Kecerdasan Buatan/ Artificial Intelligence

Ketidakpastian (Uncertainty)

Latar Belakang

- Sistem kecerdasan buatan dikembangkan dengan memiliki pengetahuan yang terbatas tentang permasalahan yang ditanganinya, maka sistem tersebut <u>dapat memiliki kesalahan dalam memberikan solusi dengan menggunakan pendekatan logika.</u>

https://slideplayer.info/slide/13260818/

- <u>Sistem tidak akan pernah mempunyai pengetahuan atau fakta secara lengkap untuk permasalahan yang ditanganinya</u>, sehingga sistem harus bekerja dalam ketidakpastian dan kesamaran.

- Oleh karena itu, sistem harus <u>menggunakan teknik-teknik khusus yang dapat menangani ketidakpastian dan kesamaran</u> dalam menyelesaikan permasalahan-permasalahan yang ditanganinya.

Definisi

- *Uncertainty* : teori ketidakpastian pada kecerdasan buatan untuk menyelesaikan permasalahan yang ada dengan pendekatan logika (mengadopsi logika manusia) dimana sistem tidak dapat mengakses seluruh fakta yang ada.
- Contoh *(sample point)* : hasil dari percobaan
- Ruang Contoh *(sample space)*: kumpulan dari semua kemungkinan titik contoh.
- Kejadian *(event)* : subset dari ruang contoh.
- Kejadian sederhana *(simple event)*: hanya ada satu elemen kejadian.
- Kejadian gabungan *(compound event)*: terdapat lebih dari dari satu kejadian
- Penalaran Deduktif dan Induktif dilihat dari populasi dan contoh (*sample*)

Ketidakpastian

Contoh aplikasi yang menggunakan ketidakpastian dalam pengambilan kesimpulan logika yang sukses adalah sistem pakar MYCIN untuk diagnosa medis dan PROPECTOR untuk ekplorasi mineral.
Salah satu contoh permasalahan yang memiliki ketidakpastian adalah permasalahan diagnosa pasien terhadap suatu penyakit. Misalkan terdapat aturan dalam mendiagosa gejala-gejala dari suatu penyakit sebagai berikut :

IF badan_demam(pasien) AND
IF badan_ruam(pasien) AND
IF badan_pegal(pasien) THEN mengalami_tifus(pasien)

Berdasarkan aturan diatas, terlihat bahwa jika ada pasien yang mengalami ketiga jenis gejala tersebut maka akan dideteksi bahwa pasien menderita penyakit tifus. Akan tetapi pada dunia nyata ketika terdapat gejala-gejala tersebut memenuhi belum tentu penyakit yang diderita adalah tifus. Bisa jadi penyakit lain memiliki gejala yang sama sehingga bisa terjadi kesalahan diagnosa. Bagaimana jika derajat gejala yang dialami seorang pasien dengan pasien lainnya bisa jadi berbeda. Kemungkinan-kemungkinan kesalahan yang ada tersebut bisa jadi terjadi dan merupakan hal ketidakpastian dan kesamaran pengetahuan dalam permasalahan ini.

Konsep Probabilitas Klasik

Probabilitas merupakan cara yang digunakan dalam menghitung ketidakpastian dengan jalan kuantitas. Dalam probabilitas klasik disebut juga dengan *a priori probability* karena berhubungan dengan game atau sistem. Secara fundamental formula probabilitas klasik seperti pada persamaan 9.1 :

$$P=W/N \qquad (9.1)$$

dimana : W adalah jumlah kemenangan dalam permainan
N adalah jumlah kemungkinan kejadian yang dilakukan dalam percobaan

Contoh permasalahan adalah pelemparan dadu yang memiliki 6 sisi dan memiliki 6 kemungkinan. Maka peluang-peluang yang mungkin adalah

$$P(1) = 1/6$$
$$P(2) = 1/6$$
$$P(3) = 1/6$$
$$P(4) = 1/6$$
$$P(5) = 1/6$$
$$P(6) = 1/6$$

Uncertainty (Ketidak pastian)

- Contoh ada sebuah agent yang perlu ke bandara karena akan terbang ke LN. Mis. action A_t = pergi ke bandara t menit sebelum pesawat terbang. Apakah A_t berhasil sampai dengan waktu cukup?
- Ada banyak masalah:
  - **Tidak tahu** keadaan jalan, kemacetan, dll. (**partially observable**).
  - **Kebenaran** informasi tidak bisa dijamin-“laporan pandangan mata” (**noisy sensor**).
  - **Ketidakpastian** dalam tindakan, mis. ban kempes (**nondeterministic**).
  - **Kalaupun** semua hal di atas bisa dinyatakan, reasoning akan **luar biasa repot**.

