Pohon Klasifikasi

Nama Lain

- Classification Tree
- Decision Tree
- Recursive Partition
- Iterative Dichotomiser

bagusco

Kegunaan

- Mengidentifikasi variabel apa yang dapat dijadikan sebagai pembeda antar kelompok
- Memprediksi keanggotaan kelompok suatu individu berdasarkan karakteristiknya

- Terapannya antara lain:
  - Marketing: Mengidentifikasi prospective customer (cross-sell, up-sell, new acquisition)
  - Risk: Credit scoring, menentukan apakah calon penerima kredit akan mampu bayar atau tidak
  - Customer Relationship: churn analysis, menentukan customer yang berpotensi akan meninggalkan jasa/produk
  - Health: menentukan tingkat resiko penyakit
  - dll

Metode lain yang setara kegunaannya

- Regresi Logistik
- Discriminant Analysis
- Support Vector Machine
- Bayesian Classifier
- Neural Network
- dll

bagusco

Outline

- Pengenalan Konsep Entropy dan Information Gain
- Pengenalan Algoritma Dasar Pohon Klasifikasi
- Menilai Kemampuan Prediksi Pohon Klasifikasi
- Pengembangan Lebih Lanjut dari Pohon Klasifikasi

1

Entropy dan Information Gain

Entropy dan Information Gain

- Andaikan sebuah gugus data D berisi individu-individu dengan dua kelas yaitu kelas YES dan NO, dengan proporsi yang YES sebesar p, dan tentusaja $(1 - p)$ lainnya tergolong kelas NO.

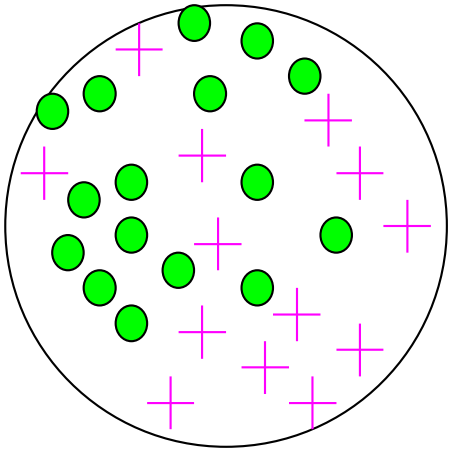
- Entropi dari gugus data tersebut adalah

  $E(D) = -p \log_2(p) - (1-p) \log_2(1-p)$

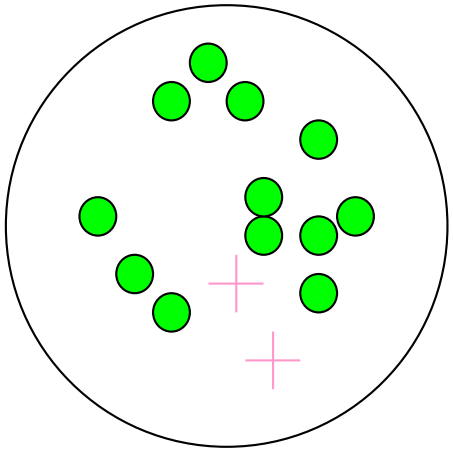
- Gugus data yang seluruh amatannya dari kelas YES akan memiliki $E(D) = 0$
- Gugus data yang seluruh amatannya dari kelas NO juga akan memiliki $E(D) = 0$
- Entropi ini adalah ukuran kehomogenan data (impurity)

Entropy dan Information Gain

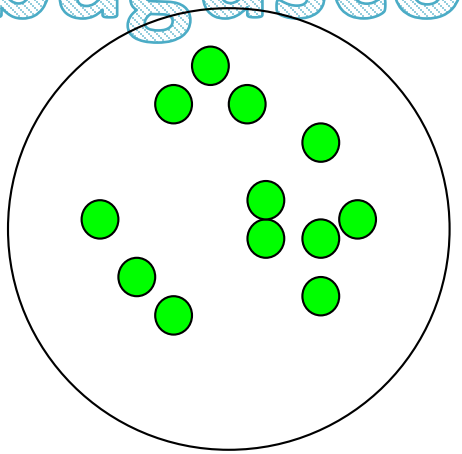
Very impure group



Less impure



Minimum impurity



Entropy dan Information Gain

- Andaikan sebuah gugus data D dibagi menjadi beberapa kelompok, misalnya D_1, D_2, …, D_k berdasarkan variabel prediktor V

- Dari setiap D_i bisa dihitung entropinya, yaitu $E(D_i)$

- Information Gain adalah

$$IG(D,V) = E(D) - \sum_{i=1}^{k} \frac{|D_i|}{|D|} E(D_i)$$

- Variabel prediktor yang kuat hubungannya akan memiliki information gain yang semakin besar

Entropy dan Information Gain

Frequency Percent Row Pct Col Pct	Table of Jenis_Kelamin by Tertarik_Beli		
Jenis_Kelamin(Jenis Kelamin)	Tertarik_Beli(Tertarik Beli)		
	tidak	tertarik	Total
perempuan	561	27	588
	51.75	2.49	54.24
	95.41	4.59	
	74.80	8.08	
laki-laki	189	307	496
	17.44	28.32	45.76
	38.10	61.90	
	25.20	91.92	
Total	750	334	1084
	69.19	30.81	100.00

E(TOTAL) = $-p \log_2(p) - (1-p) \log_2(1-p)$
= $-0.3081 \log_2(0.3081) - 0.6919 \log_2(0.6919)$
= 0.8910

E(Perempuan) = $-p \log_2(p) - (1-p) \log_2(1-p)$
= $-0.0459 \log_2(0.0459) - 0.9541 \log_2(0.9541)$
= 0.2688

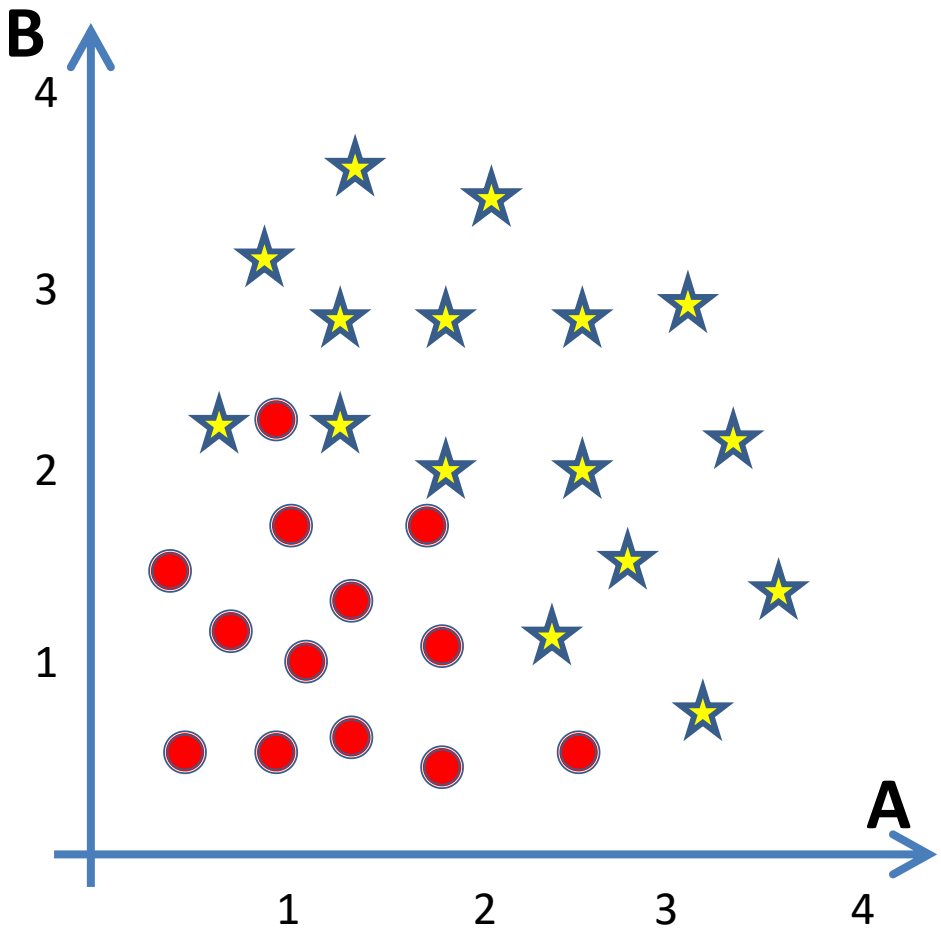
E(Laki-Laki) = $-p \log_2(p) - (1-p) \log_2(1-p)$
= $-0.6190 \log_2(0.6190) - 0.3810 \log_2(0.3810)$
= 0.9588

Information Gain dari Variabel Jenis Kelamin
IG = 0.8910 – (588/1084 * 0.2688 + 496/1084 * 0.9588)
= 0.8910 – 0.5845
= 0.3065

2

Pohon Klasifikasi

Ide Dasar



16 obs

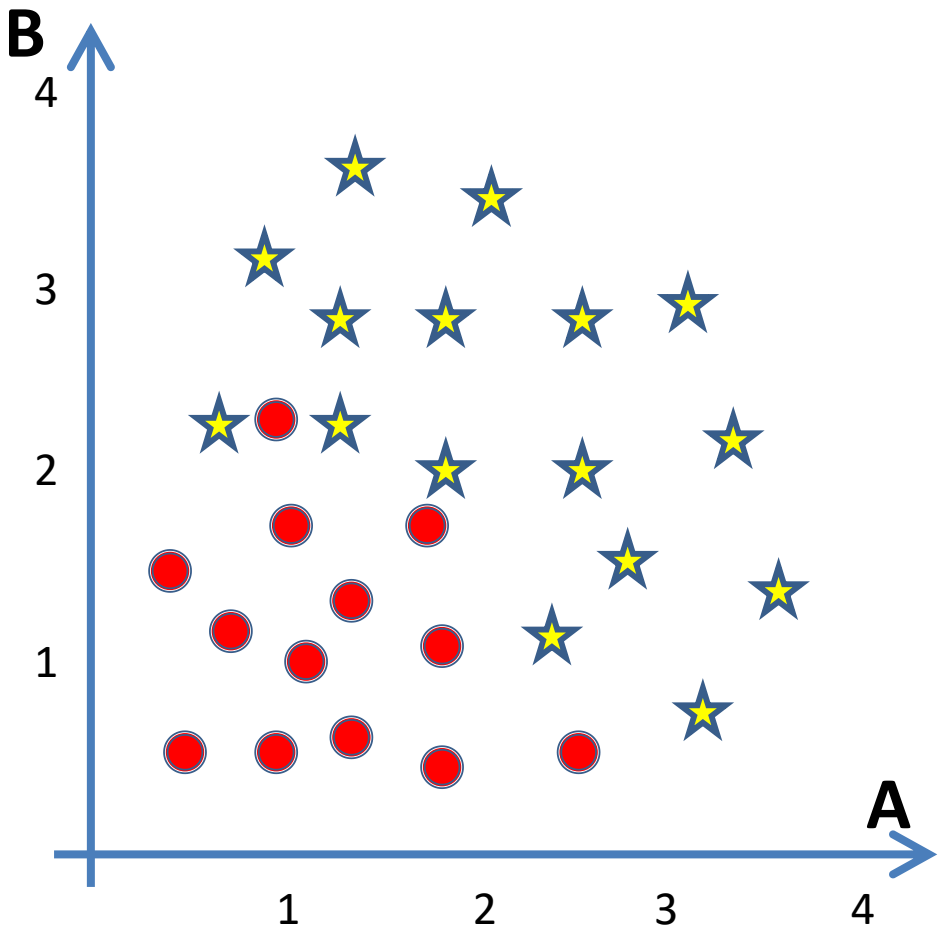
13 obs

bagusco

Mencari pemisah terbaik antara individu ★ dengan individu ●

Pemisahan dilakukan untuk masing-masing variabel, bukan kombinasinya.

