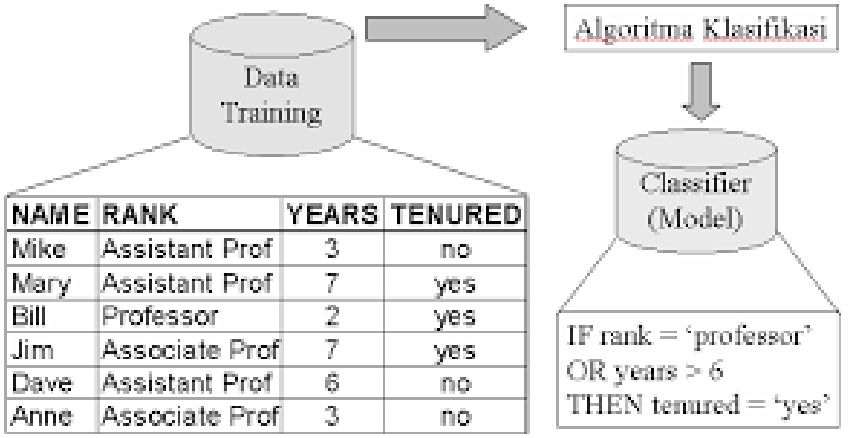
Klasifikasi

15-11-2023

Algoritma klasifikasi

- Algoritma Klasifikasi adalah salah satu metode dalam proses KDD yang ada pada tahap data mining untuk mengidentifikasi setiap kelompok berdasarkan kasus dan kemudian dikelompokkan datanya berdasarkan atribut
- Klasifikasi merupakan suatu proses pengkategorian yang digunakan untuk menentukan kelas dari data yang tidak di ketahui label class nya. Memprediksi label kelas adalah tujuan utama dari klasifikasi ini.
- Dua tahapan yang dimiliki pada poses klasifikasi, yaitu terlebih dahulu learning yang mana learning ini menganalisa sebuah data training dengan menggunakan algoritma klasifikasi, lalu proses klasifikasi yaitu memprediksi ketepatan dari klasifikasi dengan menggunakan data testing.
- Klasifikasi data dapat dimulai dengan membuat aturan klasifikasi tertentu menggunakan data latih dan data uji.



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

- Merupakan proses pembelajaran suatu fungsi tujuan yang **memetakan** tiap himpunan **atribut x** ke satu dari **label kelas y** yang didefinisikan sebelumnya.

Model Klasifikasi

Pemodelan Deskriptif

- Model Klasifikasi yang dpt berfungsi sbg alat **penjelasan** untuk membedakan **obyek-obyek** dalam **kelas-kelas** yang berbeda.

Pemodelan Prediktif

- Model klasifikasi yang dapat digunakan untuk **memprediksi label kelas** yang tidak diketahui pada suatu ***object/record***.

Klasifikasi memerlukan Training Set

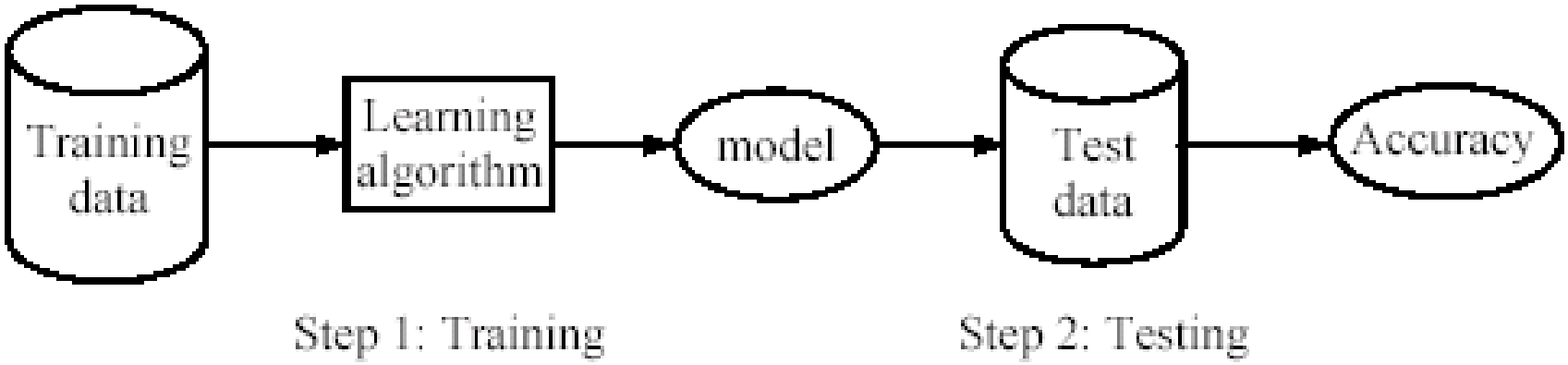
Klasifikasi adalah proses pembelajaran secara terbimbing (*supervised learning*)

Untuk melakukan klasifikasi, dibutuhkan ***training set*** sebagai data pembelajaran

Setiap **sampel** dari training set memiliki **atribut** dan ***class label***

Dua Tahapan Klasifikasi

- ***Learning* (*training*):**
  Pembelajaran menggunakan **data training** (untuk ***Naïve Bayesian Classifier***, **nilai probabilitas** dihitung dalam proses pembelajaran)
- **Testing:**
  Menguji model menggunakan data testing



Teori Bayesian: Sebagai Dasar

- **P(H|X)**, yaitu peluang hipotesa **H** berdasar kondisi **X**
- **X**:data sampel dengan kelas (label) yang tidak diketahui
- **H**:merupakan hipotesa bahwa **X** adalah data dengan kelas (label) C.
- **P(H)** : peluang dari hipotesa **H**
- **P(X)** adalah peluang dari **X** yang diamati
- **P(X|H)** : peluang **X**, berdasarkan kondisi pada hipotesa **H**

$$P(H|X)=\frac{P(X|H)P(H)}{P(X)}$$

Naïve Bayesian Classifier

- Adalah metode classifier yang berdasarkan **probabilitas** dan Teorema Bayesian dengan asumsi bahwa setiap variabel **X** bersifat bebas(*independence)*
- Dengan kata lain, **Naïve Bayesian Classifier** mengasumsikan bahwa keberadaan sebuah atribut (variabel) tidak ada kaitannya dengan beradaan atribut (variabel) yang lain

Naïve Bayesian Classifier

$$P(X \mid C_i) = \prod_{k=1}^{n} P(x_k \mid C_i)$$

- Karena asumsi **atribut tidak saling terkait** (*conditionally independent), maka:*
- Bila P(X|Ci) dapat diketahui melalui perhitungan di atas, maka klas (label) dari data sampel X adalah klas (label) yang memiliki *P(X|Ci)*P(Ci) maksimum*

Naïve Bayesian Classifier

- Namun jika atribut ke-i **bersifat kontinu**, maka P(x_i | C) diestimasi dengan fungsi densitas Gauss.
- Distribusi normal adalah distribusi dari variabel acak kontinu. Kadang-kadang distribusi normal disebut juga dengan distribusi Gauss. Distribusi ini merupakan distribusi yang paling penting dan paling banyak digunakan di bidang statistika.

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \qquad e = 2{,}7183$$

dengan μ = mean, dan σ = deviasi standar.

Naïve Bayes

- Dataset

Umur	Pendapatan	Mhs	Rating Kredit	Beli Komputer
<=30	tinggi	bukan	fair	Tdk
<=30	tinggi	bukan	excellent	tdk
30…40	tinggi	bukan	fair	ya
>40	sedang	bukan	fair	ya
>40	rendah	ya	fair	ya
>40	rendah	ya	excellent	tdk
31…40	rendah	ya	excellent	ya
<=30	sedang	bukan	fair	tdk
<=30	rendah	ya	fair	ya
>40	sedang	ya	fair	ya
<=30	sedang	ya	excellent	ya
31…40	sedang	bukan	excellent	ya
31…40	tinggi	ya	fair	ya
>40	sedang	bukan	excellent	tdk

class:

C1: Beli Komputer: **ya**

C2: Beli Komputer: **tdk**

bila data baru yg blm memiliki class sbb:

X =(umur<=30, pendapatan=sedang, mhs=ya,rating kredit= Fair)

Hitung P(Xk|Ci) utk setiap class i

- X =(**umur<=30**, pendapatan=sedang, mhs=ya, rating kredit= Fair)

Umur	Pendapatan	Mhs	Rating Kredit	Beli Komputer
<=30	tinggi	bukan	fair	tdk
<=30	tinggi	bukan	excellent	tdk
<=30	sedang	bukan	fair	tdk
<=30	rendah	ya	fair	ya
<=30	sedang	ya	excellent	ya
>40	rendah	ya	excellent	tdk
>40	sedang	bukan	fair	ya
>40	rendah	ya	fair	ya
>40	sedang	ya	fair	ya
>40	sedang	bukan	excellent	tdk
30...40	tinggi	bukan	fair	ya
31...40	rendah	ya	excellent	ya
31...40	sedang	bukan	excellent	ya
31...40	tinggi	ya	fair	ya

P(umur<=30| beli_komputer=ya) = > 2/9 = 0.220

P(umur<=30| beli_komputer=tdk) => 3/5 = 0.600

Hitung P(Xk | Ci) utk setiap class i

- X =(umur<=30, **pendapatan=sedang**, mhs=ya,rating kredit= Fair)

ID	Umur	Pendapatan	Mhs	Rating Kredit	Beli Komputer
1	>40	rendah	ya	excellent	tdk
2	<=30	rendah	ya	fair	ya
3	>40	rendah	ya	fair	ya
4	31…40	rendah	ya	excellent	ya
5	<=30	sedang	bukan	fair	tdk
6	>40	sedang	bukan	excellent	tdk
7	<=30	sedang	ya	excellent	ya
8	>40	sedang	bukan	fair	ya
9	>40	sedang	ya	fair	ya
10	31…40	sedang	bukan	excellent	ya
11	<=30	tinggi	bukan	fair	tdk
12	<=30	tinggi	bukan	excellent	tdk
13	30…40	tinggi	bukan	fair	ya
14	31…40	tinggi	ya	fair	ya

