MDL4BMF

— or —

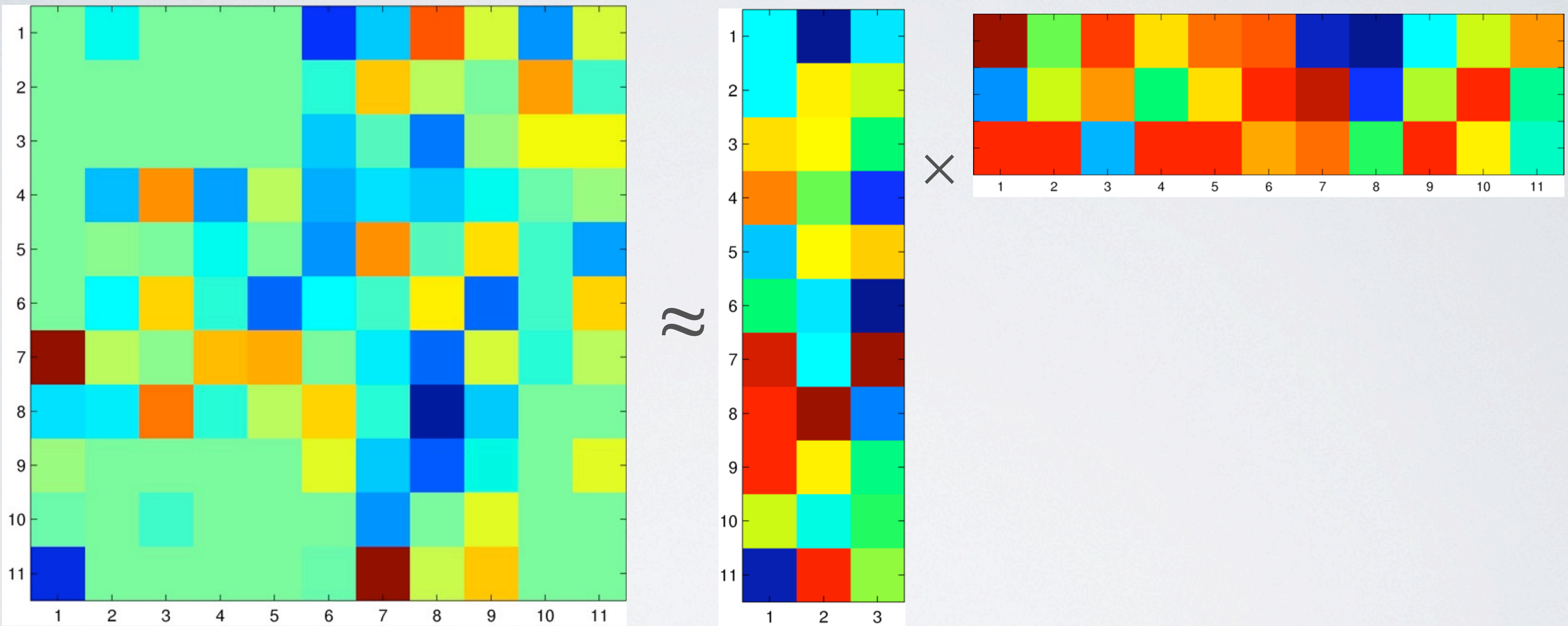
how to use the minimum description length principle
for solving the model order selection problem
for Boolean matrix factorization

