah dalam kenyataan.

False Negative (FN): Diprediksi Salah dan Benar dalam kenyataan.

Let's get closer to the metrics.

Accuracy

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{TP + TN}{TP + FP + TN + FN}$$

Akurasi digunakan untuk menghitung proporsi jumlah total prediksi yang benar. Ini adalah jumlah prediksi yang benar dibagi dengan jumlah total prediksi.

Mengapa menggunakannya? Menjadi salah satu metrik klasifikasi yang paling umum, akurasi sangat intuitif dan mudah dipahami dan diterapkan: Berkisar dari 0 hingga 100 persen atau 0 hingga 1. Jika Anda berurusan dengan kasus pemodelan sederhana, akurasi dapat membantu. Selain itu, Anda dapat menemukannya di perpustakaan ML seperti Scikit-learn untuk model klasifikasi apa pun dengan metode skor.

Jika kita mengambil model diagnosis sehat/sakit, dari semua 10.000 pasien, model tersebut dengan tepat mengklasifikasikan 9.000 pasien atau 90 persen atau 0,9 jika kita mengukur dalam rentang dari 0 sampai 1. Jadi, itulah angka akurasi kita. Penting untuk dipahami. Meskipun intuitif, metrik akurasi sangat bergantung pada spesifikasi data. Jika kumpulan data tidak seimbang (kelas-kelas dalam kumpulan disajikan secara tidak merata), hasilnya tidak akan menjadi sesuatu yang dapat Anda percayai. Misalnya, dalam rangkaian pelatihan, Anda memiliki 98 persen sampel kelas A (pasien sehat) dan hanya 2 persen sampel kelas B (pasien sakit). Model ini dapat dengan mudah memberi Anda akurasi pelatihan 98 persen hanya dengan memprediksi setiap pasien sehat meskipun mereka memiliki penyakit serius. Tak perlu dikatakan bahwa hasil miring tersebut dapat memiliki konsekuensi buruk karena orang tidak akan mendapatkan bantuan medis yang dibutuhkan.

Precision

$$\text{Precision} = \frac{\text{Number of correct positive results}}{\text{Total number of positive results}} = \frac{\text{True Positives}}{\text{True Positives + False Positives}}$$

Presisi menunjukkan berapa proporsi dari semua prediksi positif yang benar. Untuk menghitungnya, Anda membagi jumlah hasil positif yang benar (TP) dengan jumlah total semua hasil positif (TP + FP) yang diprediksi oleh pengklasifikasi.

Kembali ke contoh kita, dari semua pasien model yang didiagnosis sakit, berapa banyak yang diklasifikasikan dengan benar? Kami membagi jumlah 1.000 pasien yang benar-benar sakit dan diperkirakan sakit dengan jumlah total mereka

yang benar-benar sakit dan didiagnosis sakit (1.000) dan mereka yang sehat tetapi didiagnosis sakit (800). Hasil presisinya mencapai 55,7 persen.

Mengapa menggunakannya? Presisi bekerja dengan baik jika Anda perlu atau dapat menghindari Negatif Palsu tetapi tidak dapat mengabaikan Positif Palsu. Contoh tipikal dari ini adalah model pendeteksi spam. Tidak apa-apa jika model mengirim beberapa surat spam ke kotak masuk, tetapi mengirim email penting non-spam ke folder spam (Positif Palsu) jauh lebih buruk.

Penting untuk dipahami. Presisi adalah metrik evaluasi masuk Anda saat menangani data yang tidak seimbang. Tapi ini bukan peluru perak karena ada kasus ketika negatif palsu dan negatif benar harus diperhitungkan. Misalnya, ketika penting untuk mengetahui berapa banyak orang yang benar-benar sakit yang diklasifikasikan sebagai sehat dan dibiarkan tanpa bantuan.