Uncertainty

- Sebuah pendekatan yang murni secara logika
  - beresiko menyimpulkan dengan **salah**, mis: "A_{60} berhasil dengan waktu cukup", atau
  - kesimpulan terlalu **lemah**, mis: "A_{60} berhasil dengan waktu cukup asal nggak ada kecelakaan di tol, dan nggak hujan, dan ban nggak kempes, ..."
  - kesimpulan **tidak rational**, mis: kesimpulannya A_{1440}, tetapi terpaksa menunggu semalam di bandara (**utility theory**).
- Masalah ini bisa diselesaikan dengan **probabilistic reasoning**
  - Berdasarkan info yang ada, A_{60} akan berhasil dengan probabilitas 0.04".

Uncertainty

- Kalimat “A_{60} akan berhasil dengan probabilitas 0.04” disebut **probabilistic assertion**.
- Sebuah probabilistic assertion **merangkum** efek ketidakpastian (info tak lengkap, tak bisa dipegang, action nondeterministic, dst.) dan menyatakannya sbg. sebuah bilangan.
- Bentuk/syntax probabilistic assertion:
  - “Kalimat X bernilai true dengan probabilitas N, $0 \le N \le 1$”.
  - Pernyataan tentang knowledge atau belief state dari agent, BUKAN berarti pernyataan tentang sifat probabilistik di dunia/environment.
- Nilai probabilitas sebuah proposition bisa berubah dengan informasi baru (“evidence”):

  P(A60| tidak ada laporan kecelakaan) = 0.06

  P(A60| tidak ada laporan kecelakaan, jam 4 pagi) = 0.15

Probability

- Probabilistic reasoning:
  - Percept masuk (tambahan evidence), update nilai probabilitas.
  - **Prior/unconditional** probability: nilai sebelum evidence.
  - **Posterior/conditional** probability: nilai sesudah evidence.
  - "ASK" secara probabilistik: hitung & kembalikan posterior probability terhadap α berdasarkan evidence dari percept .
- Contoh: melempar dadu.

  α = "Nilai lemparan < 4".

  Sebelum melihat dadu:

$$P(\alpha) = \frac{1}{6} + \frac{1}{6} + \frac{1}{6} = \frac{1}{2}$$

  Setelah melihat dadu:

$$P(\alpha) = 0 \quad \text{atau} \quad 1$$

Probability

- Mengambil keputusan dlm ketidakpastian, andaikan agent mempercayai nilai-nilai sbb.:

  $P(A_{60} \mid \ldots) = 0.04$
  $P(A_{120} \mid \ldots) = 0.7$
  $P(A_{150} \mid \ldots) = 0.9$
  $P(A_{1440} \mid \ldots) = 0.999$
- Tindakan mana yang dipilih?
  - Tergantung prioritas, mis. ketinggalan pesawat vs. begadang di lobby bandara, dst.
  - **Utility theory** digunakan untuk menilai semua tindakan (mirip evaluation function).
  - **Decision theory** = utility theory + probability theory
- Sama halnya dengan logic, pendefinisian "bahasa formal" untuk menyatakan **kalimat probabilistic** harus ada : **Syntax** (bagaimana bentuk kalimatnya), **Semantics** (apakah arti kalimatnya), **Teknik & metode** melakukan reasoning.

Semantics & Syntax

- Semantics untuk kalimat probabilistic (Bayangkan semua kemungkinan dunia *possible worlds* yang terjadi).
  - Dalam logic , salah satunya adalah dunia "nyata".
  - Dalam probability , kita tidak tahu pasti yang mana, tetapi satu dunia bisa lebih mungkin dari dunia yang lain.
- Himpunan semua *possible worlds* disebut *sample space* (Ω). Masing-masing dunia alternatif disebut *sample point*, atau *atomic event* (ω).
- Contoh :
  - Jika dunia hanya berisi sebuah lemparan dadu Ω berisi 6 kemungkinan, $\omega_1 . . . \omega_6$.

Semantics & Syntax

- Sebuah **probability model** adalah *sample space* di mana tiap *sample point* diberi nilai $P(\omega)$ sehingga:
  - Setiap nilai antara 0 s/d 1.
  - Jumlah nilai seluruh sample space = 1.
- Contohnya, untuk "dunia" dengan 1 lemparan dadu :
  - $P(\omega_1) = P(\omega_2) = P(\omega_3) = P(\omega_4) = P(\omega_5) = P(\omega_6) = 1/6$
- Biasanya, dunia memiliki > 1 faktor yang tidak pasti. *Sample space* dan *probability* model menjadi multidimensi, menyatakan semua kemungkinan **kombinasi**nya.
- Contohnya, untuk "dunia" dengan 2 lemparan dadu :
  - $P(\omega_{1,1}) = P(\omega_{1,2}) = \ldots = P(\omega_{6,5}) = P(\omega_{6,6}) = 1/36$

