Drawing Clusterings of Bipartite Graphs

Thibault Marette[1] and **Stefan Neumann**[2]   [1] Ecole Normale Supérieure de Lyon, France   [2] KTH Royal Institute of Technology, Stockholm, Sweden

Abstract

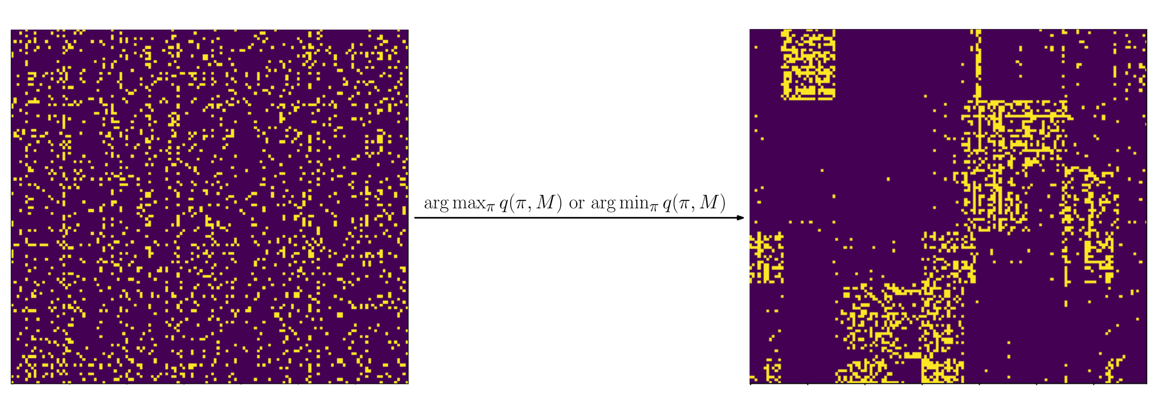
We study the drawing of a given *overlapping* clusterings of bipartite graphs. Our approach does not require a (dis-)similarity function between the clusters that shall be visualized. We present:

- An algorithm for drawing overlapping clusters
- A pre-processing routine that decreases the computational complexity
- A novel objective function to assess the drawing quality

Seriation Problem

Goal: Reorder the matrix to find "dense" blocks

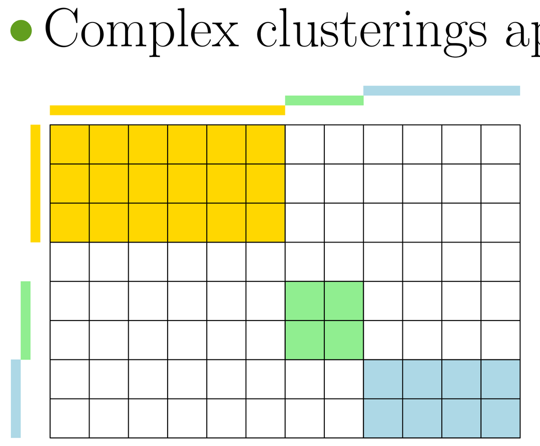
- Given a quality metric, find row and column permutations to optimize the visualization [2]
- *Pattern-agnostic*: no prior knowledge over the structure of the data
- Cannot visualize a *given* clustering



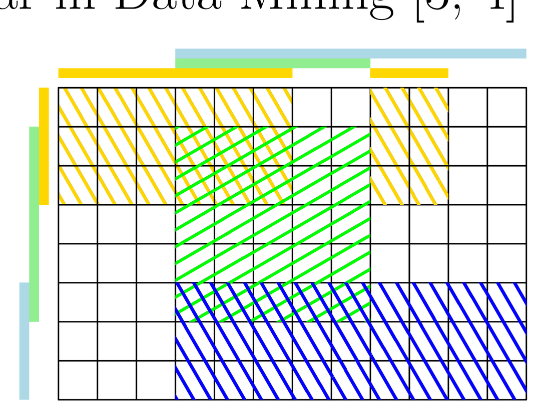
Drawing Overlapping Clusterings

In a bipartite graph $G = (R \cup C, E)$, a *cluster* is defined by $(R' \subseteq R, C' \subseteq C)$. Clustering algorithms find dense subgraphs $\{(R', C')\}_{i=1}^k$.

