

Bahan:

- Slide 6 Adversarial Network: Alpha-Beta Pruning (perhitungan) dan konsep Monte Carlo Tree Search
- Slide 7 Logical Agent: konsep entailment, konsep inferensi logika, KB untuk the wumpus worlds, cara kerja model checking dan theorem proving.
- Slide 8 Reasoning & Uncertainty: probabilitas, bayes dan naive bayes model
- Slide 9 Machine Learning: s.d hal 270 (Nonlinear Data). Konsep umum dan tipe machine learning, konsep dan perhitungan LDT dan Confusion matrix, konsep regresi linear
- dan topik tertentu yang tidak ada di slide yang saya minta kalian pelajari sendiri

Materi 6. Adversarial Search.....	2
Alpha-Beta Pruning.....	2
Monte Carlo Tree Search.....	2
Materi 7. Logical Agent.....	3
Logical Entailment.....	3
Contoh 1.....	3
Contoh 2.....	3
Logical Inference.....	3
The Wumpus World KB.....	4
Logical Operator.....	4
Wumpus World.....	4
Model Checking.....	4
Theorem Proving.....	5
Contoh.....	5
Materi 8. Reasoning with Uncertainty.....	6
Probability.....	6
Product Rules.....	6
Bayes' Rule.....	6
Naive Bayes Models.....	6
Materi 9. Machine Learning.....	7
Types of Machine Learning.....	7
LDT (Learning Decision Tree).....	8
Correctness.....	8
Gini Impurity Index.....	9
Entropy.....	9
Confusion Matrix.....	10
Linear Regression.....	11
Metrik Evaluasi.....	11
Soal Kuis.....	12
Nomor 1.....	12
Nomor 2.....	13
Nomor 3.....	14
Nomor 4.....	15
Nomor 5.....	16
Nomor 6.....	17

Materi 6. Adversarial Search

Alpha-Beta Pruning

- Pruning = memotong bagian tree yang tidak berpengaruh pada outcome
- Alpha = value of the best choice (highest value) so far untuk MAX ("at least")
- Beta = value of the best choice (lowest value) so far untuk MIN ("at most")
- Alpha beta pruning in artificial intelligence with example.
- Alpha Beta Pruning

Monte Carlo Tree Search

- Tidak menggunakan fungsi evaluasi heuristic
- Value of state = average utility atas sejumlah simulasi (playout atau rollout) permainan lengkap yang dimulai dari state tersebut
- Pure MCTS = do N simulations dimulai dari current state dan track moves mana yang memiliki highest win percentage
- Pure MCTS is not enough, we need policy
 - exploration of states yang memiliki sedikit playouts
 - Exploitation of states yang telah berhasil dalam playout sebelumnya
- MCTS maintains search tree dan mengembangkannya di setiap iterasi with 4 steps = selection, expansion, simulation, back-propagation

Materi 7. Logical Agent

Logical Entailment

- Sentence yang mengikuti secara logis dari sentence lainnya

$$\mathcal{KB} \models \alpha$$

- " α secara logis mengikuti dari KB," berarti jika semua pernyataan KB benar, maka α juga harus benar

Contoh 1

\mathcal{KB} : nothing in [1,1] and a breeze in [2,1] (from agent's percept)

\mathcal{KB} is false where [1,2] contains a pit.
There is no breeze in [1,1].

α_1 = "There is no pit in [1,2]"

in every model in which \mathcal{KB} is true, so does α_1

hence, $\mathcal{KB} \models \alpha_1$



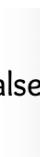
- Semua model (atau skenario) di mana persepsi agen (KB) benar, kesimpulan α_1 juga benar
- Jika KB benar, maka tidak ada lubang di [1, 2]

Contoh 2

α_2 = "There is no pit in [2,2]"

in some model in which \mathcal{KB} is true, α_2 is false

hence, \mathcal{KB} does not entail α_2



Logical Inference

- Memperoleh kesimpulan baru berdasarkan fakta yang sudah ada dalam basis pengetahuan menggunakan aturan-aturan inferensi
- Dilakukan dengan menggunakan
 - Model checking = entailment melalui semantik, enumerate all models and tunjukkan kalimat α berlaku di all models, $KB \models \alpha$
 - Theorem proving = entailment melalui syntax, apply rules of inference to KB to membuktikan pernyataan tanpa enumerating dan check all models
- Inference algorithm memiliki 2 properties
 - Sound = hanya menghasilkan entailed sentences
 - Complete = derives all entailed sentence

