Co-clustering through Optimal Transport

Charlotte Laclau, Ievgen Redko, Basarab Matei, Younès Bennani, Vincent Brault

Co-clustering through Optimal Transport

Charlotte Laclau [1] **Ievgen Redko** [2] **Basarab Matei** [1] **Younès Bennani** [1] **Vincent Brault** [3]

Abstract

In this paper, we present a novel method for co-clustering, an unsupervised learning approach that aims at discovering homogeneous groups of data instances and features by grouping them simultaneously. The proposed method uses the entropy regularized optimal transport between empirical measures defined on data instances and features in order to obtain an estimated joint probability density function represented by the optimal coupling matrix. This matrix is further factorized to obtain the induced row and columns partitions using multiscale representations approach. To justify our method theoretically, we show how the solution of the regularized optimal transport can be seen from the variational inference perspective thus motivating its use for co-clustering. The algorithm derived for the proposed method and its kernelized version based on the notion of Gromov-Wasserstein distance are fast, accurate and can determine automatically the number of both row and column clusters. These features are vividly demonstrated through extensive experimental evaluations.

1. Introduction

Cluster analysis aims to gather data instances into groups, called clusters, where instances within one group are similar among themselves while instances in different groups are as dissimilar as possible. Clustering methods have become more and more popular recently due to their ability to provide new insights into unlabeled data that may be difficult or even impossible to capture for a human being. Clustering methods, however, do not take into account the possible existing relationships between the features that describe the data instances. For example, one may consider a data matrix extracted from text corpus where each document is described by the terms appearing in it. In this case, clustering documents may benefit from the knowledge about the correlation that exists between different terms revealing their probability of appearing in the same documents. This idea is the cornerstone of *co-clustering* (Hartigan, 1972; Mirkin, 1996) where the goal is to perform clustering of both data points and features simultaneously. The obtained latent structure of data is composed of blocks usually called co-clusters. Applications of co-clustering include but are not limited to recommendation systems (George & Merugu, 2005; Deodhar & Ghosh, 2010; Xu et al., 2012), gene expression analysis (Cheng et al., 2008; Hanisch et al., 2002) and text mining (Dhillon et al., 2003a; Wang et al., 2009). As a result, these methods are of an increasing interest to the data mining community.

Co-clustering methods are often distinguished into probabilistic methods (e.g., (Dhillon et al., 2003b; Banerjee et al., 2007; Nadif & Govaert, 2008; Wang et al., 2009; Shan & Banerjee, 2010)) and metric based (e.g., (Rocci & Vichi, 2008; Ding et al., 2006)) methods. Probabilistic methods usually make an assumption that data was generated as a mixture of probability density functions where each one of them corresponds to one co-cluster. The goal then is to estimate the parameters of the underlying distributions and the posterior probabilities of each co-cluster given the data. Metric based approaches proceed in a different way and rely on introducing and optimizing a criterion commonly taking into account intra- and inter-block variances. This criterion, in its turn, is defined using some proper metric function that describes the geometry of data in the most precise way possible. Both metric and probabilistic approaches are known to have their own advantages and limitations: despite being quite efficient in modeling the data distribution, probabilistic methods are computationally demanding and hardly scalable; metric methods are less computationally demanding but present the need to choose the "right" distance that uncovers the underlying latent co-clusters' structure based on available data. Furthermore, the vast majority of co-clustering methods require the number of co-clusters to be set in advance. This is usu-

[1]CNRS, LIPN, Université Paris 13 - Sorbonne Paris Cité, France [2]CNRS UMR 5220 - INSERM U1206, Univ. Lyon 1, INSA Lyon, F-69621 Villeurbanne, France [3]CNRS, LJK, Univ. Grenoble-Alpes, France. Correspondence to: Charlotte Laclau <charlotte.laclauc@univ-grenoble-alpes.fr>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, 2017. JMLR: W&CP.

[0]The first author of this paper is now a post-doc in CNRS, LIG, Univ. Grenoble-Alpes, France

ally done using the computationally expensive exhaustive search over a large number of possible pairs of row and column clusters as in (Keribin et al., 2015; Wyse & Friel, 2012; Wyse et al., 2014).

In this paper, we address the existing issues of co-clustering methods described above by proposing a principally new approach that efficiently solves the co-clustering problem from both qualitative and computational points of view and allows the automatic detection of the number of co-clusters. We pose the co-clustering problem as the task of transporting the empirical measure defined on the data instances to the empirical measure defined on the data features. The intuition behind this process is very natural to co-clustering and consists in capturing the associations between instances and features of the data matrix. The solution of optimal transportation problem is given by a doubly-stochastic coupling matrix which can be considered as the approximated joint probability distribution of the original data. Furthermore, the coupling matrix can be factorized into three terms where one of them reflects the posterior distribution of data given co-clusters while two others represent the approximated distributions of data instances and features. We use these approximated distributions to obtain the final partitions. We also derive a kernelized version of our method that contrary to the original case, is based on an optimal transportation metric defined on the space of dissimilarity functions.

The main novelty of our work is two-fold. To the best of our knowledge, the proposed approach is a first attempt to apply entropy regularized optimal transport for co-clustering and to give its solution a co-clustering interpretation. While Wasserstein distance has already been adapted to design clustering algorithms (Cuturi & Doucet, 2014; Irpino et al., 2014), our idea is to concentrate our attention on the solution of the optimal transport given by the coupling matrix and not to minimize the quantization error with respect to (w.r.t.) Wasserstein distance. We also note that using entropy regularization leads to a very efficient algorithm that can be easily parallelized (Cuturi, 2013). Second, we show that under some plausible assumptions the density estimation procedure appearing from the use of the optimal transport results in the variational inference problem with the minimization of the reversed Kullback-Leibler divergence. The important implications of this difference w.r.t. other existing methods are explained in Section 3.

The rest of this paper is organized as follows. In Section 2, we briefly present the discrete version of the optimal transportation problem and its entropy regularized version. Section 3 proceeds with the description of the proposed approach, its theoretical analysis and algorithmic implementation. In Section 4, we evaluate our approach on synthetic and real-world data sets and show that it is accurate and substantially more efficient than the other state-of-the-art methods. Last section concludes the paper and gives a couple of hints for possible future research.

2. Background and notations

In this section, we present the formalization of the Monge-Kantorovich (Kantorovich, 1942) optimization problem and its entropy regularized version.

2.1. Optimal transport

Optimal transportation theory was first introduced in (Monge, 1781) to study the problem of resource allocation. Assuming that we have a set of factories and a set of mines, the goal of optimal transportation is to move the ore from mines to factories in an optimal way, i.e., by minimizing the overall transport cost.

