Clustering Ensemble on Reduced Search Spaces

Sandro Vega-Pons and Paolo Avesani

NeuroInformatics Laboratory (NILab), Fondazione Bruno Kessler, Trento, Italy
Centro Interdipartimentale Mente e Cervello (CIMeC), Università di Trento, Italy
{vega,avesani}@fbk.eu

Abstract. Clustering ensemble has become a very popular technique in the past few years due to its potentialities for improving the clustering results. Roughly speaking it consists in the combination of different partitions of the same set of objects in order to obtain a consensus one. A common way of defining the consensus partition is as the solution of the median partition problem. This way, the consensus partition is defined as the solution of a complex optimization problem. In this paper, we study possible prunes of the search space for this optimization problem. Particularly, we introduce a new prune that allows a dramatic reduction of the search space. We also give a characterization of the dissimilarity measures that can be used to take advantage of this prune and we proof that the *lattice metric* fits in this family. We carry out an experimental study comparing, under different circumstances, the size of the original search space and the size after the proposed prune. Outstanding reductions are obtained, which can be very beneficial for the development of clustering ensemble algorithms.

Keywords: Clustering ensemble, partition lattice, median partition, search space reduction, dissimilarity measure.

1 Introduction

Clustering ensemble has become a popular technique to deal with data clustering problems. When different clustering algorithms are applied to the same dataset, different clustering results can be obtained. Instead of trying to find the *best* one, the idea of combining these individual results in order to obtain a *consensus* has gained an increasing interest in the last years. In practice, such a procedure could produce high quality final clusterings.

In the past ten years, motivated by the success of the combination of supervised classifiers, several clustering ensemble algorithms have been proposed in the literature [1]. Different mathematical and computational tools have been used for the development of clustering ensemble algorithms. For example, there are methods based on Co-Association Matrix [2], Voting procedures [3], Genetic Algorithms [4], Graph Theory [5], Kernel Methods [6], Information Theory [7], Fuzzy Techniques [8], among others.

However, the consensus clustering, which is the final result of all clustering ensemble algorithms, is not always defined in the same way. For many methods, the consensus partition lacks of a formal definition, it is just implicitly defined as the objective function of the particular algorithm. This makes the theoretical study of the consensus partition properties to be difficult. This is the case, for example, of Relabeling and Voting [3], Graph based methods [5] and Co-association Matrix based methods [2]. On the other hand, there are some methods that use an explicit definition of the consensus partition concept. In this approach, the consensus partition is defined as the solution of an optimization problem, the problem of finding the *median partition* with respect to the clustering ensemble. Before defining this problem, we introduce the notation that will be used throughout this paper.

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of n objects and $\mathbb{P} = \{P_1, P_2, \ldots, P_m\}$ be a set of m partitions of X. A partition $P = \{C_1, C_2, \ldots, C_k\}$ of X is a set of k subsets of X (clusters) that satisfies the following properties:
(i) $C_i \neq \emptyset$, for all $i = 1, \ldots, k$;
(ii) $C_i \cap C_j = \emptyset$, for all $i \neq j$;
(iii) $\bigcup_{i=1}^{k} C_i = X$.

Furthermore, $\mathbb{P}_X$ is defined as the set of all possible partitions of X, $\mathbb{P} \subseteq \mathbb{P}_X$ and the *consensus partition* is denoted by P^*, $P^* \in \mathbb{P}_X$.

Formally, given an ensemble $\mathbb{P}$ of m partitions, the median partition is defined as:

$$P^* = \arg \min_{P \in \mathbb{P}_X} \sum_{i=1}^{m} d(P, P_i) \tag{1}$$

where d is a dissimilarity [1] measure between partitions.

Despite the median partition has been accepted in the clustering ensemble community, almost no studies about its theoretical properties have been done by scientists of this area. However, theoretical studies about the median partition problem have been carried out by the discrete mathematicians much before this problem gained interest in the machine learning community. Nevertheless, it has been mainly studied when it is defined by using the *symmetric difference distance* (or Mirkin distance) [9]. One of the most important results is the proof that the problem of finding the median partition with this distance is $\mathcal{NP}$-hard [10]. A proper analysis with other (dis)similarity measures has not been done.

