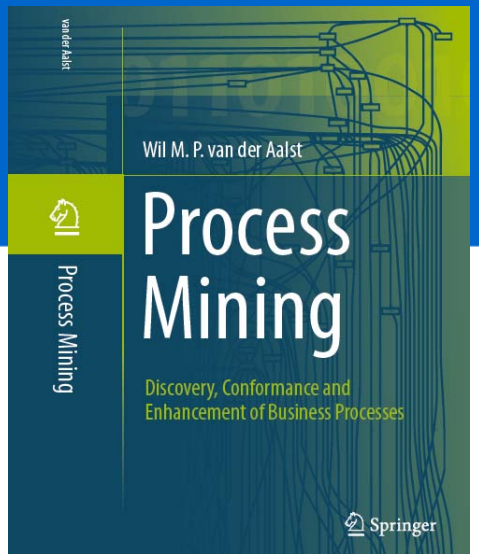
Chapter 3
Data Mining

prof.dr.ir. Wil van der Aalst
www.processmining.org



TU/e Technische Universiteit Eindhoven University of Technology

Where innovation starts

Overview

Chapter 1
Introduction

Part I: Preliminaries

Chapter 2
Process Modeling and Analysis

Chapter 3
Data Mining

Part II: From Event Logs to Process Models

Chapter 4
Getting the Data

Chapter 5
Process Discovery: An Introduction

Chapter 6
Advanced Process Discovery Techniques

Part III: Beyond Process Discovery

Chapter 7
Conformance Checking

Chapter 8
Mining Additional Perspectives

Chapter 9
Operational Support

Part IV: Putting Process Mining to Work

Chapter 10
Tool Support

Chapter 11
Analyzing “Lasagna Processes”

Chapter 12
Analyzing “Spaghetti Processes”

Part V: Reflection

Chapter 13
Cartography and Navigation

Chapter 14
Epilogue

Data mining

- **The growth of the “digital universe” is the main driver for the popularity of data mining.**
- **Initially, the term “data mining” had a negative connotation (“data snooping”, “fishing”, and “data dredging”).**
- **Now a mature discipline.**
- **Data-centric, not process-centric.**

Data set 1

Data about 860 recently deceased persons to study the effects of drinking, smoking, and body weight on the life expectancy.

drinker	smoker	weight	age
yes	yes	120	44
no	no	70	96
yes	no	72	88
yes	yes	55	52
no	yes	94	56
no	no	62	93
...	...	...	...

Questions:

- **What is the effect of smoking and drinking on a person's bodyweight?**
- **Do people that smoke also drink?**
- **What factors influence a person's life expectancy the most?**
- **Can one identify groups of people having a similar lifestyle?**

Data set 2

Data about 420 students to investigate relationships among course grades and the student's overall performance in the Bachelor program.

linear algebra	logic	program-ming	operations research	workflow systems	...	duration	result
9	8	8	9	9	...	36	cum laude
7	6	-	8	8	...	42	passed
-	-	5	4	6	...	54	failed
8	6	6	6	5	...	38	passed
6	7	6	-	8	...	39	passed
9	9	9	9	8	...	38	cum laude
5	5	-	6	6	...	52	failed
...	...	...	...	...	...	...	...

Questions:

- Are the marks of certain courses highly correlated?
- Which electives do excellent students (cum laude) take?
- Which courses significantly delay the moment of graduation?
- Why do students drop out?
- Can one identify groups of students having a similar study behavior?

Data set 3

Data on 240 customer orders in a coffee bar recorded by the cash register.

cappuccino	latte	espresso	americano	ristretto	tea	muffin	bagel
1	0	0	0	0	0	1	0
0	2	0	0	0	0	1	1
0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	1	2	0
0	0	0	1	1	0	0	0
...	...	...	...	...	...	...	...

Questions:

- **Which products are frequently purchased together?**
- **When do people buy a particular product?**
- **Is it possible to characterize typical customer groups?**
- **How to promote the sales of products with a higher margin?**

Variables

- **Data set (sample or table) consists of instances (individuals, entities, cases, objects, or records).**
- **Variables are often referred to as attributes, features, or data elements.**
- **Two types:**
  - **categorical variables:**
    - **ordinal (high-med-low, cum laude-passed-failed) or**
    - **nominal (true-false, red-pink-green)**
  - **numerical variables (ordered, cannot be enumerated easily)**

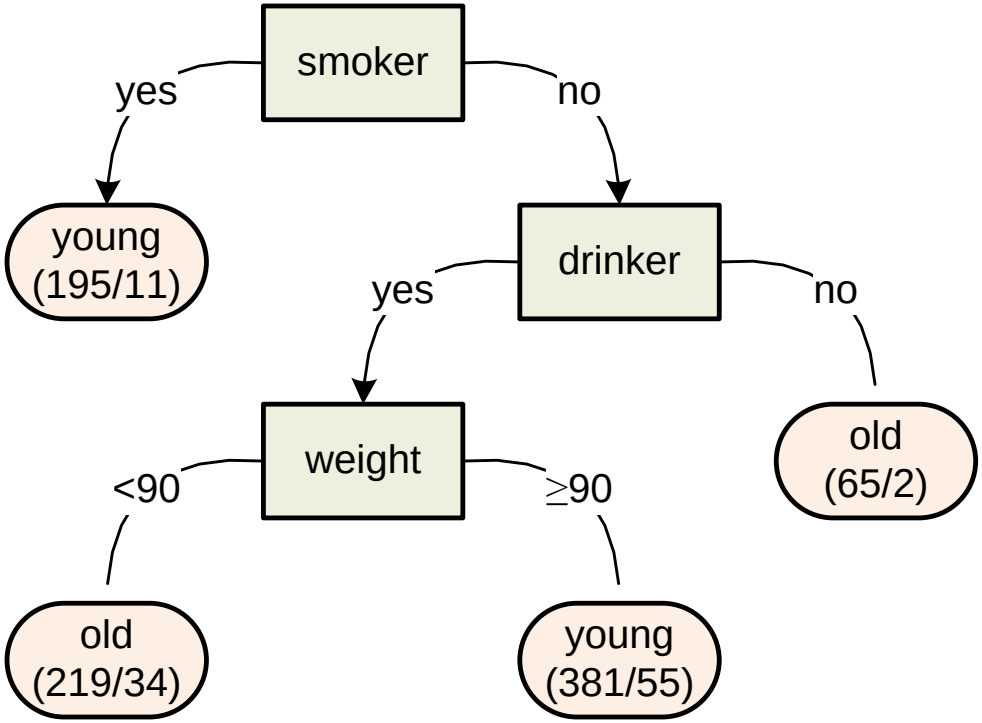
Supervised Learning

- **Labeled data, i.e., there is a response variable that labels each instance.**
- **Goal: explain response variable (dependent variable) in terms of predictor variables (independent variables).**
- **Classification techniques (e.g., decision tree learning) assume a categorical response variable and the goal is to classify instances based on the predictor variables.**
- **Regression techniques assume a numerical response variable. The goal is to find a function that fits the data with the least error.**