More formally, given two empirical probability measures[1] $\hat{\mu}_S = \frac{1}{N_S}\sum_{i=1}^{N_S}\delta_{x_i^S}$ and $\hat{\mu}_T = \frac{1}{N_T}\sum_{i=1}^{N_T}\delta_{x_i^T}$ defined as uniformly weighted sums of Dirac with mass at locations supported on two point sets $X_S = \{\boldsymbol{x}_i^S \in \mathbb{R}^d\}_{i=1}^{N_S}$ and $X_T = \{\boldsymbol{x}_i^T \in \mathbb{R}^d\}_{i=1}^{N_T}$, the Monge-Kantorovich problem consists in finding a probabilistic coupling γ defined as a joint probability measure over $X_S \times X_T$ with marginals $\hat{\mu}_S$ and $\hat{\mu}_T$ that minimizes the cost of transport w.r.t. some metric $l : X_s \times X_t \to \mathbb{R}^+$:

$$\min_{\gamma \in \Pi(\hat{\mu}_S, \hat{\mu}_T)} \langle M, \gamma \rangle_F$$

where $\langle \cdot, \cdot \rangle_F$ is the Frobenius dot product, $\Pi(\hat{\mu}_S, \hat{\mu}_T) = \{\gamma \in \mathbb{R}_+^{N_S \times N_T} | \gamma \mathbf{1} = \hat{\mu}_S, \gamma^T \mathbf{1} = \hat{\mu}_T\}$ is a set of doubly stochastic matrices and M is a dissimilarity matrix, i.e., $M_{ij} = l(\boldsymbol{x}_i^S, \boldsymbol{x}_j^T)$, defining the energy needed to move a probability mass from $\boldsymbol{x}_i^S$ to $\boldsymbol{x}_j^T$. This problem admits a unique solution γ^* and defines a metric on the space of probability measures (called the Wasserstein distance) as follows:

$$W(\hat{\mu}_S, \hat{\mu}_T) = \min_{\gamma \in \Pi(\hat{\mu}_S, \hat{\mu}_T)} \langle M, \gamma \rangle_F.$$

The Wasserstein distance has been successfully used in various applications, for instance: computer vision (Rubner et al., 2000), texture analysis (Rabin et al., 2011), tomographic reconstruction (I. Abraham & Carlier, 2016), domain adaptation (Courty et al., 2014), metric learning (Cuturi & Avis, 2014) and clustering (Cuturi & Doucet, 2014; Irpino et al., 2014). This latter application is of a particular interest as Wasserstein distance is known to be a very efficient metric due to its capability of taking into account the

[1] Due space limitation, we present only the discrete version of optimal transport. For more details on the general continuous case and the convergence of empirical measures, we refer the interested reader to the excellent monograph by (Villani, 2009).

geometry of data through the pairwise distances between samples. The success of algorithms based on this distance is also due to (Cuturi, 2013) who introduced an entropy regularized version of optimal transport that can be optimized efficiently using matrix scaling algorithm. We present this regularization below.

2.2. Entropic regularization

The idea of using entropic regularization dates back to (Schrödinger, 1931). In (Cuturi, 2013), it found its application to the optimal transportation problem through the following objective function:

$$\min_{\gamma \in \Pi(\hat{\mu}_S, \hat{\mu}_T)} \langle M, \gamma \rangle_F - \frac{1}{\lambda} E(\gamma).$$

Second term $E(\gamma) = -\sum_{i,j}^{N_S, N_T} \gamma_{i,j} \log(\gamma_{i,j})$ in this equation allows to obtain smoother and more numerically stable solutions compared to the original case and converges to it at the exponential rate (Benamou et al., 2015). Another advantage of entropic regularization is that it allows to solve optimal transportation problem efficiently using Sinkhorn-Knopp matrix scaling algorithm (Sinkhorn & Knopp, 1967).

In the next section, we explain the main underlying idea of our approach that consists in associating data instances with features through regularized optimal transport.

3. Co-clustering through optimal transport

In this section we show how the co-clustering problem can be casted in a principally new way and then solved using the ideas from the optimal transportation theory.

3.1. Problem setup

Let us denote by X and Y two random variables taking values in the sets $\{\boldsymbol{x}_r\}_{r=1}^n$ and $\{\boldsymbol{y}_c\}_{c=1}^d$, respectively, where subscripts r and c correspond to rows (instances) and columns (features). Similar to (Dhillon et al., 2003b), we assume that the joint probability distribution between X and Y denoted by $p(X, Y)$ is estimated from the data matrix $\mathcal{A} \in \mathbb{R}^{n \times d}$. We further assume that X and Y consist of instances that are distributed w.r.t. probability measures μ_r, μ_c supported on Ω_r, Ω_c where $\Omega_r \subseteq \mathbb{R}^d$ and $\Omega_r \subseteq \mathbb{R}^n$, respectively.

The problem of co-clustering consists in jointly grouping the set of features and the set of instances into homogeneous blocks by finding two assignment functions C_r and C_c that map as follows: $C_r : \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\} \to \{\hat{x}_1, \ldots, \hat{x}_g\}$, $C_c : \{\boldsymbol{y}_1, \ldots, \boldsymbol{y}_d\} \to \{\hat{y}_1, \ldots, \hat{y}_m\}$ where g and m denote the number of row and columns clusters, and discrete random variables $\hat{X}$ and $\hat{Y}$ represent the partitions induced by X and Y, i.e., $\hat{X} = C_r(X) =$ and $\hat{Y} = C_c(Y)$.

To use discrete optimal transport, we also define two empirical measures $\hat{\mu}_r$ and $\hat{\mu}_c$ based on X and Y as follows: $\hat{\mu}_r = \frac{1}{n} \sum_{i=1}^n \delta_{\boldsymbol{x}_i}$ and $\hat{\mu}_c = \frac{1}{m} \sum_{i=1}^m \delta_{\boldsymbol{y}_i}$. We are now ready to present our method.

3.2. Proposed approach

The main underlying idea of our approach is to use the optimal transportation presented above to find a probabilistic coupling of the empirical measures defined based on rows and columns of a given data matrix. More formally, for some fixed $\lambda > 0$ we solve the co-clustering problem through the following optimization procedure:

$$\gamma_\lambda^* = \text{argmin}_{\gamma \in \Pi(\hat{\mu}_r, \hat{\mu}_c)} \langle \mathrm{M}, \gamma \rangle_{\mathrm{F}} - \frac{1}{\lambda} \mathrm{E}(\gamma), \qquad (1)$$

where the matrix M is computed using the Euclidean distance, i.e., $M_{ij} = \|\boldsymbol{x}_i - \boldsymbol{y}_j\|^2$. The elements of the resulting matrix γ_λ^* provides us with the weights of associations between instances and features: similar instances and features correspond to higher values in γ_λ^*. Our intuition is to use these weights to identify the most similar sets of rows and columns that should be grouped together to form co-clusters.

