Non-Exhaustive, Overlapping Co-Clustering

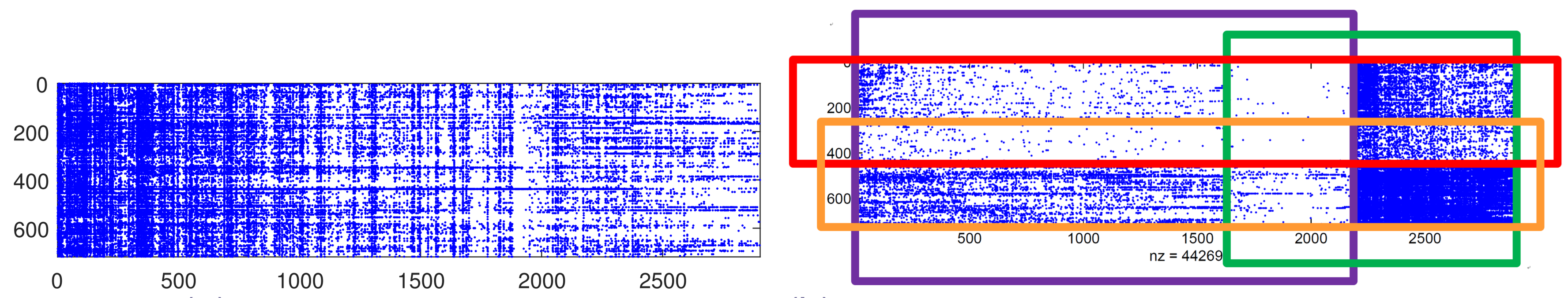
Joyce Jiyoung Whang*[1] and Inderjit S. Dhillon[2]
[1] Sungkyunkwan University (SKKU), [2] The University of Texas at Austin
ACM International Conference on Information and Knowledge Management, 2017

Main Contributions

- Non-Exhaustive, Overlapping Co-Clustering Problem:
  - Simultaneously identify a clustering of the rows as well as the columns of a two dimensional data matrix.
  - Both of the row and column clusters can overlap with each other.
  - Outliers are not assigned to any cluster.
- An intuitive objective function is proposed to formulate this problem.
- NEO-CC: an efficient iterative algorithm that optimizes the non-exhaustive, overlapping co-clustering objective function.
- Experimental results show that the NEO-CC algorithm effectively captures the underlying co-clustering structure of real-world data.

An Example on a User-Movie Rating Matrix

- Result on a user-movie rating dataset where each row represents a user and each column represents a movie.
- The NEO-CC method detects one outlier from the rows, which corresponds to a user who randomly gives ratings.



(a) The original data matrix (b) Rearrangement of the rows and the columns according to the output of the NEO-CC method.

Figure: Visualization of a user-movie rating dataset.

The NEO-CC Objective Function

- Idea: Consider the sum of squared differences between each entry and each mean of the co-clusters the data point belongs to.

$$\begin{aligned} \underset{\boldsymbol{U},\boldsymbol{V}}{\text{minimize}} \quad & \sum_{i=1}^{k}\sum_{j=1}^{l} \|D(\mathbf{u}_i)\boldsymbol{X}D(\mathbf{v}_j) - \hat{\mathbf{u}}_i\hat{\mathbf{u}}_i^T\boldsymbol{X}\hat{\mathbf{v}}_j\hat{\mathbf{v}}_j^T\|_F^2 \\ \text{subject to} \quad & \text{trace}(\boldsymbol{U}^T\boldsymbol{U}) = (1+\alpha_r)n, \\ & \textstyle\sum_{i=1}^{n} \mathbb{I}\{(\boldsymbol{U}\mathbf{1})_i = 0\} \le \beta_r n, \\ & \text{trace}(\boldsymbol{V}^T\boldsymbol{V}) = (1+\alpha_c)m, \\ & \textstyle\sum_{i=1}^{m} \mathbb{I}\{(\boldsymbol{V}\mathbf{1})_i = 0\} \le \beta_c m, \end{aligned}$$

- $\boldsymbol{U} = [u_{ij}]_{n\times k}$: the assignment matrix for row clustering
- $\boldsymbol{V} = [v_{ij}]_{m\times l}$ denote the assignment matrix for column clustering.
- $\mathbb{I}\{exp\} = 1$ if exp is true; 0 otherwise.
- $D(\mathbf{y}) = [d_{ij}]_{m\times m}$: the diagonal matrix with $d_{ii} = y_i$ $(i = 1, \ldots, m)$.
- α_r and β_r are the parameters for row clustering, and α_c and β_c are the parameters for column clustering. α_r and α_c control the amount of overlap while β_r and β_c control the degree of non-exhaustiveness.
- Example: for the entry x_{21}, the NEO-CC objective considers the squared differences between x_{21} and four different means.