P(pendapatan=sedang| beli_komputer=ya) => 4/9= 0.444

P(pendapatan=sedang| beli_komputer=tdk) => 2/5=0.400

Hitung P(Xk|Ci) utk setiap class i

X =(umur<=30, pendapatan=sedang, **mhs=ya**,rating kredit= Fair)

ID	Umur	Pendapatan	Mhs	Rating Kredit	Beli Komputer
1	<=30	sedang	bukan	fair	tdk
2	>40	sedang	bukan	excellent	tdk
3	<=30	tinggi	bukan	fair	tdk
4	<=30	tinggi	bukan	excellent	tdk
5	>40	sedang	bukan	fair	ya
6	31…40	sedang	bukan	excellent	ya
7	30…40	tinggi	bukan	fair	ya
8	>40	rendah	ya	excellent	tdk
9	<=30	rendah	ya	fair	ya
10	>40	rendah	ya	fair	ya
11	31…40	rendah	ya	excellent	ya
12	<=30	sedang	ya	excellent	ya
13	>40	sedang	ya	fair	ya
14	31…40	tinggi	ya	fair	ya

P(mhs=ya| beli_komputer=ya) => 6/9 = 0.670

P(mhs=ya| beli_komputer=tdk) => 1/5 = 0.200

Hitung P(X_k | Ci) utk setiap class i

X =(umur<=30, pendapatan=sedang, mhs=ya,**rating kredit= Fair**)

ID	Umur	Pendapatan	Mhs	Rating Kredit	Beli Komputer
1	<=30	tinggi	bukan	excellent	tdk
2	>40	sedang	bukan	excellent	tdk
3	>40	rendah	ya	excellent	tdk
4	31…40	sedang	bukan	excellent	ya
5	31…40	rendah	ya	excellent	ya
6	<=30	sedang	ya	excellent	ya
7	<=30	sedang	bukan	fair	tdk
8	<=30	tinggi	bukan	fair	tdk
9	>40	sedang	bukan	fair	ya
10	30…40	tinggi	bukan	fair	ya
11	<=30	rendah	ya	fair	ya
12	>40	rendah	ya	fair	ya
13	>40	sedang	ya	fair	ya
14	31…40	tinggi	ya	fair	ya

P(rating kredit=fair| beli_komputer=ya) => 6/9 = 0.670

P(rating kredit=fair| beli_komputer=tdk) => 2/5 = 0.400

$$P(X \mid C_i) = \prod_{k=1}^{n} P(x_k \mid C_i)$$

Hitung P(Xk | Ci) utk setiap class I	
P(umur<=30| beli_komputer=ya) = 2/9	0.222
P(umur<=30| beli_komputer=tdk) = 3/5	0.600
P(pendapatan=sedang| beli_komputer=ya) = 4/9	0.444
P(pendapatan=sedang| beli_komputer=tdk) = 2/5	0.400
P(mhs=ya| beli_komputer=ya) = 6/9	0.667
P(mhs=ya| beli_komputer=tdk) = 1/5	0.200
P(rating kredit=fair| beli_komputer=ya) = 6/9	0.667
P(rating kredit=ya| beli_komputer=tdk) = 2/5	0.400

- Hitung P(X|Ci) untuk setiap Class:
  - P(X|beli_computer="ya")=0.222 x 0.444 x 0.667 x 0.667 = 0.044
  - P(X|beli_computer="tdk")=0.600 x 0.400 x 0.200 x 0.400 = 0.019

P(X|Ci)*P(Ci):

- P(X|beli_computer="ya") * P(beli_computer="ya")
  0.044 * (9/14) = 0.028
- P(X|beli_computer="tdk") * P(beli_computer="tdk")
  0.019 * (5/14) = 0.007

X memiliki class "beli_computer=ya"

karena

P(X|beli_computer="ya") memiliki nilai maksimum pada perhitungan di atas

Contoh Kasus (2)

- Misalnya terdapat ingin diketahui apakah suatu objek masuk dalam ketegori dipilih untuk perumahan atau tidak dengan algoritma Naive Bayes Classifier. Untuk menetapkan suatu daerah akan dipilih sebagai lokasi untuk mendirikan perumahan, telah dihimpun 10 aturan.

...

- Ada 4 atribut yang digunakan, yaitu:
  - harga tanah per meter persegi (C1),
  - jarak daerah tersebut dari pusat kota (C2),
  - ada atau tidaknya angkutan umum di daerah tersebut (C3), dan
  - keputusan untuk memilih daerah tersebut sebagai lokasi perumahan (C4).

...

Probabilitas kemunculan setiap nilai untuk atribut Harga Tanah (C1)

Harga tanah	Jumlah kejadian "Dipilih"		Probabilitas	
	Ya	Tidak	Ya	Tidak
Murah	2	1	2/5	1/5
Sedang	2	1	2/5	1/5
Mahal	1	3	1/5	3/5
Jumlah	*5*	*5*	*1*	*1*

...

Probabilitas kemunculan setiap nilai untuk atribut Jarak dari Pusat Kota (C2)

Jarak Dari Pusat Kota	Jumlah kejadian "Dipilih"		Probabilitas	
	Ya	Tidak	Ya	Tidak
Dekat	3	0	3/5	0
Sedang	2	1	2/5	1/5
Jauh	0	4	0	4/5
Jumlah	*5*	*5*	*1*	*1*

...

Probabilitas kemunculan setiap nilai untuk atribut Ada Angkutan Umum (C3)

Angkutan Umum	Jumlah kejadian "Dipilih"		Probabilitas	
	Ya	Tidak	Ya	Tidak
Ada	1	3	1/5	3/5
Tidak	4	2	4/5	2/5
Jumlah	*5*	*5*	*1*	*1*

Probabilitas kemunculan setiap nilai untuk atribut Dipilih untuk perumahan (C4)

Dipilih Untuk Perumahan	Jumlah kejadian "Dipilih"		Probabilitas	
	Ya	Tidak	Ya	Tidak
Jumlah	*5*	*5*	*1/2*	*1/2*

...

- Test Set: Berdasarkan data tersebut, apabila diketahui suatu daerah dengan harga tanah **MAHAL**, jarak dari pusat kota **SEDANG**, dan **ADA** angkutan umum, maka dapat dihitung:

YA = P(Ya|Tanah=MAHAL)*P(Ya|Jarak=SEDANG)*
P(Ya|Angkutan=ADA)*P(Ya)
= 1/5 x 2/5 x 1/5 x 5/10 = 2/125 = 0,008

TIDAK=P(Tidak|Tanah=MAHAL)*P(Tidak|Jarak=SEDANG)*
P(Tidak|Angkutan=ADA)*P(Tidak)
= 3/5 x 1/5 x 3/5 x 5/10 = 2/125 = 0,036

...

- Nilai probabilitas dapat dihitung dengan melakukan normalisasi terhadap likelihood tersebut sehingga jumlah nilai yang diperoleh = 1

Probabilitas Ya = $\frac{0{,}008}{0{,}008 + 0{,}036} = 0{,}182.$

Probabilitas Tidak = $\frac{0{,}036}{0{,}008 + 0{,}036} = 0{,}818.$

Klasifikasi : TIDAK

Contoh Kasus (2)

- Untuk jenis data harga tanah dan jarak pusat kota yang kontinue, misalnya :

Aturan ke-	Harga tanah (C1)	Jarak dari pusat kota (C2)	Ada angkutan umum (C3)	Dipilih untuk perumahan (C4)
1	100	2	Tidak	Ya
2	200	1	Tidak	Ya
3	500	3	Tidak	Ya
4	600	20	Tidak	Tidak
5	550	8	Tidak	Tidak
6	250	25	Ada	Tidak
7	75	15	Ada	Tidak
8	80	10	Tidak	Ya
9	700	18	Ada	Tidak
10	180	8	Ada	Ya

...

Probabilitas kemunculan setiap nilai untuk atribut Harga Tanah (C1)

	Ya	Tidak
1	100	600
2	200	550
3	500	250
4	80	75
5	180	700
Mean (μ) Deviasi standar (σ)	212 168,8787	435 261,9637

...

Probabilitas kemunculan setiap nilai untuk atribut Jarak dari Pusat Kota (C2)

	Ya	Tidak
1	2	20
2	1	8
3	3	25
4	10	15
5	8	18
Mean (μ) Deviasi standar (σ)	4,8 3,9623	17,2 6,3008

...

Probabilitas kemunculan setiap nilai untuk atribut Angkutan Umum (C3)

Angkutan Umum	Jumlah kejadian "Dipilih"		Probabilitas	
	Ya	Tidak	Ya	Tidak
Ada	1	3	1/5	3/5
Tidak	4	2	4/5	2/5
Jumlah	*5*	*5*	*1*	*1*

• • •

Probabilitas kemunculan setiap nilai untuk atribut Dipilih untuk Perumahan (C4)

Dipilih Untuk Perumahan	Jumlah kejadian "Dipilih"		Probabilitas	
	Ya	Tidak	Ya	Tidak
Jumlah	*5*	*5*	*1/2*	*1/2*

- Apabila diberikan C1 = 300, C2 = 17, C3 = Tidak, maka:

$$f(C1=300\mid ya)=\frac{1}{\sqrt{2\pi}(168{,}8787)}e^{\frac{-(300-212)^2}{2(168{,}8787)^2}}=0{,}0021.$$

$$f(C1=300\mid tidak)=\frac{1}{\sqrt{2\pi}(261.9637)}e^{\frac{-(300-435)^2}{2(261.9637)^2}}=0{,}0013.$$

$$f(C2=17\mid ya)=\frac{1}{\sqrt{2\pi}(3.9623)}e^{\frac{-(17-4{,}8)^2}{2(3.9623)^2}}=0{,}0009.$$

$$f(C2=17\mid tidak)=\frac{1}{\sqrt{2\pi}(6{,}3008)}e^{\frac{-(17-17{,}2)^2}{2(6{,}3008)^2}}=0{,}0633.$$

Likelihood Ya = (0,0021) x (0,0009) x 4/5 x 5/10

= 0,000000756.