Despite the complexity of the problem depends on the applied (dis)similarity measure, it seems to be a hard problem for any meaningful measure [1]. The application of an exhaustive search for the optimum solution, would only be computationally feasible for very small size problems. Therefore, several heuristic procedures have been applied to face this problem, for example: simulated annealing [6, 11] and genetic algorithms [12].

Despite the good results with these heuristics, they are still designed for finding the optimum solution in the whole search space. An interesting approach is to study the properties of the problem in order to find a possible prune of the search space, reducing the complexity. In some clustering ensemble algorithms,

[1] The problem can also be equally defined by maximizing the *similarity* with all partitions, in the case that d is a similarity measure.

an intuitive simplification of the problem, called *fragment clusters* [13, 14], has been used. The idea is that, if a subset of objects has been placed in the same cluster in all partitions to be combined, it is expected to find it in the consensus partition. Therefore, a representative object (fragment object) for each of those subsets can be computed. This way, the problem is reduced to work with the fragment objects. Once the consensus result is obtained, each fragment object is replaced back by the set of objects that it represents, in order to obtain the final consensus partition. This idea has also been used in the context of ensemble of image segmentations under the name of super-pixels [15, 16].

The above explained reduction needs objects to be placed in the same cluster for all partitions. As the number of partitions increases or partitions are more independent or some noisy partitions are included in the ensemble, the probability of having such subsets of objects with the same cluster label in all partitions decreases. Therefore, this prune of the search space could be useless in practice. Stronger prunes of the search space are needed for real applications.

In this paper, we introduce a new prune that leads to a dramatic reduction of the size of the search space. The paper is structured as follows, in Section 2 we present the basic concepts on lattice theory that are needed to introduce our results. In Section 3 a relation between the dissimilarity measure used to define the median partition problem and possible prunes of the search space is establish. First, a formalization of the fragment objects based prune is given by introducing the properties to be fulfilled by the dissimilarity measure. Afterwards, we introduce a new prune of the search space and provide a family of dissimilarity measures for which this prune is possible. Moreover, we present a measure that fits in this family, which can be used in practice to take advantage of the reduction of the search space. In Section 4, both prunes of the search space are experimentally evaluated on synthetic data. The size of the reduced search spaces is compared with the size of the whole search space under different conditions. Finally, Section 5 concludes this study.

2 Partition Lattice

The cardinality of the set of all partitions $\mathbb{P}_X$ is given by the $|X|$-th Bell number [17], which can be computed by the following recursion formula $B_{n+1} = \sum_{k=0}^{n} \binom{n}{k} B_k$. The Bell number has an exponential growing as the number of objects increases, e.g. $B_3 = 5$, $B_{10} = 115975$ and $B_{100} = 4.75 \times 10^{115}$ (Much bigger than the estimation of the number of all atoms in the observable universe, around 10^{80}) [2]. Therefore, even for a relatively small number of objects, the set of all partitions of them $\mathbb{P}_X$ is huge.

Over $\mathbb{P}_X$, a partial order relation[3] $\preceq$ (called *refinement*) can be defined. For all $P, P' \in \mathbb{P}_X$, we say $P \preceq P'$ if and only if, for all cluster $C' \in P'$ there are

[2] http://www.universetoday.com/36302/atoms-in-the-universe/

[3] A binary relation which is *reflexive* ($P \preceq P$), *anti-symmetric* ($P \preceq P'$ and $P' \preceq P$ implies $P = P'$) and *transitive* ($P \preceq P'$ and $P' \preceq P''$ implies $P \preceq P''$) for all $P, P', P'' \in \mathbb{P}_X$.

clusters $C_{i_1}, C_{i_2}, \ldots, C_{i_v} \in P$ such that $C' = \bigcup_{j=1}^{v} C_{i_j}$. In this case, it is said that P is *finer* than P' or equivalently, P' is *coarser* than P.

The set of all partitions $\mathbb{P}_X$ of a finite set X, endowed with the *refinement* order ($\preceq$) is a *lattice* (see example in Fig. 1). Therefore, for each pair of partitions P, P' two binary operations are defined: *meet* ($P \wedge P'$) which is the coarsest of all partitions finer than both P and P', and *join* ($P \vee P'$) which is the finest of all partitions coarser than both P and P'.