Following (Benamou et al., 2015), this optimization problem can be equivalently rewritten in the following way:

$$\min_{\gamma \in \Pi(\hat{\mu}_r, \hat{\mu}_c)} \langle M, \gamma \rangle_F - \frac{1}{\lambda} E(\gamma) = \frac{1}{\lambda} \min_{\gamma \in \Pi(\hat{\mu}_r, \hat{\mu}_c)} \mathrm{KL}(\gamma \| \xi_\lambda),$$

where $\xi_\lambda = e^{-\lambda M}$ is the Gibbs kernel.

Finally, we can rewrite the last expression as follows:

$$\min_{\gamma \in \Pi(\hat{\mu}_r, \hat{\mu}_c)} \mathrm{KL}(\gamma \| \xi_\lambda) = \min_{\gamma \in \mathcal{C}} \mathrm{KL}(\gamma \| \xi_\lambda),$$

where $\mathcal{C} = \mathcal{C}_1 \cap \mathcal{C}_2$ is the intersection of closed convex subsets given by $\mathcal{C}_1 = \{\gamma \in \mathbb{R}^{d \times d} | \gamma \mathbf{1} = \hat{\mu}_r\}$ and $\mathcal{C}_2 = \{\gamma \in \mathbb{R}^{d \times d} | \gamma^T \mathbf{1} = \hat{\mu}_c\}$. The solution of the entropy regularized optimal transport can be obtained using Sinkhorn-Knopp algorithm and has the following form (Benamou et al., 2015):

$$\gamma_\lambda^* = \text{diag}(\boldsymbol{\alpha}) \xi_\lambda \text{diag}(\boldsymbol{\beta}), \qquad (2)$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are the scaling coefficients of the Gibbs kernel ξ_λ.

In what follows, we show that under some plausible assumptions, we can interpret these two vectors as approximated rows and columns probability density functions.

3.3. Connection to variational inference

In order to justify our approach from the theoretical point of view, we first explain how the obtained solution γ^*

can be used for co-clustering. As mentioned in (Dhillon et al., 2003b) and later in (Banerjee et al., 2007), the co-clustering can be seen as a density estimation problem where the goal is to approximate the real density $p(X, Y)$ by a simpler one depending on the obtained co-clustering in a way that it preserves the loss in the mutual information given by $I(X, Y) - I(\hat{X}, \hat{Y})$ where $I(X, Y) = \int_{XY} p(x, y) \log \frac{p(x,y)}{p(x)p(y)} dxdy$ is the mutual information. This quantity is further shown to be equal to the Kullback-Leibler divergence between the original distribution $p(X, Y)$ and $q(X, Y)$ where the latter has the following form:

$$q(x, y) = p(y|\hat{y})p(\hat{x}, \hat{y})p(x|\hat{x}).$$

From this point, one may instantly see that the solution of the optimal transport problem γ^* has a very similar form as it also represents the joint probability distribution that approximates the original probability distribution $p(x, y)$ given by the Gibbs measure ξ_λ and also factorizes into three terms. The most important difference, however, lies in the asymmetry of the KL divergence: while (Dhillon et al., 2003b) and (Banerjee et al., 2007) concentrate on minimizing $\mathrm{KL}(p(X, Y)\|q(X, Y))$, our idea is different and consists in minimizing $\mathrm{KL}(q(X, Y)\|p(X, Y))$. This approach is known in the literature as the variational inference (Bishop, 2006) and exhibits a totally different behaviour compared to the minimization of $\mathrm{KL}(p(X, Y)\|q(X, Y))$. As shown by (Bishop, 2006), in variational inference the estimated distribution $q(X, Y)$ concentrates on the modes of data and remains compact, while the minimizer of $\mathrm{KL}(p(X, Y)\|q(X, Y))$ tends to cover the whole surface of the original density and to overestimate its support. As X, Y and $\hat{X}$ and $\hat{Y}$ represent the observed and unobserved variables, respectively, the natural goal is to try to estimate the distribution $p(X, Y|\hat{X}, \hat{Y})$ of the data given the obtained co-clusters by the simpler variational distribution $q(X, Y)$. However, as the maximisation of $p(X, Y|\hat{X}, \hat{Y})$ is computationally impossible, it is common to introduce a free distribution $q(\cdot, \cdot)$ on the parameters $\widehat{X}$ and $\widehat{Y}$ in order to obtain the following decomposition:

$$\log p(X, Y) = \mathcal{L}(q(\hat{X}, \hat{Y})) + \mathrm{KL}(q(\hat{X}, \hat{Y})\|p(\hat{X}, \hat{Y}|X, Y)),$$

where the lower bound

$$\mathcal{L}(q(\hat{X}, \hat{Y})) = \int_{\hat{x}\in\hat{\mathcal{X}}} \int_{\hat{y}\in\hat{\mathcal{Y}}} q(\hat{x}, \hat{y}) \log \frac{p(x, y, \hat{x}, \hat{y})}{q(\hat{x}, \hat{y})} d\hat{x}d\hat{y}$$

is maximized when the KL divergence is minimized.

Now, if we assume that $p(\hat{X}, \hat{Y}|X, Y)$ follows the Gibbs distribution, i.e. $p(\hat{X}, \hat{Y}|X, Y) \propto e^{-\lambda M(x,y)}$, we can consider the original formulation of the regularized optimal transport as the variational inference problem:

$$\min_q \mathrm{KL}(q(\hat{X}, \hat{Y})\|p(\hat{X}, \hat{Y}|X, Y)) = \min_\gamma \mathrm{KL}(\gamma\|\xi_\lambda),$$

where the optimal coupling γ equals to the estimated joint probability $q(\hat{X}, \hat{Y})$.

At this point, we know that the coupling matrix can be seen as an approximation to the original unknown posterior density function but the question how one can use it to obtain the clustering of rows and columns has not been answered yet. In order to solve the variational inference problem, it is usually assumed that the variables $\hat{x}, \hat{y}$ are independent and thus the variational distribution $q(\hat{x}, \hat{y})$ factorizes as $q(\hat{x}, \hat{y}) = q(\hat{x})q(\hat{y})$. This assumption, however, goes against the whole idea of co-clustering that relies on the existence of a deep connection between these two variables.

To this end, we propose to consider the factorization of $q(\hat{x}, \hat{y})$ that has the following form

$$q(\hat{x}, \hat{y}) = q(x)q(\hat{x}, \hat{y}|x, y)q(y).$$

This particular form follows the idea of structured stochastic variational inference proposed in (Hoffman & Blei, 2015) where a term depicting the conditional distribution between hidden and observed variables is added to the fully factorized traditional setting presented above. As stated in (Hoffman & Blei, 2015), this term allows arbitrary dependencies between observed and hidden variables which can increase the fidelity of the approximation.

