Shift of pairwise similarities for data clustering

Morteza Haghir Chehreghani[1] (ORCID)

Abstract

Several clustering methods (e.g., *Normalized Cut* and *Ratio Cut*) divide the *Min Cut* cost function by a cluster dependent factor (e.g., the size or the degree of the clusters), in order to yield a more balanced partitioning. We, instead, investigate adding such regularizations to the original cost function. We first consider the case where the regularization term is the sum of the squared size of the clusters, and then generalize it to adaptive regularization of the pairwise similarities. This leads to shifting (adaptively) the pairwise similarities which might make some of them negative. We then study the connection of this method to *Correlation Clustering* and then propose an efficient *local search* optimization algorithm with fast theoretical convergence rate to solve the new clustering problem. In the following, we investigate the shift of pairwise similarities on some common clustering methods, and finally, we demonstrate the superior performance of the method by extensive experiments on different datasets.

Keywords Unsupervised learning · Clustering · Shift of pairwise similarities · Local search optimization · Correlation clustering

1 Introduction

Given a set of objects, clustering is concerned with grouping them in such a way that objects of the same group are more similar to each other (according to a predefined similarity measure), compared to those in different groups. This task plays a fundamental role in several data analytics applications. Examples are image segmentation (to detect the items in images), document clustering (for the purpose of document organization, topic identification or efficient information retrieval), data compression, and analysis of (e.g., transportation) networks and graphs. Clustering itself is not a specific method, rather is a general machine learning task to be addressed. The task can be solved via several methods that

---

✉ Morteza Haghir Chehreghani
  morteza.chehreghani@gmail.com

[1]   Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Sweden

differ significantly in the way they define the notion of clusters and the way they extract them. The concept of clustering is originated from anthropology and then was used in psychology (Tryon, 1939; Bailey, 1994), in particular for trait theory classification in personality psychology (Cattell, 1943).

A wide range of clustering methods introduce a cost function whose minimal solution provides a clustering solution. *K*-means is a common cost function which is defined by the within-cluster sum of squared distances from the means (Macqueen, 1967). The data can be demonstrated by a graph, whose nodes represent the objects and the edge weights are the pairwise similarities between the objects. Then, a wide range of different graph partitioning methods can be applied to produce the clusters. Arguably, the most basic graph-based method is the *Min Cut* (*Minimum K-*Cut) cost function (Leighton & Rao, 1999; Wu & Leahy, 1993), in which the goal is to partition the graph into exactly *K* connected components (clusters) such that the sum of the inter-clusters edge weights is minimal. As we will see, the *Min Cut* cost function often yields separating singleton clusters, in particular when the clusters have diverse densities. To overcome such problem, several clustering methods normalize the *Min Cut* clusters to render more balanced clusters. For example, they propose to normalize the *Min Cut* clusters by the size of the clusters (*Ratio Assoc* (Hofmann & Buhmann, 1997) and *Ratio Cut* (Chan et al., 1994)) or the degree of the clusters (*Normalized Cut* (Shi & Malik, 2000)).

We note that balanced clustering has been studied for featured-based (vectorial) data as well, in particular with the *K*-means method. The method in Malinen & Fränti (2014) develops balanced clustering via formulating it as a assignment problem by the Hungarian algorithm, which suffers from a high runtime (cubic w.r.t. the number of objects). Another work models this problem as a least square linear regression with a balance constraints and uses the method of augmented Lagrange multipliers to solve it (Liu et al., 2017). The work in Liu et al. (2018) considers *K*-means as the main clustering method and the respective cluster variances as the penalty term. Then, Lin et al. (2019) yields balanced clustering with convex regularization which makes the optimization more efficient. In the following, Ding (2020) studies balanced *K*-center, *K*-median, and *K*-means in high dimensions with theoretical approximate algorithms. Finally, Han et al. (2019) proposes a balanced clustering framework that utilizes both local and global information. However, in this paper, we consider the 'graph-based' balanced clustering variant, where we assume the clustering is applied to a given graph, instead of data features.

While most of graph clustering cost functions assume a nonnegative matrix of pairwise similarities as input, *Correlation Clustering* assumes that the similarities can be negative as well. This cost function was first introduced on graphs with only $+1$ and -1 edge weights (Bansal et al., 2004), and then it was generalized to graphs with arbitrary positive and negative edge weights (Demaine et al., 2006).

Such graph clustering cost functions are often NP-hard (Shi & Malik, 2000; Bansal et al., 2004; Demaine et al., 2006). However, the respective optimal solution can be approximated in some way. A category of methods work based on eigenvector analysis of the Laplacian matrix. *Spectral Clustering* (Shi & Malik, 2000; Ng et al., 2001) was the first method which exploits the information from eigenvectors. It forms a low-dimensional embedding by the bottom eigenvectors of the Laplacian of the similarity matrix and then applies *K*-means to produce the final clusters. A more recent method, called *Power Iteration Clustering* (PIC) (Lin & Cohen, 2010), instead of embedding the data into a *K*-dimensional space, approximates an eigenvalue-weighted linear combination of all the eigenvectors of the normalized similarity matrix via early stopping of the power iteration method. *P-Spectral Clustering* (PSC) (Bühler & Hein, 2009) is another spectral approach that

proposes a non-linear generalization of the Laplacian and then performs an iterative splitting method based on its second eigenvector.

An alternative graph-based clustering approach has been developed in the context of discrete time dynamical systems and evolutionary game theory which is based on performing replicator dynamics (Pavan & Pelillo, 2007; Ng et al., 2012; Liu et al., 2013). *Dominant Set Clustering* (DSC) (Pavan & Pelillo, 2007) is an iterative method which at each iteration, peels off a cluster by performing a replicator dynamics until its convergence. The method in Liu et al. (2013) proposes an iterative clustering algorithm in two shrink and expansion steps, which helps to extract many small and dense clusters in large datasets. The method in Bulò et al. (2011), called *InImDyn*, instead of replicator dynamics, suggests to use a population dynamics motivated from the analogy with infection and immunization processes within a population of players.

In this paper, we investigate adding the regularization terms to the *Min Cut* cost function, in order to avoid creation of small singleton sets of clusters. We first consider the case where the regularization is the sum of the squared size of the clusters, weighted by the parameter α. This regularization leads to a simple shift transformation of the input, i.e., subtracting the pairwise similarities by α, which provides a straightforward quadratic cost function. We further extend the regularization to the pairwise similarities and employ an adaptive shift of the pairwise similarities which does not require fixing a regularization parameter in advance. The size constrained *Min Cut* then constitutes a special case of the latter form. Such a shift might render some pairwise similarities to be negative. We then study the connection to *Correlation Clustering*, another cost function which performs on both positive and negative similarities, and conclude the equivalence of these two methods given the shifted (regularized) pairwise similarities in a direct and straightforward way (beyond the argument based on algorithmic reduction proposed in Demaine et al. (2006)). However, our method, called *Shifted Min Cut*, provides a principled way to deduce such negative edge weights (adaptively). Thereafter, we develop an efficient optimization method based on *local search* to solve the new optimization problem. We further discuss the fast theoretical convergence rate of this local search algorithm. In the following, we study the impact of shifting the pairwise similarities on some common flat and hierarchical clustering methods where they often exhibit an invariant behaviour with respect to the shift of pairwise similarities, unlike the basic *Min Cut* cost function. Finally, we perform extensive experiments on several real-world datasets to study the performance of *Shifted Min Cut* compared to the alternatives.

This work is an extension of our previous work (Chehreghani, 2017) wherein we additionally, (i) provide an argument on the theoretical convergence rate of the local search algorithm based on the connection to an optimized variant of Frank-Wolfe algorithm, (ii) discuss the shift of pairwise similarities on several other clustering methods, and (iii) elaborate further the existing experimental results and perform extra studies on real-world datasets. We have later found out that the work in Chen et al. (2005) suggests a similar idea for regularization of *Min Cut* in order to yield balanced clusters. However, there are several fundamental differences between Chen et al. (2005) and our work: (i) they study size constrained *Min Cut* for bi-partitioning (i.e., for only two clusters), whereas we model it for arbitrary K clusters. Then, to generate more clusters than two, they propose an iterative (sequential) bi-partitioning which might cause the re-scaling problem. (ii) Their method requires fixing critical hyperparameters often in a heuristic way, whereas our method does not include such hyperparameters. (iii) Beyond size constrained *Min Cut*, we extend the method to refined regularization of the pairwise similarities that yields an adaptive regularization (shift) of the cost function. This adaptive regularization, not only provides

adaptivity with respect to the type of the relations, but also obviates the need for fixing critical hyperparameters. (iv) We consider that the regularization renders some of the pairwise similarities to be negative, and thereby, we study the connection between such a regularized (*Shifted*) *Min Cut* method and *Correlation Clustering*. However, Chen et al. (2005) does not study such a connection. (v) To optimize the respective cost function, we employ integration of the regularizations into shifting the pairwise similarities and develop an efficient local search algorithm that enjoys a linear convergence rate. Chen et al. (2005), instead, develops approximate spectral solutions. (vi) We demonstrate the performance of the method on several real-world datasets with respect to different evaluation criteria, whereas Chen et al. (2005) only studies the mutual information evaluation criterion on two datasets. In particular, we investigate both the cost function and its optimization separately.

The rest of the paper is organized as following. In Sect. 2, we introduce the notations and the definitions. Then, in Sect. 3, we describe the regularization and the connection to shifting the pairwise similarities. In this section, we extend the method to adaptive regularization (shift) of the pairwise similarities. In Sect. 4, we study the connection between *Shifted Min Cut* and *Correlation Clustering*, and, in Sect. 5, we develop an efficient local search optimization method for the cost function. In Sect. 6, we study the consequence of shifting pairwise relations in some other (flat and hierarchical) clustering methods. In Sect. 7, we experimentally investigate the different aspects of the method on several real-world datasets, and finally, in Sect. 8, we conclude the paper.

2 Notations and definitions

The data is given by a set of n objects $\mathbf{O} = \{1, ..., n\}$ and the corresponding matrix of pairwise similarities $\mathbf{X} = \{\mathbf{X}_{ij}\}, \forall i, j \in \mathbf{O}$. Thus, the data can be represented by (an undirected) graph $\mathcal{G}(\mathbf{O}, \mathbf{X})$, where the objects $\mathbf{O}$ constitute the nodes of the graph and $\mathbf{X}_{ij}$ represents the weight of the edge between i and j. Then, the goal is to partition the objects (the graph) into K coherent groups which are distinguishable from each other. The clustering solution is encoded in $\mathbf{c} \in \{1, ..., K\}^n$, i.e., $\mathbf{c}_i$ indicates the cluster label of the ith object. The vector $\mathbf{c}$ can be also represented via the co-clustering matrix $\mathbf{H} \in \{0, 1\}^{n \times n}$.

$$\mathbf{H}_{ij} = \begin{cases} 1 & \text{iff } \mathbf{c}_i = \mathbf{c}_j \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

$\mathcal{C}$ denotes the space of all different clustering solutions.

Moreover, we assume $\mathbf{O}_k \subset \mathbf{O}$ includes the members of the kth cluster, i.e.,

$$\mathbf{O}_k := \{i \in \mathbf{O} \; : \; \mathbf{c}_i = k\} \, . \tag{2}$$

$|\mathbf{O}_k|$ refers to the size of the kth cluster.