Likelihood Tidak = (0,0013) x (0,0633) x 2/5 x 5/10

= **0,000016458.**

- Nilai probabilitas dapat dihitung dengan melakukan normalisasi terhadap likelihood tersebut sehingga jumlah nilai yang diperoleh = 1

Probabilitas Ya = $\frac{0{,}000000756}{0{,}000000756 + 0{,}000016458} = 0{,}0439.$

Probabilitas Tidak = $\frac{0{,}000016458}{0{,}000000756 + 0{,}000016458} = 0{,}9561.$

Klasifikasi : TIDAK

Naïve Bayesian: Summary

- Kekuatan:
  - Mudah diimplementasi
  - Memberikan hasil yang baik untuk banyak kasus
- Kelemahan:
  - Harus mengasumsi bahwa antar fitur tidak terkait (*independent) Dalam realita,* keterkaitan itu ada
  - Keterkaitan tersebut tidak dapat dimodelkan oleh Naïve Bayesian Classifier

Latihan

ID	OUTLOOK	TEMPERATUR	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

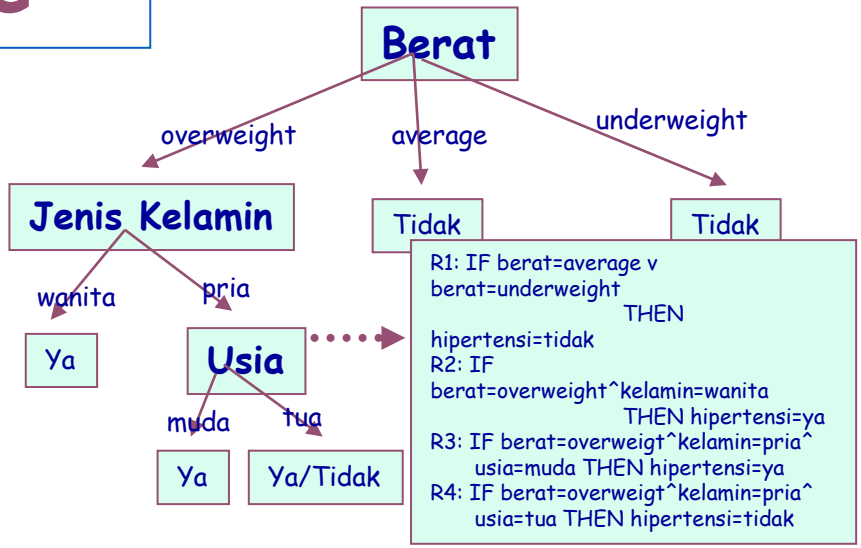
Tentukan *class label* dari X:

X =(Outlook=Rainy,Temperature=Cool, Humidity=High, Windy=False)

Konsep Decision Tree

Mengubah data menjadi pohon keputusan (*decision tree*) dan aturan-aturan keputusan (*rule*)

Nama	Usia	Berat	Kelamin	Hipertensi
Ali	muda	overweight	pria	ya
Edi	muda	underweight	pria	tidak
Annie	muda	average	wanita	tidak
Budiman	tua	overweight	pria	tidak
Herman	tua	overweight	pria	ya
Didi	muda	underweight	pria	tidak
Rina	tua	overweight	wanita	ya
Gatot	tua	average	pria	tidak

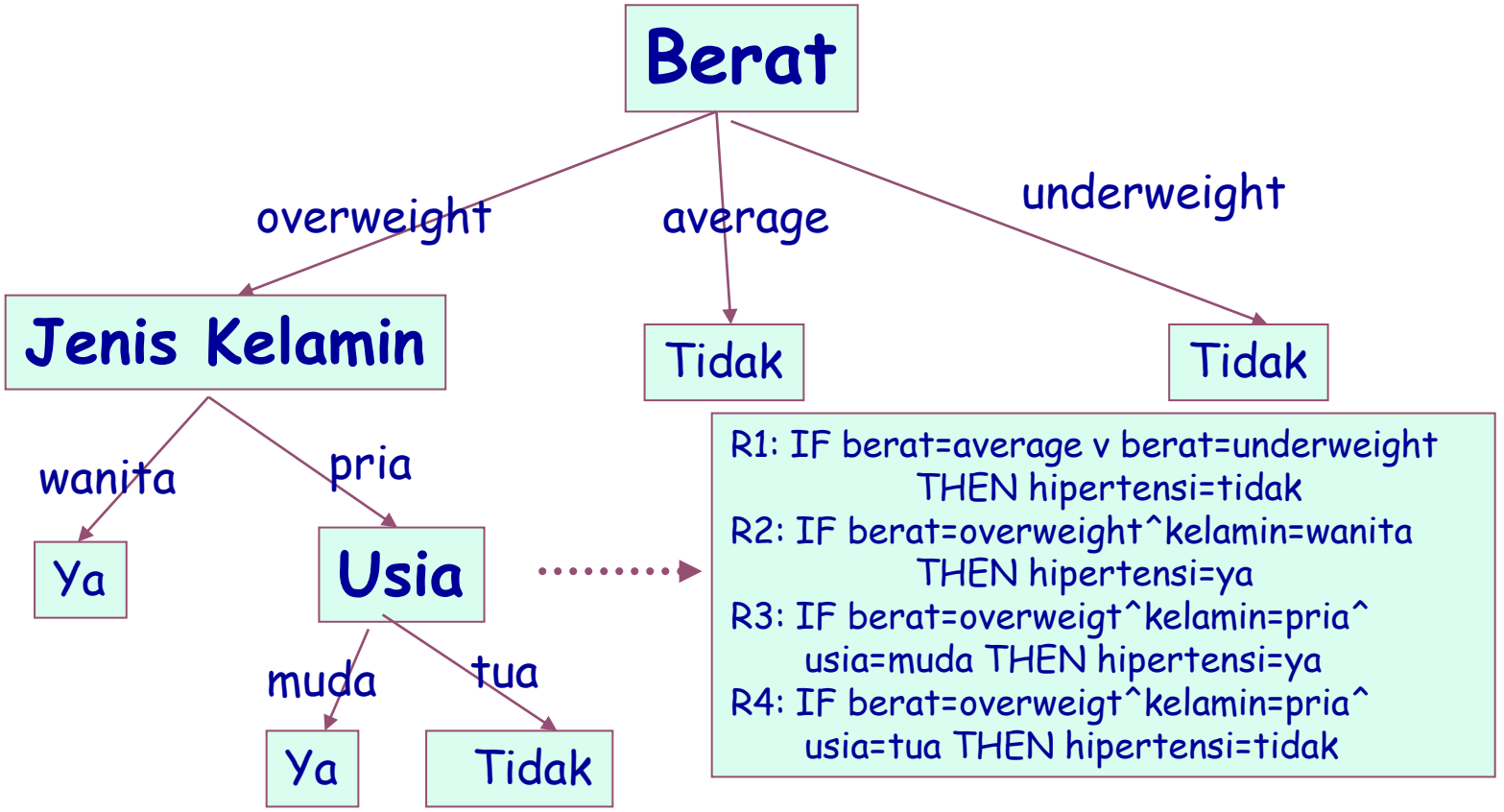


Gambaran Pemakaian Decision Tree

Membuat aturan (rule) yang dapat digunakan untuk menentukan apakah seseorang mempunyai potensi untuk menderita hipertensi atau tidak berdasarkan data usia, berat badan dan jenis kelamin.

#	Usia	Berat Badan	Kelamin	Hipertensi
1	muda	overweight	pria	ya
2	muda	underweight	pria	tidak
3	muda	average	wanita	tidak
4	tua	overweight	pria	tidak
5	tua	overweight	pria	ya
6	muda	underweight	pria	tidak
7	tua	overweight	wanita	ya
8	tua	average	pria	tidak

Gambaran Pemakaian Decision Tree

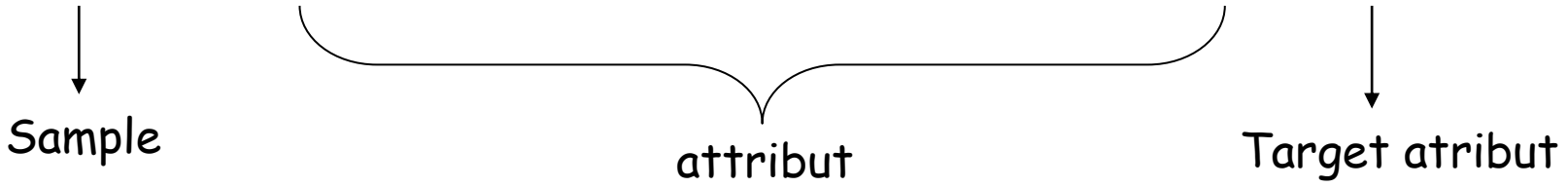


Konsep Data
Dalam Decision Tree

- Data dinyatakan dalam bentuk tabel dengan atribut dan record.
- **Atribut** menyatakan suatu parameter yang dibuat sebagai kriteria dalam pembentukan tree. Misalkan untuk menentukan main tenis, kriteria yang diperhatikan adalah cuaca, angin dan temperatur. Salah satu atribut merupakan atribut yang menyatakan data solusi per-item data yang disebut dengan **target atribut**.
- Atribut memiliki nilai-nilai yang dinamakan dengan **instance**. Misalkan atribut cuaca mempunyai instance berupa cerah, berawan dan hujan.