Unsupervised Learning

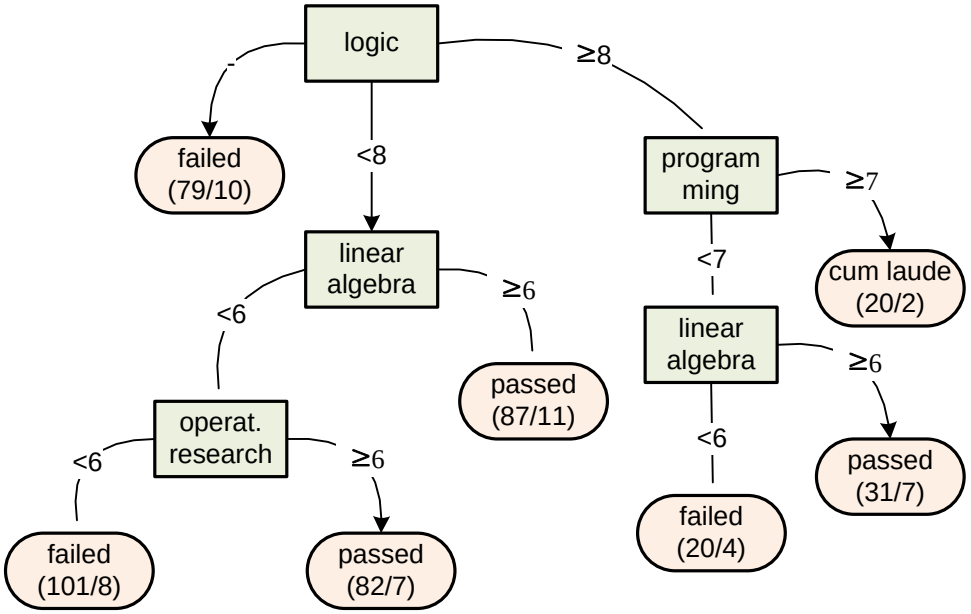
- **Unsupervised learning assumes unlabeled data, i.e., the variables are not split into response and predictor variables.**
- **Examples: clustering (e.g., k-means clustering and agglomerative hierarchical clustering) and pattern discovery (association rules)**

Decision tree learning: data set 1



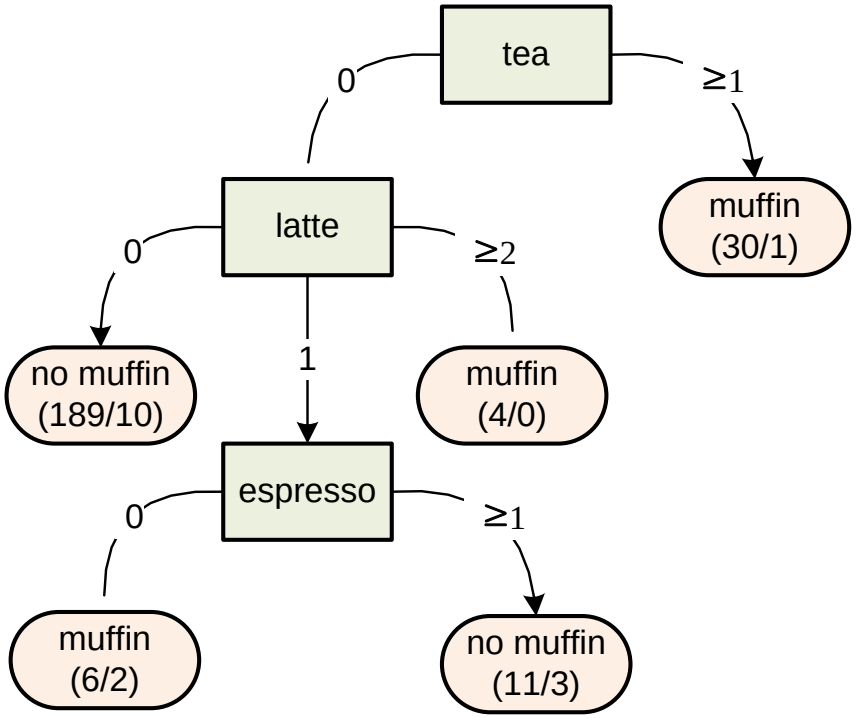
drinker	smoker	weight	age
yes	yes	120	44
no	no	70	96
yes	no	72	88
yes	yes	55	52
no	yes	94	56
no	no	62	93
…	…	…	…

Decision tree learning: data set 2



linear algebra	logic	program-ming	operations research	workflow systems	...	duration	result
9	8	8	9	9	...	36	cum laude
7	6	-	8	8	...	42	passed
-	-	5	4	6	...	54	failed
8	6	6	6	5	...	38	passed
6	7	6	-	8	...	39	passed
9	9	9	9	8	...	38	cum laude
5	5	-	6	6	...	52	failed
...	...	...	...	...	...	...	...