Ide Dasar

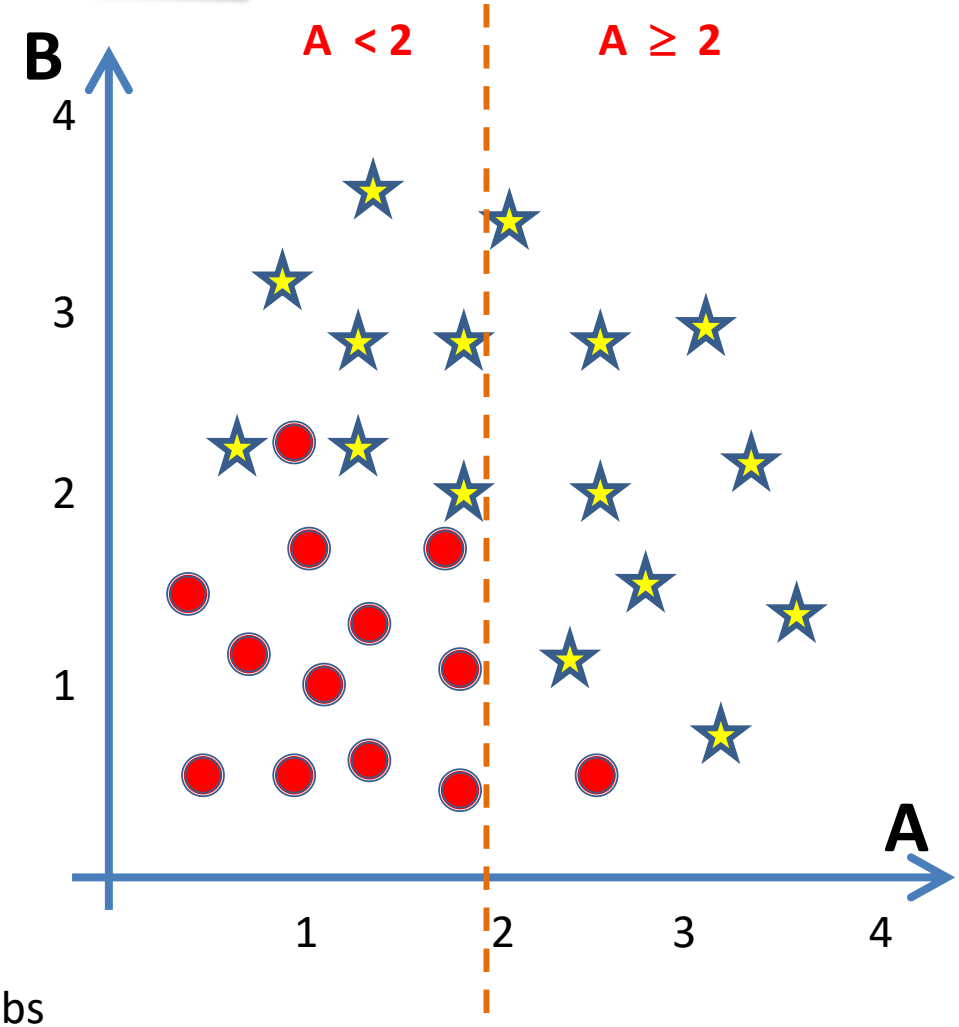


16 obs

13 obs

Pemisah yang dicari adalah yang menyebabkan data hasil pemisahannya bersifat homogen kelasnya.

Ide Dasar



Pemisahan menggunakan garis A = 2, menghasilkan dua kelompok:

Kelompok 1 A < 2

7 obs

12 obs

Kelompok 2 A ≥ 2

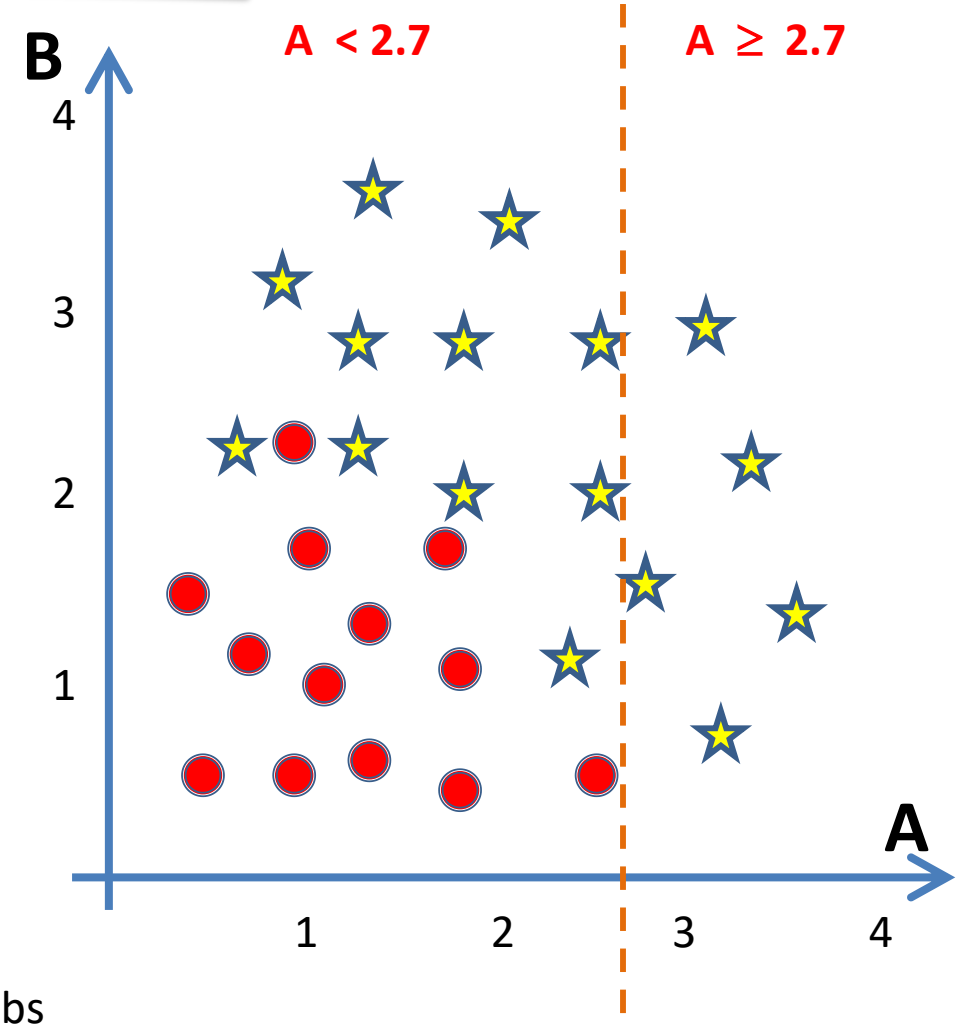
9 obs

1 obs

16 obs

13 obs

Ide Dasar



Pemisahan menggunakan garis A = 2.7, menghasilkan dua kelompok:

Kelompok 1 A < 2.7

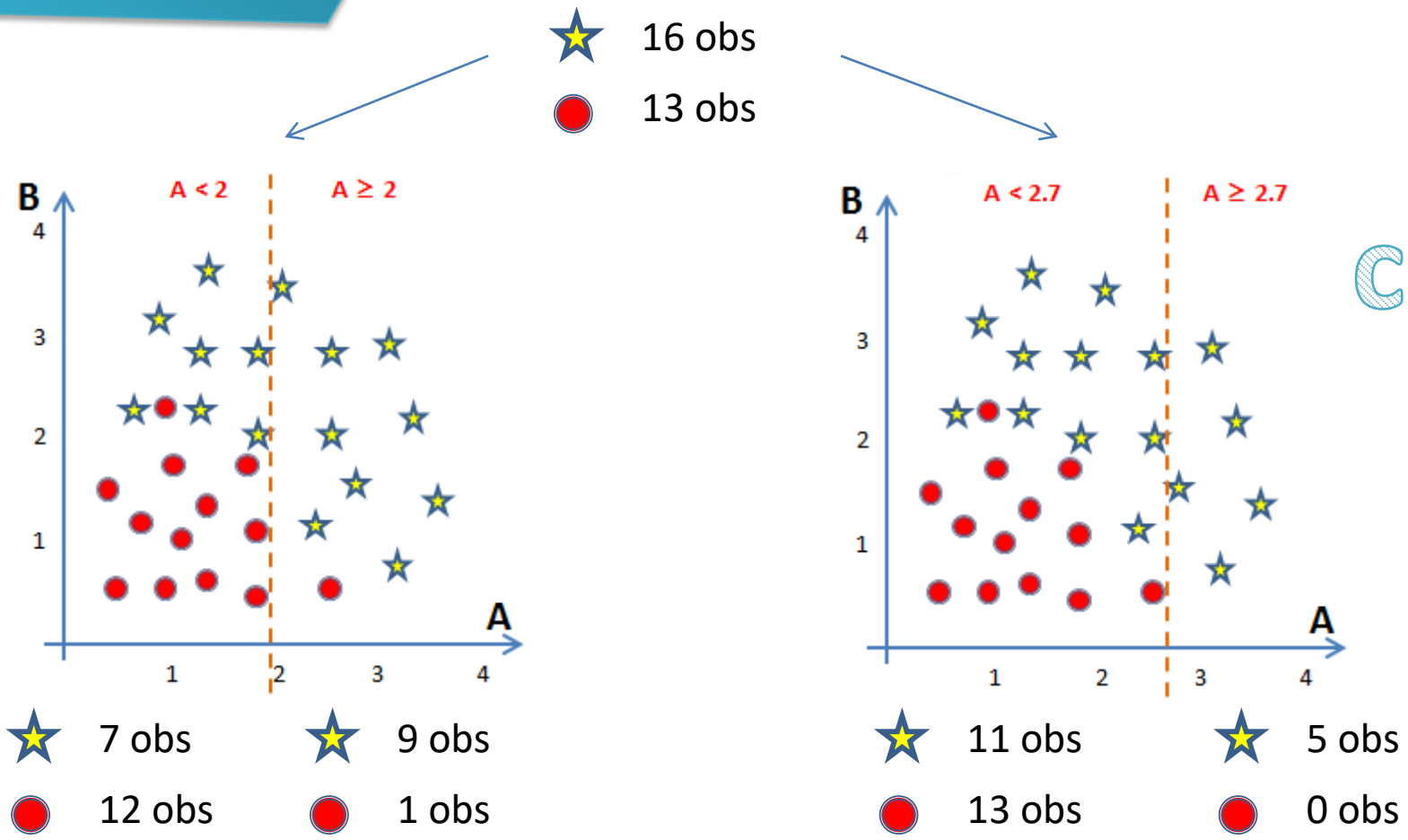
- 11 obs
- 13 obs

Kelompok 2 A ≥ 2.7

- 5 obs
- 0 obs

- 16 obs
- 13 obs

Ide Dasar



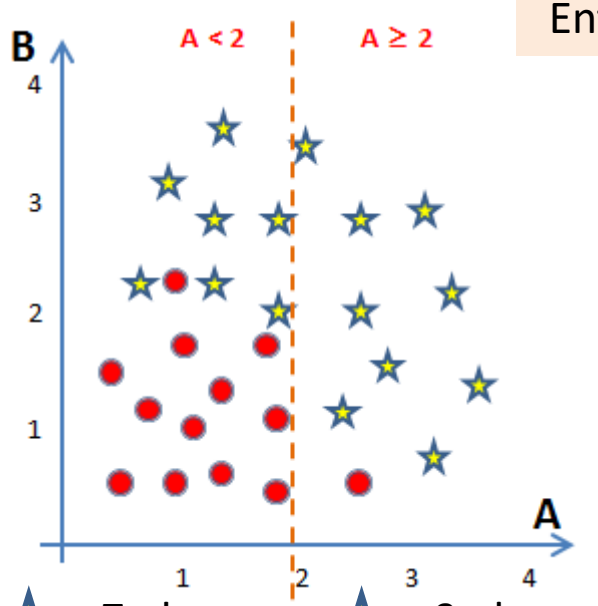
Mana yang lebih baik?