Konsep Data
Dalam Decision Tree

Nama	Cuaca	Angin	Temperatur	Main
Ali	cerah	keras	panas	tidak
Budi	cerah	lambat	panas	ya
Heri	berawan	keras	sedang	tidak
Irma	hujan	keras	dingin	tidak
Diman	cerah	lambat	dingin	ya

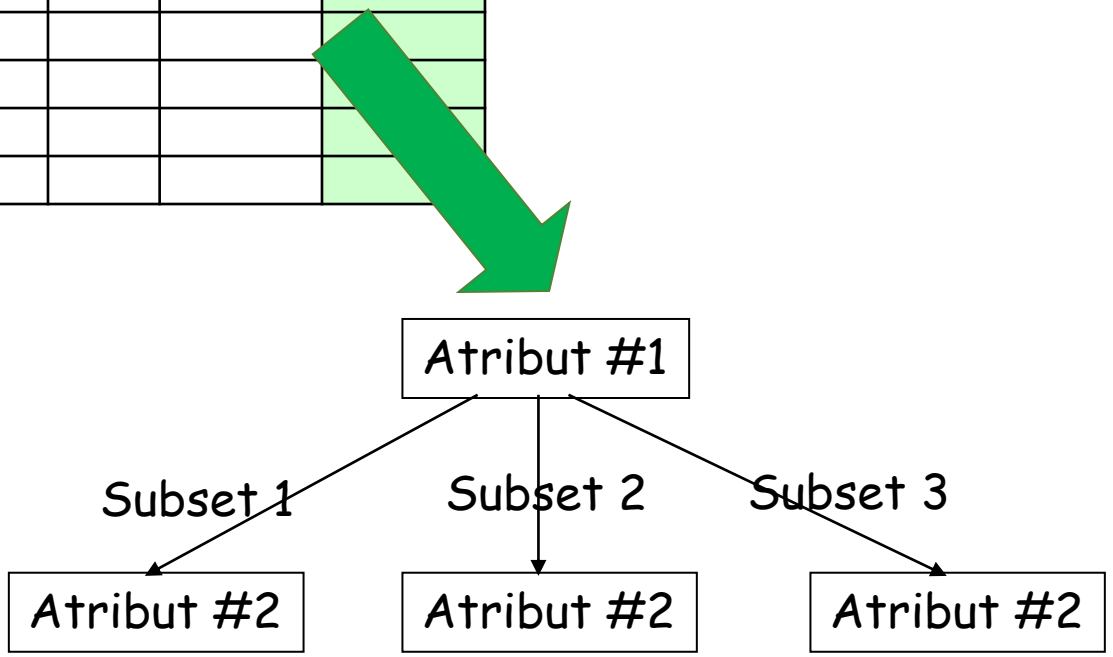


Proses Dalam Decision Tree

- Mengubah bentuk data (tabel) menjadi model tree. Dalam Modul ini menggunakan algoritma ID3.
- Mengubah model tree menjadi rule
- Menyederhanakan Rule (Pruning)

Proses Data Menjadi Tree

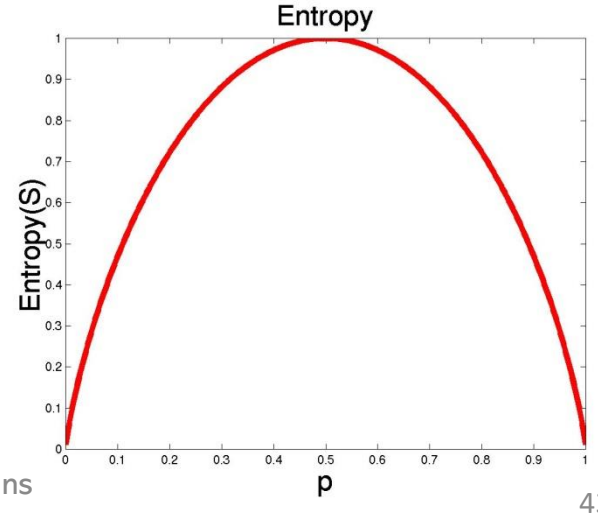
Indentity Atribut	Atribut 1	Atribut 2	Atribut 3	.....	Atribut n	Target Atribut



Entropy

- S adalah ruang (data) sample yang digunakan untuk training.
- P+ adalah jumlah yang bersolusi positif (mendukung) pada data sample untuk kriteria tertentu.
- P- adalah jumlah yang bersolusi negatif (tidak mendukung) pada data sample untuk kriteria tertentu.
- Besarnya Entropy pada ruang sample S didefinisikan dengan:

$$\text{Entropy}(S) = -p_+ \log_2 p_+ - p_- \log_2 p_-$$



Definisi Entropy

- Entropy(S) adalah jumlah bit yang diperkirakan dibutuhkan untuk dapat mengekstrak suatu kelas (+ atau -) dari sejumlah data acak pada ruang sample S.
- Entropy bisa dikatakan sebagai kebutuhan bit untuk menyatakan suatu kelas. Semakin kecil nilai Entropy maka semakin baik untuk digunakan dalam mengekstraksi suatu kelas.
- Panjang kode untuk menyatakan informasi secara optimal adalah $-\log_2 p$ bits untuk messages yang mempunyai probabilitas p.
- Sehingga jumlah bit yang diperkirakan untuk mengekstraksi S ke dalam kelas adalah:

$$-p_+ \log_2 p_+ - p_- \log_2 p_-$$

Contoh Permasalahan Penentuan Hipertensi Menggunakan Decision Tree

Data diambil dengan 8 sample, dengan pemikiran bahwa yang mempengaruhi seseorang menderita hipertensi atau tidak adalah usia, berat badan, dan jenis kelamin.

Usia mempunyai instance:
muda dan tua
Berat badan mempunyai instance:
underweight, average dan overweight
Jenis kelamin mempunyai instance:
pria dan wanita

Data Sample yang Digunakan Untuk Menentukan Hipertensi

Nama	Usia	Berat	Kelamin	Hipertensi
Ali	muda	overweight	pria	ya
Edi	muda	underweight	pria	tidak
Annie	muda	average	wanita	tidak
Budiman	tua	overweight	pria	tidak
Herman	tua	overweight	pria	ya
Didi	muda	underweight	pria	tidak
Rina	tua	overweight	wanita	ya
Gatot	tua	average	pria	tidak

Langkah Mengubah Data Menjadi Tree

- Menentukan Node Terpilih
- Menyusun Tree

Menentukan Node Terpilih

- Untuk menentukan node terpilih, gunakan nilai Entropy dari setiap kriteria dengan data sample yang ditentukan.
- Node terpilih adalah kriteria dengan Entropy yang paling kecil.

Memilih Node Awal

Usia	Hipertensi	Jumlah
muda	Ya (+)	1
muda	Tidak (-)	3
tua	ya	2
tua	tidak	2

Usia = muda

$$q_1 = -\frac{1}{4}\log_2\frac{1}{4} - \frac{3}{4}\log_2\frac{3}{4} = 0.81$$

Usia = tua

$$q_2 = -\frac{2}{4}\log_2\frac{2}{4} - \frac{2}{4}\log_2\frac{2}{4} = 1$$

Entropy untuk Usia:

$$E = \frac{4}{8}q_1 + \frac{4}{8}q_2 = \frac{4}{8}(0.81) + \frac{4}{8}(1) = 0.91$$

Memilih Node Awal

Usia	Hipertensi	Jumlah
muda	ya	1
muda	tidak	3
tua	ya	2
tua	tidak	2

Entropy = 0.91

Berat	Hipertensi	Jumlah
overweight	ya	3
overweight	tidak	1
average	ya	0
average	tidak	2
underweight	ya	0
underweight	tidak	2

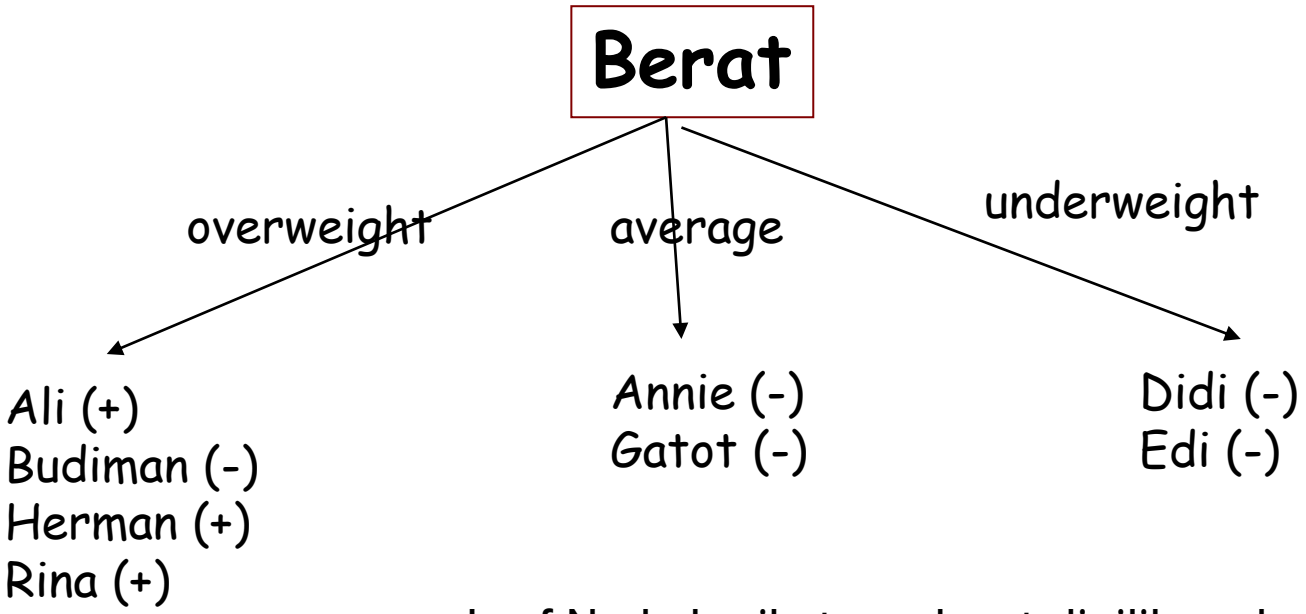
Entropy = 0.41

Kelamin	Hipertensi	Jumlah
pria	ya	2
pria	tidak	4
wanita	ya	1
wanita	tidak	1

Entropy = 0.94

Terpilih atribut BERAT BADAN sebagai node awal karena memiliki entropy terkecil

Penyusunan Tree Awal



Leaf Node berikutnya dapat dipilih pada bagian yang mempunyai nilai + dan -, pada contoh di atas hanya berat=overweight yang mempunyai nilai + dan – maka semuanya pasti mempunya leaf node. Untuk menyusun leaf node lakukan satu-persatu.