Pauli Miettinen & Jilles Vreeken

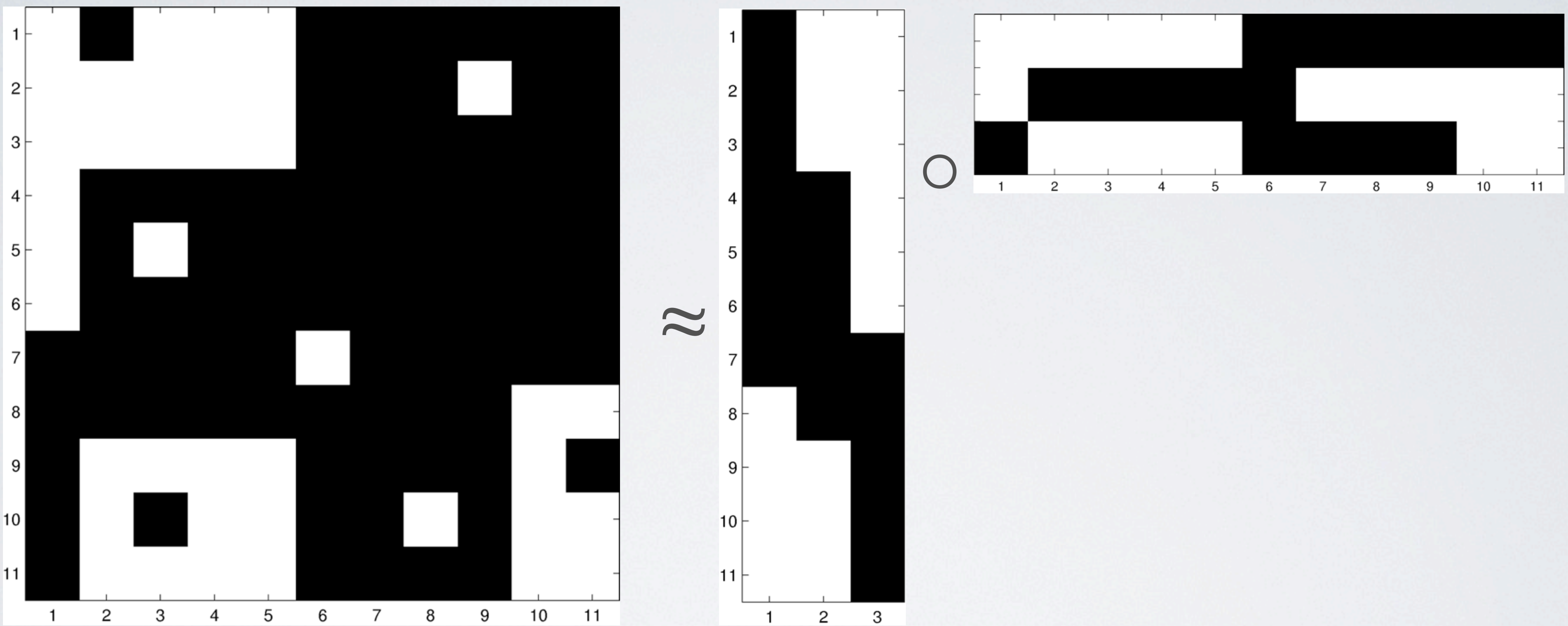
max planck institut
informatik

Universiteit
Antwerpen

MATRIX FACTORIZATIONS



BOOLEAN MATRIX FACTORIZATIONS



max planck institut
informatik

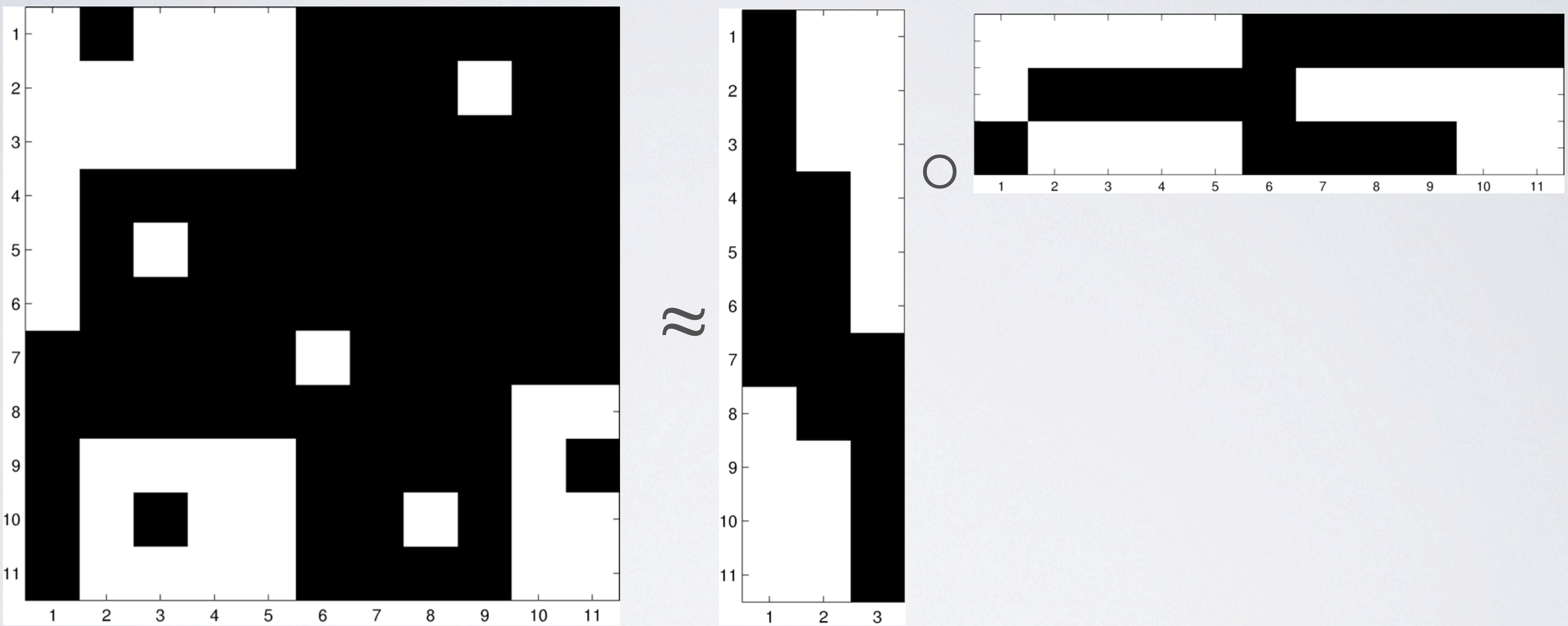
A WORD ABOUT BOOLEAN MATRIX PRODUCT

- As normal matrix product, but with addition defined as 1+1=1 (logical OR)
- Closed under binary matrices
- Corresponds to set union operation

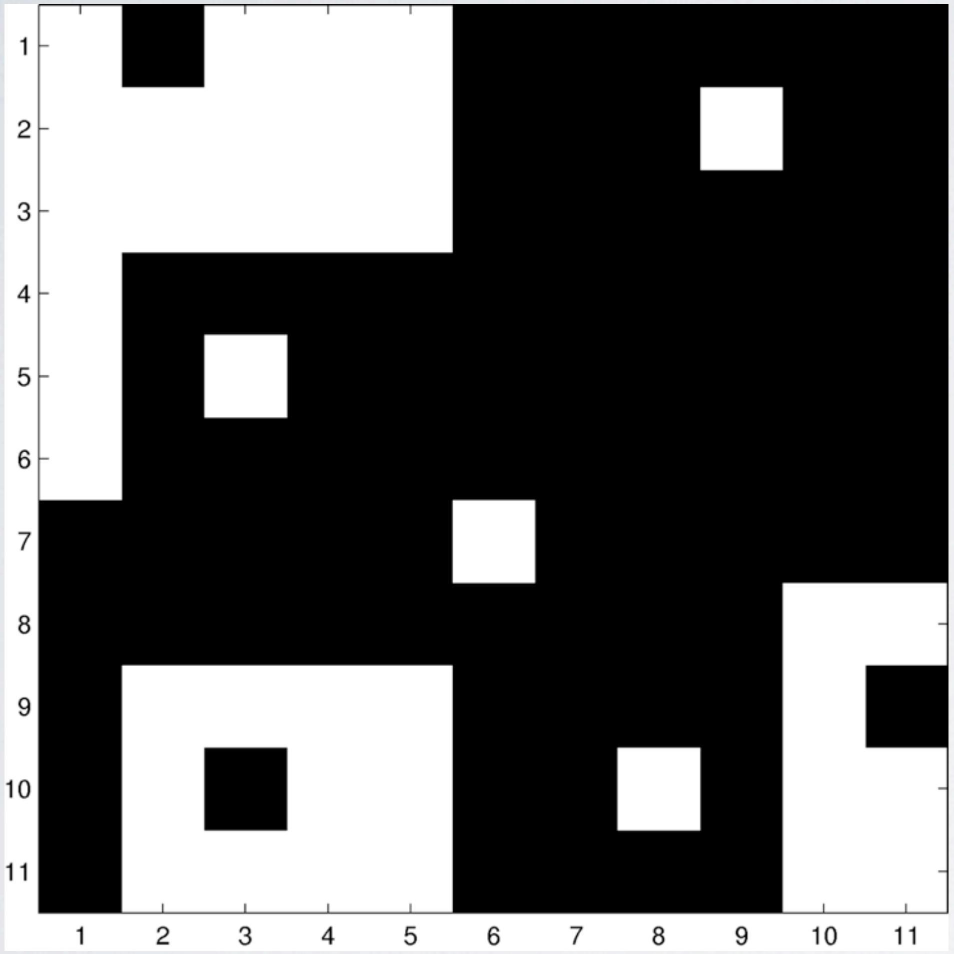
$$(\mathbf{X} \circ \mathbf{Y})_{ij} = \bigvee_{l=1}^{k} x_{il} y_{lj}$$

max planck institut informatik

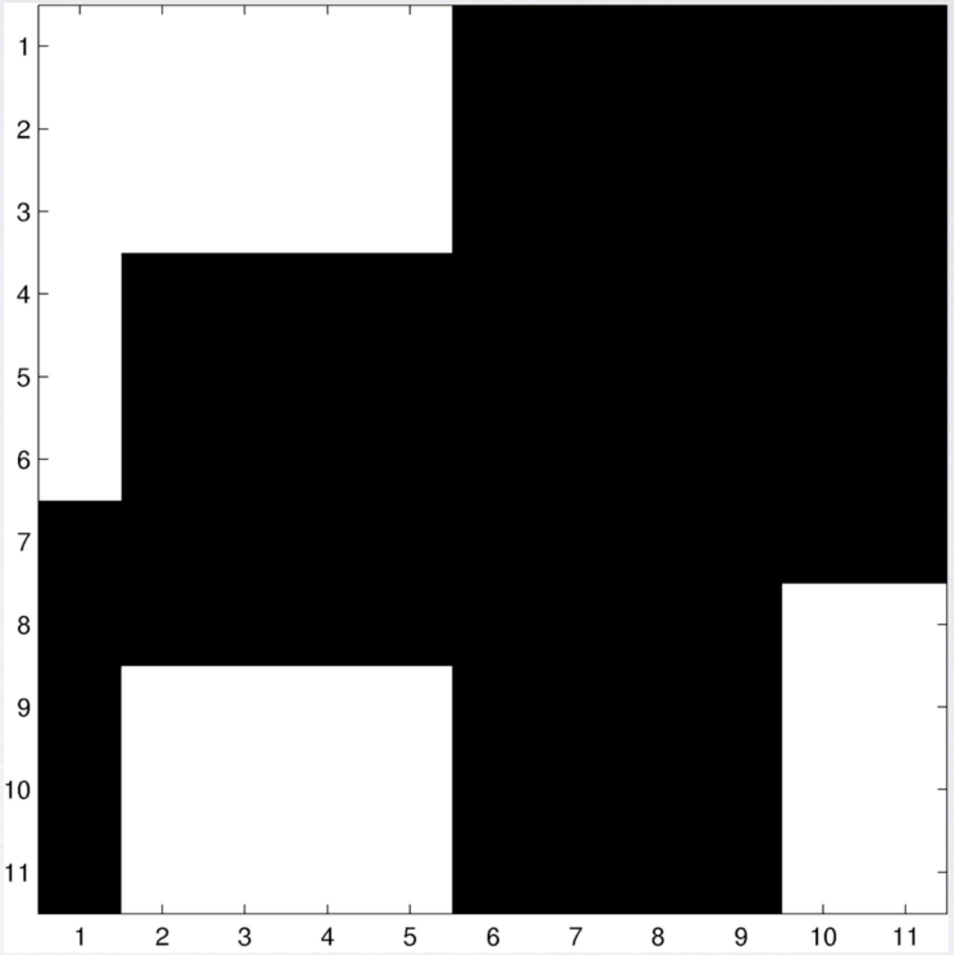
BOOLEAN MATRIX FACTORIZATIONS



BOOLEAN MATRIX FACTORIZATIONS



$\approx$



max planck institut
informatik

BOOLEAN MATRIX FACTORIZATIONS

Definition (BMF). Given an *n*-by-*m* binary matrix **A** and non-negative integer *k*, find *n*-by-*k* binary matrix **B** and *k*-by-*m* binary matrix **C** such that they minimize

$$|\mathbf{A} \otimes (\mathbf{B} \circ \mathbf{C})| = \sum_{i,j} |a_{ij} - (\mathbf{B} \circ \mathbf{C})_{ij}|$$

max planck institut
informatik

BUT WAIT, HOW DO I KNOW WHAT *K* TO USE?

