

DATA MINING

LECTURE 10

Minimum Description Length

Information Theory

Co-Clustering

MINIMUM DESCRIPTION LENGTH

Occam's razor

- Most data mining tasks can be described as creating a **model** for the data
 - E.g., the EM algorithm models the data as a mixture of Gaussians, the K-means models the data as a set of centroids.
- **What is the right model?**
- **Occam's razor:** All other things being equal, the simplest model is the best.
 - A good principle for life as well

Occam's Razor and MDL

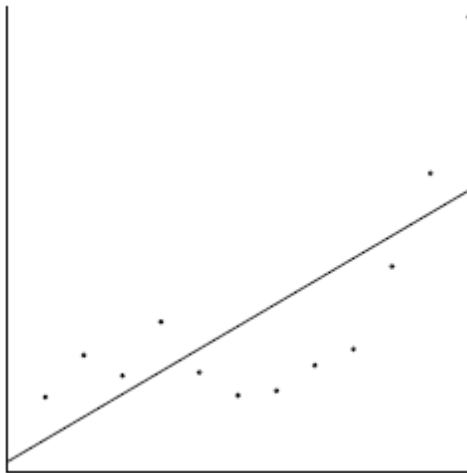
- What is a **simple** model?
- **Minimum Description Length Principle**: Every model provides a (**lossless**) **encoding** of our data. The model that gives the **shortest encoding** (**best compression**) of the data is the best.
 - Related: **Kolmogorov complexity**. Find the shortest program that produces the data (uncomputable).
 - MDL restricts the family of models considered
- Encoding cost: cost of party A to **transmit** to party B the data.

Minimum Description Length (MDL)

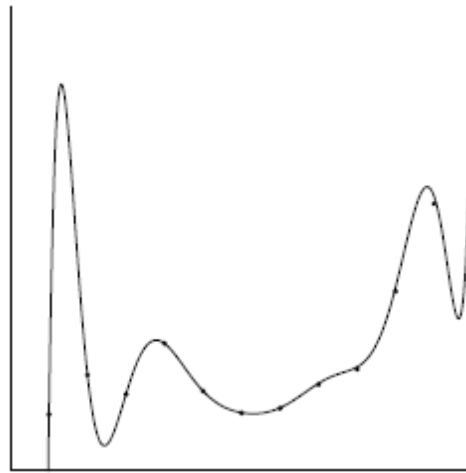
- The description length consists of two terms
 - The cost of **describing the model** (model cost)
 - The cost of **describing the data given the model** (data cost).
 - $L(D) = L(M) + L(D|M)$
- There is a **tradeoff** between the two costs
 - Very **complex models** describe the data in a lot of detail but are **expensive to describe the model**
 - Very **simple models** are cheap to describe but it is **expensive to describe the data** given the model
- This is generic idea for finding the right model
 - We use MDL as a blanket name.

Example

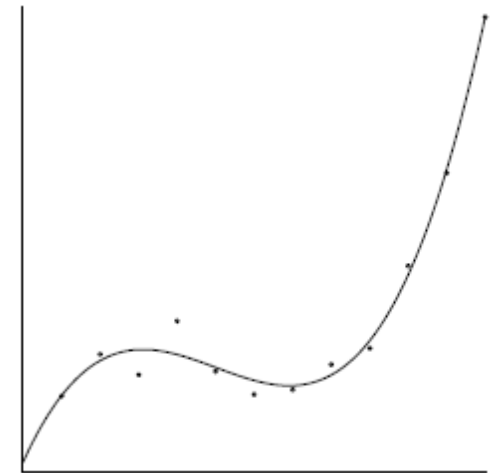
- **Regression**: find a **polynomial** for describing a set of values
 - **Model complexity** (model cost): polynomial coefficients
 - **Goodness of fit** (data cost): difference between real value and the polynomial value



Minimum model cost
High data cost



High model cost
Minimum data cost

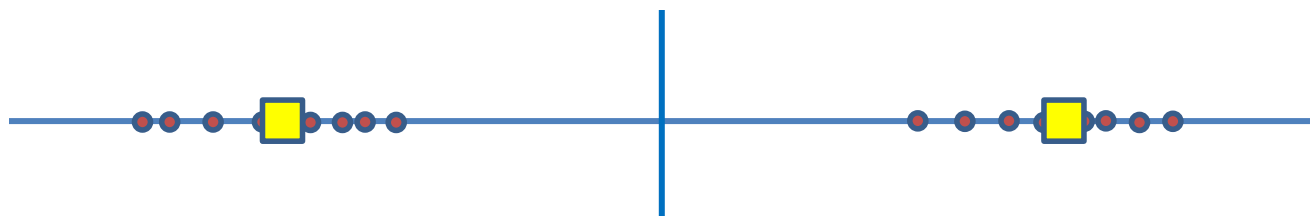


Low model cost
Low data cost

MDL avoids **overfitting** automatically!

Example

- Suppose you want to describe a set of integer numbers
 - Cost of describing a single number is proportional to the value of the number x (e.g., $\log x$).
 - How can we get an efficient description?



- Cluster integers into two clusters and describe the cluster by the centroid and the points by their distance from the centroid
 - **Model cost**: cost of the centroids
 - **Data cost**: cost of cluster membership and distance from centroid
- What are the two extreme cases?

MDL and Data Mining

- Why does the shorter encoding make sense?
 - Shorter encoding implies **regularities** in the data
 - Regularities in the data imply **patterns**
 - Patterns are interesting
- Example

000010000100001000010000100001000010000100001000010000100001000010000100001

- Short description length, just repeat 12 times 00001

0100111001010011011010100001110101111011011010101110010011100

- Random sequence, no patterns, no compression

Is everything about compression?

- Jürgen Schmidhuber: [A theory about creativity, art and fun](#)
 - Interesting Art corresponds to a novel pattern that we cannot compress well, yet it is not too random so we can learn it
 - Good Humor corresponds to an input that does not compress well because it is out of place and surprising
 - Scientific discovery corresponds to a significant compression event
 - E.g., a law that can explain all falling apples.
- Fun lecture:
 - [Compression Progress: The Algorithmic Principle Behind Curiosity and Creativity](#)

Issues with MDL

- What is the right model family?
 - This determines the kind of solutions that we can have
 - E.g., polynomials
 - Clusterings
- What is the encoding cost?
 - Determines the function that we optimize
 - Information theory

INFORMATION THEORY

A short introduction

Encoding

- Consider the following sequence

AAABBBAAACCCABACAABBBAACCCABAC

- Suppose you wanted to encode it in binary form, how would you do it?

50% A

25% B

25% C

A is 50% of the sequence
We should give it a shorter
representation

A → 0

B → 10

C → 11

This is actually provably the best encoding!