The Wumpus World KB

Logical Operator

- \neg = NOT
 - \wedge = AND
 - \vee = OR
 - \Rightarrow = implikasi (if ... then. Menghasilkan salah hanya jika pernyataan pertama benar dan pernyataan kedua salah)
 - \Leftrightarrow biimplikasi (Menghasilkan benar jika kedua pernyataan memiliki nilai kebenaran yang sama)

Wumpus World

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK			
1,1 A OK	2,1 OK	3,1	4,1

Symbols for each $[x, y]$:

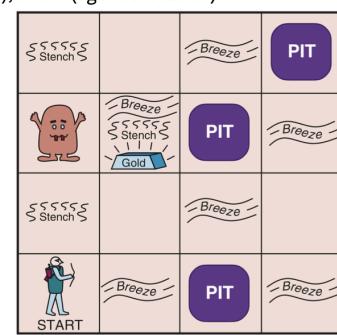
- $P_{i,j}$ is true if there is a pit in $[i, j]$
 - $B_{i,j}$ is true if there is a breeze in $[i, j]$
 - and the same for W (wumpus), S (stench), and L (agent's location)

\mathcal{KB} for the reduced Wumpus world:

- $R_1 : \neg P_{1,1}$
 - $R_2 : B_{1,1} \iff P_{1,2} \vee P_{2,1}$
 - $R_3 : B_{2,1} \iff P_{1,1} \vee P_{2,2} \vee P_{3,1}$
 - $R_4 : \neg B_{1,1}$
 - $R_5 : B_{2,1}$

Questions (based on above KB):

- $\mathcal{KB} \vDash: P_{1,2}$
 - $\mathcal{KB} \vDash: P_{2,2}$



Model Checking

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	R_1	R_2	R_3	R_4	R_5	KB
false	true	true	true	true	false	false						
false	false	false	false	false	false	true	true	true	false	true	false	false
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
false	true	false	false	false	false	false	true	true	false	true	true	false
false	true	false	false	false	false	true	true	true	true	true	true	<u>true</u>
false	true	false	false	false	true	false	true	true	true	true	true	<u>true</u>
false	true	false	false	false	true	true	true	true	true	true	true	<u>true</u>
false	true	false	false	true	false	false	true	false	false	true	true	false
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
true	false	true	true	false	true	false						

Theorem Proving

- Method checking is inefficient, tables might have exponential number of models
- Apply aturan inferensi directly ke KB tanpa berkonsultasi dengan model
 - Modus Ponens
 - And-Elimination
 - Standard logical equivalences

Contoh



How to prove there is no pit in [1,2]:

1. Apply biconditional elimination to R_2 to obtain

$$R_6 : (B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}).$$

2. Apply And-Elimination to R_6 to obtain

$$R_7 : ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}).$$

3. Logical equivalence for contrapositives gives

$$R_8 : (\neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})).$$

4. Apply Modus Ponens with R_8 and the percept R_4 (i.e., $\neg B_{1,1}$), to obtain

$$R_9 : \neg(P_{1,2} \vee P_{2,1}).$$

5. Apply De Morgan's rule, giving the conclusion

$$R_{10} : \neg P_{1,2} \wedge \neg P_{2,1}.$$

That is, neither [1,2] nor [2,1] contains a pit.

Materi 8. Reasoning with Uncertainty

Probability

- Ω = ruang sample, himpunan semua kemungkinan hasil (36 of two dices)
- ω = a particular possible world (one of 36 of two dices)
- $P(\omega)$ = probability model berasosiasi with each possible world. $0 \leq P(\omega) \leq 1$

Product Rules

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

$$P(a | b) = \frac{P(a \wedge b)}{P(b)}, \text{ holds whenever } P(b) > 0$$

Bayes' Rule

$$P(b | a) = \frac{P(a | b)P(b)}{P(a)}$$

$$P(\text{cause} | \text{effect}) = \frac{P(\text{effect} | \text{cause})P(\text{cause})}{P(\text{effect})}$$

- $P(\text{cause} | \text{effect})$ = diagnostic direction. Penalaran dari suatu efek yang diamati untuk menentukan penyebab
- $P(\text{effect} | \text{cause})$ = causal direction. Hubungan sebab-akibat

