Comparing Subspace Clusterings*
Technical Report UW-CSE-2004-10-01

Anne Patrikainen† Marina Meilă‡

Abstract

We present the first framework for comparing subspace clusterings. We propose several distance measures for subspace clusterings, including generalizations of well-known distance measures for ordinary clusterings. We describe a set of important properties for any measure for comparing subspace clusterings and give a systematic comparison of our proposed measures in terms of these properties. We validate the usefulness of our subspace clustering distance measures by comparing clusterings produced by the algorithms FastDOC, HARP, PROCLUS, ORCLUS, and SSPC. We show that our distance measures can be also used to compare partial clusterings, overlapping clusterings, hierarchical clusterings, and patterns in binary data matrices.

Keywords: Subspace clustering, Projected clustering, Distance, Feature selection, Feature extraction, Cluster validation

1 Introduction

1.1 Subspace Clustering

The goal of clustering is to group a given set of data points into *clusters* that capture some notion of similarity between the data points in each cluster. Data is represented by a number of *features*, not all of which are useful for comparing individual data points. In particular, the choice of the set of features used to represent data may highlight different facets of the similarity between the data points. Subspace clustering was introduced in order to capture this idea of "similarity examined under different representations".

Conceptually, subspace clustering algorithms work on a collection of data points described using a large number of features, and address the problem of simultaneously selecting the relevant features, and the points that are similar

*This work was done with partial support from NSF grant No. 0313339.

†HIIT Basic Research Unit, Laboratory of Computer and Information Science, Helsinki University of Technology. annep@james.hut.fi. The work has been done while the author was visiting University of Washington.

‡Department of Statistics, University of Washington. mmp@stat.washington.edu.

given these features. Formally, a *subspace cluster* can be defined as a pair (subset of data points, subspace of the feature space). The data points of the cluster are similar in the associated subspace. A *subspace clustering* is a collection of subspace clusters.[1] The first[2] subspace clustering algorithm CLIQUE [4] was published in 1998 and was soon followed by many related methods [1, 2, 3, 8, 12, 13, 14, 15, 18, 19, 23, 31, 36, 38, 41, 46, 48, 53, 60, 62, 63]. The algorithms have been applied for instance to clustering gene expression data: it is often the case that a group of genes behaves similarly only in a subset of experiments (i.e. in a subspace) [14, 15, 19, 23, 53, 60, 62]. Reviews of some of the existing subspace clustering algorithms can be found in [44, 45, 64].

1.2 Comparing Subspace Clusterings

Surprisingly, despite the multitude of subspace clustering algorithms, there are no existing methods for comparing their outputs. Pairs of ordinary clusterings (partitions of the set of data points) can be compared with numerous well-known criteria, for instance with the Clustering Error (CE), the Variation of Information (VI), or the Rand index; some of these measures have been in use at least since the seventies [54]. However, these criteria are not directly applicable to comparing subspace clusterings. This is unfortunate, since clustering comparison methods are necessary for several important tasks, including cluster validation, meta-clustering, and consensus clustering; we introduce these in Section 1.3.

To the best of our knowledge, nobody has proposed a method for comparing two subspace clusterings in a way that takes into account the data point groups and the subspaces simultaneously. In the existing literature, authors most commonly compare only the grouping of data points into clusters, ignoring the similarity or dissimilarity of the associated subspaces [3, 19, 48, 53, 64]. Note that this approach does not work in the general case, since the data point clusters might be not form a partition of the set of all data points. Sometimes the subspaces are compared and the data point groups ignored [44, 46]; this is done qualitatively in the absence of suitable comparison methods. At best, the data points and the subspaces are compared separately, and the conclusions are once again only qualitative [2, 14]. All these approaches fail to compare subspace clusterings in a fair manner.

There are various types of subspace clusters and subspace clusterings. Perhaps the most common type of subspace cluster is an *axis-aligned subspace cluster*, in which the subspace is spanned by a subset of the attributes. In this case, an equivalent representation for the cluster is a pair (subset of data points, subset of attributes). A more general type of subspace cluster is a *non-axis-aligned*

[1] Other names that have been used for the same or a closely related task are projected clustering [2, 3, 62, 63], projective clustering [1, 48], bi-clustering [33, 33], co-clustering [8, 15, 17], coupled two-way clustering [24], simultaneous clustering [47], direct clustering [27], block clustering [27], and clustering on subsets of attributes [23].

[2] In fact, related ideas had been introduced earlier in [27, 42], but CLIQUE was the first algorithm that became widely known in the research community.

subspace cluster, in which the subspaces can be arbitrarily oriented. We present distance measures for both types of subspace clusterings and also address the closely related topics of *co-clusterings* and *attribute weighted clusterings*. These categories of algorithms are described in detail in Section 4.

1.3 Applications of Comparing Clusterings

1.3.1 Cluster Validation

Cluster validation refers to quantitatively evaluating the quality of a clustering solution. It is always important to validate the clustering solution after running a clustering algorithm, since bad clusterings arise in many common occasions. Most clustering algorithms give some kind of a result even if the data set does not have any clustering structure. Many algorithms give bad results if the choice of the parameter values or the initialization is inappropriate. This kinds of clusterings can be recognized and avoided by means of cluster validation.

Cluster validation can be divided into external and internal cluster validation [54]. External cluster validation refers to comparing a clustering solution to a true clustering; internal cluster validation evaluates the clustering result without any knowledge of a true clustering.

External cluster validation [37, 49, 54] is important in evaluating the performance of a clustering algorithm on synthetic data sets. It aims to measure the quality of clustering produced by the algorithm by comparing it to a true clustering of the data. Specifically, assume that we have a data set X for which the true subspace clustering $\mathcal{T}$ is known. We have two algorithms, A and A', which have produced subspace clusterings $\mathcal{S}$ and $\mathcal{S}'$. We would like to be able to calculate the distances $d(\mathcal{S}, \mathcal{T})$ and $d(\mathcal{S}', \mathcal{T})$ to find out which of these clusterings is closer to the true clustering. However, there are currently no distance measures for subspace clusterings.

Internal cluster validation aims to measure the quality of a clustering in real-life settings, when there is no knowledge of the real clustering, or if there is uncertainty of whether the data set can be clustered at all [34, 54]. Internal cluster validation can be done by means of *point configuration based methods*, such as the Dunn index [54] or the Davies-Bouldin index [16], which assess the structure of the clustering — for instance, a clustering with compact, spherical, well-separated clusters might be judged good. Another way to do internal cluster validation is to use *stability based methods*, which measure the stability of a clustering by sampling the data [21, 34, 35]. If the clusterings on different samples agree, the clustering is judged stable and therefore good. Naturally, clustering distance measures are needed for evaluating the agreement of the clusterings on different samples. However, the point configuration based methods do not need clustering distance measures, but these methods are not currently applicable to subspace clusterings in the first place.

1.3.2 Consensus Clustering

If we have several clusterings for the same data set, we might be interested in combining these clusterings into a single *consensus clustering*[3] which should be as close to all original clusterings as possible [26, 52, 55, 56, 57]. This kind of situation might arise if we are not sure which parameter values (such as the number of clusters K) of a given clustering algorithm to use; we might just run the algorithm for several choices of the parameter values and try to find a consensus among the resulting clusterings. Also, combining several results by the same algorithm could alleviate the effect of random initialization (for instance choosing the initial cluster centroids).

As another possibility, we might run several clustering algorithms with different objective functions for the same data set and combine all resulting clusterings into a single clustering. This might allow us to capture a rich variety of features which would not be possible for any single clustering algorithm alone. Further, combining the results of several different algorithms would improve the robustness and stability of clustering and decrease the sensitivity to outliers and noise.

Clustering ensembles can also be used to combine clusterings on different attributes of the same data set. One potential application of this would be parallelization; another is distributed data mining. Yet another application would be clustering categorical data, where each categorical attribute could be viewed as a clustering of the data set. [26, 55]

Many of the above cases apply to subspace clusterings in addition to ordinary clusterings. Naturally, if we wish to find a consensus clustering which is close to all the original clusterings, we must be able to calculate the distances between the clusterings. Given this definition of a good consensus clustering, having a distance measure for subspace clusterings is essential for finding a consensus subspace clustering. Of course, different definitions for a consensus clustering cost function might be possible, and clustering distance measures might not always be necessary.

1.3.3 Meta-clustering

Meta-clustering refers to investigating the structure of a set of clusterings. Meta-clustering discards the idea of trying to derive a single good clustering for a data set; instead, it is acknowledged that the data can be well represented in several different, complementary ways. For instance, assume that a given data set has been clustered several times by different algorithms. A meta-clusterer might now observe that these clusterings form two tight groups of clusterings, and give the user a representative of each of these groups, instead of a single 'best' clustering. [7]

There are various ways to produce different clusterings for a data set: we could use different algorithms, a single algorithm with various parameter values and initializations, change metrics, use various dimensionality reduction

[3] Also known as a clustering ensemble, or an aggregate clustering.

schemes, or sample the data. Meta-clustering may be used to investigate whether some of these clusterings form tight groups, whether some of the clusterings are outliers, whether the effect of the parameter values is strong or weak, etc. For instance, it has been empirically shown by means of meta-clustering that only a small number of clustering algorithms is enough to represent a large number of clustering criteria [29].

Sometimes meta-clustering is used in a broader sense to refer to all kinds of methods that operate on sets of clusterings; according to this definition, meta-clustering would include cluster validation and consensus clusterings as special cases [26]. Whichever definition we choose, it is clear that meta-clustering is impossible without a distance measure for clusterings.

1.4 Contributions of the Article

In this article, we address the problem of comparing subspace clusterings. Specifically, we introduce a set of important properties for a subspace clustering distance measure; we propose four novel subspace clustering distance measures (RNIA and generalizations of the Rand index, CE, and VI); we describe how our distance measures can be applied to axis-aligned subspace clusterings, non-axis-aligned subspace clusterings, attribute weighted clusterings, and co-clusterings; we investigate the theoretical properties of the proposed distance measures; we show experimentally that our distance measures are useful in practice.

As discussed earlier, subspace clustering distance measures are necessary in external cluster validation (comparing a clustering to a true clustering), stability based internal cluster validation (evaluating the stability of a clustering), meta-clustering (investigating the structure of a set of clusterings), and many cases of consensus clustering (finding a representative clustering for a set of clusterings).

In addition to comparing subspace clusterings, our distance measures can be used to compare partial clusterings (clusterings on subsets of data points; see Section 3), clusterings with overlapping clusters (a data point may belong to multiple clusters; see Section 3.4), hierarchical clusterings (see Section 6), and many patterns in data matrices (see Section 5.3).

1.5 Structure of the Article

We start by comparing axis-aligned subspace clusterings in Section 3. We will extend our analysis to the case of non-axis-aligned subspace clusterings, attribute weighted clusterings, and co-clusterings in Section 4. In Section 5, we apply our distance measures to comparing subspace clusterings produced by the algorithms FastDOC, HARP, PROCLUS, ORCLUS, and SSPC on synthetic data sets. We also show how our distance measures can be used to compare patterns in binary data matrices. Finally in Section 6, we present a summary of our work and discuss future research directions.

2 Comparing Ordinary Clusterings

A *clustering*[4] $\mathcal{C}$ is a partitioning of the set of m data points into disjoint clusters $C_1, C_2, \ldots, C_K$ of sizes $m_1, m_2, \ldots, m_K$, where $\sum_i m_i = m$. Virtually all criteria for comparing clusterings are based on the so-called *confusion matrix.* Assume that we have two clusterings $\mathcal{C} = \{C_1, C_2, \ldots, C_K\}$ and $\mathcal{C}' = \{C'_1, C'_2, \ldots, C'_{K'}\}$. The confusion matrix $M = (m_{ij})$ is a $K \times K'$ matrix whose ijth element is the number of points in the intersection of clusters C_i and C'_j, i.e. $m_{ij} = |C_i \cap C'_j|$.

An intuitive way to compare clusterings is to calculate the *clustering error* (CE). It is the proportion of points which are clustered differently in $\mathcal{C}$ and $\mathcal{C}'$ after an optimal matching of clusters. In other words, it is the scaled sum of the non-diagonal elements of the confusion matrix, minimized over all possible permutations of rows and columns. In practise, we do not need to try out all possible permutations; clustering error can be computed efficiently by the Hungarian method for finding maximum weight matching in a bipartite graph [43]. Clustering error is a metric.

An important class of criteria for comparing clusterings is based on counting the pairs of points on which two clusterings agree/disagree. Each pair of data points falls in one of the four categories labeled as N_{11}, N_{10}, N_{01}, and N_{00}. The category N_{11} contains the point pairs that are in the same cluster in both $\mathcal{C}$ and $\mathcal{C}'$. The category N_{10} contains the point pairs that are in the same cluster in $\mathcal{C}$ but not in $\mathcal{C}'$. The definitions of N_{01} and N_{00} are similar. All four counts can be obtained from the confusion matrix; for instance, $2N_{11} = \sum_{i,j} m_{ij}^2 - m$.

The best-known clustering distances based on point pair counts are the Wallace indices [58], the Fowlkes-Mallows index [22], the Rand index[49], the Jaccard index [28], and the Mirkin metric [39]. For instance, the Rand index is the proportion of point pairs on which the two clusterings agree, given as

$$\mathrm{Rand}(\mathcal{C}, \mathcal{C}') = \frac{N_{11} + N_{00}}{N}, \tag{1}$$

where $N = N_{11} + N_{10} + N_{01} + N_{00}$, the total number of point pairs. Out of the methods mentioned, the quantity (1 - Rand index) and the Mirkin measure are metrics.

Variation of information (VI) is a recently proposed clustering criterion based on information theoretic concepts [37]. It measures the amount of information that we gain and lose when going from clustering $\mathcal{C}$ to another clustering $\mathcal{C}'$. It is defined as $VI(\mathcal{C}, \mathcal{C}') = \mathrm{H}(\mathcal{C}|\mathcal{C}') + \mathrm{H}(\mathcal{C}'|\mathcal{C})$, where $\mathrm{H}(\mathcal{C}|\mathcal{C}')$ is the conditional entropy of $\mathcal{C}$ given $\mathcal{C}'$. An equivalent way of writing the VI distance is

$$\mathrm{VI}(\mathcal{C}, \mathcal{C}') = \frac{1}{m} \sum_{i=1}^{K} \sum_{j=1}^{K'} m_{ij} \log \frac{m_i m'_j}{m_{ij}^2}. \tag{2}$$

The VI distance is also a metric.

[4] In this paper we only discuss hard clusterings. In a *soft clustering*, a given data point r_i has a probability $P(r_i|C_j)$ of belonging to a given cluster C_j.

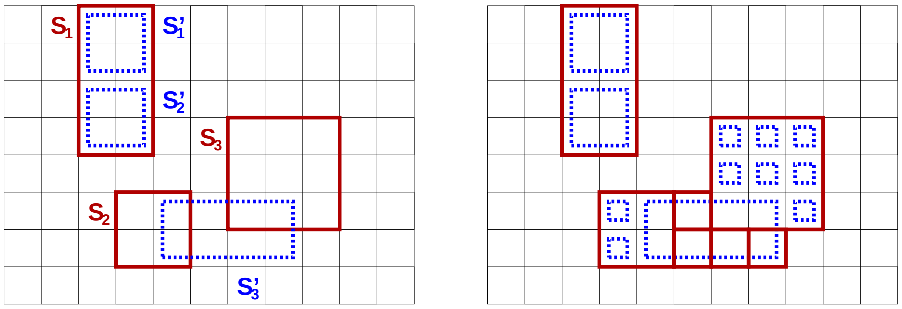


Figure 1: (Left) Two axis-aligned subspace clusterings which we wish to compare. The solid rectangles depict the clustering $\mathcal{S} = \{S_1, S_2, S_3\}$, and the dashed rectangles the clustering $\mathcal{S}' = \{S'_1, S'_2, S'_3\}$. For these clusterings, $\text{CE}(\mathcal{S}, \mathcal{S}') = 19/25$ and $\text{RNIA}(\mathcal{S}, \mathcal{S}') = 13/25$. (Right) To create a partition of the matrix elements, we have filled the non-intersecting areas with singleton clusters $S_4, \ldots, S_7$ and $S'_4, \ldots, S'_{12}$. This allows us to compute $\text{VI}(\mathcal{S}, \mathcal{S}') = 1.68$ and $1 - \text{Rand}(\mathcal{S}, \mathcal{S}') = 82/300$.

3 Comparing Axis-Aligned Subspace Clusterings

For the sake of simplicity, we start by considering axis-aligned subspace clusterings, the simplest and most popular type of subspace clusterings. We extend our analysis to non-axis-aligned subspace clusterings and other related types of clusterings in Section 4.

3.1 Size, Union, and Intersection of Axis-Aligned Subspace Clusters

Our data matrix $X = (x_{ij})$ has m rows and p columns. An *axis-aligned subspace cluster* S is a pair (R, C), where $R \subseteq \{r_1, r_2, \ldots, r_m\}$ is a subset of the rows and $C \subseteq \{c_1, c_2, \ldots, c_p\}$ is a subset of the columns. A *axis-aligned subspace clustering* $\mathcal{S}$ is a collection $\{S_1, S_2, \ldots, S_K\}$ of K subspace clusters.