Encoding

- **Prefix Codes:** no codeword is a prefix of another

A → 0

B → 10

C → 11

Uniquely directly decodable

For every code we can find a prefix code of equal length

- **Codes and Distributions:** There is one to one mapping between codes and distributions
 - If P is a distribution over a set of elements (e.g., $\{A, B, C\}$) then there exists a (prefix) code C where $L_C(x) = -\lceil \log P(x) \rceil, x \in \{A, B, C\}$
 - For every (prefix) code C of elements $\{A, B, C\}$, we can define a distribution $P(x) = 2^{-C(x)}$
- The code defined has the smallest **average codelength!**

Entropy

- Suppose we have a random variable X that takes n distinct values

$$X = \{x_1, x_2, \dots, x_n\}$$

that have probabilities $P(X) = \{p_1, \dots, p_n\}$

- This defines a code C with $L_C(x_i) = -\lceil \log p_i \rceil$. The average codelength is

$$-\sum_{i=1}^n p_i \lceil \log p_i \rceil$$

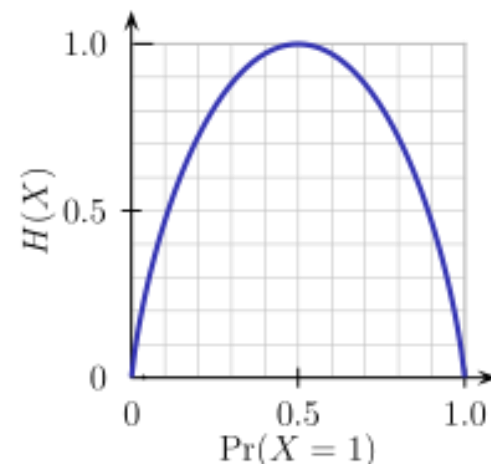
- This (more or less) is the **entropy** $H(X)$ of the random variable X

$$H(X) = -\sum_{i=1}^n p_i \log p_i$$

- **Shannon's theorem**: The entropy is a **lower bound** on the average codelength of any code that encodes the distribution $P(X)$
 - When encoding N numbers drawn from $P(X)$, the best encoding length we can hope for is $N * H(X)$
 - Reminder: **Lossless** encoding

Entropy

$$H(X) = - \sum_{i=1}^n p_i \log p_i$$



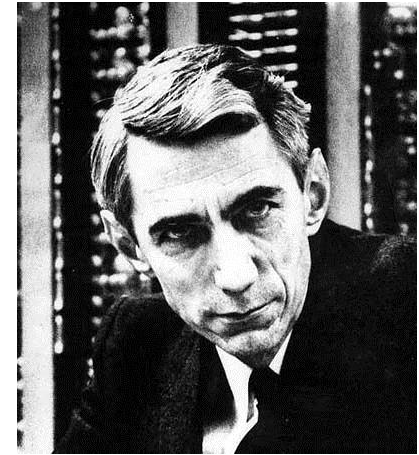
- What does it mean?
- Entropy captures different aspects of a distribution:
 - The **compressibility** of the data represented by random variable X
 - Follows from Shannon's theorem
 - The **uncertainty** of the distribution (highest entropy for uniform distribution)
 - How well can I predict a value of the random variable?
 - The **information content** of the random variable X
 - The number of bits used for representing a value is the information content of this value.

Claude Shannon

Father of Information Theory

Envisioned the idea of communication of information with 0/1 bits

Introduced the word “bit”



The word **entropy** was suggested by **Von Neumann**

- Similarity to physics, but also
- “nobody really knows what entropy really is, so in any conversation you will have an advantage”

Some information theoretic measures

- **Conditional entropy** $H(Y|X)$: the uncertainty for Y given that we know X

$$\begin{aligned} H(Y|X) &= - \sum_x p(x) \sum_y p(y|x) \log p(y|x) \\ &= - \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)} \end{aligned}$$

- **Mutual Information** $I(X,Y)$: The reduction in the uncertainty for Y (or X) given that we know X (or Y)

$$I(X,Y) = H(Y) - H(Y|X) = H(X) - H(X|Y)$$

Some information theoretic measures

- **Cross Entropy**: The cost of encoding distribution P , using the code of distribution Q

$$-\sum_x P(x) \log Q(x)$$

- **KL Divergence** $KL(P||Q)$: The increase in encoding cost for distribution P when using the code of distribution Q

$$KL(P||Q) = -\sum_x P(x) \log Q(x) + \sum_x P(x) \log P(x)$$

- Not symmetric
- Problematic if Q not defined for all x of P .

Some information theoretic measures

- **Jensen-Shannon Divergence** $JS(P,Q)$: distance between two distributions P and Q
 - Deals with the shortcomings of KL-divergence
- If $M = \frac{1}{2} (P+Q)$ is the mean distribution