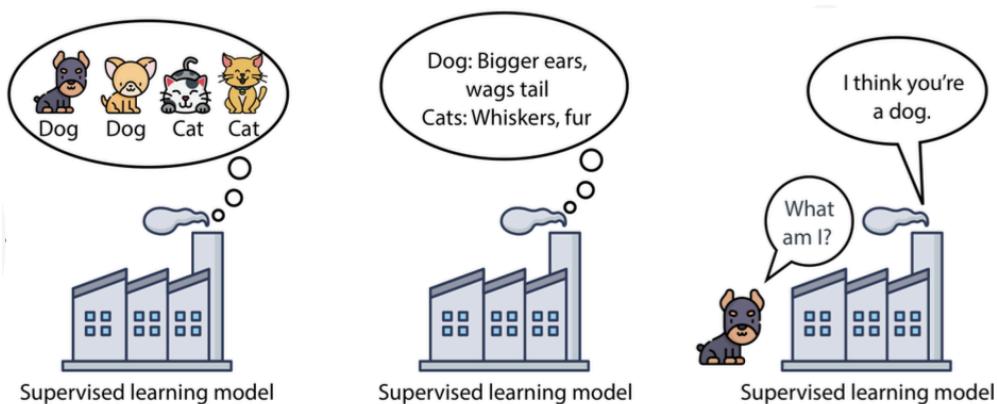
Naive Bayes Models

- Often used dimana effects are not independent
- $P(\text{Cause}, \text{Effect}_1, \dots, \text{Effect}_n) = P(\text{Cause}) \prod_i P(\text{Effect}_i | \text{cause})$

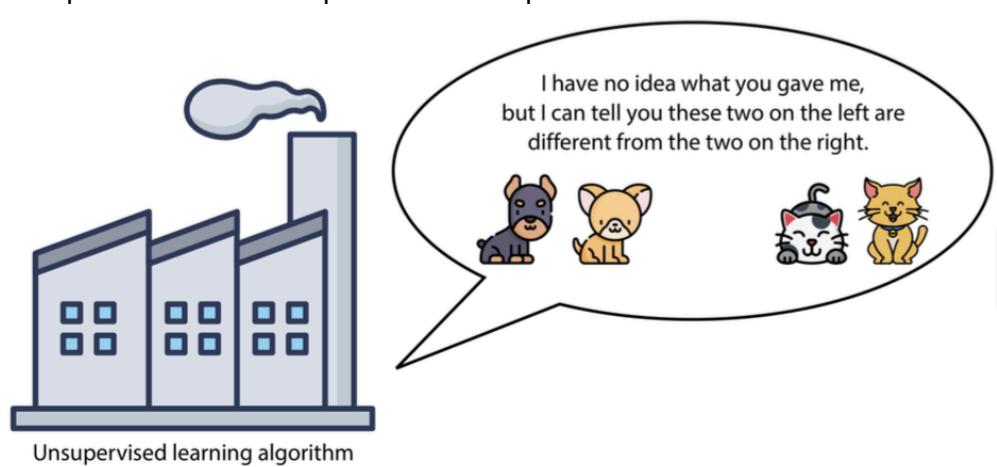
Materi 9. Machine Learning

Types of Machine Learning

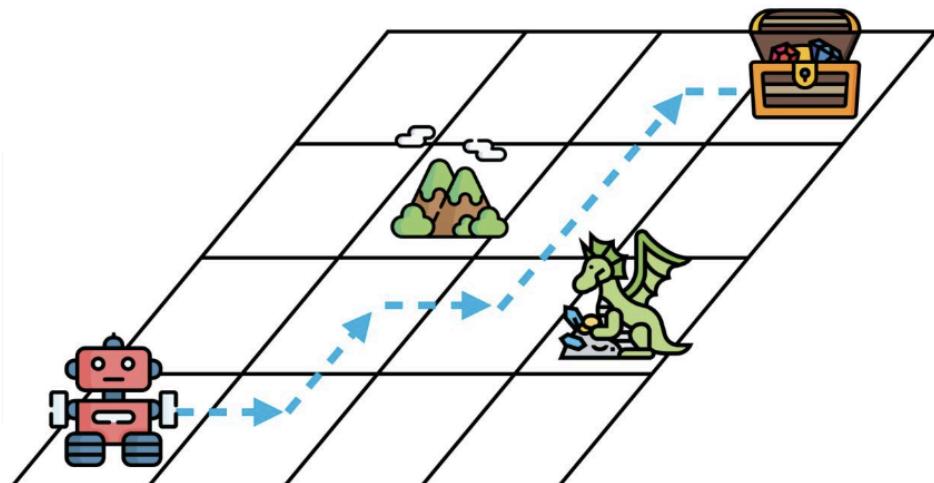
- Supervised = observes input-labeled output



- Unsupervised = learns a pattern in the input without feedback



- Reinforcement = learns from serangkaian rewards and punishments



LDT (Learning Decision Tree)



Consider the following app:



Atom Count



Beehive Finder



Check Mate Mate

and our customer data from the store ($6 \text{ out of } 2 \times 70 \times 3 = 240 \text{ combination}$):