In order to construct a clustering distance measure for axis-aligned subspace clusterings, we need to define the size, the union, and the intersection of subspace clusters and clusterings. To this end, we define the *support* of a cluster S_k as the set of matrix elements in it, given as $\text{supp}\,(S_k) = \{x_{ij} | r_i \in R_k \wedge c_j \in C_k\}$. The support of a clustering $\mathcal{S}$ is $\text{supp}\,(\mathcal{S}) = \bigcup_k \text{supp}\,(S_k)$. The size $|S_k|$ of a cluster is the number of matrix elements in its support. Similarly, the size $|\mathcal{S}|$ of a clustering is the number of matrix elements in its support. The union and the intersection of two subspace clusters are given as the union and the intersection of their supports. We denote the union of two subspace clusterings $\mathcal{S}$ and $\mathcal{S}'$ by $U = U(\mathcal{S}, \mathcal{S}') = \text{supp}\,(\mathcal{S}) \cup \text{supp}\,(\mathcal{S}')$ and the intersection by $I = I(\mathcal{S}, \mathcal{S}') = \text{supp}\,(\mathcal{S}) \cap \text{supp}\,(\mathcal{S}')$.

To simplify the analysis, we will start by considering only *disjoint subspace* *clusterings*, i.e., clusterings in which the clusters are disjoint in the sense that they do not share matrix elements: $\operatorname{supp}(S_k) \cap \operatorname{supp}(S_l) = \emptyset$ for all $k, l \in \{1, \ldots, K\}$ with $k \neq l$. It is important to note that any given data point or an attribute may still be relevant for subspaces of multiple clusters. A large number of subspace clustering algorithms are guaranteed to produce disjoint clusterings; exceptions include algorithms of the type of CLIQUE [4] and SUBCLU [31]. We discuss the extension of the analysis to non-disjoint subspace clusterings in Section 3.4.

3.2 Properties of a Distance Measure

Comparing subspace clusterings differs greatly from comparing ordinary clusterings, since subspace clusterings are not partitions of the data matrix. We will therefore start by introducing a set of properties for describing a subspace clustering comparison distance $d(\mathcal{S}, \mathcal{S}')$. We will use these properties to characterize and compare different distance measure candidates.

Metric. A distance measure d is a *metric* if it satisfies three axioms: positivity, symmetry, and triangle inequality.

Positive. For all $\mathcal{S}$ and $\mathcal{S}'$ we have $d(\mathcal{S}, \mathcal{S}') \geq 0$ and $d(\mathcal{S}, \mathcal{S}') = 0$ if and only if $\mathcal{S} = \mathcal{S}'$.

Symmetric. For all $\mathcal{S}$ and $\mathcal{S}'$ we have $d(\mathcal{S}, \mathcal{S}') = d(\mathcal{S}', \mathcal{S})$.

Triangle inequality. For all $\mathcal{S}$, $\mathcal{S}'$, and $\mathcal{S}''$ we have $d(\mathcal{S}, \mathcal{S}') \leq d(\mathcal{S}, \mathcal{S}'') + d(\mathcal{S}'', \mathcal{S}')$.

Label permutation invariant. An alternative way to consider an axis-aligned subspace clustering $\mathcal{S} = \{S_1, S_2, \ldots, S_K\}$ is to view it as a function $\mathcal{S} : \{x_{ij}\} \to \{1, 2, \ldots, K\}$. This function assigns cluster indices to data matrix elements: we refer to the cluster index $\mathcal{S}(x_{ij})$ as the *cluster label* of the element x_{ij}. A *permutation* $\rho : \{1, 2, \ldots, K\} \to \{1, 2, \ldots, K\}$ is a bijection from the set of cluster labels to the set of cluster labels. Let us permute the cluster labels of $\mathcal{S}$ by ρ and denote the resulting clustering by $\mathcal{S}^\rho$. In other words, we have $\mathcal{S}^\rho(x_{ij}) = \rho(\mathcal{S}(x_{ij}))$. A clustering distance measure d is *label permutation invariant* if $d(\mathcal{S}^\rho, \mathcal{S}'^\pi) = d(\mathcal{S}, \mathcal{S}')$ for any permutations ρ and π and any clusterings $\mathcal{S}$ and $\mathcal{S}'$.

Penalty for non-intersecting area. Consider two clusterings $\mathcal{S}$ and $\mathcal{S}'$, the data matrix elements in their union U, in their intersection I, and in $U \backslash I$. Let us refer to $|U \backslash I|$ as the *non-intersecting area* of these two clusterings. Consider adding one or more unclustered data matrix elements $x_{ij} \notin U$ to $U \backslash I$. In effect, we increase the non-intersecting area of the two clusterings while keeping everything else unchanged. The new matrix elements might have been added to $\mathcal{S}$ only, $\mathcal{S}'$ only, or to both clusterings. Let us denote the new clusterings by $\mathcal{S}^U$ and $\mathcal{S}'^U$; note that one of these clusterings might in fact be equal to the original one. A distance measure d *penalizes for non-intersecting area* if $d(\mathcal{S}^U, \mathcal{S}'^U) > d(\mathcal{S}, \mathcal{S}')$ for any clusterings $\mathcal{S}$ and $\mathcal{S}'$.

Background independent. Consider two clusterings $\mathcal{S}$ and $\mathcal{S}'$ on the data matrix X of size $m \times p$. Let us introduce an alternative notation $\mathcal{S}_X$ and $\mathcal{S}'_X$ in order to emphasize that we have clusterings on X. Now consider adding m'

rows and p' columns to X and denote the new $(m+m') \times (p+p')$ data matrix by X'. The matrix element sets $\mathrm{supp}\,(\mathcal{S})$ and $\mathrm{supp}\,(\mathcal{S}')$ are included in X', so we can write $\mathcal{S}_{X'}$ and $\mathcal{S}'_{X'}$ for the same clusterings on the larger data matrix X'. A distance measure d is *background independent* if $d(\mathcal{S}_X, \mathcal{S}'_X) = d(\mathcal{S}_{X'}, \mathcal{S}'_{X'})$ for any clusterings $\mathcal{S}$ and $\mathcal{S}'$ and $m' > 0$ or $p' > 0$.[5]

Scale invariant. Consider scaling the data matrix X by a constant $k \in \mathbb{Z}^+$, or in other words, introducing k copies of each row and column of the matrix. Let us denote the scaled data matrix by kX. While X has rows $\{r_1, \ldots, r_m\}$ and columns $\{c_1, \ldots, c_p\}$, kX has rows $\{r_{11}, r_{12}, \ldots, r_{1k}, \ldots, r_{m1}, r_{m2}, \ldots, r_{mk}\}$ and columns $\{c_{11}, r_{12}, \ldots, c_{1k}, \ldots, c_{m1}, c_{m2}, \ldots, c_{mk}\}$. Now consider a subspace clustering $\mathcal{S}$ on X and its scaled version $k\mathcal{S}$ on kX. If a cluster $S_i = (R_i, C_i)$ of $\mathcal{S}$ has the row r_j in its row set R_i, then the cluster $kS_i = (kR_i, kC_i)$ of $k\mathcal{S}$ has the rows $r_{j1}, r_{j2}, \ldots, r_{jk}$ in its row set kR_i. Similarly, if a cluster $S_i = (R_i, C_i)$ of $\mathcal{S}$ has the column c_j in its column set C_i, then the cluster $kS_i = (kR_i, kC_i)$ of $k\mathcal{S}$ has the columns $c_{j1}, c_{j2}, \ldots, c_{jk}$ in its column set kC_i. A distance measure d is *scale invariant* if $d(k\mathcal{S}, k\mathcal{S}') = d(\mathcal{S}, \mathcal{S}')$ for all clusterings $\mathcal{S}$ and $\mathcal{S}'$ and for all $k \in \mathbb{Z}^+$.

Copy invariant. Consider two subspace clusterings $(\mathcal{S}, \mathcal{S}')$. Next consider introducing a disjoint copy of this pair of clusterings in a large data matrix X, resulting in a pair of 'double clusterings' $(\mathcal{S}^D, \mathcal{S}'^D)$. Unlike in the scale invariance property, we are not altering the size of the data matrix here. In other words, introduce a copy $\tilde{\mathcal{S}}$ of $\mathcal{S}$ and a copy $\tilde{\mathcal{S}}'$ of $\mathcal{S}'$ such that the new cluster sizes $\{\tilde{m}_i\}$, $\{\tilde{m}'_i\}$ and cluster intersection sizes $\{\tilde{m}_{ij}\}$ equal to the old ones ($\{m_i\}$,$\{m'_i\}$,$\{m_{ij}\}$) and that $\mathrm{supp}\,(\mathcal{S}) \cap \mathrm{supp}\,(\tilde{\mathcal{S}}) = \emptyset$ and $\mathrm{supp}\,(\mathcal{S}') \cap \mathrm{supp}\,(\tilde{\mathcal{S}}') = \emptyset$. Then the 'double clusterings' are given by $\mathcal{S}^D = \{S_1, S_2, \ldots, S_K, \tilde{S}_1, \tilde{S}_2, \ldots, \tilde{S}_K\}$ and $\mathcal{S}'^D = \{S'_1, S'_2, \ldots, S'_{K'}, \tilde{S}'_1, \tilde{S}'_2, \ldots, \tilde{S}'_{K'}\}$ A distance measure d is *copy invariant* if $d(\mathcal{S}, \mathcal{S}') = d(\mathcal{S}^D, \mathcal{S}'^D)$ for all clusterings $\mathcal{S}$ and $\mathcal{S}'$.

Requires partitioning. Consider two clusterings $\mathcal{S}$ and $\mathcal{S}'$ and the set of matrix elements in their union U. In the general case, neither of these clusterings is a partition of U, since the elements in $\mathrm{supp}\,(\mathcal{S}) \backslash \mathrm{supp}\,(\mathcal{S}')$ are not clustered by $\mathcal{S}'$, and the elements in $\mathrm{supp}\,(\mathcal{S}') \backslash \mathrm{supp}\,(\mathcal{S})$ are not clustered by $\mathcal{S}$. A subspace clustering distance measure d that *requires the clusterings to be partitions* of the data matrix elements in the union U therefore forces us to modify $\mathcal{S}$ and $\mathcal{S}'$ before we are able to compute $d(\mathcal{S}, \mathcal{S}')$. In order to transform $\mathcal{S}$ into a partition of the data matrix elements in U, we assign each $x_{ij} \notin \mathrm{supp}\,(\mathcal{S}') \backslash \mathrm{supp}\,(\mathcal{S})$ to a new singleton cluster. If $\mathcal{S}$ originally had K clusters, its modified version therefore has $K + |\mathrm{supp}\,(\mathcal{S}') \backslash \mathrm{supp}\,(\mathcal{S})|$ clusters. Similar transformation is done for $\mathcal{S}'$. See Fig. 1 for an example.

Multiple cluster coverage penalty. Consider two clusterings $\mathcal{S}$ and $\mathcal{S}'$ such that $\mathcal{S} = \{S_1\}$ consists only of a single cluster and $\mathcal{S}' = \{S'_1, \ldots, S'_{K'}\}$

[5] To motivate the background independence property, let us consider the following. The distance $d(\mathcal{S}, \mathcal{S}')$ should not be affected by the size of the data matrix X, only the size of the union U of the two clusterings. For instance, increasing the size of X by adding noise rows and columns should not move the clusterings $\mathcal{S}$ and $\mathcal{S}'$ closer to each other.

consists of K' disjoint clusters which all have the same size, with $\text{supp}(\mathcal{S}) = \text{supp}(\mathcal{S}')$. The clustering $\mathcal{S}'$ therefore clusters the same matrix elements than $\mathcal{S}$ but uses multiple clusters to cover the area. For an example of the case $K = 2$, see Fig. 1. A distance measure d *penalizes for multiple cluster coverage* if $d(\mathcal{S}, \mathcal{S}') \neq 0$ for $K' > 1$.

Generalizable. A distance measure for axis-aligned subspace clusterings is *generalizable* if it can also be applied to non-axis-aligned subspace clusterings and other related types of clusterings.

Handles ordinary clusterings. A distance measure $d(\mathcal{S}, \mathcal{S}')$ *handles ordinary clusterings* if it produces sensible results in the case in which $\mathcal{S}$ and $\mathcal{S}'$ are ordinary clusterings (partitions of the same element set).

Easy to compute.

Intuitive and understandable.

3.3 Distance Measures for Subspace Clusterings

We now present methods for comparing subspace clusterings by generalizing well-known distance measures for ordinary clusterings. We consider the Clustering Error (CE), the Rand index (as a well-known representative of the point pair counting based methods), and the Variation of Information (VI). In addition, we introduce a new distance measure, relative non-intersecting area (RNIA). We define and briefly discuss each of these four distance measures below; a comprehensive comparison of their properties is presented in Table 1.

Since we are comparing subspace clusterings, we consider the set of the data matrix elements $\{x_{ij}\}$ as our base element set, instead of the set of data points (rows).

3.3.1 Clustering Error

Consider subspace clusterings $\mathcal{S} = \{S_1, S_2, \ldots, S_K\}$ and $\mathcal{S}' = \{S'_1, S'_2, \ldots, S'_{K'}\}$ of K and K' clusters, respectively. Recall from Section 2 that a confusion matrix $M = (M_{ij})$ is a $K \times K'$ matrix in which m_{ij} is the number of data matrix elements shared by the clusters S_i and S'_j. More formally, $m_{ij} = |\text{supp}(S_i) \cap \text{supp}(S'_j)|$. Note, however, that in the case of subspace clusters, the rows and the columns of M do not necessarily sum up to the cluster sizes. That is, $\sum_i m_{ij} \leq |S'_j|$ and $\sum_j m_{ij} \leq |S_i|$.

Let us transform M into a square matrix by adding rows or columns of zeroes if necessary and use the Hungarian method [43] to find a permutation of the cluster labels such that the sum of the diagonal elements of M is maximized. Denote this maximized sum by D_{max}. Now, we define the clustering error (CE) for subspace clusterings as

$$\text{CE}(\mathcal{S}, \mathcal{S}') = \frac{|U| - D_{max}}{|U|}. \tag{3}$$

In the case of ordinary clusterings (partitions of the rows of the data matrix), the clustering error defined here is the clustering error of Section 2.

For the two clusterings of Fig. 1, the confusion M is

$$\begin{array}{c|ccc} & S'_1 & S'_2 & S'_3 \\ \hline S_1 & 4 & 4 & 0 \\ S_2 & 0 & 0 & 2 \\ S_3 & 0 & 0 & 2 \end{array} .$$

We also have $|U| = 25$, $D_{max} = 6$, and thus $\mathrm{CE}(\mathcal{S}, \mathcal{S}') = 19/25$.

3.3.2 Rand Index

The Rand index for ordinary clusterings is based on counting pairs of data points, but the Rand index for subspace clusterings is based on counting pairs of matrix elements. Recall that we need the quantities N_{11} (the number of pairs of matrix elements in the same cluster in both $\mathcal{S}$ and $\mathcal{S}'$), N_{00} (the number of pairs of matrix elements in a different cluster in both $\mathcal{S}$ and $\mathcal{S}'$) and N (the total number of pairs of matrix elements) for calculating the value of the Rand index according to Eq. 1.

Since we want our distance measure to be background independent, we will only count pairs of matrix element in U, the union of the two clusterings. Therefore $N = |U|(|U| - 1)/2$. However, to compute the values of N_{01} and N_{10}, $\mathcal{S}$ and $\mathcal{S}'$ have to be partitions of U. To this end, we will make $\mathcal{S}$ a partition by filling the non-intersecting area $U \backslash \mathrm{supp}\,(\mathcal{S})$ with extra singleton clusters. We will similarly convert $\mathcal{S}'$ into a partition of U.

If we considered this non-intersecting area area as a single big extra cluster, we would end up having zero distance between many different clusterings[6]. The only way around this difficulty seems to be to fill the non-intersecting area with extra singleton clusters, as illustrated in Fig. 1. After this filling procedure for both clusterings, we are able to compute the values of N_{11} and N_{00} as usual, for instance with the help of the confusion matrix.

When we discuss the Rand index, we actually consider $1 - \mathrm{Rand}(\mathcal{S}, \mathcal{S}') = (N_{01} + N_{10})/N$, since this is a proper distance measure: It assumes zero values for identical clusterings and positive values for non-identical clusterings.

If we transform the two clusterings of Fig. 1 into partitions by adding singleton clusters, the resulting confusion matrix M is

$$\begin{array}{c|ccc|ccccc|c} & S'_1 & S'_2 & S'_3 & S'_4 & S'_5 & S'_6 & \dots & S'_{12} & \\ \hline S_1 & 4 & 4 & 0 & 0 & 0 & 0 & \dots & 0 & 8 \\ S_2 & 0 & 0 & 2 & 1 & 1 & 0 & \dots & 0 & 4 \\ S_3 & 0 & 0 & 2 & 0 & 0 & 1 & \dots & 1 & 9 \\ \hline S_4 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 & 1 \\ S_5 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 & 1 \\ S_6 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 & 1 \\ S_7 & 0 & 0 & 1 & 0 & 0 & 0 & \dots & 0 & 1 \\ \hline & 4 & 4 & 8 & 1 & 1 & 1 & \dots & 1 & 25 \end{array} .$$

[6] For instance, consider two single-cluster clusterings $\mathcal{S} = \{S_1\}$ and $\mathcal{S}' = \{S'_1\}$ such that $\mathrm{supp}\,(S_1) \cap \mathrm{supp}\,(S'_1) = \emptyset$.