$$JS(P, Q) = \frac{1}{2} KL(P||M) + \frac{1}{2} KL(Q||M)$$

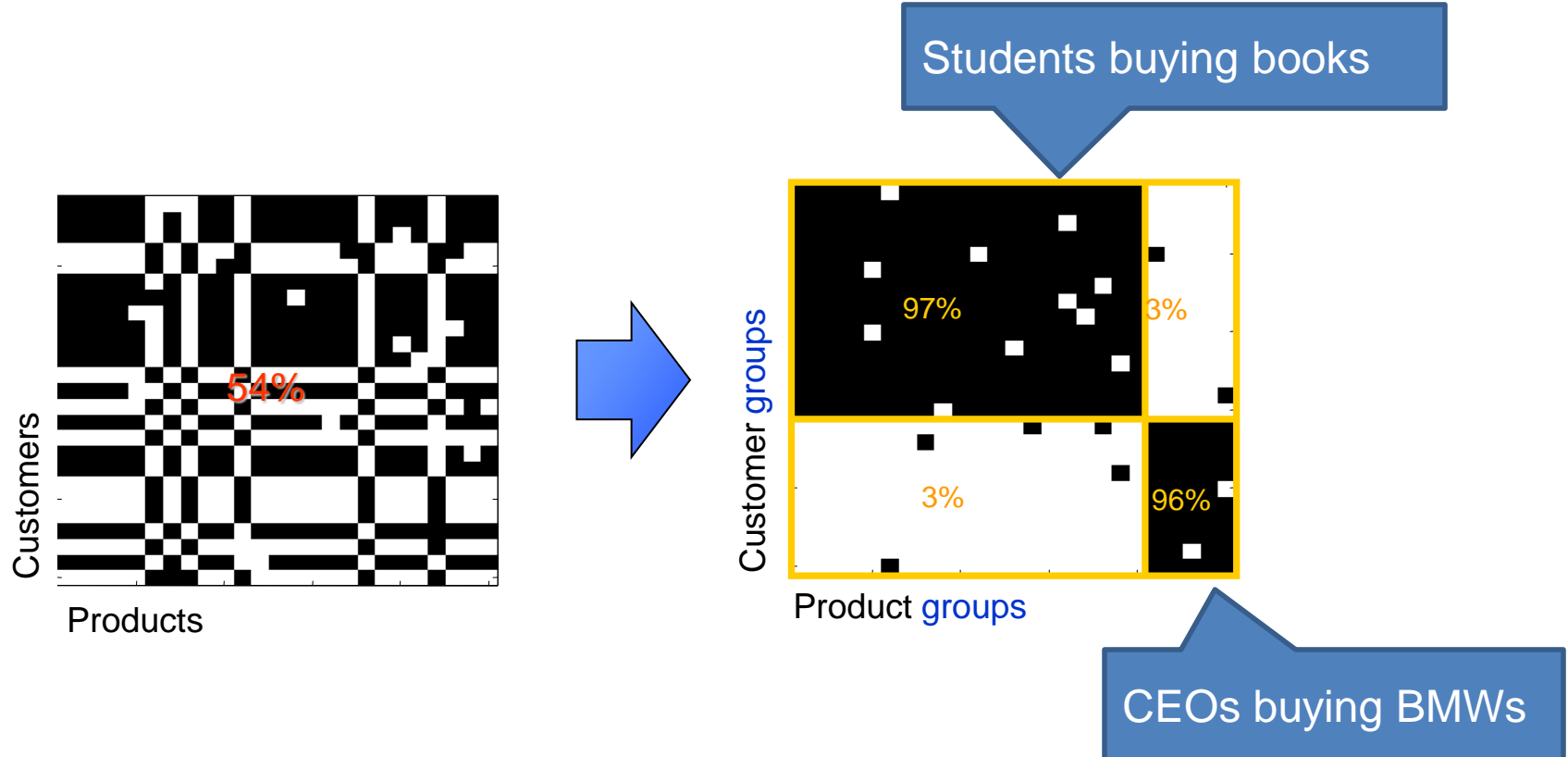
- Jensen-Shannon is a metric

USING MDL FOR CO-CLUSTERING (CROSS-ASSOCIATIONS)

Thanks to Spiros Papadimitriou.

Co-clustering

- Simultaneous grouping of rows and columns of a matrix into homogeneous groups

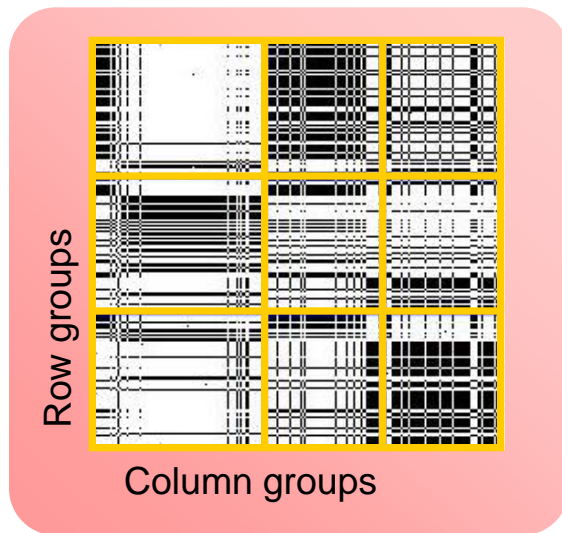


Co-clustering

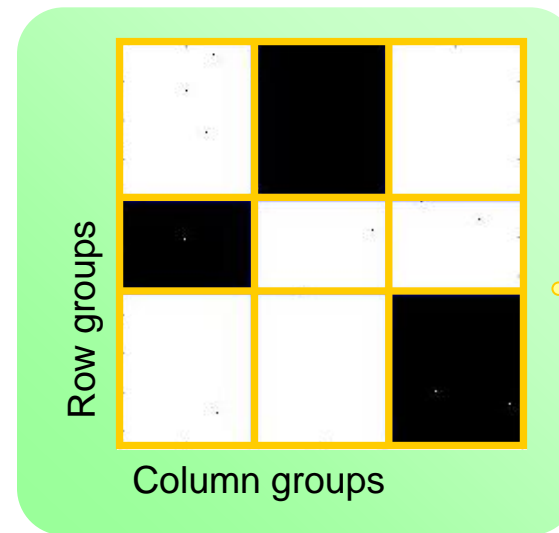
- **Step 1:** How to define a “good” partitioning?
Intuition and formalization
- **Step 2:** How to find it?

