

# When Pattern met Subspace Cluster — a Relationship Story

Jilles Vreeken<sup>1</sup> and Arthur Zimek<sup>2</sup>

<sup>1</sup> Advanced Database Research and Modeling, Universiteit Antwerpen, Belgium  
<http://www.adrem.ua.ac.be/~jvreeken>  
[jilles.vreeken@ua.ac.be](mailto:jilles.vreeken@ua.ac.be)

<sup>2</sup> Ludwig-Maximilians-Universität München, Germany  
<http://www.dbs.ifi.lmu.de/~zimek>  
[zimek@dbs.ifi.lmu.de](mailto:zimek@dbs.ifi.lmu.de)

**Abstract.** While subspace clustering emerged as an application of pattern mining and some of its early advances have probably been inspired by developments in pattern mining, over the years both fields progressed rather independently. In this paper, we identify a number of recent developments in pattern mining that are likely to be applicable to alleviate or solve current problems in subspace clustering and vice versa.

## 1 Introduction

Arguably, the two main proponents of exploratory data mining research are clustering and pattern mining. Both share the aim of the field—extracting interesting, non-trivial, and previously unknown knowledge from data—yet, they are orthogonal in their approach, or at least appear so at first glance.

Pattern mining, to start with, is concerned with developing theory and algorithms for identifying groups of attributes and some selection criteria on those; such that the most ‘interesting’ attribute-values combinations in the data are returned. That is, by that selection we identify objects in the data that somehow stand out. The prototypical example is supermarket basket analysis, in which by frequent itemset mining we identify items that are frequently sold together—with the infamous *beer* and *nappies* as an example of an interesting pattern.

Clustering, on the other hand, in general aims at finding groups of similar objects in a database. Aside from algorithmic variations in the process of identifying these groups, the major differences between various clustering approaches is in the actual meaning of ‘similar’. Especially in high dimensional data the notion of similarity is not a trivial one. The so-called ‘curse of dimensionality’ is often given as the main motivation for ‘subspace clustering’ [34], where our goal is to identify both sets of objects, as well as subsets of attributes (subspaces), on which those objects are measured to be highly similar. As such, we see that both pattern mining and subspace clustering identify sub-parts of the data.

In this paper we explore and discuss a number of connections between these two active fields of research. We argue that research in subspace clustering,

having a common origin with pattern mining and sharing some early ideas, has deviated from the route of pattern mining subsequently. Interestingly, both fields now face problems already studied by the other. Here, we would like to point out interesting recent research topics on pattern mining where research on subspace clustering can possibly benefit from, and vice versa. For example, the explosion in numbers of results, and reducing their redundancy, are currently open problems in subspace clustering but have recently been studied in detail by the pattern mining community. On the other hand, the notion of alternative results, as well as the generalization beyond binary data, are topics where pattern miners may draw much inspiration from recent work in (subspace) clustering.

The goal of this paper is to identify a number of developments in these fields that should not go unnoticed; we are convinced that solutions for pattern mining problems are applicable in subspace clustering, and vice versa. In other words, it is time to meet the relatives.

The remainder of this paper is organized as follows. First, we discuss the background of subspace clustering, and how it relates to pattern mining. Next, we go into the similarities between their results. Section 4 discusses developments in pattern mining that are interesting with regard to subspace clustering—and vice versa in Section 5. We round up with conclusions in Section 6.

## 2 It’s a Family Affair

### 2.1 The Curse

The so-called ‘curse of dimensionality’ is often credited for causing problems in similarity computations in high dimensional data, and, hence, is given as motivation for specialized approaches such as ‘subspace clustering’ [34]. Let us consider two aspects of the ‘curse’ that are often confused in the literature: (i) the concentration effect of  $L_p$ -norms and (ii) the presence of irrelevant attributes.

Regarding the concentration effect (i), the key result of [10] states that, if the ratio of the variance of the length of any point vector  $\mathbf{x} \in \mathbb{R}^d$  (denoted by  $\|\mathbf{x}\|$ ) with the length of the mean point vector (denoted by  $E[\|\mathbf{x}\|]$ ) converges to zero with increasing data dimensionality, then the proportional difference between the farthest-point distance  $D_{max}$  and the closest-point distance  $D_{min}$  (the *relative contrast*) vanishes, i.e., all distances concentrate around a mean, and look alike. This observation is often quoted for motivating subspace clustering as a specialized procedure. It should be noted, though, that the problem is neither well enough understood (see e.g. [20]) nor actually relevant when the data follows different, well separated distributions [8, 9, 29].