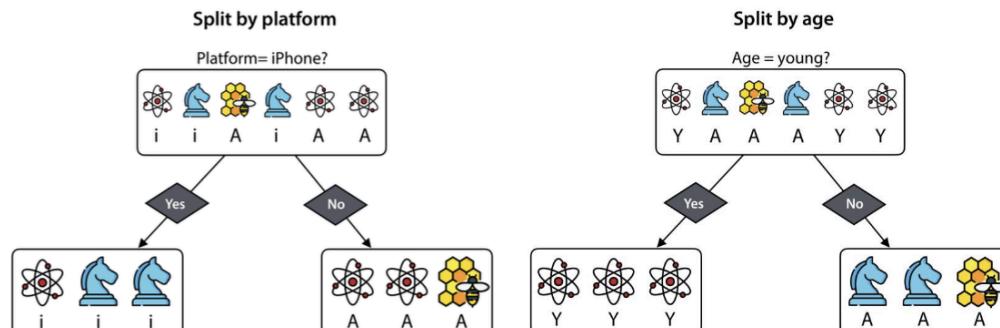
Platform	Age	App
iPhone	15	Atom Count
iPhone	25	Check Mate Mate
Android	32	Beehive Finder
iPhone	35	Check Mate Mate
Android	12	Atom Count
Android	14	Atom Count

Which app would you recommend for:

- o Customer 1: a 13-year-old iPhone user
- o Customer 2: a 28-year-old iPhone user
- o Customer 3: a 34-year-old Android user

Simplify data, $<20 = \text{"young"}$ otherwise "adult"

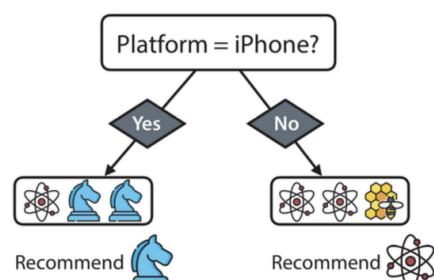
Which is the important attribute? Platform or age?



Which split is better?

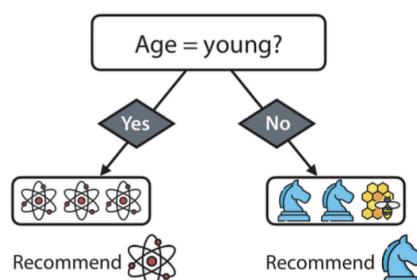
Correctness

Classifier 1: By platform



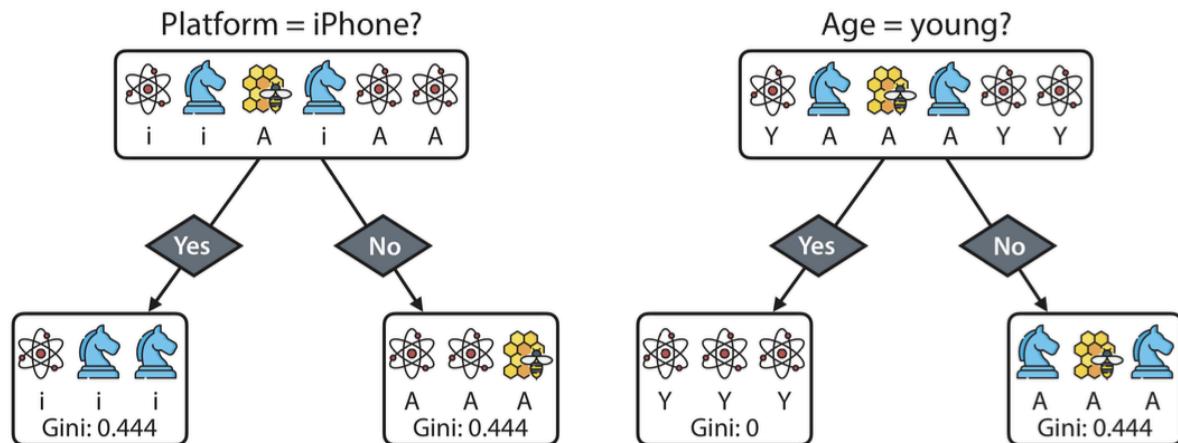
Correct 4 out of 6 times
Accuracy: 66.67%