Definition (BMF). Given an *n*-by-*m* binary matrix **A** and non-negative integer *k*, find *n*-by-*k* binary matrix **B** and *k*-by-*m* binary matrix **C** such that they minimize

$$|\mathbf{A} \otimes (\mathbf{B} \circ \mathbf{C})| = \sum_{i,j} |a_{ij} - (\mathbf{B} \circ \mathbf{C})_{ij}|$$

max planck institut informatik

BUT WAIT, HOW DO I KNOW WHAT *K* TO USE?

Definition (BMF). Given an *n*-by-*m* binary matrix **A** and non-negative integer *k*, find *n*-by-*k* binary matrix **B** and *k*-by-*m* binary matrix **C** such that they minimize

$$|\mathbf{A} \otimes (\mathbf{B} \circ \mathbf{C})| = \sum_{i,j} |a_{ij} - (\mathbf{B} \circ \mathbf{C})_{ij}|$$

BUT WAIT, HOW DO I KNOW WHAT *K* TO USE?

Definition (BMF). Given an *n*-by-*m* binary matrix **A** and non-negative integer *k*, find *n*-by-*k* binary matrix **B** and *k*-by-*m* binary matrix **C** such that they minimize

$$|\mathbf{A} \otimes (\mathbf{B} \circ \mathbf{C})| = \sum_{i,j} |a_{ij} - (\mathbf{B} \circ \mathbf{C})_{ij}|$$

N.B. This is nothing special to BMF!

max planck institut
informatik

PRINCIPLES OF GOOD *K*

- **Goal:** Separate noise from structure
- We assume data has BMF-type structure
  - There are *k* factors explaining the BMF structure
  - Rest of the data does not follow the BMF structure (noise)
- But how to decide where structure ends and noise starts?

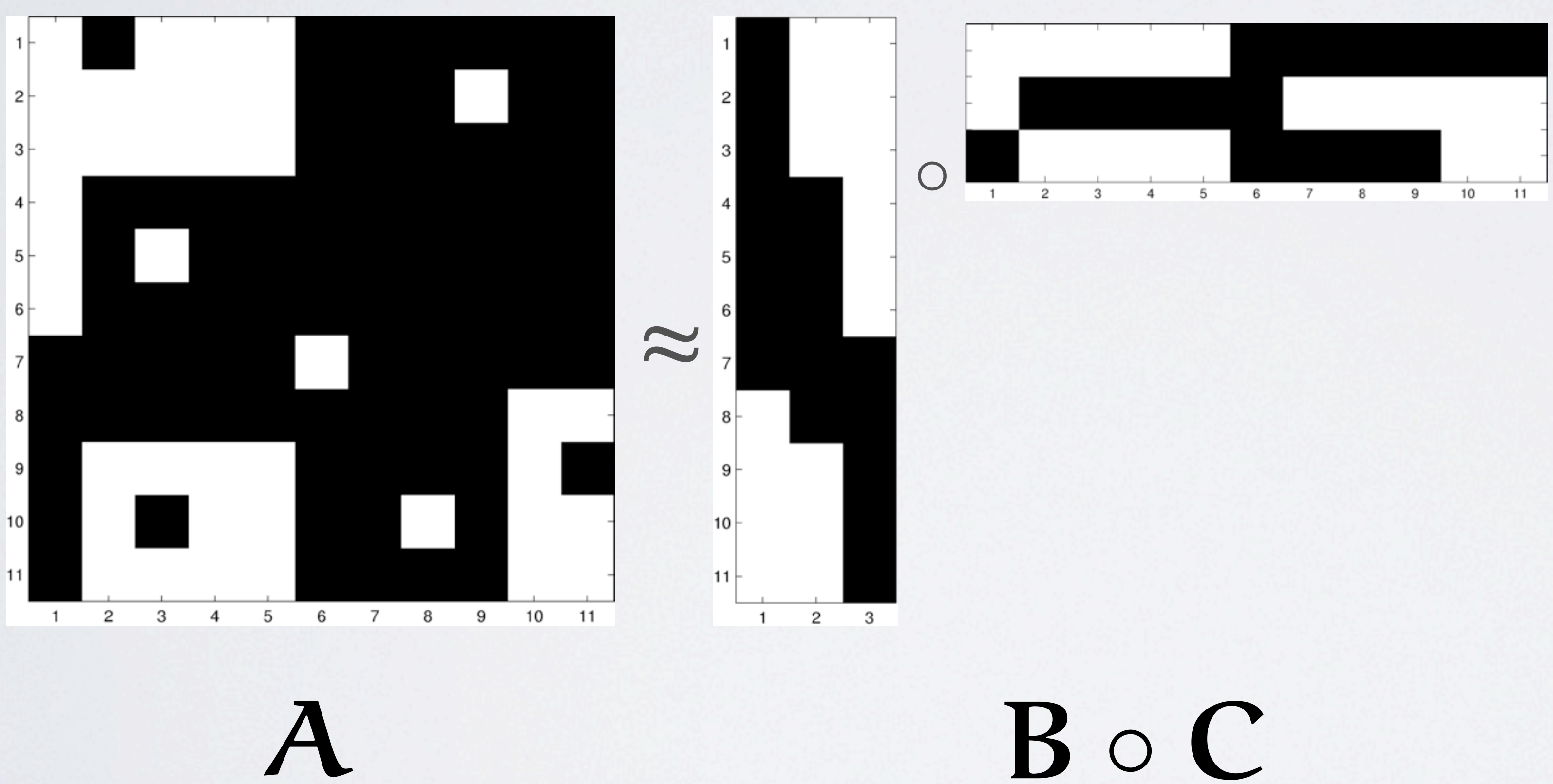
ENTER MDL

THE MINIMUM DESCRIPTION LENGTH PRINCIPLE

- Selecting k = model order selection problem
- The best model (order) is the one that allows us to represent the data with least number of bits
- **Intuition:** Using factor matrices to represent the BMF structure in the data saves space, but using them to represent noise wastes space