Regarding the separation of clusters, the second problem (ii) is far more important for subspace clustering: In order to find structures describing phenomena, abundances of highly detailed data are collected. Among the features of a high dimensional data set, for any given query object, many attributes can be expected to be irrelevant to describing that object. Irrelevant attributes can easily obscure clusters that are clearly visible when we consider only the relevant

‘subspace’ of the dataset. Hence, they interfere with the performance of similarity measures in general, but in a far more fundamental way for clustering. The relevance of certain attributes may differ for different groups of objects within the same data set. Thus, many subsets of objects are defined only by some of the available attributes, and the irrelevant attributes (‘noise’) will interfere with the efforts to find these subsets. This second problem is actually the true motivation for designing specialized methods to look for clusters in subspaces of a dataset.

## 2.2 Variants

In general, in subspace clustering similarity is defined in some relation to subsets or combinations of attributes or dimensions of database objects. Hence, a clustering with  $n$  clusters for a database  $\mathcal{D} \times \mathcal{A}$ , with the set of objects  $\mathcal{D}$  and with the full set of attributes  $\mathcal{A}$ , can be seen as a set  $\mathcal{C} = \{(\mathcal{C}_1, \mathcal{A}_1), \dots, (\mathcal{C}_n, \mathcal{A}_n)\}$ , where  $\mathcal{C}_i \subseteq \mathcal{D}$  and  $\mathcal{A}_i \subseteq \mathcal{A}$ , i.e., a cluster is defined w.r.t. a set of objects and w.r.t. a set of attributes (i.e., a subspace).

Subspace clustering algorithms are typically split into two groups; in ‘projected clustering’ objects belong to at most one cluster, while ‘subspace clustering’ (in a more narrow sense) seeks to find all possible clusters in all available subspaces, allowing overlap [34]. The distinction (and terminology) originates from the two pioneering papers in the field, namely CLIQUE [2] for ‘subspace clustering’ and PROCLUS [1] for projected clustering; and the two definitions allow a broad field of hybrids. Since we are interested in the relationship between pattern mining and subspace clustering, we will let aside projected clustering and hybrid approaches and concentrate on subspace clustering in the narrower sense as defined in [2]. In this setting, subspace clustering is usually related to a bottom-up traversal of the search space of all possible subspaces, i.e., starting with all one dimensional subspaces, two-dimensional combinations of these subspaces, three dimensional combinations of the two dimensional subspaces and so on, all (relevant) subspaces are searched for clusters residing therein.

Considering CLIQUE, we find the intuition of subspace clustering promoted there closely related to pattern mining. To this end, we consider frequent itemset mining [3], in which we consider binary transaction data, where transactions are sets of items  $A, B, C$ , etc. The key idea of APRIORI [3] is to start with itemsets of size 1 that are frequent, and exclude all itemsets from the search that cannot be frequent anymore, given the knowledge which smaller itemsets are frequent. For example, if we count a 1-itemset containing  $A$  less than the given minimum support threshold, all 2-itemsets, 3-itemsets, etc. containing  $A$  (e.g.,  $\{A, B\}, \{A, C\}, \{A, B, C\}$ ) cannot be frequent either and need not be considered. While theoretically the search space remains exponential, in practice searching becomes feasible even for very large datasets.

Transferring this problem to subspace clustering, each attribute represents an item, and each subspace cluster is then an itemset containing the items representing the attributes of the subspace. This way, finding itemsets with support 1 relates to finding all combinations of attributes constituting a subspace of at least

one cluster. This observation is the rationale of most bottom-up subspace clustering approaches: subspaces containing clusters are determined starting from all 1-dimensional subspaces accommodating at least one cluster, employing a search strategy similar to that of itemset mining algorithms. To apply any efficient algorithm, the cluster criterion must implement a downward closure property (i.e. (anti-)monotonicity): *If subspace  $\mathcal{A}_i$  contains a cluster, then any subspace  $\mathcal{A}_j \subseteq \mathcal{A}_i$  must also contain a cluster.* The anti-monotonic reverse implication, *if a subspace  $\mathcal{A}_j$  does not contain a cluster, then any superspace  $\mathcal{A}_i \supseteq \mathcal{A}_j$  also cannot contain a cluster,* can subsequently be used for pruning.