Classifier 2: By age



Correct 5 out of 6 times
Accuracy: 83.33%

Gini Impurity Index

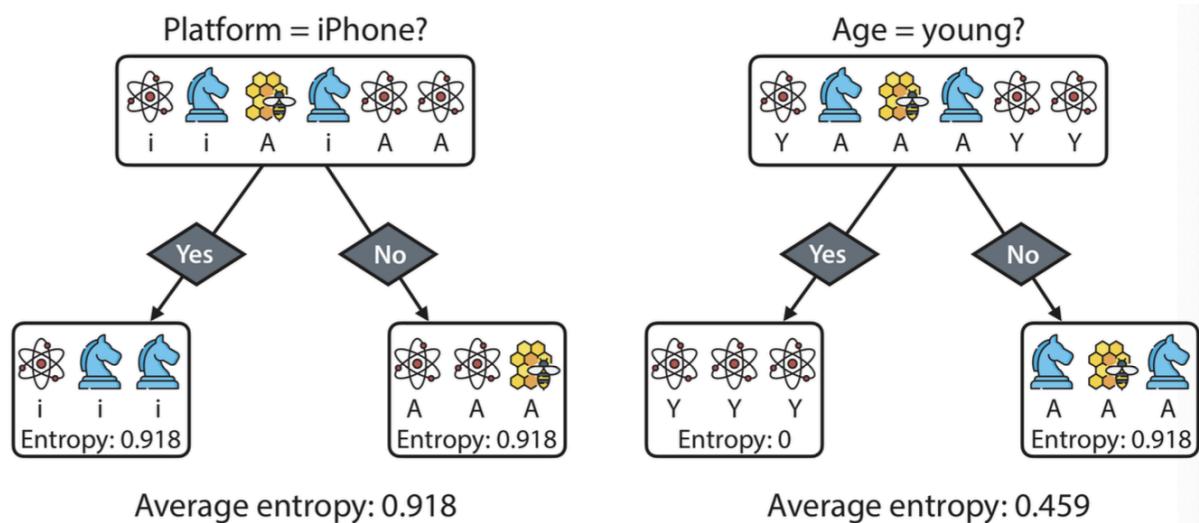


Average Gini impurity index: 0.444

- Iphone, yes = $1 - (\frac{1}{3})^2 - (\frac{2}{3})^2 = 0.444$
- Iphone, no = $1 - (\frac{2}{3})^2 - (\frac{1}{3})^2 = 0.444$
- Young, yes = $1 - (3/3)^2 = 0$
- Young, no = $1 - (\frac{2}{3})^2 - (\frac{1}{3})^2 = 0.444$
- Average iphone = $(0.444 + 0.444) / 2 = 0.444$
- Average young = $(0 + 0.444) / 2 = 0.222$

Average Gini impurity index: 0.222

Entropy

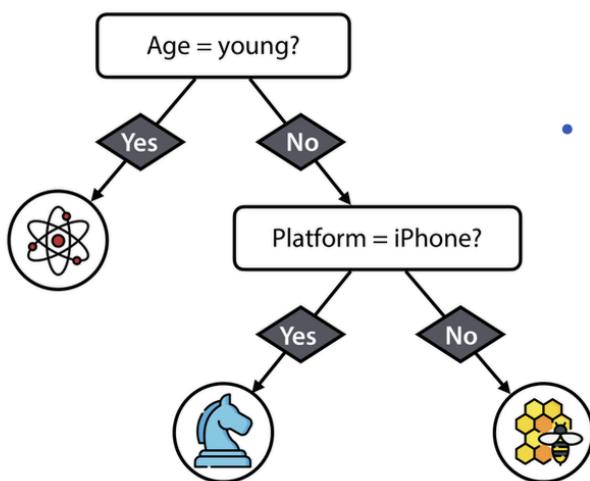


Average entropy: 0.918

- Iphone, yes = $-1 \times (\frac{1}{3} \log_2(\frac{1}{3}) + (\frac{2}{3}) \log_2(\frac{2}{3})) = 0.918$
- Iphone, no = $-1 \times (\frac{2}{3} \log_2(\frac{2}{3}) + (\frac{1}{3}) \log_2(\frac{1}{3})) = 0.918$
- Young, yes = $-1 \times (3/3 \log_2(3/3)) = 0$
- Young, no = $-1 \times (\frac{2}{3} \log_2(\frac{2}{3}) + (\frac{1}{3}) \log_2(\frac{1}{3})) = 0.918$
- Average iphone = $(0.918 + 0.918) / 2 = 0.918$
- Average young = $(0 + 0.918) / 2 = 0.459$

Average entropy: 0.459

Final decision tree



- Which app would you recommend for:
 - Customer 1: a 13-year-old iPhone user
 - Customer 2: a 28-year-old iPhone user
 - Customer 3: a 34-year-old Android user



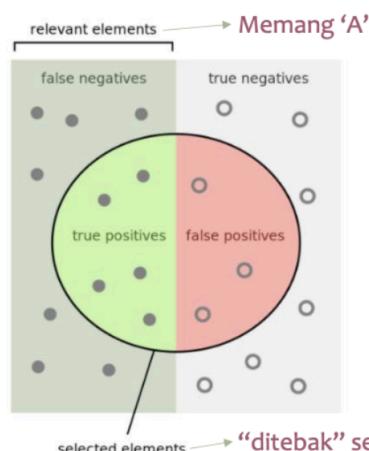
Confusion Matrix

Contoh: Diberikan 10 sample binatang, 6 kucing dan 4 anjing. Berdasarkan ciri-cirinya, classifier belajar untuk "menebak" apakah binatang tersebut kucing atau bukan. Setelah classifier melalui tahap learning, classifier salah "menebak" 2 kucing sebagai anjing dan 1 anjing sebagai kucing. 7 lainnya "ditebak" dengan benar.