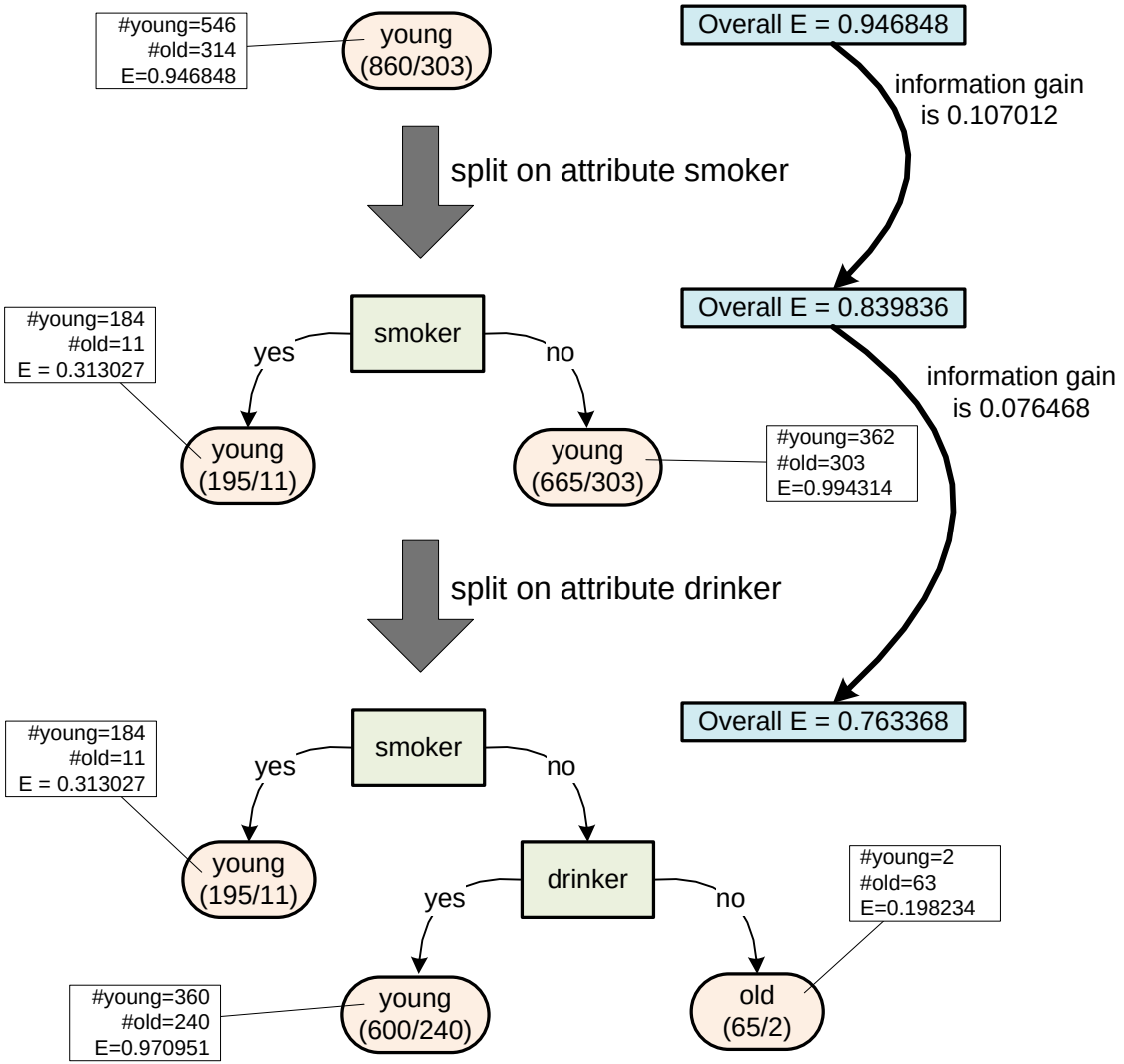
Decision tree learning: data set 3



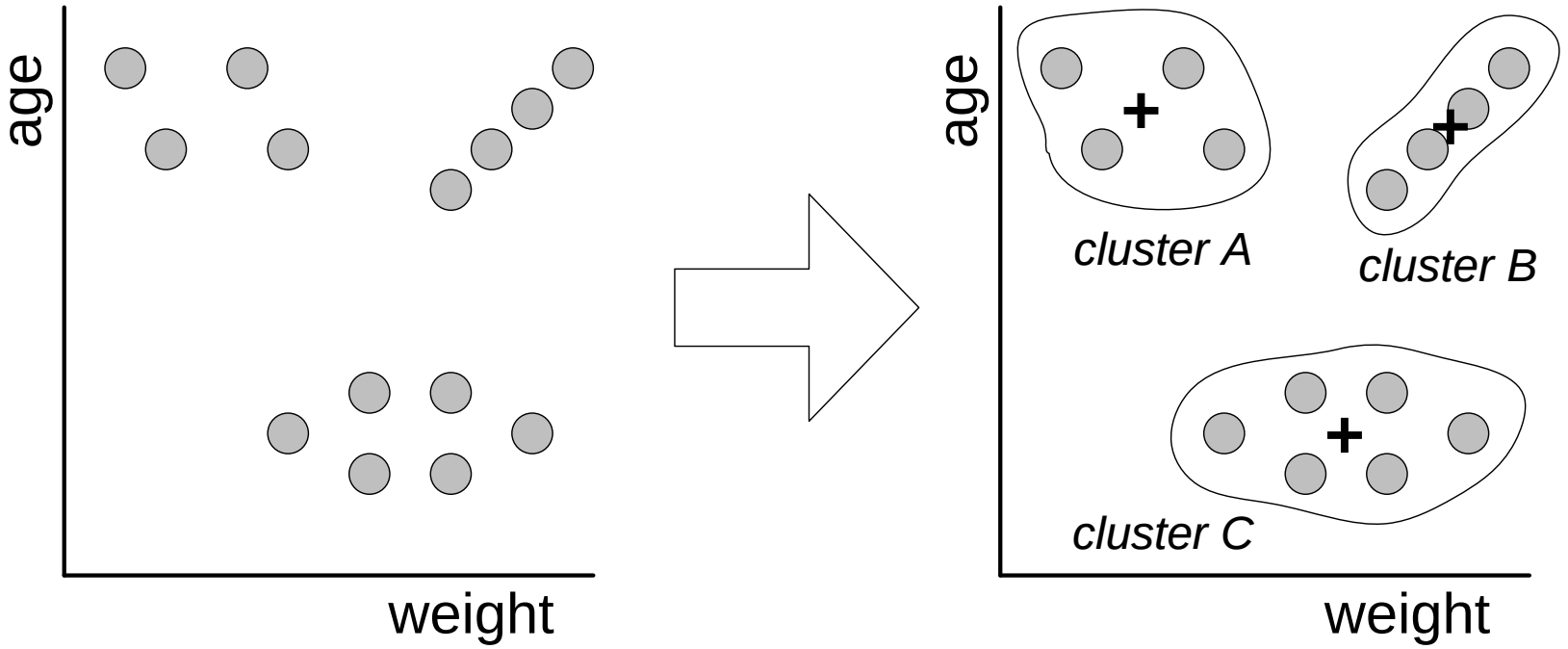
cappuccino	latte	espresso	americano	ristretto	tea	muffin	bagel
1	0	0	0	0	0	1	0
0	2	0	0	0	0	1	1
0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	1	2	0
0	0	0	1	1	0	0	0
…	…	…	…	…	…	…	…

Basic idea

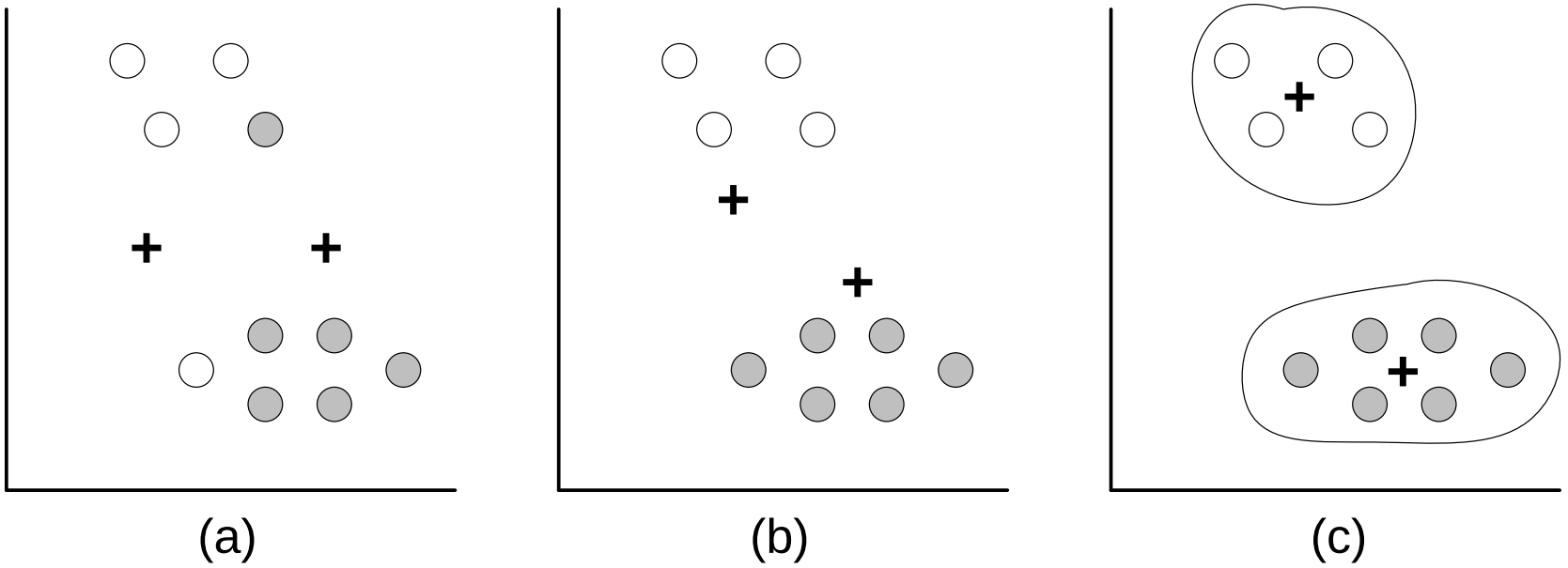
- **Split the set of instances in subsets such that the variation within each subset becomes smaller.**
- **Based on notion of entropy or similar.**
- **Minimize average entropy; maximize information gain per step.**