Recall

$$\text{Recall} = \frac{\text{Number of correct positives}}{\text{Number of all positives}} = \frac{\text{True Positives}}{\text{True Positives + False Negatives}}$$

Recall menunjukkan proporsi prediksi positif yang benar dari semua hal positif yang dapat dibuat oleh model. Untuk menghitungnya, bagi semua True Positives dengan jumlah semua True Positives dan False Negatives dalam kumpulan data. Dengan cara ini, ingatan memberikan indikasi prediksi positif yang terlewatkan, tidak seperti metrik presisi yang kami jelaskan di atas.

Dalam contoh kita, ini menjawab pertanyaan, "Dari semua orang yang benar-benar sakit, berapa banyak model yang didiagnosis sakit dengan benar?" Jadi, dengan mengikuti rumus, Anda akan mendapatkan 83,3 persen prediksi benar model dari semua hal positif. Semakin dekat ingatan ke 1, semakin baik model Anda karena tidak melewatkan hal positif yang sebenarnya.

Mengapa menggunakannya? Dalam model kami, Anda ingin menemukan semua orang sakit, jadi tidak apa-apa jika model tersebut mendiagnosis beberapa orang sehat sebagai orang sakit. Mereka mungkin akan dikirim untuk mengikuti

beberapa tes tambahan, yang menjengkelkan tetapi tidak kritis. Tetapi jauh lebih buruk jika model tersebut mendiagnosis beberapa orang sakit sebagai orang sehat dan memulangkan mereka tanpa pengobatan. Metrik penarikan lebih baik dalam hal ini daripada presisi karena meningkatkan jumlah orang dengan penyakit yang diprediksi dengan benar dan menerima perawatan mereka.

Penting untuk dipahami. Sama seperti presisi, metrik penarikan bukanlah solusi satu ukuran untuk semua. Jika kita mengambil contoh pendeteksi spam, prediksi yang benar akan lebih sedikit daripada presisi.

F1 Score

$$\text{F1 Score} = \text{Harmonic mean of Precision and Recall} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} = \frac{2\,TP}{2TP + FP + FN}$$

Skor F1 mencoba menemukan keseimbangan antara presisi dan daya ingat dengan menghitung rata-rata harmoniknya. Ini adalah ukuran akurasi tes di mana nilai tertinggi yang mungkin adalah 1. Hal ini menunjukkan presisi dan daya ingat yang sempurna.

Mengapa menggunakannya? Beberapa orang mungkin berpikir bahwa untuk menyeimbangkan presisi dan recall, kita cukup memilih hasil rata-rata. Meskipun ini bisa menjadi cara, ada peluang bagus untuk mendapatkan akurasi prediksi yang salah. Skor F1 adalah metrik yang lebih rumit yang memungkinkan Anda mendapatkan hasil yang mendekati kenyataan pada masalah klasifikasi yang tidak seimbang. Misalnya, dalam model medis kami, rata-ratanya adalah 69,5 persen sedangkan Skor F1 adalah 66,76 persen.

Penting untuk dipahami. Berbeda dengan Skor F1 yang tinggi, yang rendah tidak terlalu informatif: Ini hanya memberi tahu Anda tentang kinerja di ambang batas. Dengannya, Anda tidak akan mengerti apakah itu kesalahan penarikan atau kesalahan presisi.

Specificity

Spesifisitas adalah proporsi negatif aktual yang telah diidentifikasi dengan benar oleh model dari semua negatif. Ini menunjukkan Rasio Negatif Sejati yang dihitung sebagai semua Negatif Sejati dibagi dengan jumlah Negatif Sejati dan Positif Palsu dalam kumpulan data. Dalam matriks kami, kekhususan menjawab pertanyaan, "Dari semua orang yang benar-benar sehat, berapa banyak yang diprediksi oleh model dengan benar?" Spesifisitas pada dasarnya kebalikan dari ingatan.