Penentuan Leaf Node Untuk Berat=Overweight

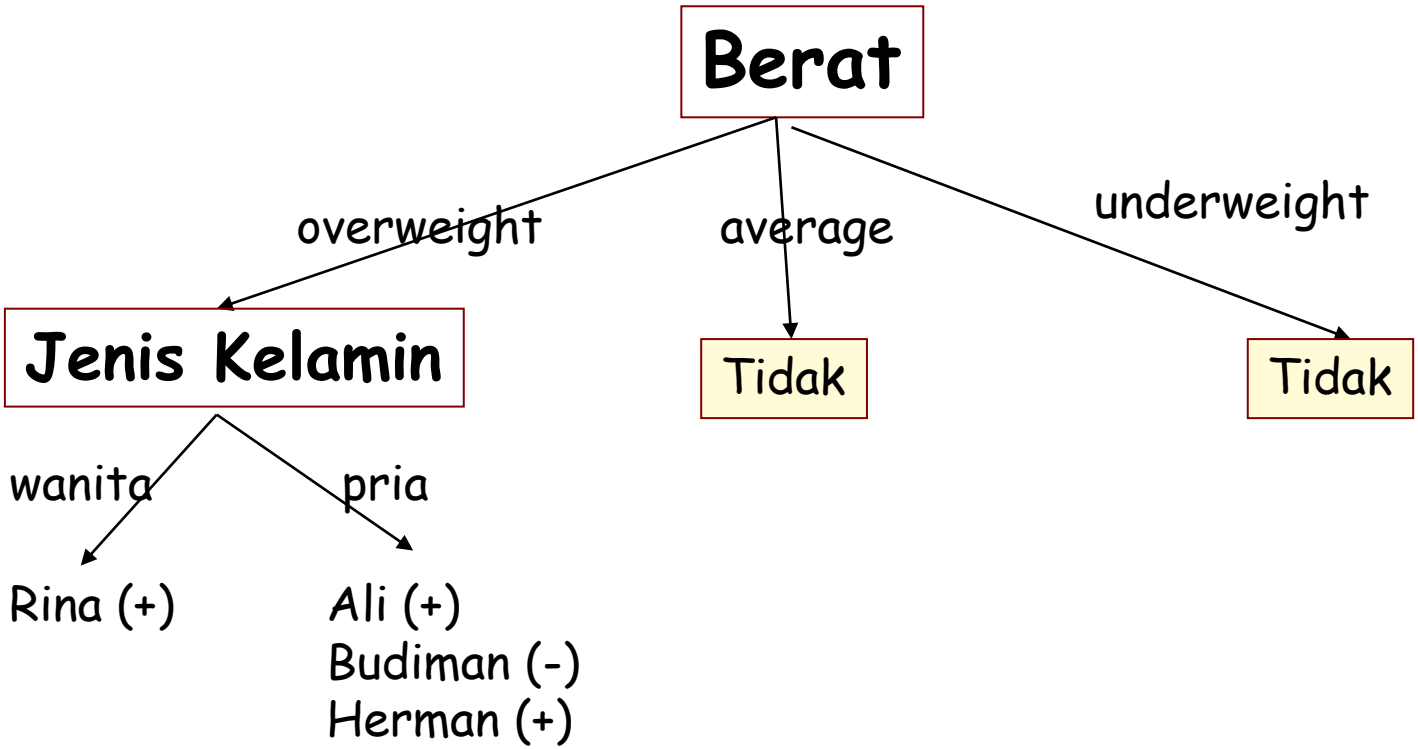
Data Training untuk berat=overweight

Nama	Usia	Kelamin	Hipertensi
Ali	muda	pria	ya
Budiman	tua	pria	tidak
Herman	tua	pria	ya
Rina	tua	wanita	ya

Usia	Hipertensi	Jumlah
muda	ya	1
	tidak	0
tua	ya	2
	tidak	1
Entropy =		0,69

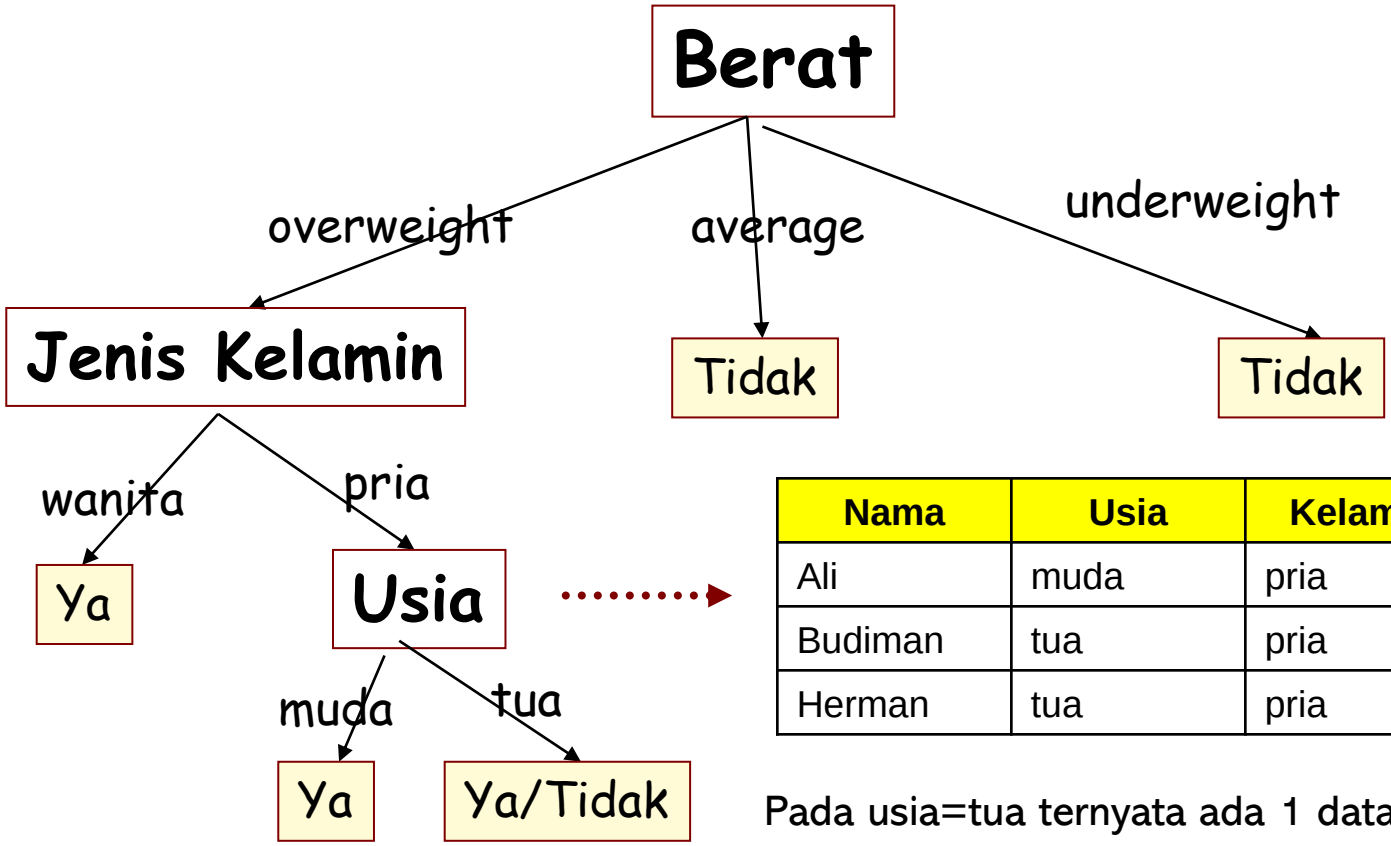
Kelamin	Hipertensi	Jumlah
pria	ya	2
	tidak	1
wanita	ya	1
	tidak	0
Entropy =		0,69

Penyusunan Tree



Leaf Node Usia dan Jenis Kelamin memiliki Entropy yang sama, sehingga tidak ada cara lain selain menggunakan pengetahuan pakar atau percaya saja pada hasil acak.

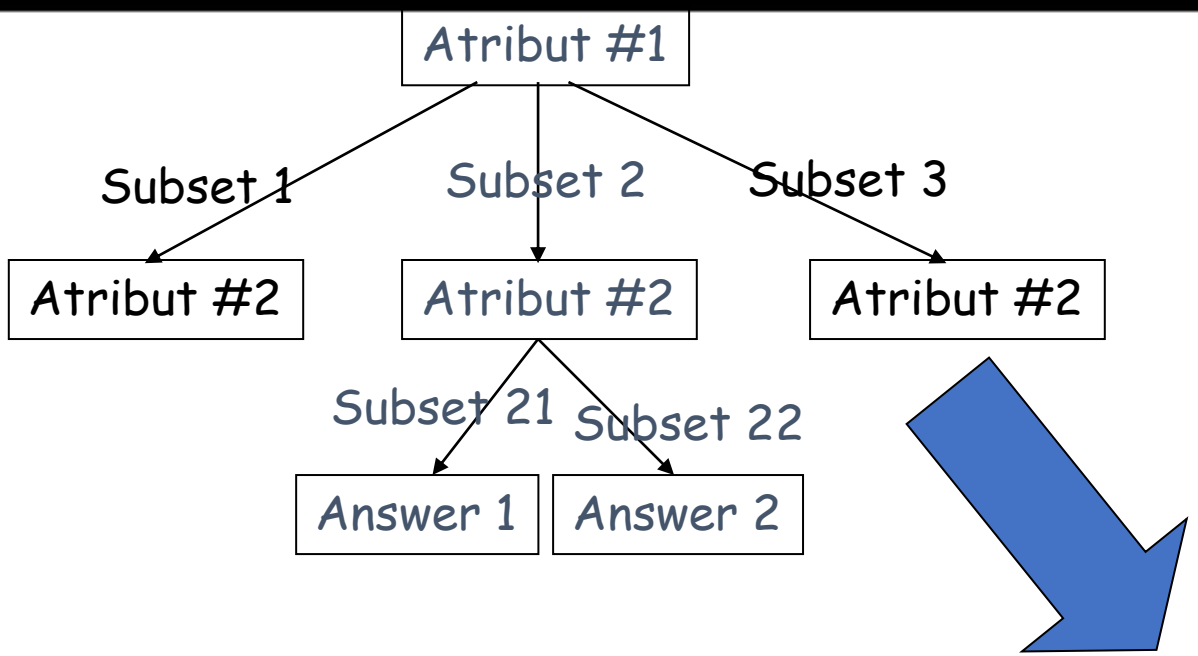
Hasil Tree



Nama	Usia	Kelamin	Hipertensi
Ali	muda	pria	ya
Budiman	tua	pria	tidak
Herman	tua	pria	ya