- Tiles are colored according to which clusters the corresponding rows and columns belong to
- Projection over the rows and columns
- Complex clusterings appear in Data Mining [3, 4]



simple clustering (disjoint clusters)



complex clustering (overlapping clusters)

Problem Summary

- *Input:*
  - The bipartite graph G
  - *A complex clustering*, i.e., clusters $\{(R'_i, C'_i)\}_{i=1}^k$
- *Output:*
  - A permutation of the rows and of the columns that visualizes the given clustering $\{(R'_i, C'_i)\}_{i=1}^k$

To get the permutations, we maximize an objective function that measures the drawing quality:

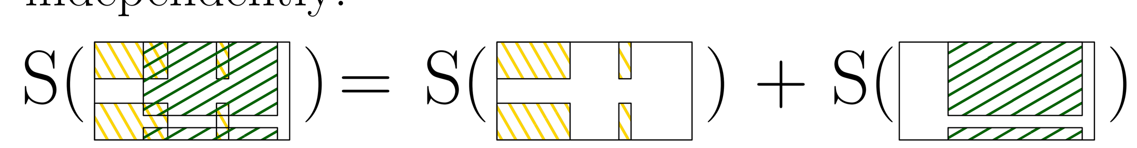
$$\arg\max_{\pi_R, \pi_C} \sum_{i=0}^{k} S(\text{cluster}_i, (\pi_R, \pi_C)),$$

where

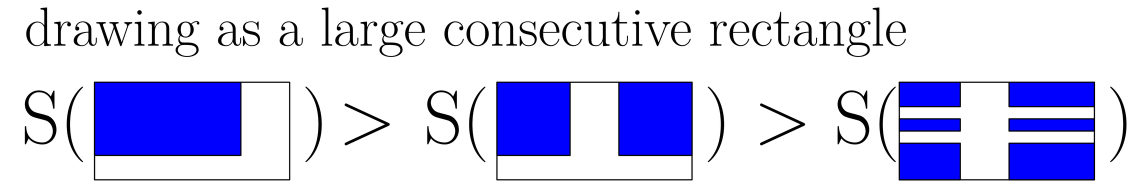
$$S(\text{cluster}) = \sum_{\text{rect} \in \mathcal{R}(cluster)} \text{area}(\text{rect})^2.$$

It is founded on two properties:

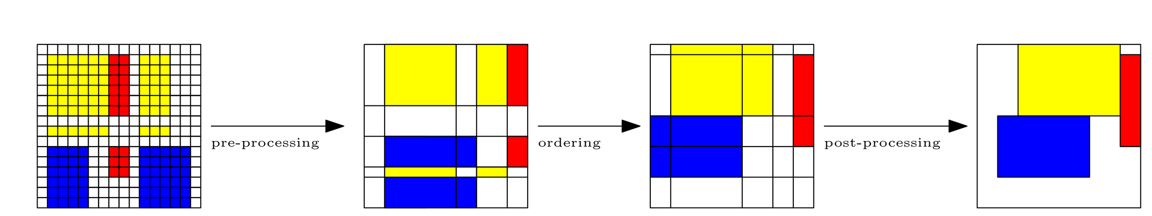
- The score of two clusters should be evaluated independently:



- A cluster is well drawn when it appears in the drawing as a large consecutive rectangle