Mengapa menggunakannya? Spesifisitas harus menjadi metrik pilihan Anda jika Anda harus mencakup semua hasil negatif sebenarnya dan Anda tidak dapat mentolerir hasil positif palsu apa pun. Mari kita ambil contoh lain yang akan sedikit dibesar-besarkan untuk etalase. Katakanlah, Anda membuat model deteksi penipuan di mana semua orang yang aktivitas kartu kreditnya ditandai sebagai penipuan (positif) akan langsung masuk penjara. Tentu saja, Anda tidak ingin memenjarakan orang yang tidak bersalah, yang berarti positif palsu di sini tidak dapat diterima.

Performance metrics for regression problems

Metrik yang digunakan untuk mengevaluasi kinerja model regresi. Tidak seperti klasifikasi, regresi memberikan keluaran dalam bentuk nilai numerik, bukan kelas, sehingga Anda tidak dapat menggunakan akurasi klasifikasi untuk evaluasi. Metrik untuk regresi melibatkan penghitungan skor kesalahan untuk meringkas keterampilan prediksi model. Seberapa jauh prediksi model berasal dari nilai data aktual atau data kebenaran dasar.

Mean Squared Error (MSE)

$$\text{MSE} = \frac{1}{N} \times \sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

N - number of data samples
y_i - actual data value
$\hat{y}_i$ - predicted data value

Mungkin metrik yang paling umum untuk masalah regresi, MSE atau Mean Squared Error dari prediksi bertujuan menemukan kesalahan kuadrat rata-rata antara nilai aktual dan nilai prediksi. Katakanlah kita membuat model regresi pembelajaran mendalam untuk memprediksi jumlah hari yang akan dihabiskan pasien tertentu di rumah sakit. Hari-hari yang benar-benar dihabiskan pasien di rumah sakit dilambangkan sebagai y_i, sedangkan jumlah yang diprediksi kami tunjukkan dengan $\hat{y}_i$. MSE untuk tiga pasien adalah sebagai berikut.

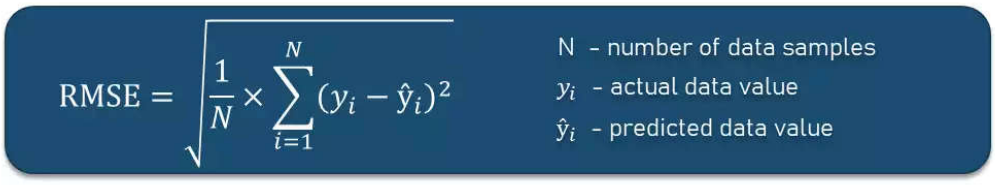
$$MSE = \frac{1}{3} \times \left((10-9)^2 + (11-5)^2 + (5-7)^2\right) = \frac{1+36+4}{3} = \frac{41}{3} = 13,(6)$$

Dengan asumsi kita memiliki data aktual bahwa pasien A menghabiskan 10 hari di rumah sakit, pasien B menghabiskan 11 hari, dan pasien C — 5 hari dengan nilai data prediksi masing-masing 9, 5, dan 7, MSE untuk model ini adalah 13. (6). Jumlah kesalahannya cukup besar mengingat tugas kami. Hasil seperti itu menandai bahwa model perlu diperbaiki. Semakin kecil nilai MSE, semakin tinggi akurasi model prediksi untuk mendeskripsikan data.

Mengapa menggunakannya? MSE bagus untuk mengoptimalkan model dan mudah dihitung. Angka MSE merupakan indikator bagi data scientist yang ingin meningkatkan daya prediksi model karena mereka dapat membandingkan MSE dari setiap iterasi, memilih persamaan yang menghasilkan kesalahan prediksi terkecil.