3 Shift of pairwise similarities for clustering

Different graph-based clustering methods often consider the *Min Cut* cost function as a base method which is defined by

Fig. 1 The *Min Cut* cost function has a bias to split small (singleton) sets of objects. Any cut that splits one of the objects on the right half will have smaller cost than the cut that splits the objects into the left and right halves. The figure has been adapted from Shi and Malik (2000)
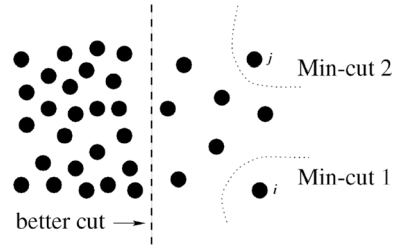
$$R^{MC}(c, \mathbf{X}) = \sum_{k=1}^{K} \sum_{\substack{k'=1, \\ k' \neq k}}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij}, \tag{3}$$

This cost function has a tendency to split small sets of objects, since the cost increases with the number of inter-cluster edge weights, i.e., the edges connecting the different clusters. Figure 1 illustrates such a situation for two clusters (Shi & Malik, 2000). We assume that the edge weights are inversely proportional to the distances between the objects. It is observed *Min Cut* favors splitting objects i or j, instead of performing a more balanced split. In fact, any cut that splits one of the objects on the right half will yield a smaller cost than the cut that partitions the objects into the left and right halves. This issue is particularly problematic when the intra-cluster edge weights are heterogeneous among different clusters. Thus, several methods propose to normalize the *Min Cut* clusters by a cluster depending factor, e.g., the size of clusters (*Ratio Assoc* (Hofmann & Buhmann, 1997) and *Ratio Cut* (Chan et al., 1994)) or the degree of clusters (*Normalized Cut* (Shi & Malik, 2000)).

We investigate an alternative approach to yield the occurrence of more balanced clusters. Instead of normalizing (dividing) the *Min Cut* cost function by a cluster-dependent function, we consider adding such a regularization to the original cost function, i.e.,

$$R^{new}(\mathbf{c}, \mathbf{X}, \alpha) = R^{MC}(\mathbf{c}, \mathbf{X}) + \alpha \cdot r(\mathbf{c}, \mathbf{X}), \tag{4}$$

where $r(\mathbf{c}, \mathbf{X})$ indicates the regularization. Note that this formulation involves the two free choices α and $r(\mathbf{c}, \mathbf{X})$, thereby, it yields a richer family of alternative methods. We first focus on the case where $r(\mathbf{c}, \mathbf{X})$ is the sum of the squared size of the clusters,[1] i.e.,

$$R^{new}(\mathbf{c}, \mathbf{X}) = R^{MC}(\mathbf{c}, \mathbf{X}) + \alpha \sum_{k=1}^{K} |\mathbf{O}_k|^2. \tag{5}$$

Thereby,

1. If $\alpha < 0$, then the term $\alpha \sum_{k=1}^{K} |\mathbf{O}_k|^2$ is minimal when only the singleton clusters (objects) are separated. Thus, this choice does not help to avoid occurrence of singleton clusters, rather, it accelerates.

---

[1] The *Min Cut* cost function is quadratic with respect to the number of edges, therefore, to be consistent, we use the squared form of the cluster cardinalities.

Fig. 2 The impact of the shift parameter α on the results of the *Shifted Min Cut* cost function. A very large α might yield splitting large clusters, instead of separating true small clusters
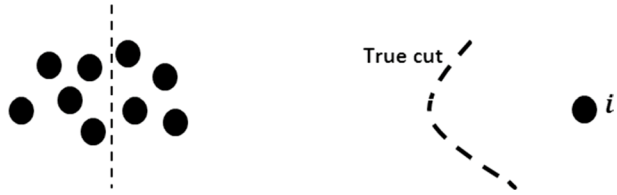
2. If $\alpha > 0$, then $\alpha \sum_{k=1}^{K} |\mathbf{O}_k|^2$ is minimal for balanced clusters, i.e., when $|\mathbf{O}_k| \approx n/K$, $\forall k \in \{1, ..., K\}$. This leads to equalize the size of clusters. We note that $|\mathbf{O}_k|$'s are integer numbers, but n/K is not necessarily an integer. Thus, we may arbitrarily set some of the $|\mathbf{O}_k|$'s to $\lceil n/K \rceil$ and some others to $\lfloor n/K \rfloor$ such that $\sum_{k=1}^{K} |\mathbf{O}_k| = n$. The order would not change the minimum.

The cost function in Eq. (5) can be further written as

$$
\begin{aligned}
R^{new}(\mathbf{c}, \mathbf{X}, \alpha) &= R^{MC}(\mathbf{c}, \mathbf{X}) + \alpha \sum_{k=1}^{K} |\mathbf{O}_k|^2 \\
&= \sum_{k=1}^{K} \sum_{k' \neq k}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij} + \alpha \sum_{k=1}^{K} |\mathbf{O}_k|^2 \\
&= \sum_{k=1}^{K} \sum_{k' \neq k}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij} + \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \mathbf{X}_{ij} - \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \mathbf{X}_{ij} + \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \alpha \\
&= \sum_{k=1}^{K} \sum_{k'=1}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij} - \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} (\mathbf{X}_{ij} - \alpha) \\
&= \underbrace{\sum_{i,j \in \mathbf{O}} \mathbf{X}_{ij}}_{constant} - \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} (\mathbf{X}_{ij} - \alpha) \\
&= - \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} (\mathbf{X}_{ij} - \alpha) + constant \, .
\end{aligned}
\tag{6}
$$

Therefore, we define

$$
R^{SMC}(\mathbf{c}, \mathbf{X}, \alpha) = - \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} (\mathbf{X}_{ij} - \alpha)
$$

Thus, we employ a shifted variant of *Min Cut* cost function (called *Shifted Min Cut*), wherein all pairwise similarities are subtracted by a positive parameter α, such that some of the pairwise similarities might become negative. It makes sense that the regularization on the size of the clusters becomes connected to the pairwise similarities, as, at the end, pairwise relations are responsible for creating the clusters. Thus, by tuning them properly, one should be able to obtain the desired balanced clusters. Thereby, the cluster level regularization is effectively applied to the representation space, where, as will be discussed, it yields modelling and computational advantages.

This formulation provides a rich family of alternative clustering methods where different regularizations are induced by different values of α. However, choosing a very large α can lead to equalizing the size of the clusters that are inherently very unbalanced in size. For example, consider the dataset shown in Fig. 2. We assume that the edge weights are inversely proportional to the pairwise distances. Then, we subtract all pairwise similarities by a very large number. Therefore, the pairwise similarities become very large but negative numbers which renders *Shifted Min Cut* to produce equal-size clusters, even though a correct cut should separate only the object i from the rest. Thus, in practice one needs to examine different values of α, and choose the one that yields the best results, or is preferred by the user. However, this procedure might be computationally expensive, and, moreover, the user might not be able to validate the correct solution among many different alternatives, due to lack of enough prior knowledge, supervision or side information. For this reason, we employ a particular shift of pairwise similarities which takes the connectivity of the objects into account and does not need fixing any free parameter.

Adaptive shift of pairwise similarities Different pairwise similarities might need different shifts, depending on the type and the density of the clusters that the respective objects belong to. Therefore, we relax the constraints of the formulation in Eq. (6) and consider a separate shift parameter for every pairwise similarity $\mathbf{X}_{ij}$.

$$R^{SMC}(\mathbf{c}, \mathbf{X}, \{\alpha_{ij}\}) = -\sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} (\mathbf{X}_{ij} - \alpha_{ij}), \qquad \alpha_{ij} = \alpha_{ji}. \tag{7}$$

The formulation in Eq. (7) already involves the formulation in Eq. (6) as a special case where all α_{ij}'s are fixed by a constat. To determine α_{ij}'s properly, a reasonable approach is to shift the pairwise similarity $\mathbf{X}_{ij}$ between i and j adaptively with respect to the similarities between i and all the other objects and as well as the similarities between j and the other objects. For this purpose, we shift $\mathbf{X}_{ij}$ such that the sum of the pairwise similarities between i and all the other objects becomes zero, and the same holds for j too. In this way, we have

$$\alpha_{ij} = \frac{1}{n} \sum_{p=1}^{n} \mathbf{X}_{ip} + \frac{1}{n} \sum_{p=1}^{n} \mathbf{X}_{pj} - \frac{1}{n^2} \sum_{p=1}^{n} \sum_{q=1}^{n} \mathbf{X}_{pq}. \tag{8}$$

Summing up the regularizations for all pairs of objects, we have (we assume $\mathbf{X}$ is symmetric):

$$\sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \alpha_{ij} = \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \left(\frac{1}{n} \sum_{p=1}^{n} \mathbf{X}_{ip} + \frac{1}{n} \sum_{p=1}^{n} \mathbf{X}_{pj} - \frac{1}{n^2} \sum_{p=1}^{n} \sum_{q=1}^{n} \mathbf{X}_{pq} \right)$$
$$= \frac{2}{n} \sum_{k=1}^{K} |\mathbf{O}_k| deg(k) - \frac{\beta}{n^2} \sum_{k=1}^{K} |\mathbf{O}_k|^2, \tag{9}$$

where $deg(k)$ is the degree of cluster k, i.e., $deg(k) = \sum_{i \in \mathbf{O}_k} \sum_{p=1}^{n} \mathbf{X}_{ip}$, and constant β is the sum of the given pairwise similarities, i.e., $\beta = \sum_{p=1}^{n} \sum_{q=1}^{n} \mathbf{X}_{pq}$. Therefore, the adaptive regularization yields a tradeoff between the size of the clusters and the degree of the clusters. The former is used in *Ratio Assoc* and the latter in *Normalized Cut*, both in the denominator. However, here a combination of these two is assumed, but as additive terms.