Semantics & Syntax

- Di dalam dunia multidimensi, terkadang kita hanya tertarik dengan 1 dimensi (mis. lemparan dadu pertama)
- Sebuah **event** A adalah sembarang subset dari Ω.
- Probability A adalah jumlah probability sample point anggotanya.

$$\mathrm{P}(A) = \sum_{\omega \in A} \mathrm{P}(\omega)$$

- Contohnya, $\mathrm{P}(\text{dadu}_1 = 5) = 6x\dfrac{1}{36} = \dfrac{1}{6}$
- Event juga bisa menyatakan probability dari deskripsi parsial.
- Contoh: untuk satu lemparan dadu, $\mathrm{P}(\text{dadu} \geq 4) = 3x\dfrac{1}{6} = \dfrac{1}{2}$

Semantics & Syntax

- Nilai probabilitas diberikan kepada sebuah proposition.
- Agar proposition dapat diperinci, kita definisikan **random variable**, yang merepresentasikan suatu “aspek” dari sebuah dunia.
- Contohnya, dalam kasus melempar dadu :
  - Bisa ada **random variable** bernama *hasil_lemparan*.
- Secara formal, random variable adalah fungsi yang memetakan setiap sample point ke dalam ranah, mis. boolean, integer, real.
- Contohnya :
  - *hasil_lemparan* adalah fungsi yang memetakan ω_1 ke integer 1, ω_2 keinteger 2, ω_3 ke integer 3, dst.
- Sekarang semua proposition berupa pernyataan tentang satu atau lebih *random variable*.

Semantics & Syntax

- Domain sebuah random variable bisa:
  - **boolean**, mis: Ganjil(ω_1) = true
  - **diskrit**, mis: Weather (ω) $\in$ (sunny, rain, cloudy, snow)
  - **takhingga**, mis: integer (**diskrit**), real (**kontinyu**)
- Sebuah probability model P menghasilkan probability distribution untuk sembarang random variable : $P(X = x_i) = \sum_{\omega : X(\omega) = x_i} P(\omega)$
- Contoh dgn. dadu:

$$P(\text{Ganjil} = \text{true}) = \frac{1}{6} + \frac{1}{6} + \frac{1}{6} = \frac{1}{2}$$

- Contoh dgn cuaca:

P(Weather = sunny) = 0.7

P(Weather = rain) = 0.2

P(Weather = cloudy) = 0.08

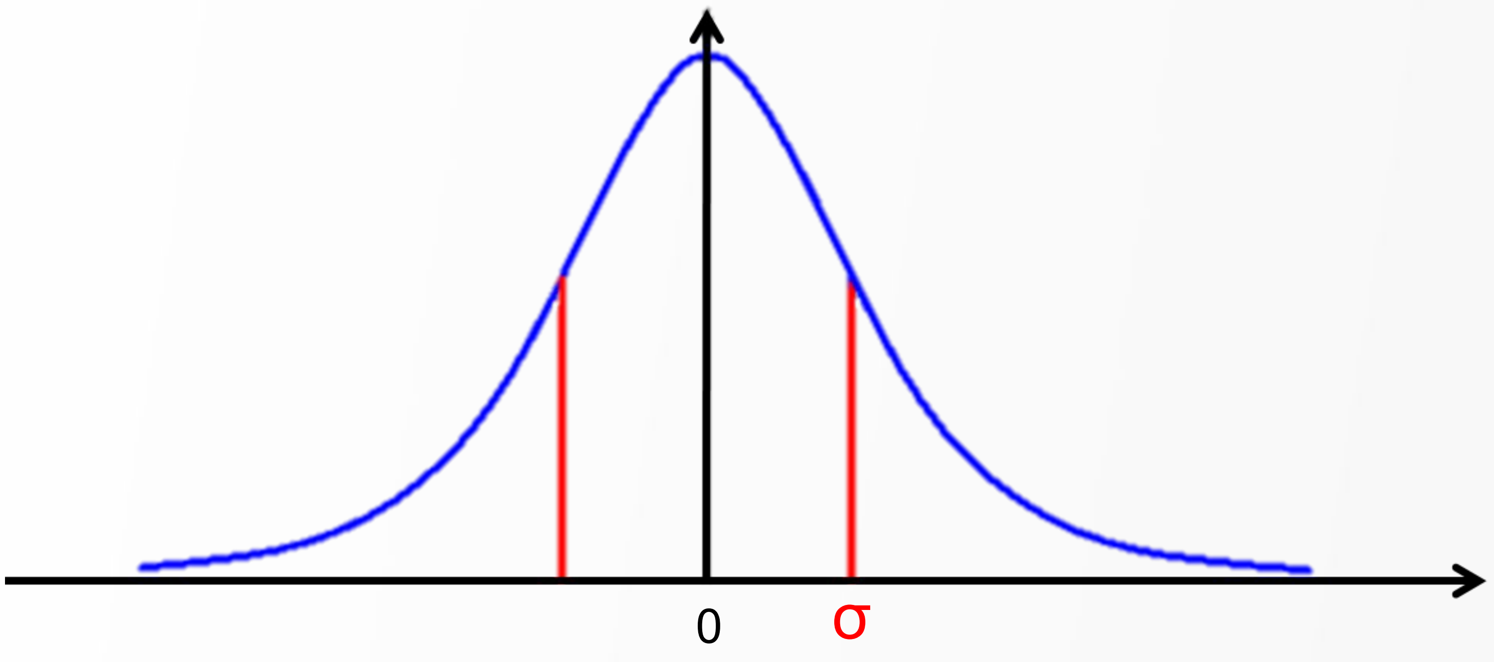
P(Weather = snow) = 0.02

atau disingkat P(Weather) = (0.7, 0.2, 0.08, 0.02)

Semantics & Syntax

- Contoh distribution untuk variable real & kontinyu yang banyak ditemui dalam dunia nyata adalah fungsi **Gaussian**:

$$P(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$



Semantics & Syntax

- Dalam AI, seringkali sample point didefinisikan oleh nilai sekumpulan random variable.
- Jadi, sample space berisi semua kemungkinan kombinasi nilai semua variable.
- **Joint probability distribution** dari sehimpunan random variable memberikan nilai probability untuk setiap sample point tersebut.
- Andaikan kita tertarik mengamati hubungan **cuaca** dengan **sakit gigi**, berikut **contoh joint probability distribution**-nya:

Weather	=	sunny	rain	cloudy	snow
Toothache	= true	0.144	0.02	0.016	0.02
Toothache	= false	0.576	0.08	0.064	0.08

Semantics & Syntax

- Sebuah **proposition** adalah pernyataan tentang nilai dari satu atau lebih random variable.
- Bayangkan proposition sebagai event (himpunan sample point) di mana ia bernilai true.
- Untuk 2 buah random variable boolean A dan B :
  - Event a = himpunan sample point di mana $A(\omega) = true$
  - Event $\neg a$ = himpunan sample point di mana $A(\omega) = false$
  - Event $a \land b$ = himpunan sample point di mana $A(\omega)$ dan $B(\omega) = true$
  - Event $a \lor b$ = himpunan sample point di mana $A(\omega)$ atau $B(\omega) = true$

Semantics & Syntax

- Contoh yang memilukan, bayangkan masalah dokter gigi, di mana ada 3 random variable:
  - **Cavity**: apakah pasien memiliki gigi berlubang atau tidak?
  - **Toothache**: apakah pasien merasa sakit gigi atau tidak?
  - **Catch**: apakah pisau dokter nyangkut di gigi pasien atau tidak?
- Joint probability distribution sbb.:

	toothache		**¬ toothache**	
	catch	¬ catch	catch	¬ catch
cavity	0.108	0.012	0.072	0.008
¬ cavity	0.016	0.64	0.144	0.576

- Prior vs. posterior probability
  - **Prior**: Nilai probability tanpa informasi spesifik (unconditional).
  - **Posterior**: Nilai probability jika sesuatu informasi spesifik diketahui (conditional).

Semantics & Syntax

- Prior vs. posterior probability
  - **Prior**: Nilai probability tanpa informasi spesifik (unconditional).

    Contoh: P(cavity), P(toothache ∧ catch), dst.
  - **Posterior**: Nilai probability jika sesuatu informasi spesifik diketahui (conditional).

    Contoh: P(cavity | toothache).

    ("Peluang jika seseorang sakit gigi maka giginya berlubang")
- Definisi *conditional probability*:

$$P(a \mid b) = \frac{P(a \wedge b)}{P(b)}$$

untuk P(b) ≠ 0

- Perumusan alternatif (**Product rule**):

$$P(a \wedge b) = P(a \mid b)P(b) = P(b \mid a)P(a)$$

Inference

- Dengan *joint probability distribution*, probability sembarang proposition bisa dihitung sbg. jumlah probability *sample point* di mana ia bernilai true. Contoh :

	toothache		¬ toothache	
	catch	¬ catch	catch	¬ catch
cavity	0.108	0.012	0.072	0.008
¬ cavity	0.016	0.064	0.144	0.576