Ide Dasar

16 obs

13 obs

Entropy = 0.9923



7 obs

12 obs

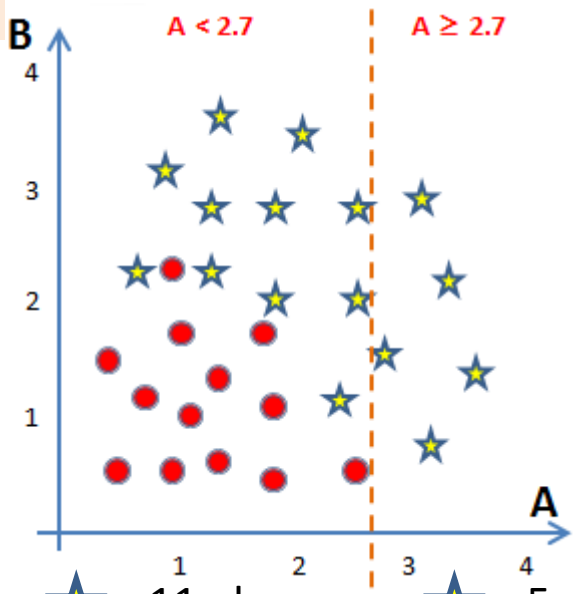
E = 0.9495

9 obs

1 obs

E = 0.4690

Information Gain = 0.2085



11 obs

13 obs

E = 0.9950

5 obs

0 obs

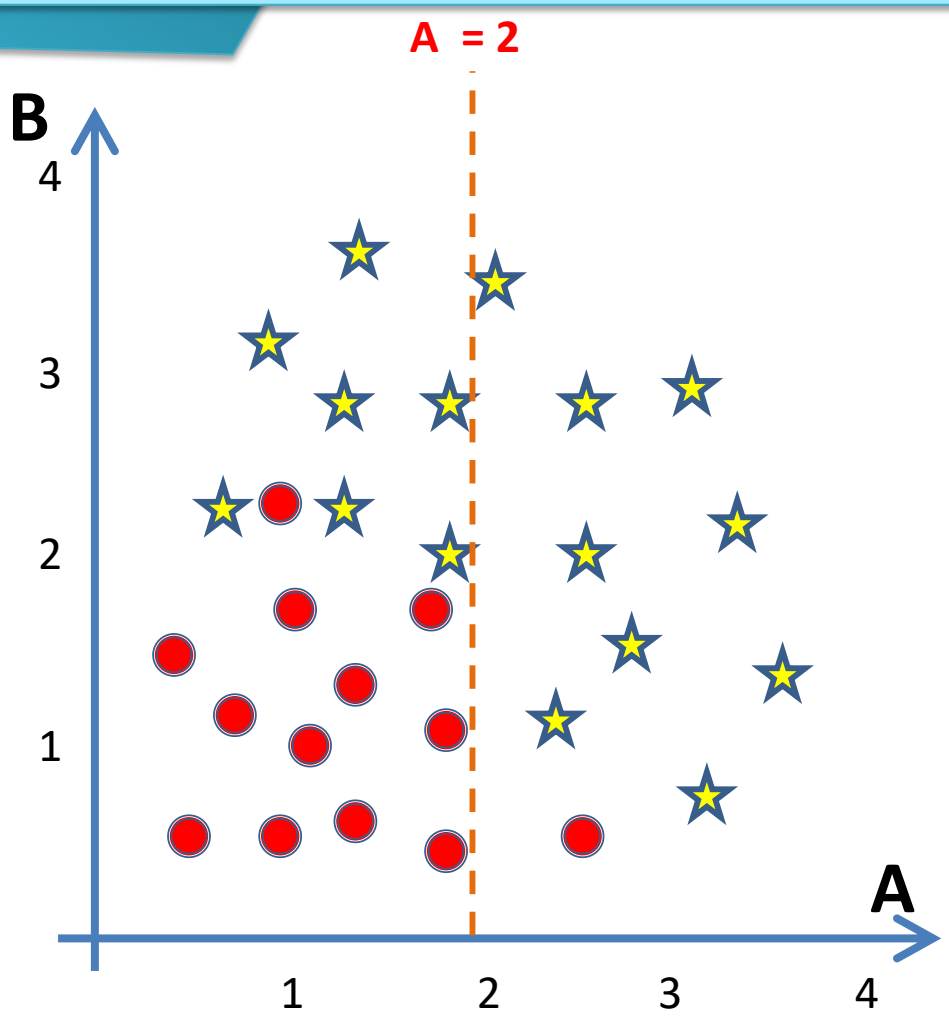
E = 0

Information Gain = 0.1688

CO

Ide Dasar

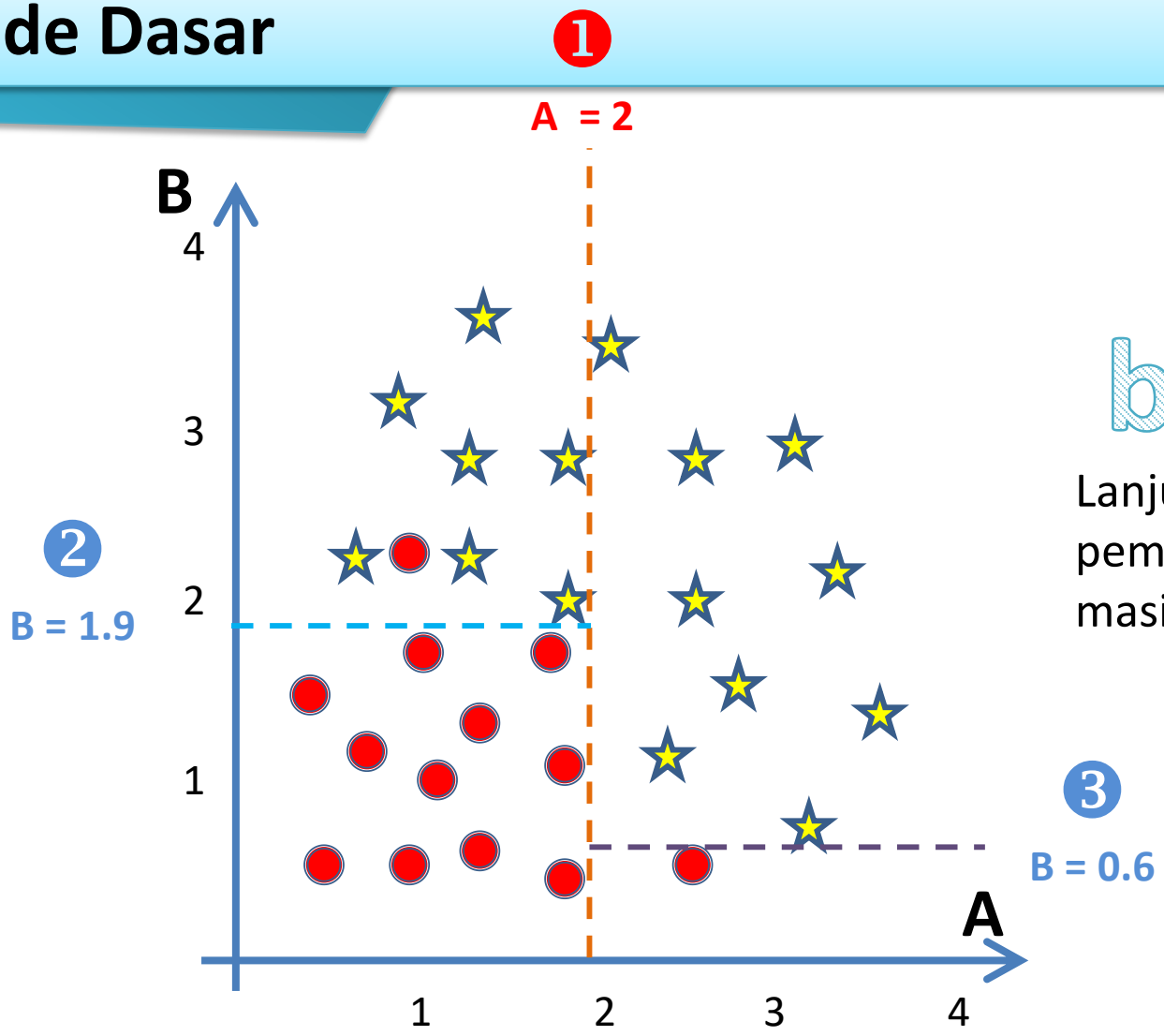
❶



bagusco

Lanjutkan mencari pemisahan untuk masing-masing kelompok….

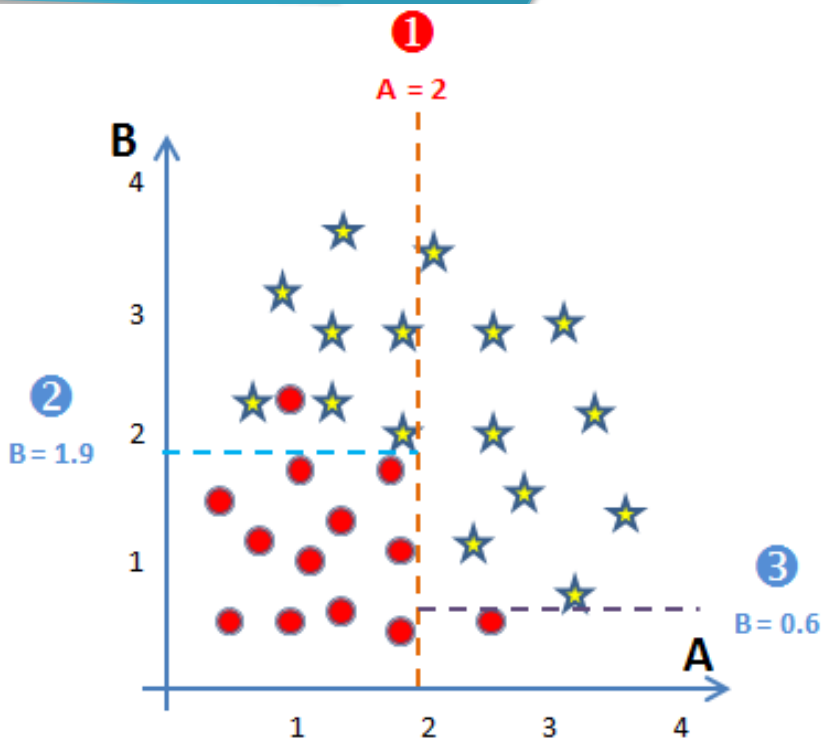
Ide Dasar

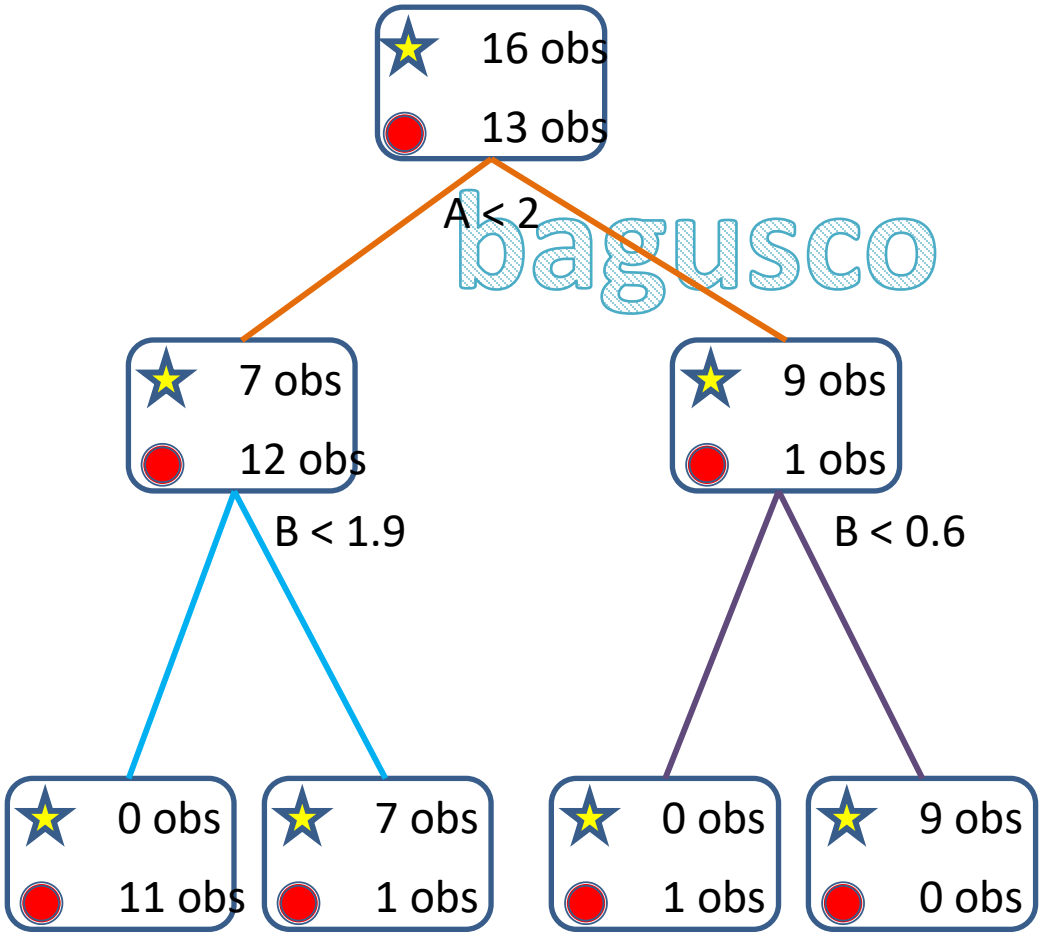


bagusco

Lanjutkan mencari pemisahan untuk masing-masing kelompok….

Representasi Hasil Pemisahan



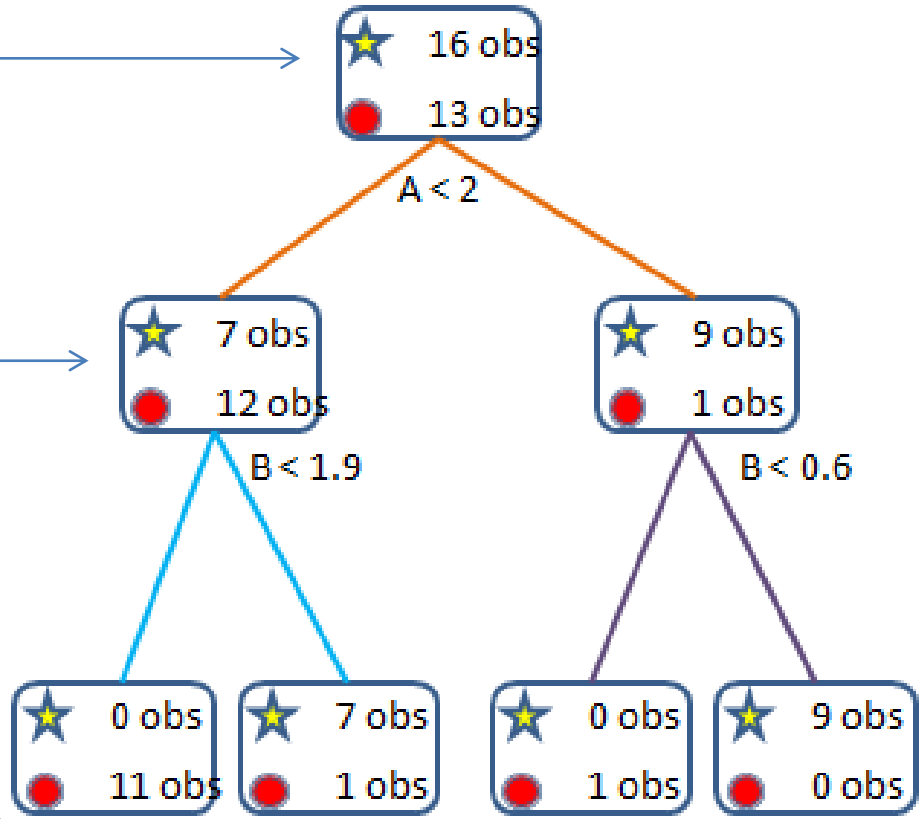


Beberapa Istilah

root
(simpul akar)

node
(simpul)

leaf
(simpul akhir)