Therefore, the new shifted similarity $\mathbf{S}_{ij}$ is obtained by

$$\mathbf{S}_{ij} = \mathbf{X}_{ij} - \frac{1}{n}\sum_{p=1}^{n}\mathbf{X}_{ip} - \frac{1}{n}\sum_{p=1}^{n}\mathbf{X}_{pj} + \frac{1}{n^2}\sum_{p=1}^{n}\sum_{q=1}^{n}\mathbf{X}_{pq}. \tag{10}$$

It is easy to check that $\mathbf{S}$ is symmetric, provided that $\mathbf{X}$ is symmetric. It can be shown that sum of the rows and the columns of $\mathbf{S}$ are equal to zero. For example, for a fixed row i we have

$$
\begin{aligned}
\sum_{j=1}^{n}\mathbf{S}_{ij} &= \sum_{j=1}^{n}\mathbf{X}_{ij} - \frac{1}{n}\sum_{j=1}^{n}\sum_{p=1}^{n}\mathbf{X}_{ip} - \frac{1}{n}\sum_{j=1}^{n}\sum_{p=1}^{n}\mathbf{X}_{pj} + \frac{1}{n^2}\sum_{j=1}^{n}\sum_{p=1}^{n}\sum_{q=1}^{n}\mathbf{X}_{pq}\\
&= \sum_{j=1}^{n}\mathbf{X}_{ij} - \frac{n}{n}\sum_{p=1}^{n}\mathbf{X}_{ip} - \frac{1}{n}\sum_{j=1}^{n}\sum_{p=1}^{n}\mathbf{X}_{pj} + \frac{n}{n^2}\sum_{p=1}^{n}\sum_{q=1}^{n}\mathbf{X}_{pq}\\
&= 0 + 0.
\end{aligned}
\tag{11}
$$

The adaptive shift in Eq. (10) can be written in matrix form as

$$\mathbf{S} = \mathbf{TXT}, \tag{12}$$

where the $n \times n$ matrix $\mathbf{T}$ is defined by

$$\mathbf{T} = \mathbf{I}_n - \frac{1}{n}\mathbf{U}. \tag{13}$$

$\mathbf{U}$ is an $n \times n$ matrix whose all elements are 1.[2]

Thus, according to Eqs. (6) and (7), the new cost function is written by

$$R^{SMC}(\mathbf{c}, \mathbf{S}) = -\sum_{k=1}^{K}\sum_{i,j\in\mathbf{O}_k}\mathbf{S}_{ij} \tag{14}$$

$$\equiv \sum_{k=1}^{K}\sum_{k'\neq k}^{K}\sum_{i\in\mathbf{O}_k}\sum_{j\in\mathbf{O}_{k'}}\mathbf{S}_{ij}. \tag{15}$$

As an alternative to the adaptive shift, a proper shift can be obtained by investigating few pairwise relations by a user (i.e., a kind of weak supervision). In this setting, the user tells us how the actual pairwise relations should look like for a small subset of them, i.e., weather they are in the same cluster (positive shift) or different clusters (negative shift). Then, given this feedback, we can generalize them to all the pairwise relations. We may train a model, e.g. a neural network, which learns the shift depending on the specifications of the respective edge and objects. Such an approach can be even combined with our method for adaptive shift of pairwise similarities, where the later is used as an initial guess for the shifted pairwise relations and then they are fine tuned further using the user feedbacks if needed. This formulation also provides a convenient way to encode constraints and prior knowledge such as 'objects x and y must be together', and 'objects p and q must be in different clusters'.

---

[2] Due to shift invariance of *Ratio Assoc*, a similar shift is used in Roth et al. (2003) to render the respective eigenvalues non-negative and thus obtain an embedding for the pairwise relations. However, here, the shift is used for a totally different purpose and it yields changing the size of clusters.

4 Relation to correlation clustering

Correlation Clustering is a clustering cost function that partitions a graph with positive and negative edge weights. The cost function sums the disagreements, i.e., the sum of negative intra-cluster edge weights plus the sum of positive inter-cluster edge weights. The respective cost function on general graphs is defined by Demaine et al. (2006)

$$R^{CC}(c, \mathbf{X}) = \sum_{(i,j) \in E^{<+>}} \mathbf{X}_{ij}(1 - \mathbf{H}_{ij}) - \sum_{(i,j) \in E^{<->}} \mathbf{X}_{ij}\mathbf{H}_{ij}, \tag{16}$$

where $E^{<->}$ and $E^{<+>}$ respectively indicate the set of the edges with negative and with positive weights. The approximation scheme in Demaine et al. (2006) reduces *Min Cut* to *Correlation Clustering* in order to obtain a logarithmic approximation factor for *Correlation Clustering*. It also develops a reduction from *Correlation Clustering* to *Min Cut* to conclude the equivalence of these two cost functions. Here, we elaborate that these two cost functions are identical and represent the same objective (given the shifted pairwise similarities) in a direct and straightforward way without using the more complicated reduction argument. In addition, Demaine et al. (2006) assumes that the number of clusters is hidden in the cost function (as defined in Eq. (16)). However, we study the equivalence for any arbitrary number of clusters K. As shown in Chehreghani et al. (2012), Frank et al. (2011), optimizing *Correlation Clustering* without a constraint on the number of clusters can lead to overfitting and unrobust solutions, whereas fixing the number of clusters may avoid these issues. Therefore, we consider the setting where the number of clusters K is explicitly specified in the cost function and the user has the possibility to fix it in advance. Finally, the reduction-based argument in Demaine et al. (2006) yields the equivalence of the optimal solutions between *Min Cut* and *Correlation Clustering* and the respective approximation and hardness results. We, in addition, conclude the equivalence of any local optimal solution for the two cost functions, which is important when using *local search* algorithms to optimize the cost functions.

For a fixed K, the *Correlation Clustering* cost function can be written as Chehreghani et al. (2012), Frank et al. (2011)

$$R^{CC}(c, \mathbf{X}) = \underbrace{\frac{1}{2} \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} (|\mathbf{X}_{ij}| - \mathbf{X}_{ij})}_{a} + \underbrace{\frac{1}{2} \sum_{k=1}^{K} \sum_{\substack{k'=1, \\ k' \neq k}}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} (|\mathbf{X}_{ij}| + \mathbf{X}_{ij})}_{b}. \tag{17}$$

The first term (called a) sums the intra-cluster negative edge weights, whereas the second term (called b) sums the inter-cluster positive edge weights. We separately expand each term.

$$
\begin{aligned}
a &= \frac{1}{2} \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} |\mathbf{X}_{ij}| - \frac{1}{2} \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \mathbf{X}_{ij} \\
&= \frac{1}{2} \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} |\mathbf{X}_{ij}| - \underbrace{\frac{1}{2} \sum_{k=1}^{K} \sum_{k'=1}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij}}_{\text{constant}} + \frac{1}{2} \sum_{k=1}^{K} \sum_{\substack{k'=1, \\ k' \neq k}}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij}.
\end{aligned}
\tag{18}
$$

Similarly, we expand term *b*.

$$b = \frac{1}{2} \sum_{k=1}^{K} \sum_{\substack{k'=1, \\ k' \neq k}}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} |\mathbf{X}_{ij}| + \frac{1}{2} \sum_{k=1}^{K} \sum_{\substack{k'=1, \\ k' \neq k}}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij} . \tag{19}$$

Then, by summing *a* and *b* we obtain

$$R^{CC}(c, \mathbf{X}) = constant$$

$$+ \frac{1}{2} \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} |\mathbf{X}_{ij}| + \underbrace{\frac{1}{2} \sum_{k=1}^{K} \sum_{\substack{k'=1, \\ k' \neq k}}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} |\mathbf{X}_{ij}|}_{constant} + \underbrace{\sum_{k=1}^{K} \sum_{\substack{k'=1, \\ k' \neq k}}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij}}_{R^{MC}(\mathbf{c},\mathbf{X})} . \tag{20}$$

Thus, *Correlation Clustering* and *Min Cut* are equivalent cost functions, i.e.,

1. The cost functions share the same optimal solution, i.e., $\arg\min_{\mathbf{c}} R^{MC}(\mathbf{c}, \mathbf{X}) = \arg\min_{\mathbf{c}} R^{CC}(\mathbf{c}, \mathbf{X})$.
2. The costs differences are the same, i.e., $\forall \mathbf{c} \in \mathcal{C} : R^{MC}(\mathbf{c}, \mathbf{X}) - \min_{\mathbf{c}} R^{MC}(\mathbf{c}, \mathbf{X}) = R^{CC}(\mathbf{c}, \mathbf{X}) - \min_{\mathbf{c}} R^{CC}(\mathbf{c}, \mathbf{X})$. This is in particular relevant when defining for example a Boltzmann distribution over the solution space $\mathcal{C}$.

Thus, *Correlation Clustering*, similar to *Shifted Min Cut*, is an extension of *Min Cut* which deals with both negative and positive edge weights. However, there are fundamental differences between these two methods:

1. *Correlation Clustering* assumes that the matrix of pairwise positive and negative similarities is given (which might be nontrivial), whereas *Shifted Min Cut* proposes a principled way to yield clustering of positive and negative similarities via regularizing the base *Min Cut* cost function. Thus, *Shifted Min Cut* provides an explicit and straightforward interpretation of the clustering problem.
2. The form of the *Shifted Min Cut* cost function expressed in Eq. (14) provides efficient function evaluations (e.g., for optimization) compared to the *Correlation Clustering* cost function in Eq. (17) or the base *Min Cut* cost function in Eq. (3). The cost functions in Eqs. (3) and (17) are quadratic with respect to *K*, the number of clusters, whereas the cost function in Eq. (14) is linear.

5 Optimization of the *shifted min cut* cost function

Finding the optimal solution of the standard *Min Cut* with non-negative edge weights, i.e., when $\mathbf{X}_{ij} \geq 0, \forall i,j$, is well-studied, for which there exist several polynomial time algorithms, e.g., $\mathcal{O}(n^4)$ (Goldschmidt & Hochbaum, 1994) and $\mathcal{O}(n^2 \log^3 n)$ (Karger & Stein, 1996). However, finding the optimal solution of the *Shifted Min Cut* cost function, wherein some edge weights are negative, is NP-hard (Bansal et al., 2004; Demaine et al., 2006) and

even is APX-hard (Demaine et al., 2006). Therefore, we develop a *local search* method which computes a local minimum of the cost function in Eq. (14). The effectiveness of such a greedy strategy is well studied for different clustering cost functions, e.g., *K*-means (Macqueen, 1967), kernel *K*-means (Schölkopf et al., 1998) and in particular several graph partitioning methods (Dhillon et al., 2004, 2005).[3] In this approach, we start with a random clustering solution and then we iteratively assign each object to the cluster that yields a maximal reduction in the cost function. We repeat this procedure until no further change of assignments is achieved during a complete round of investigation of the objects, i.e., then a local optimal solution is attained.

At each iteration of the aforementioned procedure, one needs to evaluate the cost of assigning every object to each of the clusters. The cost function is quadratic, thus a single evaluation might take $\mathcal{O}(Kn^2)$ runtime. Thereby, if the local search converges after t iterations, then, the total runtime will be $\mathcal{O}(tKn^3)$ for n objects, which might be computationally expensive.

However, we do not need to recalculate the cost function for every individual evaluation. Let $R^{SMC}(\mathbf{c}_{o \to l}, \mathbf{S})$ denote the cost of the clustering solution $\mathbf{c}$ wherein object o is assigned to cluster l. At each step of the local search algorithm, we need to evaluate the cost $R^{SMC}(\mathbf{c}_{o \to l'}, \mathbf{S}), l' \neq l$ given $R^{SMC}(\mathbf{c}_{o \to l}, \mathbf{S})$.