- **P(toothache) = 0.108 + 0.012 + 0.016 + 0.064 = 0.2**

Inference

- Dengan *joint probability distribution*, probability sembarang proposition bisa dihitung sbg. jumlah probability *sample point* di mana ia bernilai true. Contoh :

	toothache		**¬ toothache**	
	catch	¬ catch	catch	¬ catch
cavity	0.108	0.012	0.072	0.008
¬ cavity	0.016	0.064	0.144	0.576

- **P(cavity ∨ toothache) = 0.108 + 0.012 + 0.072 + 0.008 + 0.016 + 0.064**
  = 0.28

Inference

- Dengan *joint probability distribution*, probability sembarang proposition bisa dihitung sbg. jumlah probability *sample point* di mana ia bernilai true. Contoh :

	toothache		¬ toothache	
	catch	¬ catch	catch	¬ catch
cavity	0.108	0.012	0.072	0.008
¬ cavity	0.016	0.064	0.144	0.576

- **Bisa juga menghitung conditional probability:**

$$P(\neg \text{cavity} \mid \text{toothache}) = \frac{P(\neg \text{cavity} \wedge \text{toothache})}{P(\text{toothache})}$$

$$= \frac{0.016 + 0.064}{0.108 + 0.012 + 0.016 + 0.064} = 0.4$$

Naïve Bayes

- Algoritma Naive Bayes Merupakan pengklasifikasian statistik yang dapat digunakan untuk memprediksi probabilitas keanggotaan suatu class. Bayesian Classification didasarkan pada teorema Bayes yang memiliki kemampuan klasifikasi serupa decision tree dan neural network.

- Teorema Bayes akhirnya dikembangkan dengan berbagai ilmu termasuk untuk penyelesaian masalah sistem pakar dengan menetukan nilai probabilitas dari hipotesa pakar dan nilai evidence yang didapatkan fakta yang didapat dari objek yang diagnosa.

- Teorema ini menerangkan hubungan antara probabilitas terjadinya peristiwa A dengan syarat peristiwa B telah terjadi dan probabilitas terjadinya peristiwa B dengan syarat peristiwa A telah terjadi. Teorema ini didasarkan pada prinsip bahwa tambahan informasi dapat memperbaiki probabilitas.

Keuntungan Naive Bayesian

- Menangani kuantitatif dan data diskrit
- Kokoh untuk titik noise yang diisolasi, misalkan titik yang dirata – ratakan ketika mengestimasi peluang bersyarat data.
- Hanya memerlukan sejumlah kecil data pelatihan untuk mengestimasi parameter (rata – rata dan variansi dari variabel) yang dibutuhkan untuk klasifikasi.
- Menangani nilai yang hilang dengan mengabaikan instansi selama perhitungan estimasi peluang
- Cepat dan efisiensi ruang
- Kokoh terhadap atribut yang tidak relevan

Kekurangan Naive Bayesian

- Tidak berlaku jika probabilitas kondisionalnya adalah nol, apabila nol maka probabilitas prediksi akan bernilai nol juga
- Mengasumsikan variabel bebas

Aturan Bayes

- Mendefinisikan fitur untuk setiap objek dengan : $P(x \mid \omega_1)$ & $P(x \mid \omega_2)$: (Probabilitas kodisional objek (x) terhadap kelas (ω_j) / Likelihood).

$$P(\omega_j \mid x) = \frac{P(x \mid \omega_j) P(\omega_j)}{P(x)}$$

$$\text{Posterior} = \frac{\text{Likelihood} * \text{Prior}}{\text{Evidence}}$$

$$P(x) = \sum_{j=1}^{2} P(x \mid \omega_j) P(\omega_j)$$

Case Study

- Diketahui probabilitas ikan Sea Bass dan Salmon masing-masing adalah $P(\omega_2) = 2/3$ dan $P(\omega_1) = 1/3$. Tentukan hasil keputusan klasifikasi jika input x = 13, dimana probabilitas likelihood-nya masing-masing $P(x \mid \omega_1) = 0.28$ dan $P(x \mid \omega_2) = 0.17$!
- Penyelesaian :

$$P(\omega_1|x) = \frac{P(x|\omega_1)P(\omega_1)}{P(x)}$$

$$= \frac{(0,28)(1/3)}{P(x|\omega_1)P(\omega_1) + P(x|\omega_2)P(\omega_2)}$$

$$= \frac{(0,28)(1/3)}{((0,28)(1/3)) + ((0,17)(2/3))}$$

$$= \frac{0,0924}{0,0924 + 0,1139} = \frac{0,0924}{0,2063} = 0,4479$$

$$P(\omega_2|x) = \frac{P(x|\omega_2)P(\omega_2)}{P(x)}$$

$$= \frac{(0,17)(2/3)}{P(x|\omega_1)P(\omega_1) + P(x|\omega_2)P(\omega_2)}$$

$$= \frac{(0,17)(2/3)}{((0,28)(1/3)) + ((0,17)(2/3))}$$

$$= \frac{0,1139}{0,0924 + 0,1139} = \frac{0,1139}{0,2063} = 0,5521$$

$$Kelas(x) = \underset{x \in [\omega_1, \omega_2]}{\arg\max} P(\omega_j|x)$$

Karena $\omega_2 > \omega_1$, maka data uji x termasuk kelas ω_2 (ikan Sea Bass).