Following (Bishop, 2006), the optimal estimated densities $q(x)$ and $q(y)$ are controlled by the direction of the smallest variance of $p(x)$ and $p(y)$ respectively. Furthermore, $q(x)$ and $q(y)$ are proportional to the joint densities $p(\hat{y}, y)$ and $p(\hat{x}, x)$, i.e., $q(x) \propto p(\hat{y}, y)$ and $q(y) \propto p(\hat{x}, x)$. Bearing in mind the equivalence between γ^*_λ and $q(\hat{x}, \hat{y})$, this brings us to the following important conclusions: (1) the matrices $\mathrm{diag}(\boldsymbol{\alpha})$ and $\mathrm{diag}(\boldsymbol{\beta})$ can be seen as the approximated densities $p(\hat{Y}, Y)$ and $p(\hat{X}, X)$; (2) vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ represent the approximated densities $p(X)$ and $p(Y)$ obtained by summing X and Y out of $p(\hat{X}, X)$ and $p(\hat{Y}, Y)$, respectively.

According to (Laird, 1978), the non-parametric estimate of the mixing distribution is a piecewise step function where the number of steps depend on the number of components in the mixture. In the cluster analysis, we can assume that $p(X)$ and $p(Y)$ consist of g and m components, respectively. Then, our goal is to detect these steps based on the estimates given by $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ to obtain the desired partitions.

3.4. Kernelized version and Gromov-Wasserstein distance

In this part, we introduce the kernelized version of our method and compare it to the original formulation of our algorithm. In order to proceed, we first define two similarity matrices $K_r \in \mathbb{R}^{n\times n}$ and $K_c \in \mathbb{R}^{d\times d}$ associated to empirical measures $\hat{\mu}_r, \hat{\mu}_c$ thus forming metric-measure

spaces as in (Mémoli, 2011). Matrices K_r and K_c are defined by calculating the pairwise distances or similarities between rows and columns, respectively, without the restriction of them being positive or calculated based on a proper distance function satisfying the triangle inequality. The entropic Gromov-Wasserstein discrepancy in this case is defined as follows (Peyré et al., 2016):

$$\begin{aligned}\mathrm{GW}(K_r, K_c, \hat{\mu}_r, \hat{\mu}_c) &= \min_{\gamma \in \Pi_{\hat{\mu}_r, \hat{\mu}_c}} \Gamma_{K_r, K_c}(\gamma) - \lambda E(\gamma) \\ &= \min_{T \in \Pi_{\hat{\mu}_r, \hat{\mu}_c}} \sum_{i,j,k,l} L(K_{r_{i,j}}, K_{c_{k,l}}) \gamma_{i,j} \gamma_{k,l} - \lambda E(\gamma).\end{aligned}$$

where γ is a coupling matrix between two similarity matrices and $L : \mathbb{R} \times \mathbb{R} \to \mathbb{R}_+$ is an arbitrary lost-function, usually the quadratic-loss or Kullback-Leibler divergence.

Based on this definition, one may define the problem of the entropic Gromov-Wasserstein barycenters for similarity or distance matrices K_r and K_c as follows:

$$\min_{K, \gamma_r, \gamma_c} \sum_{i=\{r,c\}} \varepsilon_i \Gamma_{K, K_i}(\gamma_i) - \lambda E(\gamma_i) \tag{3}$$

where K is the computed barycenter and $\gamma_r \in \Pi_{\hat{\mu}, \hat{\mu}_r}$, $\gamma_c \in \Pi_{\hat{\mu}, \hat{\mu}_c}$ are the coupling matrices that align it with K_r and K_c, respectively. ε_i are the weighting coefficients summing to one, i.e., $\sum_{i=\{r,c\}} \varepsilon_i = 1$ that determine our interest in more accurate alignment between K_r and K or K_c and K.

The intuition behind this optimization procedure for co-clustering with respect to original formulation given in (1) is the following: while in (1) we align rows with columns directly, in (3) our goal is to do it via an intermediate representation given by the barycenter K that is optimally aligned with both K_r and K_c. In this case, we obtain the solutions γ_r and γ_c that, similar to (2), can be decomposed as follows:

$$\gamma_r^* = \mathrm{diag}(\boldsymbol{\alpha_r}) \xi_r \mathrm{diag}(\boldsymbol{\beta_r}),\ \gamma_c^* = \mathrm{diag}(\boldsymbol{\alpha_c}) \xi_c \mathrm{diag}(\boldsymbol{\beta_c})$$

where $\xi_r = e^{-\lambda M_r}$ and $\xi_c = e^{-\lambda M_c}$ are Gibbs kernels calculated between the barycenter and row and column similarity matrices using any arbitrary loss-function L as explained before. Finally, based on the analysis presented above, we further use vectors $\boldsymbol{\beta_r}$ and $\boldsymbol{\beta_c}$ to derive row and column partitions.

3.5. Detecting the number of clusters

In order to detect the steps (or jumps) in the approximated marginals, we propose to adapt a procedure introduced in (Matei & Meignen, 2012) for multiscale denoising of piecewise smooth signals. This method is of particular interest for us as it determines the significant jumps in the vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ without knowing their number and location, nor a specific threshold to decide the significance of a jump. As the proposed procedure deals with non-decreasing functions, we first sort the values of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ in the ascending order. Since the procedure is identical for both vectors, we only describe it for the vector $\boldsymbol{\alpha}$.

We consider that the elements $\{\alpha_i\}_{i=1}^n$ of $\boldsymbol{\alpha}$ are the local averages of a piecewise continuous function $v : [0,1[\subset \mathbb{R} \to \mathbb{R}$ on the intervals $I_i^n = [i/n, (i+1)/n[$ defined by the uniform subdivision of step $1/n$ of the interval $[0,1[$. More precisely: $\alpha_i^n = n \int_{I_i^n} v(t)dt, i = 0, \ldots, n-1$. The detection strategy is based on the following cost function: $F(I_i^n) = \sum_{l=i-1}^{i} |\alpha_{l+1}^n - \alpha_l^n|$ defined for each interval. Therefore, we get the list of the interval suspicious to contain a jump for the subdivision of order n as follows:

$$L^n = \{i^*; i^* = \mathrm{argmax_i F(I_i^n)}\}.$$

This detection should be refined in order to get only significant jumps in our vector $\boldsymbol{\alpha}$. To this end we use the multiscale representation of $\boldsymbol{\alpha}$ as in (Harten, 1989) and we perform this detection on each scale. On the first scale, we get a coarse version of $\boldsymbol{\alpha}$ by averaging:

$$\alpha_i^{n/2} = \frac{1}{2}(\alpha_{2i}^n + \alpha_{2i+1}^n), i = 0, \ldots, n/2 - 1.$$

Now, by considering the coarse version of $\boldsymbol{\alpha}$, we obtain a second list $L^{n/2}$ of suspicious intervals as before. After that, these two lists merge in the list L_{jumps} as follows: a jump will be considered in the interval I_{2i}^n or I_{2i+1}^n if the interval $I_i^{n/2}$ is also detected as suspicious at the coarse scale. This procedure is iterated $[\log_2 n]$ times and a jump is observed if a chain of detection exists from fine to coarse scales. Finally, the number of clusters is obtained by $g = |L_{\text{jumps}}| + 1$.