Based on this confusion matrix, we get $N = 300$, $N_{11} = 14$, $N_{00} = 204$, $N_{01} = 26$, $N_{10} = 56$, and $1 - \mathrm{Rand}(\mathcal{S}, \mathcal{S}') = 82/300$.

3.3.3 Variation of Information

Like the Rand index, the Variation of Information (VI) also requires the clusterings to be partitions. For reasons similar to the case of the Rand index, we fill the non-intersecting areas of the clusterings with extra singleton clusters. After this step, we can easily compute the confusion matrix and thereby calculate the VI distance according to Eq. 2. For the two clusterings of Fig. 1, $\mathrm{VI}(\mathcal{S}, \mathcal{S}') = 1.68$.

3.3.4 Relative Non-Intersecting Area

If we want to compare a subspace clustering $\mathcal{S}$ to a true clustering $\mathcal{S}'$, a simple approach would be to calculate the *precision*, the *recall*, and the *F-measure*, used widely in the information retrieval literature to measure the success of the retrieval task [50]. Retrieval is similar to subspace clustering in that it aims to extract a subset of the data that is alike in some respect, while the rest of the data is not assumed to be grouped in any way. Hence, a subspace clustering is like the unsupervised retrieval of several disjoint groups.

Using our subspace clustering notation, recall is defined as $|I|/\mathrm{supp}\,(\mathcal{S}')$; it measures how big part of the matrix elements of the true clustering $\mathcal{S}'$ is retrieved (covered) by the clustering $\mathcal{S}$. Precision is defined as $|I|/\mathrm{supp}\,(\mathcal{S})$; it measures the proportion of the matrix elements in the clustering $\mathcal{S}$ that belong to the true clustering $\mathcal{S}'$. The F-measure is just the geometric mean of the precision and the recall.

A big drawback of these measures is that they are not symmetric. A symmetric alternative is the *relative non-intersecting area (RNIA)*[7] of the two clusterings:

$$\mathrm{RNIA}(\mathcal{S}, \mathcal{S}') = \frac{|U| - |I|}{|U|}. \tag{4}$$

For the two clusterings of Fig. 1, $\mathrm{RNIA}(\mathcal{S}, \mathcal{S}') = 19/25$.

3.3.5 Comparing Distance Measures

Table 1 is a summary of the various properties of CE, Rand index, VI, and RNIA. Rand index and VI have some serious drawbacks: They do not satisfy all metric axioms, and they fail to penalize for the non-intersecting area in certain special cases.[8] RNIA behaves better, even though it does not satisfy all metric axioms either and cannot be used for comparing ordinary clusterings (it always gives zero distance for partitions of the same data set). The properties of CE

[7] Also known as the *symmetric difference* of two sets.

[8] Consider clusterings $\mathcal{S}$ and $\mathcal{S}'$. Add a few clusters of size 2 to $\mathcal{S}$, such that these new clusters are not included in $\mathrm{supp}\,(\mathcal{S}')$. After the addition, $d(\mathcal{S}, \mathcal{S}')$ might decrease, contradicting intuition.

	CE	RNIA	VI	1-Rand
Positive	✓	–	✓	✓
Symmetric	✓	✓	✓	✓
Triangle inequality	✓	✓	–	–
Label permutation invariant	✓	✓	✓	✓
Penalty for non-intersecting area	✓	✓	–	–
Background independent	✓	✓	✓	✓
Scale invariant	✓	✓	–	–
Copy invariant	✓	✓	✓	–
Lower bound	0	0	0	0
Upper bound	1	1	$\log(U)$	1
Requires partitioning	–	–	✓	✓
Multiple cluster coverage penalty	$\frac{K-1}{K}U$	0	$\log(K)$	$\frac{U(K-1)}{K(U-1)}$
Generalizable	✓	✓	✓	–
Handles ordinary clusterings	✓	–	✓	✓
Easy to compute	✓	✓	✓	✓
Intuitive and understandable	✓	✓	✓	✓

Table 1: Subspace clustering comparison properties of the Clustering Error (CE), the Relative Non-Intersecting Area (RNIA), the Variation of Information (VI), and 1-Rand. The proofs of the bold-face properties can be found in the Appendix; the rest of the proofs are straightforward.

are superior to the properties of the other distance measures; note specifically that CE is the only metric we have.

We will focus on CE and RNIA in the rest of this article, since these two measures have more desirable properties than VI or Rand.

A final point to note is that RNIA simply measures the intersecting area of the two clusterings and loses a lot of information in doing that. In fact, since CE requires one-to-one matching between clusters, but RNIA rewards for all overlaps, we have the following proposition.

Proposition 1. *For all $\mathcal{S}$ and $\mathcal{S}'$ we have $CE(\mathcal{S}, \mathcal{S}') \geq RNIA(\mathcal{S}, \mathcal{S}')$.*

3.4 Comparing Non-Disjoint Clusterings

In a non-disjoint subspace clustering $\mathcal{S}$, some of the clusters share matrix elements. We can compare this kinds of clusterings by duplicating certain matrix elements in a way such that the result clusterings become disjoint. The previously introduced methods can be then applied.

More specifically, consider a matrix element x_{ij} that belongs to the support of $n_{ij}^{\mathcal{S}}$ clusters of the clustering $\mathcal{S}$ and to the support of $n_{ij}^{\mathcal{S}'}$ clusters of the clustering $\mathcal{S}'$. To make the clustering $\mathcal{S}$ disjoint, we need to have $n_{ij}^{\mathcal{S}}$ copies of the matrix element x_{ij}. To make the clustering $\mathcal{S}'$ disjoint, we need to have $n_{ij}^{\mathcal{S}'}$ copies of x_{ij}. To make both clusterings disjoint simultaneously, $\max(n_{ij}^{\mathcal{S}}, n_{ij}^{\mathcal{S}'})$ copies of x_{ij} are needed.

Essentially, the element duplication procedure corresponds to redefining the union size of two clusterings $\mathcal{S}$ and $\mathcal{S}'$ as

$$|U| = \sum_{i,j} \max(n_{ij}^{\mathcal{S}}, n_{ij}^{\mathcal{S}'}) \tag{5}$$

and the intersection size as

$$|I| = \sum_{i,j} \min(n_{ij}^{\mathcal{S}}, n_{ij}^{\mathcal{S}'}). \tag{6}$$

Plugging in these definitions for $|U|$ and $|I|$, RNIA can be computed straightforwardly using Eq. 4. As for CE, the cluster intersection matrix can be formed and its diagonal sum maximized as usual. After this, Eq. 3 can be used together with the above definition of $|U|$.

4 Comparing Other Types of Clusterings

In this section, we first extend our analysis to comparing non-axis-aligned subspace clusterings. The comparison scheme we propose includes axis-aligned subspace clusterings as a special case. Further, we observe that similar principles can be applied to comparing attribute weighted clusterings. Lastly, we address the problem of comparing co-clusterings.

4.1 Comparing Non-Axis-Aligned Subspace Clusterings

A *non-axis-aligned subspace cluster* S is a pair (R, W), where $R \subseteq \{r_1, r_2, \ldots, r_m\}$ is a subset of the data points and W is a collection of vectors $\{w_1, w_2, \ldots, w_D\}$, $w_i \in \mathbb{R}^p$. The vectors in W form a basis for a subspace of the original p-dimensional data space. We use W also to denote this subspace. A *non-axis-aligned subspace clustering* $\mathcal{S}$ is a collection $\{S_1, S_2, \ldots, S_K\}$ of K non-axis aligned subspace clusters.

To simplify the analysis, we require analogously to the axis-aligned case that the clusters of a non-axis-aligned subspace clustering are disjoint. By this we mean that if two clusters share data points, the associated subspaces must be orthogonal. An extension to the case of non-disjoint clusterings seems possible but complicated, and we leave it for future studies.

We can compare non-axis-aligned subspace clusterings using CE or RNIA just as we compared axis-aligned ones if we first define the size of a cluster and the union and the intersection of two clusters. These are introduced next, by means of the principal angles between two subspaces.

4.1.1 Principal Angles

Let us consider two subspaces of $\mathbb{R}^n$, $\mathcal{F}$ and $\mathcal{G}$, such that $p = \dim \mathcal{F} \geq \dim \mathcal{G} = q \geq 1$. The q *principal angles* $\theta_1, \theta_2, \ldots, \theta_q \in [0, \pi/2]$ can be used to measure the similarity of the subspaces. The angles can be defined sequentially for $k = 1, 2, \ldots, q$ by

$$\cos(\theta_k) = \max_{w \in \mathcal{F}, v \in \mathcal{G}} w^T v \tag{7}$$

with

$$(w_k, v_k) = \arg \max_{w \in \mathcal{F}, v \in \mathcal{G}} w^T v \tag{8}$$

subject to

$$\begin{array}{l} ||w|| = ||v|| = 1, \quad w^T w_i = 0, \quad v^T v_i = 0, \\ i = 1, 2, \ldots, k-1. \end{array} \tag{9}$$

The vectors $w_1, w_2, \ldots, w_q$ and $v_1, v_2, \ldots, v_q$ are referred to as the *principal vectors.* A pair of vectors (w_i, v_i) is referred to as a *principal pair.*

Let us clarify the definition a bit. Given two subspaces $\mathcal{F}$ and $\mathcal{G}$, we first find vectors $w_1 \in \mathcal{F}$ and $v_1 \in \mathcal{G}$ such that the angle between these vectors is as small as possible. This angle is referred to as the first principal angle θ_1. We now proceed to finding vectors $w_2 \in \mathcal{F}$ and $v_2 \in \mathcal{G}$ such that the angle between these vectors is minimized, given the additional restriction that w_2 has to be orthogonal with w_1, and that v_2 has to be orthogonal with v_1. This gives us the second principal angle θ_2. We continue finding principal angles this way. The vectors $w_1, w_2, \ldots$ form an orthogonal set in the subspace $\mathcal{F}$, and the vectors $v_1, v_2, \ldots$ form an orthogonal set in the subspace $\mathcal{G}$. Due to this restriction, the maximum number of these vectors (and hence principal angles) is naturally $\min(\dim \mathcal{F}, \dim \mathcal{G})$. An illustrative example of principal angles is given in Fig. 2.

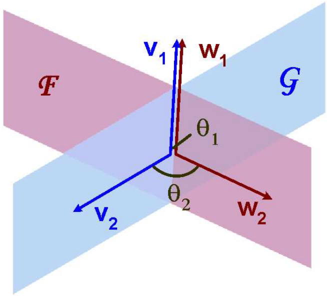


Figure 2: Example of principal angles between two subspaces. We have two 2-dimensional subspaces $\mathcal{F}$ and $\mathcal{G}$ in a 3-dimensional space. We thus have two principal angles θ_1 and θ_2 between these subspaces. The first principal angle θ_1 is the angle between the vectors w_1 and v_1; it is naturally zero. The second principal angle θ_2 is the angle between the vectors w_2 and v_2. The vectors w_1, w_2 are orthogonal and lie on the subspace $\mathcal{F}$. Similarly, the vectors v_1, v_2 are orthogonal and lie on the subspace $\mathcal{G}$.

Singular value decomposition (SVD) is a convenient way to compute the principal angles. Let the matrices $Q_{\mathcal{F}} \in \mathbb{R}^{n\times p}$ and $Q_{\mathcal{G}} \in \mathbb{R}^{n\times q}$ contain orthonormal bases for the subspaces $\mathcal{F}$ and $\mathcal{G}$, respectively. The SVD of $Q_{\mathcal{F}}^T Q_{\mathcal{G}}$ is

$$Y^T Q_{\mathcal{F}}^T Q_{\mathcal{G}} Z = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_q), \tag{10}$$

where $Y \in \mathbb{R}^{p\times p}$ and $Z \in \mathbb{R}^{q\times q}$ are orthogonal matrices. The ith singular value σ_i is just the cosine of the ith principal angle, i.e. $\sigma_i = \cos(\theta_i)$.

The cosines of the principal angles are also known as canonical correlations and have important applications for instance in statistics, econometrics, and geology. Principal angles can also be used to solve certain constrained optimization problems. [6, 11, 20, 42]

4.1.2 Size, Union, and Intersection of Non-Axis-Aligned Subspace Clusters

In order to use CE, RNIA, or other distance measures with non-axis-aligned subspace clusterings, we need to define the size of a non-axis-aligned subspace cluster and the union and intersection of two such clusters. In case of axis-aligned subspace clusters, our base elements were the matrix elements, or the pairs (data point, attribute). Analogously, the base element here is (data point l, basis vector w).

We count the number of these base elements in a cluster $S_i = (R_i, W_i)$, whose size becomes naturally $m_i = |R_i| \cdot \dim(W_i)$. Continuing with the analogy, we define the intersection of two clusters $S_i = (R_i, W_i) \in \mathcal{S}$ and $S'_j = (R'_j, V_j) \in \mathcal{S}'$

as $m_{ij} = |R_i \cap R'_j| \sum_{k=1}^{q} \sigma_k^2$, where q is the minimum of the dimensions of W_i and V_j and $\{\sigma_k\}$ are the principal angles between W_i and V_j.

This seemingly arbitrary definition for the intersection of two non-axis-aligned subspace clusters has a twofold motivation. First, it is geometrically consistent, as Theorem 1 shows, and in the axis-aligned case it coincides with the previously defined m_{ij} (the number of matrix elements shared by the two clusters). The second, probabilistic, motivation comes from viewing m_{ij}/m_i as the probability of seeing label j in $\mathcal{S}'$ if we randomly pick a point from cluster $S_i \in \mathcal{S}$. This view is used in the original definition of the VI distance and is implicit also in the other distance measures we have considered [37].

Let us investigate the probabilistic motivation in more detail. For any vector w denote by $\Pi_V\, w$ the projection of w on subspace V. Note that $||\Pi_V\, w||_2^2 = \cos^2(w, V)||w||_2^2$. We imagine the following sampling process:

1. Pick uniformly a point $l \in R_i$, and if $l \notin R'_j$, stop with 0 successes.
2. Else, pick a random orthonormal basis $\mathcal{B}_{W_i}$ in W_i.
3. For each $w \in \mathcal{B}_{W_i}$, "map w probabilistically to V_j" by counting a success with probability $\cos^2(w, V_j)$ and a failure with probability $\sin^2(w, V_j)$.

It can be shown that the expected number of successes in m_i trials is equal to m_{ij} as defined above.

We have to make sure that the size of the intersection of a subspace cluster S_i with the clusters of the other clustering does not exceed the size of S_i. The following theorem shows this; the proof is presented in the Appendix.

Theorem 1. *Assume that we have a p-dimensional subspace $\mathcal{F}$ and k orthogonal subspaces $\mathcal{G}_1, \mathcal{G}_2, \ldots, \mathcal{G}_k$ of dimensionalities $q_1, q_2, \ldots, q_k$. We compute the principal angles $\theta^1_{\mathcal{F},\mathcal{G}_i}, \ldots, \theta^{a_i}_{\mathcal{F},\mathcal{G}_i}$ between all subspace pairs $\mathcal{F}$, $\mathcal{G}_i$; here $a_i = \min(p, q_i)$. It always holds that $\sum_{i=1}^{k} \sum_{j=1}^{a_i} \cos^2(\theta^j_{\mathcal{F},\mathcal{G}_i}) \leq p$. Equality is attained if and only if $\mathcal{F}$ admits an orthogonal decomposition*[9] *over the subspaces $\{\mathcal{G}\}$.*

We have defined the cluster size m_i for a non-axis-aligned subspace cluster. We have also defined the size m_{ij} of the intersection of two non-axis-aligned subspace clusters, motivated this definition, and made sure that it is geometrically consistent. We now proceed to defining the size of the intersection and union of two non-axis-aligned subspace clusterings.

The size of the intersection $|I|$ of two non-axis-aligned subspace clusterings is naturally defined as the sum of the intersections of the cluster pairs: $|I| = \sum_{ij} m_{ij}$. The union size $|U|$ of two clusterings can be defined as $|U| = \sum_i m_i + \sum_j m_j - \sum_{ij} m_{ij}$.

After these definitions, we can calculate the distances between two non-axis-aligned subspace clusterings with CE or RNIA simply by using the familiar Eqs. 3 and 4. To compute CE, we only need the confusion matrix M, which can be constructed using the cluster intersection sizes m_{ij} and the union size

[9] In other words, the subspace $\mathcal{F}$ is spanned by the collection of the basis vectors for subspaces $\{\mathcal{G}_i\}_i$.

$|U|$. To compute RNIA, only the union size $|U|$ and the intersection size $|I|$ are needed.