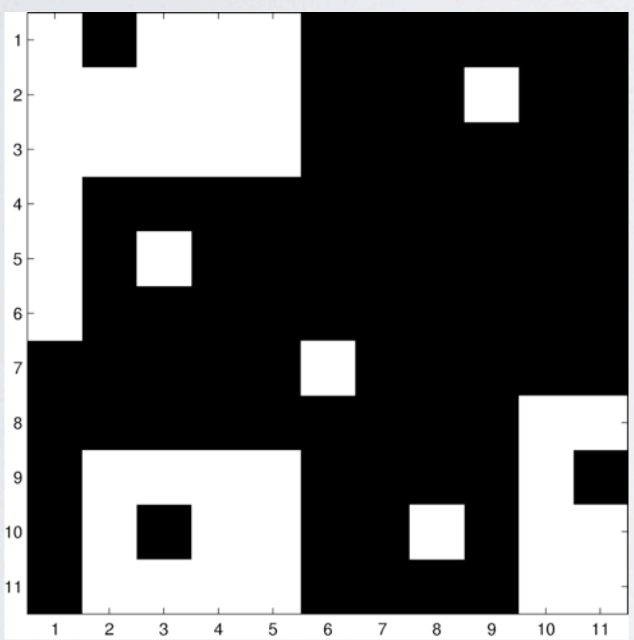
FITTING BMF TO MDL

- MDL requires exact representation

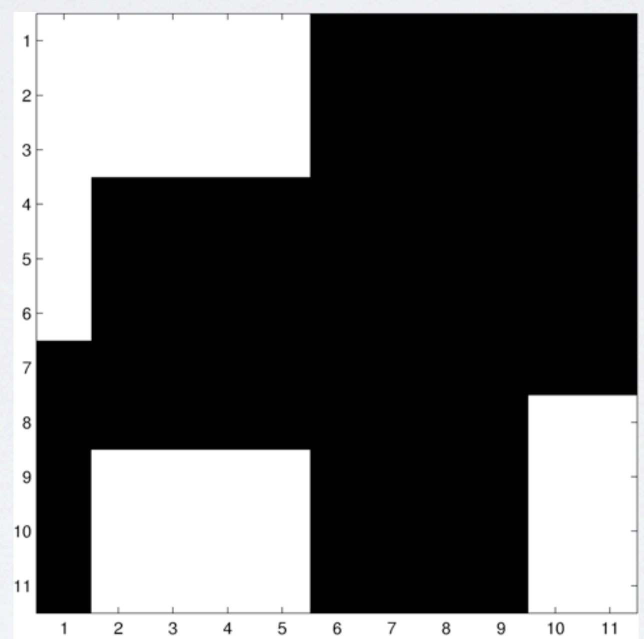


FITTING BMF TO MDL

- MDL requires exact representation



$\approx$

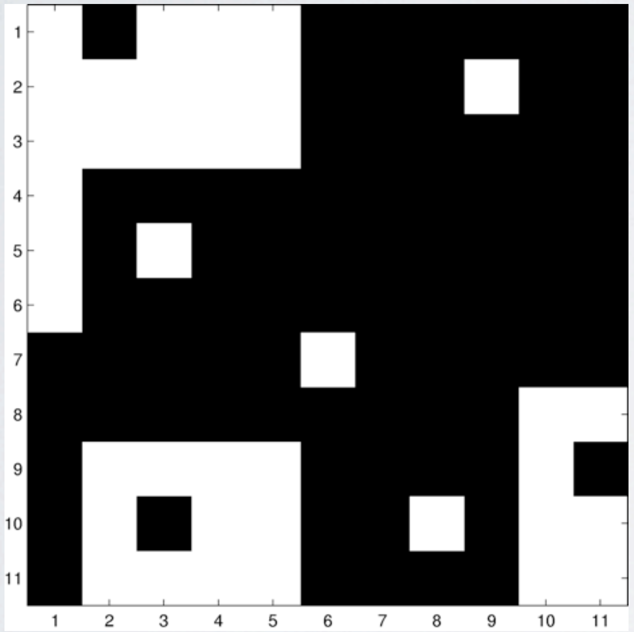


$\mathbf{A}$ $\qquad$ $\mathbf{B} \circ \mathbf{C}$

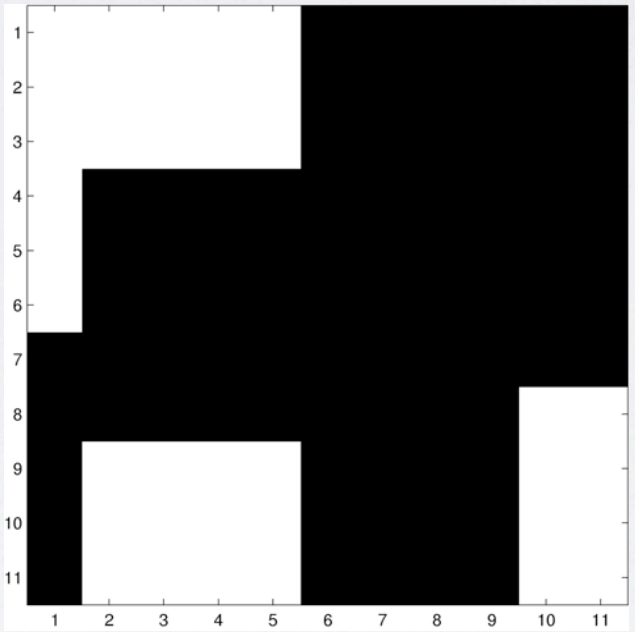
max planck institut informatik

FITTING BMF TO MDL

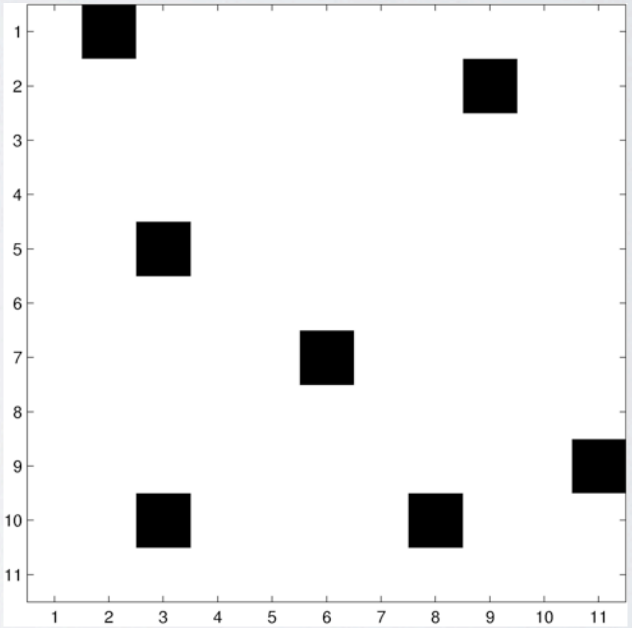
- MDL requires exact representation



$=$