The cost $R^{SMC}(\mathbf{c}_{o \to l}, \mathbf{S})$ is written by

$$R^{SMC}(\mathbf{c}_{o \to l}, \mathbf{S}) = -\sum_{k=1}^{K} \sum_{\substack{i,j \in \mathbf{O}_k \\ i,j \neq o}} \mathbf{S}_{ij} - \sum_{\substack{i \in \mathbf{O}_l \\ i \neq o}} (\mathbf{S}_{io} + \mathbf{S}_{oi}) - \mathbf{S}_{oo}. \tag{21}$$

Similarly, the cost $R^{SMC}(\mathbf{c}_{o \to l'}, \mathbf{S}), l' \neq l$ is obtained by

$$\begin{aligned} R^{SMC}(\mathbf{c}_{o \to l'}, \mathbf{S}) &= -\sum_{k=1}^{K} \sum_{\substack{i,j \in \mathbf{O}_k \\ i,j \neq o}} \mathbf{S}_{ij} - \sum_{\substack{i \in \mathbf{O}_{l'} \\ i \neq o}} (\mathbf{S}_{io} + \mathbf{S}_{oi}) - \mathbf{S}_{oo} \\ &= R^{SMC}(\mathbf{c}_{o \to l}, \mathbf{S}) + \sum_{\substack{i \in \mathbf{O}_l \\ i \neq o}} (\mathbf{S}_{io} + \mathbf{S}_{oi}) - \sum_{\substack{i \in \mathbf{O}_{l'} \\ i \neq o}} (\mathbf{S}_{io} + \mathbf{S}_{oi}). \end{aligned} \tag{22}$$

Thus, given $R^{SMC}(\mathbf{c}_{o \to l}, \mathbf{S})$ the runtime of a new evaluation of the cost function $R^{SMC}(\mathbf{c}_{o \to l'}, \mathbf{S})$ is $\mathcal{O}(n)$. Hence, the total runtime of the local search method will be $\mathcal{O}(tn^2)$. Therefore, at the beginning, we compute a random initial solution, wherein each object is assigned randomly to one of K clusters, and compute the respective cost. At each iteration, we use Eq. (22) to investigate the cost of assigning an object to the other clusters than the current one. Then, we assign the object to the cluster that yields a maximal reduction in the cost. We might repeat the local search algorithm with several random initializations and at end, choose a solution with a minimal cost. Note that even the efficient evaluation and

---

[3] Consistently, with *Correlation Clustering* we observe a significantly better performance of the local search algorithm compared to approximation schemes such as those proposed in Bansal et al. (2004), Demaine et al. (2006).

optimization of the variants in Eqs. (3) and (17) would yield $\mathcal{O}(tKn^2)$ total runtime, i.e., K times slower than the variant expressed in Eq. (14).

We note that this technique can be employed with other optimization or inference methods as well, such as MCMC methods and simulated annealing.

On the convergence rate of the local search optimization With the co-authors, we have shown in Thiel et al. (2019) that for *Correlation Clustering*, Frank-Wolfe optimization with line search for the update parameter (to find the optimal learning rate) is equivalent to the local search algorithm. On the other hand, we have established convergence rate of $\mathcal{O}(\frac{1}{t})$ for Frank-Wolfe optimization applied to *Correlation Clustering* (Thiel et al., 2019) (t indicates the optimization step). As discussed before, given the shifted pairwise similarities, *Shifted Min Cut* is equivalent to *Correlation Clustering*. Thus, the same argument holds for the aforementioned local search algorithm for *Shifted Min Cut*, i.e., *Shifted Min Cut* enjoys the convergence rate of $\mathcal{O}(\frac{1}{t})$. This convergence rate should be compared with the convergence rate of $\mathcal{O}(\frac{1}{\sqrt{t}})$ for general non-convex (non-concave) functions (Reddi et al., 2016) that applies to many other clustering objectives such *Ratio Assoc*, *Normalized Cut* and *Dominant Set Clustering*, i.e., optimizing *Shifted Min Cut* yields a faster theoretical convergence rate compared to many other alternatives.

6 Shift analysis of other clustering methods

In this section, we investigate the impact of shifting the pairwise similarities on some common flat and hierarchical clustering methods.

Shift of pairwise similarities for flat clustering It is obvious that *K*-means and Gaussian Mixture Models (GMMs) are invariant with respect to the shift of data features. Since these methods perform directly on the data features, shifting refers to adding constant α to all the features. Under this shift, the centroids (in *K*-means) and the means (in GMM) are shifted by α as well, but their proportional distances stay the same. The other parameters, i.e., the clustering assignments (in *K*-means), and the assignment probabilities, covariance matrices and weights (in GMM) do not change. One might assume that by shift only the location of the clusters is affected without modifying the cluster memberships. A similar argument applies to a density-based clustering method such as DBSCAN (Ester et al., 1996) wherein shifting data features does not modify the clustering solution, except a consistent shift of the geographical locations of the clusters together.

As discussed in Roth et al. (2003), when shifting the pairwise similarities by α, the *Ratio Assoc* and *Ratio Cut* cost functions stay invariant, i.e., their optimal solutions stay the same. By shifting the pairwise similarities by α, the *Ratio Assoc* cost function is written as

$$R^{SRA}(\mathbf{c}, \mathbf{X}, \alpha) = -\sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \frac{\mathbf{X}_{ij} + \alpha}{|\mathbf{O}_k|}$$

$$= -\sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \frac{\mathbf{X}_{ij}}{|\mathbf{O}_k|} - \sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \frac{\alpha}{|\mathbf{O}_k|}$$

$$= -\sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \frac{\mathbf{X}_{ij}}{|\mathbf{O}_k|} - \sum_{k=1}^{K} \frac{\alpha |\mathbf{O}_k|^2}{|\mathbf{O}_k|} \qquad (23)$$

$$= \underbrace{-\sum_{k=1}^{K} \sum_{i,j \in \mathbf{O}_k} \frac{\mathbf{X}_{ij}}{|\mathbf{O}_k|}}_{R^{RA}(\mathbf{c}, \mathbf{X})} - \underbrace{\alpha n}_{constant}.$$

Therefore, the *Ratio Assoc* cost function is *invariant* under shifting the pairwise similarities. Similar to *Ratio Assoc*, the *Shifted Ratio Cut* cost function can be written as

$$R^{SRC}(\mathbf{c}, \mathbf{X}, \alpha) = \sum_{k=1}^{K} \frac{\sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O} \setminus \mathbf{O}_k} \mathbf{X}_{ij} + \alpha}{|\mathbf{O}_k|}$$

$$= \sum_{k=1}^{K} \frac{\sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O} \setminus \mathbf{O}_k} \mathbf{X}_{ij}}{|\mathbf{O}_k|} + \sum_{k=1}^{K} \frac{\sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O} \setminus \mathbf{O}_k} \alpha}{|\mathbf{O}_k|}$$

$$= \sum_{k=1}^{K} \frac{\sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O} \setminus \mathbf{O}_k} \mathbf{X}_{ij}}{|\mathbf{O}_k|} + \sum_{k=1}^{K} \frac{\alpha |\mathbf{O}_k|(n - |\mathbf{O}_k|)}{|\mathbf{O}_k|} \qquad (24)$$

$$= \underbrace{\sum_{k=1}^{K} \frac{\sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O} \setminus \mathbf{O}_k} \mathbf{X}_{ij}}{|\mathbf{O}_k|}}_{R^{RC}(\mathbf{c}, \mathbf{X})} + \underbrace{\alpha n(K-1)}_{constant}.$$

Thereby, both *Ratio Assoc* and *Ratio Cut* cost functions are *invariant* under shifting the pairwise similarities. One can show that this holds in general for every clustering cost function that normalizes the clusters by the size of the clusters, i.e., size-normalized (divided) clustering cost functions stay *invariant* with respect to the shift of pairwise similarities.

On the other hand, the *Normalized Cut* cost function when the pairwise similarities are shifted is written by

$$R^{SNC}(\mathbf{c}, \mathbf{X}, \alpha) = \sum_{k=1}^{K} \frac{\sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O} \setminus \mathbf{O}_k} \mathbf{X}_{ij} + \alpha}{\sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}} \mathbf{X}_{ij} + \alpha}. \qquad (25)$$

It turns out that this cost function *is not* shift invariant in general, contrary to the two previous alternatives. However, for the special case of almost balanced clusters, i.e., $|\mathbf{O}_k| \approx n/K, \ \forall 1 \le k \le K,$[4] and similar intra-cluster similarity distribution among all clusters, all the row-sums of the similarity matrix $\mathbf{X}$ tend to be close to each other. The objects then share the same degree, i.e., $\sum_{j=1}^{n} \mathbf{X}_{ij} \approx constant$. In this case, the *Normalized Cut* cost

---

[4] Similar to *Shifted Min Cut*, n/K might not be an integer number. Then, we consider $\lceil n/K \rceil$ and $\lfloor n/K \rfloor$ instead of n/K.

functions becomes equivalent to the *Ratio Assoc* cost function (Roth et al., 2003). This analysis explains the similar performance of such graph partitioning methods in large-scale comparison studies, e.g., for image segmentation, where clusters have balanced and similar structures (Soundararajan & Sarkar, 2001; Roth et al., 2003).

Ratio Cut, despite normalizing the cut by the size of clusters, intends to separate small clusters, as demonstrated in Shi and Malik (2000), Chehreghani (2013). For this reason, *Normalized Cut* has proposed to normalize the cut by the degree of the clusters, rather than the size of the clusters. An alternative way to overcome this problem is to apply a stronger constraint on the size of the clusters. Using this idea, *P-Spectral Clustering* (Bühler & Hein, 2009) proposes a nonlinear generalization of spectral clustering based on the second eigenvector of the graph p-Laplacian which is then interpreted as a generalization of graph clustering models such as *Ratio Cut*. *P-Spectral Clustering* is an iterative clustering procedure that at each step performs a bi-partitioning of one of the existing clusters until K clusters are constructed using a nonlinear spectral method. The underlying respective cost function for bi-partitioning into two sets $\mathbf{O}_a$ and $\mathbf{O}_b$ is given by ($p > 1$)

$$R^{PSC}(c, \mathbf{X}) = \sum_{i \in \mathbf{O}_a} \sum_{j \in \mathbf{O}_b} \mathbf{X}_{ij} \left(\frac{1}{|\mathbf{O}_a|^{\frac{1}{p-1}}} + \frac{1}{|\mathbf{O}_b|^{\frac{1}{p-1}}} \right)^{p-1}. \tag{26}$$

In Chehreghani (2013), we have introduced *Adaptive Ratio Cut* (ARC) as a generalization of the cost function to yield K clusters:

$$R^{ARC}(c, \mathbf{X}) = \sum_{k=1}^{K} \sum_{k'=k+1}^{K} \sum_{i \in \mathbf{O}_k} \sum_{j \in \mathbf{O}_{k'}} \mathbf{X}_{ij} \left(\frac{1}{|\mathbf{O}_k|^{\frac{1}{p-1}}} + \frac{1}{|\mathbf{O}_{k'}|^{\frac{1}{p-1}}} \right)^{p-1}. \tag{27}$$

For the special case of $p = 2$, *Adaptive Ratio Cut* is equivalent to the standard *Ratio Cut* cost function. However, unlike *Ratio Cut*, it is easy to see that *Adaptive Ratio Cut* is not shift invariant, as the shift parameter α cannot be factored out from the cost function.

Shifted Dominant Set Clustering. This clustering method computes the clusters via performing replicator dynamics. It has been shown that the solutions of a replicator dynamics correspond to the solutions of the following quadratic program (Schuster & Sigmund, 1983; Weibull, 1997).

$$\max_{\mathbf{v}} f(\mathbf{v}) = \mathbf{v}^{\mathsf{T}} \mathbf{X} \mathbf{v}, \quad \text{s.t. } \mathbf{v} \geq \mathbf{0}, \sum_{i=1}^{n} \mathbf{v}_i = 1, \tag{28}$$

where the n-dimensional *characteristic vector* $\mathbf{v}$ determines the participation of the objects to the solution.