Level 0

Level 1

depth

Level 2

Algoritma Dasar Pohon Klasifikasi

- Tahap 1:
  Mencari pemisahan/penyekatan (splitting) terbaik di setiap variabel

bagusco

- Tahap 2:
  Menentukan variabel terbaik untuk penyekatan

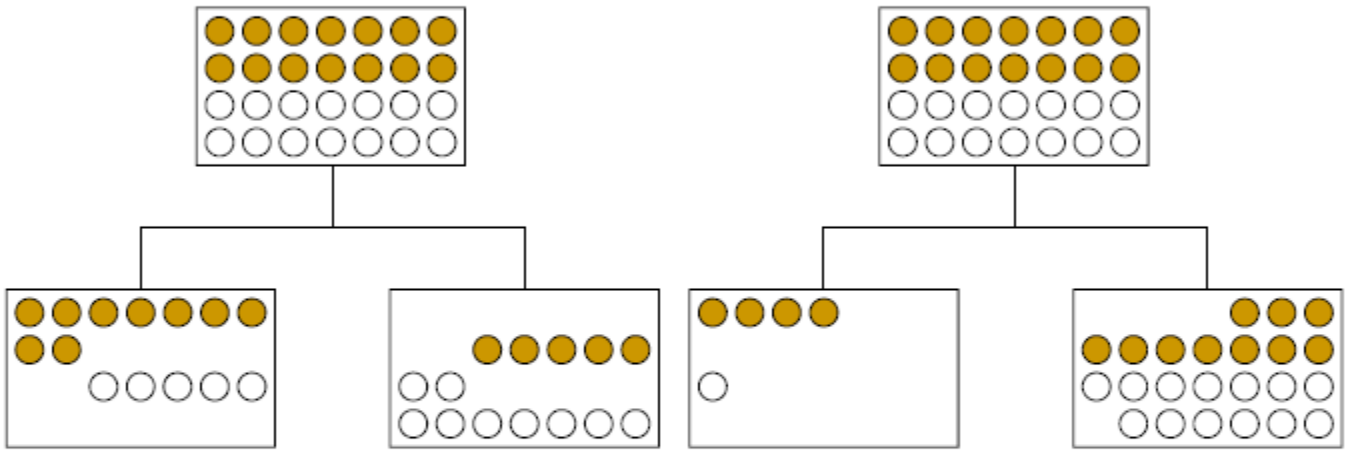
- Tahap 3:
  Melakukan penyekatan berdasarkan hasil dari Tahap 2, dan memeriksa apakah sudah waktunya menghentikan proses

Lakukan tiga tahapan di atas untuk setiap simpul dan hasil sekatannya

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Which Is the Better Split?

A or B



Two children: Both are 64% pure.

The sizes are exactly the same.

Two children: One is 80% pure and the other 57%.

However, the sizes are quite different.

18

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Gini: Easy Measure to Explain

Gini, used in the social sciences and economics, is the probability that two things chosen at random from a population will be the same (a measure of purity).

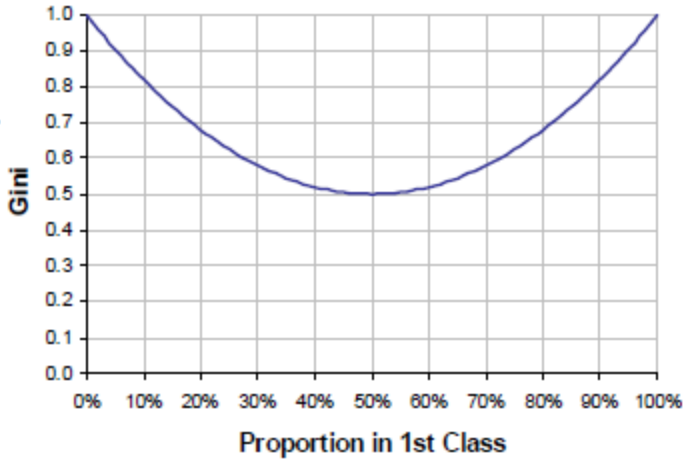
A pure population has a Gini index of 1.

If there are two groups equally represented, then the Gini index is 0.5.

The Gini index is the sum of the square of the proportions:

$$p_1^2 + p_2^2$$

The goal is to maximize Gini.



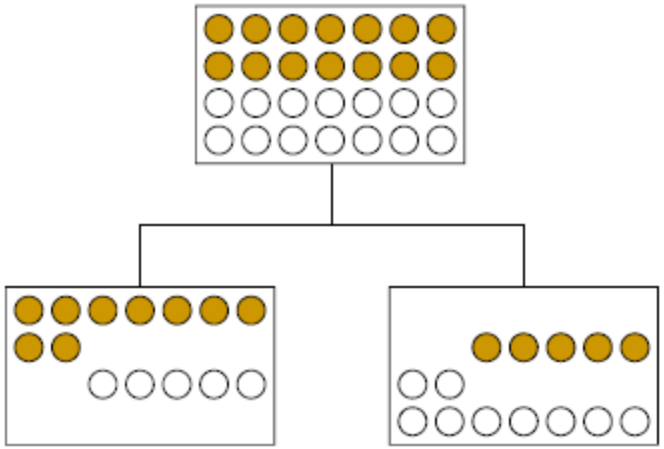
19

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Gini for Candidate Split A

The Gini score is the weighted sum of the Gini index of each child (weighted by the size of the split).

Gini for the root node is 0.5 ($0.5^2+0.5^2$).



Gini score for either child

$(5/14)^2 + (9/14)^2 =$
$0.128 + 0.413 = 0.541$

Evaluate the split with the weighted average of Gini values for all children (easy in this case).

$0.5*0.541 + 0.5*0.541 =$
0.541

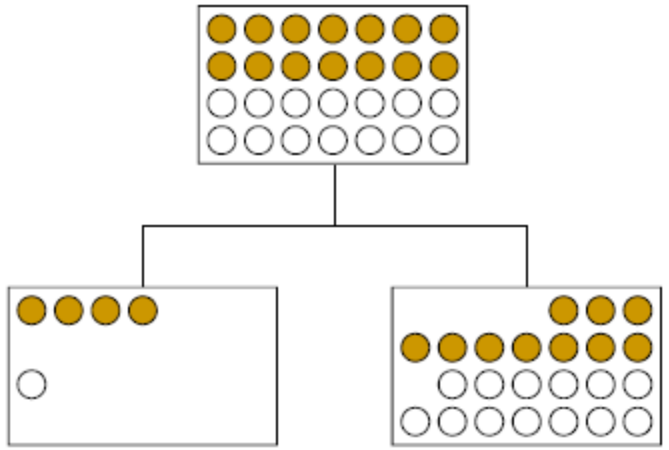
20

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Comparing the Splits with Gini

Candidate split B:

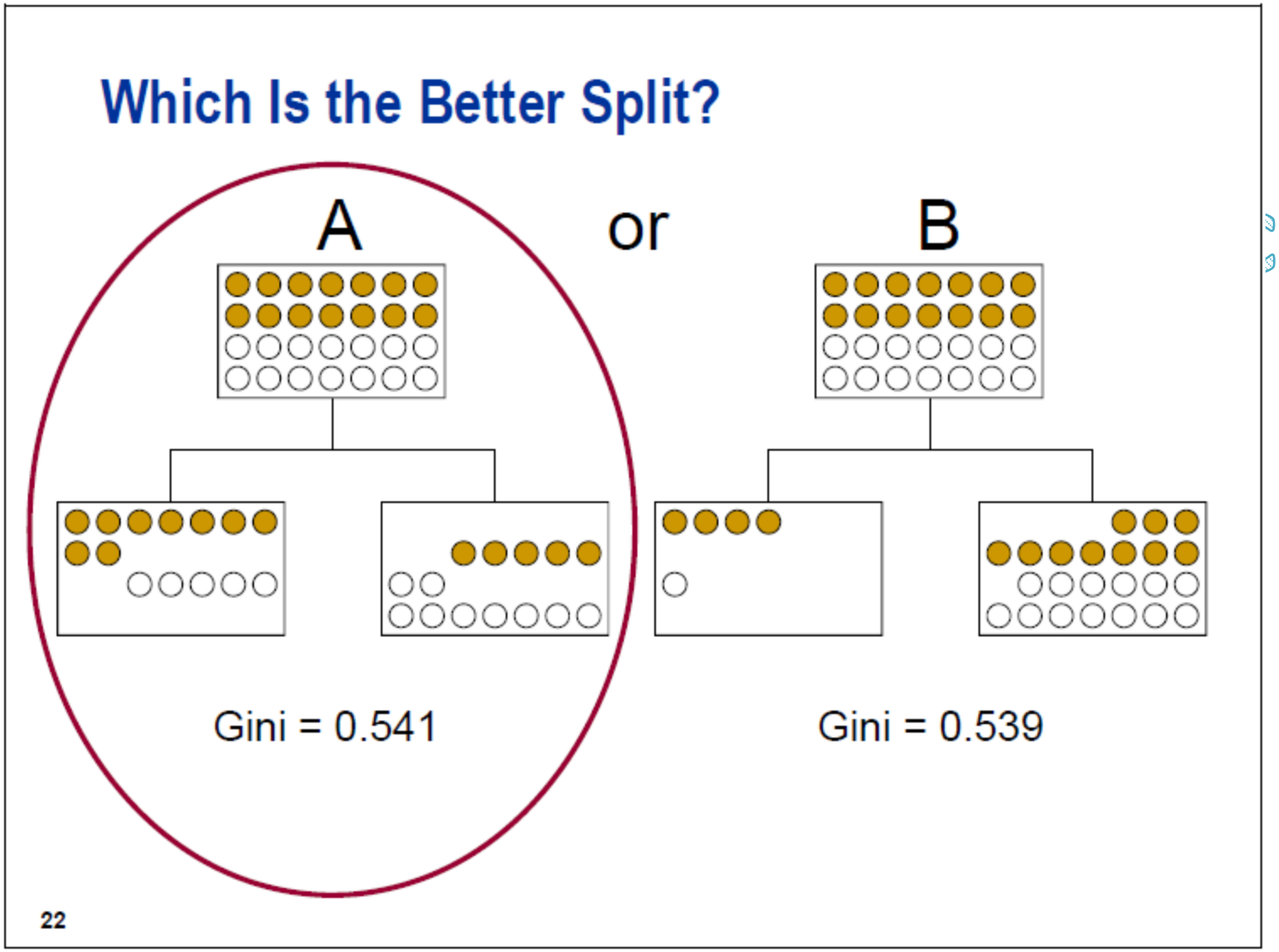
- Gini for the left child is $(1/5)^2 + (4/5)^2 =$ $0.04 + 0.64 = 0.68.$
- Gini for the right child is $(10/23)^2 + (13/23)^2 =$ $0.189 + 0.319 = 0.508.$



- Gini for the split is the weighted average: $(5/28)*Gini_{left} + (23/28)*Gini_{right} =$ 0.539

21

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq



Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Entropy: More Difficult Measure to Explain

Entropy is used in information theory to measure the amount of information stored in a given number of bits.

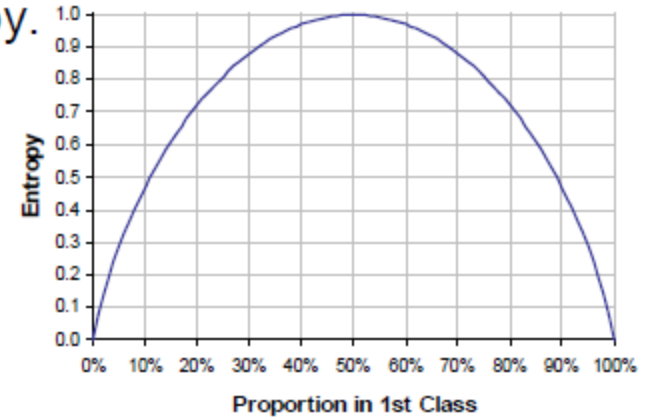
A pure population has an entropy of 0.

If there are two groups equally represented, then the entropy is 1.

The calculation for entropy is shown here:

$$-1*(p_1 \log_2 (p_1) + p_2 \log_2 (p_2)).$$

The goal is to minimize entropy.



23

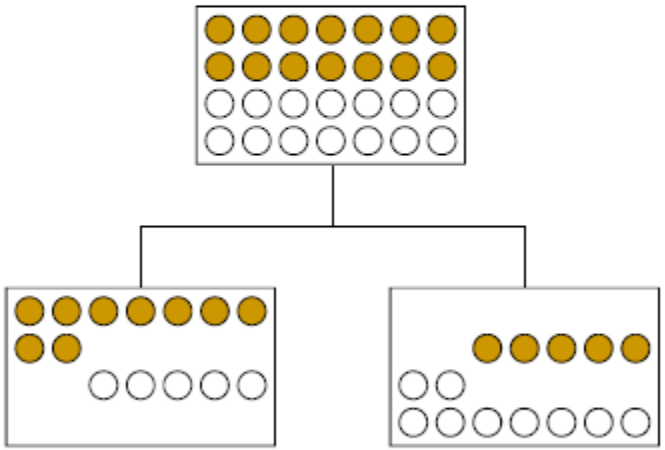
28

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Entropy for Candidate Split A

The entropy is the weighted sum of the entropy of each child (weighted by the size of the split).