Algorithm Overview

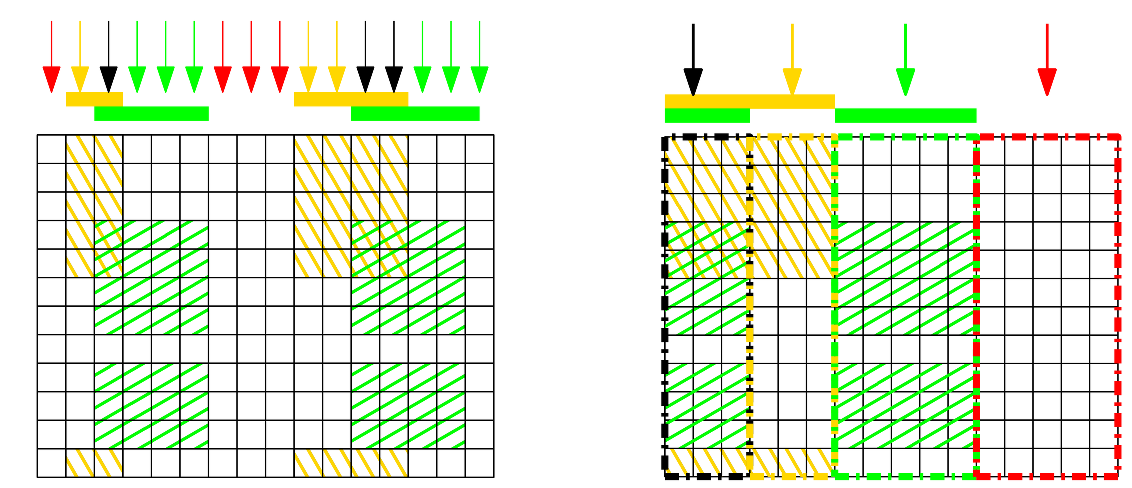


- *Pre-processing*: Reduce the computational complexity by creating *blocks* of rows and columns
- *Ordering*: Provide an ordering *of the blocks* to draw clusters as good as possible.
  There are two components in this step:
  - *Objective function*: Assert the quality of the ordering
  - *Optimization algorithm*: Find the best permutation possible to maximize the objective function
- *Post-processing*: Augment the ordering of the previous phase to get a better global picture

Pre-processing

Goal: Utilize the clustering information to reduce the computational complexity of the problem.

- Form *blocks* of columns (and rows) that belong exactly to the same set of clusters



Consequence: We only need to reorder the blocks! This reorders the rows and columns implicitly.

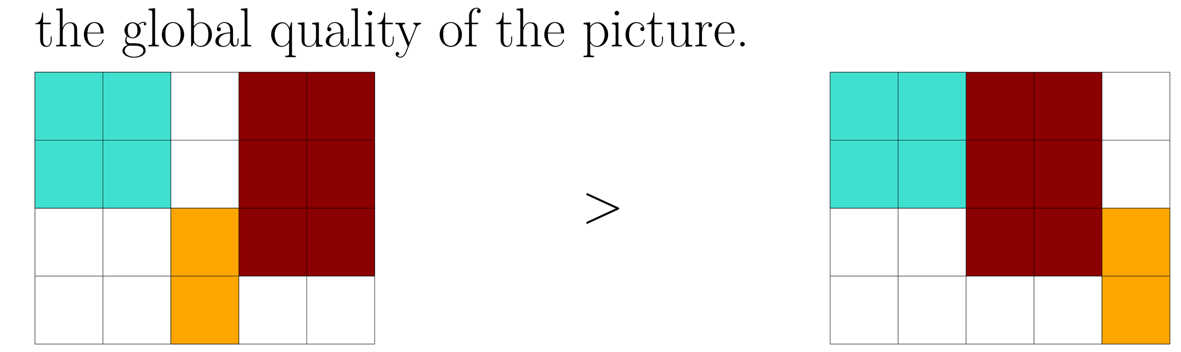
Ordering

Goal: Compute an ordering of the blocks to draw the clusters as good as possible.

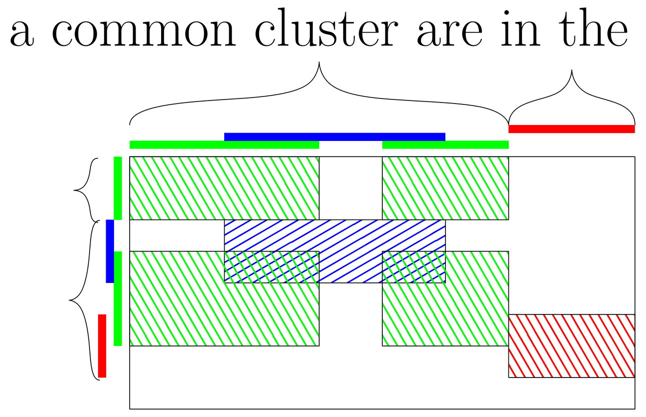
- We maximize the objective function detailed on the left column

Post-processing

Goal: Reorder *groups of* blocks such that the objective function does not decrease, but we increase the global quality of the picture.