$\otimes$

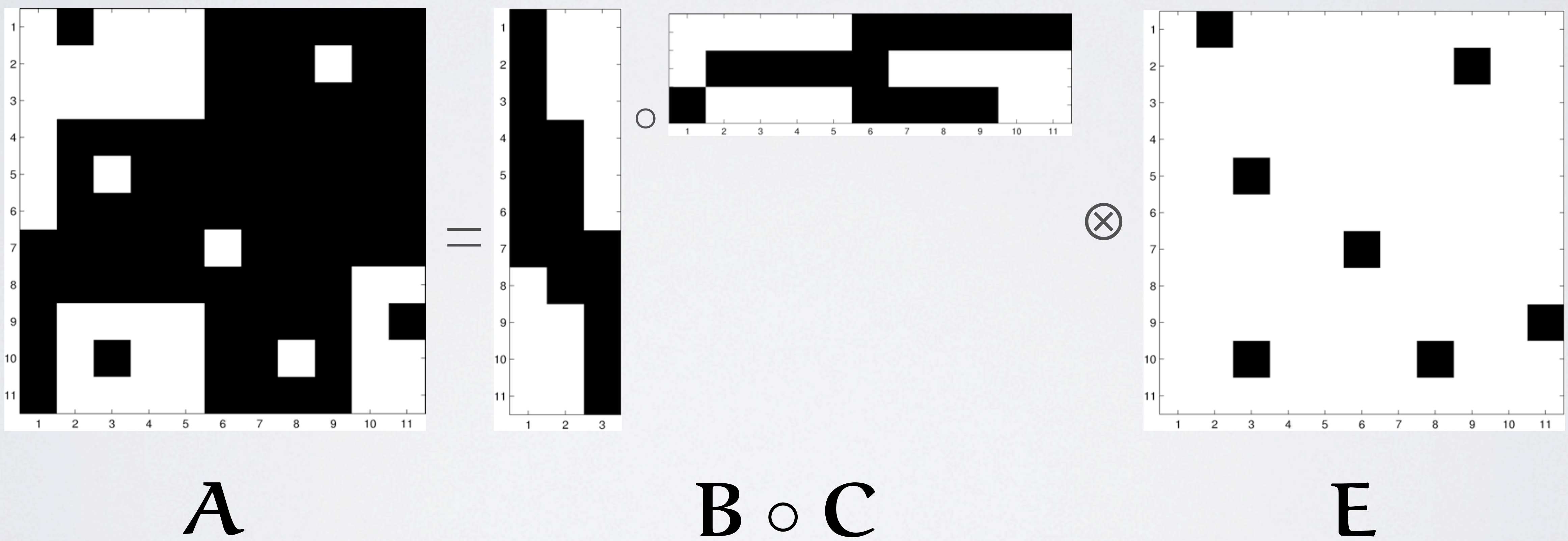


$$\mathbf{A} \qquad \mathbf{B} \circ \mathbf{C} \qquad \mathbf{E}$$

max planck institut
informatik

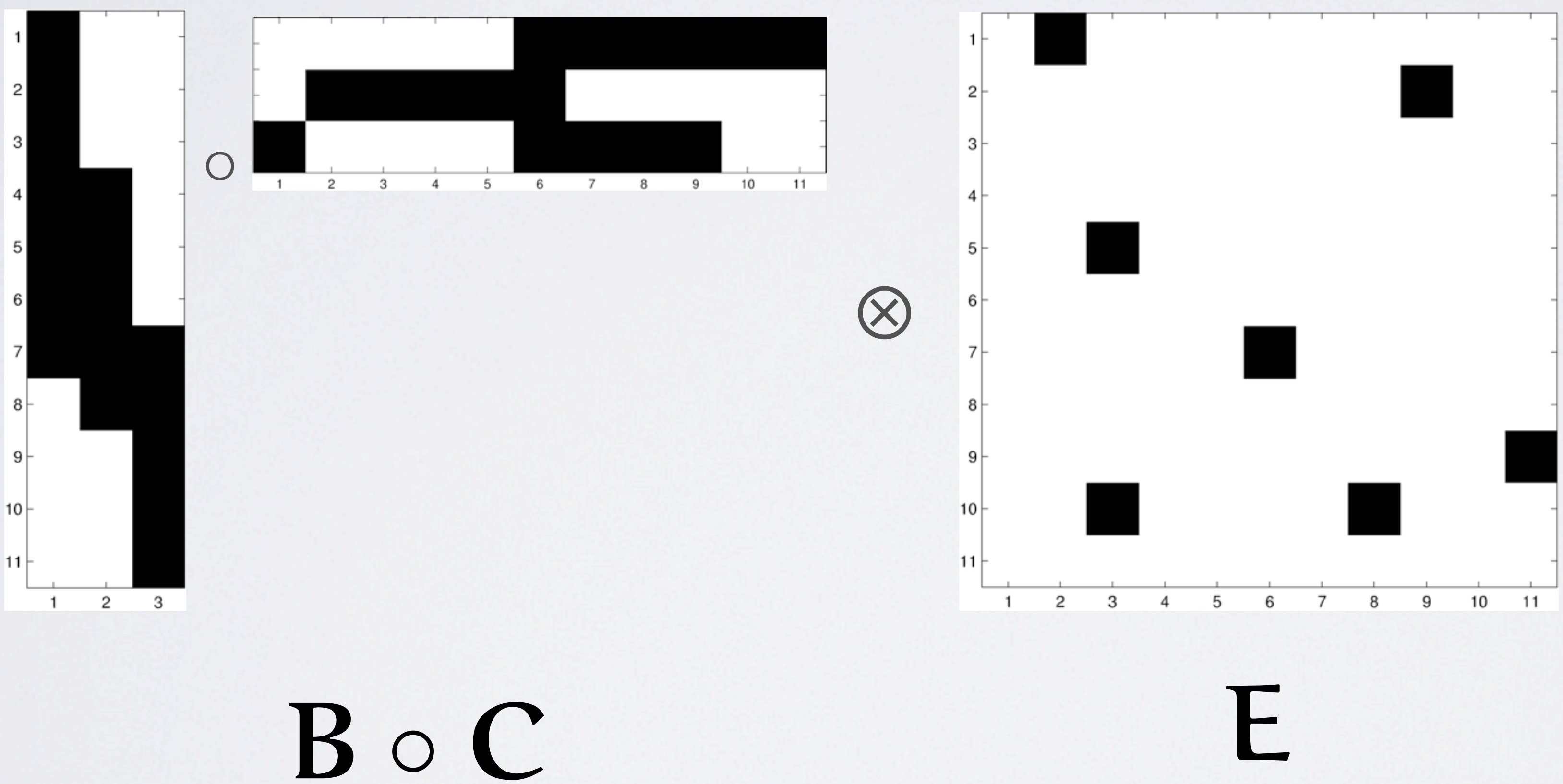
FITTING BMF TO MDL

- MDL requires exact representation



FITTING BMF TO MDL

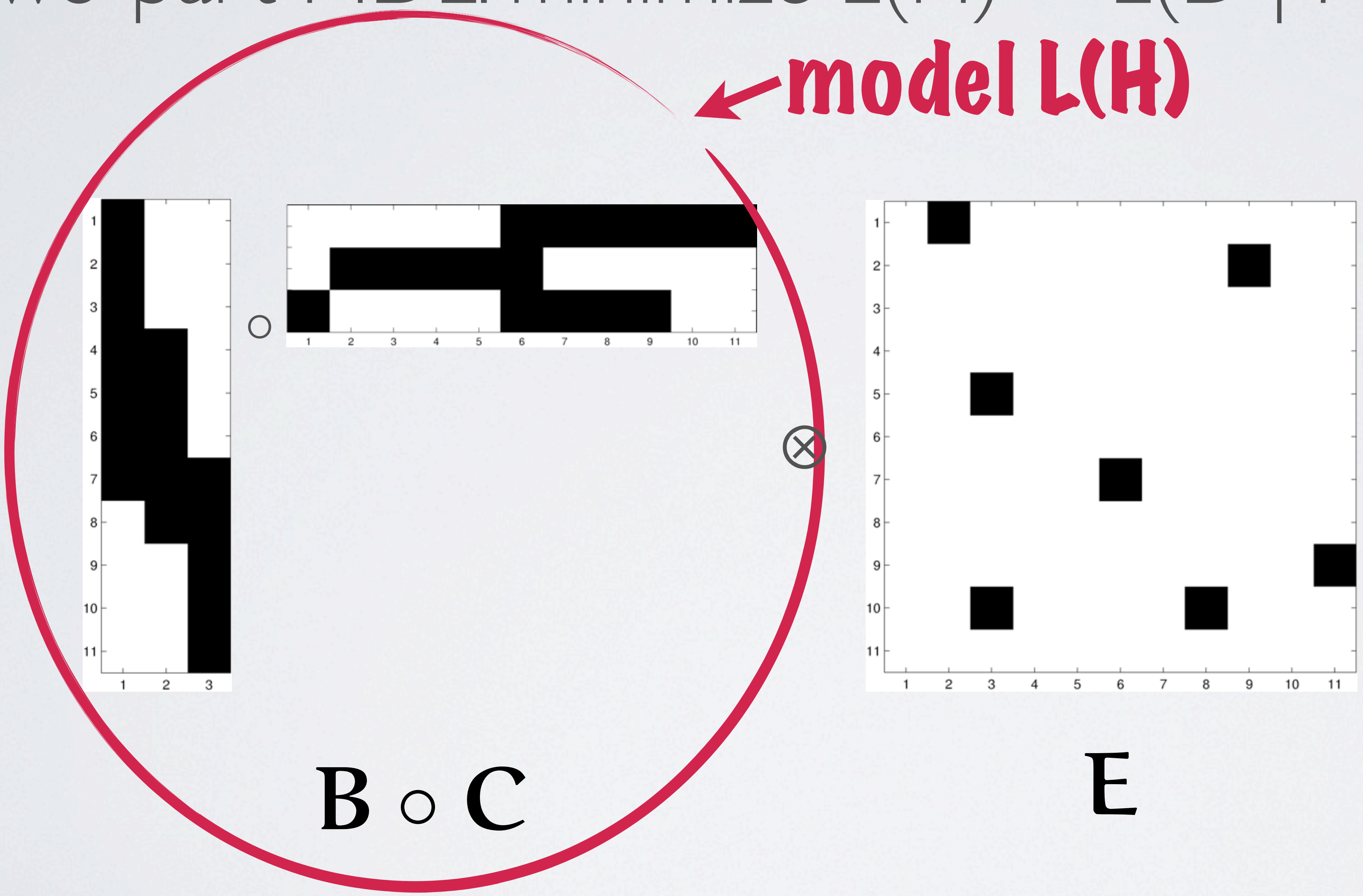
- Two-part MDL: minimize L(H) + L(D | H)



max planck institut informatik

FITTING BMF TO MDL

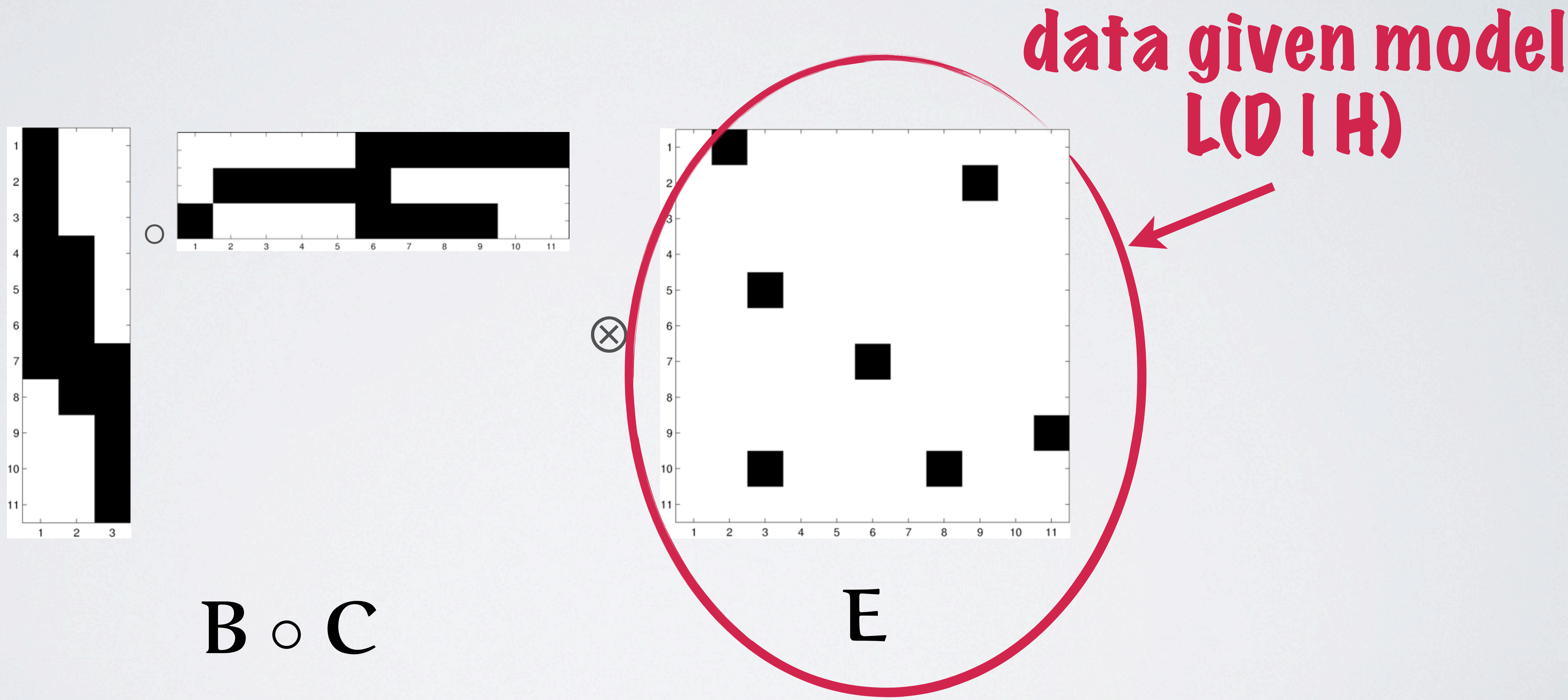
- Two-part MDL: minimize L(H) + L(D | H)