Clearly, this is a rather naïve use of the concept of frequent itemsets in subspace clustering. What constitutes a good subspace clustering result is defined here apparently in close relationship to the design of the algorithm, i.e., the desired result appears to be defined according to the expected result (as opposed to: in accordance to what makes sense) — see the discussion in [34]. Resulting clusters are usually highly redundant and, hence, mostly useless.

This issue is strongly related to the so-called pattern explosion. Taking frequent itemset mining as an example, we see that for high minimal support thresholds, only trivial results are returned, but that for lower thresholds we end up with enormous amounts of results—a collection that is highly redundant, and many of the returned patterns are variations of the same theme. Recently, pattern miners have started to acknowledge they have been asking the wrong question: instead of asking for *all* patterns that satisfy some constraints, we should ask for small, non-redundant, and high quality *sets of patterns*—where by high-quality we mean that each of the patterns in the set satisfy the thresholds we set on interestingness or similarity, and the set is optimal with regard to some criterion, e.g. mutual information [32], compression [49], area [21].

Research on subspace clustering inherited all the deficiencies from this originally ill-posed problem. However, early research on subspace clustering as follow-ups of CLIQUE apparently also tried to transfer improvements from pattern mining. As an example, ENCLUS [14] uses several quality criteria for subspaces, not only implementing the downward closure property, but also an upward closure (i.e., allowing search for interesting subspaces as specializations as well as generalizations). This most probably relates to the concept of positive and negative borders known from closed frequent itemsets [46]. Both can be seen as implementations of the classical concept of version spaces [38].

### 3 I Say Pattern, You Say Subspace Cluster

Methods aside, there are two notions we have to discuss that do, or do not, make the two fields different. First and foremost, what is a result? And, second, can we relate interestingness and similarity? To start with the former, in subspace clustering, a single result defined by the Cartesian product of objects  $C \subseteq \mathcal{D}$  and attributes  $A \subseteq \mathcal{A}$ . In order to be considered as a result, each of the objects in the selection should be similar to the others, over the selected attributes, according to the employed similarity function. In order to make a natural connection to

pattern mining, we adopt a visual approach; if we are allowed to re-order both attributes and objects freely, we can reorder  $\mathcal{D}$  and  $\mathcal{A}$  such that  $C$  and  $A$  define a rectangle in the data, or a *tile*. In pattern mining, the notion of a tile has become very important in recent years [17, 21, 23, 33]. Originally the definition of a pattern was very much along the lines of an SQL query, posing selection criteria on which objects in the data are considered to *support* the pattern or not. As such, beyond whether they contribute to such a global statistic, the selected objects were not really taken into account. In many recent papers, however, the supporting objects are explicitly taken into account, and by doing so, patterns also naturally define *tiles*. In the next section we will link this approach to the reduction of redundancy. So, both fields identify *tiles*, sub-parts of the data. However, both have different ways of arriving at these tiles. In pattern mining, results are typically selected by some measure of ‘interestingness’—of which support, the number of selected objects, is the most well-known example. In subspace clustering, on the other hand, we measure results by how similar the selected objects are over the considered attributes. Clearly, while this may lead to discovering rather different tiles, it is important to realize that both approaches do find tiles, and provide some statistics for the contents of each tile.