Thus, to study the impact of the shift on DSC, we consider the shifted variant of the quadratic program. In Chehreghani (2016) we have elaborated the impact of such a shift based on the off-diagonal shift argument in Pavan and Pelillo (2003). It yields

$$f(\mathbf{v}, \alpha) = \mathbf{v}^\mathrm{T}(\mathbf{X} + \alpha\, \mathbf{e}\mathbf{e}^\mathrm{T})\mathbf{v}$$
$$= \mathbf{v}^\mathrm{T}\mathbf{X}\mathbf{v} + \mathbf{v}^\mathrm{T}\alpha\, \mathbf{e}\mathbf{e}^\mathrm{T}\mathbf{v}$$
$$= \mathbf{v}^\mathrm{T}\mathbf{X}\mathbf{v} + \alpha\, \underbrace{(\mathbf{v}^\mathrm{T}\mathbf{e})}_{=1}\, \underbrace{(\mathbf{e}^\mathrm{T}\mathbf{v})}_{=1} \tag{29}$$
$$= \mathbf{v}^\mathrm{T}\mathbf{X}\mathbf{v} + \alpha\, ,$$

where $\mathbf{e} = (1, 1, ...1)^\mathrm{T}$ is a vector of ones.

Therefore, *Dominant Set Clustering* is *invariant* under shifting the pairwise similarities.

However, it has been proposed in Pavan and Pelillo (2003) to shift the diagonal entries of the similarity matrix by a negative value, in order to obtain coarser clusters, which yields computing a hierarchy of clusters. The clusters obtained from the unshifted similarity matrix appear at the lowest level of the hierarchy. The larger the negative shift is the coarser the clusters are. Performing a negative shift is equivalent to adding the same shift but with a positive sign to the off-diagonal pairwise similarities. Thereby, the shifted matrix is still non-negative and has a null diagonal, i.e. satisfies the conditions of *Dominant Set Clustering*.

One can think of performing a negative shift on the off-diagonal pairwise similarities to compute a finer representation of the clusters. However, this type of shift might violate the non-negativity and null diagonal constraints. On the other hand, according to our experiments, a negative shift is effectively equivalent to applying a larger cut-off threshold when peeling off the clusters. In Chehreghani (2016) we have proposed such a shift to accelerate the appearance of clusters for DSC.

Shift of pairwise similarities for hierarchical clustering. Hierarchical clustering methods, unlike flat clustering, produce clusters at multiple levels. A main category of such methods first consider each object in a separate cluster, and then at each step, combine the two clusters with a *minimal* distance according to some criterion until only one cluster is left at the highest level.

A cluster at an arbitrary level is represented by a set of objects belong to that, e.g., by $\mathbf{u}$ or $\mathbf{v}$. A hierarchical clustering solution can be represented by a dendrogram (tree) T such that,

(i) each node $\mathbf{v}$ in T consists of a non-empty subset of the objects that belong to cluster $\mathbf{v}$, and (ii) the overlapping clusters have a parent-child relation, i.e., one is the (grand) parent of the other.

We use $dist(\mathbf{u}, \mathbf{v})$ to refer to the inter-cluster distance between clusters $\mathbf{u}$ and $\mathbf{v}$. It can be defined according to different criteria. Three common criteria for hierarchical clustering are *single* linkage, *complete* linkage and *average* linkage. Given the matrix of (inter-object) pairwise dissimilarities $\mathbf{D} = \{\mathbf{D}_{ij}\}, i,j \in \mathbf{O}$, the *single* linkage criterion (Sneath, 1957) defines the distance between every two clusters as the distance between their nearest members:

$$dist(\mathbf{u}, \mathbf{v}) = \min_{i \in \mathbf{u}, j \in \mathbf{v}} \mathbf{D}_{ij}\, . \tag{30}$$

On the other hand, *complete* linkage (Lance & Williams, 1967) considers the distance between their farthest members:

$$dist(\mathbf{u}, \mathbf{v}) = \max_{i \in \mathbf{u}, j \in \mathbf{v}} \mathbf{D}_{ij}\, . \tag{31}$$

Finally, *average* linkage (Sokal & Michener, [1958]) uses the average of the inter-cluster distances as the distance between the two clusters:

$$dist(\mathbf{u}, \mathbf{v}) = \sum_{i \in \mathbf{u}, j \in \mathbf{v}} \frac{\mathbf{D}_{ij}}{|\mathbf{u}||\mathbf{v}|} \ . \tag{32}$$

In the following we show that these methods, which perform based on pairwise inter-cluster distances, are shift-invariant (Proposition [1]).

Proposition 1 *Single linkage, complete linkage and average linkage methods are invariant with respect to the shift of the pairwise dissimilarities* $\mathbf{D}$ *by constant* α.

Proof Let us show the shifted pairwise dissimilarities by $\mathbf{D}^\alpha$, i.e., $\mathbf{D}_{ij}^\alpha = \mathbf{D}_{ij} + \alpha, \forall i, j \in \mathbf{O}$.

- With shifting all the pairwise dissimilarities by α, the $dist(\mathbf{u}, \mathbf{v})$ function for *single* linkage is defined as

$$dist(\mathbf{u}, \mathbf{v}) = \min_{i \in \mathbf{u}, j \in \mathbf{v}} \mathbf{D}_{ij}^\alpha = \min_{i \in \mathbf{u}, j \in \mathbf{v}} \mathbf{D}_{ij} + \alpha. \tag{33}$$

  Thus, if $dist(\mathbf{u}, \mathbf{v}) \leq dist(\mathbf{u}, \mathbf{w})$ holds with respect to $\mathbf{D}$, then it would also hold with respect to $\mathbf{D}^\alpha$ and vice versa, as they differ only by a constant in both sides of the inequality. Thus, shifting the pairwise dissimilarities by α does not change the order of merging the intermediate clusters and hence the final dendrogram will remain the same.
- With shifting all the pairwise dissimilarities by α, the $dist(\mathbf{u}, \mathbf{v})$ function for *complete* linkage is defined as

$$dist(\mathbf{u}, \mathbf{v}) = \max_{i \in \mathbf{u}, j \in \mathbf{v}} \mathbf{D}_{ij}^\alpha = \max_{i \in \mathbf{u}, j \in \mathbf{v}} \mathbf{D}_{ij} + \alpha. \tag{34}$$

  Thus, with the same argument as with *single* linkage, shifting the pairwise dissimilarities by α does not change the final *complete* linkage dendrogram.
- With shifting all the pairwise dissimilarities by α, the $dist(\mathbf{u}, \mathbf{v})$ function in *average* linkage is defined as

$$\begin{aligned} dist(\mathbf{u}, \mathbf{v}) &= \sum_{i \in \mathbf{u}, j \in \mathbf{v}} \frac{\mathbf{D}_{ij}^\alpha}{|\mathbf{u}||\mathbf{v}|} \\ &= \sum_{i \in \mathbf{u}, j \in \mathbf{v}} \frac{\mathbf{D}_{ij} + \alpha}{|\mathbf{u}||\mathbf{v}|} \\ &= \left(\sum_{i \in \mathbf{u}, j \in \mathbf{v}} \frac{\mathbf{D}_{ij}}{|\mathbf{u}||\mathbf{v}|} \right) + \alpha. \end{aligned} \tag{35}$$

  Thus, we use the same argument as in with *single* linkage and *complete* linkage, and conclude that shifting the pairwise dissimilarities by α does not change the final *average* linkage dendrogram.

$\square$

Another category of hierarchical clustering methods such as *centroid* linkage and *Ward* linkage perform directly on data features, instead of pairwise dissimilarities. *Centroid* linkage computes a representative for each cluster and defines the inter-cluster

distances according to those representatives. Similar to the case of *K*-means, shifting the data features by a constant does not change the pairwise inter-cluster distances. The *Ward* linkage (Ward, 1963) aims at minimizing the within-cluster variance at each step, i.e., the *dist*(**u**, **v**) is defined as

$$dist(\mathbf{u}, \mathbf{v}) = \frac{|\mathbf{u}||\mathbf{v}|}{|\mathbf{u}| + |\mathbf{v}|}||\mathbf{g_u} - \mathbf{g_v}||^2, \tag{36}$$

where $\mathbf{g_u}$ denotes the centroid vector of cluster **u**. Therefore, due to shift invariance of variance, the *Ward* linkage is also invariant with respect to the shift of data features. Thereby, we can state Proposition 2 as following.

Proposition 2 *Centroid linkage and Ward linkage are invariant with respect to the shift of data features.*

Finally, it is notable that some of the improvements proposed for hierarchical clustering still preserve the invariance property with respect to the shift of pairwise distances. For example, in order to improve the robustness of hierarchical clustering, it is suggested in Chehreghani et al. (2008) to first apply *K*-means with many centroids (of order of *n*) and then apply the aforementioned hierarchical methods. Since both steps, i.e., *K*-means clustering and hierarchical clustering, are invariant with respect to the shift, thus one can conclude that the entire procedure remains invariant as well. The work in Chehreghani (2021) studies extracting all mutual linkages at every step of hierarchical clustering, instead of the smallest one, in order to provide adaptivity to diverse shapes of clusters. Since this contribution is independent of the way the inter-cluster distances are defined, then this strategy yields invariant clustering with respect to the shift of pairwise distances for methods such as *single* linkage, *complete* linkage and *average* linkage.

7 Experiments

We empirically investigate the performance of *Shifted Min Cut* and compare the results against several alternatives. We perform the experiments under identical computational settings on a core i7-4600U Intel machine with 2.7 GHz CPU and 8.00 GB internal memory.

Data We first perform our experiments on several UCI datasets (Lichman, 2013), chosen from different domains and contexts with different type of features.

1. *Breast Tissue* contains 106 electrical impedance measurements of the breast tissue samples in 6 types (clusters) each with 10 features. The types or clusters are 'car' (carcinoma, 21 measurements), 'fad' (fibro-adenoma, 15 measurements), 'mas' (mastopathy 18 measurements), 'gla' (glandular, 16 measurements), 'con' (connective, 12 measurements) and 'adi' (adipose 22 measurements). The features are real valued with no missing value.
2. *Cloud* consists of 2048 vectors, where each vector includes 10 parameters in two types (each of size 1024) representing AVHRR images. The vectors (attributes) are real-valued and there are no missing values. The target clusters are balanced.
3. *Ecoli* a biological dataset on the cellular localization sites of 7 types (clusters) of proteins which includes 336 samples. The samples are represented by 8 real-valued features. The size of the clusters are: 143, 77, 3, 7, 35, 20 and 52,

4. *Forest Type Mapping* a remote sensing dataset of 523 samples with 27 real-valued attributes collected from forests in Japan and grouped in 4 different forest types (clusters). The clusters are: 's' ('Sugi' forest, 159 samples), 'h' ('Hinoki' forest, 86 samples), 'd' ('Mixed deciduous' forest, 195 samples), 'o' ('Other' non-forest land, 83 samples).

5. *Heart* dataset of heart disease that involves 303 instances each with 75 attributes. The attributes are diverse: categorical, integer and real where the categorical attributes are treated using one-hot encoding. The missing values are estimated by the median of the respective feature. Cluster distributions are: 164, 55, 36, 35 and 13.

6. *Lung Cancer* high-dimensional lung cancer data with 32 instances (with distribution 9 and 23) and 56 integer features. There are few missing values estimated using the median of the respective feature.

7. *Parkinsons* contains 197 biomedical voice measurements from 31 people each represented by 23 real-valued attributes that correspond to voice recordings. In the dataset, there are 48 healthy samples and 147 other samples that belong to one of 23 people with Parkinson's disease.

8. *Pima Indians Diabetes* the data of 768 female patients from Pima Indian heritage with 8 attributes. The attributes include the number of pregnancies of the patient, their BMI, insulin level, age, and so on, and they are either real numbers or integers. 268 samples out of 768 haze the outcome 1 and the others (500 samples) have the outcome 0.