Entropy for the root node is 1 (-(0.5*log(0.5) + 0.5*log(0.5))).



Entropy value for either child:

-((5/14)log(5/14) + (9/14) log(9/14)) =
-(-0.5305 + -0.4098) = 0.9403

Evaluate the split with the weighted average of entropy values for all children:

0.5*0.9403 + 0.5*0.9403 = 0.9403

Information gain is
1 – 0.9403 = 0.0597.

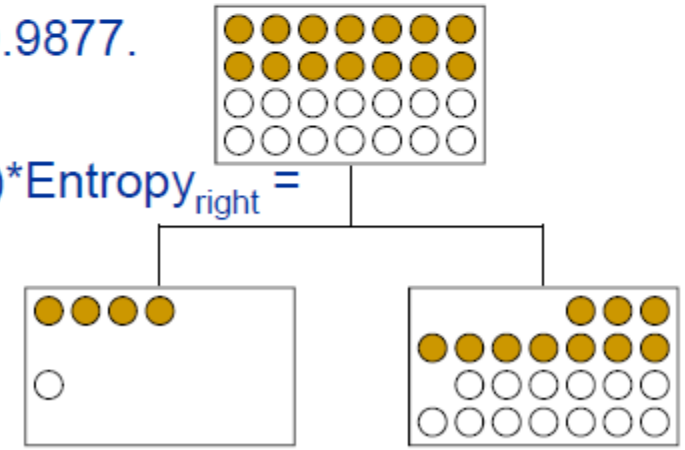
24

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Comparing the Splits with Entropy

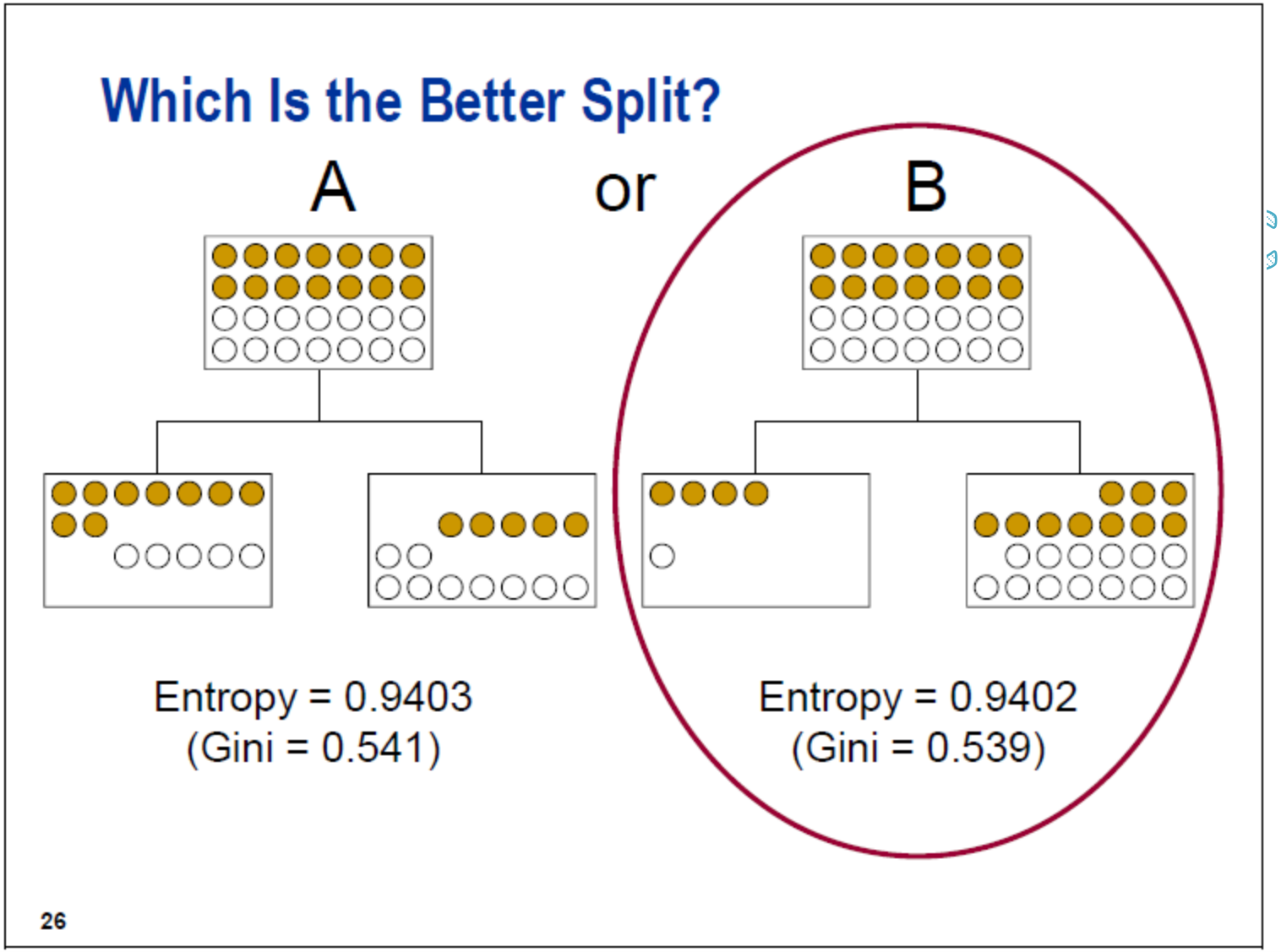
Candidate split B:

- Entropy for the left child is
  -1*((1/5)*log(1/5) + (4/5)log(4/5)) =
  -1*(-0.4644 + -0.2575) = 0.7219.
- Entropy for the right child is
  -1*((10/23)*log(10/23) + (13/23)log(13/23)) =
  -1*(-0.5225 + -0.4652) = 0.9877.
- Entropy for the split is
  $(5/28)*\text{Entropy}_{left} + (23/28)*\text{Entropy}_{right}$ =
  0.9402.



25

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq



Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Chi-Square Is from Statistics

The chi-square test is an important test in statistics to measure the probability that observed frequencies in a sample are due only to sampling variation.

Chi-square is always relative to the proportion in the original population (the parent).

If the proportions in both children are the same as the parent, then the chi-square value is 0.

If both children are pure, then the chi-square value is high. (For a 50%-50% population, the value is the population size.)

The calculation for chi-square is done for each child:

$(c_1 – \text{expected}(c_1))^2/\text{expected}(c_1) + (c_2 – \text{expected}(c_2))^2/\text{expected}(c_2)$

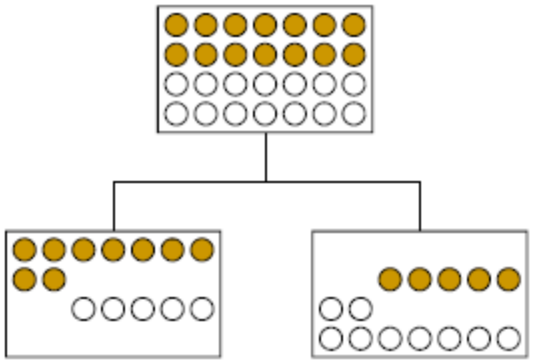
In this calculation, c_1 is the number of instances of class 1 in one child and $\text{expected}(c_1)$ is the expected number given the proportion in the parent.

The goal is to maximize chi-square.

27

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq

Chi-Square for Candidate Splits

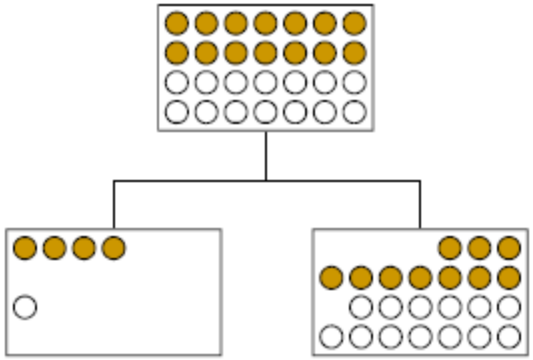


The expected value of dark or light is 7 in each child. So, the chi-square value for each child is as shown below:

$(9-7)^2/7 + (5-7)^2/7 = 4/7 + 4/7 = 1.1429$

The overall chi-square value is the sum for each child:

$1.1429 + 1.1429 = 2.2857$



The expected value of dark or light is 2.5 for the left child and 11.5 for the right child. The chi-square values are as follows:

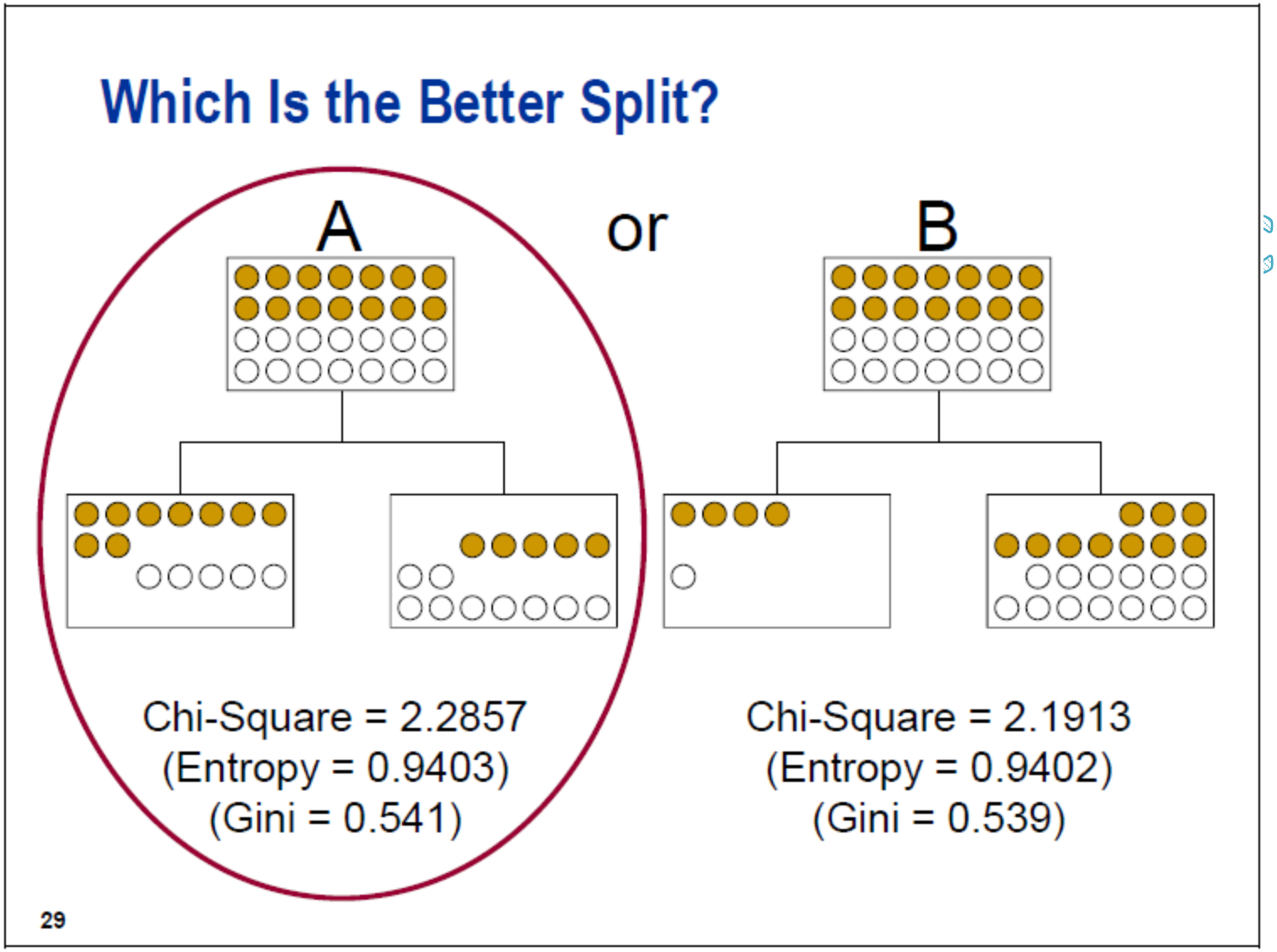
Left: $(4-2.5)^2/2.5 + (1-2.5)^2/2.5 = 2.25/2.5 + 2.25/2.5 = 1.800$

Right: $(10-11.5)^2/11.5 + (13-11.5)^2/11.5 = 2.25/11.5 + 2.25/11.5 = 0.3913$

Overall: $1.8000 + 0.3913 = 2.1913$

28

Ilustrasi lain terkait dengan Gini, Information Gain, dan Chi-Sq



Kriteria Penghentian Proses Pemisahan

- Simpul berisi amatan yang berasal dari satu kelas variabel respon
- Simpul berisi amatan yang seluruh variabel prediktornya identik
- Simpul berisi amatan yang kurang dari ukuran simpul minimal yang ditentukan di awal
- Kedalaman pohon sudah mencapai kedalaman maksimal