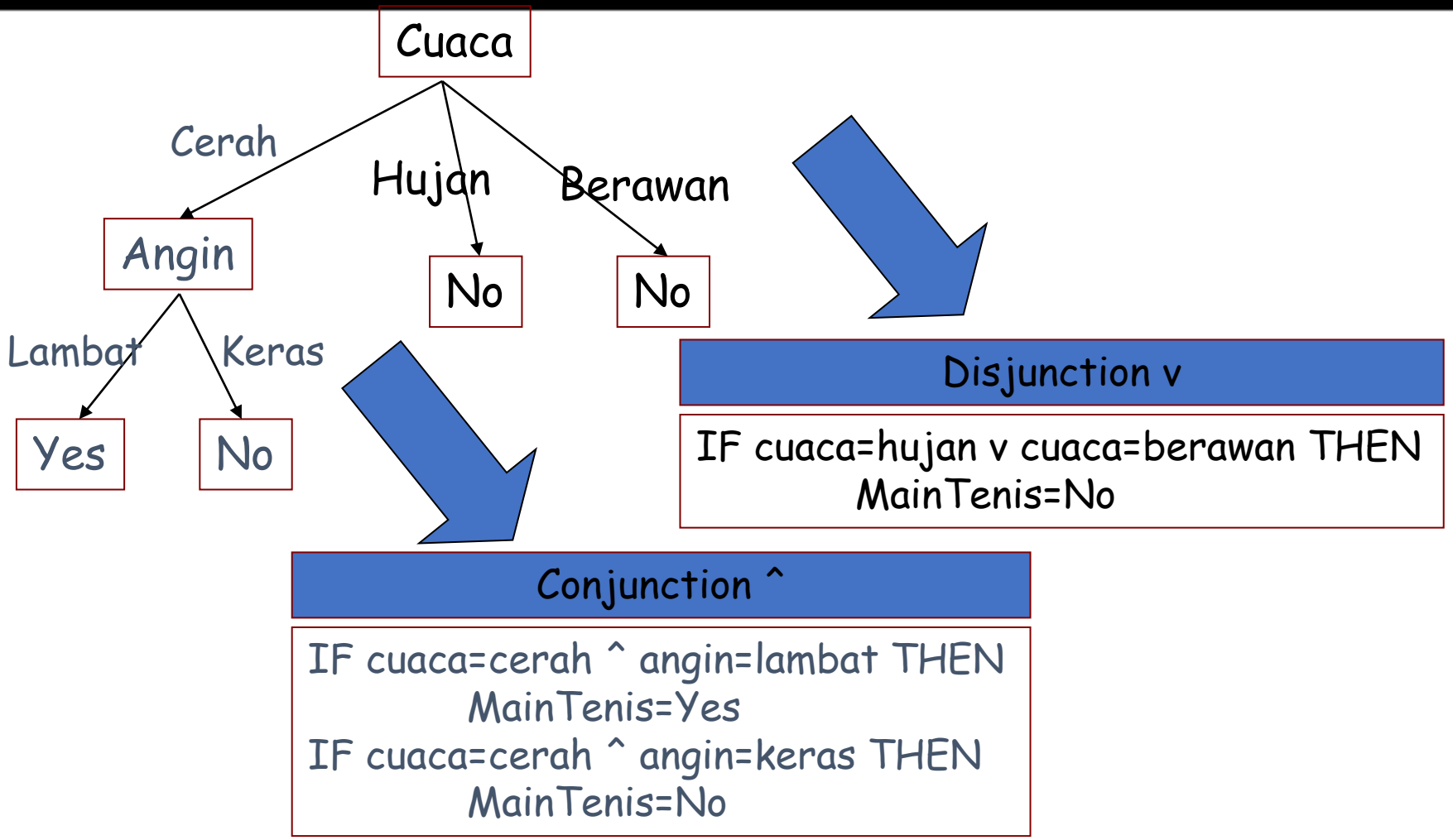
Pada usia=tua ternyata ada 1 data menyatakan ya dan 1 data menyatakan tidak, keadaan ini perlu dicermati. Pilihan hanya dapat ditentukan dengan campur tangan seoranng pakar.

Mengubah Tree Menjadi Rules

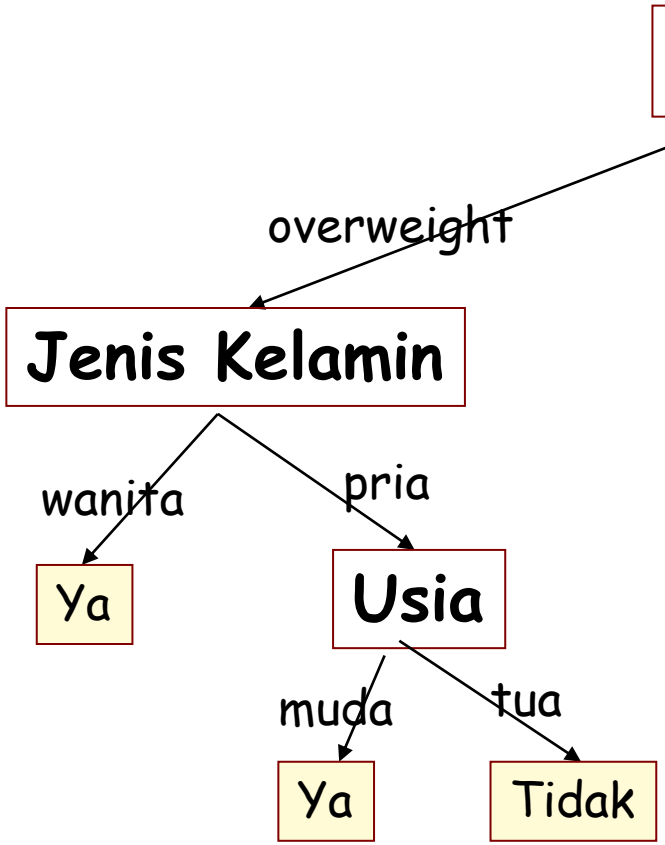


If atribut#1=subset2 ^ atribut#2=subset21
then answer=answer1
If atribut#1=subset2 ^ atribut#2=subset22
then answer=answer2

Conjunction & Disjunction



Mengubah Tree Menjadi Rule



R1: IF berat=average v berat=underweight
THEN hipertensi=tidak
R2: IF berat=overweight^kelamin=wanita
THEN hipertensi=ya
R3: IF berat=overweigt^kelamin=pria^
usia=muda THEN hipertensi=ya
R4: IF berat=overweigt^kelamin=pria^
usia=tua THEN hipertensi=tidak

Hasil Prediksi Pada Data Training

Nama	Usia	Berat	Kelamin	Hipertensi	Prediksi
Ali	muda	overweight	pria	ya	ya
Edi	muda	underweight	pria	tidak	tidak
Annie	muda	average	wanita	tidak	tidak
Budiman	tua	overweight	pria	tidak	tidak
Herman	tua	overweight	pria	ya	tidak
Didi	muda	underweight	pria	tidak	tidak
Rina	tua	overweight	wanita	ya	ya
Gatot	tua	average	pria	tidak	tidak

Kesalahan (e) = 12.5 %
(1 dari 8 data)

Decision Tree

- Diciptakan oleh Ross Quinlan
- ID3, C4.5, C5.0
- Model direpresentasikan dalam bentuk tree

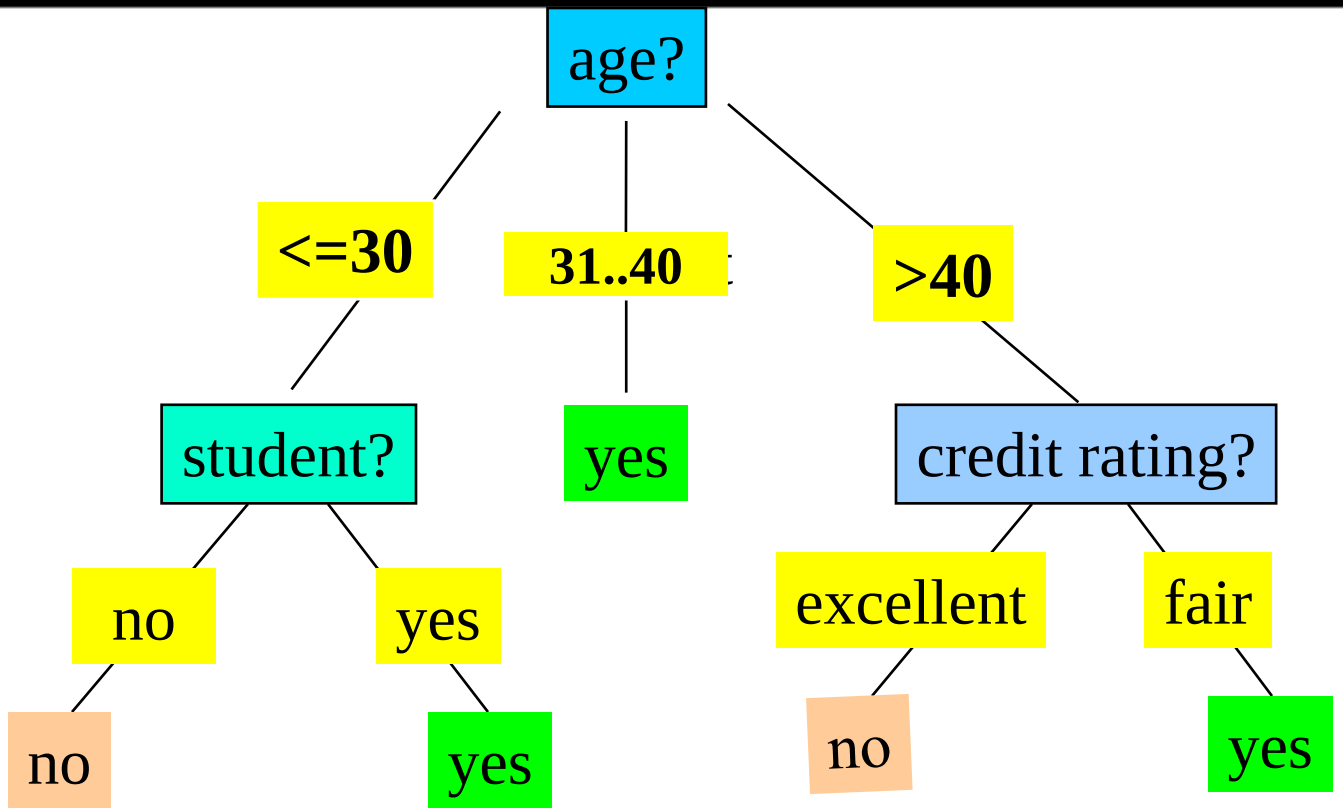
Decision Tree: Contoh Input (Data Latih)

age	income	student	credit_rating	buys_computer
<=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