4.1.3 Example of Non-Axis-Aligned Subspace Clusterings

Assume that we have two non-axis-aligned subspace clusterings, $S = \{S_1, S_2\}$ and $S' = \{S'_1, S'_2\}$, defined as follows.

$$\begin{array}{rcl} S_1 & = & (\{1,2,3\}, \{[1,1,0,0]^T/\sqrt{2}\}) \\ S_2 & = & (\{5,6,7\}, \{[0,0,1,0]^T, \\ & & [0,1,0,1]^T/\sqrt{2}, [1,-2,0,2]^T/3\}) \\ S'_1 & = & (\{2,3,4,5\}, \\ & & \{[2,0,0,1]^T/\sqrt{(}5), [0,1,0,0]^T\}) \\ S'_2 & = & (\{5,6,7\}, \\ & & \{[0,0,1,0]^T, [-1,0,0,2]^T/\sqrt{5}\}) \end{array}$$

We immediately observe that the cluster sizes are $m_1 = 3 \cdot 1 = 3$, $m_2 = 3 \cdot 3 = 9$, $m'_1 = 4 \cdot 2 = 8$, and $m'_2 = 3 \cdot 2 = 6$.

We write the orthonormal basis vectors in the matrices Q_{S_1}, Q_{S_2}, $Q_{S'_1}$, and $Q_{S'_2}$. We need these matrices for calculating the principal angles between the subspaces. For instance, the principal angle $\theta^1_{S_1,S'_1}$ is the only non-zero singular value of the matrix $Q^T_{S_1} Q_{S'_1}$. The principal angles are $\theta^1_{S_1,S'_1} = 0.32$ (18.44°), $\theta^1_{S_1,S'_2} = 1.25$ (71.57°), $\theta^1_{S_2,S'_1} = 0$ (0°), $\theta^2_{S_2,S'_1} = 0.89$ (50.77°), $\theta^1_{S_2,S'_2} = 0$ (0°), and $\theta^2_{S_2,S'_2} = 0.69$ (39.23°).

These numbers allow us to compute the intersections between the cluster pairs. For instance, for S_1 and S'_1 we get the intersection size $|I|_{S_1,S'_1} = 2 \cdot \cos(0.32) = 1.90$. Similarly, $|I|_{S_1,S'_2} = 0$, $|I|_{S_2,S'_1} = 1.63$, and $|I|_{S_2,S'_2} = 5.31$. The union area of the two clusterings is $|U| = \sum_i m_i + \sum_j m'_j - \sum_{i,j} |I|_{S_i,S'_j} = 26 - 8.84 = 17.16$.

Finally, $\text{RNIA}(\mathcal{S}, \mathcal{S}') = (17.16 - 8.84)/17.16 = 0.48$ and $\text{CE}(\mathcal{S}, \mathcal{S}') = (17.16 - 7.21)/17.16 = 0.58$.

4.2 Comparing Attribute Weighted Clusterings

An *attribute weighted cluster* S is a pair (R, b), where $R \subseteq \{r_1, r_2, \ldots, r_m\}$ is a subset of the data points and b is a vector $[b_1, b_2, \ldots, b_p]^T$, where $b_i \geq 0$ and $\sum_{i=1}^p b_i = 1$. The vector b defines an importance weight for each column (attribute). An *attribute weighted clustering* $\mathcal{S}$ is a collection of K attribute weighted clusters $\{S_1, S_2, \ldots, S_K\}$. The algorithms COSA [23] and LAC [19] produce this type of clusterings.

We can easily utilize CE and RNIA to compare attribute weighted clusterings if we first define the sizes, the unions and the intersections for this kind of clusterings. We define the size of the cluster (R, b) as $|R|$, the number of data points in the cluster. To define the intersection of two clusters, we first need the intersection of two attribute weight vectors. We define this as

$$b_1 \cap b_2 = 1 - \frac{1}{2}\sum_{i=1}^{p} |b_{1i} - b_{2i}|. \tag{11}$$

Note that this could be defined in various ways; The present definition corresponds to the *variation distance* commonly used with discrete probability distributions [5].

It always holds that $0 \leq b_1 \cap b_2 \leq 1$. Now the intersection of two clusters (R_1, b_1) and (R_2, b_2) is given as $|R_1 \cap R_2| \cdot |b_1 \cap b_2|$. To bound the sum of the intersection sizes, we require that if two clusters of a clustering share data points, the inner product of the associated attribute weight vectors has to be zero. We previously introduced an analogous definition for non-axis-aligned subspace clusters.

4.3 Comparing Co-Clusterings

Recall that a *co-clustering* $\mathcal{S} = (\mathcal{R}, \mathcal{C})$ is a simultaneous partitioning of the rows and the columns of the data matrix; $\mathcal{R} = \{R_1, R_2, \ldots, R_L\}$ denotes the collection of row clusters and $\mathcal{C} = \{C_1, C_2, \ldots, C_M\}$ the collection of column clusters.

Since co-clusterings are always partitionings of the data matrix elements, we can straightforwardly use any of the ordinary clustering distance measures of Section 2. For instance, we simply write $\mathrm{VI}(\mathcal{S}, \mathcal{S}')$ for the VI distance between two co-clusterings $\mathcal{S}$ and $\mathcal{S}'$. In calculating the VI (or any other distance) for co-clusterings, we consider each data matrix element as a data point with a cluster label, and the co-clustering as a partition of the data matrix elements.

It is possible to derive relationships for the distances between two co-clusterings and their corresponding row and column clusterings. For instance, the following propositions hold. The proofs are presented in the Appendix.

Proposition 2. *For all co-clusterings $\mathcal{S}, \mathcal{S}'$ we have $VI(\mathcal{S}, \mathcal{S}') = VI(\mathcal{R}, \mathcal{R}') + VI(\mathcal{C}, \mathcal{C}')$.*

Proposition 3. *For all co-clusterings $\mathcal{S}, \mathcal{S}'$ we have $CE(\mathcal{S}, \mathcal{S}') \geq CE(\mathcal{R}, \mathcal{R}') + CE(\mathcal{C}, \mathcal{C}') - CE(\mathcal{R}, \mathcal{R}')CE(\mathcal{C}, \mathcal{C}')$.*

Proposition 4. *For all co-clusterings $\mathcal{S}, \mathcal{S}'$ we have $RNIA(\mathcal{S}, \mathcal{S}') = 0$.*

4.3.1 Example of Co-Clusterings

Consider two co-clusterings $\mathcal{S}$ and $\mathcal{S}'$. We have $R = \{\{1, 2, 3, 4\}, \{5, 6, 7\}, \{8\}\}$,
$C = \{\{1, 2, 3\}, \{4, 5, 6\}, \{7, 8\}$, $R' = \{\{1, 2, 4\}, \{3, 6, 7\}, \{5, 8\}\}$, and
$C' = \{\{1, 2, 5, 6\}, \{3, 4, 7\}, \{8\}\}$.

The VI distances are $\mathrm{VI}(\mathcal{R}, \mathcal{R}') = 0.93$, $\mathrm{VI}(\mathcal{C}, \mathcal{C}') = 1.41$, and $\mathrm{VI}(\mathcal{S}, \mathcal{S}') = \mathrm{VI}(\mathcal{R}, \mathcal{R}') + \mathrm{VI}(\mathcal{C}, \mathcal{C}') = 2.34$. The CE distances are $\mathrm{CE}(\mathcal{R}, \mathcal{R}') = 2/8$, $\mathrm{CE}(\mathcal{C}, \mathcal{C}') = 4/8$, and $\mathrm{CE}(\mathcal{S}, \mathcal{S}') = \mathrm{CE}(\mathcal{R}, \mathcal{R}') + \mathrm{CE}(\mathcal{C}, \mathcal{C}') - \mathrm{CE}(\mathcal{R}, \mathcal{R}')\mathrm{CE}(\mathcal{C}, \mathcal{C}') = 40/64$. All RNIA distances are zero.

5 Experimental Results

We now demonstrate how our distance measures can be used to calculating distances between clusterings produced by a variety of subspace clustering algorithms on synthetic data sets. In our first experiment, detailed in Section 5.1, we compare the algorithms by the means of external cluster validation. We also show how the use of the subspace clustering distance measures can give more information than traditional row-based or column-based comparison approaches. In Section 5.2, we describe an experiment illustrating how subspace clustering distance measures can be used for studying the stability of various clusterings produced by a single algorithm on a given data set. These results indicate that our distance measures are useful for internal cluster validation. As our third experiment, described in Section 5.3, we show how our distance measures can be used to compare patterns in binary data matrices.

5.1 External Cluster Validation

5.1.1 Data Sets and Algorithms

We compare the performance of four algorithms, PROCLUS [2], FASTDOC [48], HARP [62], and ORCLUS [3] on synthetic data sets. The first three algorithms produce axis-aligned subspace clusterings, and ORCLUS produces non-axis-aligned clusterings. We compare clusterings produced by these algorithms using our extended CE and RNIA distance measures, which were the two candidates possessing the most desirable theoretical properties (see Section 3.3.5 for details).

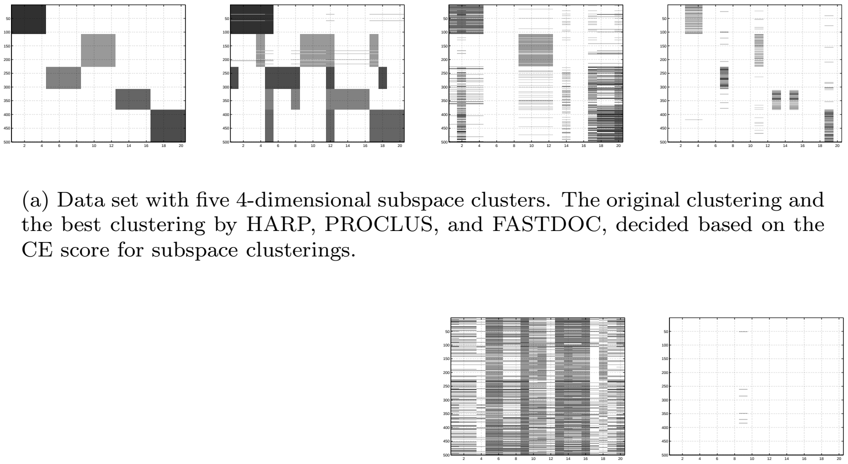
We use the clustering results from [61, 62] that compared the algorithms across 8 synthetic data sets. Each data set has 500 data points, 20 attributes, and 5 axis-aligned subspace clusters. The corresponding row clusters form a partition of the data points (rows). The number of data points in each cluster varies from 15% to 25% of the total number of points. The 8 data sets differ in the dimensionality of the subspace clusters. In the first data set, the dimensionality of all subspace clusters is 4, in the second data set it is 6, and finally in the 8th data set, the subspaces are 18-dimensional. In an attribute relevant to a subspace, the standard deviation of the within-cluster data is between 3% and 5% of the global standard deviation on that attribute.[10] No noise is added. For each data set, we have several clustering results for each algorithm corresponding to various parameter values (except for HARP, which is deterministic and has no input parameters).

5.1.2 Qualitative Comparison of the Algorithms

Some of the clustering results by HARP, PROCLUS, and FASTDOC are visualized in Fig. 3 together with the original clustering.[11] To illustrate the full range

[10] Only the ratio of the standard deviations affects the performance of the algorithms; the magnitudes as such do not have an effect.

[11] Since ORCLUS produces non-axis-aligned clusters, its results cannot be visualized here.



(a) Data set with five 4-dimensional subspace clusters. The original clustering and the best clustering by HARP, PROCLUS, and FASTDOC, decided based on the CE score for subspace clusterings.

(b) Data set with five 4-dimensional subspace clusters. The worst clustering by PROCLUS and FASTDOC, decided based on the CE score for subspace clusterings.

Figure 3: Various subspace clusterings for a data set with 500 data points, 20 dimensions, and five 4-dimensional subspace clusters. Each small picture illustrates a subspace clustering. Each subspace cluster is represented by a different shade of gray (the colors do not imply correspondence between clusters of different clusterings), and the unclustered background is white. The definition of the 'best' clustering depends on the choice of the distance for PROCLUS and FASTDOC. We only have one HARP clustering, since HARP is a deterministic algorithm without input parameters.

of clusterings that we analyzed, we have chosen to plot the best and the worst clusterings produced by each algorithm. We only consider CE distances here, since the RNIA results are very similar. The output clusters by the various algorithms are clearly different from each other and the dependence of PROCLUS and FASTDOC on the parameter values is strong. Visual inspection of the clustering results on other data sets supports these observations.

5.1.3 Comparing Subspace Clusterings, Row Clusterings, and Column Clusterings

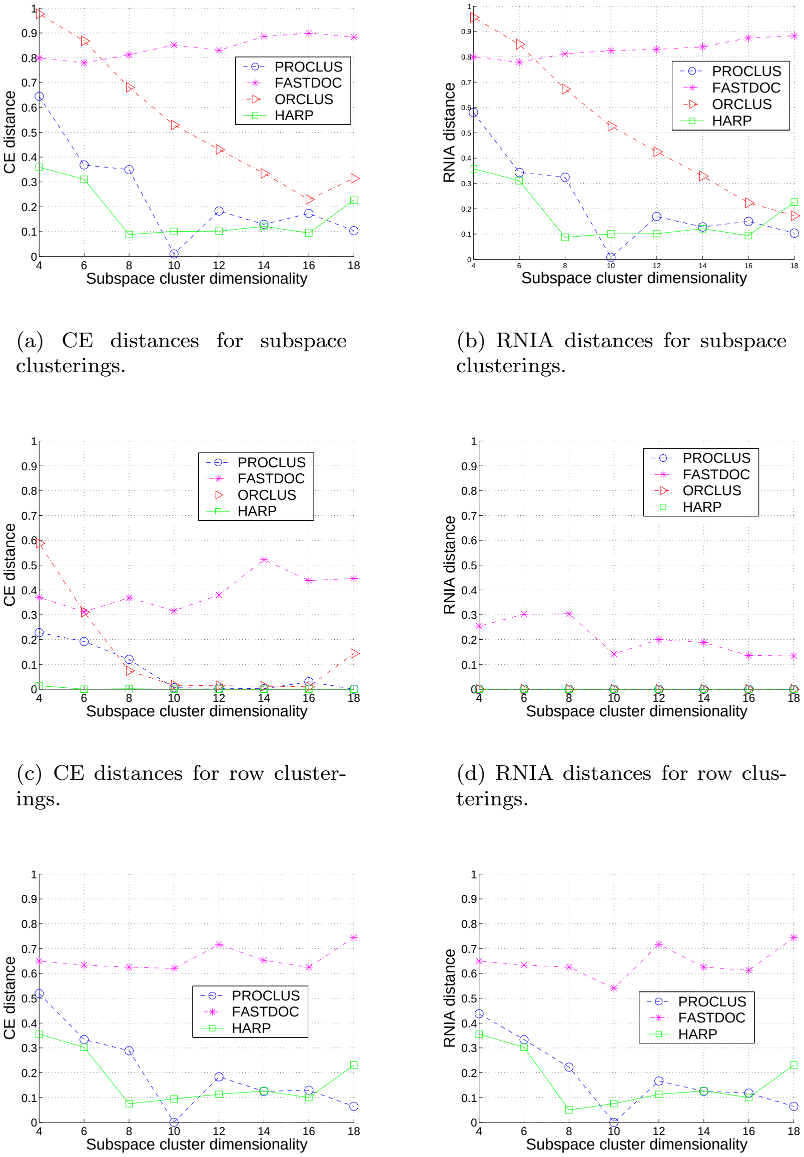
Let us now consider the difference between the CE distance for subspace clusterings, row clusterings, and column clusterings. Fig. 4 shows a comparison between the algorithms on all data sets using six different distance measures.[12] In the first column, we have CE distance for subspace clusterings, CE distance using row information only, and CE distance using column information only.[13] In the second column, the corresponding results for RNIA are shown. Note that the column clusterings are non-disjoint, so we need to apply the element duplication procedure from Section 3.4 here.

The figure clearly brings out differences between the subspace clustering distances, the row clustering distances, and the column clustering distances, indicating that it is indeed worthwhile to pay attention to the choice of the clustering type to compare. For instance, according to the row clustering results, HARP performs well for all data sets and always gives the best result. This is somewhat misleading, since the other two distance measures reveal that HARP's choice of subspaces leaves room for improvement. These results show that, irrespective of whether we wish to compare algorithms to each other or analyze the performance of a given algorithm across data sets, the row, column, or subspace based distance measures give different information.

The results of CE and RNIA are very similar on the subspace clustering distances (top row of Fig. 4) and on the column clustering distances (bottom row of Fig. 4). However, the RNIA scores for PROCLUS, ORCLUS, and HARP are zero when the row clustering distances are computed. This is natural, since all three algorithms give a partition of the full set of data points, and since the original clustering contains a partition of the same set, we do not have any non-intersecting area between the clusterings. On the other hand, FASTDOC does not produce a partition of the set of data points, which is why its RNIA scores are non-zero.

[12] We have chosen to plot only the best clustering results of each algorithm, since the dependence on parameter values is strong, and it does not make sense in this case to plot the means with error bars.

[13] Since we do not yet have a method for handling non-disjoint non-axis-aligned clusterings, the column distance measure for ORCLUS is not shown.



(a) CE distances for subspace clusterings.

(b) RNIA distances for subspace clusterings.

(c) CE distances for row clusterings.

(d) RNIA distances for row clusterings.

(e) CE distances for column clusterings.

(f) RNIA distances for column clusterings.