For example, in Fig. 1, if $P_1 = \{\{a,b,c,d\}\}$, $P_2 = \{\{a,b\},\{c,d\}\}$, $P_3 = \{\{a,b,c\},\{d\}\}$, $P_4 = \{\{a,b\},\{c\},\{d\}\}$, then: $P_4 \preceq P_2 \preceq P_1$; $P_2 \wedge P_3 = P_4$ and $P_2 \vee P_3 = P_1$.

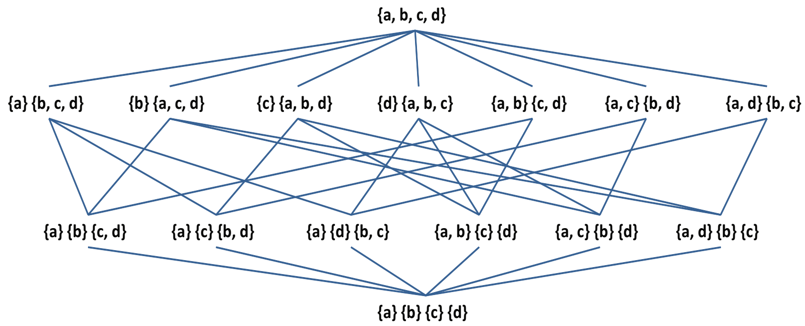

Fig. 1. Hasse diagram or graphical representation of the lattice associated to the set of partitions $\mathbb{P}_X$ of the set $X = \{a,b,c,d\}$.

Among several properties, the partition lattice $(\mathbb{P}_X, \preceq)$ satisfies the property of being an *atomic lattice*. The partitions P_{xy}, composed of a cluster containing only the objects x and y, and the remaining clusters containing only one object, are called *atoms*. For example, in Figure 1, $P_{ab} = \{\{a,b\},\{c\},\{d\}\}$ and $P_{bc} = \{\{a\},\{b,c\},\{d\}\}$ are two atoms. The partition lattice is atomic because, every partition P is the join of the elementary partitions P_{xy} for all pair of objects x, y which are in the same cluster in P.

An important concept that is needed to understand the results proposed in this paper is the *q-quota* rules [18]. Given a real number $q \in [0,1]$, the *q-quota* rule c_q is defined in the following way:

$$c_q(\mathbb{P}) = \bigvee \{P_{xy} : \gamma(xy, \mathbb{P}) > q\} \tag{2}$$

where $\gamma(xy, \mathbb{P}) = \frac{N(xy,\mathbb{P})}{m}$ and $N(xy, \mathbb{P})$ is the number of times that the objects $x, y \in X$ are in the same cluster in the partitions in $\mathbb{P}$.

Two interesting cases of *q-quota* rules are the following:

- **unanimity rule**: $u(\mathbb{P}) = \bigvee \{P_{xy} : \gamma(xy, \mathbb{P}) = 1\}$
- **majority rule**: $m(\mathbb{P}) = \bigvee \{P_{xy} : \gamma(xy, \mathbb{P}) > 0.5\}$

Notice that any *q-quota* rule is a partition of $\mathbb{P}_X$. For example: $u(\mathbb{P})$ is the partition obtained by the join of all atoms P_{xy}, such that the objects x and y are placed in the same cluster in all partitions in $\mathbb{P}$. In the same way, $m(\mathbb{P})$ is the join of all atoms P_{xy} such that x and y are in the same cluster in more than half of the partitions in $\mathbb{P}$. Next we present a toy example.

Example 1. Let $X = \{1, 2, 3, 4, 5, 6\}$ be a set of objects and $\mathbb{P} = \{P_1, P_2, P_3, P_4, P_5\}$ be a set of partitions of X, such that:
$P_1 = \{\{1, 2\}, \{3, 4\}, \{5, 6\}\}$, $P_2 = \{\{1, 2, 4\}, \{3, 5, 6\}\}$, $P_3 = \{\{1, 2, 3\}, \{4, 5, 6\}\}$, $P_4 = \{\{1, 3, 4\}, \{2, 5, 6\}\}$, $P_5 = \{\{1, 3\}, \{2, 4\}, \{5, 6\}\}$.