We observe that in pattern mining the selection of the objects ‘belonging’ to the pattern is very strict—and that as such those objects will exhibit high similarity over the subspace of attributes the pattern identifies. For example, in standard frequent itemset mining, transactions (i.e., objects) are only selected if they are a strict superset of the pattern at hand—and in fault-tolerant itemset mining typically only very few attribute-value combinations are allowed to deviate from the template the pattern identifies. Linking this to similarity, in this strict selection setting, it is easy to see that for the attributes identified by the pattern, the selected objects are highly similar. The same also holds for subgroup discovery, a supervised branch of pattern mining. In subgroup discovery the patterns typically strongly resemble SQL range-based selection queries, where the goal is to identify those patterns (intention) that select objects (extension) that correlate strongly to some target attribute(s). Intuitively, the more strict the selection criteria are per attribute, the more alike the selected objects will be on those attributes. So, in the traditional sense, patterns identified as interesting by a measure using support, are highly likely to correspond to highly-similar subspace clusters, the more strict conditions the pattern defines on its attributes. The other way around, we can say that the higher the similarity of a subspace cluster, the easier it will be to define a pattern that covers the same area of the database. And, the larger this highly-similar subspace cluster is, the more likely it is that it will be discovered by pattern mining using any support-based interestingness measure.

Besides this link, it is interesting to consider what the main differences are. In our view, it is a matter of perspective; whether to take a truly local stance at the objects, and from *within* a tile, like in subspace clustering, or, whether to take a slightly more global stance and look at how we can select those objects by defining conditions on the attributes. Further, we remark that both interest-

ingness and similarity are very vague concepts, for which many proposals exist. A unifying theory, likely from a statistical point of view, would be very welcome.

## 4 Advances of Interest in Pattern Mining

In this section we discuss some recent advances in pattern mining research that may likewise be applicable for issues in subspace clustering.

### 4.1 Summarizing Sets of Patterns

As touched upon in Section 2, reducing redundancy has been studied for a long period of time in pattern mining. Very roughly speaking, two main approaches can be distinguished: summarizing the result set, and summarizing the data.

In this subsection we discuss the former, in which we find well-known examples. The main idea of this approach is that we have a set of results  $\mathcal{F}$ , consisting of results that have passed the constraints that we have set, e.g. they all pass the interestingness threshold. Now, with the goal of reducing redundancy in  $\mathcal{F}$ , we want to select a subset  $\mathcal{S} \subseteq \mathcal{F}$  such that  $\mathcal{S}$  contains as much information on the whole of  $\mathcal{F}$  while being minimal in size.

Perhaps the most well-known examples of this approach are *closed* [46] and *maximal* [7] frequent itemsets, by which we only allow elements  $X \in \mathcal{F}$  into  $\mathcal{S}$  for which no superset exists that has the same support, or no superset exists that does not meet the mining constraints, respectively. As such, for closed sets, given  $\mathcal{S}$  we can reconstruct  $\mathcal{F}$  without loss of information—and for maximal sets we can reconstruct only the itemsets, not their frequencies. Non-derivable itemsets [13] follow a slightly different approach, and only provide those itemsets for which their frequency cannot be derived from the rest. While the concepts of closed and maximal have been applied in subspace clustering, non-derivability has not been explored yet, to the best of our knowledge.

Reduction by closure only works well when data are highly structured, and it deteriorates rapidly with noise. A recent improvement is margin-closedness [39], where we consider elements into the closure for which our measurement falls within a given margin. This provides strong reduction in redundancy, and higher noise resistance; we expect it to be applicable for subspace clusters.

Perhaps not trivially translatable to subspace clusters, another branch of summarization is that of picking or creating a number of representative results. Yan et al. [51] choose  $\mathcal{S}$  such that the error of predicting the frequencies in  $\mathcal{F}$  is minimized. Here, it may well be reasonable to replace frequency with similarity. There are some attempts in this direction, e.g. in biclustering [48].

More examples exist, but for reasons of space, we continue to a more important development of reducing redundancy.

### 4.2 Pattern Set Mining

While the above-mentioned techniques do reduce redundancy, they typically still result in large collections of patterns, that still do contain many variations of the

same theme. As stated in Section 2, a recent major insight in pattern mining is that we were asking the wrong question. Instead of asking for all patterns that satisfy some constraints, we should be asking for a small non-redundant group of high-quality patterns. With this insight, the attention shifted from attempting to summarize the full result  $\mathcal{F}$ , to provide a good summarization of the *data*. In terms of subspace clustering, this means that we would select that group of subspace clusters such that we can approximate (explain, describe, etc.) the data optimally. Here we discuss a few examples of such *pattern set mining* techniques that we think are applicable to subspace clustering in varying degrees.