$$X = \begin{array}{c} \begin{array}{ccccc} \mathbf{x}^c_1 & \mathbf{x}^c_2 & \mathbf{x}^c_3 & \mathbf{x}^c_4 & \mathbf{x}^c_5 \end{array} \\ \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} \end{bmatrix} \end{array} \begin{array}{c} \mathbf{x}^r_1 \\ \mathbf{x}^r_2 \\ \mathbf{x}^r_3 \\ \mathbf{x}^r_4 \end{array} \quad U = \begin{array}{c} \begin{array}{cc} \mathcal{C}^r_1 & \mathcal{C}^r_2 \end{array} \\ \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \end{array} \begin{array}{c} \mathbf{x}^r_1 \\ \mathbf{x}^r_2 \\ \mathbf{x}^r_3 \\ \mathbf{x}^r_4 \end{array} \quad V = \begin{array}{c} \begin{array}{cc} \mathcal{C}^c_1 & \mathcal{C}^c_2 \end{array} \\ \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \end{array} \begin{array}{c} \mathbf{x}^c_1 \\ \mathbf{x}^c_2 \\ \mathbf{x}^c_3 \\ \mathbf{x}^c_4 \\ \mathbf{x}^c_5 \end{array}$$

(a) Data matrix $\boldsymbol{X}$, row clustering $\boldsymbol{U}$, and column clustering $\boldsymbol{V}$

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} \end{bmatrix} \;\Downarrow\; \left\{x_{21} - \left(\frac{x_{11}+x_{12}+x_{13}+x_{21}+x_{22}+x_{23}}{6}\right)\right\}^2$$

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} \end{bmatrix} \;\Downarrow\; \left\{x_{21} - \left(\frac{x_{11}+x_{14}+x_{21}+x_{24}}{4}\right)\right\}^2$$

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} \end{bmatrix} \;\Downarrow\; \left\{x_{21} - \left(\frac{x_{21}+x_{22}+x_{23}+x_{31}+x_{32}+x_{33}}{6}\right)\right\}^2$$

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} \end{bmatrix} \;\Downarrow\; \left\{x_{21} - \left(\frac{x_{21}+x_{24}+x_{31}+x_{34}}{4}\right)\right\}^2$$

(b) The contribution of x_{21} to the NEO-CC objective

Figure: The NEO-CC objective considers the differences between each entry and the co-cluster means the entry belongs to.

The NEO-CC Algorithm

Input: $\boldsymbol{X} \in \mathbb{R}^{n\times m}$, k, l, α_r, α_c, β_r, β_c
Output: Row clustering $\boldsymbol{U} \in \{0,1\}^{n\times k}$, Column clustering $\boldsymbol{V} \in \{0,1\}^{m\times l}$

Initialize $\boldsymbol{U}$, $\boldsymbol{V}$, and $t = 0$.
while not converged **do**
  Update row clustering by computing the distance between a data point $\mathbf{x}_p \in \mathcal{X}^r$ for p=1, $\cdots$, n and a row cluster $\mathcal{C}^r_q$ for $q = 1, \cdots, k$.
  Update column clustering by computing the distance between a data point $\mathbf{x}_p \in \mathcal{X}^c$ for p=1, $\cdots$, m and a column cluster $\mathcal{C}^c_q$ for q=1, $\cdots$, l.
end while

- Example: the distance between a data point and a row cluster.

$$U = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad V = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} \end{bmatrix} \quad X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ x_{21} & x_{22} & x_{23} & x_{24} & x_{25} \\ x_{31} & x_{32} & x_{33} & x_{34} & x_{35} \\ x_{41} & x_{42} & x_{43} & x_{44} & x_{45} \end{bmatrix}$$

$$\Downarrow$$

$$dist(\mathbf{x}^r{}_1, \mathcal{C}^r{}_1) = \|\mathbf{x}^r{}_{11} - \mathbf{m}_{11}\|^2 + \|\mathbf{x}^r{}_{12} - \mathbf{m}_{12}\|^2$$

$$\mathbf{x}^r{}_{11} = \begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{m}_{11} = \begin{bmatrix} \frac{x_{11}+x_{12}+x_{13}+x_{21}+x_{22}+x_{23}}{6} \\ \frac{x_{11}+x_{12}+x_{13}+x_{21}+x_{22}+x_{23}}{6} \\ \frac{x_{11}+x_{12}+x_{13}+x_{21}+x_{22}+x_{23}}{6} \\ 0 \\ 0 \end{bmatrix} \quad \mathbf{x}^r{}_{12} = \begin{bmatrix} x_{11} \\ 0 \\ 0 \\ x_{14} \\ 0 \end{bmatrix} \quad \mathbf{m}_{12} = \begin{bmatrix} \frac{x_{11}+x_{14}+x_{21}+x_{24}}{4} \\ 0 \\ 0 \\ \frac{x_{11}+x_{14}+x_{21}+x_{24}}{4} \\ 0 \end{bmatrix}$$

Figure: The distance between $\mathbf{x}^r_1$ and a row cluster $\mathcal{C}^r_1$.

- We can theoretically prove that the NEO-CC algorithm monotonically decreases the NEO-CC objective function.