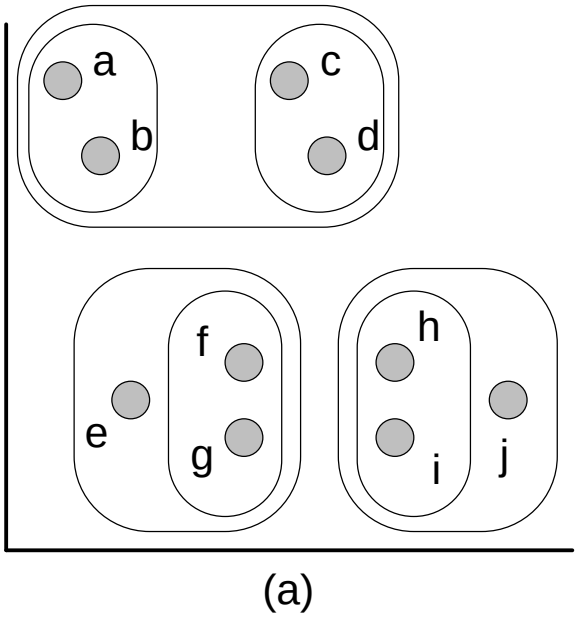
Clustering



k-means clustering

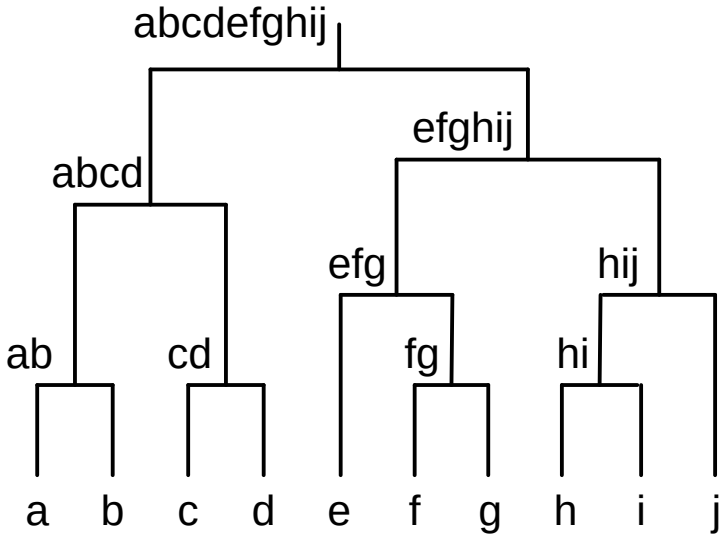


Agglomerative hierarchical clustering



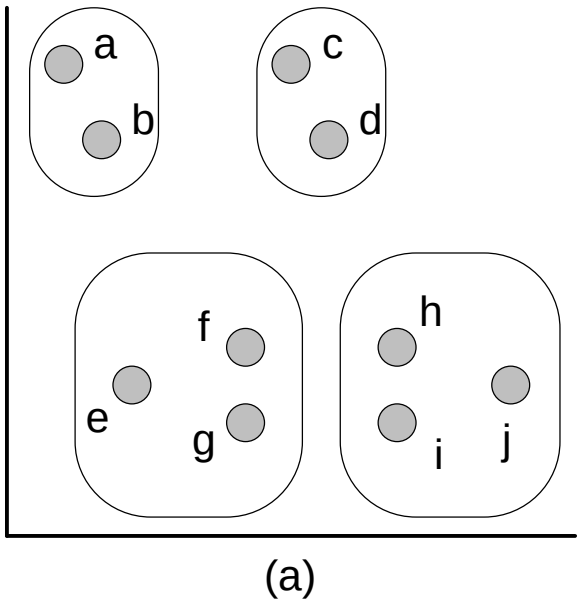
(a)

dendrogram

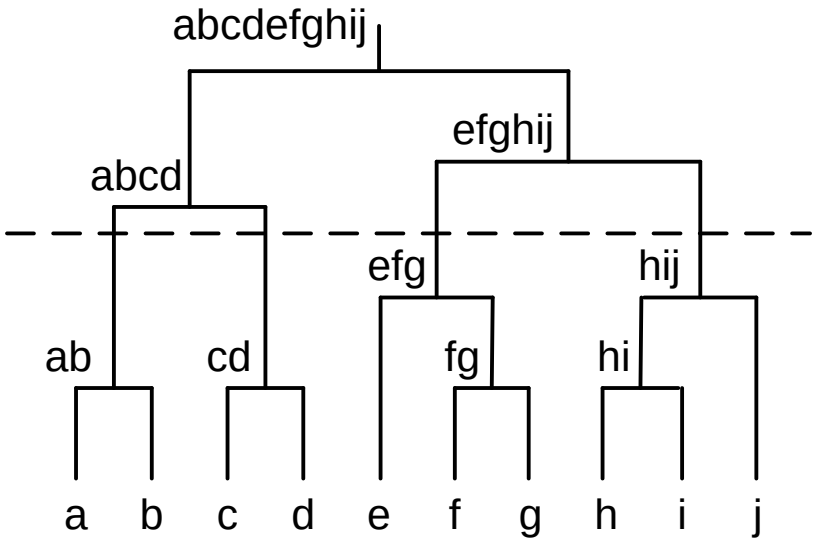


(b)

Levels introduced by agglomerative hierarchical clustering



(a)



(b)

Any horizontal line in dendrogram corresponds to a concrete clustering at a particular level of abstraction

Association rule learning

- **Rules of form "IF X THEN Y"** $X \Rightarrow Y$

$$support(X \Rightarrow Y) = N_{X \wedge Y} / N$$

$$confidence(X \Rightarrow Y) = N_{X \wedge Y} / N_X$$

$$lift(X \Rightarrow Y) = \frac{N_{X \wedge Y} / N}{(N_X / N)\ (N_Y / N)} = \frac{N_{X \wedge Y}\ N}{N_X\ N_Y}$$