Co-clustering

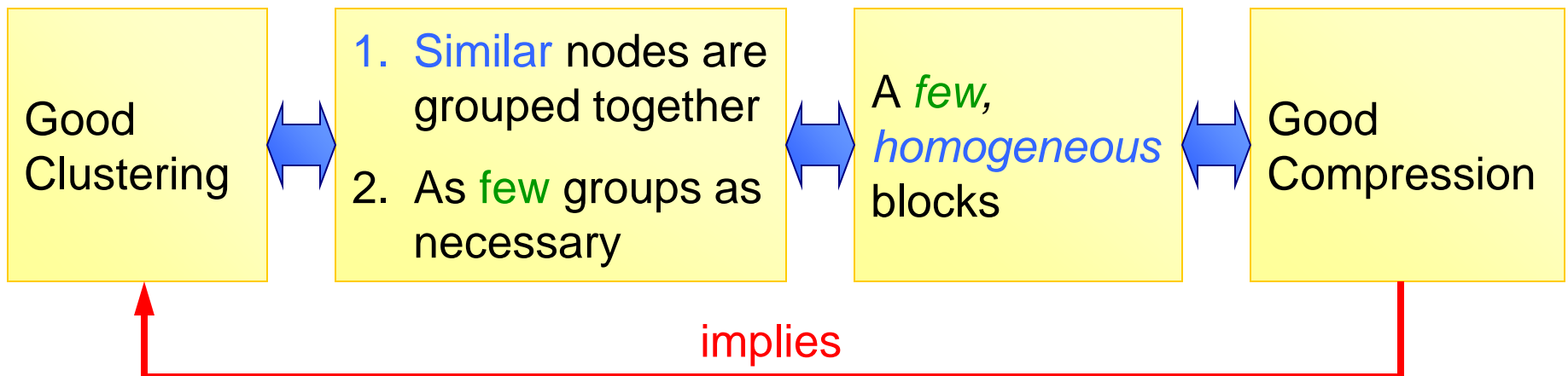
Intuition



versus



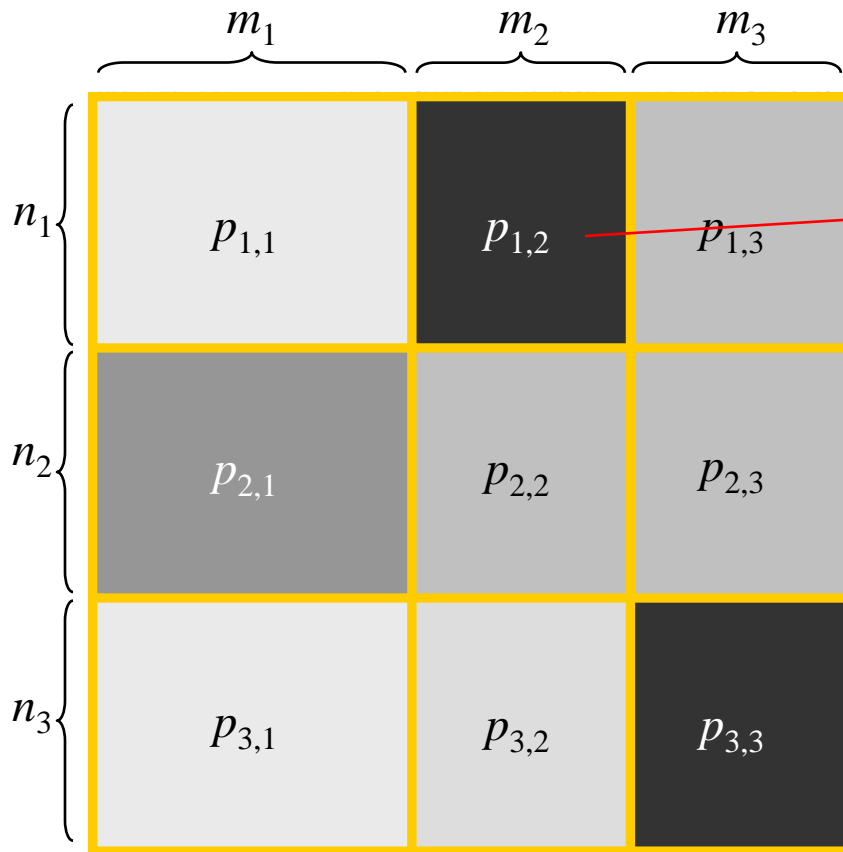
Why is this better?



Co-clustering

MDL formalization—Cost objective

$\ell = 3$ col. groups



$n \times m$ matrix

$$p_{i,j} := \frac{e_{i,j}}{n_i m_j}$$

└ density of ones

$n_1 m_2 H(p_{1,2})$ bits for (1,2)
 block size ← └ ┘ entropy

$$\sum_{i,j} n_i m_j H(p_{i,j}) \quad \text{bits total}$$

data cost

+
 model cost

$$n H\left(\frac{n_1}{n}, \dots, \frac{n_k}{n}\right) + m H\left(\frac{m_1}{m}, \dots, \frac{m_\ell}{m}\right)$$


row-partition description col-partition description

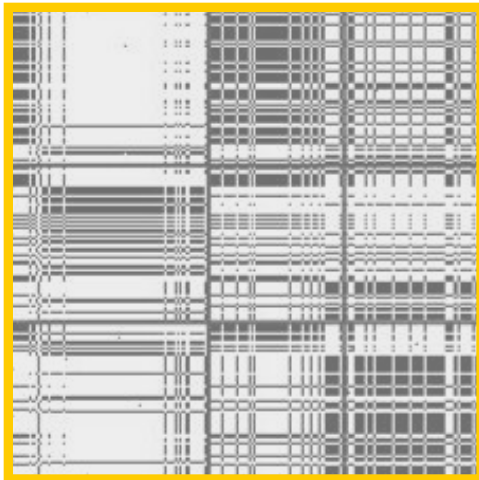
$$+ \log^* k + \log^* \ell + \sum_{i,j} \lceil \log n_i m_j \rceil$$

transmit #partitions transmit #ones $e_{i,j}$

Co-clustering

MDL formalization—Cost objective

one row group 
one col group



high

low

code cost
(block contents)

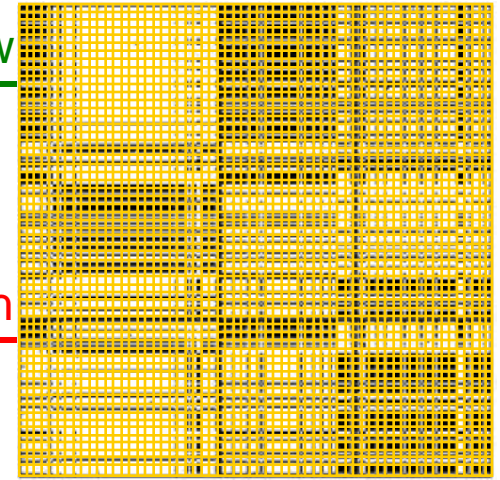
+

description cost
(block structure)

low

high

n row groups 
 m col groups

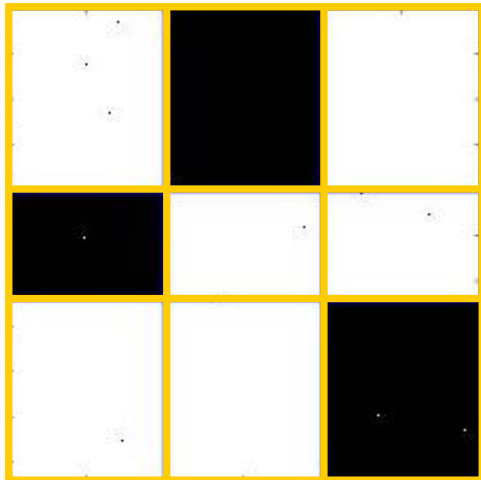


Co-clustering

MDL formalization—Cost objective

$k = 3$ row groups ✓

$\ell = 3$ col groups



low



code cost
(block contents)