The most straightforward technique we discuss is tiling [21]. It proposes to not just consider itemsets, but also the transactions they occur in—the same notion we adopted in Section 3. The main idea here is that patterns that cover a large area are more interesting than patterns of a small area, where area is defined as the product of the number of items and number of transactions that support the itemset. Most importantly, the authors give an algorithm for approximating the optimal *tiling* of the data—those  $k$  tiles that together cover as much of the data as possible. As the paper only considers exact tiles, for which exactly what the data values are, namely 1s, the returned tilings are good summarizations of the data. It is not trivially translated to subspace clustering. One could extract a cluster profile, e.g. a centroid, and take the deviation between the current summary and the real data into account—something that one could minimize.

In this direction, other promising approaches take cues from Information Theory, the Minimum Description Length principle [25] in particular. That is, they approach the pattern set selection problem from the perspective of lossless compression; the best set of patterns is that set of patterns that together compress the data best. Gionis et al. [23] propose a hierarchical model for discovering informative regions (patterns, subspaces) in the data by employing MDL. It does not consider a candidate set  $\mathcal{F}$ , but looks for interesting regions directly, assuming a given fixed order of the attributes and objects. The hierarchical nature potentially does link strongly to subspace clustering, where we could consider nested clusters—a related method for clusters was proposed by Böhm et al [12]. Siebes et al. [49] proposed the KRIMP algorithm to approximate the set of itemsets that together optimally compress the data from a candidate collection  $\mathcal{F}$ . The resulting code tables have been shown to be of very high quality, while reducing the number of patterns up to 7 orders of magnitude [49]. In turn, Krontonias and De Bie [33] combine the ideas of MDL and Tiling, although they do not simply accumulate tiles to maximize the covered area of the database. Instead, they measure how informative candidate results are with regard to a static probabilistic background model, while also taking their complexity into account. In other words, how many bits does adding result  $X$  save us when explaining the data, and how many does it cost to understand  $X$ .

Each of the above methods have, as of yet, only been defined for (single and multi-table) binary data. However, MDL theory does exist for richer data types, and we would like to point out the strong potential for reducing redundancy in subspace clustering by aiming at that set of subspace clusters that together

describe the data best. That is, those clusters by which we can encode the data and the model most succinctly. Note that this approach naturally allows for overlapping clusters, as well as refinements (a big general cluster, and a smaller sub-region of it)—results will be selected if they provide sufficient extra information by which the data can be compressed better than without, while not costing too much to be described themselves.

### 4.3 Significance and Randomization

Perhaps the largest problem in exploratory data mining is validation. Unlike in settings where there is a clear formal goal, such as in many supervised machine learning, our goal is as ill-defined as to find ‘interesting things’. Like in clustering a plethora of different similarity measures have been considered, all of which may identify some interesting interplay between objects, also in pattern mining a broad spectrum of interestingness measures have been discussed, yet there is no gold standard by which we can compare results.

One approach that recently received ample attention in pattern mining is that of statistical significance. If a result can be easily explained by background knowledge, it will most likely not be interesting to the end user, never mind how large its support or similarity. Webb [50] proposes to rank patterns depending on their individual statistical significance. A more general framework was proposed by Gionis et al. [22], who propose to investigate significance of results *in general* through randomization. To this end, they introduce swap randomization as a means to sample random binary datasets of the same row and column margins as the original data, and so calculate empirical  $p$ -values. Ojala et al. [44, 45] gave variants for numerical data, easing the use of the model for subspace clustering. Hanhijarvi et al. [28] extended the framework such that more complex background information, such as cluster densities and itemset frequencies, can be entered into the model—making the approach applicable for iterative data mining. De Bie [17] proposed to model these probability distributions over datasets *analytically*, by employing the Maximum Entropy principle. A main advantage is that this allows for the calculation of exact  $p$ -values.

As of yet, with the exception of the latter, each of the above have already been formalized for a wide variety of data types, and, hence, we expect these methods to be rather easily applicable for assessing whether a subspace cluster, subspace clustering or multiple clustering is significant—whether in light of some basic properties of the data, or with regard to more involved known structure.

## 5 Interesting Advances in Subspace Clustering

In this section we discuss advances in subspace clustering that may be of particular worth in progressing pattern mining research.