In this case $u(P) = \{\{1\}, \{2\}, \{3\}, \{4\}, \{5, 6\}\}$ since 5 and 6 are the only elements that are grouped in the same cluster in all partitions. On the other hand, $m(\mathbb{P}) = \{\{1, 2, 3\}, \{4\}, \{5, 6\}\} = P_{12} \vee P_{13} \vee P_{56}$. Notice that objects 2 and 3 are in the same cluster in $m(\mathbb{P})$ even though P_{23} is not a majority atom, i.e. $\gamma(23, \mathbb{P}) = 1/5 < 0.5$. This is a chaining effect of the fact that P_{12} and P_{13} are majority atoms.

3 Methods

The two rules defined in the previous Section (unanimity and majority rules) allow the definition of two different subsets of the partition space $\mathbb{P}_X$. Let us consider $\mathbb{U}_X \subseteq \mathbb{P}_X$ the set of all partitions coarser than $u(\mathbb{P})$, i.e. $\mathbb{U}_X = \{P \in \mathbb{P}_X : u(\mathbb{P}) \preceq P\}$. Analogously, $\mathbb{M}_X \subseteq \mathbb{P}_X$ is defined as the set of all partitions coarser than $m(\mathbb{P})$, i.e. $\mathbb{M}_X = \{P \in \mathbb{P}_X : m(\mathbb{P}) \preceq P\}$. It is not difficult to verify that $\mathbb{M}_X \subseteq \mathbb{U}_X \subseteq \mathbb{P}_X$, because any atom P_{xy} satisfying $P_{xy} \preceq u(\mathbb{P})$ also holds $P_{xy} \preceq m(\mathbb{P})$.

In this section we will describe the conditions such that the median partition can be searched just in the reduced spaces $\mathbb{U}_X$ and $\mathbb{M}_X$. The median partition problem could have more than one solution, therefore equation (1) should be written in a more precise way as follows:

$$\mathcal{M}_{\mathbb{P}} = \arg \min_{P \in \mathbb{P}_X} \sum_{i=1}^{m} d(P, P_i) \tag{3}$$

where $\mathcal{M}_{\mathbb{P}}$ is the set of all median partitions. If we only consider the reduced search space $\mathbb{U}_X$, the median partition problem is defined as:

$$\mathcal{M}_{\mathbb{U}} = \arg \min_{P \in \mathbb{U}_X} \sum_{i=1}^{m} d(P, P_i) \tag{4}$$

In the same way, when only $\mathbb{M}_X$ is considered as search space, the median partition problem is given by:

$$\mathcal{M}_{\mathbb{M}} = \arg \min_{P \in \mathbb{M}_X} \sum_{i=1}^{m} d(P, P_i) \tag{5}$$

Another concept that we will use is the *sum-of-dissimilarities* (SoD):

Definition 1. *Given a set of partitions $\mathbb{P} \subset \mathbb{P}_X$ and a dissimilarity $d : \mathbb{P}_X \times \mathbb{P}_X \to \mathbb{R}$, the sum-of-dissimilarities of a partition P to $\mathbb{P}$ ($SoD(P)$) is defined as:*

$$SoD(P) = \sum_{i=1}^{m} d(P, P_i)$$

Notice that a median partition P^* is an element of $\mathbb{P}_X$ with a minimum SoD value, i.e. $P^* = \arg\min_{P \in \mathbb{P}_X} SoD(P)$.

In Section 3.1 we present a family of dissimilarity functions for which $\mathcal{M}_{\mathbb{P}} = \mathcal{M}_{\mathbb{U}}$ and therefore the reduced search space $\mathbb{U}_X$ can be used instead of $\mathbb{P}_X$. In Section 3.2 a family of dissimilarity functions for which $\mathcal{M}_{\mathbb{P}} = \mathcal{M}_{\mathbb{M}}$ is also presented. We also prove that the *lattice metric* belongs to this family of functions.