+

low

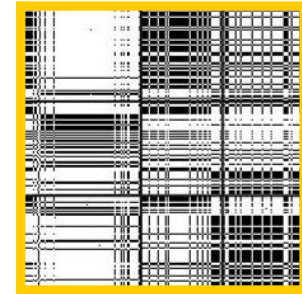
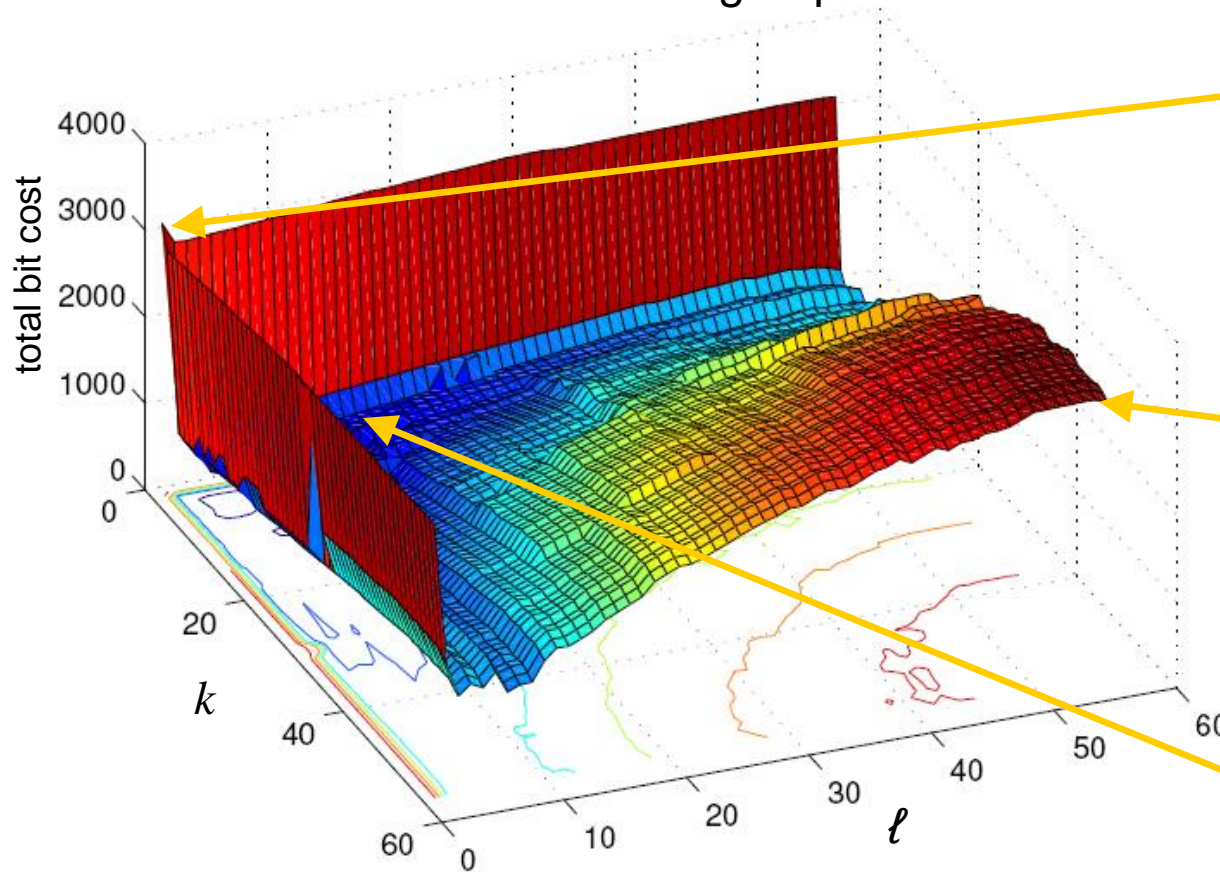


description cost
(block structure)

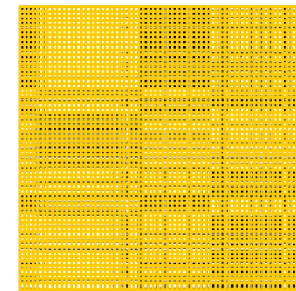
Co-clustering

MDL formalization—Cost objective

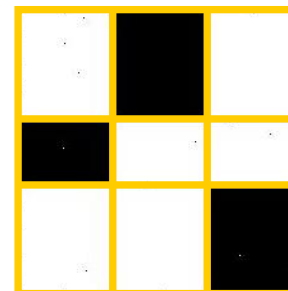
Cost vs. number of groups



one row group
one col group



n row groups
 m col groups



$k = 3$ row groups
 $l = 3$ col groups

Co-clustering

- **Step 1:** How to define a “good” partitioning?
Intuition and formalization
- **Step 2:** How to find it?

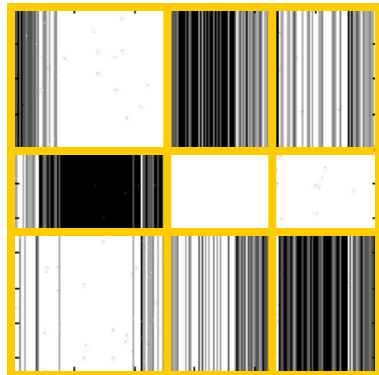
Search for solution

Overview: assignments w/ fixed number of groups (shuffles)