Figure 4: Distances between PROCLUS output and the true clustering, FastDOC output and the true clustering, etc., for eight different data sets (cluster dimensionalities $4, 6, \ldots, 18$). Only the best clustering results of each algorithm are shown. Since we do not yet have a method for handling non-disjoint non-axis-aligned clusterings, the column distance for ORCLUS is not shown.

5.2 Internal Cluster Validation

5.2.1 Data Sets and Algorithms

We conducted another experiment on a new data set consisting of 1000 rows, 100 columns, and 5 subspace clusters. The goal here is to compare the performance of PROCLUS and SSPC [63] over samples of this data set with various parameter settings. Our non-disjoint axis-aligned subspace clusters are 10-dimensional and each cluster has approximately 200 rows. The row clusters form a partition of the set of all rows. The standard deviation of the within-cluster data is between 3% and 5% of the global standard deviation. Five samples of this data set were created by removing 10% of the rows and 10% of the columns.[14] These 5 samples were then clustered by PROCLUS and SSPC. PROCLUS was run with 9 different parameter values for each sample, resulting in 45 clusterings, and SSPC was run with 10 different parameter values per sample, resulting in 50 clusterings.

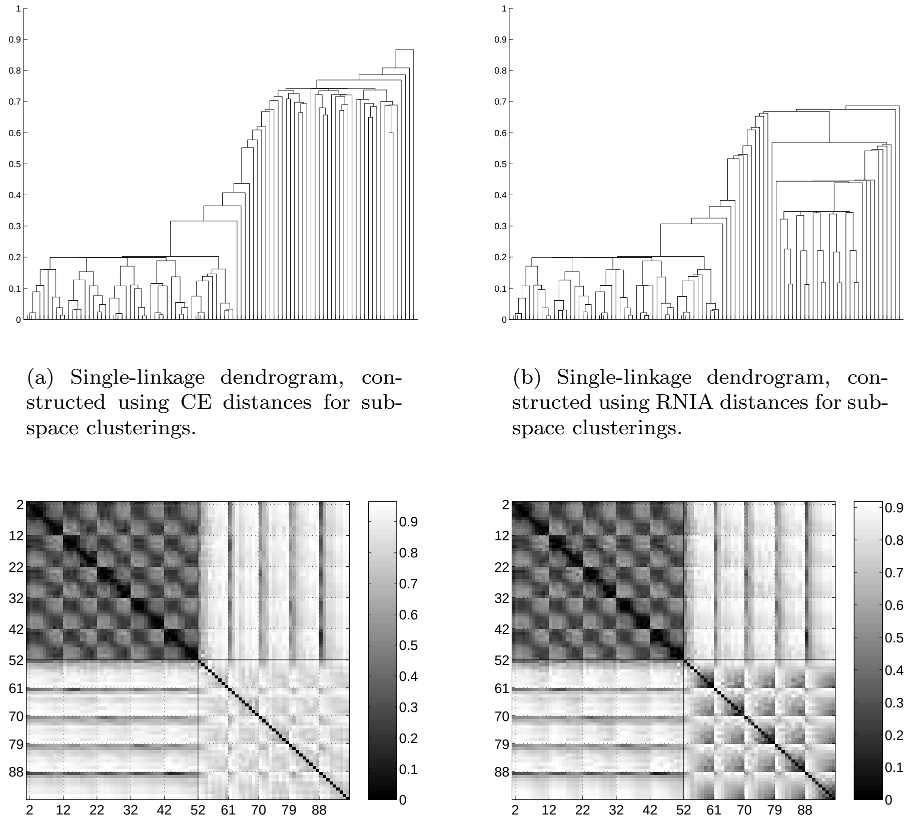
5.2.2 Results

We have computed the pairwise distances for all 96 clusterings (the true clustering, 50 SSPC clusterings, and 45 PROCLUS clusterings) using the subspace clustering CE distance and the subspace clustering RNIA distance. Fig. 5 (a) shows a single-linkage dendrogram produced by an agglomerative hierarchical clustering algorithm based on the subspace clustering CE distances between all pairs of clusterings. The five groups on the left correspond to the SSPC clusterings on the five samples; the true clustering is included in the fifth group. The PROCLUS clusterings lie on the right-hand side of the dendrogram and do not seem to contain any clear structure.

The subspace CE distance matrix in Fig. 5 (c) supports our findings. Clustering 1 is the true clustering, clusterings 2–51 are the SSPC clusterings, and clusterings 52–96 are the PROCLUS clusterings. The SSPC and the PROCLUS clusterings are separated by vertical and horizontal lines. The clusterings are ordered by sample: the clusterings 2–11 are the SSPC clusterings for the first sample, the clusterings 12–21 are the SSPC clusterings for the second sample, and so on. Similarly, the clusterings 52–60 are the PROCLUS clusterings for the first sample, the clusterings 61–69 are the PROCLUS clusterings for the second sample, etc. It is clear that the SSPC clusterings are much closer to each other than the PROCLUS clusterings, which do not exhibit clear clustering structure. In each sample, the SSPC clusterings are clustered into two groups by parameter value, as the dendrogram of Fig. 5 (a) also shows.

The right column of Fig. 5 shows similar results using the subspace RNIA distance. As the dendrogram in Fig. 5 (b) illustrates, the RNIA results differ from the CE results. As before, the five clusters on the left-hand side of the dendrogram correspond to the SSPC clusterings for the five samples. However,

[14] An extensive series of experiments would be needed to determine a good sample size and to see whether sampling both the rows and the columns is necessary. In this article, we present only preliminary experiments on internal cluster validation.



(a) Single-linkage dendrogram, constructed using CE distances for subspace clusterings.

(b) Single-linkage dendrogram, constructed using RNIA distances for subspace clusterings.

(c) Distance matrix ordered by sample, constructed using CE distances for subspace clusterings.

(d) Distance matrix ordered by sample, constructed using RNIA distances for subspace clusterings.

Figure 5: Representations of the pairwise distances for 50 clusterings by SSPC (10 parameter values for each of the 5 samples) and 45 clusterings by PROCLUS (9 parameter values for each of the 5 samples). In the dendrograms, the closely grouped clusterings on the left are produced by SSPC, and the more loosely grouped clusterings on the right are produced by PROCLUS. In the distance matrices, 1: True clustering. 2–51: SSPC clusterings. 52–96: PROCLUS clusterings. The horizontal/vertical lines mark the borders between the SSPC clusterings and the PROCLUS clusterings.

now also some of the PROCLUS results seem to exhibit clustering structure; five groups of four PROCLUS clusterings each are visible.

The distance matrix in Fig. 5 (d) brings more light in the situation. The five groups correspond to the five samples, and in each sample, the clusterings corresponding to the four last parameter values are clustered together. This is easy to explain. The parameter value required by PROCLUS is the average dimensionality of the output subspace clusters. The parameter values 6-9 correspond to higher dimensionalities 60-90, and if the clusters are so high-dimensional as to fill almost the whole 100-dimensional data matrix, they are bound to be close to each other in the RNIA sense; the relative overlap is very high, no matter how bad the clusters are. However, this undesirable phenomenon does not show in the CE results, since CE requires a one-to-one matching between the clusters.

This experiment demonstrates how subspace clustering distance measures could be used for stability-based internal cluster validation for subspace clusterings. We have shown that omputing pairwise distances between clusterings is able to provide us infromation on the stability of algorithms. Based on our experiments, it is clear that SSPC is more stable algorithm than PROCLUS, since the SSPC results vary less across samples. It is important to note that the choice of the subspace clustering distance measure does matter: CE is a better choice than RNIA in the case of high-dimensional subspace clusters.

We have further shown that subspace clustering distance measures are useful in other ways; noticing that the SSPC clusterings corresponding to various parameter values fall into two groups gives us more information on how the SSPC algorithm works; this is an example of meta-clustering.

5.3 Patterns in a Binary Data Set

We will now demonstrate that our distance measures are not limited to comparing subspace clusterings but can be used for other important tasks as well. As an example, let us consider comparing patterns in sparse binary data matrices. This kinds of matrices are commonly encountered in data mining; examples include market basket data, web log data, and text data in bag-of-words format. Ways to represent patterns in binary matrices include error-tolerant frequent itemsets [59], dense itemsets [51], geometric and combinatorial tiles [25], alpha/beta concepts [10], and conjunctive clusters [40].

Some of the pattern representations listed above can be directly viewed as axis-aligned-subspace clusterings and compared as such. Many others can be straightforwardly converted into axis-aligned subspace clusterings. To demonstrate this, we have computed geometric tiles, error-tolerant frequent itemsets (ETIs), and dense itemsets for a newsgroup posting data set. Geometric tiles are just rectangular sets of data matrix elements, so we do not have to do anything to convert these into subspace clusters.[15] However, error-tolerant frequent itemsets and dense itemsets are attribute sets. In these cases, we can quite naturally compute the set of rows for which a given itemset is frequent/dense. This

[15] In our experiments, we leave out the background tile which covers the whole data matrix; see the original paper for details.

way, we are able to derive a rectangular set of matrix elements corresponding to each itemset.

Our data set, which has been originally used in [30], consists of 348 newsgroup postings and 16 terms. The entry (i, j) in the data matrix has a value '0' or '1' depending on whether the jth term appears in the ith newsgroup posting. The postings are collected from 4 newsgroups whose topics are religion, cryptography, medicine, and space. The terms are chosen to reflect the topics: they are 'god', 'christ', 'bibl', 'church', 'secur', 'kei', 'encrypt', 'public', 'effet', 'medic', 'patient', 'doctor', 'space', 'nasa', 'orbit', and 'launch'. See Fig. 6 (a) for a visualization of the data set. We have chosen a small data set because the dense itemsets algorithm and the basic version of the error-tolerant frequent itemsets algorithm are exponential with respect to the number of attributes.

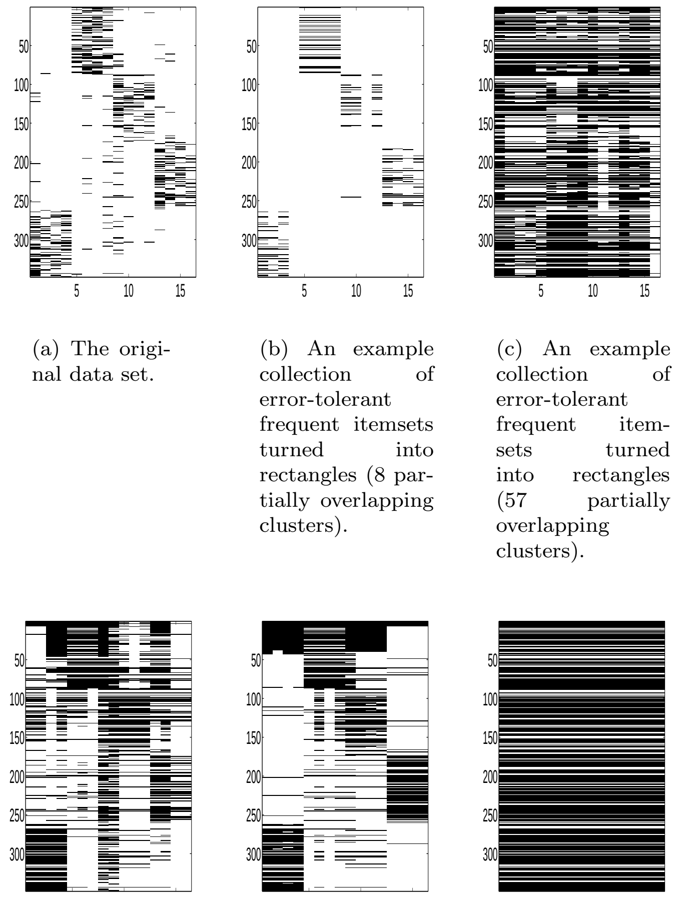
We have computed geometric tiles for the newsgroup data set with 9 different parameter values, error-tolerant frequent itemsets with 15 parameter values, and dense itemsets with 20 parameter values. We then converted these results into collections of rectangular sets of data matrix elements; we will refer to these rectangular sets as clusters and the collections of these sets as clusterings. Some examples of the results are visualized in Fig. 6. The examples show that the dependence on parameter values is significant for ETIs and dense itemsets. It is therefore important to be able to compare these results quantitatively. This way, we are able to see which algorithms are stable, which parameter values produce similar results, and which algorithms resemble each other.

Fig. 7 shows the pairwise CE and RNIA distances for all pairs of clusterings. The CE and RNIA distances behave in a similar way, even though RNIA places the clusterings closer to each other, as expected. The results for dense itemsets form a few clear clusters, the results for geometric tiles are all very close to each other, and the results for error-tolerant frequent itemsets vary a lot with respect to the parameter value. The tilings are closer to the dense itemsets than the ETIs.

6 Conclusion

In this article, we have addressed the problem of comparing subspace clusterings. We have done a comprehensive literature survey on subspace clustering articles and observed that there is currently no satisfactory way to compare subspace clusterings. We have motivated our work by arguing that comparing clusterings is of crucial importance in external and internal cluster validation, meta-clustering, and consensus clusterings.

Since comparing subspace clusterings is more general task than comparing ordinary clusterings, we have introduced a set of theoretical properties important for a subspace clustering distance. We have introduced four candidates for comparing subspace clusterings, namely CE, RNIA, VI, and Rand, and characterized them in terms of their theoretical properties. CE, VI, and Rand are generalizations of existing methods for comparing ordinary clusterings, and RNIA is a novel retrieval measure.



(a) The original data set.

(b) An example collection of error-tolerant frequent itemsets turned into rectangles (8 partially overlapping clusters).

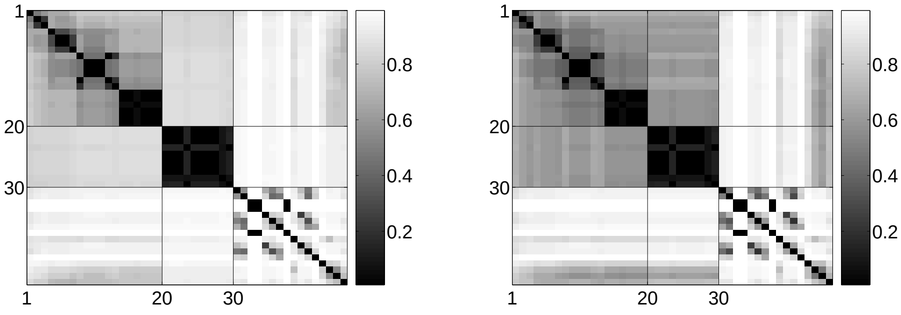
(c) An example collection of error-tolerant frequent itemsets turned into rectangles (57 partially overlapping clusters).

(d) An example collection of dense itemsets turned into rectangles (13 partially overlapping clusters).

(e) An example collection of dense itemsets turned into rectangles (7 partially overlapping clusters).

(f) An example collection of geometric tiles (10 partially overlapping clusters).

Figure 6: Some example results by our algorithms for the newsgroup data set. The binary data matrix is of size 348 newsgroup postings × 16 terms; the '1's are colored black. This simple visualization does not show the difference between the clusters or the overlap of the clusters.



(a) Distance matrix constructed using subspace CE as the distance measure.

(b) Distance matrix constructed using RNIA as the distance measure.

Figure 7: Representations of the pairwise distances for dense itemsets (1-20), tilings (21-30), and error-tolerant frequent itemsets (31-46) for the newsgroup data. The horizontal/vertical lines denote the borders between these three categories of results.

Out of these four distance measure candidates, we have chosen to use CE and RNIA in our experiments, since these two measures possess the most desirable theoretical properties. In the experiments, we have compared clusterings by five well-known algorithms: FASTDOC, HARP, PROCLUS, ORCLUS, and SSPC. We have demonstrated how our measures can be used in both external and internal cluster validation. We have also shown that comparing subspace clusterings gives different information than comparing the corresponding row and column clusterings. Further, we have noted that comparing the row or the column clusterings, which might not be partitions of sets of all rows/columns, is often not even possible with the ordinary clustering comparison measures. Our experiments have demonstrated that CE is a better choice for a distance measure than RNIA in the case of high-dimensional subspace clusters.

It turns out that the distance measures we have proposed for axis-aligned subspace clusterings are useful for comparing other types of clusterings also. In designing the distance measures, we have not restricted ourselves in any way to consider only rectangular sets of matrix elements as clusters. Hence, our distance measures are applicable to any *partial clusterings*: clusterings on subsets of data points. We are not aware of any existing methods for comparing partial clusterings. These kinds of clusterings commonly arise in stability-based internal cluster validation, where we want to compare clusterings on samples of the data. Most previous approaches have compared the clusterings only at their intersection [9]. Partial clusterings might also arise in distributed databases [52].

Our distance measures could also be useful for comparing hierarchical clusterings. Comparing hierarchical clusterings is interesting, since hierarchies are commonly used in bioinformatics for instance to represent the evolutionary relations of proteins [32]. Also, the subspace clustering algorithms HARP and COSA produce hierarchical clusterings; it would be exciting to see how these clusterings compare to ordinary hierarchical clusterings. Hierarchical clusterings are commonly described by dendrograms, which in turn can be viewed as non-disjoint clusterings, i.e., clusterings in which a given data point may belong to several clusters. More specifically, a hierarchical clustering for N data points is a collection of N partitions of these N points; each data point thus belongs to N clusters simultaneously. We are currently studying the properties of our distance measures in comparing hierarchical clusterings.