Special case: market basket analysis

cappuccino	latte	espresso	americano	ristretto	tea	muffin	bagel
1	0	0	0	0	0	1	0
0	2	0	0	0	0	1	1
0	0	1	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	1	2	0
0	0	0	1	1	0	0	0
…	…	…	…	…	…	…	…

$$tea \wedge latte \Rightarrow muffin$$

$$tea \Rightarrow muffin \wedge bagel$$

Example
(people that order tea and latte also order muffins)

$$tea \wedge latte \Rightarrow muffin, \text{ i.e., } X = tea \wedge latte \text{ and } Y = muffin$$

$$support(X \Rightarrow Y) = N_{X \wedge Y}/N = N_{tea \wedge latte \wedge muffin}/N = 15/240 = 0.0625$$

$$confidence(X \Rightarrow Y) = N_{X \wedge Y}/N_X = N_{tea \wedge latte \wedge muffin}/N_{tea \wedge latte} = 15/20 = 0.75$$

$$lift(X \Rightarrow Y) = \frac{N_{X \wedge Y}\, N}{N_X\, N_Y} = \frac{N_{tea \wedge latte \wedge muffin}\, N}{N_{tea \wedge latte}\, N_{muffin}} = \frac{15 \times 240}{20 \times 40} = 4.5$$

- **Support should be as high as possible (but will be low in case of many items).**
- **Confidence should be close to 1.**
- **High lift values suggest a positive correlation (1 if independent).**

Brute force algorithm

Association rules can now be generated as follows:

1. Generate all *frequent item-sets*, i.e., all sets Z such that $N_Z/N \geq minsup$ and $|Z| \geq 2$.
2. For each frequent item-set Z consider all partitionings of Z into two non-empty subsets X and Y. If $confidence(X \Rightarrow Y) \geq minconf$, then keep the rule $X \Rightarrow Y$. If $confidence(X \Rightarrow Y) < minconf$, then discard the rule.
3. Output the rules found.

Apriori (optimization based on two observations)

1. If an item-set is *frequent* (i.e., an item-set with a support above the threshold), then all of its non-empty subsets are also frequent. Formally, for any pair of non-empty item-sets X, Y: if $Y \subseteq X$ and $N_X/N \geq minsup$, then $N_Y/N \geq minsup$.
2. If, for any k, I_k is the set of all frequent item-sets with cardinality k and $I_l = \emptyset$ for some l, then $I_k = \emptyset$ for all $k \geq l$.

1. Create I_1. This is the set of singleton frequent item-sets, i.e., item-sets with a support above the threshold *minsup* containing just one element.
2. $k := 1$
3. If $I_k = \emptyset$, then output $\bigcup_{i=1}^{k} I_i$ and end. If $I_k \neq \emptyset$, continue with the next step.
4. Create C_{k+1} from I_k. C_{k+1} is the candidate set containing item-sets of cardinality $k+1$. Note that one only needs to consider elements that are the union of two item-sets A and B in I_k such that $|A \cap B| = k$ and $|A \cup B| = k+1$.
5. For each candidate frequent item-set $c \in C_{k+1}$: examine all subsets of c with k elements; delete c from C_{k+1} if any of the subsets is not a member of I_k.
6. For each item-set c in the pruned candidate frequent item-set C_{k+1}, check whether c is indeed frequent. If so, add c to I_{k+1}. Otherwise, discard c.
7. $k := k+1$ and return to Step 3.

Sequence mining

$$X \Rightarrow Y$$

customer	seq. number	timestamp	items
Wil	1	02-01-2011:09.02	{*cappuccino*}
	2	03-01-2011:10.06	{*espresso,muffin*}
	3	05-01-2011:15.12	{*americano,cappuccino*}
	4	06-01-2011:11.18	{*espresso,muffin*}
	5	07-01-2011:14.24	{*cappuccino*}
	6	07-01-2011:14.24	{*americano,cappuccino*}
Mary	1	30-12-2010:11.32	{*tea*}
	2	30-12-2010:12.12	{*cappuccino*}
	3	30-12-2010:14.16	{*espresso,muffin*}
	4	05-01-2011:11.22	{*bagel,tea*}
Bill	1	30-12-2010:14.32	{*cappuccino*}
	2	30-12-2010:15.06	{*cappuccino*}
	3	30-12-2010:16.34	{*bagel,espresso,muffin*}
	4	06-01-2011:09.18	{*ristretto*}
	5	06-01-2011:12.18	{*cappuccino*}
...	...	...	...

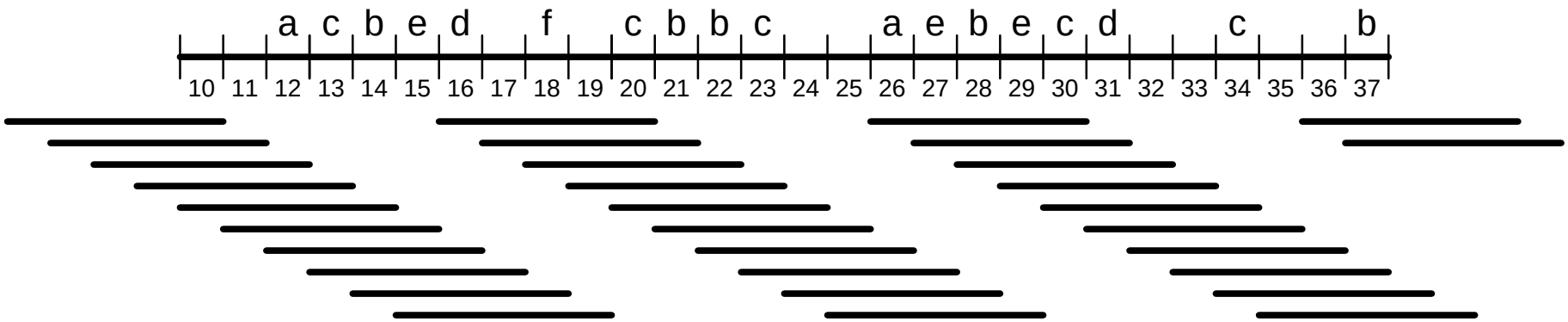
$$X = \langle \{cappuccino\}, \{espresso\} \rangle$$

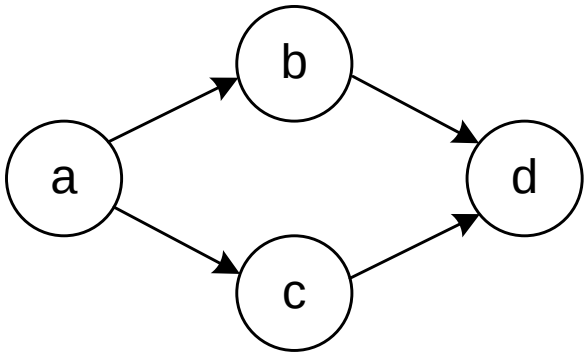
$$Y = \langle \{cappuccino\}, \{espresso\}, \{latte, muffin\} \rangle$$

$$support(X \Rightarrow Y) = N_{X \wedge Y}/N$$

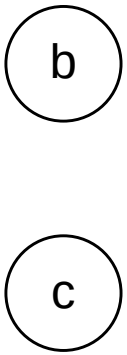
$$confidence(X \Rightarrow Y) = N_{X \wedge Y}/N_X$$

Episode mining
(32 time windows of length 5)