Ilustrasi Sederhana

- Gunakan “data tree.csv”

- Variabel :
- "No"
- "Jenis.Kelamin"
- "Single"
- "Tinggal.di.Kota"
- "usia"
- "Perokok"
- "Budget"
- "Kesukaan"
- "Tertarik.Beli."

bagusco

Ilustrasi Sederhana

```
setwd("D:/bagusco/bagusco/Kuliah S2 --- Pemodelan Klasifikasi/Genap 2017 2018")
data <- read.csv("data tree.csv")


data$tertarik <- factor(data$Tertarik.Beli., levels = c(0, 1), labels=c("tidak", "tertarik"))
data$jk <- factor(data$Jenis.Kelamin, levels=c(0,1), labels=c("p", "l"))
data$tempattinggal <- factor(data$Tinggal.di.Kota, levels = c(0,1), labels = c("desa", "kota"))
data$single <- factor(data$Single,levels = c(0,1), labels = c("Menikah", "Single"))
data$merokok <- factor(data$Perokok, levels = 0:1, labels = c("Tidak", "Ya"))
```

```
setwd("D:/bagusco/bagusco/Kuliah S2 --- Pemodelan Klasifikasi/Genap 2017 2018")
data <- read.csv("data tree.csv")

library(discretization)
entropy_total <- ent(data$tertarik)

entropy_lakilaki <- ent(data$tertarik[data$jk == "l"])
entropy_perempuan <- ent(data$tertarik[data$jk == "p"])
IG_jk <- entropy_total - length(data$tertarik[data$jk == "l"])*entropy_lakilaki / nrow(data) -
length(data$tertarik[data$jk == "p"])*entropy_perempuan / nrow(data)
```

bagusco

IG_jk → 0.21

```
entropy_merokok <- ent(data$tertarik[data$merokok == "Ya"])
entropy_tidakmerokok <- ent(data$tertarik[data$merokok == "Tidak"])
IG_merokok <- entropy_total - length(data$tertarik[data$merokok ==
"Ya"])*entropy_merokok  / nrow(data) - length(data$tertarik[data$merokok ==
"Tidak"])*entropy_tidakmerokok  / nrow(data)
```

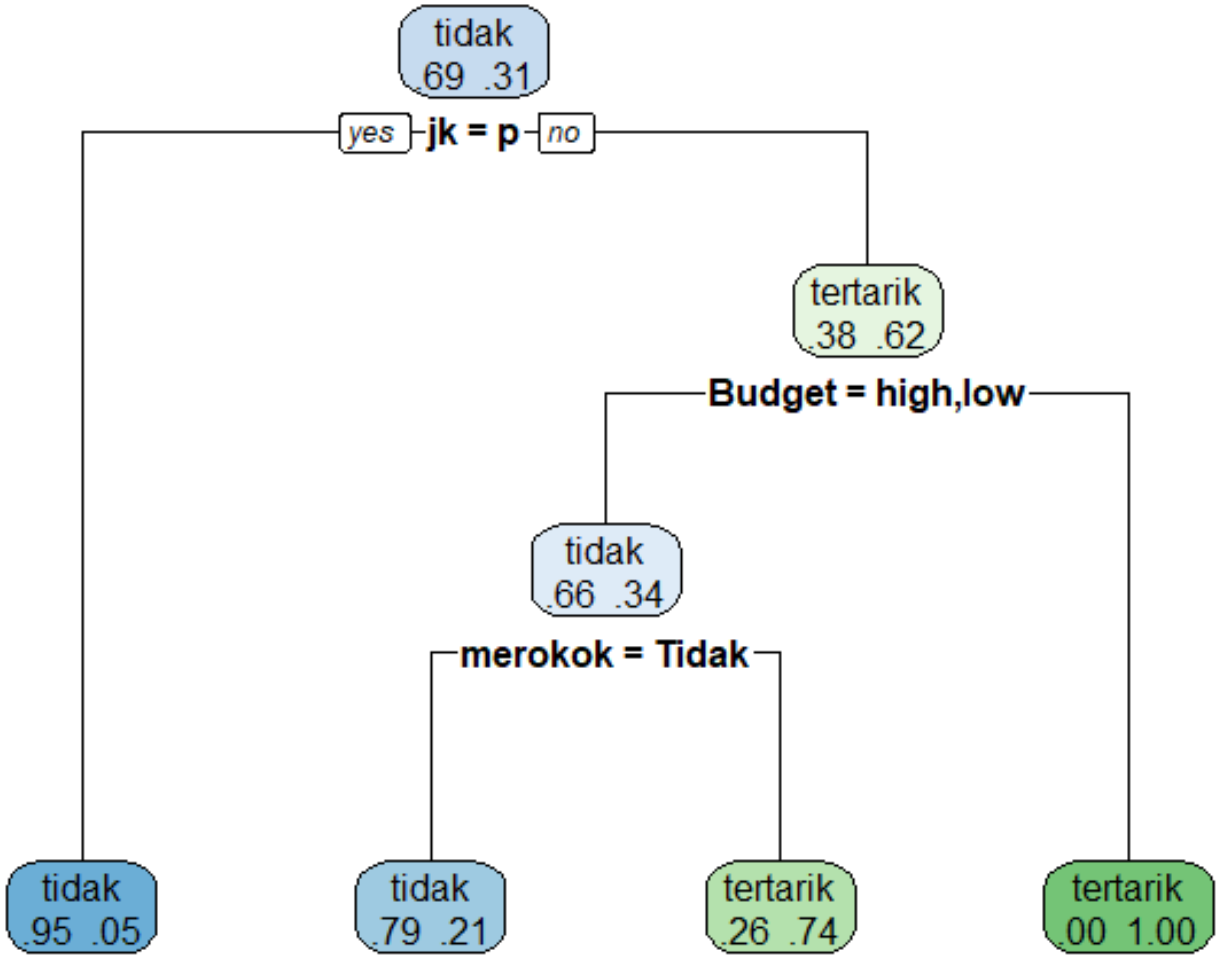

IG_merokok → 0.07

38

Ilustrasi Sederhana

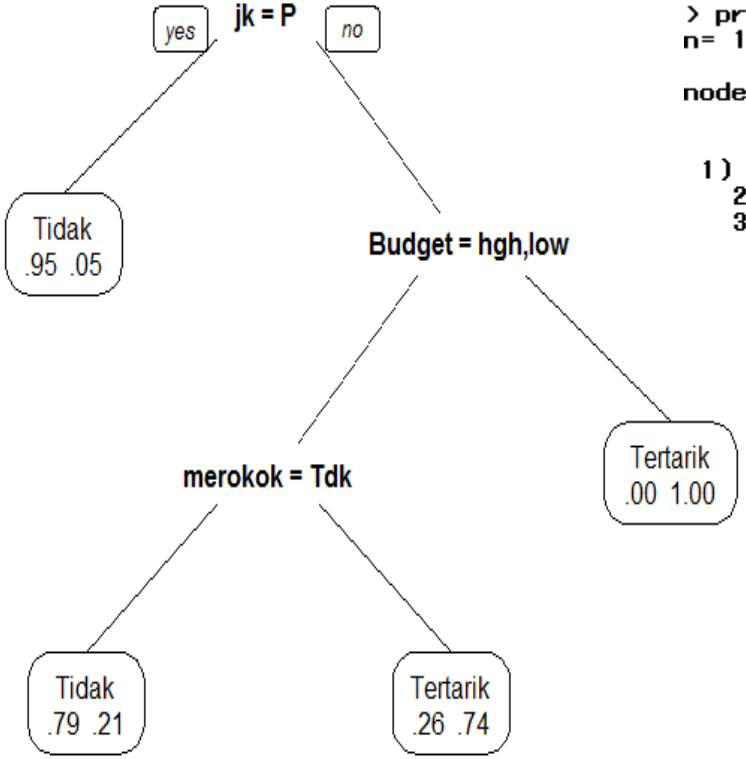
```
library(rpart)
library(rpart.plot)

model = rpart(tertarik ~  jk + tempattinggal + single + usia + merokok + Budget,
              data = data, method="class",
              control = rpart.control(minsplit = 100, cp = 0))
print(model)
rpart.plot(model, extra=4)
```

Grafik


tidak
.69 .31
yes
jk = p
no
tertarik
.38 .62
Budget = high,low
tidak
.66 .34
merokok = Tidak
tidak
.95 .05
tidak
.79 .21
tertarik
.26 .74
tertarik
.00 1.00

Ilustrasi Sederhana

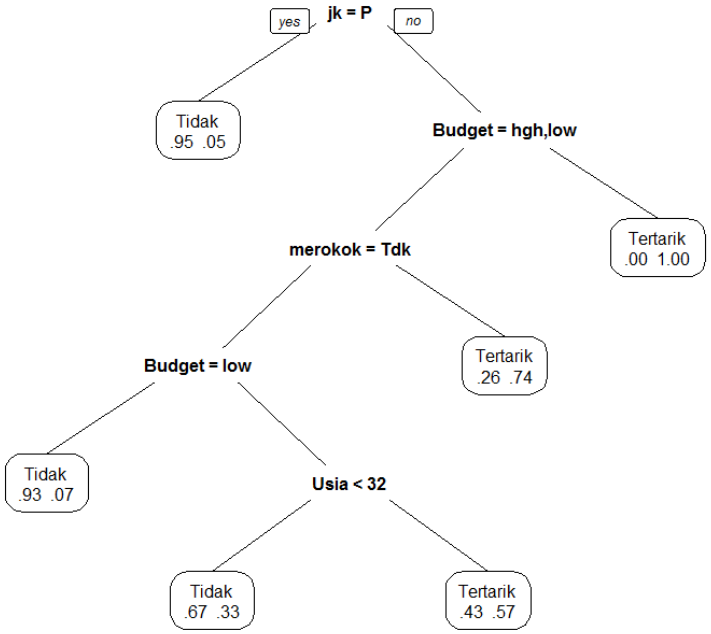
```
model = rpart(tertarik ~  jk + kota + single + usia + merokok + Budget,
                          data = a.data, method="class",
                          control = rpart.control(minsplit = 100, cp = 0))
```




```
> print(model)
n= 1084

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 1084 334 Tidak (0.691881919 0.308118081)
   2) jk=P 588  27 Tidak (0.954081633 0.045918367) *
   3) jk=L 496 189 Tertarik (0.381048387 0.618951613)
     6) Budget=high,low 283  95 Tidak (0.664310954 0.335689046)
      12) merokok=Tidak 217  46 Tidak (0.788018433 0.211981567) *
      13) merokok=Ya 66  17 Tertarik (0.257575758 0.742424242) *
     7) Budget=medium 213   1 Tertarik (0.004694836 0.995305164) *
```

Ilustrasi Sederhana

model = rpart(tertarik ~ jk + kota + single + usia + merokok + Budget,
data = a.data, method="class",
control = rpart.control(minsplit = **50**, cp = **0**))




```
> print(model)
n= 1084

node), split, n, loss, yval, (yprob)
      * denotes terminal node

 1) root 1084 334 Tidak (0.691881919 0.308118081)
   2) jk=P 588  27 Tidak (0.954081633 0.045918367) *
   3) jk=L 496 189 Tertarik (0.381048387 0.618951613)
     6) Budget=high,low 283  95 Tidak (0.66▒310954 0.335689046)
      12) merokok=Tidak 217  46 Tidak (0.788018433 0.211981567)
        24) Budget=low 147  11 Tidak (0.925170068 0.074829932) *
        25) Budget=high 70  35 Tidak (0.500000000 0.500000000)
          50) Usia< 31.5 21   7 Tidak (0.666666667 0.333333333) *
          51) Usia>=31.5 49  21 Tertarik (0.428571429 0.571428571) *
      13) merokok=Ya 66  17 Tertarik (0.257575758 0.742424242) *
     7) Budget=medium 213   1 Tertarik (0.004694836 0.995305164) *
```

3

Menilai Kebaikan Pohon Klasifikasi

Prediksi Variabel Respon

- Untuk setiap individu yang diketahui nilai-nilai variabel prediktor yang muncul pada pohon klasifikasi, kita dapat melakukan prediksi kelas variabel respon. Misalnya jika diketahui usia, jenis kelamin, apakah merokok, dan klasifikasi budget dari seseorang, maka kita dapat memprediksi apakah orang tersebut akan tertarik atau tidak.