Penting untuk dipahami. MSE secara umum lebih sensitif terhadap outlier — poin data yang secara signifikan menyimpang dari distribusi reguler nilai dalam data. Karena kesalahan mengkuadratkan, model dapat dihukum lebih jika membuat prediksi yang sangat berbeda dari nilai sebenarnya yang sesuai. Jika model berurusan dengan tugas-tugas di mana outlier ekstrem harus diperhatikan, ada baiknya memilih MSE sebagai metrik karena pasti akan memperhatikannya. Katakanlah, Anda sedang membangun model peramalan permintaan untuk pengecer yang barangnya memiliki masa kedaluwarsa yang pendek. Pengecer seperti itu tidak dapat membiarkan diri mereka memiliki terlalu banyak atau terlalu sedikit barang di toko. MSE akan membantu Anda menemukan kesalahan yang lebih besar yang menghilangkan situasi defisit atau surplus.

Root Mean Squared Error (RMSE)



RMSE atau Root Mean Squared Error adalah perpanjangan dari MSE yang memungkinkan Anda menghilangkan kesalahan kuadrat dengan menghitung akar kuadrat dari hasil MSE.

Mengapa menggunakannya? Terkadang kami memiliki masalah dengan MSE karena unit kuadrat. Misalnya, jika Anda membuat model yang memprediksi harga tiket pesawat dalam dolar, skor kesalahan MSE akan memiliki satuan "dolar kuadrat". Hal ini dapat mempengaruhi efektivitas interpretasi kinerja model. RMSE, di sisi lain, akan memiliki unit "dolar" seperti variabel target dengan mengambil akar kuadrat dari MSE.

Penting untuk dipahami. Seperti halnya MSE, nilai RMSE yang sempurna adalah 0,0 atau mendekatinya, yang berarti bahwa semua prediksi sama persis dengan nilai yang diharapkan. Tapi itu jarang terjadi. Selain itu, itu juga tidak kuat untuk outlier.

Mean Absolute Error (MAE)

$$\text{MAE} = \frac{1}{N} \times \sum_{i=1}^{N} |y_i - \hat{y}_i|$$

N - number of data samples
y_i - actual data value
$\hat{y}_i$ - predicted data value

MAE atau Mean Absolute Error adalah rata-rata selisih antara nilai aktual dan nilai prediksi. Ini hanya memberikan ukuran seberapa jauh prediksi yang dibuat oleh model dari keluaran aktual.

Mengapa menggunakannya? Karena MAE tidak mengkuadratkan kesalahan, MAE tidak memberikan bobot yang lebih besar dan lebih kecil untuk kesalahan yang lebih besar vs kesalahan yang lebih kecil, tidak seperti MSE atau RMSE. Jika kesalahan yang lebih besar tidak memainkan peran penting dalam hasil, MAE bisa menjadi solusi yang bagus. Misalnya, untuk pengecer elektronik, memiliki defisit atau surplus 10 unit karena model meramalkan 10 unit lebih dan kurang yang akan terjual bukanlah masalah besar. Jadi, memperhatikan kesalahan besar dan kecil dalam model peramalan permintaan Anda bukanlah suatu keharusan.

$$MAE = \frac{1}{3} \times (|10 - 9| + |11 - 5| + |5 - 7|) = \frac{1 + 6 + 2}{3} = \frac{9}{3} = 3$$

Jika kita mengambil contoh yang disebutkan di atas dengan tiga pasien, MAE akan menjadi 3, yang merupakan hasil yang cukup bagus seperti RMSE 3,7 karena mendekati nol daripada angka MSE. Dan seperti yang kita ketahui, semakin kecil angka kesalahan, semakin baik kinerja model. Jadi, apakah ini metrik yang sempurna? TIDAK.

Penting untuk dipahami. Dengan MAE, kami hanya mendapatkan kesalahan absolut rata-rata di semua nilai. Karena absolut adalah fungsi matematika yang hanya membuat angka menjadi positif, perbedaan antara yang diharapkan dan yang

diprediksi — apakah itu positif atau negatif — selalu dipaksa menjadi positif saat menghitung MAE. Sama seperti MSE dan RMSE, hasilnya dari 0 hingga tak terhingga.