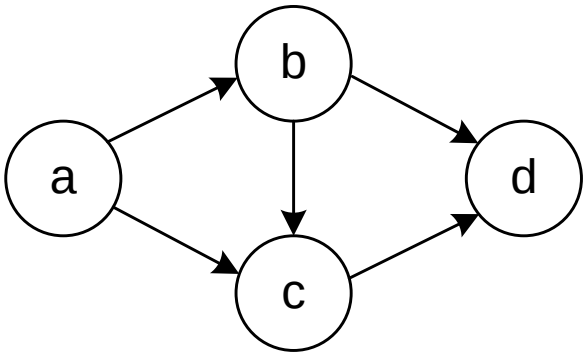




E1

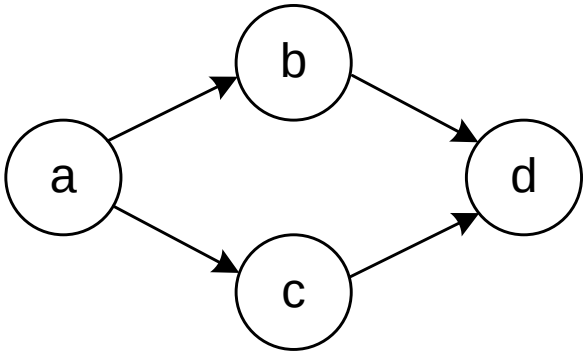


E2

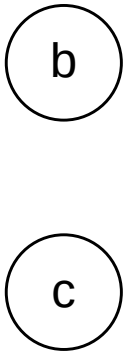


E3

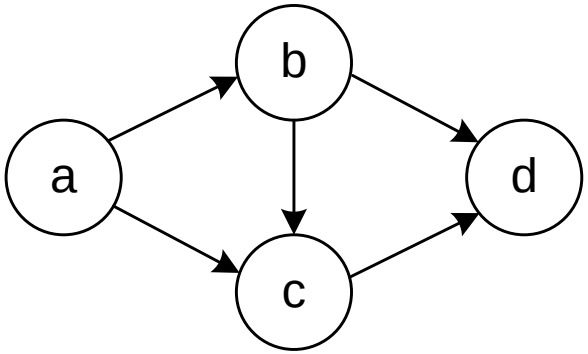
Occurrences



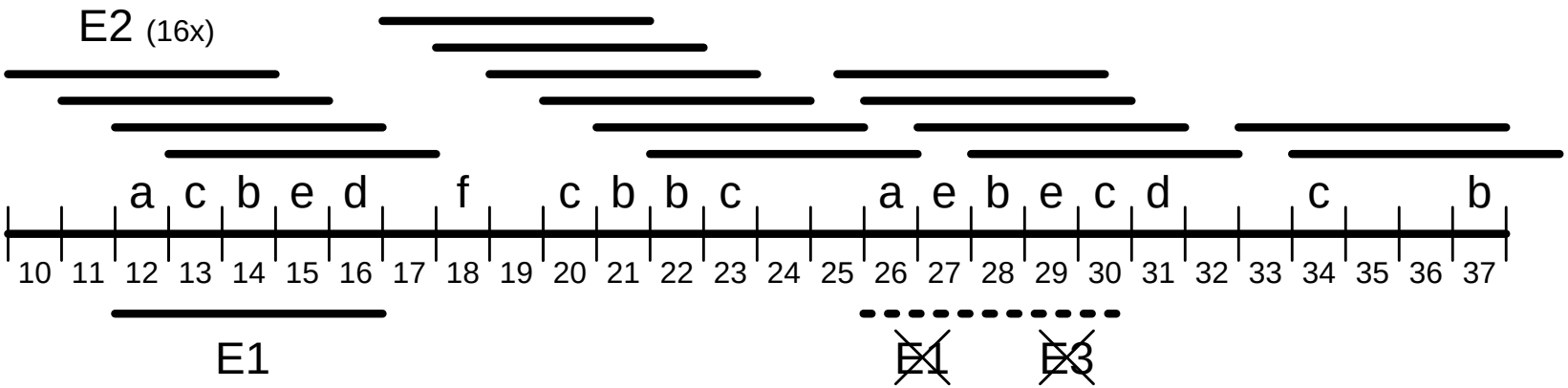
E1



E2



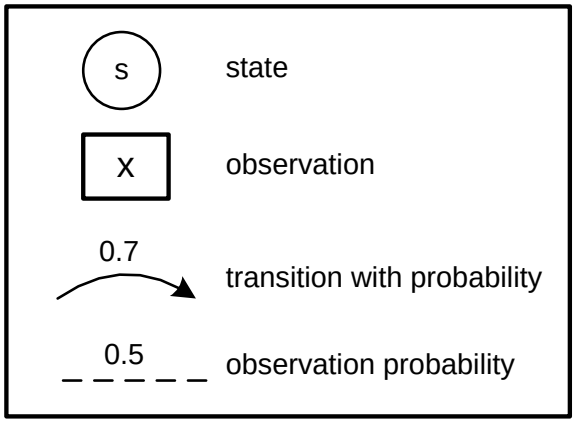
E3

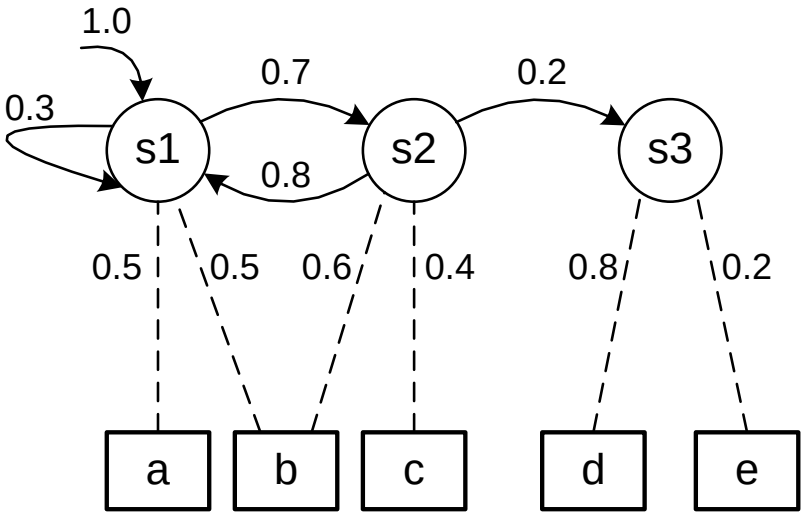


$E2 \Rightarrow E1$ has a confidence of $1/16$

Hidden Markov models

- **Given an observation sequence, how to compute the probability of the sequence given a hidden Markov model?**
- **Given an observation sequence and a hidden Markov model, how to compute the most likely "hidden path" in the model?**
- **Given a set of observation sequences, how to derive the hidden Markov model that maximizes the probability of producing these sequences?**