## 5.1 Half of Success is Knowing When to Stop

In early approaches to subspace clustering, the fixed grid, that allows an easy translation to frequent itemsets, introduces bias towards certain cluster properties. Thus, it has found major interest in research on subspace clustering. The MAFIA [43] method uses an adaptive grid, while its generation of subspace clusters is similar to CLIQUE. Another variant, nCluster [37], allows overlapping windows of length  $\delta$  as 1-dimensional units of the grid. SUBCLU [31] uses the DBSCAN cluster model of density-connected sets [18], letting go the grid-approach completely. Nevertheless, density-connected sets satisfy the downward closure property. This enables SUBCLU to search for density-based clusters in subspaces also in an APRIORI-like style. A global density threshold, as used by SUBCLU and the grid-based approaches, leads to a bias towards a certain dimensionality: a tighter threshold separates clusters from noise well in low dimensions, but tends to loose clusters in higher dimensions. A more relaxed threshold detects high dimensional clusters but produces an excessive amount of low dimensional clusters. This problem has been of major interest in the research on subspace clustering in the recent years. See e.g. [4, 42], where the density-threshold is adapted in turn to the dimensionality of the subspace currently being scrutinized during the run of the algorithm.

A problem closely related to choosing the appropriate density level is the redundancy issue, that also found much interest recently [5, 26, 41]. These approaches aim at reporting only the most representative of a couple of redundant subspace clusters. While technically the approaches differ, in concept, adaptive density-thresholds show high similarity with selection of patterns based on statistical significance [33, 50]. Significance of subspace clusters, though, has only been addressed once so far [40]. Hence, we regard it as highly likely that both approaches can learn from each other's endeavours.

## 5.2 Alternatively...

A recent development in both fields is finding alternatives to results. The techniques we employ in exploratory data mining can only seldom be shown to provide optimal results, instead typically returning good results heuristically. However, while one good result might shed light on one aspect of the data, it might ignore other parts of the data—for which other results will be informative. A clustering result that can be judged valid by known structures may even be completely uninteresting [19]. Hence a proposal to improve cluster evaluation relies on the *deviation* of a clustering from known structures instead of judging the coverage of known structures [35].

This is almost literally the approach taken in the subfield of alternative clustering, where one wants to discover a good clustering that is orthogonal from what we already found, or, alternatively, where we want to find  $n$  good clusterings each of which be different from any other. Approaches for finding alternative clusterings mostly use ensemble techniques [6, 15, 16, 24]. A key requirement for building good ensembles is a source of diversity for the ensemble members.

Clearly, using different feature subsets (i.e., subspaces) can be a very good source of diversity and actually has occasionally been used in alternative clustering as one possibility to find different clustering solutions [47]. Alternative clustering approaches typically seek diversity constrained by non-redundancy. Hence, allowing some degree of redundancy could be meaningful, such as allowing overlap between clusters. In different subspaces, one subset of objects could belong to two different, yet meaningful, clusters. While this would increase their redundancy, reporting both of them instead of only one would not necessarily be useless. Considerations in this direction can be found w.r.t. subspace clustering [27].

Also it has been conceded that preserving known properties or concepts is desirable when seeking different clusters [47]. As searching for subspaces that are (at least partially) different from subspaces of already found clusters, the more specialized area of multiview clustering [11, 30] is also of interest here, and can be seen as a special case of seeking alternative results. The constraint here is the orthogonality of subspaces. It can also be seen as a special case of subspace clustering allowing maximal overlap yet seeking minimally redundant clusters by accommodating different concepts (as proposed e.g. in [27]). These approaches highlight that highly overlapping clusters in different subspaces (i.e., certain subsets of objects may belong to several clusters simultaneously) need not be redundant nor meaningless (see also the discussion in [19, 36]).

## 6 Conclusion

There exist strong links between Subspace Clustering and Pattern Mining, although the topics of research within the two fields have diverged over time. We argued the case that both fields are not as different as they might think, and moreover, that both can learn much from the experience gained by the other. In other words, we say it is time for the two fields to meet again. To this end, we gave a (far from complete) overview of proposals from the one field that we find have strong potential to advance research in the other, and vice versa.