original groups



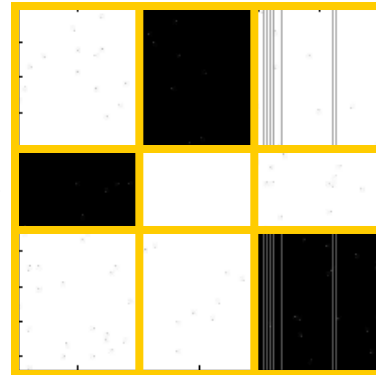
row shuffle



reassign all rows,
holding column
assignments fixed



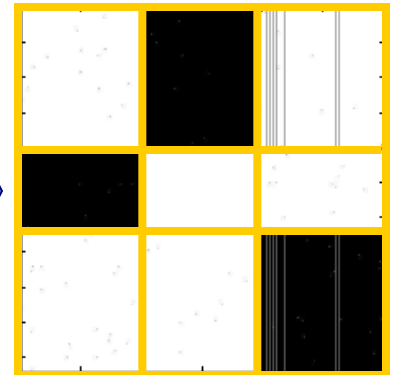
column shuffle



reassign all columns,
holding row
assignments fixed



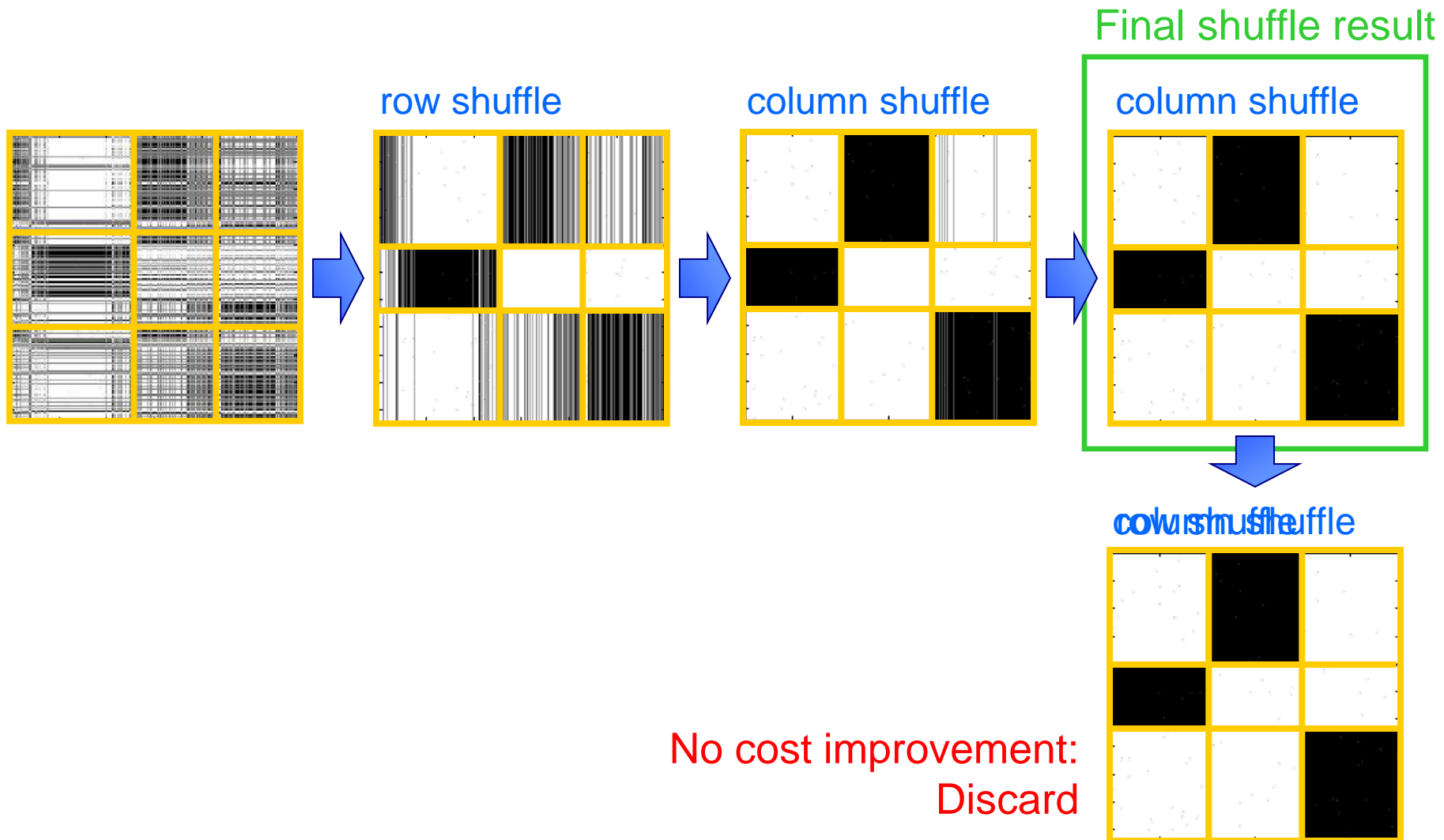
row shuffle



No cost improvement:
Discard

Search for solution

Overview: assignments w/ fixed number of groups (shuffles)



Search for solution

Shuffles

$p_{1,1}$	$p_{1,2}$	$p_{1,3}$
$p_{2,1}$	$p_{2,2}$	$p_{2,3}$
$p_{3,1}$	$p_{3,2}$	$p_{3,3}$

Similarity (“KL-divergences”) of row fragments to blocks of a row group

Assign to second row-group

Iteration

each part that, for all

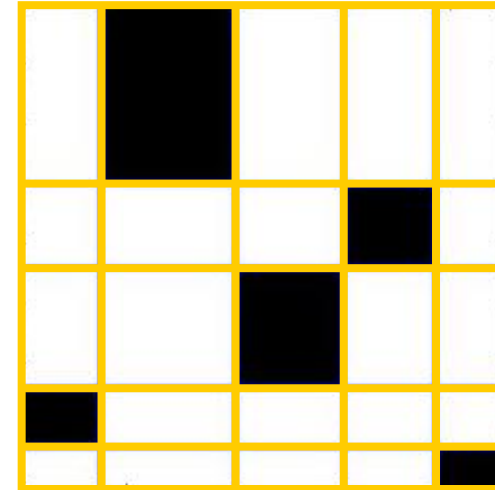
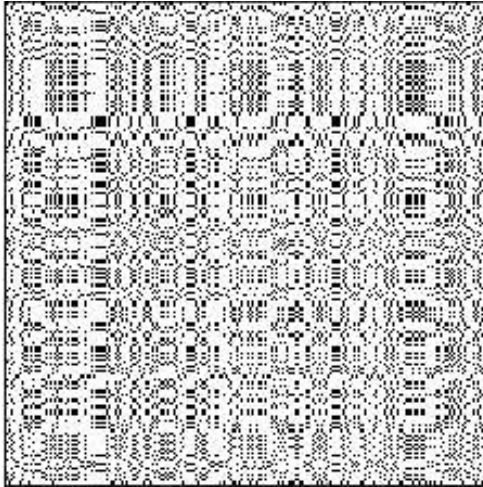
$$- \sum_{j=1}^{\ell} \left(\nu_j \log p_{i^*,j} + (n - \nu_j) \log(1 - p_{i^*,j}) \right)$$

$$\leq - \sum_{j=1}^{\ell} \left(\nu_j \log p_{i,j} + (n - \nu_j) \log(1 - p_{i,j}) \right)$$

Search for solution

Overview: number of groups k and ℓ (splits & shuffles)

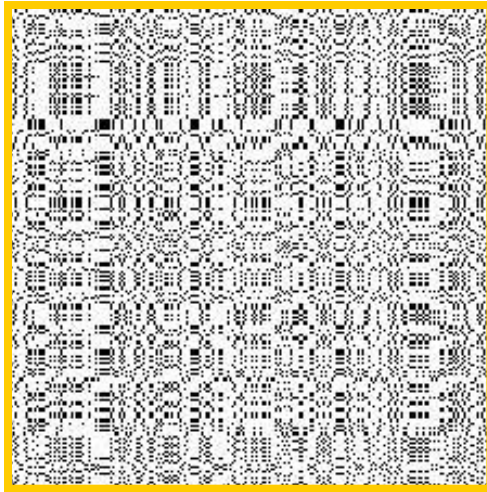
$$k = 5, \ell = 5$$



Search for solution

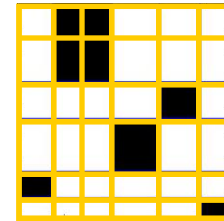
Overview: number of groups k and ℓ (splits & shuffles)