9. *SPECTF* describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images with 44 integer attributes (values from the 0 to 100) about the heart of 267 patients. The diagnosis is binary with the distribution of 55 and 212 samples.

10. *Statlog ACA (Australian Credit Approval)* contains information of 690 credit card applications each described with 14 features (with cluster size 383 and 307). The features are categorical and numerical where for categorical features we use one-hot encoding. The few missing values are estimated using the median of the respective feature.

11. *Teaching Assistant* consists of evaluations of teaching performance over 5 semesters of 151 teaching assistant assignments. The scores are divided into 3 roughly equal-sized categories ('low', 'medium' and 'high') to form the target variables which are used as the cluster labels. The attributes are categorical and integer, where we use one-hot encoding for categorical attributes. There are no missing values.

12. *User Knowledge Modeling* contains the 403 students' knowledge status on Electrical DC Machines with 5 integer attributes grouped in 4 categories. The labels and the cluster distributions are: 'very Low': 50, 'low': 129, 'middle': 122 and 'high': 130. There are no missing values.

In these datasets, the objects are represented by vectors. Thus, to obtain the pairwise similarity matrix $\mathbf{X}$, we first compute the pairwise squared Euclidean distances between the vectors and obtain matrix $\mathbf{D}$. Then, as proposed in Chehreghani ([2016](#)), we convert the pairwise distances $\mathbf{D}$ to the similarity matrix $\mathbf{X}$ via $\mathbf{X}_{ij} = \max(\mathbf{D}) - \mathbf{D}_{ij} + \min(\mathbf{D})$, where the max(.) and min(.) operations respectively give the maximum and the minimum of the elements in $\mathbf{D}$. An alternative transformation is an exponential function in the form of $\mathbf{S}_{ij} = \exp(-\frac{\mathbf{X}_{ij}}{\sigma^2})$, which requires fixing the free parameter σ in advance. However, this task is nontrivial in unsupervised learning and the appropriate values of σ coincide in a very narrow range (Luxburg, [2007](#)). The other alternative is the cosine

similarity, which suits better to textual and document datasets. On our datasets, we consistently obtain better results with the aforementioned transformation.

Methods We compare *Shifted Min Cut* against several alternative methods developed for clustering. We consider the following methods: (i) *Dominant Set Clustering* (DSC), (ii) *InImDyn*, (iii) *P-Spectral Clustering* (PSC), (iv) *Gaussian Mixture Model* (GMM), (v) *K-means*, (vi) *Power Iteration Clustering* (PIC), and vii) *Spectral Clustering* (SC).

The chosen baselines belong to different clustering approaches which cover a wide range of alternative viewpoints for clustering, e.g., those based on a cost function, probabilistic methods, game-theoretic methods and spectral methods. With the GMM method, we obtain the probabilistic assignment of the objects to the clusters. Then, we assign each object to the most probable cluster. The developed clustering perspective can potentially be combined with the recent developments proposed in particular for cost-based clustering methods. For example, a category of recent clustering methods aim to combine deep representation learning methods with clustering (Demetriou et al., 2020; Yang et al., 2019), or develop approximate and distributed clustering methods. Such contributions are orthogonal to our contribution and, in principle, can be combined with *Shifted Min Cut* as well. On the other hand, considering the relation between *Shifted Min Cut* and *Correlation Clustering*, with the co-authors, we have recently (Thiel et al., 2019) studied the performance of the local search optimization compared to a wide range of approximate methods developed for *Correlation Clustering* and have demonstrated both efficiency and effectiveness for the local search method.

Evaluation criteria We have access to the ground truth solutions for the datasets. These labels may play the role of an expert (reference) that tells us the desired clustering solution. Thus, we can use them to evaluate the results of different methods. We note that we do not employ them to infer the clustering solution, they are only used for evaluation. Therefore, we are still in unsupervised setting which assumes no data label is used to obtain the results. This evaluation procedure is recommended in Manning (2008) consistent with several studies, e.g., (Dhillon et al., 2004; Lin & Cohen, 2010; Liu et al., 2013; Thiel et al., 2019; Yang et al., 2019). Thereby, we compare the true (given) clustering labels and the estimated solutions to investigate quantitatively the performance of each method. For this purpose, we consider three criteria:

1. Adjusted Mutual Information (Vinh et al., 2010): the mutual information between the two estimated and true clustering solutions,
2. Adjusted Rand score (Hubert & Arabie, 1985): the similarity between the solutions, and
3. V-measure (Rosenberg & Hirschberg, 2007): the harmonic mean of homogeneity and completeness.

We compute the adjusted variant of these criteria such that they give zero scores for random solutions.

Results We study the performance of different methods from two perspectives in order to distinguish between the quality of a method/costs function and its optimization. The former implies how good a particular method/cost function is (given that it can be optimized in a proper way) while the later focuses on the optimization aspects of the method/cost function. We run each method 100 times with different random initializations. In the first type of study, we choose the best solution in terms of the cost or likelihood among the 100 different runs for each method. We note that we do not choose the best results in terms of

Table 1 Performance of different methods with respect to the adjusted Mutual Information criterion

Dataset	ShiftedMinCut	DSC	InImDyn	PSC	GMM	*K*-means	PIC	SP
Breast tissue	0.4196	0.4305	0.4196	0.3606	0.3276	0.1809	0.4123	**0.4507**
Cloud	**1.0000**	0.3812	0.3543	0.3098	0.8511	0.3056	0.8597	0.8406
Ecoli	0.5414	0.4731	0.4368	0.5074	**0.5800**	0.5685	0.0542	0.4743
Forest type	0.4352	0.2960	0.3109	0.2704	0.3877	**0.5197**	0.3875	0.3163
Heart	**0.1570**	0.0698	0.0602	0.0594	0.0813	0.0813	0.0509	0.1078
Lung cancer	0.2362	0.0850	0.0859	0.0713	0.1684	0.1997	0.0380	**0.2473**
Parkinsons	**0.1957**	0.0738	0.0761	0.0511	0.0484	0.0136	0.0153	0.1631
Pima Indians diabetes	0.1178	0.0561	0.0533	0.0368	0.0003	0.0257	**0.1226**	0.1200
SPECTF	**0.1570**	0.0698	0.0602	0.0419	0.0813	0.0813	0.0509	0.1078
Statlog ACA	**0.3907**	0.1607	0.1498	0.1392	0.0038	0.0038	0.3570	0.3683
Teaching assistant	**0.1041**	0.0268	0.0357	0.0123	0.0353	0.0130	0.0470	0.0123
User knowledge modeling	0.2926	0.1107	0.1198	0.0441	**0.6100**	0.2139	0.2454	0.1169

The best result among different methods are shown in bold

Shifted Min Cut yields the best results in most of the cases

Table 2 Performance of different methods with respect to the adjusted Rand score

Dataset	ShiftedMinCut	DSC	InImDyn	PSC	GMM	*K*-means	PIC	SP
Breast tissue	**0.3546**	0.2929	0.2907	0.3100	0.2085	0.0943	0.3125	0.3037
Cloud	**1.0000**	0.3573	0.3117	0.2926	0.8991	0.2429	0.9065	0.8899
Ecoli	**0.6801**	0.4068	0.3299	0.5145	0.5574	0.4944	0.0378	0.4132
Forest type	0.3983	0.2225	0.2426	0.2027	0.3285	**0.4987**	0.3560	0.2454
Heart	**0.0917**	0.0608	0.0449	0.0578	0.0467	0.0337	0.0721	0.0588
Lung cancer	0.2962	0.0689	0.0664	0.0215	0.1698	0.2294	0.0949	**0.4201**
Parkinsons	**0.1275**	0.0129	0.0360	0.0234	0.0123	0.0162	0.0178	0.0419
Pima Indians diabetes	0.1535	0.0454	0.0460	0.0381	0.0010	0.0744	**0.1617**	0.1200
SPECTF	**0.0917**	0.0608	0.0429	0.0570	0.0617	0.0617	0.0434	0.0898
Statlog ACA	**0.4913**	0.1485	0.1307	0.1332	0.0022	0.0022	0.4550	0.4710
Teaching assistant	**0.1170**	0.0112	0.0127	0.0129	0.0220	0.0089	0.0322	0.0129
User knowledge modeling	0.2912	0.0713	0.0778	0.0503	**0.5680**	0.1675	0.1449	0.1053

The best result among different methods are shown in bold

Shifted Min Cut leads to better clustering solutions on most of the datasets

the evaluation criteria. This helps to gain a sense that the optimization is done properly, and we may not suffer from very poor local optima, and therefore, we can investigate the performance of the method or the cost function regardless of its optimization.

Tables 1, 2 and 3 show the results of the first type of study for different clustering methods on the UCI datasets respectively with respect to the Mutual Information criterion, the Rand score and the V-measure. We observe that on most of the datasets, *Shifted Min Cut* yields the best scores. In the cases that the method is not the best, it is usually among top

Table 3 Performance of different methods with respect to the adjusted V-measure

Dataset	ShiftedMinCut	DSC	InImDyn	PSC	GMM	K-means	PIC	SP
Breast tissue	**0.5563**	0.4914	0.4999	0.4756	0.4389	0.2895	0.5230	0.5142
Cloud	**1.0000**	0.5525	0.5239	0.5116	0.8520	0.3388	0.8605	0.8417
Ecoli	**0.6396**	0.5339	0.5169	0.5203	0.6192	0.6321	0.1139	0.5364
Forest type	0.4835	0.3591	0.3866	0.2729	0.3937	**0.5279**	0.3982	0.3244
Heart	**0.1788**	0.1186	0.1061	0.0689	0.0896	0.0896	0.0644	0.1131
Lung cancer	0.2730	0.2175	0.2310	0.1038	0.2030	0.2356	0.1117	**0.2987**
Parkinsons	**0.2196**	0.1255	0.1327	0.0814	0.0130	0.0105	0.0291	0.1798
Pima Indians diabetes	0.1227	0.0867	0.0838	0.0693	0.0013	0.0295	**0.1276**	0.1200
SPECTF	**0.1788**	0.1186	0.1061	0.0819	0.0896	0.0896	0.0644	0.1131
Statlog ACA	**0.3927**	0.2396	0.2267	0.2068	0.0099	0.0099	0.3632	0.3720
Teaching assistant	**0.1156**	0.0615	0.0770	0.0253	0.0583	0.0263	0.0606	0.0253
User knowledge modeling	0.3384	0.1527	0.1697	0.0573	**0.6191**	0.2278	0.2603	0.1325

The best result among different methods are shown in bold

In a consistent way to the two previous evaluation criteria, the *Shifted Min Cut* method provides the best clustering results on most of the datasets

Table 4 Average performance (and the standard deviation shown in brackets) for different methods over 100 runs with respect to adjusted Mutual Information, where *Shifted Min Cut* often yields the most promising results