max planck institut
informatik

FITTING BMF TO MDL

- Two-part MDL: minimize L(H) + L(D | H)



ENCODING THE MODEL

- Model includes factor matrices **B** and **C** and their dimensions (*n*, *m*, and *k*)
- Each factor (row of **B** and column of **C**) is encoded using an optimal prefix code

ENCODING THE MODEL

- Model includes factor matrices **B** and **C** and their dimensions (*n*, *m*, and *k*)
- Each factor (row of **B** and column of **C**) is encoded using an optimal prefix code

max planck institut
informatik

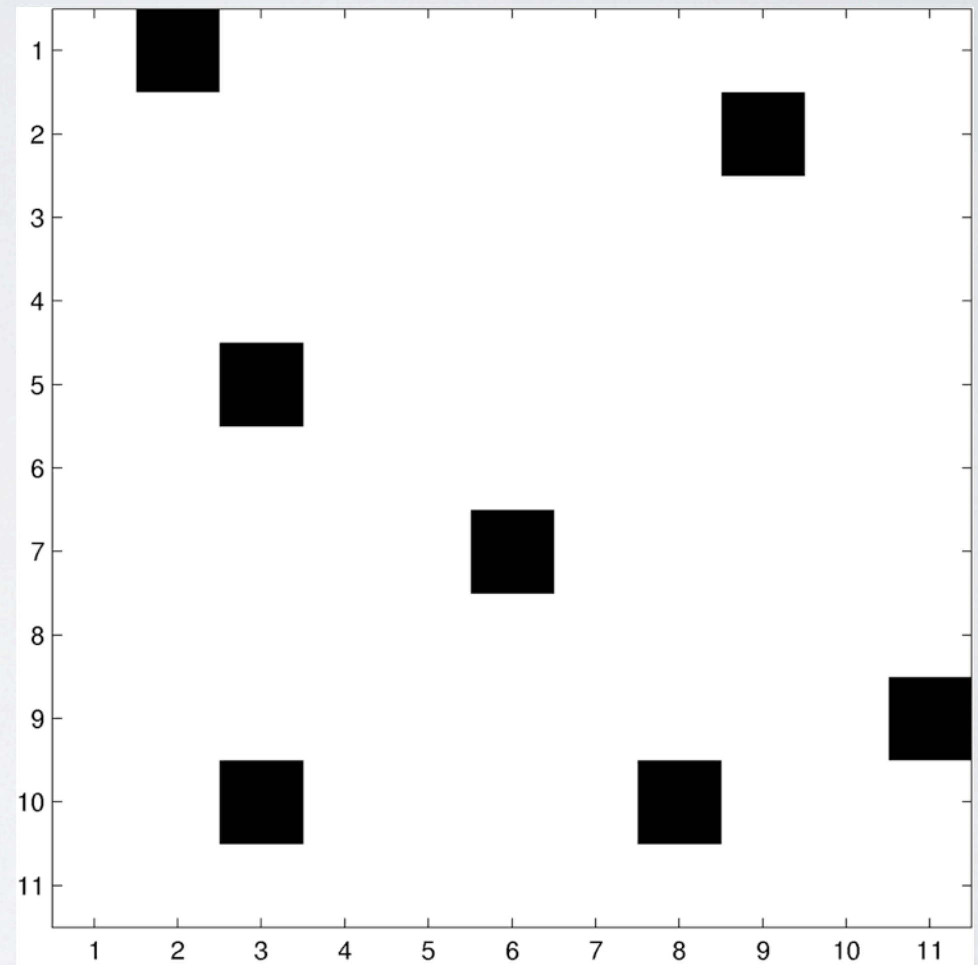
ENCODING THE ERROR

Four different methods to encode **E**:

1. Naïve Factors
2. Naïve Indices
3. Naïve Exclusive OR
4. Typed Exclusive OR

ENCODING THE ERROR

1. Naïve Factors
   Factor **E** and encode factors similar to **B**
2. Naïve Indices
   Send the indices of errors (log(*nm*) bits each)
3. Naïve Exclusive OR
   Send the value of each element of **E** using optimal prefix codes to 1 and 0

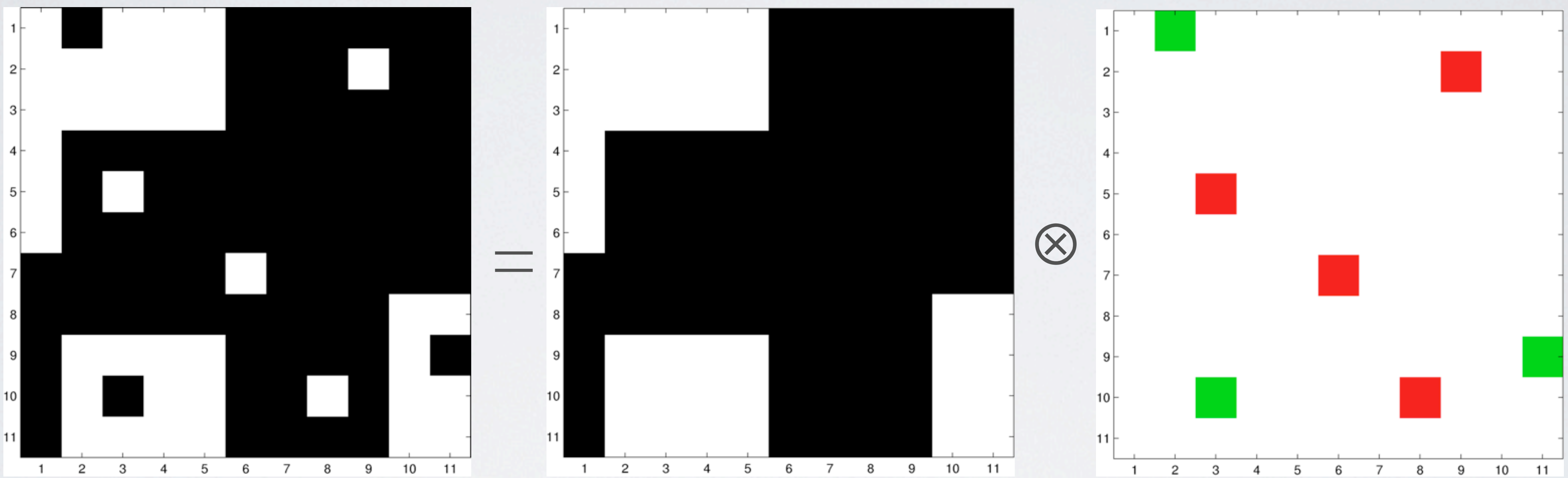


E

ENCODING THE ERROR

4. Typed Exclusive OR
   - Divide error matrix **E** into over-covering ($\mathbf{E}^-$) and under-covering ($\mathbf{E}^+$) parts ($\mathbf{E} = \mathbf{E}^- + \mathbf{E}^+$)
   - Encode $\mathbf{E}^-$ and $\mathbf{E}^+$ separately using optimal prefix indices
     - $\mathbf{E}^-$ cannot have more 1s than $\mathbf{B} \circ \mathbf{C}$
   - Saves space compared to naïve XOR

EXAMPLE OF TYPED XOR



$\mathbf{A}$ $\qquad$ $\mathbf{B} \circ \mathbf{C}$ $\qquad$ $\mathbf{E}$

HOW HARD CAN IT BE?