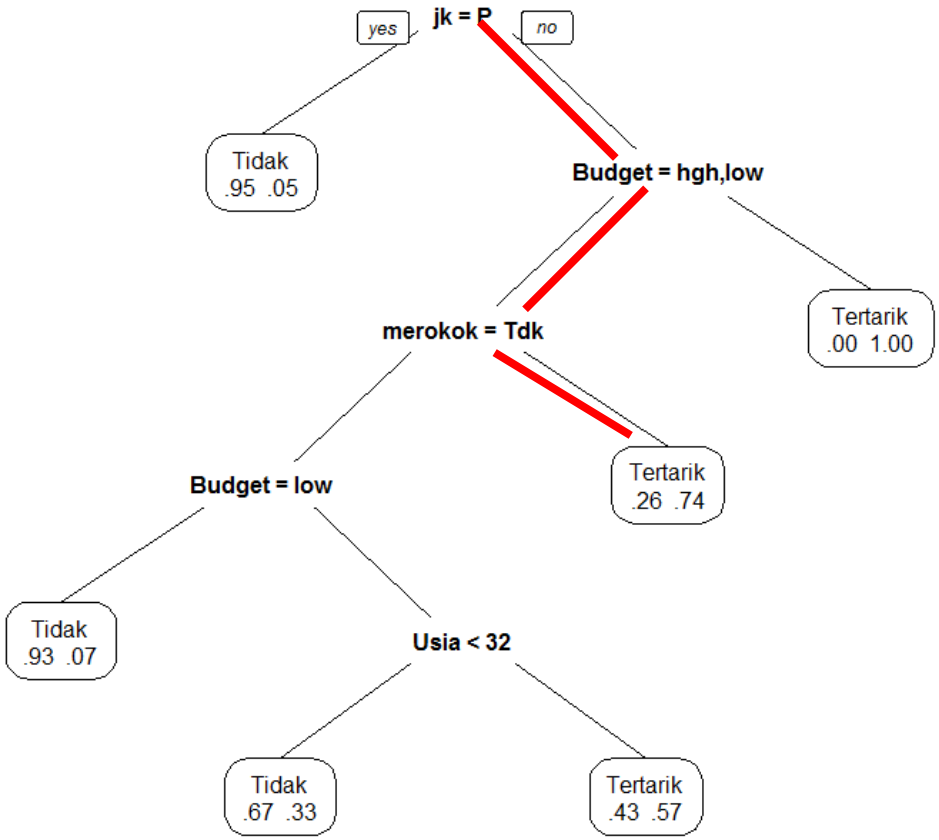
- Bagaimana caranya? Gunakan alur pencabangan yang ada pada pohon klasifikasi sampai berhenti di simpul akhir. Berdasarkan simpul akhir itulah kita prediksi dia masuk ke kategori apa.

Prediksi Variabel Respon

Misal

- Jenis Kelamin = Laki-Laki
- Budget Low
- Merokok
- Usia 25 tahun

- Probability TERTARIK = 0.74

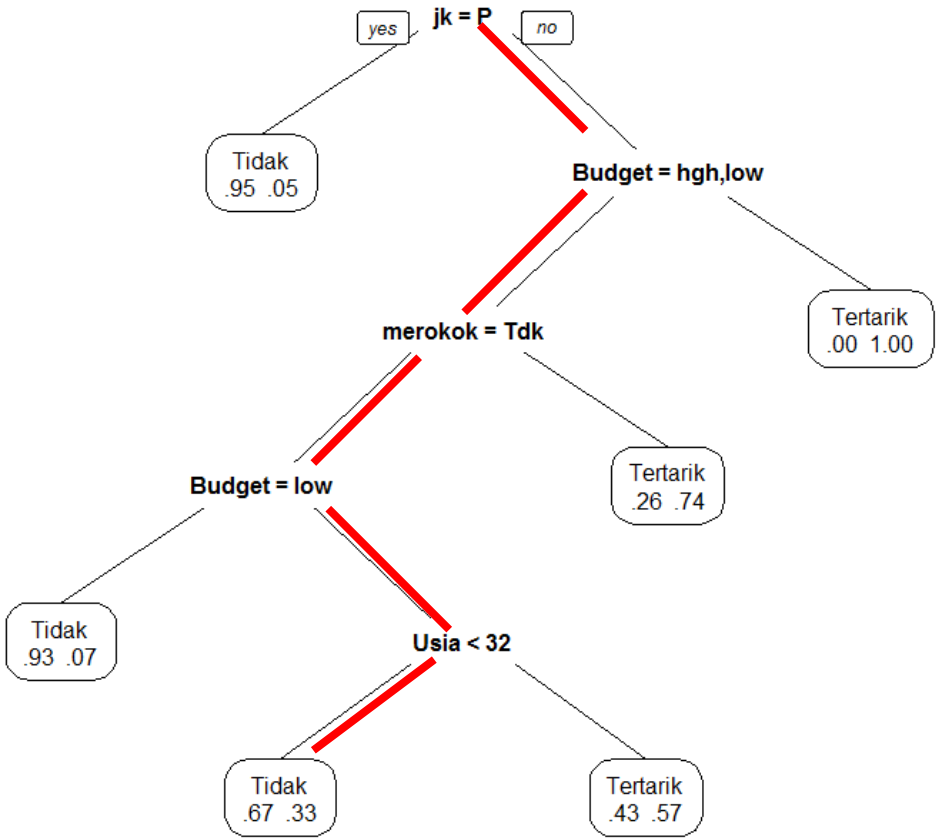


Prediksi Variabel Respon

Misal

- Jenis Kelamin = Laki-Laki
- Budget high
- Tidak Merokok
- Usia 25 tahun

- Probability TERTARIK = 0.33



Prediksi Variabel Respon

```
model = rpart(tertarik ~  jk + kota + single + usia + merokok + Budget,
                          data = a.data, method="class",
                          control = rpart.control(minsplit = 50, cp = 0))
```


prob_prediksi <- predict(model, newdata=data, type = 'prob')
head(prob_prediksi, n=10)

```
       Tidak   Tertarik
1  0.925170068 0.07482993
2  0.954081633 0.04591837
3  0.954081633 0.04591837
4  0.954081633 0.04591837
5  0.004694836 0.99530516
6  0.954081633 0.04591837
7  0.004694836 0.99530516
8  0.925170068 0.07482993
9  0.954081633 0.04591837
10 0.428571429 0.57142857
```

Prediksi Variabel Respon

Andaikan digunakan batasan 0.5 untuk mengelompokkan ketertarikan, sehingga kalau
Prob(Tertarik) > 0.5 → Tertarik
Prob(Tertarik) ≤ 0.5 → tidak

bagusco

Maka kita akan dapatkan

	Tidak	Tertarik		Prediksi
1	0.925170068	0.07482993	→	Tidak
2	0.954081633	0.04591837	→	Tidak
3	0.954081633	0.04591837	→	Tidak
4	0.954081633	0.04591837	→	Tidak
5	0.004694836	0.99530516	→	Tertarik
6	0.954081633	0.04591837	→	Tidak
7	0.004694836	0.99530516	→	Tertarik
8	0.925170068	0.07482993	→	Tidak
9	0.954081633	0.04591837	→	Tidak
10	0.428571429	0.57142857	→	Tertarik

Prediksi Variabel Respon

Perbandingan antara respon yang sebenarnya dengan dugaan

bagusco

	Tertarik_beli	dugaan	
1	Tidak	Tidak	
2	Tidak	Tidak	
3	Tidak	Tidak	
4	Tidak	Tidak	
5	Tertarik	Tertarik	
6	Tidak	Tidak	
7	Tertarik	Tertarik	
8	Tidak	Tidak	
9	Tidak	Tidak	
10	Tidak	Tertarik	→ salah prediksi

Kebaikan pohon klasifikasi

Kebaikan dapat dilihat dari seberapa tinggi kemampuan pohon klasifikasi menghasilkan dugaan yang sama dengan kondisi yang sesungguhnya.


```
prediksi <- ifelse (prob_prediksi[,2] > 0.5, "tertarik", "tidak")
table(data$tertarik, prediksi)
```

	prediksi	
	tertarik	tidak
tidak	39	711
tertarik	289	45

Kebaikan pohon klasifikasi

```
library(caret)
confusionMatrix(prediksi, data$tertarik,
positive="tertarik")
```

```
Confusion Matrix and Statistics

          Reference
Prediction tidak tertarik
  tidak      711       45
  tertarik    39      289

               Accuracy : 0.9225
                 95% CI : (0.905, 0.9377)
    No Information Rate : 0.6919
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.8173
 Mcnemar's Test P-Value : 0.5854

            Sensitivity : 0.8653
            Specificity : 0.9480
         Pos Pred Value : 0.8811
         Neg Pred Value : 0.9405
             Prevalence : 0.3081
         Detection Rate : 0.2666
   Detection Prevalence : 0.3026
      Balanced Accuracy : 0.9066

       'Positive' Class : tertarik
```

Kebaikan pohon klasifikasi

Cut-Off	Accuracy	Sensitivity	Specificity
0.3	91.60%	88.62%	92.93%
0.5	92.25%	86.53%	94.80%
0.6	91.60%	78.14%	97.60%