$$k = 1, \ell = 1$$



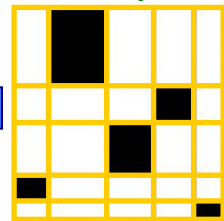
No cost improvement:
Discard

shuffle
col split

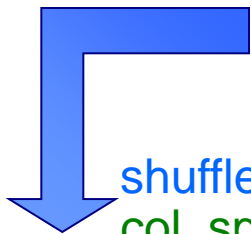


$k = 5, \ell = 5$

shuffle
row split



$k = 5, \ell = 5$

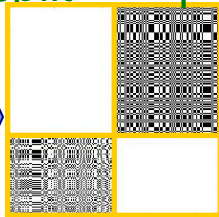


shuffle shuffle
col. split row split



$k=1, \ell=2$

shuffle
col. split



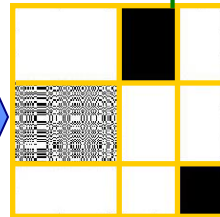
$k=2, \ell=2$

shuffle
col. split



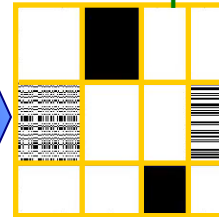
$k=2, \ell=3$

shuffle
row split



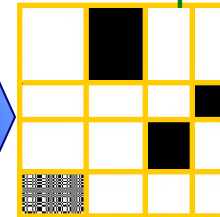
$k=3, \ell=3$

shuffle
col. split



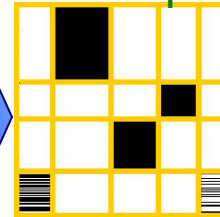
$k=3, \ell=4$

shuffle
row split



$k=4, \ell=4$

shuffle
col. split



$k=4, \ell=5$

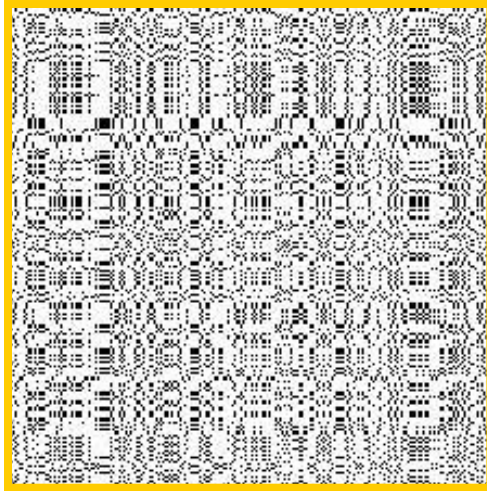
Split:
Increase k or ℓ

Shuffle:
Rearrange rows or cols

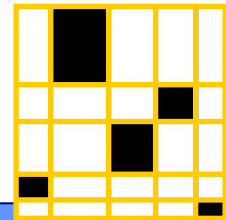
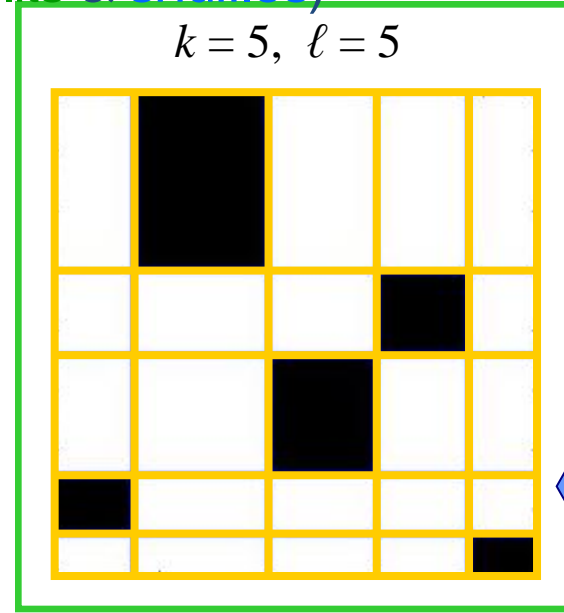
Search for solution

Overview: number of groups k and ℓ (splits & shuffles)

$k = 1, \ell = 1$

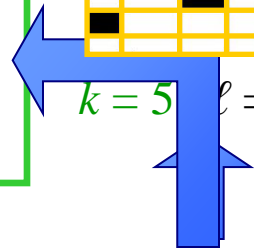
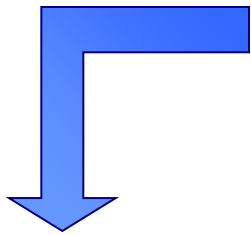