3.1 Prune of the search space based on Unanimity Rule

Definition 2. *A dissimilarity measure between partitions $d_u : \mathbb{P}_X \times \mathbb{P}_X \to \mathbb{R}$ is said to be* u-atomic*, if for every pair of partitions $P, P' \in \mathbb{P}_X$ and every atom P_{xy} such that $P_{xy} \npreceq P$ and $P_{xy} \preceq P'$, then $d(P \vee P_{xy}, P') < d(P, P')$.*

Proposition 1. *Let $\mathbb{P} \subset \mathbb{P}_X$ be a set of partitions and d_u be an* u-atomic *dissimilarity function. For all partition $P \in \mathbb{P}_X$ and every atom P_{xy} such that $P_{xy} \npreceq P$ and $P_{xy} : \gamma(xy, \mathbb{P}) = 1$, we have:*

$$SoD(P \vee P_{xy}) < SoD(P)$$

Proof. $SoD(P) = d_u(P, P_1) + \ldots + d_u(P, P_m)$ for all $P_i \in \mathbb{P}$, with $i = 1, \ldots, m$. In the same way, $SoD(P \vee P_{xy}) = d_u(P \vee P_{xy}, P_1) + \ldots + d_u(P \vee P_{xy}, P_m)$.

As $\gamma(xy, \mathbb{P}) = 1$, all partitions $P_i \in \mathbb{P}$ hold $P_{xy} \preceq P_i$. For each term in both equations we have $d_u(P \vee P_{xy}, P_i) < d_u(P, P_i)$ based on the definition of *u-atomic* dissimilarity and therefore $SoD(P \vee P_{xy}) < SoD(P)$. □

Proposition 2. *Let $\mathbb{P} \subset \mathbb{P}_X$ be a set of partitions, $u(\mathbb{P})$ be the unanimity rule and d_u be an* u-atomic *function. Every median partition $P^* \in \mathcal{M}_{\mathbb{P}}$ holds $u(\mathbb{P}) \preceq P^*$.*

Proof. Let us assume that $P^* \in \mathcal{M}_{\mathbb{P}}$ is a median partition and $u(\mathbb{P}) \npreceq P^*$. Then, there is at least one atom $P_{xy} \preceq u(\mathbb{P})$ such that $P_{xy} \npreceq P^*$. According to Proposition 3, P^* would not be a median element because the partition $P^* \vee P_{xy}$ would have a smaller SoD value. Therefore, the assumption is false and we conclude that $u(\mathbb{P}) \preceq P^*$ □

Corollary 1. *Let $\mathbb{P} \subset \mathbb{P}_X$ be a set of partitions and d_u be an* atomic *function, then problems* (3) *and* (4) *have the same set of solutions ($\mathcal{M}_{\mathbb{P}} = \mathcal{M}_{\mathbb{U}}$).*

Proof. This is a direct consequence of Proposition 2 and equations (3) and (4). If all solutions of equation (3) are coarser than $m(\mathbb{P})$, they are in $\mathbb{U}_X$. Therefore, problems (3) and (4) are equivalent. □

In practice, there is a simple way to reduce the search space from $\mathbb{P}_X$ to $\mathbb{U}_X$. First, $u(\mathbb{P})$ should be computed and for each cluster a representative element y_i is defined. This way, a set of objects Y with $|Y| \leq |X|$ is obtained, and the corresponding set of partitions $\mathbb{P}_Y$ will be equivalent to $\mathbb{U}_X$. This is exactly the idea of *fragment clusters*. As we have previously mentioned, this idea has been intuitively used before and have also been proven to be valid for some common dissimilarity measures [13] such as: Mutual Information and Mirkin distance. In this section, we have presented the notion of *u-atomic* function and we have proven that for any *u-atomic* dissimilarity measure this prune of the search space can be used.

3.2 Prune of the search space based on the Majority Rule

Definition 3. *A dissimilarity measure between partitions* $d_m : \mathbb{P}_X \times \mathbb{P}_X \to \mathbb{R}$ *is said to be* m-atomic*, if for every pair of partitions* $P, P' \in \mathbb{P}_X$ *and every atom* P_{xy} *such that* $P_{xy} \npreceq P$*, there is a constant real value* $c > 0$ *such that the following properties hold:*

- *(i) if* $P_{xy} \preceq P'$*, then* $d_m(P \vee P_{xy}, P') \leq d(P, P') - c$
- *(ii) if* $P_{xy} \npreceq P'$*, then* $d_m(P \vee P_{xy}, P') \leq d(P, P') + c$

Notice that following this definition any *m-atomic* function is also *u-atomic*.