**Acknowledgements** Jilles Vreeken is supported by a Post-Doctoral Fellowship of the Research Foundation – Flanders (FWO).

## References

1. C. C. Aggarwal, C. M. Procopiuc, J. L. Wolf, P. S. Yu, and J. S. Park. Fast algorithms for projected clustering. In *Proc. SIGMOD*, 1999.
2. R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan. Automatic subspace clustering of high dimensional data for data mining applications. In *Proc. SIGMOD*, 1998.
3. R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proc. SIGMOD*, 1994.
4. I. Assent, R. Krieger, E. Müller, and T. Seidl. DUSC: dimensionality unbiased subspace clustering. In *Proc. ICDM*, 2007.

5. I. Assent, E. Müller, S. Günemann, R. Krieger, and T. Seidl. Less is more: Non-redundant subspace clustering. In *Proc. ACM SIGKDD Workshop MultiClust*, 2010.
6. E. Bae and J. Bailey. COALA: a novel approach for the extraction of an alternate clustering of high quality and high dissimilarity. In *Proc. ICDM*, 2006.
7. R. Bayardo. Efficiently mining long patterns from databases. In *Proc. SIGMOD*, pages 85–93, 1998.
8. K. P. Bennett, U. Fayyad, and D. Geiger. Density-based indexing for approximate nearest-neighbor queries. In *Proc. KDD*, 1999.
9. T. Bernecker, M. E. Houle, H.-P. Kriegel, P. Kröger, M. Renz, E. Schubert, and A. Zimek. Quality of similarity rankings in time series. In *Proc. SSTD*, 2011.
10. K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft. When is “nearest neighbor” meaningful? In *Proc. ICDT*, 1999.
11. S. Bickel and T. Scheffer. Multi-view clustering. In *Proc. ICDM*, 2004.
12. C. Böhm, F. Fiedler, A. Oswald, C. Plant, B. Wackersreuther, and P. Wackersreuther. ITCH: information-theoretic cluster hierarchies. In *Proc. ECML PKDD*, 2010.
13. T. Calders and B. Goethals. Non-derivable itemset mining. *Data Min. Knowl. Disc.*, 14(1):171–206, 2007.
14. C. H. Cheng, A. W.-C. Fu, and Y. Zhang. Entropy-based subspace clustering for mining numerical data. In *Proc. KDD*, pages 84–93, 1999.
15. X. H. Dang and J. Bailey. Generation of alternative clusterings using the CAMI approach. In *Proc. SDM*, 2010.
16. I. Davidson, S. S. Ravi, and L. Shamis. A SAT-based framework for efficient constrained clustering. In *Proc. SDM*, 2010.
17. T. De Bie. Maximum entropy models and subjective interestingness: an application to tiles in binary databases. *Data Min. Knowl. Disc.*, 2010.
18. M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proc. KDD*, 1996.
19. I. Färber, S. Günemann, H.-P. Kriegel, P. Kröger, E. Müller, E. Schubert, T. Seidl, and A. Zimek. On using class-labels in evaluation of clusterings. In *Proc. ACM SIGKDD Workshop MultiClust*, 2010.
20. D. Francois, V. Wertz, and M. Verleysen. The concentration of fractional distances. *IEEE TKDE*, 19(7):873–886, 2007.
21. F. Geerts, B. Goethals, and T. Mielikäinen. Tiling databases. In *Proc. DS’04*, pages 278–289, 2004.
22. A. Gionis, H. Mannila, T. Mielikäinen, and P. Tsaparas. Assessing data mining results via swap randomization. *ACM TKDD*, 1(3), 2007.
23. A. Gionis, H. Mannila, and J. K. Seppänen. Geometric and combinatorial tiles in 0-1 data. In *Proc. ECML PKDD’04*, 2004.
24. D. Gondek and T. Hofmann. Non-redundant data clustering. In *Proc. ICDM*, 2004.
25. P. D. Grünwald. *The Minimum Description Length Principle*. MIT Press, 2007.
26. S. Günemann, I. Färber, E. Müller, and T. Seidl. ASCLU: alternative subspace clustering. In *Proc. ACM SIGKDD Workshop MultiClust*, 2010.
27. S. Günemann, E. Müller, I. Färber, and T. Seidl. Detection of orthogonal concepts in subspaces of high dimensional data. In *Proc. CIKM*, 2009.
28. S. Hanhijärvi, M. Ojala, N. Vuokko, K. Puolamäki, N. Tatti, and H. Mannila. Tell me something I don’t know: randomization strategies for iterative data mining. In *Proc. KDD*, pages 379–388, 2009.