We have demonstrated that in addition to comparing clusterings, our distance measures can be used to compare certain types of patterns in data matrices. We have shown comparisons for geometric tiles, dense itemsets, and error-tolerant frequent itemsets for a binary data matrix. Our distance measures provide a way to derive results on the stability and similarity of algorithms that produce this kinds of patterns.

There are still more avenues to be explored in the future. Weighting the rows and the columns of the data matrix is another potentially useful feature. We have only discussed hard clusterings; probabilistic subspace clusterings would require separate analysis.[16] Finally, depending on the definition of the closeness of the data points in the subspace (for instance, distance-based [2] or pattern-based [60]), the rows and the columns of the data matrix may or may not be symmetric, and a successful comparison method should take this into account.

Appendix 1: Proofs for Table 1

6.1 Triangle Inequality

Theorem 2. *(Triangle inequality for CE.) $CE(\mathcal{A},\mathcal{B}) \leq CE(\mathcal{A},\mathcal{C}) + CE(\mathcal{B},\mathcal{C})$ for any subspace clusterings $\mathcal{A},\mathcal{B},\mathcal{C}$.*

To prove this theorem, we first show some preliminary results. For simplicity, we adopt a shorthand notation and write $\mathcal{A}$ instead of $\operatorname{supp}(\mathcal{A})$, $\mathcal{A}\backslash\mathcal{B}$ instead of $\operatorname{supp}(\mathcal{A}) \setminus \operatorname{supp}(\mathcal{B})$, etc. Also, we write $\mathcal{A}_{\mathcal{A}\cap\mathcal{B}}$ for the part of the clustering $\mathcal{A}$ in $\mathcal{A}\cap\mathcal{B}$, $\mathcal{A}_{\mathcal{A}\backslash\mathcal{B}}$ for the part of the clustering $\mathcal{A}$ in $\mathcal{A}\backslash\mathcal{B}$, etc. Also, let us define a *cluster label vector* u for a clustering of mp points and K clusters as a vector of size $mp \times 1$ where $u_i = k$ if the ith point belongs to the kth cluster. Here $i \in \{1, 2, \ldots, mp\}$ and $k \in \{1, 2, \ldots, K\}$.

Proposition 5. *If $\mathcal{A}$ and $\mathcal{B}$ are arbitrary subspace clusterings and $\mathcal{C}' \subseteq (\mathcal{A}\cup\mathcal{B})$, then $CE(\mathcal{A},\mathcal{B}) \leq CE(\mathcal{A},\mathcal{C}') + CE(\mathcal{B},\mathcal{C}')$.*

Proof. (Proposition 5) Let us write $\mathrm{H}(u, v)$ for the Hamming distance between two cluster label vectors u and v, or in other words, the total number of differ-

[16] However, we are only aware of one algorithm producing probabilistic subspace clusterings [46].

ences between these two vectors. Then the CE distance becomes

$$\mathcal{C}E(\mathcal{A},\mathcal{B}) = \\ \frac{\min \mathrm{H}(\mathcal{A}_{\mathcal{A}\cap\mathcal{B}},\mathcal{B}_{\mathcal{A}\cap\mathcal{B}}) + |\mathcal{A}\backslash\mathcal{B}| + |\mathcal{B}\backslash\mathcal{A}|}{|\mathcal{A}\cup\mathcal{B}|},$$

where the minimum is taken over all permutations of the cluster labels. Let us consider three subspace clusterings $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}'$ such that $\mathcal{A}$ and $\mathcal{B}$ are arbitrary and $\mathcal{C}' \subseteq (\mathcal{A}\cup\mathcal{B})$. We consider $\mathcal{C}'$ in three disjoint parts: $\mathcal{C}' = \mathcal{C}'_{\mathcal{A}\cap\mathcal{B}}\cup\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}\cup\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}$. Let us fix the cluster labels of $\mathcal{C}'$ and choose the permutation of labels in $\mathcal{A}$ to minimize $\mathrm{H}(\mathcal{A}_{\mathcal{A}\cap\mathcal{C}'},\mathcal{C}'_{\mathcal{A}\cap\mathcal{C}'})$ and the permutation of labels in $\mathcal{B}$ to minimize $\mathrm{H}(\mathcal{B}_{\mathcal{B}\cap\mathcal{C}'},\mathcal{C}'_{\mathcal{B}\cap\mathcal{C}'})$. Using these labels, we have

$$\begin{aligned}
&\mathrm{CE}(\mathcal{A},\mathcal{C}') + \mathrm{CE}(\mathcal{B},\mathcal{C}') - \mathrm{CE}(\mathcal{A},\mathcal{B}) \\
&\quad\geq \frac{\mathrm{H}(\mathcal{A}_{C'_{\mathcal{A}\cap\mathcal{B}}},\mathcal{C}'_{A\cap B}) + \mathrm{H}(\mathcal{A}_{C'_{\mathcal{A}\backslash\mathcal{B}}},\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}})}{|\mathcal{A}\cup\mathcal{C}'|} \\
&\qquad+ \frac{|\mathcal{A}| - |\mathcal{C}'_{\mathcal{A}\cap\mathcal{B}}| - |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}| + |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}|}{|\mathcal{A}\cup\mathcal{C}'|} \\
&\qquad+ \frac{\mathrm{H}(\mathcal{B}_{C'_{\mathcal{A}\cap\mathcal{B}}},\mathcal{C}'_{\mathcal{A}\cap\mathcal{B}}) + \mathrm{H}(\mathcal{B}_{C'_{\mathcal{B}\backslash\mathcal{A}}},\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}})}{|\mathcal{B}\cup\mathcal{C}'|} \\
&\qquad+ \frac{|\mathcal{B}| - |\mathcal{C}'_{\mathcal{A}\cap\mathcal{B}}| - |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}| + |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}|}{|\mathcal{B}\cup\mathcal{C}'|} \\
&\qquad- \frac{\mathrm{H}(\mathcal{A}_{\mathcal{A}\cap\mathcal{B}},\mathcal{B}_{\mathcal{A}\cap\mathcal{B}}) + |\mathcal{A}\backslash\mathcal{B}| + |\mathcal{B}\backslash\mathcal{A}|}{|\mathcal{A}\cup\mathcal{B}|}.
\end{aligned}$$

Above, the first two terms correspond to $\mathrm{CE}(\mathcal{A},\mathcal{C}')$ and the second two terms correspond to $\mathrm{CE}(\mathcal{B},\mathcal{C}')$. Due to the choice of the labels, the quantity in the third term is greater than or equal to $\mathrm{CE}(\mathcal{A},\mathcal{B})$, hence the inequality.

Next, we notice that $|\mathcal{A}| = |\mathcal{A}\cap\mathcal{B}| + |\mathcal{A}\backslash\mathcal{B}|$, that $|\mathcal{B}| = |\mathcal{A}\cap\mathcal{B}| + |\mathcal{B}\backslash\mathcal{A}|$, and that $\mathrm{H}(\mathcal{A}_{\mathcal{A}\cap\mathcal{B}},\mathcal{B}_{\mathcal{A}\cap\mathcal{B}}) \leq |\mathcal{A}\cap\mathcal{B}| + \mathrm{H}(\mathcal{A}_{C'_{\mathcal{A}\cap\mathcal{B}}},\mathcal{B}_{C'_{\mathcal{A}\cap\mathcal{B}}}) - |\mathcal{C}'_{\mathcal{A}\cap\mathcal{B}}|$. We substitute these in the above equation, rearrange the terms, and obtain

$$\begin{aligned}
&\mathrm{CE}(\mathcal{A},\mathcal{C}') + \mathrm{CE}(\mathcal{B},\mathcal{C}') - \mathrm{CE}(\mathcal{A},\mathcal{B}) \\
&\geq \frac{\mathrm{H}(\mathcal{A}_{C'_{\mathcal{A}\cap\mathcal{B}}},\mathcal{C}'_{A\cap B})}{|\mathcal{A}\cup\mathcal{C}'|} + \frac{\mathrm{H}(\mathcal{B}_{C'_{\mathcal{A}\cap\mathcal{B}}},\mathcal{C}'_{A\cap B})}{|\mathcal{B}\cup\mathcal{C}'|} \\
&\quad - \frac{\mathrm{H}(\mathcal{A}_{C'_{\mathcal{A}\cap\mathcal{B}}},\mathcal{B}_{C'_{\mathcal{A}\cap B}})}{|\mathcal{A}\cup\mathcal{B}|} \\
&\quad + \frac{\mathrm{H}(\mathcal{A}_{C'_{\mathcal{A}\setminus\mathcal{B}}},\mathcal{C}'_{\mathcal{A}\setminus\mathcal{B}})}{|\mathcal{A}\cup\mathcal{C}'|} + \frac{\mathrm{H}(\mathcal{B}_{C'_{\mathcal{B}\setminus\mathcal{A}}},\mathcal{C}'_{\mathcal{B}\setminus\mathcal{A}})}{|\mathcal{B}\cup\mathcal{C}'|} \\
&\quad + \frac{(|\mathcal{A}\cap\mathcal{B}| - |\mathcal{C}'_{\mathcal{A}\cap\mathcal{B}}|) + (|\mathcal{A}\backslash\mathcal{B}| - |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}|) + |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}|}{|\mathcal{A}\cup\mathcal{C}'|} \\
&\quad + \frac{(|\mathcal{A}\cap\mathcal{B}| - |\mathcal{C}'_{\mathcal{A}\cap\mathcal{B}}|) + (|\mathcal{B}\backslash\mathcal{A}| - |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}|) + |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}|}{|\mathcal{B}\cup\mathcal{C}'|} \\
&\quad - \frac{|\mathcal{A}\backslash\mathcal{B}| + |\mathcal{B}\backslash\mathcal{A}| + (|\mathcal{A}\cap\mathcal{B}| - |C'_{\mathcal{A}\cap\mathcal{B}}|)}{|\mathcal{A}\cup\mathcal{B}|}.
\end{aligned}$$

We know that the triangle inequality holds for CE with ordinary clusterings (partitions), so in the equation above, the first three terms sum up to 0 or more. The Hamming distance is always nonnegative, so the fourth and the fifth terms above are also greater than or equal to 0. We also notice that $|\mathcal{A}\backslash\mathcal{B}| - |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}| \geq 0$ and $|\mathcal{B}\backslash\mathcal{A}| - |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}| \geq 0$. Finally, it holds that $|\mathcal{A}\cap\mathcal{B}| - |C'_{\mathcal{A}\cap\mathcal{B}}| \geq 0$. These observations lead us to

$$\begin{aligned}
&\mathrm{CE}(\mathcal{A},\mathcal{C}') + \mathrm{CE}(\mathcal{B},\mathcal{C}') - \mathrm{CE}(\mathcal{A},\mathcal{B}) \\
&\geq \frac{(|\mathcal{A}\backslash B| - |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}|) + |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}|}{|\mathcal{A}\cup\mathcal{C}'|} \\
&\quad + \frac{(|\mathcal{B}\backslash\mathcal{A}| - |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}|) + |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}|}{|\mathcal{B}\cup\mathcal{C}'|} \\
&\quad - \frac{|\mathcal{A}\backslash\mathcal{B}| + |\mathcal{B}\backslash\mathcal{A}|}{|\mathcal{A}\cup\mathcal{B}|}.
\end{aligned}$$

We lastly observe that $|A\cup\mathcal{B}| \geq |A\cup\mathcal{C}'|$ and that $|A\cup\mathcal{B}| \geq |B\cup\mathcal{C}'|$, which helps to complete the proof:

$$\begin{aligned}
&\mathrm{CE}(\mathcal{A},\mathcal{C}') + \mathrm{CE}(\mathcal{B},\mathcal{C}') - \mathrm{CE}(\mathcal{A},\mathcal{B}) \\
&\geq \frac{|\mathcal{A}\backslash B| - |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}| + |\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}| + |\mathcal{B}\backslash\mathcal{A}|}{|\mathcal{A}\cup\mathcal{B}|} \\
&\quad + \frac{-|\mathcal{C}'_{\mathcal{B}\backslash\mathcal{A}}| + |\mathcal{C}'_{\mathcal{A}\backslash\mathcal{B}}| - |\mathcal{A}\backslash\mathcal{B}| + |\mathcal{B}\backslash\mathcal{A}|}{|\mathcal{A}\cup\mathcal{B}|} \\
&= 0.
\end{aligned}$$

□

Proposition 6. *If $\mathcal{A}$ and $\mathcal{B}$ are arbitrary subspace clusterings and $\mathcal{C} = \mathcal{C}' \cup \mathcal{C}''$, where $\mathcal{C}' \subset (\mathcal{A} \cup \mathcal{B})$ and $\mathcal{C}'' \cap (\mathcal{A} \cup \mathcal{B}) = \emptyset$, then $CE(\mathcal{A}, \mathcal{C}) \geq CE(\mathcal{A}, \mathcal{C}')$ and $CE(\mathcal{B}, \mathcal{C}) \geq CE(\mathcal{B}, \mathcal{C}')$.*

Proof. (Proposition 6)

$$\begin{aligned}
&\mathcal{C}E(\mathcal{A}, \mathcal{C}) \\
&= \frac{\mathrm{H}(\mathcal{A}_{\mathcal{A}\cap\mathcal{C}'}, \mathcal{C}'_{\mathcal{A}\cap\mathcal{C}'}) + |\mathcal{A}\backslash\mathcal{C}'| + |\mathcal{C}'\backslash\mathcal{A}| + |\mathcal{C}''|}{|\mathcal{A} \cap \mathcal{C}'| + |\mathcal{A}\backslash\mathcal{C}'| + |\mathcal{C}'\backslash\mathcal{A}| + |\mathcal{C}''|} \\
&\geq \frac{\mathrm{H}(\mathcal{A}_{\mathcal{A}\cap\mathcal{C}'}, \mathcal{C}'_{\mathcal{A}\cap\mathcal{C}'}) + |\mathcal{A}\backslash\mathcal{C}'| + |\mathcal{C}'\backslash\mathcal{A}|}{|\mathcal{A} \cap \mathcal{C}'| + |\mathcal{A}\backslash\mathcal{C}'| + |\mathcal{C}'\backslash\mathcal{A}|} \\
&= \mathrm{CE}(\mathcal{A}, \mathcal{C}').
\end{aligned}$$

The case of $\mathcal{B}$ and $\mathcal{C}$ can be proven analogously. □

Proof. (Theorem 2) Let us choose arbitrary subspace clusterings $\mathcal{A}$, $\mathcal{B}$, and $\mathcal{C}$, for which $\mathcal{C} = \mathcal{C}' \cup \mathcal{C}''$ such that $\mathcal{C}' \subseteq (\mathcal{A} \cup \mathcal{B})$ and $\mathcal{C}'' \cap (\mathcal{A} \cup \mathcal{B}) = \emptyset$. By Propositions 5 and 6, $\mathrm{CE}(\mathcal{A}, \mathcal{C}) + \mathrm{CE}(\mathcal{B}, \mathcal{C}) \geq \mathrm{CE}(\mathcal{A}, \mathcal{C}') + \mathrm{CE}(\mathcal{B}, \mathcal{C}') \geq \mathrm{CE}(\mathcal{A}, \mathcal{B})$. □

Theorem 3. *(Triangle inequality for RNIA.) $RNIA(\mathcal{A}, \mathcal{B}) \leq RNIA(\mathcal{A}, \mathcal{C}) + RNIA(\mathcal{B}, \mathcal{C})$ for any subspace clusterings $\mathcal{A}, \mathcal{B}, \mathcal{C}$.*

Proof. (Theorem 3) Let us pick subspace clusterings $\mathcal{A}, \mathcal{B}, \mathcal{C}$ for a data matrix $X = (x_{ij})$. Let us write n_A for the number of the elements of X that are clustered only by $\mathcal{A}$, n_{AB} for the number of the elements of X that are clustered by $\mathcal{A}$ and $\mathcal{B}$ but not $\mathcal{C}$, and n_{ABC} for the number of the elements of X that are clustered by all three clusterings. We define n_B, n_C, n_{AC}, and n_{BC} similarly. Now, we can write

$$\begin{aligned}
&\mathrm{RNIA}(\mathcal{A}, \mathcal{C}) + \mathrm{RNIA}(\mathcal{B}, \mathcal{C}) - \mathrm{RNIA}(\mathcal{A}, \mathcal{B}) \\
&= \frac{n_A + n_C + n_{AB} + n_{BC}}{n_A + n_C + n_{AB} + n_{BC} + n_{AC} + n_{ABC}} \\
&+ \frac{n_B + n_C + n_{AC} + n_{AB}}{n_B + n_C + n_{AC} + n_{AB} + n_{BC} + n_{ABC}} \\
&- \frac{n_A + n_B + n_{AC} + n_{BC}}{n_A + n_B + n_{AC} + n_{BC} + n_{AB} + n_{ABC}}.
\end{aligned}$$

Once the expression is fully expanded, all negative terms disappear. Since the expression is always greater than zero, the triangle inequality holds. □

Example 1. *(Triangle inequality for VI.) VI does not satisfy the triangle inequality in the case of subspace clusterings.*

We show by counterexample that VI does not satisfy the triangle inequality. Consider three subspace clusterings $\mathcal{A} = (A_1) = (\{1, 2\}, \{1, 2, 3, 4\})$, $\mathcal{B} = (B_1) = (\{2, 3\}, \{1, 2, 3, 4\})$, and $\mathcal{C} = (C_1, C_2) = (\{1, 2\}, \{1, 2, 3, 4\}, \{1, 2\}, \{6\})$. We have $\mathrm{VI}(\mathcal{A}, \mathcal{B}) = 8/3 \log 2 \approx 1.84$, $\mathrm{VI}(\mathcal{A}, \mathcal{C}) = 1/5 \log 2 \approx 0.14$, and $\mathrm{VI}(\mathcal{B}, \mathcal{C}) = 17/7 \log 2 \approx 1.68$. Thus $\mathrm{VI}(\mathcal{A}, \mathcal{C}) + \mathrm{VI}(\mathcal{B}, \mathcal{C}) < \mathrm{VI}(\mathcal{A}, \mathcal{B})$, showing that the triangle inequality does not hold. It does not hold for these three clusterings even for VI distances scaled by the logarithm of the union area.