Dataset	ShiftedMinCut	GMM	K-means	PIC	SP
Breast tissue	0.4191(0.0031)	0.3031(0.0365)	0.1794(0.0371)	0.3883(0.0470)	**0.4393(0.0129)**
Cloud	**1.0000(0.0000)**	0.8510(0.0152)	0.3019(0.0141)	0.8588(0.0060)	0.8392(0.0071)
Ecoli	0.5304(0.0119)	**0.5792(0.0374)**	0.5673 (0.0294)	0.0493(0.0177)	0.4728(0.0032)
Forest type	0.4085(0.0416)	0.3542(0.0290)	**0.5169(0.0330)**	0.3881(0.0147)	0.3124(0.0010)
Heart	**0.1562(0.0024)**	0.0817(0.0025)	0.0832(0.0019)	0.0528(0.0133)	0.0820(0.0304)
Lung cancer	0.2174(0.0684)	0.1678(0.0719)	0.1720(0.0455)	0.0460(0.0397)	**0.2389(0.0250)**
Parkinsons	**0.1932(0.0046)**	0.0380(0.0241)	0.0256(0.0305)	0.0169(0.0134)	0.1443(0.0251)
Pima Indians diabetes	0.0972(0.0329)	0.0003(0.0008)	0.0285(0.0020)	**0.1233(0.0042)**	0.1186(0.0049)
SPECTF	**0.1561(0.0026)**	0.0821(0.0020)	0.0806(0.0031)	0.0503(0.0155)	0.1019(0.0229)
Statlog ACA	**0.3842(0.0279)**	0.0116(0.0031)	0.0087(0.0033)	0.3521(0.0045)	0.3648(0.0037)
Teaching assistant	**0.0934(0.0380)**	0.0346(0.0123)	0.0045(0.0085)	0.0412(0.0124)	0.0106(0.0073)
User knowledge modeling	0.2735(0.0601)	**0.5536(0.0962)**	0.2026(0.0648)	0.2470(0.0131)	0.1043(0.0239)

The best result among different methods are shown in bold

choices. DSC and InImDyn perform very similarly, consistent to the results in Bulò et al. (2011). PIC works well only when there are few clusters in the dataset. The reason is that it computes an one-dimensional embedding of the data and then applies *K*-means. However, such an embedding might confuse some clusters when there exist many of them in the dataset (Chehreghani, 2016). PSC is significantly slower than the other methods and also

Table 5 Average performance (and the standard deviation) for different methods with respect to adjusted Rand score

Dataset	ShiftedMinCut	GMM	K-means	PIC	SP
Breast tissue	**0.3543(0.0051)**	0.1756(0.0354)	0.1071(0.0361)	0.2642(0.0697)	0.3030(0.0311)
Cloud	**1.0000(0.0000)**	0.8636(0.0309)	0.2419(0.0015)	0.9067(0.0018)	0.8865(0.0029)
Ecoli	**0.6826(0.0113)**	0.5303(0.0611)	0.4598(0.0686)	0.0193(0.0175)	0.3939(0.0098)
Forest type	0.3894(0.0267)	0.2792(0.0427)	**0.4880(0.0360)**	0.3429(0.0243)	0.2388(0.0009)
Heart	**0.0885(0.0041)**	0.0471(0.0022)	0.0259(0.0029)	0.0518(0.0212)	0.0498(0.0113)
Lung cancer	0.2546(0.0798)	0.1594(0.0949)	0.2079(0.0705)	0.0844(0.0297)	**0.4172(0.0133)**
Parkinsons	**0.1268(0.0079)**	0.0282(0.0286)	0.0217(0.0389)	0.0159(0.0121)	0.0387(0.0081)
Pima Indians diabetes	0.1317(0.0329)	0.0010(0.0003)	0.0736(0.0027)	**0.1593(0.0028)**	0.1135(0.0058)
SPECTF	**0.0903(0.0036)**	0.0598(0.0033)	0.0608(0.0042)	0.0516(0.0248)	0.0894(0.057)
Statlog ACA	**0.4825(0.0485)**	0.0031(0.0007)	0.0030(0.0008)	0.4525(0.0042)	0.4590(0.0162)
Teaching assistant	**0.0883(0.0421)**	0.0248(0.0061)	0.0024(0.0069)	0.0248(0.0115)	0.0201(0.0097)
User knowledge modeling	0.2729(0.0520)	**0.4972(0.0788)**	0.1542(0.0377)	0.1495(0.0121)	0.1046(0.0174)

The best result among different methods are shown in bold

Consistent with adjusted Mutual Information, *Shifted Min Cut* yields the best results on most of the datasets

Table 6 Average performance (and the standard deviation) for different methods with respect to adjusted V-measure

Dataset	ShiftedMinCut	GMM	K-means	PIC	SP
Breast tissue	**0.5558(0.0023)**	0.4015(0.0326)	0.2711(0.0329)	0.5193(0.0190)	0.5113(0.0174)
Cloud	**1.0000(0.0000)**	0.8329(0.0137)	0.3402(0.0011)	0.8602(0.0011)	0.8395(0.0028)
Ecoli	**0.6380(0.0149)**	0.6029(0.0355)	0.6007(0.0432)	0.0884(0.0372)	0.5296(0.0031)
Forest type	0.4712(0.0402)	0.3835(0.0178)	**0.5211(0.0327)**	0.3936(0.0147)	0.3169(0.0017)
Heart	**0.1761(0.0037)**	0.0841(0.0053)	0.0864(0.0019)	0.0608(0.0142)	0.1066(0.0036)
Lung cancer	0.2578(0.0667)	0.2004(0.0559)	0.2194(0.0623)	0.1094(0.0350)	**0.2858(0.0096)**
Parkinsons	**0.2151(0.0036)**	0.0134(0.0102)	0.0152(0.0096)	0.0216(0.0142)	0.1506(0.0243)
Pima Indians diabetes	0.1026(0.0328)	0.0011(0.0006)	0.0217(0.0058)	**0.1242(0.0042)**	0.1205(0.0016)
SPECTF	**0.1729(0.0040)**	0.0850(0.0028)	0.0891(0.0015)	0.0592(0.0058)	0.0974(0.0108)
Statlog ACA	**0.3845(0.0463)**	0.0040(0.0030)	0.0033(0.0051)	0.3627(0.0015)	0.3655(0.0147)
Teaching assistant	**0.1062(0.0375)**	0.0506(0.0123)	0.0170(0.0086)	0.0535(0.0110)	0.0284(0.0072)
User knowledge modeling	0.3027(0.0592)	**0.5834(0.0674)**	0.2250(0.0540)	0.2612(0.0131)	0.1257(0.0236)

The best result among different methods are shown in bold

We observe that the average results are consistent with the results from the first type of study and *Shifted Min Cut* performs well compared to the alternatives

yields suboptimal results, as reported by several previous studies as well. Other methods are efficient and perform within few seconds.

Table 7 Performance of different methods on DS1

	ShiftedMinCut	DSC	InImDyn	PSC	GMM	*K*-means	PIC	SP
Mutual Information	0.6046	0.2951	0.2786	0.1572	**0.6124**	0.5880	0.5994	0.3525
Rand score	**0.4632**	0.2101	0.2067	0.0762	0.2945	0.2434	0.2640	0.1384
V-measure	**0.8107**	0.4884	0.4655	0.3119	0.7996	0.7854	0.7733	0.5874

The best result among different methods are shown in bold

On this dataset, *Shifted Min Cut* yields superior results compared to the alternatives

Table 8 Performance of different methods on DS2 where *Shifted Min Cut* leads to the best overall performance

	ShiftedMinCut	DSC	InImDyn	PSC	GMM	*K*-means	PIC	SP
Mutual Information	**0.5621**	0.3011	0.3180	0.3572	0.4793	0.5103	0.4272	0.5475
Rand score	0.5316	0.2780	0.2621	0.2752	0.4648	0.4512	0.4484	**0.5528**
V-measure	**0.6179**	0.3126	0.3441	0.3270	0.5245	0.5523	0.5164	0.5715

The best result among different methods are shown in bold

Table 9 Average performance (and the standard deviation shown in brackets) for different methods over 100 runs with respect to different evaluation criteria on DS1

	ShiftedMinCut	GMM	*K*-means	PIC	SP
Mutual Information	0.5785(0.0352)	**0.5922(0.0294)**	0.5773(0.0467)	0.5367(0.0386)	0.3493(0.0460)
Rand score	**0.4239(0.0506)**	0.2872(0.0377)	0.2278(0.0491)	0.2614(0.0195)	0.1146(0.0207)
V-measure	**0.7914(0.0345)**	0.7806(0.0272)	0.7590(0.0338)	0.7568(0.0248)	0.5506(0.0512)

The best result among different methods are shown in bold

Table 10 Average performance (and the standard deviation) for different methods over 100 runs with respect to different evaluation criteria on DS2

	ShiftedMinCut	GMM	*K*-means	PIC	SP
Mutual Information	**0.5617(0.0218)**	0.4427(0.0465)	0.4519(0.0421)	0.4185(0.0307)	0.5118(0.0535)
Rand score	0.5185(0.0266)	0.4502(0.0379)	0.4327(0.0289)	0.4151(0.0342)	**0.5424(0.0372)**
V-measure	**0.5891(0.0335)**	0.5081(0.0507)	0.5122(0.0410)	0.4870(0.0441)	0.5385(0.0463)

The best result among different methods are shown in bold

On both DS1 and DS2, *Shifted Min Cut* yields more promising results in overall

In the second type of study, in order to investigate the optimization itself, we report the average scores and the respective standard deviations over the 100 different runs for each method. We note that DSC, InImDyn and PSC are non-randomized algorithmic procedures that do not show randomness in the performance and their results are stable among different runs. Therefore, we do not need to report their results here. Tables 4, 5 and 6 show such optimization variability results for different UCI datasets (i.e., the average results and the respective standard deviations shown in brackets). We observe that the results are consistent among different runs and the better methods in Tables 1, 2 and 3 perform well on average

too, i.e., the results from the first type of study and the second type of study are consistent in overall. In particular, *Shifted Min Cut* yields the most promising results in this type of study as well. The results also confirm the effectiveness of the optimization based on local search, a method that is nowadays used widely in different machine learning paradigms.

Experiments on real-world data In the following, we investigate the performance of different clustering methods on two real-world datasets:

1. *DS1* This dataset, collected by a document processing company, contains the vectors of 675 scanned documents, wherein each document is represented in a 4096 dimensional space using different textual, image, structural and other features. The documents are placed within 56 clusters with different sizes, that makes the clustering task challenging. The size of the clusters varies from few documents to more than 200 documents. The features are real-valued.

2. *DS2* In this dataset, we collect articles about 5 different Computer Science subjects: 'artificial intelligence', 'software', 'hardware', 'networks' and 'algorithms'. For each category, we collect 1500 articles, thus in total there are 7500 articles in this dataset. We computer the *tf-idf* vectors for each article, thus the attributes are numerical. There are no missing values.

Similar to the experiments on the UCI datasets, we first study the performance of the methods when the optimization is performed properly, i.e., when we pick the best results in terms of the cost function or the likelihood over 100 different runs. Tables 7 and 8 show the performance of different clustering methods with respect to the evaluation criteria on DS1 and DS2. We observe that only *Shifted Min Cut* yields high scores with respect to all criteria. In most of the cases, *Shifted Min Cut* results in the best scores. Otherwise, it is still competitive compared to the best choice.

Finally, we study the optimization variability, i.e., the average results and the respective standard deviations among the 100 runs. The results with respect to different evaluation criteria are shown in Tables 9 and 10 corresponding to DS1 and DS2. Similar to the experiments on the UCI datasets, we observe that the optimization variability results follow the same trend as the results in Tables 7 and 8. This indicates that the average results are consistent with the results obtained based on the best values of the cost function or the likelihood. On the other hand, *Shifted Min Cut* yields the most promising results either in average or when choosing the best solutions in terms of cost/likelihood.