- MDL itself is an approximation of Kolmogorov complexity
- Finding minimum-error BMF is NP hard (even to approximate)
- But how hard it is to find the MDL-optimal decomposition?
  - Not necessarily minimum-error decomposition
  - Hardness depends on encoding
  - We know that there exists an encoding for which it is NP-hard to find the MDL-optimal decomposition

AN ALGORITHM FOR BMF: ASSO

- **The Good**
  - Asso is hierarchical and deterministic
    - The k^{th} factor does not change the previous $k - 1$ factors
- **The Bad**
  - Asso is heuristic
- **The Ugly**
  - Asso requires extra parameter t — but MDL can be used to find this, too

EXPERIMENTS

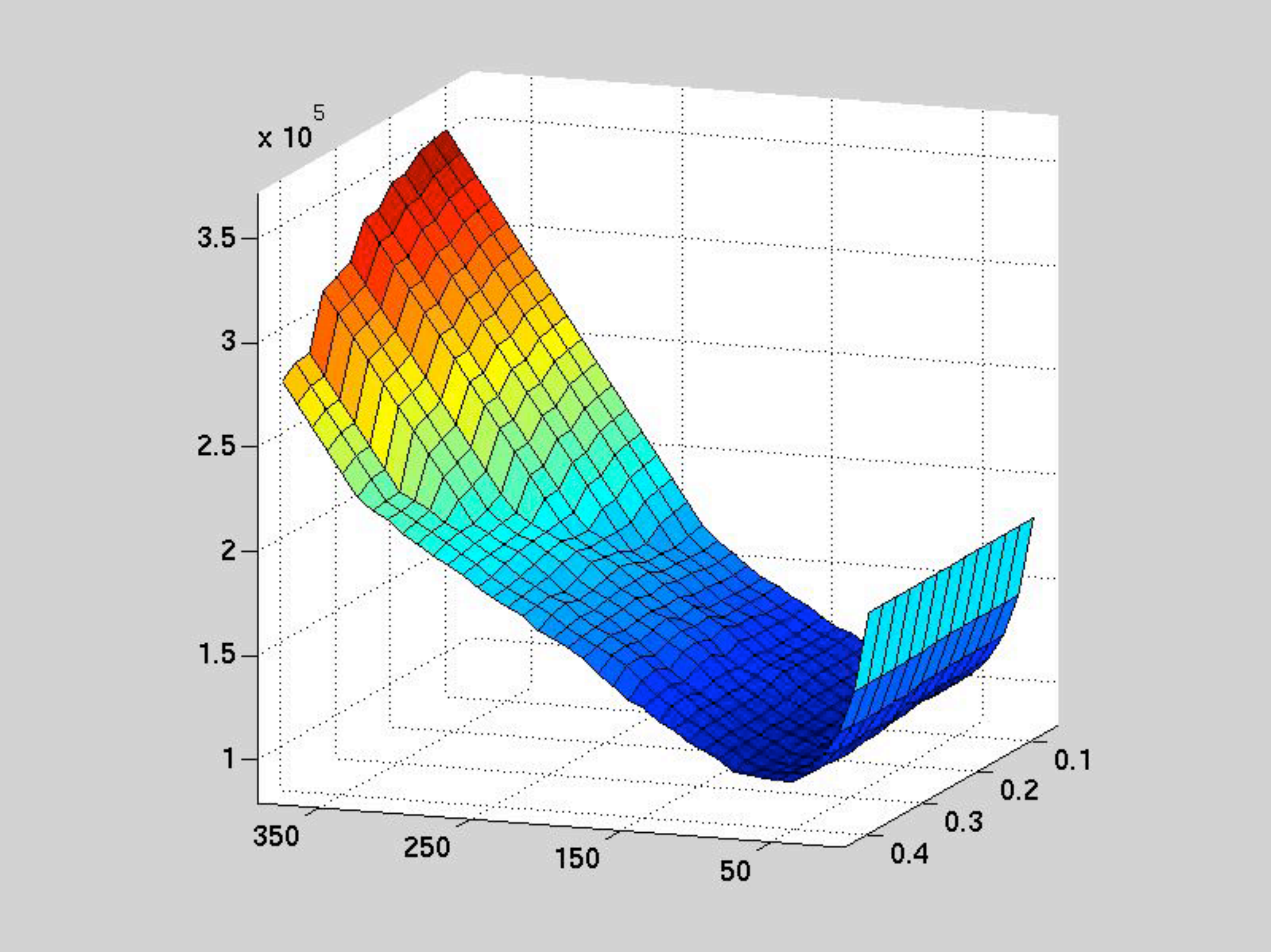
HASN'T THIS BEEN DONE BEFORE?

- Model order selection for matrix factorizations is studied before (mostly with SVD/PCA)
- Methods such as Guttman–Kaiser criterion (c. 1950) or Cattell's scree test (1966) are not suitable
  - Poor performance and need for subjective decisions
- We tried Cross Validation, but it did not work
  - Well-known problem with matrix factorizations, recent ECML'11 paper to address this