Contoh Kasus NB

- Perhitungan metode naive bayes untuk sistem pakar penentuan kerusakan pada laptop, pada tahap awal kita harus mempunyai data kerusakan dan gejala laptop terlebih dahulu. Kerusakan laptop yang dibahas disini adalah tentang kerusakan dibagian hardware didalam laptop. Berikut adalah data yang disajikan.

- **Data kerusakan laptop :**

  K1 = IC Charger Rusak
  K2 = IC Power Rusak
  K3 = Resistor Rusak
  K4 = Kapasitor Rusak
  K5 = Mofset Rusak
  K6 = Embeded Controller Rusak

- **Data gejala yang timbul :**

  G1 = Indikator pengisian baterai nyala tapi laptop tidak bisa dinyalakan.
  G2 = Indikaor pengisian baterai mati, laptop tidak bisa dinyalakan.
  G3 = Indikaor pengisian baterai nyala, bisa dinyalakan tapi tidak tampil pada layar.
  G4 = Input seperti USB tidak berfungsi

  Keterangan :
  K = Kerusakan
  G = Gejala

- Selanjutnya dari data gejala dan kerusakan diatas kita menentukan tabel keputusan antara kerusakan dan gejala yang timbul. Tabel keputusan berfungsi untuk menentukan laptop tersebut mengalami kerusakan apa, berdasarkan gejala yang timbul. Berikut adalah tabel keputusan yang sudah ditentukan.

Gejala	Kerusakan					
	K1	K2	K3	K4	K5	K6
G1	1	0	1	0	0	0
G2	0	1	0	0	0	1
G3	0	0	1	0	0	0
G4	0	1	0	1	1	1

Keterangan :

1 = Gejala muncul
0 = Tidak ada gejala yang muncul

- **Contoh Kasus :**

  Misalnya gejala yang tampak pada laptop ada dua gejala yaitu :

  G1 : Indikator pengisian baterai nyala tapi laptop tidak bisa dinyalakan, dan
  G3 : Indikator pengisian baterai nyala, bisa dinyalakan tapi tidak tampil pada layar.

  Berdasarkan gejala yang muncul tersebut maka langkah perhitungannya adalah sebagai berikut :

- **Langkah 1 : menentukan penyakit yang muncul berdasarkan tabel keputusan**

  Berdasarkan gejala yang muncul G1 dan G3 , maka bisa dilihat dari tabel keputusan indikasi kerusakan yang akan di prediksi yaitu K1 dan K3. karena pada K1 terdapat G1 dan G3 yang bernilai 1 dan pada K3 terdapat G3 yang bernilai 1.

  Maka untuk tahap selanjutnya yang di hitung menggunakan algoritma naive bayes adalah menghitung nilai probabilitas gejala dari K1 dan K3.

$$\text{Rumus Probabilitas K1} = \frac{\text{jumlah kemungkinan kerusakan yang muncul}}{\text{jumlah semua kerusakan}} = \frac{1}{6} = 0.16$$

Keterangan :
Angka 1 di dapatkan dari prediksi minimal kerusakan yang muncul
Angka 6 di dapatkan dari jumlah semua kerusakan yang ada pada tabel keputusan

- Rumus menghitung probabilitas gejala yang muncul

  G1 : Indikator pengisian baterai nyala tapi laptop tidak bisa dinyalakan.

$$G1 = \frac{\text{jumlah kemungkinan}}{\text{jumlah kemungkinan kerusakan akibat gejala}} = \frac{1}{2} = 0.5$$

G3 : Indikator pengisian baterai nyala, bisa dinyalakan tapi tidak tampil pada layar.

$$G3 = \frac{\text{jumlah kemungkinan}}{\text{jumlah kemungkinan kerusakan akibat gejala}} = \frac{0}{2} = 0$$

- Keterangan :
  jumlah kemungkinan *= jumlah gejala G1/G3 yang muncul pada K1 di tabel keputusan*

  jumlah kemungkinan kerusakan akibat gejala *= kerusakan yang muncul yang di akibatkan gejala dalam perhitungan kali ini didapatkan 2 kerusakan yang muncul yaitu K1 dan K3*