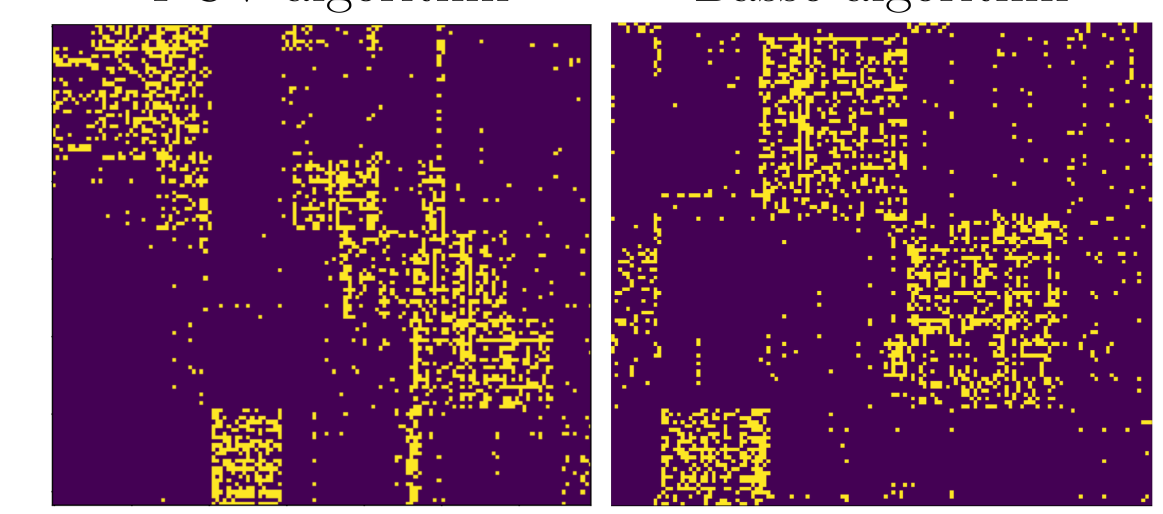


→ Partition the columns so that any pair of columns that have a common cluster are in the same set

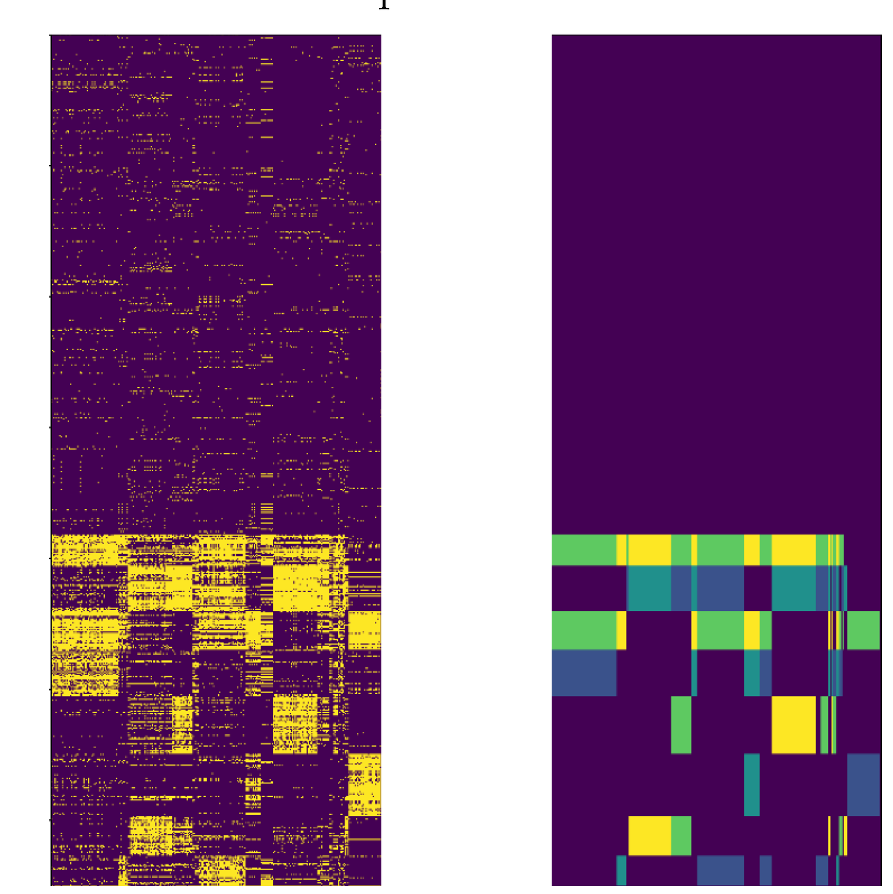


Results

- Use the visualization to compare clusterings



- Assess the local imperfections of a clustering



References

[1] Corinna Vehlow, Fabian Beck, and D. Weiskopf. Visualizing group structures in graphs: A survey. *Computer Graphics Forum*, 36, 2017.

[2] Panpan Xu, Nan Cao, Huamin Qu, and J. Stasko. Interactive visual co-cluster analysis of bipartite graphs. *PacificVis*, pages 32–39, 2016.

[3] Pauli Miettinen and Stefan Neumann. Recent developments in boolean matrix factorization. In *IJCAI*, pages 4922–4928, 2020.

[4] S.C. Madeira and A.L. Oliveira. Biclustering algorithms for biological data analysis: a survey. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 1(1):24–45, 2004.

Contact Information: thibault.marette@ens-lyon.fr

Supported by the ERC Advanced Grant REBOUND (834862) and the EC H2020 RIA project SoBigData++ (871042).