Masalah

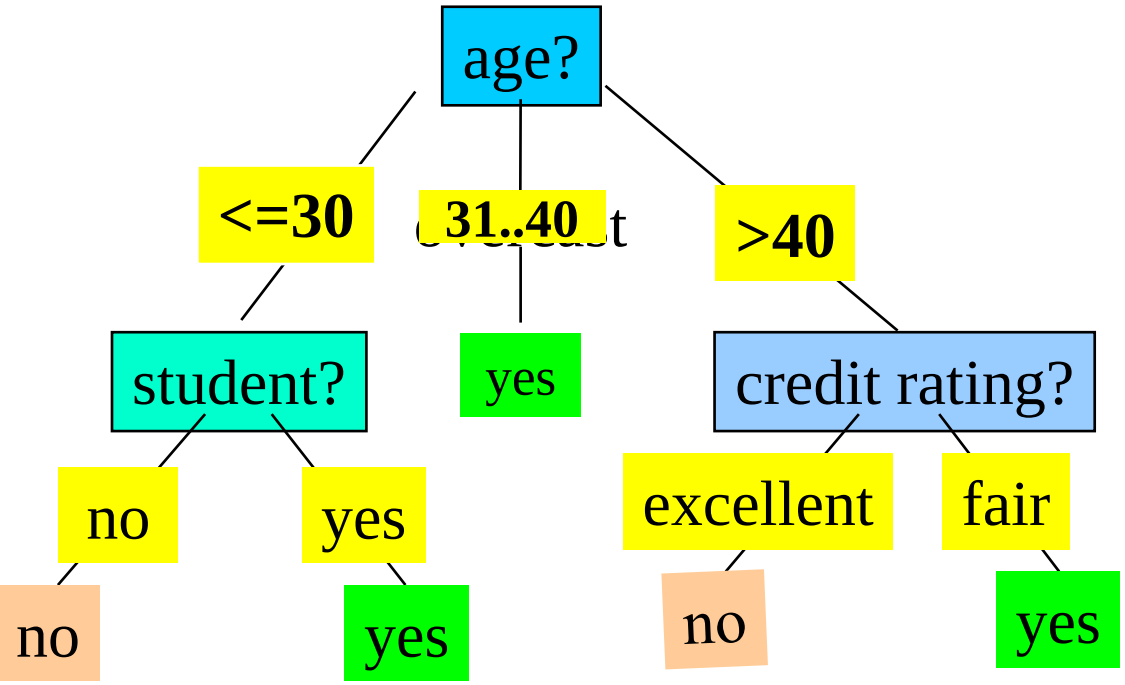
- Bagaimana dari data latih tersebut dapat diperoleh model yang bisa mengklasifikasikan secara otomatis?

Model: Decision Tree



Dari data latih, model ini dibangkitkan secara otomatis…

Tree Dapat Direpresentasikan sebagai Rule



IF ((age<=30) and
(student))

OR

(age=31..40)

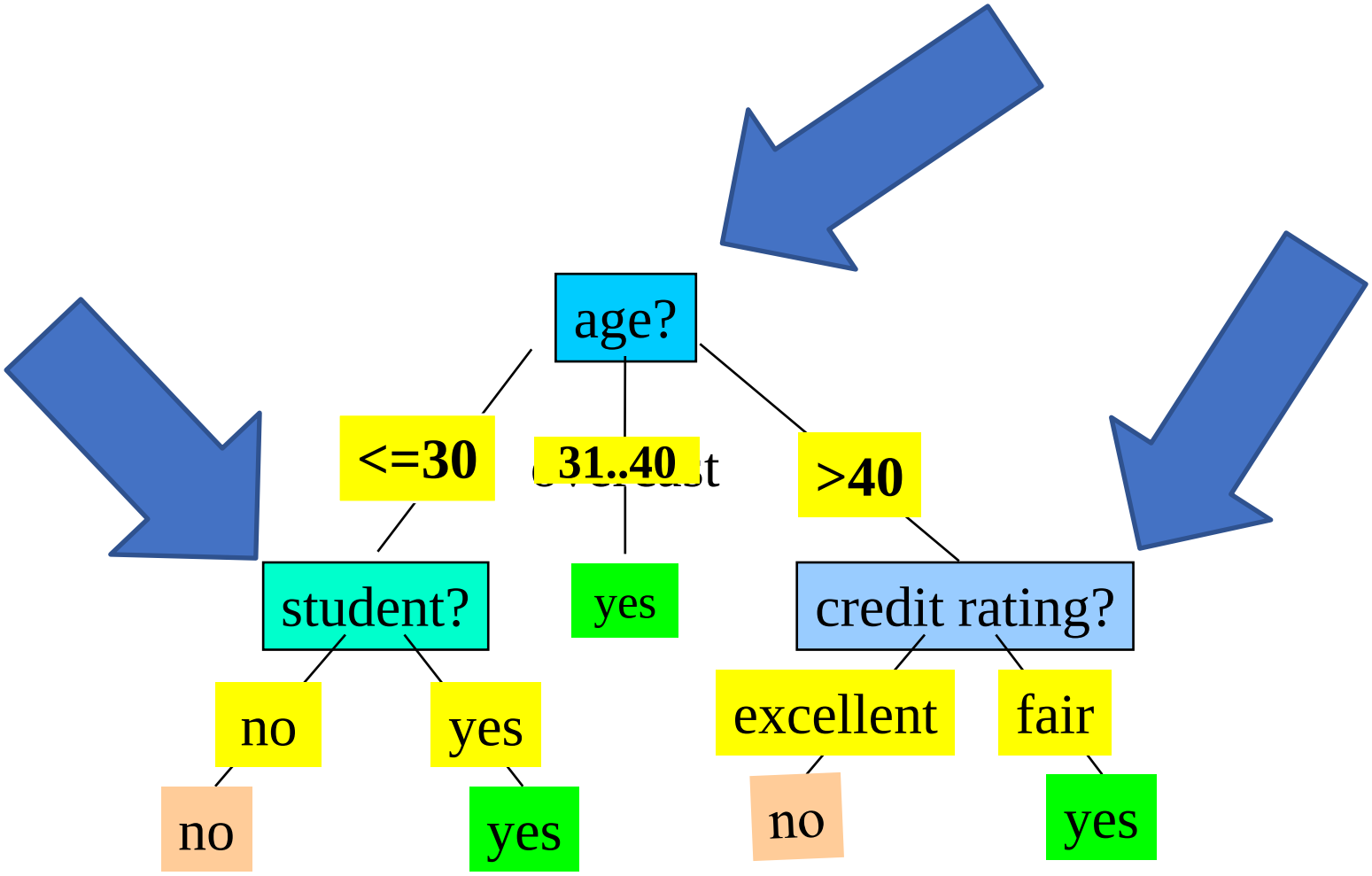
OR

(age>40) and
(credit_rating=fair)

THEN

BELI_PC=YES

Bagaimana cara pemilihan urutan atribut?



Cara Pemilihan Atribut

- Entrophy: Ukuran kemurnian, semakin murni, semakin homogen, semakin rendah nilainya.
- Information Gain: pengurangan entropy disebabkan oleh partisi berdasarkan suatu atribut.

  Semakin besar info gain = atribut itu semakin membuat homogen = semakin bagus

  Idenya → pilih atribut dengan info gain yg paling besar

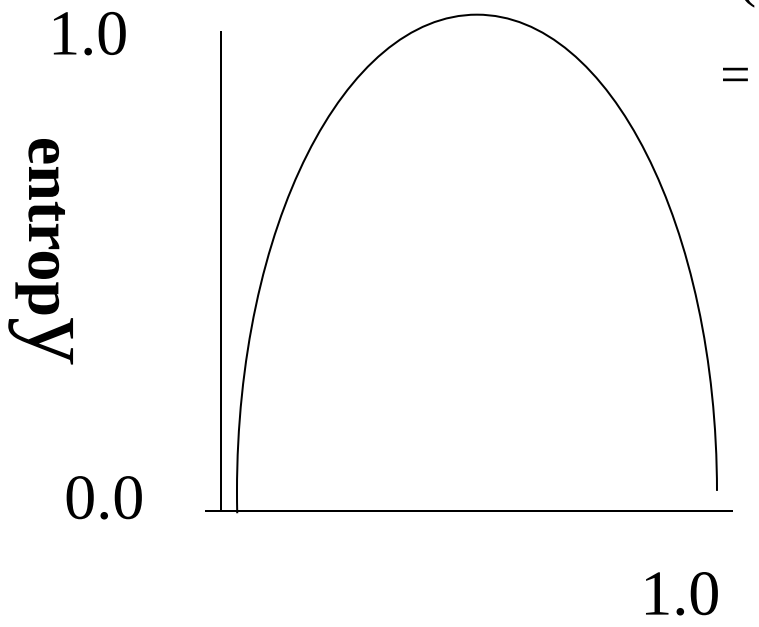
Entrophy untuk dua kelas: + dan -

$$Entropy(S) \equiv -p_{\oplus} \log_2 p - p_{\Theta} \log_2 p_{\Theta}$$

Entropy([9+,5-] ((9 positif, 5 neg)) =

$-(9/14) \log_2(9/14) - (5/14) \log_2(5/14)$

$= 0.940$



Entropy([9+,5-]) = 0.940

Entropy([7+,7-]) = 1

Entropy([14+,0]) = 0

Entroy([0+,14-]) = 0

Entrophy untuk kelas > 2

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

Info (D) = Entrophy (D) (istilah dibuku J. HAN)

Information Gain

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times I(D_j)$$

$$Gain(A) = Info(D) - Info_A(D)$$

Gain(A) seberapa besar entropy berkurang akibat atribut A. Makin besar makin bagus.