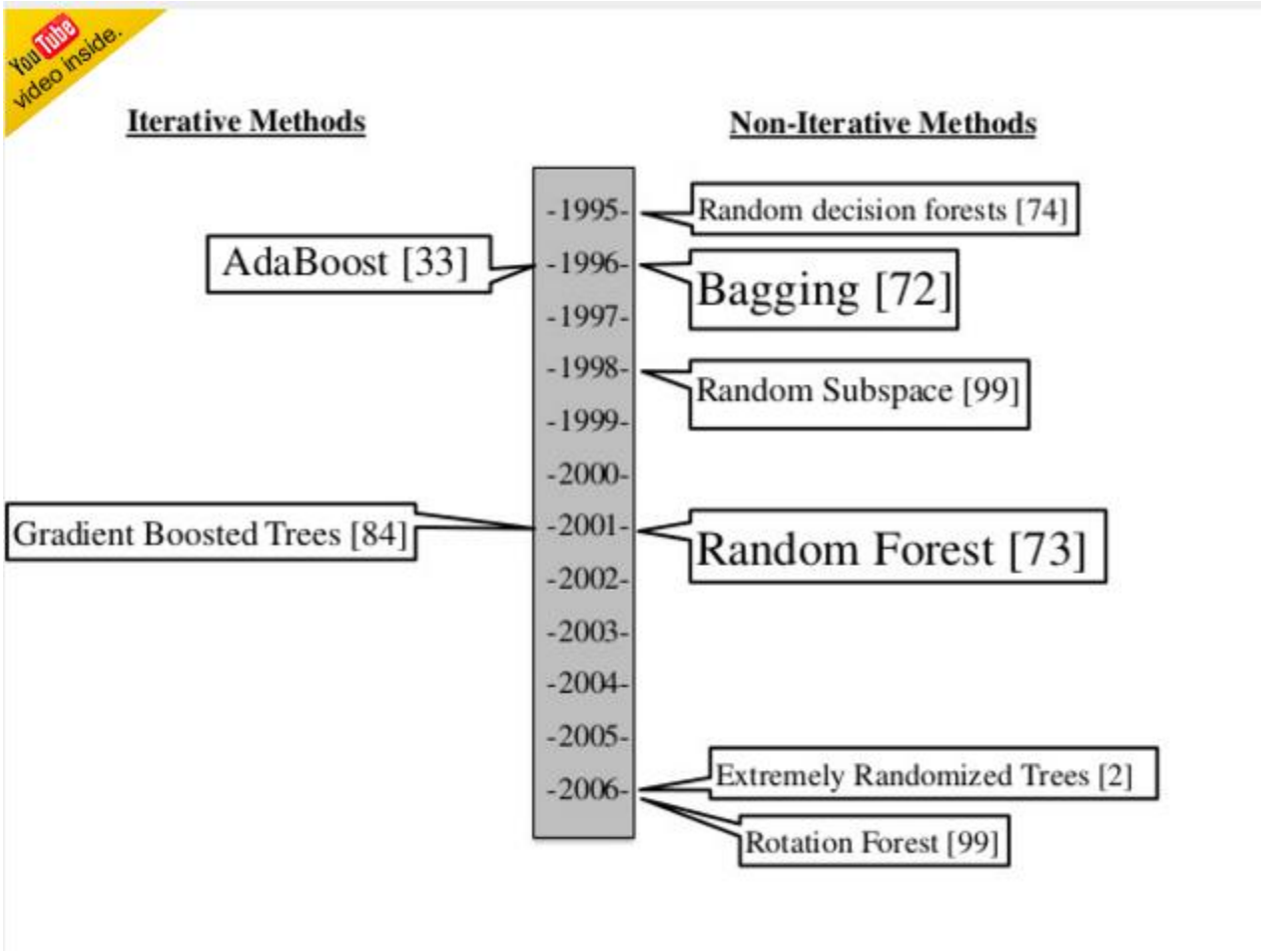
4

Perkembangan Lebih Lanjut

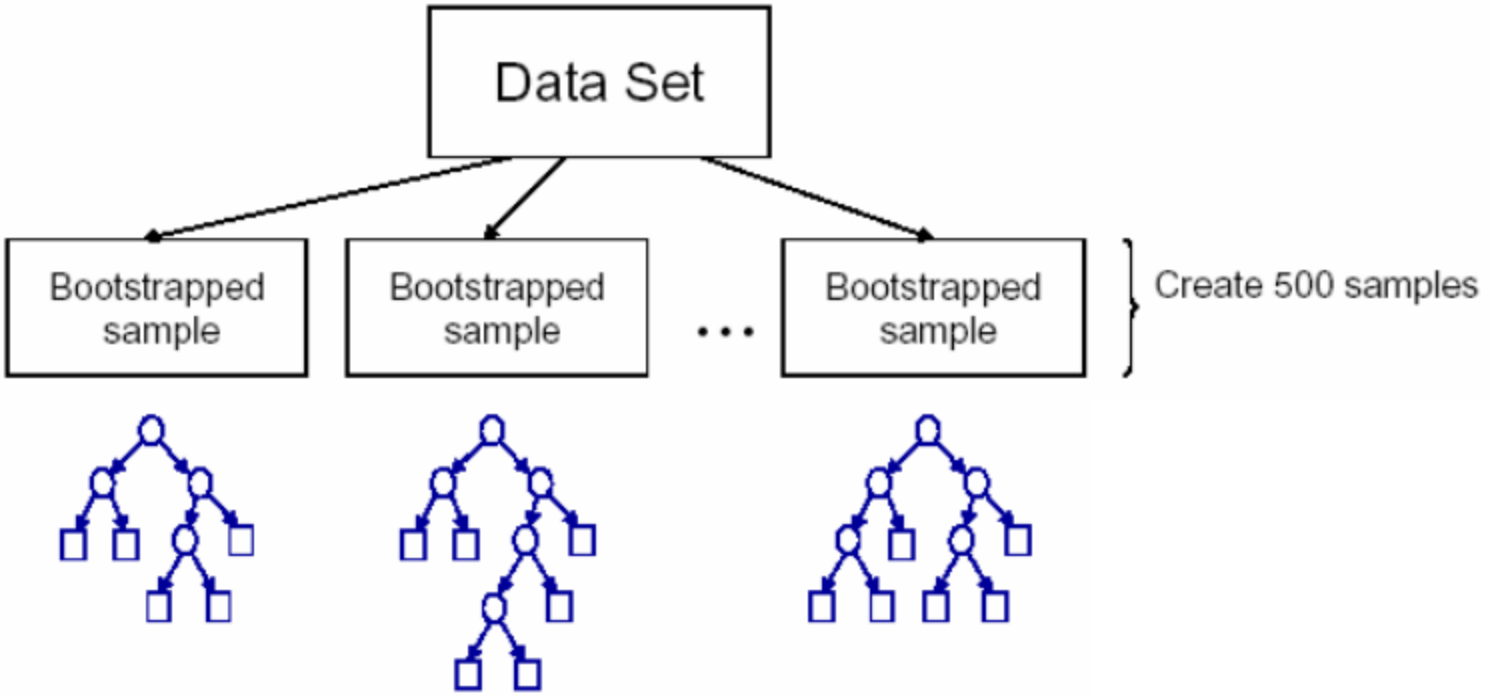
Perkembangan Lebih Lanjut: Ensemble Tree

- Bagging (Bootstrap Aggregating)
  - Breiman, L (1996). Bagging predictors. *Machine Learning* 24 (2): 123–140
- Boosting
  - Freund, Y., & Schapire, R. E. (1996, July). Experiments with a new boosting algorithm. In *ICML* (Vol. 96, pp. 148-156).
- Random Forest
  - Breiman L (2001). Random Forests. *Machine Learning*, 45, 5-32.
  - Ho, Tin Kam (1995). *Random Decision Forests*. Proceedings of the 3rd International Conference on Document Analysis and Recognition, Montreal, QC, 14–16 August 1995. pp. 278–282
  - Ho, Tin Kam (1998). "The Random Subspace Method for Constructing Decision Forests". *IEEE Transactions on Pattern Analysis and Machine Intelligence* **20** (8): 832–844
- Rotation Forest
  - Rodriguez, J. J., Kuncheva, L. I., & Alonso, C. J. (2006). Rotation forest: A new classifier ensemble method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, *28*(10), 1619-1630.

Perkembangan Lebih Lanjut

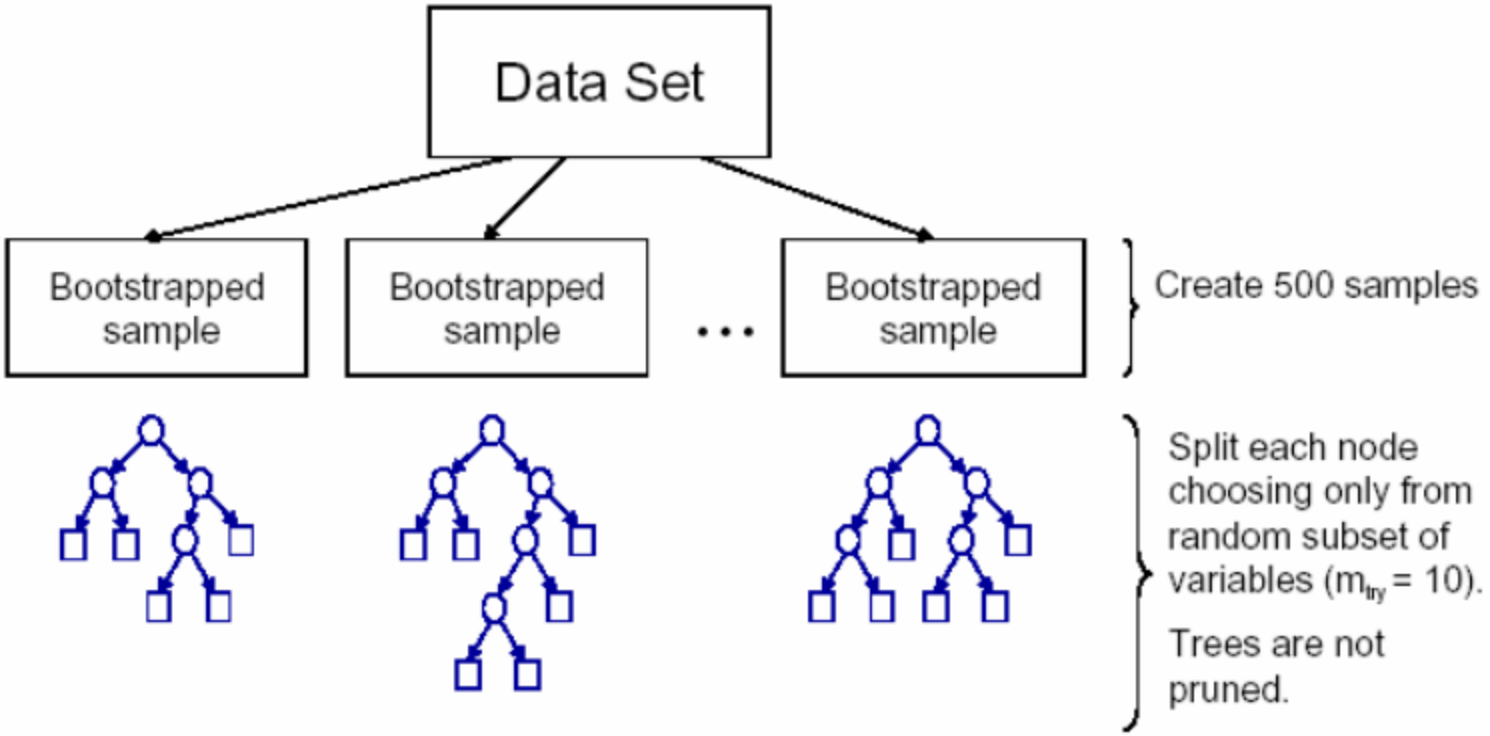


Bagging



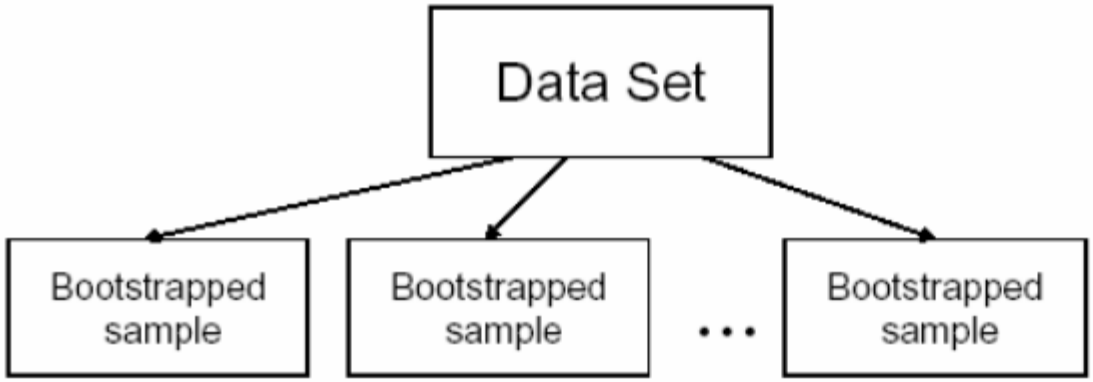
To classify new observation, use majority vote from the forest.

Random Forest



To classify new observation, use majority vote from the forest.

Rotation Forest



Create 500 samples

SCO

Buat beberapa grup veriabel prediktor

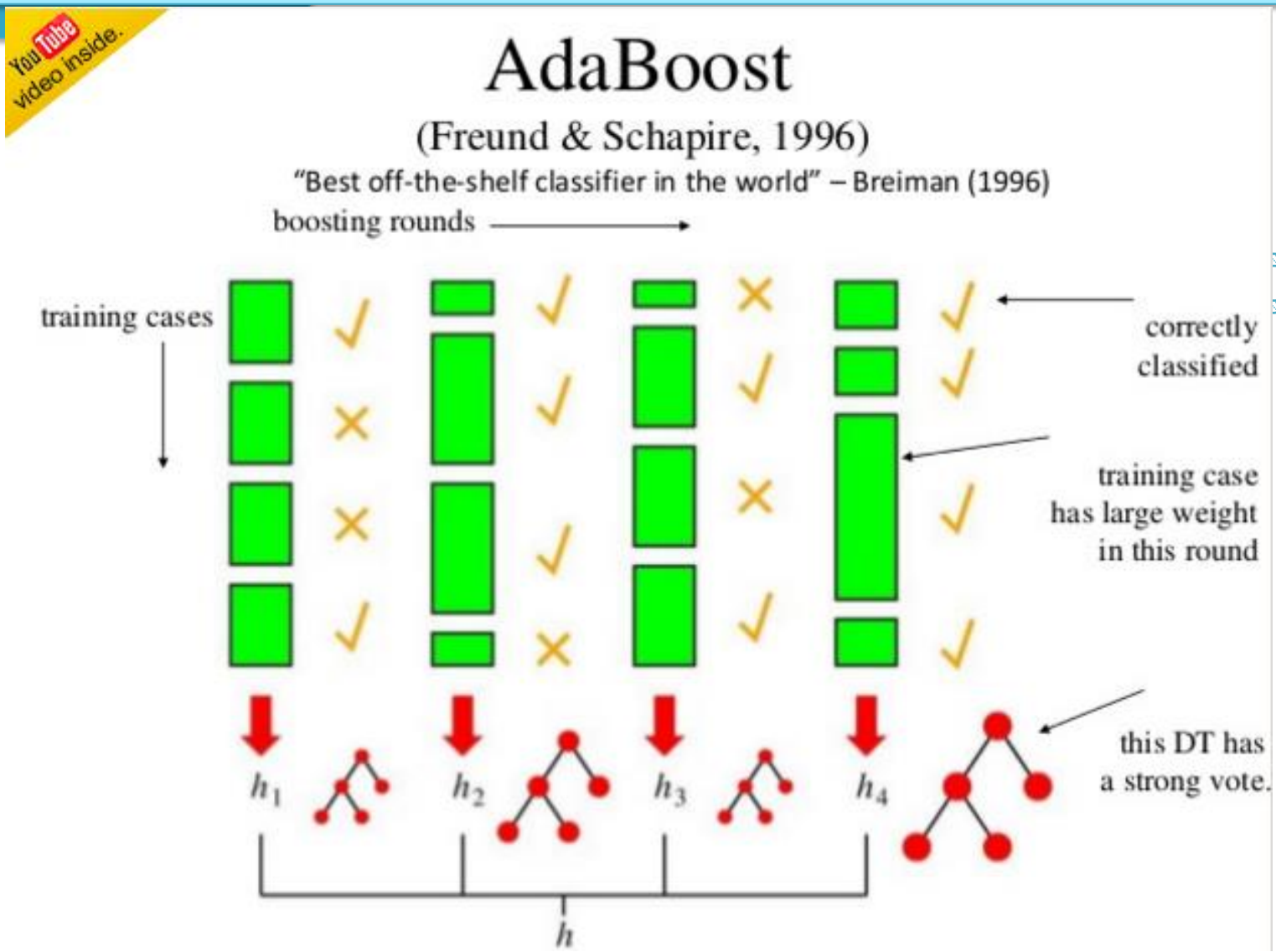
Lakukan PCA pada setiap group

Gunakan koef PC untuk merotasi data prediktor

Buat pohon klasifikasi berdasarkan data yang telah dirotasi

To classify new observation, use majority vote from the forest.

Boosting


AdaBoost
(Freund & Schapire, 1996)
"Best off-the-shelf classifier in the world" – Breiman (1996)
boosting rounds
training cases
correctly classified
training case has large weight in this round
this DT has a strong vote.
h_1
h_2
h_3
h_4
h

Empirical study

- Pendekatan ensemble tree menghasilkan prediksi yang lebih baik dibandingkan pohon klasifikasi tunggal

bagusco

- Ensemble tree banyak digunakan untuk menangani masalah-masalah:
  - Ketidakseimbangan Kelas (Imbalanced Class)
  - Curse of Dimensionality
  - Klasifikasi Multi-Kelas

Beyond Classification Tasks

- Regression tree (Breiman *et al.*, 1984)
- Survival tree (Bou-Hamad *et al.*, 2011)
- Clustering tree (Blockeel *et al.*, 1998)
- Recommendation tree (Gershman *et al.*, 2010):
- Markov model tree (Antwarg *et al.*, 2012)
- ….

Thank you!