Karena yang dibahas kucing, maka **POSITIVE = KUCING**

		True Class	
		Positive	Negative
Predicted Class	Positive	TP 4	FP 1
	Negative	FN 2	TN 3

$Accuracy = \frac{TP + TN}{population}$
 $Precision = \frac{TP}{predicted(+)}$
 $Recall = \frac{TP}{true(+)}$



Precision: keakuratan/tebakan benar dari semua tebakan 'A' —

Recall: keakuratan/tebakan benar dari semua actual 'A' —

Bahan bacaan: <https://towardsdatascience.com/accuracy-precision-recall-or-f1-33fb37c5cb9>

- TP = beneran kucing
- TN = beneran anjing
- FP = anjing dikira kucing
- FN = kucing dikira anjing
- Accuracy = $(TP + TN) / population$
- Recall = $TP / (TP + FN)$
- Precision = $TP / (TP + FP)$

Linear Regression

- Memodelkan hubungan antara satu atau lebih variabel independen (prediktor) dengan variabel dependen (respon)

Metrik Evaluasi

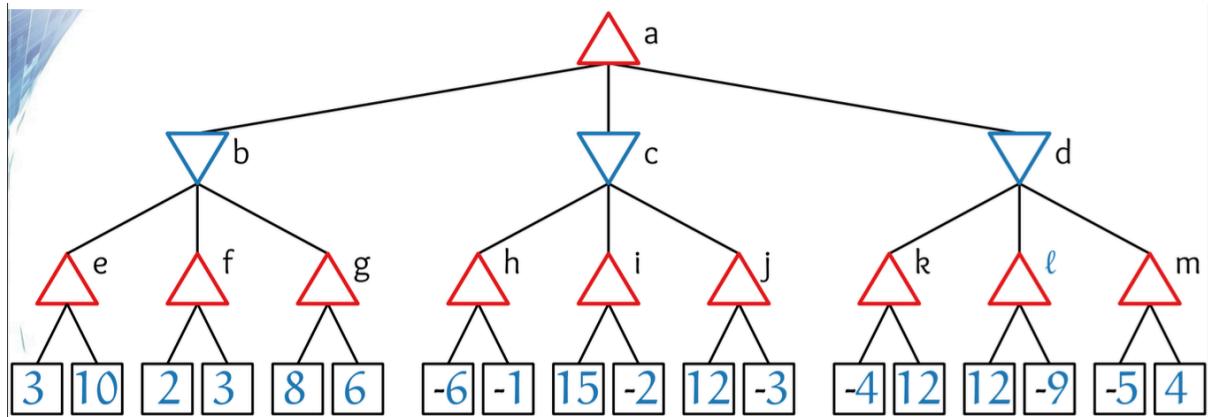
- a. RMSE (Root Mean Squared of Error)
 - Seberapa baik model regresi memprediksi nilai target dibandingkan dengan nilai aktualnya
- b. SSR (Sum of Squares for Regression)
 - Mengukur jumlah variasi dalam variabel dependen (target) yang dapat dijelaskan oleh model regresi
 - Semakin besar, semakin bagus
- c. SSE (Sum of Squares for Error)
 - Mengukur jumlah variasi dalam variabel dependen (target) yang tidak dapat dijelaskan oleh model regresi
 - Semakin kecil, semakin bagus
- d. SSTO (Sum of Squares Total)
 - Mengukur total variasi dalam variabel dependen tanpa mempertimbangkan pengaruh dari variabel independen
 - Jika nilainya kecil, hubungan antar variabel kurang signifikan

Soal Kuis

Nomor 1

Diberikan sebuah pohon permainan. Segitiga merah adalah pemain **Max** yang selalu berusaha memaksimalkan evaluation function. Segitiga (terbalik) biru adalah pemain **Min** yang selalu berusaha meminimalkan evaluation function

Menggunakan algoritma $\alpha - \beta$ pruning, sebutkan edge mana saja yang dipotong dan sebutkan nilai α dan β di parent node-nya pada saat itu! Tuliskan pasangan node-nya untuk menandakan edge, misal: "(e,3) = {\mathbf{\alpha}=2, \mathbf{\beta}=3}", jika edge antara e dan 3 dipotong!