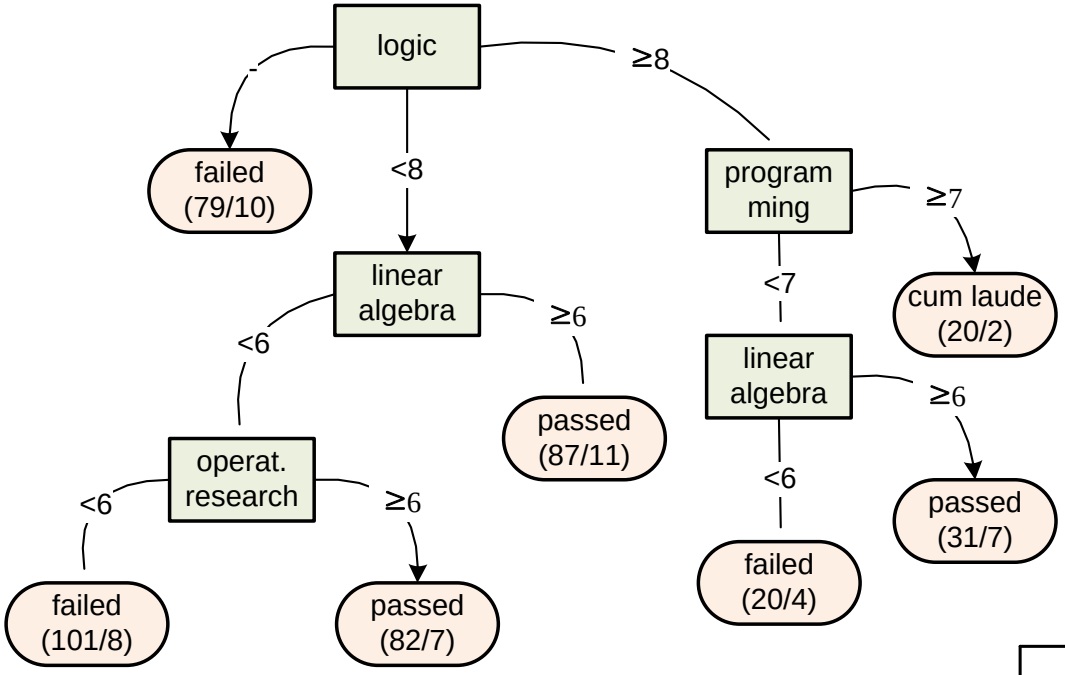




Relation between data mining and process mining

- **Process mining: about end-to-end processes.**
- **Data mining: data-centric and not process-centric.**
- **Judging the quality of data mining and process mining: many similarities, but also some differences.**
- **Clearly, process mining techniques can benefit from experiences in the data mining field.**
- **Let us now focus on the quality of mining results.**

Confusion matrix



		predicted class		
		failed	passed	cum laude
actual class	failed	178	22	0
	passed	21	175	2
	cum laude	1	3	18

Confusion matrix: metrics

		predicted class		
		+	-	
actual class	+	*tp*	*fn*	*p*
	-	*fp*	*tn*	*n*
		p'	*n'*	*N*

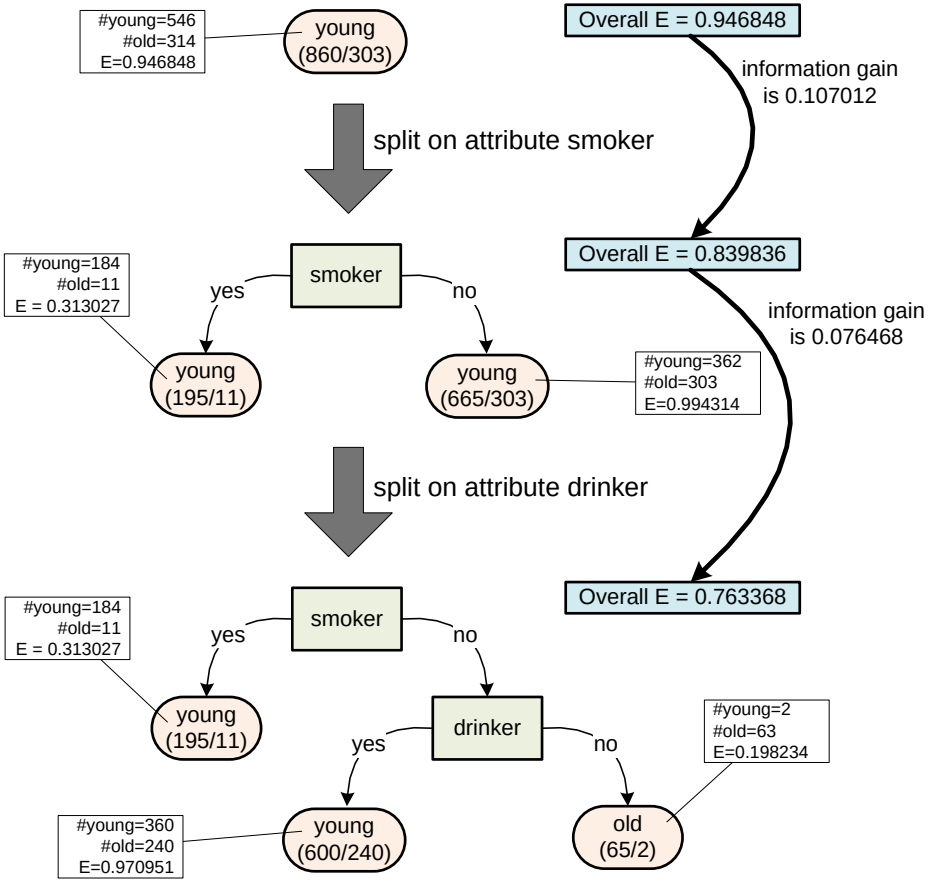
(a)

name	*formula*
error	*(fp+fn)/N*
accuracy	*(tp+tn)/N*
tp-rate	*tp/p*
fp-rate	*fp/n*
precision	*tp/p'*
recall	*tp/p*

(b)

tp is the number of true positives, i.e., instances that are correctly classified as positive.
fn is the number of false negatives, i.e., instances that are predicted to be negative but should have been classified as positive.
fp is the number of false positives, i.e., instances that are predicted to be positive but should have been classified as negative.
tn is the number of true negatives, i.e., instances that are correctly classified as negative.