Proposition 3. *Let* $\mathbb{P} \subset \mathbb{P}_X$ *be a set of partitions and* d_m *be an* m-atomic *function. For all partition* $P \in \mathbb{P}_X$ *and every atom* P_{xy} *such that* $P_{xy} \npreceq P$ *and* $P_{xy} : \gamma(xy, \mathbb{P}) > 0.5$*, we have:*

$$SoD(P \vee P_{xy}) < SoD(P)$$

Proof. $SoD(P) = d_m(P, P_1) + \ldots + d_m(P, P_m)$ for all $P_i \in \mathbb{P}$, with $i = 1, \ldots, m$. In the same way, $SoD(P \vee P_{xy}) = d_m(P \vee P_{xy}, P_1) + \ldots + d_m(P \vee P_{xy}, P_m)$.

As $\gamma(xy, \mathbb{P}) > 0.5$ there are $t > m/2$ partitions $P_i \in \mathbb{P}$ such that $P_{xy} \preceq P_i$, and therefore according to definition 3, $d_m(P \vee P_{xy}, P_i) \leq d_m(P, P_i) - c$. On the other hand, there are $l < m/2$ partitions $P_j \in \mathbb{P}$ such that $P_{xy} \npreceq P_j$ and $d_m(P \vee P_{xy}, P') \leq d_m(P, P') + c$.

Therefore, $SoD(P \vee P_{xy}) \leq SoD(P) - t \cdot c + l \cdot c$, and taking into account that $t > l$ and $c > 0$ we have that: $SoD(P \vee P_{xy}) < SoD(P)$ and the proposition is proven. □

Proposition 4. *Let* $\mathbb{P} \subset \mathbb{P}_X$ *be a set of partitions,* $m(\mathbb{P})$ *be the majority rule and* d_m *be a* m-atomic *function. Every median partition* $P^* \in \mathcal{M}_{\mathbb{P}}$ *holds* $m(\mathbb{P}) \preceq P^*$*.*

Proof. The proof is analogous to the one for Proposition 2. Let us assume that $P^* \in \mathcal{M}_{\mathbb{P}}$ is a median partition and $m(\mathbb{P}) \npreceq P^*$. Then, there is at least one atom $P_{xy} \preceq m(\mathbb{P})$ such that $P_{xy} \npreceq P^*$. According to Proposition 3, P^* would not be a median element because the partition $P^* \vee P_{xy}$ would have a smaller SoD value. Therefore, the assumption is false and we conclude that $m(\mathbb{P}) \preceq P^*$ □

Corollary 2. *Let $\mathbb{P} \subset \mathbb{P}_X$ be a set of partitions and d_m be a* m-atomic *function, then problems* (3) *and* (5) *have the same set of solutions ($\mathcal{M}_\mathbb{P} = \mathcal{M}_\mathbb{M}$).*

Proof. This is a direct consequence of Proposition 4 and equations (3) and (5). □

We have proven that if the median partition problem is defined with a *m-atomic* function, any solution of the problem will be found in the reduced search space $\mathbb{M}_X$. As in the case of the fragment clusters prune, there is a simple way to reduce the search space from $\mathbb{P}_X$ to $\mathbb{M}_X$. In this case, $m(\mathbb{P})$ should be first computed and for each cluster a representative element y_i is defined. This way, a set of objects Y with $|Y| \leq |X|$ is obtained, and the corresponding set of partitions $\mathbb{P}_Y$ will be equivalent to $\mathbb{M}_X$.

So far, we have presented the notion of *m-atomic* function and we have proven that for any *m-atomic* dissimilarity measure this prune of the search space can be applied. Now, we present an existing distance between partitions and we prove that it is *m-atomic*.

Definition 4. *(Lattice Metric [18]) The function $\delta : \mathbb{P}_X \times \mathbb{P}_X \to \mathbb{R}$ defined as $\delta(P, P') = |P| + |P'| - 2|P \vee P'|$, where $|P|$ denotes the number of clusters in partition P, is called* lattice metric.