$k = 5, \ell = 5$

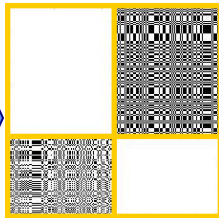


$k = 5, \ell = 5$

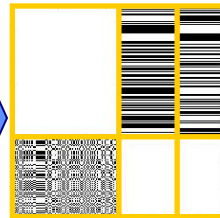
Final result



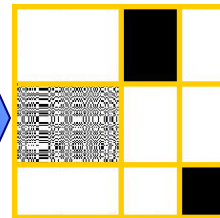
$k=1, \ell=2$



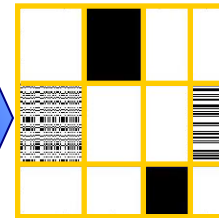
$k=2, \ell=2$



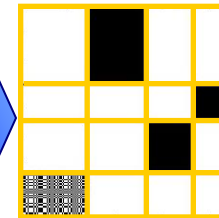
$k=2, \ell=3$



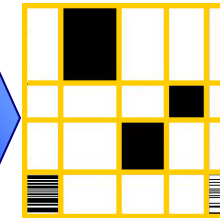
$k=3, \ell=3$



$k=3, \ell=4$



$k=4, \ell=4$



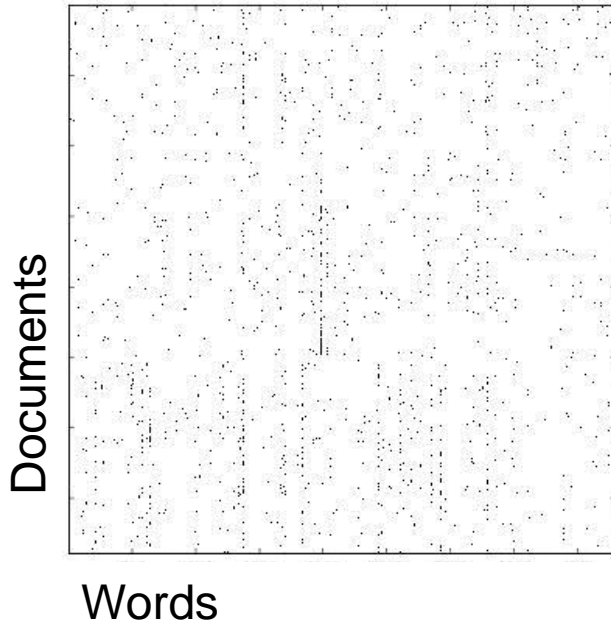
$k=4, \ell=5$

Split:
Increase k or ℓ

Shuffle:
Rearrange rows or cols

Co-clustering

CLASSIC



CLASSIC corpus

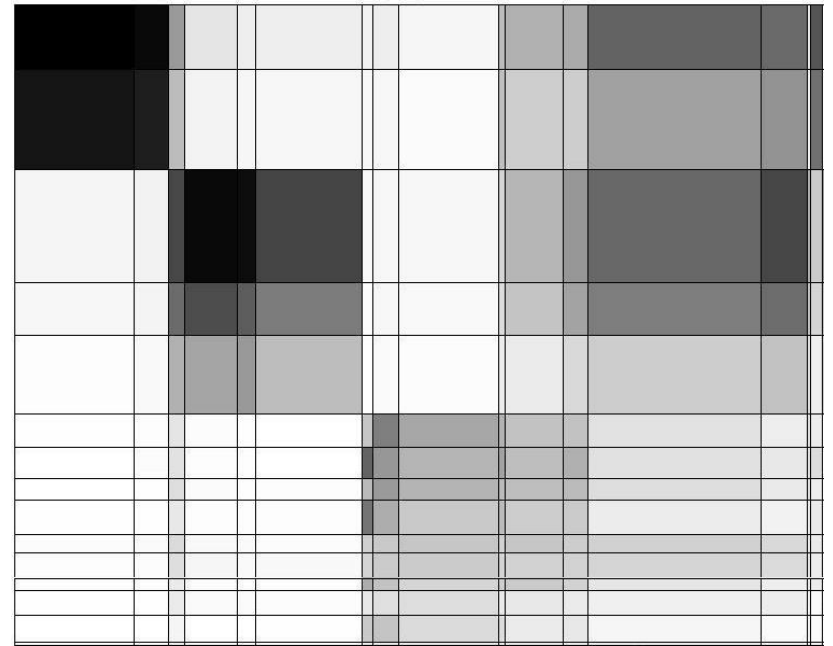
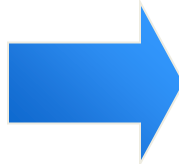
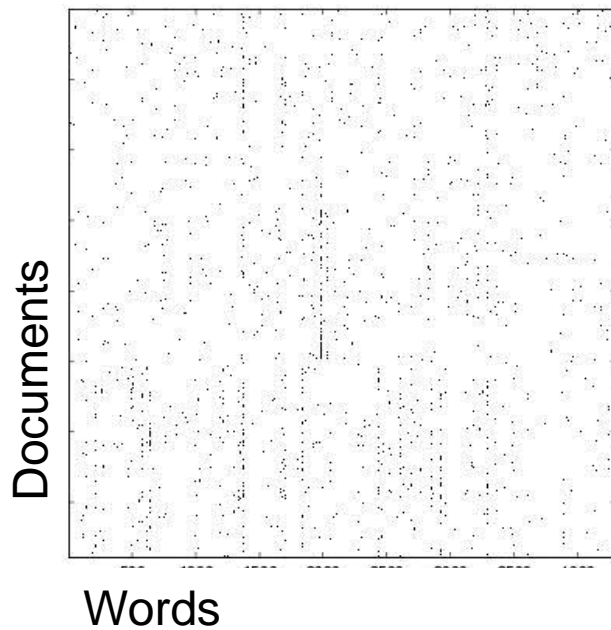
- 3,893 documents
- 4,303 words
- 176,347 “dots” (edges)

Combination of 3 sources:

- **MEDLINE** (medical)
- **CISI** (info. retrieval)
- **CRANFIELD** (aerodynamics)

Graph co-clustering

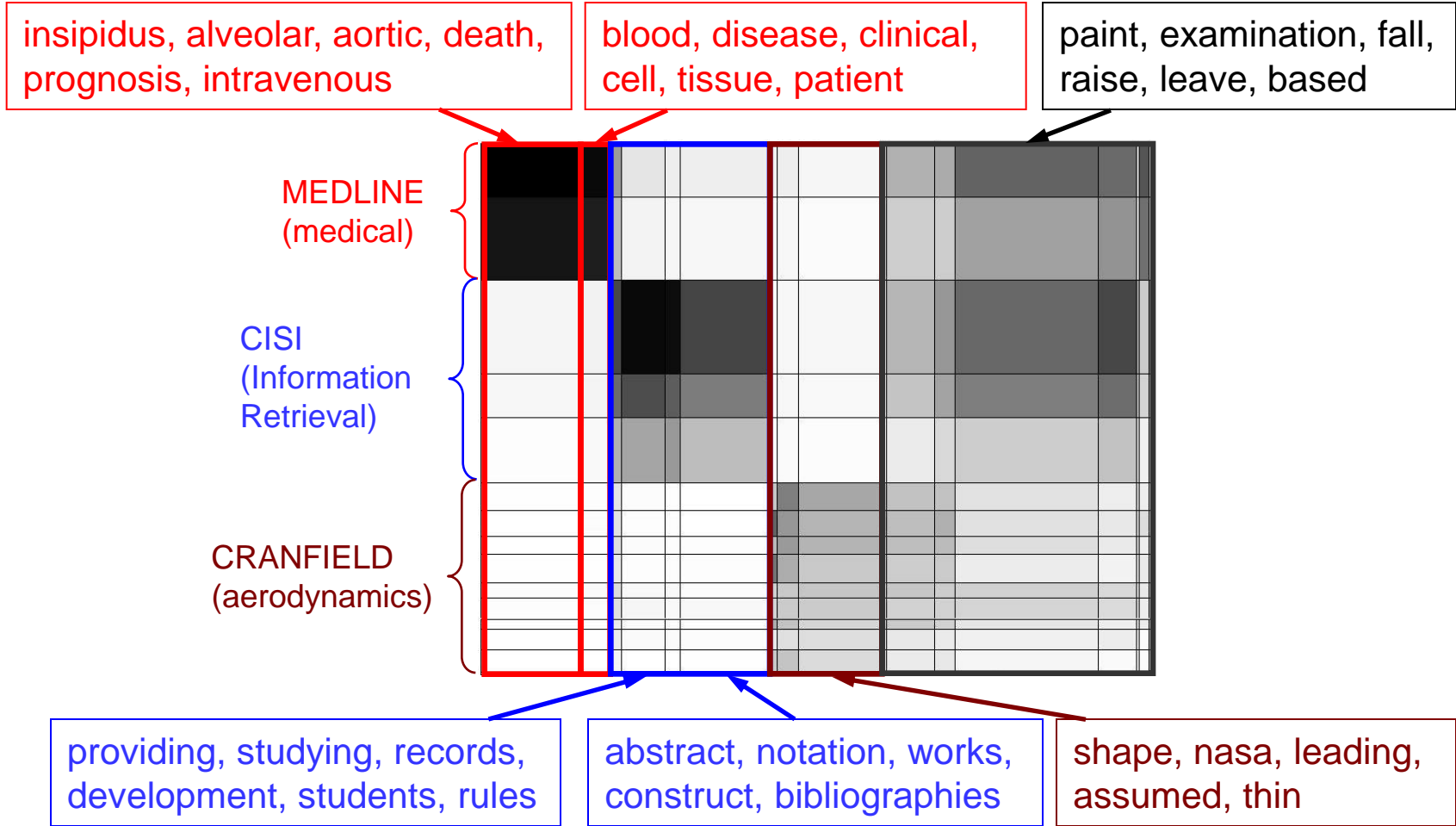
CLASSIC



“CLASSIC” graph of documents & words:
 $k = 15, \ell = 19$

Co-clustering

CLASSIC



“CLASSIC” graph of documents & words:
 $k = 15, \ell = 19$

Co-clustering

CLASSIC

Document cluster #	Document class			Precision
	CRANFIELD	CISI	MEDLINE	
1	0	1	390	0.997
2	0	0	610	1.000
3	2	676	9	0.984
4	1	317	6	0.978
5	3	452	16	0.960
6	207	0	0	1.000
7	188	0	0	1.000
8	131	0	0	1.000
9	209	0	0	1.000
10	107	2	0	0.982
11	152	3	2	0.968
12	74	0	0	1.000
13	139	9	0	0.939
14	163	0	0	1.000
15	24	0	0	1.000
Recall	0.996	0.990	0.968	

0.999

0.975

0.94-1.00

0.987

0.97-0.99