Example



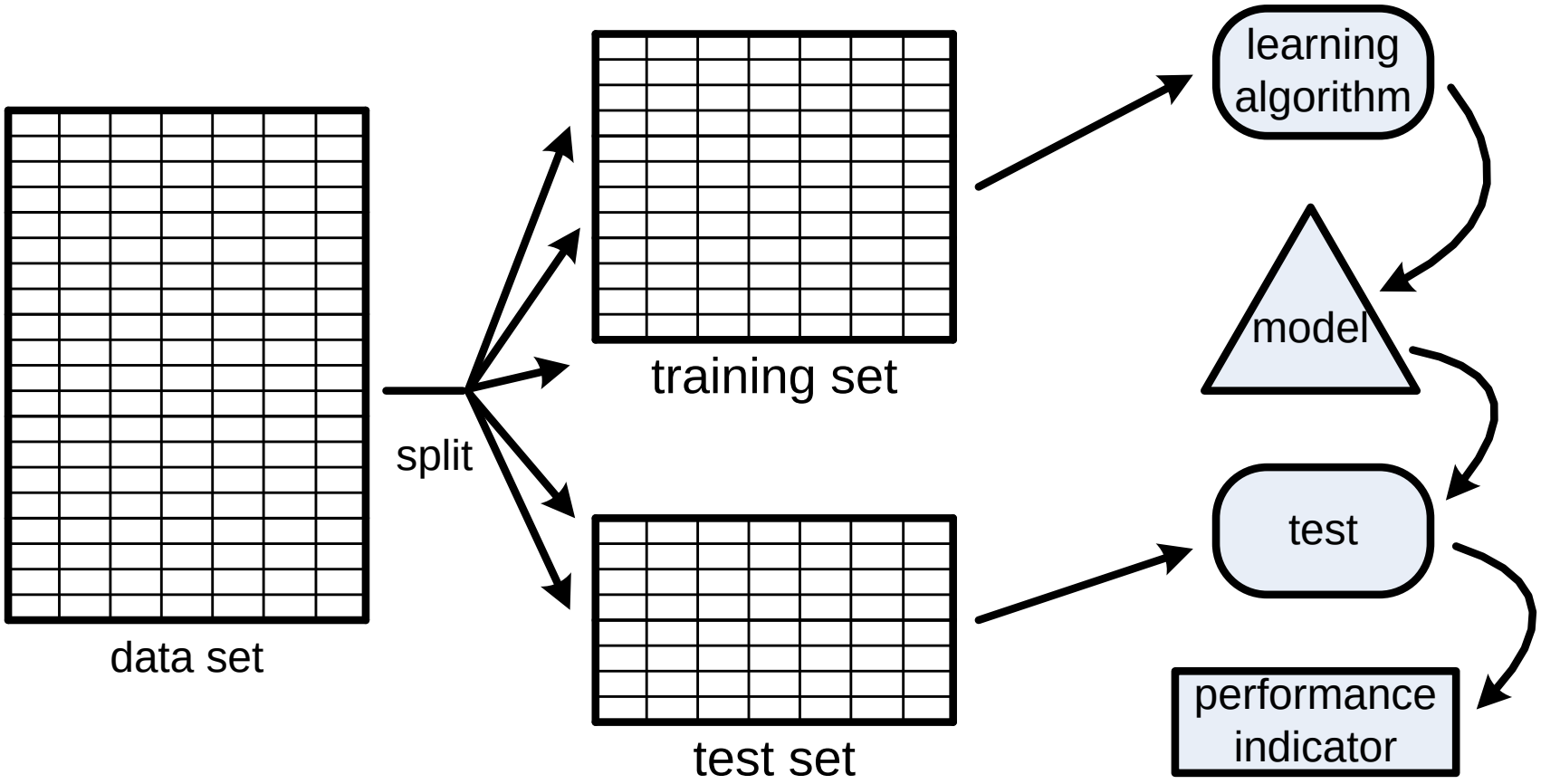
		predicted class	
		young	old
actual class	young	546	0
	old	314	0

(a)

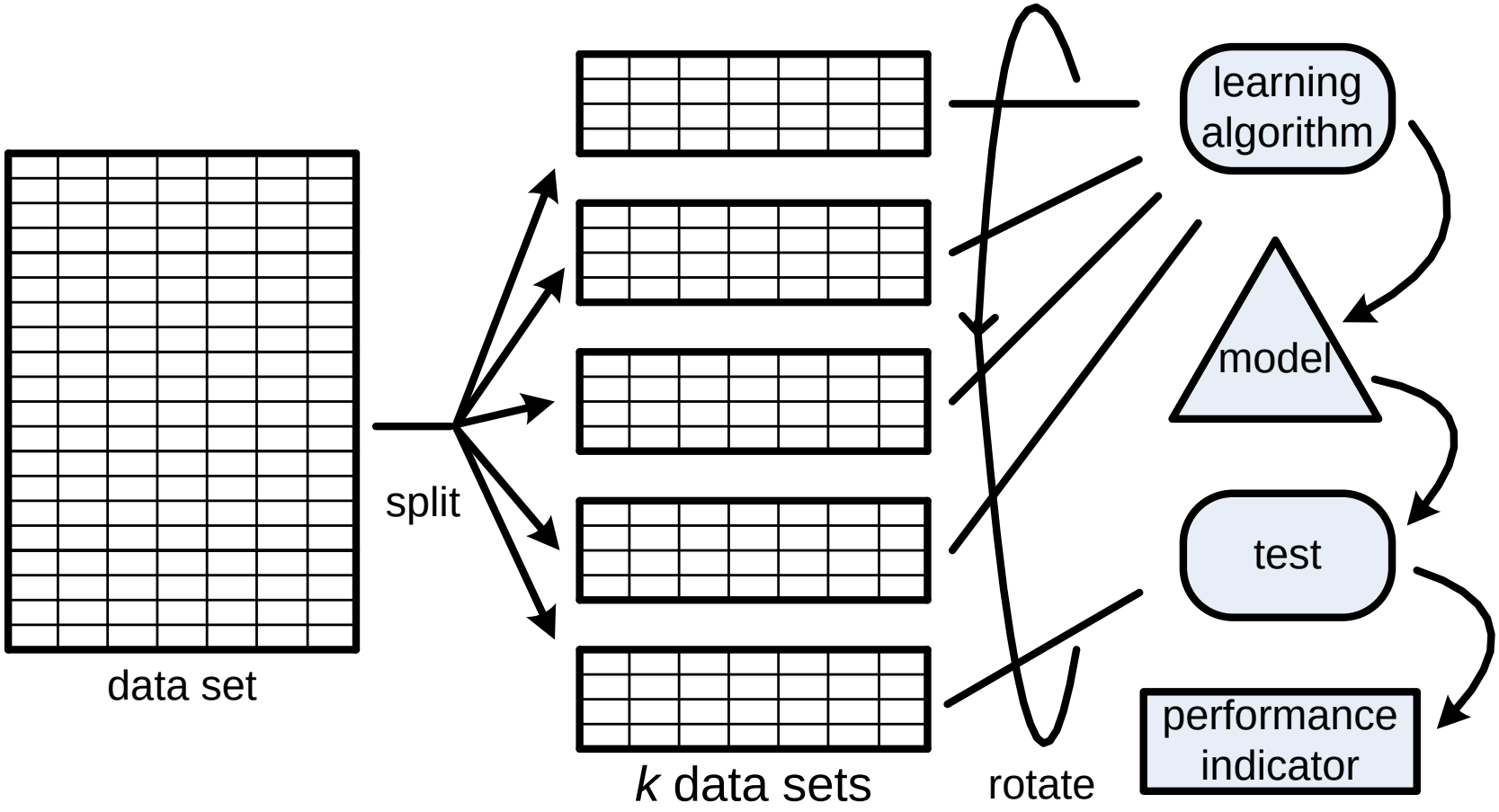
		predicted class	
		young	old
actual class	young	544	2
	old	251	63

(b)

Cross-validation



k-fold cross-validation



Occam's Razor

- Principle attributed to the 14thcentury English logician William of Ockham.
- The principle states that "one should not increase, beyond what is necessary, the number of entities required to explain anything", i.e., one should look for the "simplest model" that can explain what is observed in the data set.
- The Minimal Description Length (MDL) principle tries to operationalize Occam's. In MDL performance is judged on the training data alone and not measured against new, unseen instances. The basic idea is that the "best" model is the one that minimizes the encoding of both model and data set.