Experimental Results

- Datasets: user-movie rating matrices, an yeast gene expression dataset, and a social network with node attributes.
- Compare the clustering performance of the NEO-CC method with other state-of-the-art co-clustering and one-way clustering methods.
- The NEO-CC algorithm achieves the highest F_1 scores.
  - The performance of NEO-CC is even better than NEO-lrsdp.
  - Co-clustering enables us to perform an implicit dimensionality reduction – performing an implicit regularized clustering.

Table: F_1 scores (%) on the real-world datasets.

		IPM	ROCC	MSSR1	MSSR2	NEO-iter	NEO-lrsdp	NEO-CC
ML1	average	22.4	55.7	43.8	44.2	56.3	56.4	**58.1**
ML2	best	36.2	53.3	50.6	50.5	56.8	56.8	**58.8**
	worst	18.6	53.3	50.2	48.2	56.8	56.8	**58.1**
	average	26.7	53.3	50.5	49.4	56.8	56.8	**58.4**
Yeast	best	N/A	15.0	17.4	19.3	36.6	39.1	**40.7**
	worst	N/A	12.8	16.4	18.0	35.6	**39.0**	36.2
	average	N/A	14.3	16.9	18.5	36.0	39.1	**40.0**
Facebook	best	N/A	26.9	30.6	31.8	34.7	37.6	**37.7**
	worst	N/A	24.0	28.7	27.3	33.3	33.7	**37.1**
	average	N/A	25.2	29.7	29.7	33.9	35.9	**37.3**

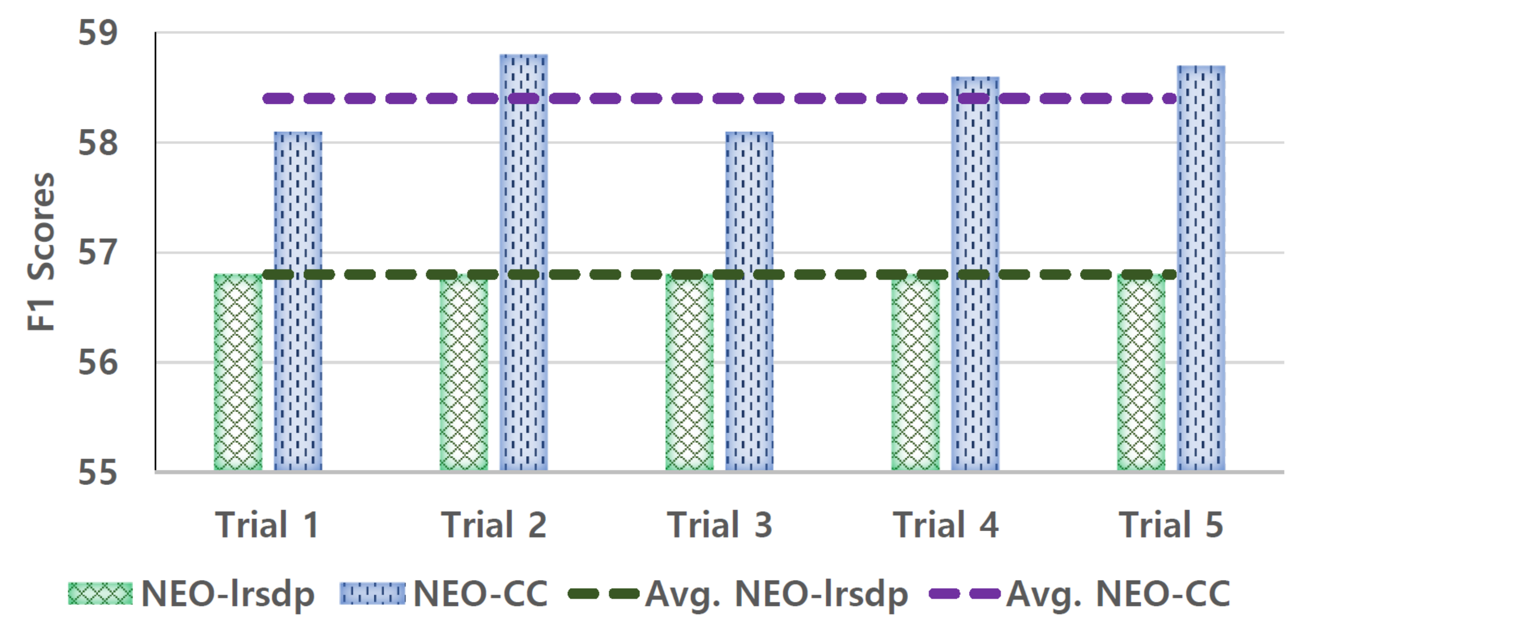


Figure: F_1 scores of the best baseline method (NEO-lrsdp) and the NEO-CC method on the ML2 dataset.

Conclusions & Future Work

- The NEO-CC method provides a principled way to capture the underlying co-clustering structure of real-world data.
- We plan to investigate a low-rank semidefinite programming for the NEO-CC method to develop a more sophisticated algorithm.

Joyce Jiyoung Whang and Inderjit S. Dhillon. Non-Exhaustive, Overlapping Co-Clustering. In CIKM, 2017.