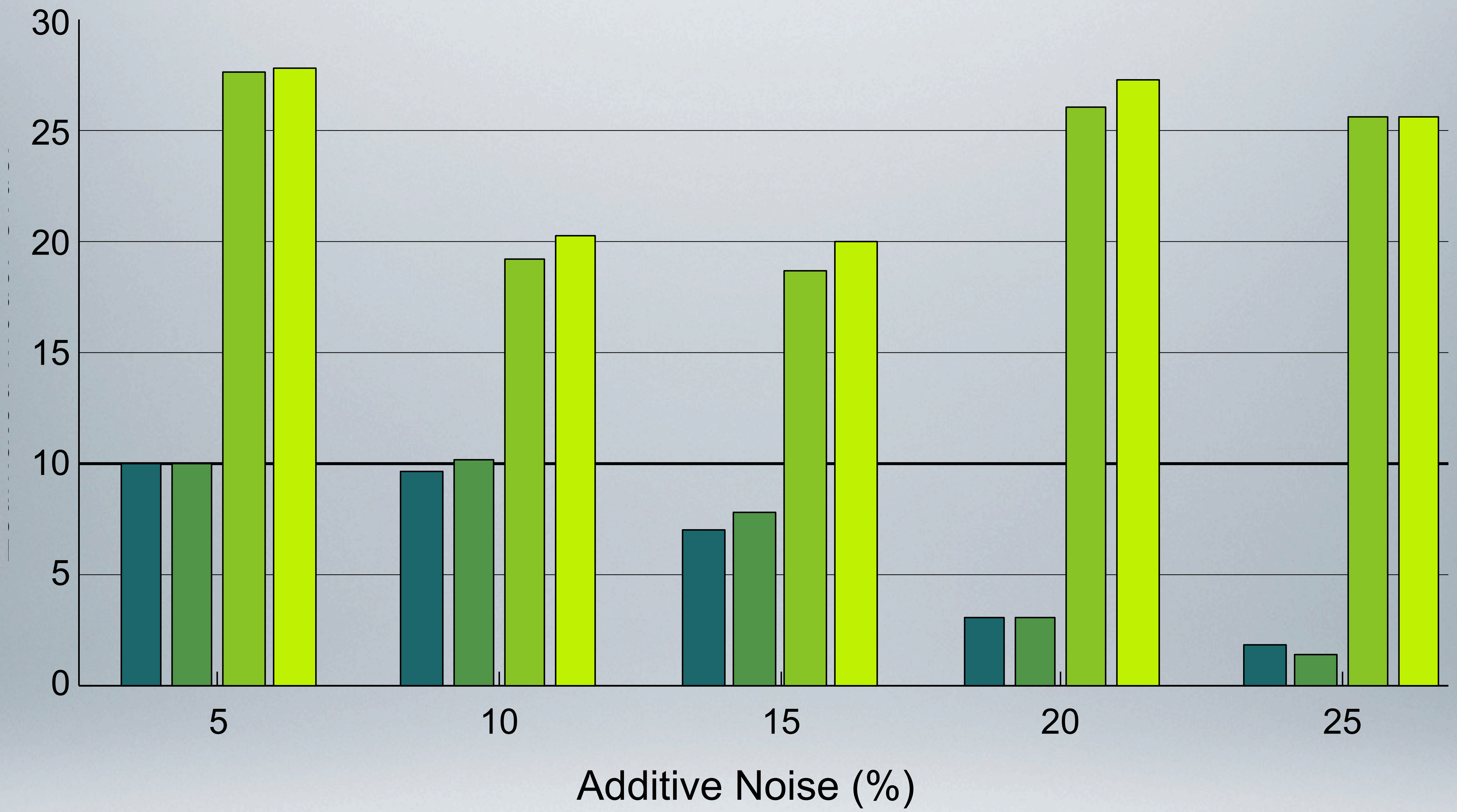
max planck institut
informatik

THE DNA DATA

max planck institut
informatik

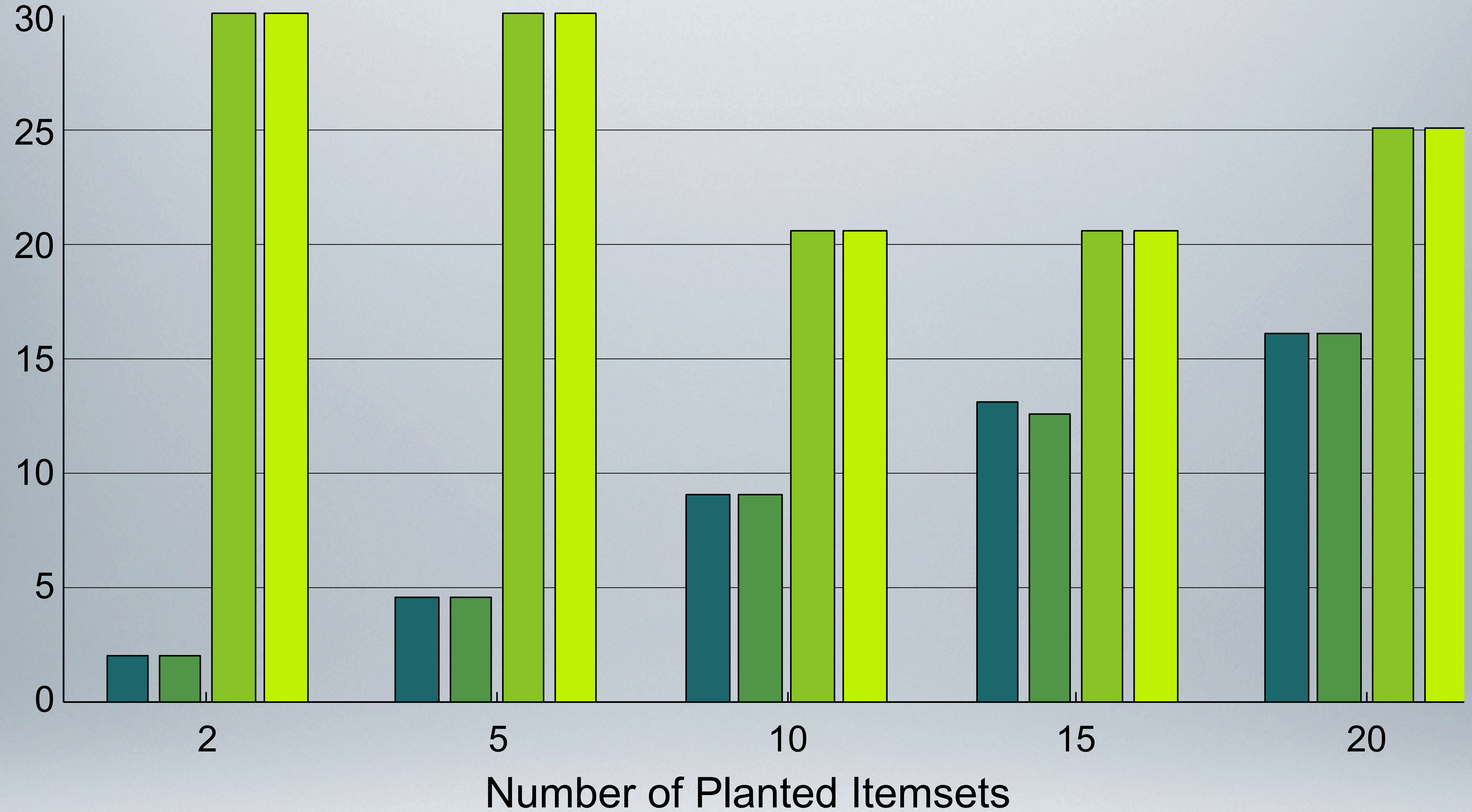


THE DNA DATA



Typed XOR | Naïve XOR | Naive Indices | Naive Factors

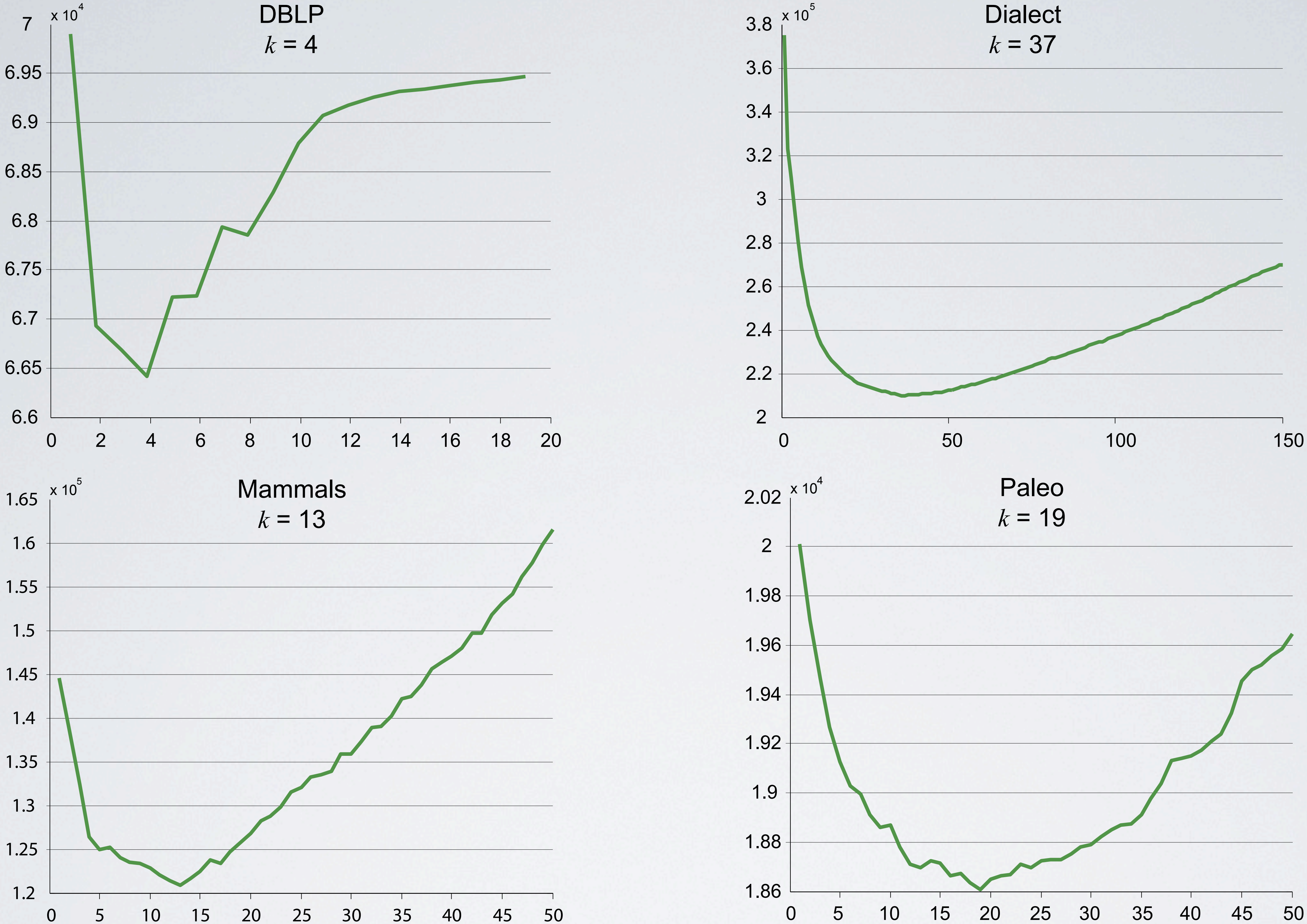
EFFECTS OF NOISE



Typed XOR | Naïve XOR | Naive Indices | Naive Factors

EFFECTS OF *K*

max planck institut
informatik



REAL-WORLD DATA

max planck institut informatik

CONCLUSIONS

- MDL works well for BMF, even with many layers of approximations
  - Allows to find new kind of information about the data
  - The MDL formulation can be used with any algorithm, not just Asso
- Future work with better encodings and extending to variations of BMF

Thank You!

- MDL works well for BMF, even with many layers of approximations
  - Allows to find new kind of information about the data
  - The MDL formulation can be used with any algorithm, not just Asso
- Future work with better encodings and extending to variations of BMF