Example 2. *(Triangle inequality for Rand.) The Rand index does not satisfy the triangle inequality in the case of subspace clusterings.*

We show by counterexample that (1-Rand) does not satisfy the triangle inequality. Consider the example clusterings from Example 1. For these clusterings, we have $1-\text{Rand}(\mathcal{A},\mathcal{B}) = (22+22)/66 \approx 0.67$, $1-\text{Rand}(\mathcal{A},\mathcal{C}) = 1/45 \approx 0.02$, and $1-\text{Rand}(\mathcal{B},\mathcal{C}) = 45/91 \approx 0.49$. The triangle inequality does not hold, since $1-\text{Rand}(\mathcal{A},\mathcal{C}) + 1 - \text{Rand}(\mathcal{B},\mathcal{C}) < 1 - \text{Rand}(\mathcal{A},\mathcal{B})$.

6.2 Penalty for Non-Intersecting Area

Consider adding $k \geq 1$ units of non-intersecting area to two subspace clusterings $\mathcal{A}$ and $\mathcal{B}$ and denote the resulting clusterings by $\mathcal{A}^U$ and $\mathcal{B}^U$ (note that one of these might actually equal to the original clustering). Are our distance measures able to penalize for this added non-intersecting area?

Theorem 4. *(Penalty for Non-Intersecting Area with CE.) CE$(\mathcal{A}^U,\mathcal{B}^U) \geq$ CE$(\mathcal{A},\mathcal{B})$ for all subspace clusterings $\mathcal{A}$, $\mathcal{B}$.*

Proof.

$$\begin{aligned}
\text{CE}(\mathcal{A}^U,\mathcal{B}^U) &= \frac{(|U|+k) - D_{max}}{(|U|+k)} \\
&\geq \frac{|U| - D_{max}}{|U|} \\
&= \text{CE}(\mathcal{A},\mathcal{B}).
\end{aligned}$$

□

Theorem 5. *(Penalty for Non-Intersecting Area with RNIA.) RNIA$(\mathcal{A}^U,\mathcal{B}^U) \geq$ RNIA$(\mathcal{A},\mathcal{B})$ for all subspace clusterings $\mathcal{A}$, $\mathcal{B}$.*

Proof.

$$\begin{aligned}
\text{RNIA}(\mathcal{A}^U,\mathcal{B}^U) &= \frac{(|U|+k) - |I|}{(|U|+k)} \\
&\geq \frac{|U| - |I|}{|U|} \\
&= \text{RNIA}(\mathcal{A},\mathcal{B}).
\end{aligned}$$

□

Example 3. *(Penalty for Non-Intersecting Area with VI.) VI does not always penalize for non-intersecting area.*

We show by counterexample that VI does not always penalize for the non-intersecting area. Consider the clusterings in Example 1. We have the same intersection size but larger non-intersecting area for clusterings $\mathcal{B}$ and $\mathcal{C}$ than for clusterings $\mathcal{B}$ and $\mathcal{A}$, so $\mathcal{B}$ and $\mathcal{C}$ should be farther apart. Despite this, $\mathrm{VI}(\mathcal{B},\mathcal{C}) < \mathrm{VI}(\mathcal{A},\mathcal{B})$. This is true also if we scale the VI distances by the logarithms of the appropriate union areas.

Example 4. *(Penalty for Non-Intersecting Area with Rand.) The Rand index does not always penalize for non-intersecting area.*

Consider the clusterings from Example 2. With similar reasoning as in Example 3, the fact that $1 - \mathrm{Rand}(\mathcal{B},\mathcal{C}) < 1 - \mathrm{Rand}(\mathcal{A},\mathcal{B})$ shows that (1-Rand) fails to penalize for the growing non-intersecting area.

6.3 Scale Invariance

Consider scaling all areas by a constant $c \geq 1$.

Theorem 6. *(Scale Invariance for CE.) $CE(c\mathcal{A}, c\mathcal{B}) = CE(\mathcal{A},\mathcal{B})$ for all subspace clusterings $\mathcal{A}$, $\mathcal{B}$.*

Proof.

$$\begin{aligned}\mathrm{CE}(c\mathcal{A}, c\mathcal{B}) &= \frac{c|U| - cD_{max}}{c|U|}\\ &= \frac{|U| - D_{max}}{|U|}\\ &= \mathrm{CE}(\mathcal{A},\mathcal{B}).\end{aligned}$$

□

Theorem 7. *(Scale Invariance for RNIA.) $RNIA(c\mathcal{A}, c\mathcal{B}) = RNIA(\mathcal{A},\mathcal{B})$ for all subspace clusterings $\mathcal{A}$, $\mathcal{B}$.*

Proof.

$$\begin{aligned}\mathrm{RNIA}(c\mathcal{A}, c\mathcal{B}) &= \frac{c|U| - c|I|}{c|U|}\\ &= \frac{|U| - |I|}{|U|}\\ &= \mathrm{RNIA}(\mathcal{A},\mathcal{B}).\end{aligned}$$

□

Example 5. *(Scale Invariance for VI.) VI is not scale invariant in the case of subspace clusterings.*

We show this by counterexample.

$$
\begin{aligned}
&\mathrm{VI}(c\mathcal{A}, c\mathcal{B}) \\
&\quad = \frac{1}{c|U|}\left(\sum_i (cm_i)\log(cm_i) + \sum_j (cm'_j)\log(cm'_j) - 2\sum_i\sum_j (cm_{ij})\log(cm_{ij})\right) \\
&\quad = \mathrm{VI}(\mathcal{A}, \mathcal{B}) + \log c \left(\sum_i m_i + \sum_j m'_j - 2\sum_i\sum_j m_{ij}\right) \\
&\quad \geq \mathrm{VI}(\mathcal{A}, \mathcal{B}),
\end{aligned}
$$

since for subspace clusterings, $\sum_i m_i \geq \sum_i \sum_j m_{ij}$ and $\sum_j m'_j \geq \sum_i \sum_j m_{ij}$. The equality (and thus scale invariance) only holds for ordinary clusterings, for which $m_i = \sum_j m_{ij}$ and $m'_j = \sum_i m_{ij} \quad \forall\, i, j$.

Example 6. *(Scale Invariance for Rand.) The Rand index is not scale invariant in the case of subspace clusterings.*

We show this by counterexample.

$$
\begin{aligned}
1 - \mathrm{Rand}(c\mathcal{A}, c\mathcal{B}) &= \frac{\sum_{i=1}^{L'}\sum_{j=1}^{L}\sum_{k=j+1}^{L}(cm_{ij})(cm_{ik})}{(cm)((cm)-1)/2} \\
&= \frac{\sum_{i=1}^{L}\sum_{j=1}^{L'}\sum_{k=j+1}^{L'}(cm_{ji})(cm_{ki})}{(cm)((cm)-1)/2} \\
&= \frac{\sum_{i=1}^{L'}\sum_{j=1}^{L}\sum_{k=j+1}^{L} m_{ij}m_{ik} + \sum_{i=1}^{L}\sum_{j=1}^{L'}\sum_{k=j+1}^{L'} m_{ji}m_{ki}}{m(m-1/c)/2} \\
&\leq 1 - \mathrm{Rand}(\mathcal{A}, \mathcal{B}).
\end{aligned}
$$

The equality holds only for $c = 1$, i.e. when there is no scaling.

6.4 Copy Invariance

Consider introducing two disjoint copies of the same clustering $\mathcal{S}$ in a large data matrix. Denote the new 'double clustering' by $\mathcal{S}^D$.

Theorem 8. *(Copy Invariance for CE.) CE$(\mathcal{A}^D, \mathcal{B}^D) =$ CE$(\mathcal{A}, \mathcal{B})$ for all subspace clusterings $\mathcal{A}$, $\mathcal{B}$.*

Proof.

$$
\begin{aligned}
\mathrm{CE}(\mathcal{A}^D, \mathcal{B}^D) &= \frac{2|U| - 2D_{max}}{2|U|} \\
&= \frac{|U| - D_{max}}{|U|} \\
&= \mathrm{CE}(\mathcal{A}, \mathcal{B}).
\end{aligned}
$$

□

Theorem 9. *(Copy Invariance for RNIA.) RNIA($\mathcal{A}^D, \mathcal{B}^D$) = RNIA($\mathcal{A}, \mathcal{B}$) for all subspace clusterings $\mathcal{A}$, $\mathcal{B}$.*

Proof.

$$\begin{aligned}\mathrm{RNIA}(\mathcal{A}^D, \mathcal{B}^D) &= \frac{2|U| - 2|I|}{2|U|} \\ &= \frac{|U| - |I|}{|U|} \\ &= \mathrm{RNIA}(\mathcal{A}, \mathcal{B}).\end{aligned}$$

□

Theorem 10. *(Copy Invariance for VI.) VI($\mathcal{A}^D, \mathcal{B}^D$) = VI($\mathcal{A}, \mathcal{B}$) for all subspace clusterings $\mathcal{A}$, $\mathcal{B}$.*

Proof.

$$\begin{aligned}&\mathrm{VI}(\mathcal{A}^D, \mathcal{B}^D) \\ &\quad = \frac{1}{2|U|}\left(2\sum_i m_i \log m_i + 2\sum_j m'_j \log m'_j - 2 \cdot 2\sum_i \sum_j m_{ij} \log m_{ij}\right) \\ &\quad = \mathrm{VI}(\mathcal{A}, \mathcal{B}).\end{aligned}$$

□

Example 7. *(Copy Invariance for Rand.) The Rand index is not copy invariant.*

We show this by counterexample. Note that (1-Rand) can also be written as

$$\begin{aligned}1 - \mathrm{Rand}(\mathcal{A}, \mathcal{B}) &= \frac{1/2(\sum_{i=1}^{L'} \sum_{j=1}^{L} \sum_{k=1}^{L} m_{ij} m_{ik} - \sum_{i=1}^{L} \sum_{j=1}^{L'} m_{ij}^2)}{m(m-1)/2} \\ &+ \frac{1/2(\sum_{i=1}^{L} \sum_{j=1}^{L'} \sum_{k=1}^{L'} m_{ji} m_{ki} - \sum_{i=1}^{L} \sum_{j=1}^{L'} m_{ij}^2)}{m(m-1)/2}\end{aligned}$$

Using this, we have

$$\begin{aligned}1 - \mathrm{Rand}(\mathcal{A}^D, \mathcal{B}^D) &= \frac{1/2(2\sum_{i=1}^{L'} 2\sum_{j=1}^{L} 2\sum_{k=1}^{L} m_{ij} m_{ik} - 2\sum_{i=1}^{L} 2\sum_{j=1}^{L'} m_{ij}^2)}{(2m)((2m)-1)/2} \\ &+ \frac{1/2(2\sum_{i=1}^{L} 2\sum_{j=1}^{L'} 2\sum_{k=1}^{L'} m_{ji} m_{ki} - 2\sum_{i=1}^{L} 2\sum_{j=1}^{L'} m_{ij}^2)}{(2m)((2m)-1)/2} \\ &= \frac{1/2(2\sum_{i=1}^{L'} \sum_{j=1}^{L} \sum_{k=1}^{L} m_{ij} m_{ik} - \sum_{i=1}^{L} \sum_{j=1}^{L'} m_{ij}^2)}{m(m-1/2)/2} \\ &+ \frac{1/2(2\sum_{i=1}^{L} \sum_{j=1}^{L'} \sum_{k=1}^{L'} m_{ji} m_{ki} - \sum_{i=1}^{L} \sum_{j=1}^{L'} m_{ij}^2)}{m(m-1/2)/2} \\ &\neq 1 - \mathrm{Rand}(\mathcal{A}, \mathcal{B})\end{aligned}$$

in the general case.

6.5 Multiple Cluster Coverage Penalty

Consider two subspace clusterings $\mathcal{A} = (A_1)$ and $\mathcal{B} = (B_1, \ldots, B_K)$ such that the clusters B_i are disjoint, of equal size $|A_1|/K = |U|/K$, and fully cover $\operatorname{supp} A_1$.

Theorem 11. *(Multiple Cluster Coverage Penalty for CE.)* $CE(\mathcal{A}, \mathcal{B}) = \frac{K-1}{K}|U|$.

Proof. We allow only one of the clusters B_i to be matched with A_1, so $K-1$ clusters are left unmatched, and the clustering error becomes $\mathrm{CE}(\mathcal{A}, \mathcal{B}) = \frac{K-1}{K}|U|$. □

Theorem 12. *(Multiple Cluster Coverage Penalty for RNIA.)* $RNIA(\mathcal{A}, \mathcal{B}) = 0$.

Proof. In this case $I = U$ and therefore $\mathrm{RNIA}(\mathcal{A}, \mathcal{B}) = 0$. □

Theorem 13. *(Multiple Cluster Coverage Penalty for VI.)* $VI(\mathcal{A}, \mathcal{B}) = \log K$.

Proof. In this case, $M = (m_{1j})$ for $j = 1, \ldots, K$, where $m_{1j} = |U|/K$. Also, $m_1 = |U|$ and $m'_j = |U|/K$. We thus get

$$\begin{aligned}\mathrm{VI}(\mathcal{A}, \mathcal{B}) &= \frac{1}{|U|}\left[|U|\log|U| + K(|U|/K)\log(|U|/K) - 2K(|U|/K)\log(|U|/K)\right]\\ &= \log K.\end{aligned}$$

□

Theorem 14. *(Multiple Cluster Coverage Penalty for Rand.)* $1 - Rand(\mathcal{A}, \mathcal{B}) = [|U|(K-1)][K(|U|-1)]$.

Proof. In this case, $N = |U|(|U|-1)/2$, $N_{11} = 1/2(|U|/K)(|U|/K-1)K$, and $N_{00} = 0$, so (1-Rand) becomes

$$\begin{aligned}1 - \mathrm{Rand}(\mathcal{A}, \mathcal{B}) &= 1 - \frac{N_{00} + N_{11}}{N}\\ &= 1 - \frac{1/2(|U|/K)(|U|/K-1)K}{|U|(|U|-1)/2}\\ &= \frac{|U|(K-1)}{K(|U|-1)}.\end{aligned}$$

□

Appendix 2: Proof for Theorem 1 of Section 5

Proposition 7. *If (u_j, v_j) is a principal pair, then $\sigma_j u_j = \Pi_{\mathcal{F}} v_j$ where $\Pi_{\mathcal{F}} x$ represents the projection of vector x on the subspace $\mathcal{F}$.*

Proof. (Proposition 7) True for $j = 1$ by the definition of the projection operation. For $j > 1$, we have by the definition of (u_j, v_j) that $u_j = \sigma_j^{-1} \Pi_{(u_1,\ldots u_{j-1})^{\perp_{\mathcal{F}}}} v_j$ where $(u_1, \ldots u_{j-1})^{\perp_{\mathcal{F}}}$ represents the orthogonal complement of $(u_1, \ldots u_{j-1})$ in $\mathcal{F}$. By the elementary "3 perpendicular theorem" we also have that $v_j \perp u_{j'}$ for any $j' < j$. Therefore, $\Pi_{(u_1,\ldots u_{j-1})} v_j = 0$ and $\Pi_{\mathcal{F}} v_j = \sigma_j u_j$. □

Proposition 8. *If U is a $n \times a$ matrix of rank at most p, then $||U||_F^2 \leq p||U||_2^2$, where $||\ ||_F$ represents the Frobenius norm.*

Proof. (Proposition 8) $||U||_F^2 = \operatorname{tr} U^T U \overset{1}{=} \sum_{j=1}^{p} \lambda_i(U^T U) \leq p\lambda_{max}(U^T U) = p||U||_2^2$. Equality $\overset{1}{=}$ holds because at most p eigenvalues of $U^T U$ are non-zero. □

Proof. (Theorem 1) Let $\sigma_{ij} = \cos(\theta^j_{\mathcal{F},\mathcal{G}_i})$ and let (u_{ij}, v_{ij}), $u_{ij} \in \mathcal{F}$, $v_{ij} \in \mathcal{G}_i$, $i = 1, \ldots K$, $j = 1, \ldots a_i$ be the principal vectors. Note that $\{v_{ij}\}$ form an orthonormal system and denote by V the $n \times a$ matrix $V = [v_{11}\, v_{12}\, \ldots v_{Ka_K}]$. Define $\tilde{u}_{ij} = \Pi_{\mathcal{F}} v_{ij}$; $\tilde{u}_{ij}$ is a vector of length σ_{ij} with the same direction as u_{ij} by Proposition 7. Form the matrix U having $\tilde{u}_{ij}$, $i = 1, \ldots K$, $j = 1, \ldots a_i$ as columns. Then $||U||_F^2 = \operatorname{tr} U^T U = \sum_{ij} ||\tilde{u}_{ij}||^2 = \sum_{ij} \sigma_{ij}^2$.