29. M. E. Houle, H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek. Can shared-neighbor distances defeat the curse of dimensionality? In *Proc. SSDBM*, 2010.
30. P. Jain, R. Meka, and I. S. Dhillon. Simultaneous unsupervised learning of disparate clusterings. *Stat. Anal. Data Min.*, 1(3):195–210, 2008.
31. K. Kailing, H.-P. Kriegel, and P. Kröger. Density-connected subspace clustering for high-dimensional data. In *Proc. SDM*, 2004.
32. A. J. Knobbe and E. K. Y. Ho. Pattern teams. In *Proc. PKDD*, 2006.
33. K.-N. Kontonasios and T. DeBie. An information-theoretic approach to finding informative noisy tiles in binary databases. In *Proc. SDM*, 2010.
34. H.-P. Kriegel, P. Kröger, and A. Zimek. Clustering high dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM TKDD*, 3(1):1–58, 2009.
35. H.-P. Kriegel, E. Schubert, and A. Zimek. Evaluation of multiple clustering solutions. In *Proc. ECML PKDD Workshop MultiClust*, 2011.
36. H.-P. Kriegel and A. Zimek. Subspace clustering, ensemble clustering, alternative clustering, multiview clustering: What can we learn from each other? In *Proc. ACM SIGKDD Workshop MultiClust*, 2010.
37. G. Liu, J. Li, K. Sim, and L. Wong. Distance based subspace clustering with flexible dimension partitioning. In *Proc. ICDE*, 2007.
38. T. M. Mitchell. Version spaces: A candidate elimination approach to rule learning. In *Proc. IJCAI*, 1977.
39. F. Moerchen, M. Thies, and A. Ultsch. Efficient mining of all margin-closed itemsets with applications in temporal knowledge discovery and classification by compression. *KAIS*, 2010.
40. G. Moise and J. Sander. Finding non-redundant, statistically significant regions in high dimensional data: a novel approach to projected and subspace clustering. In *Proc. KDD*, 2008.
41. E. Müller, I. Assent, S. Günemann, R. Krieger, and T. Seidl. Relevant subspace clustering: Mining the most interesting non-redundant concepts in high dimensional data. In *Proc. ICDM*, 2009.
42. E. Müller, I. Assent, R. Krieger, S. Günemann, and T. Seidl. DensEst: density estimation for data mining in high dimensional spaces. In *Proc. SDM*, 2009.
43. H.S. Nagesh, S. Goil, and A. Choudhary. Adaptive grids for clustering massive data sets. In *Proc. SDM*, 2001.
44. M. Ojala. Assessing data mining results on matrices with randomization. In *Proc. ICDM*, 2010.
45. M. Ojala, N. Vuokko, A. Kallio, N. Haiminen, and H. Mannila. Randomization methods for assessing data analysis results on real-valued matrices. *Stat. Anal. Data Min.*, 2(4):209–230, 2009.
46. N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal. Discovering frequent closed itemsets for association rules. In *Proc. ICDT*, 1999.
47. Z. J. Qi and I. Davidson. A principled and flexible framework for finding alternative clusterings. In *Proc. KDD*, 2009.
48. E. Segal, B. Taskar, A. Gasch, N. Friedman, and D. Koller. Rich probabilistic models for gene expression. *Bioinformatics*, 17(Suppl. 1):S243–S252, 2001.
49. J. Vreeken, M. van Leeuwen, and A. Siebes. Krimp: Mining itemsets that compress. *Data Min. Knowl. Disc.*, 23(1):169–214, 2010.
50. G. I. Webb. Discovering significant patterns. *Mach. Learn.*, 68(1):1–33, 2007.
51. X. Yan, H. Cheng, J. Han, and D. Xin. Summarizing itemset patterns: a profile-based approach. In *Proc. KDD*, 2005.