  Perhitungan Probabilitas K3 (Resistor Rusak)

  Rumus menghitung probailitas nilai K3

$$\text{Rumus Probabilitas K3} = \frac{\text{jumlah kemungkinan kerusakan yang muncul}}{\text{jumlah semua kerusakan}} = \frac{1}{6} = 0.16$$

Keterangan :
Angka 1 di dapatkan dari prediksi minimal kerusakan yang muncul
Angka 6 di dapatkan dari jumlah semua kerusakan yang ada pada tabel keputusan

Rumus menghitung probabilitas gejala yang muncul

G1 : Indikator pengisian baterai nyala tapi laptop tidak bisa dinyalakan.

$$G1 = \frac{\text{jumlah kemungkinan}}{\text{jumlah kemungkinan kerusakan akibat gejala}} = \frac{1}{2} = 0.5$$

G3 : Indikator pengisian baterai nyala, bisa dinyalakan tapi tidak tampil pada layar.

$$G3 = \frac{\text{jumlah kemungkinan}}{\text{jumlah kemungkinan kerusakan akibat gejala}} = \frac{1}{2} = 0.5$$

Keterangan :

jumlah kemungkinan *= jumlah gejala G1/G3 yang muncul pada K3 di tabel keputusan*

jumlah kemungkinan kerusakan akibat gejala *= kerusakan yang muncul yang di akibatkan gejala dalam perhitungan kali ini didapatkan 2 kerusakan yang muncul yaitu K1 dan K3*

- **Langkah 3 : Menghitung nilai bayes berdasarkan probabilitas kerusakan dan gejala yang timbul**
  Dari nilai probabilitas diatas selanjutnya tahap perhitungan nilai bayes dengan rumus sebagai berikut

Menghitung Nilai Bayes

$$K(K1 \mid G1) = \frac{[K\,(G1 \mid K1) * K\,(K1)]}{[K\,(G1 \mid K1) * K\,(K1) + K\,(G1 \mid K3) * K\,(K3)}$$

$$= \frac{0{,}5 \times 0{,}16}{0{,}5 \times 0{,}16 + 0{,}5 \times 0{,}16} = \frac{0.08}{0{,}16} = 0.5$$

$$K(K1 \mid G3) = \frac{[K\,(G3 \mid K1) * K\,(K1)]}{[K\,(G3 \mid K1) * K\,(K1) + K\,(G3 \mid K3) * K\,(K3)}$$

$$= \frac{0 \times 0{,}16}{0 \times 0{,}16 + 0{,}5 \times 0{,}16} = \frac{0}{0{,}08} = 0$$

Total nilai bayes dari K1 yaitu :

Total K1 = K(K1 | G1) + K(K1 | G3)
Total K1 = 0.5 + 0 = 0.5

Menghitung Nilai Bayes K3

$$K(K3 \mid G1) = \frac{[K(G1 \mid K3) * K(K3)]}{[K(G1 \mid K1) * K(K1) + K(G1 \mid K3) * K(K3)}$$

$$= \frac{0{,}5 \times 0{,}16}{0{,}5 \times 0{,}16 + 0{,}5 \times 0{,}16} = \frac{0.08}{0{,}16} = 0.5$$

$$K(K3 \mid G3) = \frac{[K(G3 \mid K3) * K(K3)]}{[K(G3 \mid K1) * K(K1) + K(G3 \mid K3) * K(K3)}$$

$$= \frac{0{,}5 \times 0{,}16}{0{,}5 \times 0{,}16 + 0{,}5 \times 0{,}16} = \frac{0{,}08}{0{,}16} = 0{,}5$$

Total nilai bayes dari K3 yaitu :

Total K3 = K(K3 | G1) + K(K3 | G3)
Total K3 = 0.5 + 0.5 = 1

Menjumlahkan hasil nilai bayes dari K1 dan K3

Hasil Total = Total Bayes K1 + Total Bayes K3
= 0.5 + 1
= 1.5

Langkah 4 : Menghitung presentase nilai prediksi kerusakan

Dari perhitungan hasil total didapatkan nilai 1.5 . Angka tersebut nantinya di gunakan sebagai pembagi masing-masing nilai bayes dari K1 dan K3 untuk di ketehaui presentasenya. Berikut ini adalah hasil yang didapatkan dari perhitungan tersebut.