It remains to show that $||U||_F^2 \leq p$. But, in matrix notation, $U = HV$ where H is the symmetric, idempotent ($H^2 = H$) matrix representing the projection onto $\mathcal{F}$. It is easy to verify that $||V||_2 = 1$ and $||H||_2 = 1$. Therefore, $||U||_2 \leq ||H||_2 ||V||_2 = 1$ and by virtue of Proposition 8 we obtain $||U||_F^2 \leq p$. □

Appendix 3: Proofs for Section 5.2

Consider two co-clusterings $\mathcal{S} = (S_{ij})$, $\mathcal{S}' = (S'_{ij})$ together with the corresponding row clusterings $\mathcal{R} = (R_i)$, $\mathcal{R}' = (R'_i)$ and the column clusterings $\mathcal{C} = (C_i)$, $\mathcal{C}' = (C'_i)$. For the row clusterings, we define the cluster sizes as $r_i = |R_i|$, $r'_i = |R'_i|$, and the cluster intersection sizes as $r_{ij} = |R_i \cap R'_j|$. Similarly, for the column clusterings, we have $c_i = |C_i|$, $c'_i = |C'_i|$, and $c_{ij} = |C_i \cap C'_j|$. For the co-clusterings, the sizes of the clusters are defined as $m_{ij} = |S_{ij}| = r_i c_j$, $m'_{ij} = |S'_{ij}| = r'_i c'_j$, and the intersection of two co-clusters is $m_{ijkl} = |S_{ij} \cap S'_{kl}| = r_{ik} c_{kl}$. Recall that m and p stand for the number of data matrix rows and columns, respectively. Also recall that D_{max} is the sum of the diagonal elements of the co-clustering confusion matrix after an optimal permutation of the co-cluster labels. Let us write D^R_{max} for the corresponding sum for the row clustering confusion matrix and D^C_{max} for the column clustering confusion matrix.

Theorem 15. *(CE and Co-Clusterings) $CE(\mathcal{S}, \mathcal{S}') \geq CE(\mathcal{R}, \mathcal{R}') + CE(\mathcal{C}, \mathcal{C}') - CE(\mathcal{R}, \mathcal{R}')CE(\mathcal{C}, \mathcal{C}')$ for any co-clusterings $\mathcal{S} = (\mathcal{R}, \mathcal{C})$, $\mathcal{S}' = (\mathcal{R}', \mathcal{C}')$.*

Proof. Fix the permutation of the co-cluster labels that minimizes $\mathrm{CE}(\mathcal{S}, \mathcal{S}')$. However, the same permutation of the labels might not minimize $\mathrm{CE}(\mathcal{R}, \mathcal{R}')$ and/or $\mathrm{CE}(\mathcal{C}, \mathcal{C}')$. If it does, an equality is attained, but we do not know yet if this is the case. Meanwhile, we have

$$\begin{aligned}
\mathrm{CE}(\mathcal{S}, \mathcal{S}') &= \frac{mp - D_{max}}{mp} \\
&= \frac{mp - \sum_i \sum_j m_{ijij}}{mp} \\
&= \frac{mp - \sum_i \sum_j r_{ii} c_{jj}}{mp} \\
&= \frac{mp - \sum_i r_{ii} \sum_j c_{jj}}{mp} \\
&\geq \frac{mp - D^R_{max} D^C_{max}}{mp} \\
&= \frac{m - D^R_{max}}{m} + \frac{p - D^C_{max}}{p} + \frac{(m - D^R_{max})(p - D^C_{max})}{mp} \\
&= \mathrm{CE}(\mathcal{R}, \mathcal{R}') + \mathrm{CE}(\mathcal{C}, \mathcal{C}') - \mathrm{CE}(\mathcal{R}, \mathcal{R}')\mathrm{CE}(\mathcal{C}, \mathcal{C}').
\end{aligned}$$

□

Theorem 16. *(RNIA and Co-Clusterings) RNIA*$(\mathcal{S}, \mathcal{S}') = 0$ *for any co-clusterings* $\mathcal{S} = (\mathcal{R}, \mathcal{C})$, $\mathcal{S}' = (\mathcal{R}', \mathcal{C}')$.

Proof. For co-clusterings it always holds that $I = U$. □

Theorem 17. *(VI and Co-Clusterings) VI*$(\mathcal{S}, \mathcal{S}') =$ *VI*$(\mathcal{R}, \mathcal{R}') +$ *VI*$(\mathcal{C}, \mathcal{C}')$ *for any co-clusterings* $\mathcal{S} = (\mathcal{R}, \mathcal{C})$, $\mathcal{S}' = (\mathcal{R}', \mathcal{C}')$.

Proof.

$$\begin{aligned}
\mathrm{VI}(\mathcal{S}, \mathcal{S}') &= \frac{1}{mp} \sum_i \sum_j \sum_k \sum_l m_{ijkl} \log \frac{m_{ij} m'_{kl}}{m^2_{ijkl}} \\
&= \frac{1}{mp} \sum_i \sum_j \sum_k \sum_l r_{ik} c_{jl} \log \frac{r_i c_j r'_k c'_l}{r^2_{ik} c'^2_{jl}} \\
&= \frac{1}{mp} \sum_i \sum_k r_{ik} \log \frac{r_i r'_k}{r^2_{ik}} \sum_j \sum_l c_{jl} \\
&+ \frac{1}{mp} \sum_j \sum_l c_{jl} \log \frac{c_j c'_l}{c^2_{jl}} \sum_i \sum_k r_{ik} \\
&= \frac{1}{m} \sum_i \sum_k r_{ik} \log \frac{r_i r'_k}{r^2_{ik}} + \frac{1}{p} \sum_j \sum_l c_{jl} \log \frac{c_j c'_l}{c^2_{jl}} \\
&= \mathrm{VI}(\mathcal{R}, \mathcal{R}') + \mathrm{VI}(\mathcal{C}, \mathcal{C}').
\end{aligned}$$

We get this by noticing that $\sum_j \sum_l c_{jl} = p$ and that $\sum_i \sum_k r_{ik} = m$. □

Acknowledgments

We are grateful to Kevin Yip from the University of Hong Kong for providing us the subspace clustering results we used in our experiments. We greatly appreciate the work that Jouni Seppänen from Helsinki University of Technology has done in running tiling and dense itemset experiments for us. We also wish to thank Profs. Heikki Mannila and Anna Karlin for providing Anne Patrikainen an opportunity to visit University of Washington.

References

[1] P. K. Agarwal and N. H. Mustafa. K-means projective clustering. In *Proceedings of the Twenty-third ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 155–165, 2004.

[2] C. Aggarwal, C. Procopiuc, J. Wolf, P. Yu, and J. Park. A framework for finding projected clusters in high dimensional spaces. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 1999.

[3] C. C. Aggarwal and P. S. Yu. Finding generalized projected clusters in high dimensional spaces. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 70–81, 2000.

[4] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 94–105, 1998.

[5] D. Aldous and J. Fill. Reversible Markov Chains and random walks on graphs. `http://www.stat.berkeley.edu/users/aldous/RWG/book.html`.

[6] M. E. Argentati. Principal angles between subspaces. `http://www-math.cudenver.edu/~aknyazev/teaching/rico/talk_defense.pdf`.

[7] P. Artigas, A. Goldenberg, A. Likhodedov, and R. Caruana. Meta clustering. `http://www-2.cs.cmu.edu/~artigas/classproj/mlproj.ps`, 2000.

[8] A. Banerjee, I. Dhillon, J. Ghosh, S. Merugu, and D. S. Modha. A generalized maximum entropy approach to bregman co-clustering and matrix approximation. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

[9] A. Ben-Hur, A. Elisseeff, and I. Guyon. A stability based method for discovering structure in clustered data. In *Pacific Symposium on Biocomputing*, pages 6–17, 2002.

[10] J. Besson, C. Robardet, and J-F. Boulicaut. Mining alpha-beta concepts as relevant bi-sets from transactional data. In *Proceedings of the 3rd International Workshop on Knowledge Discovery in Inductive Databases KDID'04*, 2004.

[11] A. Björk and G. H. Golub. Numerical methods for computing angles between linear subspaces. *Mathematical Computation*, 27:579–594, 1973.

[12] C. Böhm, K. Kailing, P. Kröger, and A. Zimek. Computing clusters of correlation connected objects. In *Proceedings of the 2004 ACM SIGMOD International Conference on Management of Data*, pages 455–466, 2004.

[13] C. H. Cheng, A. W.-C. Fu, and Y. Zhang. Entropy-based subspace clustering for mining numerical data. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 84–93, 1999.

[14] Y. Cheng and G. M. Church. Biclustering of expression data. In *Proceedings of the Eighth International Conference on Intelligent Systems for Molecular Biology*, 2000.

[15] H. Cho, I. S. Dhillon, Y. Guan, and S. Sra. Minimum sum-squared residue co-clustering of gene expression data. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, 2004.

[16] D. L. Davies and D. W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(2):224–227, 1979.

[17] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretic co-clustering. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.

[18] I. S. Dhillon, S. Mallela, and D. S. Modha. Information-theoretic co-clustering. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.

[19] C. Domeniconi, D. Papadopoulos, D. Gunopulos, and S. Ma. Subspace clustering of high dimensional data. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, 2004.

[20] Z. Drmac. On principal angles between subspaces of euclidean space. *Siam Journal of Matrix Analysis Applications*, 22(1):173–194, 2000.

[21] S. Dudoit and J. Fridlyand. A prediction-based resampling method for estimating the number of clusters in a dataset. *Genome Biology*, 3(7), 2002.

[22] E. B. Fowlkes and C. L. Mallows. A method for comparing two hierarchical clusterings. *Journal of the American Statistical Association*, 78(383):553–569, 1983.

[23] J. H. Friedman and J. J. Meulman. Clustering objects on subsets of attributes. *Journal of the Royal Statistical Society B*, 66:1–25, 2004.

[24] G. Getz, E. Levine, and E. Domany. Coupled two-way clustering analysis of gene microarray data. *Proceedings of the National Academy of Sciences, USA*, 97:12079–12084, 2000.

[25] A. Gionis, H. Mannila, and J. K. Seppänen. Geometric and combinatorial tiles in 0-1 data. In *Proceedings of the European Conference on Principles and Practice of Knowledge Dicovery in Databases*, 2004.

[26] A. Gionis, H. Mannila, and P. Tsaparas. Clustering aggregation. In *Proceedings of the 21st International Conference on Data Engineering*, 2005.

[27] A. Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.

[28] P. Jaccard. The distribution of flora in the alpine zone. *The New Phytologist*, 11(2):37–50, 1912.

[29] A. K. Jain, A. Topchy, M. H. Law, and J. Buhmann. Landscape of clustering algorithms. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 260–263, 2004.

[30] A. Kaban, E. Bingham, and T. Hirsimäki. Learning to read between the lines: The aspect Bernoulli model. In *Proceedings of the 4th SIAM International Conference on Data Mining*, pages 462–466, 2004.

[31] K. Kailing, H.-P. Kriegel, and P. Kröger. Density-connected subspace clustering for high-dimensional data. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, pages 246–257, 2004.

[32] N. Kaplan, O. Sasson, U. Inbar, M. Friedlich, M. Fromer, H. Fleischer, E. Portugaly, N. Linial, and M. Linial. Protonet 4.0: A hierarchical classification of one million protein sequences. *Nucleic Acids Research*, 33:216–218, 2005.

[33] Y. Kluger, R. Basri, J. T. Chang, and M. Gerstein. Spectral biclustering of microarray data: coclustering genes and conditions. *Genome Research*, 13(4):703–16, April 2003.

[34] T. Lange, V. Roth, M. L. Braun, and J. M. Buhmann. Stability-based validation of clustering solutions. *Neural Computation*, 16:1299–1323, 2004.

[35] E. Levine and E. Domany. Resampling method for unsupervised estimation of cluster validity. *Neural Computation*, 13:2573–2593, 2001.

[36] J. Liu, W. Wang, and J. Yang. A framework for ontology-driven subspace clustering. In *Proceedings of the Tenth ACM SIGKDD International conference on Knowledge Discovery and Data Mining*, 2004.

[37] M. Meila. Comparing clusterings by the variation of information. In *Proceedings of the Sixteenth Annual Conference on Computational Learning Theory*, pages 173–187, 2003.

[38] A. A. Melkman and E. Shaham. Sleeved coclustering. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

[39] B. Mirkin. *Mathematical classification and clustering.* Kluwer Academic Press, 1996.

[40] N. Mishra, D. Ron, and R. Swaminathan. A new conceptual clustering framework. *Machine Learning*, 56(1–3):115–151, 2004.

[41] H. Nagesh, S. Goil, and A. Choudhary. MAFIA: Efficient and scalable subspace clustering for very large data sets. Technical Report 9906-010, Northwestern University, 1999.

[42] E. Oja and J. Parkkinen. On subspace clustering. In *Prceedings of the Seventh International Conference on Pattern Recognition*, 1984.

[43] C. H. Papadimitriou and K. Steiglitz. *Combinatorial Optimization, Algorithms and Complexity.* Prentice-Hall, 1982.

[44] L. Parsons, E. Haque, and H. Liu. Evaluating subspace clustering algorithms. In *Proceedings of the Fourth SIAM International Conference on Data Mining, Workshop on Clustering High Dimensional Data and its Applications*, 2004.

[45] L. Parsons, E. Haque, and H. Liu. Subspace clustering for high dimensional data: A review. *ACM SIGKDD Explorations Newsletter*, 6(1), 2004.

[46] A. Patrikainen and H. Mannila. Subspace clustering of high dimensional binary data — a probabilistic approach. In *Proceedings of the Fourth SIAM International Conference on Data Mining, Workshop on Clustering High Dimensional Data and its Applications*, 2004.

[47] K.S. Pollard and M. J. van der Laan. Statistical inference for simultaneous clustering of gene expression data. *Mathematical Biosciences*, 176(1):99–121, 2002.

[48] C. M. Procopiuc, M. T. Jones, P. K. Agarwal, and T. M. Murali. A Monte carlo algorithm for fast projective clustering. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, 2002.

[49] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66:846–850, 1971.

[50] C. Van Rijsbergen. *Information Retrieval.* Butterworths, 2nd edition, 1979.

[51] J. K. Seppänen and H. Mannila. Dense itemsets. In *Proceedings of ACM SIGKDD Internation Conference on Knowledge Discovery and Data Mining*, 2004.

[52] A. Strehl and J. Ghosh. Cluster ensembles — a knowledge reuse framework for combining multiple partititons. *Journal on Machine Learning Research*, 3:583–617, 2002.

[53] A. Tanay, R. Sharan, and R. Shamir. Discovering statistically significant biclusters in gene expression data. *Bioinformatics*, 18:136–144, 2002.

[54] S. Theodoridis and K. Koutroumbas. *Patter Recognition*. Academic Press, 1999.

[55] A. Topchy, A.K. Jain, and W. Punch. A mixture model for clustering ensembles. In *Proceedings of the Fourth SIAM International Conference on Data Mining*, 2004.

[56] A. Topchy, M. H. Law, A. K. Jain, and A. Fred. Analysis of consensus partition in cluster ensemble. In *Proceedings of The Fourth IEEE International Conference on Data Mining*, 2004.

[57] A. Topchy, B. Minaei-Bidgoli, A.K. Jain, and W. Punch. Adaptive clustering ensembles. In *Proceedings of the 17th International Conference on Pattern Recognition*, pages 272–275, 2004.

[58] D. L. Wallace. Comment. *Journal of the American Statistical Association*, 78(383):569–576, 1983.

[59] C. Yang, U. Fayyad, and P. S. Bradley. Efficient discovery of error-tolerant frequent itemsets in high dimensions. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2000.

[60] J. Yang, W. Wang, H. Wang, and P. S. Yu. Delta-cluster: Capturing subspace correlation in a large data set. In *Proceedings of the 18th IEEE International Conference on Data Engineering*, pages 517–528, 2002.

[61] K. Y. Yip. HARP: A practical projected clustering algorithm for mining gene expression data. Master's thesis, The University of Hong Kong, Pokfulam Road, Hong Kong, 2004. `http://www.csis.hku.hk/~ylyip/papers/thesis.pdf`.

[62] K. Y. Yip, D. W. Cheung, and M. K. Ng. HARP: A practical projected clustering algorithm. *IEEE Transactions on Knowledge and Data Engineering*, 16(11):1387–1397, 2004.

[63] K. Y. Yip, D. W. Cheung, and M. K. Ng. On discovery of extremely low-dimensional clusters using semi-supervised projected clustering. In *Proceedings of the 21st International Conference on Data Engineering*, 2005.

[64] K. Y. Yip, M. K. Ng, and D. W. Cheung. A review on projected clustering algorithms. *International Journal of Applied Mathematics*, 13:35–47, 2003.