3.6. Algorithmic implementation

We now briefly summarize the main steps of both `CCOT` and `CCOT-GW` methods and discuss their peculiarities with respect to each other. The pseudocode of both approaches in Matlab are presented in Algorithm 1 and Algorithm 2, respectively.

`CCOT` First step of our algorithm consists in calculating the cost matrix M and using it to obtain the optimal coupling matrix γ_λ^* by applying the regularized optimal transport. In order to calculate M, row and column instances should both lie in a space of the same dimension. This condition, however, is verified only if the matrix $\mathcal{A}$ is squared which occurs rarely in the real-world applications. To overcome this issue, we first subsample the original data set $\mathcal{A}$ in a way that allows us to equalize the number of rows and columns and operate with two sets of the same dimension. If we assume that $n > d$ then this new reduced data set is denoted by $D \in \mathbb{R}^{d \times d}$. We repeat the sampling procedure until every individual is picked at least once.

The next step is to perform for each $i = 1, \dots, n_s$ the jump detection on the sorted vectors $\boldsymbol{\alpha}_i$ and $\boldsymbol{\beta}_i$ to obtain two lists of the jumps locations $L^{\alpha_i}_{\text{jumps}}$ and $L^{\beta_i}_{\text{jumps}}$ and to define the number of row and column clusters g and m. By using them, we obtain the resulting row partition:

$$C^i_r(\boldsymbol{x}_r) = \begin{cases} 1, & r \leq L^{\alpha_i}_{\text{jumps}}(1) \\ \dots & \\ k, & L^{\alpha_i}_{\text{jumps}}(k-1) < r \leq L^{\alpha_i}_{\text{jumps}}(k) \\ \dots & \\ |L^{\alpha_i}_{\text{jumps}}| + 1, & r > L^{\alpha_i}_{\text{jumps}}(|L^{\alpha_i}_{\text{jumps}}|). \end{cases}$$

The partition for columns $C^i_c(\boldsymbol{y}_c)$ is obtained in the same way. Finally, we apply the majority vote over all samples partitions to obtain C_r and C_c. Regarding complexity, both Sinkhorn-Knopp algorithm used to solve the regularized optimal transport (Knight, 2008) and the proposed jump detection techniques are known to converge at the linear rate multiplied by the number of samples, i.e., $\mathcal{O}(n_s d)$. On the other hand, the calculation of modes of the clustering obtained on the generated samples for both features and data instances has the complexity $\mathcal{O}(n_s(n+d))$. In the end, the complexity of the whole algorithm is $\mathcal{O}(n_s(n+d))$. We also note that in the real-world applications, we usually deal with scenarios where $n \gg d$ ("big data") or $d \ll n$ ("small" data) thus reducing the overall complexity to $\mathcal{O}(n_s n)$ and $\mathcal{O}(n_s d)$, respectively. This makes our approach even more computationally attractive.

Algorithm 1 Co-clustering through Optimal Transport (CCOT)

Input : $\mathcal{A}$ - data matrix, λ - regularization parameter, n_s - number of sampling
Output: C_r, C_c - partition matrices for rows and columns, g, m - number of row and column clusters

$[\mathtt{n}, \mathtt{d}] = \mathtt{size}(\mathtt{Z})$
for $i = 1$ **to** n_s **do**
  $\mathtt{D_i} = \mathtt{datasample}(\mathtt{Z}, \mathtt{d})$
  $\mathtt{M_i} \leftarrow \mathtt{pdist2}(\mathtt{D_i}, \mathtt{D_i^T})$
  $[\boldsymbol{\alpha}_\mathtt{i}, \boldsymbol{\beta}_\mathtt{i}, \gamma^*] \leftarrow \mathtt{optimal\_transport}(\mathtt{M_i}, \lambda)$
  $[\mathtt{L}^{\alpha_\mathtt{i}}_{\text{jumps}}, \mathtt{C^i_r}, \mathtt{g}] \leftarrow \mathtt{jump\_detection}(\mathtt{sort}(\boldsymbol{\alpha}_\mathtt{i}))$
  $[\mathtt{L}^{\beta_\mathtt{i}}_{\text{jumps}}, \mathtt{C^i_c}, \mathtt{m}] \leftarrow \mathtt{jump\_detection}(\mathtt{sort}(\boldsymbol{\beta}_\mathtt{i}))$
$\mathtt{C_r} \leftarrow \mathtt{mode}(\mathtt{C^i_r})$
$\mathtt{C_c} \leftarrow \mathtt{mode}(\mathtt{C^i_c})$

CCOT-GW As it can be seen from Algorithm 2, CCOT-GW allows to overcome the important disadvantage of CCOT that consists in the need to perform sampling to cluster all data objects. On the other hand, the computational complexity of CCOT is only $\mathcal{O}(n_s d)$, while for CCOT-GW it scales as $\mathcal{O}(n^2 d + d^2 n)$. We also note that CCOT-GW offers a great flexibility in terms of the possible data representation used at its input. One may easily consider using any arbitrary kernel function to calculate similarity matrices or even learn them beforehand using multiple-kernel learning approaches.

Algorithm 2 Co-clustering through Optimal Transport with Gromov-Wasserstein barycenters (CCOT-GW)

Input : $\mathcal{A}$ - data matrix, λ - regularization parameter, ε_r, ε_c - weights for barycenter calculation
Output: C_r, C_c - partition matrices for rows and columns, g, m - number of row and column clusters

$\mathtt{K_r} \leftarrow \mathtt{pdist2}(\mathtt{Z}, \mathtt{Z})$
$\mathtt{K_c} \leftarrow \mathtt{pdist2}(\mathtt{Z^T}, \mathtt{Z^T})$
$[\boldsymbol{\beta}_\mathtt{r}, \boldsymbol{\beta}_\mathtt{c}, \gamma^*_\mathtt{r}, \gamma^*_\mathtt{c}] \leftarrow \mathtt{gw\_barycenter}(\mathtt{K_r}, \mathtt{K_c}, \lambda, \varepsilon_\mathtt{r}, \varepsilon_\mathtt{c})$
$[\mathtt{L}^{\beta_\mathtt{r}}_{\text{jumps}}, \mathtt{C_r}, \mathtt{g}] \leftarrow \mathtt{jump\_detection}(\mathtt{sort}(\boldsymbol{\beta}_\mathtt{r}))$
$[\mathtt{L}^{\beta_\mathtt{c}}_{\text{jumps}}, \mathtt{C_c}, \mathtt{m}] \leftarrow \mathtt{jump\_detection}(\mathtt{sort}(\boldsymbol{\beta}_\mathtt{c}))$

4. Experimental evaluations

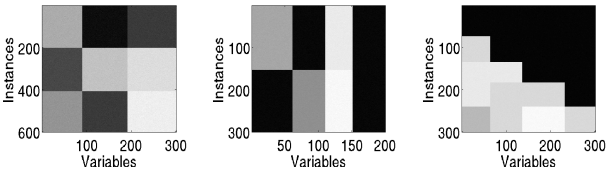
In this section, we provide empirical evaluation for the proposed algorithms.