Nomor 2

Wumpus World: agen mulai di posisi kiri bawah (1,1). Di posisi (1,2) dan (2,1), agen merasakan ada Stench. Simbol Stench → S & Wumpus → W.

- Tuliskan seluruh KB (R1, R2, dst) yang dimiliki agen saat ini (jumlahnya sedikit)! R yg ditulis hanya yang mengandung simbol W dan/atau S!
- Buatlah tabel kebenaran yg kolomnya menyatakan seluruh simbol W, S, dan R. Perhatikan bahwa baris pada tabel tidak banyak karena kita tidak perlu menuliskan semua kombinasi TRUE/FALSE untuk setiap simbol karena ada yang diketahui TRUE atau FALSE dari aturan pada Wumpus dan hasil sensor agen.
- Berdasarkan tabel, apakah $\text{KB} \models W_{2,2}$? Jelaskan jawaban anda maksimum dalam dua kalimat!

JAWABAN

1,2	2,2
1,1	2,1

a.

$$R1 = \text{negasi } S_{1,1}$$

$$R2 = S_{1,1} \Leftrightarrow W_{1,2} \vee W_{2,1}. \text{ Tidak perlu karena } S_{1,1} = \text{false}$$

$$R3 = S_{1,2} \Leftrightarrow W_{1,1} \vee W_{1,3} \vee W_{2,2}$$

$$R4 = S_{2,1} \Leftrightarrow W_{1,1} \vee W_{2,2} \vee W_{3,1}$$

$$R5 = \text{negasi } W_{1,1}$$

$$R6 = S_{1,2}$$

$$R7 = S_{2,1}$$

$W_{1,3}$	$W_{2,2}$	$W_{3,1}$	$S_{1,2}$	$S_{2,1}$	$R_3 : S_{1,2} \Leftrightarrow (W_{1,1} \vee W_{1,3} \vee W_{2,2})$	$R_4 : S_{2,1} \Leftrightarrow (W_{1,1} \vee W_{2,2} \vee W_{3,1})$
F	T	F	T	T	<input checked="" type="checkbox"/> (karena $W_{2,2} = T$)	<input checked="" type="checkbox"/> (karena $W_{2,2} = T$)
T	F	F	T	T	<input checked="" type="checkbox"/> (karena $W_{1,3} = T$)	<input type="checkbox"/> (karena tidak ada $W_{2,2}$ atau $W_{3,1}$)
F	F	T	T	T	<input type="checkbox"/> (karena tidak ada $W_{1,3}$ atau $W_{2,2}$)	<input checked="" type="checkbox"/> (karena $W_{3,1} = T$)

b.

- Iya, karena $S_{1,2}$ dan $S_{2,1}$ hanya bisa dijelaskan jika $W_{2,2} = \text{True}$

Nomor 3

- Mobil L diproduksi oleh tiga pabrik berbeda: X (50% dari seluruh V), Y (30%), dan Z (20%). Persentase mobil L berwarna hitam yang diproduksi oleh pabrik X: 40%, pabrik Y: 20%, dan pabrik Z: 10%.

Berapa probabilitas sebuah mobil L berwarna hitam diproduksi oleh pabrik X?

- Benda V diproduksi oleh tiga pabrik berbeda: X (60% dari seluruh V), Y (30%), dan Z (10%). Persentase benda V yang rusak saat produksi dari pabrik X: 2%, dari pabrik Y: 4%, dan dari pabrik Z: 8%.

Berapa probabilitas sebuah benda V yang diketahui sudah rusak saat produksi berasal dari pabrik X?

JAWABAN

1.

$$\begin{aligned}P(X) &= 0.50 \\P(Y) &= 0.30 \\P(Z) &= 0.20 \\P(H|X) &= 0.40 \\P(H|Y) &= 0.20 \\P(H|Z) &= 0.10\end{aligned}$$

$$P(X|H) = P(H|X)P(X) / P(H)$$

$$\begin{aligned}P(H) &= P(H|X)P(X) + P(H|Y)P(Y) + P(H|Z)P(Z) \\&= (0.40 \times 0.50) + (0.20 \times 0.30) + (0.10 \times 0.20) \\&= 0.20 + 0.06 + 0.02 = 0.28\end{aligned}$$

$$P(X|H) = 0.40 \times 0.50 / 0.28 = 0.714$$

2.