Contoh Pemilihan Atribut

- Class P: buys_computer = “yes”
- Class N: buys_computer = “no”

$$Info(D) = I(9,5) = -\frac{9}{14}\log_2(\frac{9}{14}) - \frac{5}{14}\log_2(\frac{5}{14}) = 0.940$$

age	p_i	n_i	$I(p_i, n_i)$
<=30	2	3	0.971
31…40	4	0	0
>40	3	2	0.971

age	income	student	credit_rating	buys_computer
<=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

$$Info_{age}(D) = \frac{5}{14}I(2,3) + \frac{4}{14}I(4,0) + \frac{5}{14}I(3,2) = 0.694$$

$\frac{5}{14}I(2,3)$ berarti ada 5 dari 14 “age <=30” dgn 2 yes dan 3 no.

Gain (Age) = Info(D) – Info age (D)
=0.940 – 0.694 = 0.246

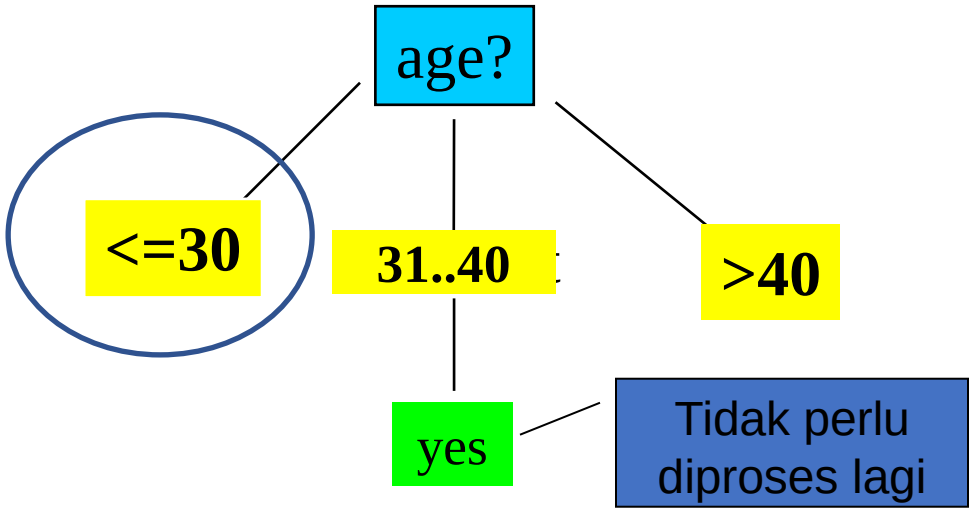
$$Gain(income) = 0.029$$
$$Gain(student) = 0.151$$
$$Gain(credit_rating) = 0.048$$

Pemilihan Atribut (lanj)

Gain (Age) = 0.246 ← yang terbesar, dipilih
Gain (income)=0.029
Gain(student)=0.151
Gain(credit_rating) =0.048

Setelah AGE, atribut apa selanjutnya?

Diproses untuk setiap cabang selama masih ada > 1 kelas



Selanjutnya... proses data yang <=30

Pemilihan Atribut (lanj)

Selanjutnya... proses data age<=30

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
<=30	medium	yes	excellent	yes

$$Info(D) = I(2,3) = -\frac{2}{5}\log_2(\frac{2}{5}) - \frac{3}{5}\log_2(\frac{3}{5}) = 0.97$$

Gain(age) tidak perlu dihitung lagi, hitung gain(student), gain(credit_rating)

$$Info_{student}(D) = \frac{3}{5}I(0,3) + \frac{2}{5}I(2,0) = 0$$

Gain (student) = Info(D) – Info$_{student}$(D)
=0.97 – 0 = 0.97

Pemilihan Atribut (lanj)

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
<=30	medium	yes	excellent	yes

hitung gain(credit_rating)

$$Info(D) = I(2,3) = -\frac{2}{5}\log_2(\frac{2}{5}) - \frac{3}{5}\log_2(\frac{3}{5}) = 0.97$$

$$Info_{credit_rating}(D) = \frac{3}{5}I(1,2) + \frac{2}{5}I(1,1) = 0.95$$

Gain (credit_rating) = Info(D) – Info$_{student}$(D)
=0.97 – 0.95 = 0.02

$$Info_{income}(D) = \frac{2}{5}I(0,2) + \frac{2}{5}I(1,1) + \frac{1}{5}I(1,0) = 0.4$$

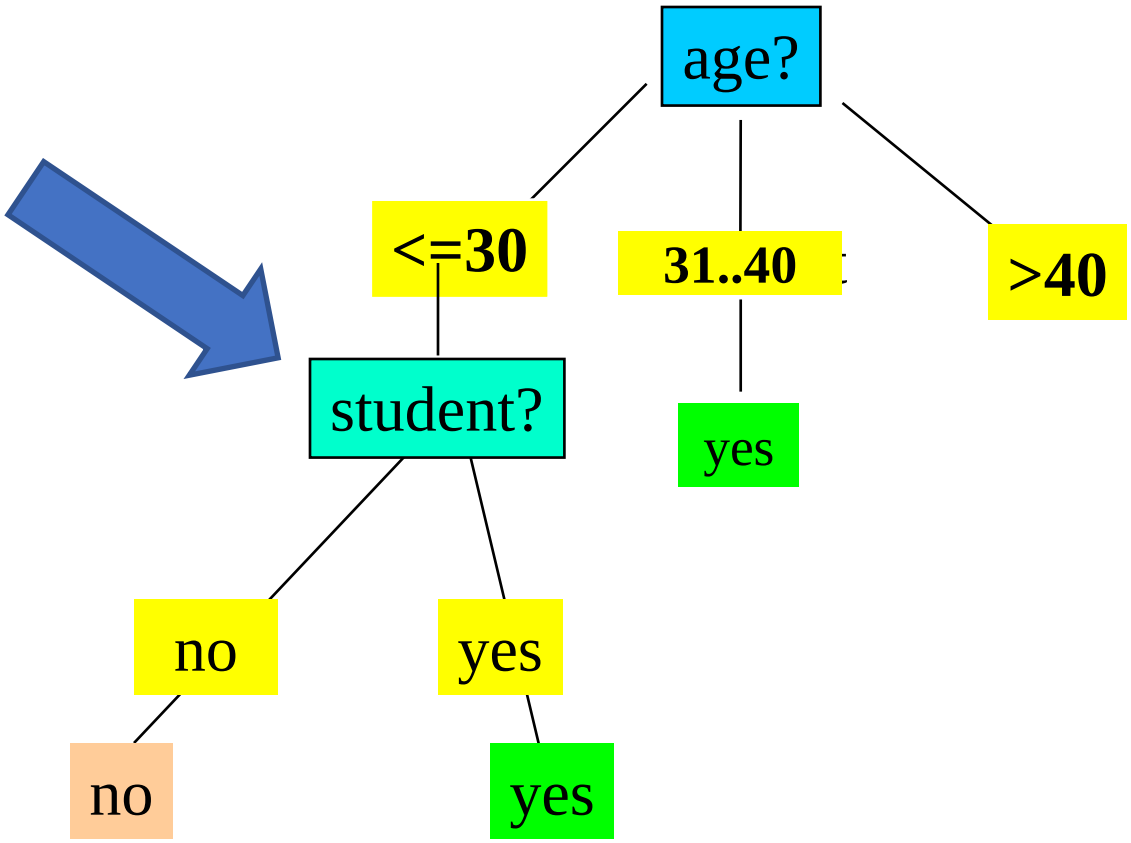
Pilihan Atribut (lanj)

Bandingkan semua gain, ambil yang paling besar

Gain (studet) = 0.97
Gain (credit_rating = 0.02
Gain (income) = 0.4

Paling besar
student

Pemilhan Atribut (lanj)



Latihan

No	Kelas	Kulit Buah	Warna	Ukuran	Bau
1	Aman	Kasar	Coklat	Besar	keras
2	Aman	Kasar	Hijau	Besar	keras
3	Berbahaya	Halus	Merah	Besar	Lunak
4	Aman	Kasar	Hijau	Besar	Lunak
5	Aman	Kasar	Merah	Kecil	Keras
6	Aman	Halus	Merah	Kecil	Keras
7	Aman	Halus	Coklat	Kecil	Keras
8	Berbahaya	Kasar	Hijau	Kecil	Lunak
9	Berbahaya	Halus	Hijau	Kecil	Keras
10	Aman	Kasar	Merah	Besar	Keras
11	Aman	Halus	Coklat	Besar	Lunak
12	Berbahaya	Halus	Hijau	Kecil	Keras
13	Aman	Kasar	Merah	Kecil	Lunak
14	Berbahaya	Halus	Merah	Besar	Keras
15	Aman	Halus	Merah	Kecil	Keras
16	Berbahaya	Kasar	Hijau	Kecil	Keras

Mengapa Decision Tree?

- Mudah diimplementasikan
- Hipotesis yang dihasilkan mudah dipahami
- Efisien

Decision Tree Cocok untuk Masalah:

- Data dalam bentuk atribut-nilai. Kondisi ideal adalah jika isi nilai jumlahnya sedikit. Misalnya: “panas”, “sedang”, “dingin”.
- Output diskrit.
- Training data dapat tidak lengkap

Masalah DT

- Overfitting: terlalu mengikuti training data
  - Terlalu banyak cabang, merefleksikan anomali akibat noise atau outlier.
  - Akurasi rendah untuk data baru
- Dua pendekatan untuk menghindari overfitting
  - Prepruning: Hentikan pembuatan tree di awal. Tidak mensplit node jika goodness measure dibawah threshold.
    - Sulit untuk menentukan threshold
  - Postpruning: Buang cabang setelah tree jadi
    - Menggunakan data yang berbeda dengan training untuk menentukan pruned tree yang terbaik.

Supervised vs. Unsupervised Learning

- Supervised learning (classification)
  - Supervision: Data pelatihan mengandung label kelas.
  - Data diklasifikasikan menggunakan model.
- Unsupervised learning (clustering)
  - Data pelatihan tidak mengandung label kelas
  - Mencari kelas atau cluster di dalam data.
  - Akan dijelaskan terpisah

SELESAI