4.1. Synthetic data

Simulation setting We simulate data following the generative process of the Gaussian Latent Block Models (for details see (Govaert & Nadif, 2013)) and we consider four scenarios with different number of co-clusters, degree of separation and size. Table 1 and Figure 1 present the characteristics of theta simulated data sets and their visualization showing the different co-clustering structures.

Table 1. Size ($n \times d$), number of co-clusters ($g \times m$), degree of overlapping ([+] for well-separated and [++] for ill-separated co-clusters) and the proportions of co-clusters for simulated data sets.

Data set	$n \times d$	$g \times m$	Overlapping	Proportions
D1	600×300	3×3	[+]	Equal
D2	600×300	3×3	[+]	Unequal
D3	300×200	2×4	[++]	Equal
D4	300×300	5×4	[++]	Unequal



(a) D2 (b) D3 (c) D4

Figure 1. D2, D3 and D4 reorganized w.r.t. the true partitions.

We use several state-of-the-art co-clustering algorithms as baselines including ITCC (Dhillon et al., 2003b), Double K-Means (DKM) (Rocci & Vichi, 2008), Orthogonal Nonnegative Matrix Tri-Factorizations (ONTMF) (Ding et al., 2006), the Gaussian Latent Block Models (GLBM) (Nadif & Govaert, 2008; Govaert & Nadif, 2013) and Residual Bayesian Co-Clustering (RBC) (Shan & Banerjee, 2010).

Table 2. Mean ($\pm$ standard-deviation) of the co-clustering error (CCE) obtained for all configurations. "-" indicates that the algorithm cannot find a partition with the requested number of co-clusters. P-values obtained using the non-parametric test of Wilcoxon (Wilcoxon, 1945) that imply significant differences are printed in bold (significance level of 0.05).

Data set	Algorithms								
	K-means	NMF	DKM	Tri-NMF	GLBM	ITCC	RBC	CCOT	CCOT-GW
D1	$\mathbf{.018 \pm .003}$	$.042 \pm .037$	$.025 \pm .048$	$.082 \pm .063$	$.021 \pm .011$	$.021 \pm .001$	$.017 \pm .045$	$.018 \pm .013$	$\mathbf{.004 \pm .002}$
D2	$.072 \pm .044$	$.083 \pm .063$	$.038 \pm .000$	$.052 \pm .065$	$.032 \pm .041$	$.047 \pm .042$	$.039 \pm .052$	$.023 \pm .036$	$\mathbf{.011 \pm .056}$
D3	–	–	$.310 \pm .000$	–	$.262 \pm .022$	$.241 \pm .031$	–	$.031 \pm .027$	$\mathbf{.008 \pm .001}$
D4	$.126 \pm .038$	–	$.145 \pm .082$	–	$.115 \pm .047$	$.121 \pm .075$	$.102 \pm .071$	$.093 \pm .032$	$\mathbf{.079 \pm .031}$

We also report the results of K-means and NMF, run on both modes of the data matrix, as clustering baseline. To assess the performance of all compared methods, we compute the co-clustering error (CCE) (Patrikainen & Meila, 2006) defined as follows:

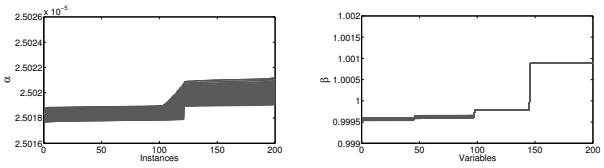
$$\text{CCE}((\mathbf{z}, \mathbf{w}), (\hat{\mathbf{z}}, \hat{\mathbf{w}})) = e(\mathbf{z}, \hat{\mathbf{z}}) + e(\mathbf{w}, \hat{\mathbf{w}}) - e(\mathbf{z}, \hat{\mathbf{z}}) \times e(\mathbf{w}, \hat{\mathbf{w}}),$$

where $\hat{\mathbf{z}}$ and $\hat{\mathbf{w}}$ are the partitions of instances and variables estimated by the algorithm; $\mathbf{z}$ and $\mathbf{w}$ are the true partitions and $e(\mathbf{z}, \hat{\mathbf{z}})$ (resp. $e(\mathbf{w}, \hat{\mathbf{w}})$) denotes the error rate, i.e., the proportion of misclassified instances (resp. features).

For all configurations, we generate 100 data sets and compute the mean and standard deviation of the CCE over all sets. As all the approaches we compared with are very sensitive to the initialization, we run them 50 times with random initializations and retain the best result according to the corresponding criterion. RBC is initialized with K-means. Regarding CCOT we set n_s to 1000 for all configurations except D4 which has the same number of rows and columns, and therefore does not require any sampling. For CCOT-GW, we use Gaussian kernels for both rows and columns with σ computed as the mean of all pairwise Euclidean distances between vectors (Kar & Jain, 2011). Finally, we let both CCOT and CCOT-GW detect automatically the number of co-clusters, while for all other algorithms we set the number of clusters to its true value.

Co-clustering performance We report the mean (and standard deviation) of co-clustering errors obtained in Table 2. Based on these results, we observe that on D1, which has a clear block structure, all algorithms perform equally well, however CCOT-GW gives the best results, closely followed by CCOT and K-means. Regarding D2, D3 and D4, which have more complicated structure than D1, both CCOT and CCOT-GW significantly outperform all other algorithms and this difference is all the more important on D3 and D4 where some of the compared algorithms are unable to find a partition with the desired number of clusters.

Furthermore, we argued that one of the strengths of our method is its ability to detect automatically the number of co-clusters by applying a jump detection algorithm on $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. From Figure 2 one can observe that the plots of these vectors, obtained with CCOT, with their elements sorted in the ascending order reveal clear steps that correspond to the correct number of clusters and also illustrate their proportions and the degree of overlapping. The same observation is valid for CCOT-GW. Both approaches correctly identified the number of clusters in most cases and CCOT is slightly more accurate than CCOT-GW when the proportions of co-clusters are unbalanced.