$$\begin{aligned}P(X) &= 0.60 \\P(Y) &= 0.30 \\P(Z) &= 0.10\end{aligned}$$

$$\begin{aligned}P(V|X) &= 0.02 \\P(V|Y) &= 0.04 \\P(V|Z) &= 0.08\end{aligned}$$

$$P(X|V) = P(V|X)P(X) / P(V)$$

$$\begin{aligned}P(V) &= P(V|X)P(X) + P(V|Y)P(Y) + P(V|Z)P(Z) \\&= (0.02 \times 0.60) + (0.04 \times 0.30) + (0.08 \times 0.10) \\&= 0.012 + 0.012 + 0.008 = 0.032\end{aligned}$$

$$P(X|V) = 0.02 \times 0.60 / 0.032 = 0.375$$

Nomor 4

1. Diberikan data restoran di tabel. Buatlah LDT untuk merekomendasikan pemberian tip atau tidak.

Seat	Type	Tip
Indoor	Breakfast	Yes
Indoor	Dinner	Yes
Outdoor	Lunch	No
Indoor	Lunch	No
Outdoor	Breakfast	Yes
Indoor	Dinner	No
Outdoor	Lunch	No
Outdoor	Dinner	Yes

2. Diberikan data instalasi sebuah aplikasi X di tabel. Buatlah LDT untuk merekomendasikan instalasi X atau tidak.

Type	OS	V?
PC	Linux	Yes
PC	MacOS	Yes
Laptop	Windows	No
PC	Windows	No
Laptop	Linux	Yes
PC	MacOS	No
Laptop	Windows	No
Laptop	MacOS	Yes

Pemilihan atribut menggunakan Gini Impurity Index.

Langkah-Langkah pemilihan atribut di setiap node harus diuraikan dengan lengkap sekalipun sudah jelas bagaimana bentuk pohonnya.

Jangan lupa gambar LDT final dengan daun yang menunjukkan klasifikasi akhir.

JAWABAN

Nomor 5

Model machine learning digunakan di sebuah gudang untuk mengklasifikasikan apakah sebuah paket itu besar (positif) atau kecil (negatif). Paket besar harus diidentifikasi dengan tepat karena perlu penanganan khusus!

Hasil model dari 100 buah paket:

- Prediksi: 40 paket diprediksi besar: hanya 30 yang benar besar
- Aktual (keadaan sebenarnya) terdapat 65 paket kecil

- a. Gambar dan isi (cth: TP = 17, dst) Confusion Matrix dari data di atas. Berikan label yang sesuai untuk kolom dan baris pada matrix tersebut!
- b. Dalam kasus ini, jelaskan dengan singkat (maks 4 kalimat) manakah yang lebih penting: recall atau precision?
- c. Sesuai jawaban (b), tulis rumus & hitung berapa recall/precision-nya!

JAWABAN

- a.
 - TP = beneran paket besar. 30
 - FP = paket kecil disangka paket besar. 10
 - TN = beneran paket kecil. 55
 - FN = paket besar disangka paket kecil. 5

	Positive	Negative
Positive	TP = 30	FN = 5
Negative	FP = 10	TN = 55

- b. Recall lebih penting, karena kesalahan dalam mengidentifikasi paket besar (False Negative) bisa menyebabkan masalah dalam penanganan khusus

- c.
 - Recall = $TP / (TP + FN) = 30 / (30 + 5) = 30/35$
 - Precision = $TP / (TP + FP) = 30 / (30 + 10) = 30/40$

Nomor 6

- Hitunglah harmonic mean dari {2, 4, 10, 20}
- Hitunglah geometric mean dari {2, 4, 10, 20}
- Apa kepanjangan dari RMSE? Mengapa E harus di-S? Jelaskan dengan singkat maksimum 3 kalimat!
- Apa kepanjangan dari RMSE? Mengapa MSE harus di-R? Jelaskan dengan singkat maksimum 3 kalimat!

JAWABAN

- $$\begin{aligned} & 1/2 + 1/4 + 1/10 + 1/20 \\ & = 10/20 + 5/20 + 2/20 + 1/20 \\ & = 18/20 = 9/10 = 0.9 \\ & n / 0.9 = 4 / 0.9 = 4.444 \end{aligned}$$
- $$\begin{aligned} & 2 \times 4 \times 10 \times 20 = 1600 \\ & = 1600 \text{ akar n atau sama dengan } 1600 \text{ pangkat } 1/4 \\ & = (40^2)^{1/4} \\ & = 40^{1/2} \\ & = \sqrt{40} = 2\sqrt{10} \end{aligned}$$
- RMSE adalah Root Mean Squared Error
Menghindari nilai negatif
- Karena hasil MSE masih dalam satuan kuadrat, mengembalikan unit ke satuan asli variabel target untuk memudahkan interpretasi error