1. Kerusakan Pada IC charger (K1)

$$= \frac{Total\ Bayes\ K1}{Total\ Hasil} \times 100\%$$

$$= \frac{0.5}{1.5} \times 100\% = \mathbf{33{,}3\ \%}$$

2. Kerusakan Pada Resistor (K3)

$$= \frac{Total\ Bayes\ K3}{Total\ Hasil} \times 100\%$$

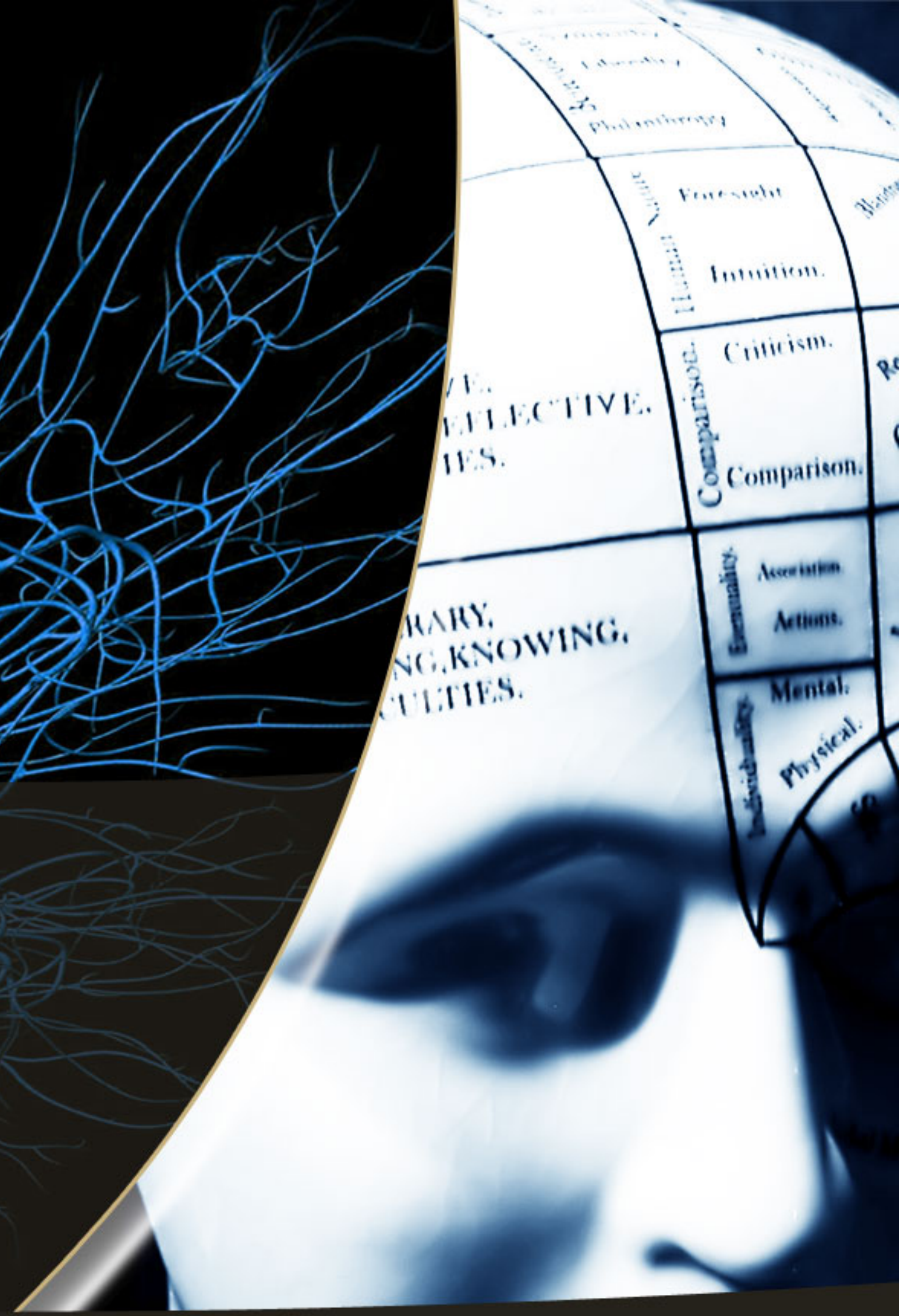
$$= \frac{1}{1.5} \times 100\% = \mathbf{66{,}6\ \%}$$

- Dari hasil presentase diatas maka didapatkan nilai presentase tertinggi adalah hasil kerusakan yang didapatkan. Dengan demikian jika ada laptop yang mengalami gejala kerusakan G1 (Indikator pengisian baterai nyala tapi laptop tidak bisa dinyalakan.) dan G3

  (Indikator pengisian baterai nyala, bisa dinyalakan tapi tidak tampil pada layar.). Maka laptop tersebut mengalami kerusakan K3

  (Kerusakan Pada Resistor).

Selesai


Foresight
Intuition.
Criticism.
Comparison.
Mental.
Physical.