(a) (b)

Data Set	CCOT		CCOT-GW	
	g	m	g	m
D1	**100**	**100**	98	100
D2	**83**	**97**	81	72
D3	**99**	**98**	93	97
D4	**73**	86	70	**87**

(c)

Figure 2. Vectors (a) $\boldsymbol{\alpha}$ and (b) $\boldsymbol{\beta}$ obtained with CCOT on D3. (c) Number of times CCOT and CCOT-GW correctly detect the number of co-clusters (g and m) over 100 trials.

To summarize, CCOT and CCOT-GW outperform all the other baselines for the considered data structures and present two important advantages: (1) they do not suffer from the initialization issues, (2) they are able to detect automatically the number co-clusters.

4.2. MovieLens

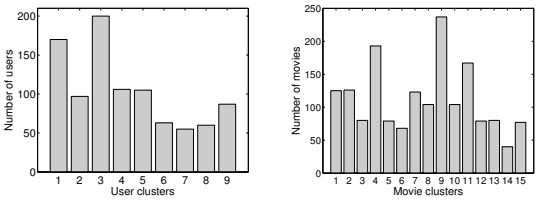
Data and setting MOVIELENS-100K[2] is a popular benchmark data set that consists of user-movie ratings, on a scale of one to five, collected from a movie recommendation service gathering 100,000 ratings from 943 users on 1682 movies. In the context of co-clustering, our goal is to find homogeneous subgroups of users and films in order to further recommend previously unseen movies that were

[2]https://grouplens.org/datasets/movielens/100k/

highly rated by the users from the same group.

We set the regularization parameters for CCOT and CCOT−GW using the cross-validation; the number of samplings for CCOT is set to 500 (as the dimensions of the data set are quite balanced); the weights for the barycenter in CCOT−GW are set to $\varepsilon = (0.5, 0.5)$.

Results In what follows we only present figures and results obtained by CCOT−GW as both algorithms return the same number of blocks and the partitions are almost identical (with a normalized mutual information between partitions above 0.8). CCOT−GW automatically detects a structure consisting of 9×15 blocks, that corresponds to 9 user clusters and 15 movie clusters. From Figure 3, one can observe that the users and the movies are almost equally distributed across clusters, except for two user and three movie clusters which have a larger size than others.



(a) (b)

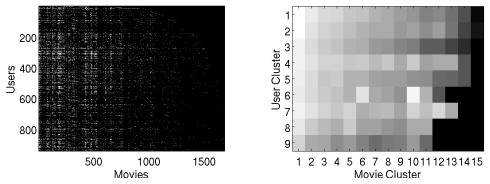
Figure 3. Distribution of the number of (a) users and (b) movies across the clusters obtained with CCOT−GW.

Figure 4 shows the original data set as well as a summarized version where each block is represented by its mean rating value (the lighter the block, the higher the ratings), revealing a structure into homogeneous groups. One can observe that the first movie cluster consists of films for which all users agree on giving high ratings (most popular movies) while the last movie cluster consists of the movies with very low ratings. We also report the 5 best rated movies in those two clusters in Table 3. One can easily see that popular movies, such that both Star Wars episodes are in M1 while M5 is composed of movies that were less critically acclaimed.

Table 3. Top 5 of movies in clusters M1 and M15.

M1	M15
Star Wars (1977)	Amytiville: A New Generation (1993)
The Lion King (1994)	Amytiville: It's About Time (1992)
Return of the Jedi (1983)	Ninjas: High Noon at Mega Mountain (1998)
Contact (1997)	Sudden Manhattan (1996)
Raiders of the lost ark (1981)	Dream Man (1995)

We can make similar observations for the interpretation of user clusters. For instance, the last two user clusters include users that tend to give less good ratings to movies than the average population. Also, we note that block $(6, 10)$ corresponds to users who liked movies from M10 better than the rest of the users. These observations are also very similar to the results reported by (Banerjee et al., 2007), where the authors proposed a detailed study of a 10×20 blocks structure for this data set. Additional results can be found in the Supplementary material.



(a) (b)

Figure 4. (a) MOVIELENS matrix; (b) the matrix summarized by the mean rating (0 ratings are excluded) for each block obtained with CCOT−GW. Darker shades indicate lower values.

5. Conclusions and future perspectives

In this paper we presented a novel approach for co-clustering based on the entropy regularized optimal transport. Our method is principally different from other co-clustering methods and consists in finding a probabilistic coupling of the empirical measures defined based on the data instances and features. We showed how this procedure can be seen as the variational inference problem and that the inferred distribution can be used to obtain the row and feature partitions. The resulting algorithm is not only more accurate than other state-of-the-art methods but also fast and capable of automatically detecting the number of co-clusters. We also presented an extended version of our algorithm that makes use of the optimal transportation distance defined on similarity matrices associated to the rows' and columns' empirical measures.

In the future, our work can be continued in multiple directions. First, we would like to extend our method in order to deal with the online setting where the goal is to classify a new previously unseen observation without the need to do the co-clustering of the data set that includes it. This can be done using a recent approach proposed in (Perrot et al., 2016) that allows to update the learned coupling matrix using the out-of-sample observations without recomputing it using all the data. We believe that this extension will make our algorithm attractive for the exploitation in real-time industrial recommendation systems due to its computational efficiency. We would also like to study the generalization properties of our algorithm in a spirit similar to the results obtained in (Maurer & Pontil, 2010). This latter work presents a rare case where the generalization bounds are derived for some famous unsupervised learning algorithms.

Acknowledgements

This work has been supported by the ANR project COCLICO, ANR-12-MONU-0001.

References

Banerjee, Arindam, Dhillon, Inderjit, Ghosh, Joydeep, Merugu, Srujana, and Modha, Dharmendra S. A generalized maximum entropy approach to bregman co-clustering and matrix approximation. *Journal of Machine Learning Research*, 8:1919–1986, December 2007.

Benamou, Jean-David, Carlier, Guillaume, Cuturi, Marco, Nenna, Luca, and Peyré, Gabriel. Iterative Bregman Projections for Regularized Transportation Problems. *SIAM Journal on Scientific Computing*, 2(37):A1111–A1138, 2015.

Bishop, Christopher M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., 2006.

Cheng, Kin-On, Law, Ngai-Fong, Siu, Wan-Chi, and Liew, Alan W. Identification of coherent patterns in gene expression data using an efficient biclustering algorithm and parallel coordinate visualization. *BMC Bioinformatics*, 9:210, 2008.

Courty, Nicolas, Flamary, Rémi, and Tuia, Devis. Domain adaptation with regularized optimal transport. In *Proceedings ECML/PKDD 2014*, pp. 1–16, 2014.