8 Conclusion

This paper investigates an alternative approach for regularizing the *Min Cut* cost function in order to avoid the appearance of singleton clusters, where the regularization term is added to the cost function, instead of dividing the *Min Cut* clusters by a cluster dependent factor. We, in particular, studied the case where the regularization term leads to subtracting the pairwise similarities by the regularization factor. Then, we only need to apply the base *Min Cut*, but on the (adaptively) shifted similarities instead of the original data. In the following, we developed an efficient *local search* algorithm to optimize (locally) the *Shifted Min Cut* cost function and studied its fast theoretical convergence rate. Thereafter,

we discussed that unlike *Min Cut*, many other common clustering cost functions are invariant with respect to the shift of pairwise similarities. Finally, we performed extensive experiments on several UCI and real-world datasets to demonstrate the superior performance of *Shifted Min Cut* according to different evaluation criteria.

Author contributions There is only one author for this publication who has performed all the steps, i.e., developing and implementing the ideas, the experimental studies and the write up.

Data availability Most of the datasets (twelve datasets) are public datasets belonging to the UCI data repository. There are two datasets that are collected by third parties. We are aiming to have an agreement and publish those datasets along with the code (if possible). The datasets do not contain any sensitive information, and all the personal information have been removed.

Declarations

Conflict of interest Not Applicable. No conflict of interest occurs.

Ethics approval This research is mainly focused on conceptual and methodological developments in unsupervised learning and clustering. Clustering is usually used for data management and exploratory data analytics. Thus, this contribution provides methods to further understand, explore and explain the data and obtain deeper insights. Such possibilities can be used for example to understand gender-specific features, data irregularities, private and sensitive information and explainability aspects. On the other hand, the use of clustering for data management and summarization provides a systematic way to compress the data to yield more efficient data precessing in terms of energy and memory usage. This, itself, can be helpful for better environmental conditions. These properties are critical when dealing with large amount of data, in particular for environment friendly solutions. Finally, we would like to emphasize that in this work the experimental studies use the datasets which do not contain any private and sensitive information.

Consent to participate Not Applicable. There is no human study in this research.

Consent for publication Not Applicable. No human study is performed in this research. There is no sensitive information.

Code availability The code will be available through the author's home page and will be maintained there with a reference to this publication.

References

Bailey, K. (1994). *Numerical taxonomy and cluster analysis*. SAGE Publications.
Bansal, N., Blum, A., & Chawla, S. (2004). Correlation clustering. *Machine Learning, 56*(1–3), 89–113.

Bühler, T., & Hein, M. (2009). Spectral clustering based on the graph p-laplacian. In *Proceedings of the 26th annual international conference on machine learning*, ICML '09, pp. 81–88. ACM.

Bulò, S. R., Pelillo, M., & Bomze, I. M. (2011). Graph-based quadratic optimization: A fast evolutionary approach. *Computer Vision and Image Understanding, 115*(7), 984–995.

Cattell, R. B. (1943). The description of personality: Basic traits resolved into clusters. *The Journal of Abnormal and Social Psychology, 38*(4), 476–506.

Chan, P. K., Schlag, M. D. F., & Zien, J. Y. (1994). Spectral k-way ratio-cut partitioning and clustering. *IEEE Transactions on CAD of Integrated Circuits and Systems, 13*(9), 1088–1096.

Chehreghani, M. H. (2013). *Information-theoretic validation of clustering algorithms*. PhD thesis, ETH Zurich.

Chehreghani, M. H. (2017). Clustering by shift. In *2017 IEEE international conference on data mining, ICDM*, pp. 793–798.

Chehreghani, M. H. (2021). Reliable agglomerative clustering. In *International joint conference on neural networks (IJCNN)*. IEEE.

Chehreghani, M. H., Busetto, A. G., & Buhmann, J. M. (2012). Information theoretic model validation for spectral clustering. In *Proceedings of the fifteenth international conference on artificial intelligence and statistics, AISTATS*, vol. 22, pp. 495–503.

Chehreghani, M. H. (2016). Adaptive trajectory analysis of replicator dynamics for data clustering. *Machine Learning, 104*(2–3), 271–289.

Chehreghani, M. H., Abolhassani, H., & Chehreghani, M. H. (2008). Improving density-based methods for hierarchical clustering of web pages. *Data & Knowledge Engineering, 67*(1), 30–50.

Chen, Y., Zhang, Y., & Ji, X. (2005). Size regularized cut for data clustering. *Advances in Neural Information Processing Systems (NIPS), 18,* 211–218.

Demaine, E. D., Emanuel, D., Fiat, A., & Immorlica, N. (2006). Correlation clustering in general weighted graphs. *Theoretical Computer Science, 361*(2–3), 172–187.

Demetriou, A., Aåg, H., Rahrovani, S., & Chehreghani, M. H. (2020). A deep learning framework for generation and analysis of driving scenario trajectories. CoRR, arXiv: 2007.14524.

Dhillon, I. S., Guan, Y., & Kulis, B. (2004). Kernel k-means: Spectral clustering and normalized cuts. In *Proceedings of the tenth ACM SIGKDD international conference on knowledge discovery and data mining*, KDD '04, pp. 551–556. ACM.

Dhillon, I. S., Guan, Y., & Kulis, B. (2005). A unified view of kernel k-means, spectral clustering and graph cuts. Technical Report TR-04-25.

Ding, H. (2020). Faster balanced clusterings in high dimension. *Theoretical Computer Science, 842,* 28–40.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the second international conference on knowledge discovery and data mining (KDD)*, pp. 226–231.

Frank, M., Chehreghani, M. H., & Buhmann, J. M. (2011). The minimum transfer cost principle for model-order selection. In *European conference on machine learning and knowledge discovery in databases (ECML-PKDD)*, Lecture Notes in Computer Science, pp. 423–438.

Goldschmidt, O., & Hochbaum, D. S. (1994). A polynomial algorithm for the k-cut problem for fixed k. *Mathematics of Operations Research, 19*(1), 24–37.

Han, J., Liu, H., & Nie, F. (2019). A local and global discriminative framework and optimization for balanced clustering. *IEEE Transactions on Neural Networks and Learning Systems, 30*(10), 3059–3071.

Hofmann, T., & Buhmann, J. M. (1997). Pairwise data clustering by deterministic annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 19*(1), 1–14.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of Classification, 2*(1), 193–218.

Karger, D. R., & Stein, C. (1996). A new approach to the minimum cut problem. *Journal of the ACM (JACM), 43*(4), 601–640.

Lance, G. N., & Williams, W. T. (1967). A general theory of classificatory sorting strategies. *The Computer Journal, 9*(4), 373–380.

Leighton, T., & Rao, S. (1999). Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *Journal of the ACM (JACM), 46*(6), 787–832.

Lichman, M. (2013). UCI machine learning repository.

Lin, F., & Cohen, W. W. (2010). Power iteration clustering. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 655–662.

Lin, W., He, Z., & Xiao, M. (2019). Balanced clustering: A uniform model and fast algorithm. In *Proceedings of the twenty-eighth international joint conference on artificial intelligence (IJCAI)*, pp. 2987–2993. International Joint Conferences on Artificial Intelligence Organization.

Liu, H., Han, J., Nie, F., & Li, X. (2017). Balanced clustering with least square regression. In *Proceedings of the thirty-first AAAI conference on artificial intelligence*, pp. 2231–2237. AAAI Press.

Liu, H., Huang, Z., Chen, Q., Li, M., Fu, Y., & Zhang, L. (2018). Fast clustering with flexible balance constraints. In *IEEE international conference on big data (big data)*, pp. 743–750.

Liu, H., Latecki, L. J., & Yan, S. (2013). Fast detection of dense subgraphs with iterative shrinking and expansion. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(9), 2131–2142.

Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and Computing, 17*(4), 395–416.

Macqueen, J. (1967). Some methods for classification and analysis of multivariate observations. In *5-th Berkeley symposium on mathematical statistics and probability*, pp. 281–297.

Malinen, M. I., & Fränti, P. (2014). Balanced k-means for clustering. In P. Fränti, G. Brown, M. Loog, F. Escolano, & M. Pelillo (Eds.), *Structural, syntactic, and statistical pattern recognition*, Lecture Notes in Computer Science, vol. 8621, pp. 32–41. Berlin, Heidelberg: Springer. https://doi.org/10.1007/978-3-662-44415-3_4.

Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to information retrieval*. Cambridge University Press.

Ng, A. Y., Jordan, M. I., & Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. *Advances in Neural Information Processing Systems, 14,* 849–856.

Ng, B., McKeown, M. J., & Abugharbieh, R. (2012). Group replicator dynamics: A novel group-wise evolutionary approach for sparse brain network detection. *IEEE Transactions on Medical Imaging, 31*(3), 576–585.

Pavan, M., & Pelillo, M. (2003). Dominant sets and hierarchical clustering. In *9th IEEE international conference on computer vision (ICCV)*, pp. 362–369.

Pavan, M., & Pelillo, M. (2007). Dominant sets and pairwise clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(1), 167–172.

Reddi, S. J., Sra, S., Póczos, B., & Smola, A. J. (2016). Stochastic frank-wolfe methods for nonconvex optimization. In *54th annual Allerton conference on communication, control, and computing, Allerton 2016, Monticello, IL, USA, September 27–30, 2016*, pp. 1244–1251.

Rosenberg, A., & Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP-CoNLL*, pp. 410–420. ACL.

Roth, V., Laub, J., Kawanabe, M., & Buhmann, J. M. (2003). Optimal cluster preserving embedding of nonmetric proximity data. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25*(12), 1540–1551.

Schölkopf, B., Smola, A., & Müller, K.-R. (1998). Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation, 10*(5), 1299–1319.

Schuster, P., & Sigmund, K. (1983). Replicator dynamics. *Journal of Theoretical Biology, 100,* 533–538.

Shi, J., & Malik, J. (2000). Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22*(8), 888–905.

Sneath, P. H. A. (1957). The application of computers to taxonomy. *Journal of General Microbiology, 17,* 201–226.

Sokal, R. R., & Michener, C. D. (1958). A statistical method for evaluating systematic relationships. *University of Kansas Science Bulletin, 38,* 1409–1438.

Soundararajan, P., & Sarkar, S. (2001). Investigation of measures for grouping by graph partitioning. In *Proceedings of conference on computer vision and pattern recognition (CVPR)*, pp. 239–246.

Thiel, E., Chehreghani, M. H., & Dubhashi, D. P. (2019). A non-convex optimization approach to correlation clustering. In *The thirty-third AAAI conference on artificial intelligence, AAAI*, pp. 5159–5166.

Tryon, R. C. (1939). *Cluster analysis: Correlation profile and orthometric (factor) analysis for the isolation of unities in mind and personality*. Edwards Brother, Incorporated, Lithoprinters and Publishers.

Vinh, N. X., Epps, J., & Bailey, J. (2010). Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance. *The Journal of Machine Learning Research, 11,* 2837–2854.

Ward, J. H. (1963). Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association, 58*(301), 236–244.

Weibull, J. W. (1997). *Evolutionary game theory*. MIT Press, Cambridge, Mass. [u.a.].

Wu, Z., & Leahy, R. (1993). An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 15*(11), 1101–1113.

Yang, L., Cheung, N.-M., Li, J., & Fang, J. (2019). Deep clustering by Gaussian mixture variational autoencoders with graph embedding. In *International conference on computer vision (ICCV)*, pp. 6439–6448.