Cuturi, Marco. Sinkhorn distances: Lightspeed computation of optimal transport. In *Proceedings NIPS*, pp. 2292–2300, 2013.

Cuturi, Marco and Avis, David. Ground metric learning. *Journal of Machine Learning Research*, 15(1):533–564, 2014.

Cuturi, Marco and Doucet, Arnaud. Fast computation of wasserstein barycenters. In *Proceedings ICML*, pp. 685–693, 2014.

Deodhar, M. and Ghosh, J. Scoal: A framework for simultaneous co-clustering and learning from complex data. *ACM Transactions on Knowledge Discovery from Data*, 4(3):1–31, 2010.

Dhillon, Inderjit S., Mallela, Subramanyam, and Kumar, Rahul. A divisive information theoretic feature clustering algorithm for text classification. *Journal of Machine Learning Research*, 3:1265–1287, 2003a.

Dhillon, Inderjit S., Mallela, Subramanyam, and Modha, Dharmendra S. Information-theoretic co-clustering. In *Proceedings ACM SIGKDD*, pp. 89–98, 2003b.

Ding, C., Li, T., Peng, W., and Park, H. Orthogonal nonnegative matrix tri-factorizations for clustering. In *Proceedings ACM SIGKDD*, pp. 126–135, 2006.

George, T. and Merugu, S. A scalable collaborative filtering framework based on co-clustering. In *Proceedings ICDM*, pp. 625–628, 2005.

Govaert, G. and Nadif, M. *Co-clustering*. John Wiley & Sons, 2013.

Hanisch, Daniel, Zien, Alexander, Zimmer, Ralf, and Lengauer, Thomas. Co-clustering of biological networks and gene expression data. *BMC Bioinformatics*, 18(suppl 1):145–154, 2002.

Harten, Amiram. Eno schemes with subcell resolution. *Journal of Computational Physics*, 83:148–184, 1989.

Hartigan, J. A. Direct Clustering of a Data Matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.

Hoffman, Matthew D. and Blei, David M. Stochastic structured variational inference. In *Proceedings AISTATS*, volume 38, pp. 361–369, 2015.

I. Abraham, R. Abraham, M. Bergounioux and Carlier, G. Tomographic reconstruction from a few views: a multi-marginal optimal transport approach. *Applied Mathematics and Optimization*, pp. 1–19, 2016.

Irpino, Antonio, Verde, Rosanna, and De Carvalho, Francisco de A.T. Dynamic clustering of histogram data based on adaptive squared wasserstein distances. *Expert Systems with Applications*, 41(7):3351–3366, 2014.

Kantorovich, Leonid. On the translocation of masses. In *C.R. (Doklady) Acad. Sci. URSS(N.S.)*, volume 37(10), pp. 199–201, 1942.

Kar, Purushottam and Jain, Prateek. Similarity-based learning via data driven embeddings. In *NIPS*, pp. 1998–2006, 2011.

Keribin, Christine, Brault, Vincent, Celeux, Gilles, and Govaert, Gérard. Estimation and selection for the latent block model on categorical data. *Statistics and Computing*, 25(6):1201–1216, 2015.

Knight, Philip A. The sinkhorn-knopp algorithm: Convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275, March 2008.

Laird, N. Nonparametric maximum likelihood estimation of a mixing distribution. *Journal of the American Statistical Association*, 73:805–811, 1978.

Matei, Basarab and Meignen, Sylvain. Nonlinear cell-average multiscale signal representations: Application to signal denoising. *Signal Processing*, 92:2738–2746, 2012.

Maurer, Augusto and Pontil, Massimiliano. K -dimensional coding schemes in hilbert spaces. *IEEE Trans. Information Theory*, 56(11):5839–5846, 2010.

Mémoli, Facundo. Gromov-wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, 11(4):417–487, 2011.

Mirkin, Boris Grigorievitch. *Mathematical classification and clustering*. Nonconvex optimization and its applications. Kluwer academic publ, Dordrecht, Boston, London, 1996.

Monge, Gaspard. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences*, pp. 666–704, 1781.

Nadif, M. and Govaert, G. Algorithms for model-based block gaussian clustering. In *DMIN'08, the 2008 International Conference on Data Mining*, 2008.

Patrikainen, A. and Meila, M. Comparing subspace clusterings. *IEEE Transactions on Knowledge and Data Engineering*, 18(7):902–916, July 2006.

Perrot, Michaël, Courty, Nicolas, Flamary, Rémi, and Habrard, Amaury. Mapping estimation for discrete optimal transport. In *NIPS*, pp. 4197–4205, 2016.

Peyré, Gabriel, Cuturi, Marco, and Solomon, Justin. Gromov-wasserstein averaging of kernel and distance matrices. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, pp. 2664–2672, 2016.

Rabin, Julien, Peyré, Gabriel, Delon, Julie, and Bernot, Marc. Wasserstein barycenter and its application to texture mixing. In *Proceedings SSVM*, volume 6667, pp. 435–446, 2011.

Rocci, R. and Vichi, M. Two-mode multi-partitioning. *Computational Statistics and Data Analysis*, 52(4): 1984–2003, 2008.

Rubner, Yossi, Tomasi, Carlo, and Guibas, Leonidas J. The earth mover's distance as a metric for image retrieval. *International Journal on Computer Vision*, 40(2):99–121, 2000.

Schrödinger, E. Uber die umkehrung der naturgesetze. *Sitzungsberichte Preuss. Akad. Wiss. Berlin. Phys. Math.*, 144:144–153, 1931.

Shan, Hanhuai and Banerjee, Arindam. Residual bayesian co-clustering for matrix approximation. In *Proceedings of the SIAM International Conference on Data Mining, SDM 2010, April 29 - May 1, 2010, Columbus, Ohio, USA*, pp. 223–234, 2010.

Sinkhorn, Richard and Knopp, Paul. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific Journal of Mathematics*, 21:343–348, 1967.

Villani, Cédric. *Optimal transport : old and new*. Grundlehren der mathematischen Wissenschaften. Springer, Berlin, 2009.

Wang, Pu, Domeniconi, Carlotta, and Laskey, Kathryn. Latent dirichlet bayesian co-clustering. *Machine Learning and Knowledge Discovery in Databases*, pp. 522–537, 2009.

Wilcoxon, F. Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83, December 1945.

Wyse, Jason and Friel, Nial. Block clustering with collapsed latent block models. *Statistics and Computing*, 22(2):415–428, 2012.

Wyse, Jason, Friel, Nial, and Latouche, Pierre. Inferring structure in bipartite networks using the latent block model and exact ICL. *ArXiv e-prints*, 2014.

Xu, Bin, Bu, Jiajun, Chen, Chun, and Cai, Deng. An exploration of improving collaborative recommender systems via user-item subgroups. In *Proceedings WWW*, pp. 21–30, 2012.