Proposition 5. *The lattice metric $\delta : \mathbb{P}_X \times \mathbb{P}_X \to \mathbb{R}$ is* m-atomic.

Proof. Let $P, P_{xy}, P_{zt}, P' \in \mathbb{P}_X$ be 4 partitions such that P_{xy} and P_{zt} are two atoms holding $P_{xy} \npreceq P$, $P_{xy} \preceq P'$; and $P_{zt} \npreceq P$, $P_{zt} \npreceq P'$. We have to proof that there is a constant c value such that:

(i) $\delta(P \vee P_{xy}, P') \leq \delta(P, P') - c$, and *(ii)* $\delta(P \vee P_{zt}, P') \leq \delta(P, P') + c$

Working on *(i)*, we have:

$$|P \vee P_{xy}| + |P'| - 2|P \vee P_{xy} \vee P'| \leq |P| + |P'| - 2|P \vee P'| - c$$

as $P_{xy} \preceq P'$, then $P \vee P_{xy} \vee P' = P \vee P'$. Therefore:

$$|P \vee P_{xy}| \leq |P| - c$$

as $P_{xy} \npreceq P$ we have $|P \vee P_{xy}| = |P| - 1$ because $|P \vee P_{xy}|$ means the join of two clusters in $|P|$. Thus, we obtain

$$c \leq 1$$

Now, working on *(ii)*:

$$|P \vee P_{zt}| + |P'| - 2|P \vee P_{tz} \vee P'| \leq |P| + |P'| - 2|P \vee P'| + c$$

$$-1 - 2|P \vee P_{tz} \vee P'| \leq -2|P \vee P'| + c$$

$$c \geq 2|P \vee P'| - 1 - 2|P \vee P_{tz} \vee P'|$$

the right-hand side of the inequality takes the higher value when $P_{zt} \npreceq P \vee P'$, and in this case $|P \vee P_{tz} \vee P'| = |P \vee P'| - 1$. Therefore,

$$c \geq 2|P \vee P'| - 1 - 2(|P \vee P'| - 1)$$

$$c \geq 1$$

Taking into account both results, we conclude that δ is a m-atomic function with $c = 1$. □

This means that if the median partition problem is defined with the *lattice metric* δ the search space of the problem is reduced to $\mathbb{M}_X$. Notice that this metric corresponds to the minimum path length metric in the neighboring graph of the lattice (see Figure 1), when all edges have a weight equal to 1 [18]. Therefore, we could say that this is an informative measure to compare partitions that takes into account the lattice structure of the partition space. Furthermore, it allows a pruning of the search space for the median partition problem.

4 Experimental Results and Discussion

Let $X = \{x_1, \ldots, x_n\}$ be a set of n objects where $x_i \in \mathbb{R}^d$ is a vector in multidimensional space. We assume that each dimension of the vector $x_i = (x_{i,1}, \ldots, x_{i,d})$ is drawn from the uniform distribution $x_{i,j} \sim U(0,1)$ and are mutually independent. We generated synthetic datasets for $d = 3$ and $n = 1000 (= 10^3)$, $3375 (= 15^3)$, $8000 (= 20^3)$, $15625 (= 25^3)$, $27000 (= 30^3)$. Objects in the datasets lie inside a 3-dimensional cube starting at the origin of the cartesian coordinates system and with edge length of 1.

In order to generate different partitions of this dataset, we use a simple clustering algorithm based on cuts of the cube by random hyperplanes. Furthermore, we model different dependency between partitions in the ensemble by clustering the dataset taking into account different subsets of the dimensions of the objects representation.

We carry out three different kinds of experiments to illustrate the behavior of the proposed method for pruning the search space. In Section 4.1 we work with different dataset sizes (n) and three different levels of dependency between partitions. In Section 4.2, we vary the number of clusters (k) in the partitions to be combined and finally, in Section 4.3, we carry out experiments with different amount of partitions (m) in the cluster ensemble. For all experiments we show:

- $|X|$: Number of objects in the dataset. $|X| = n$.
- $|\mathbb{P}_X|$: Size of the original search space for the median partition problem.
- $|u(\mathbb{P})|$: Number of clusters in the unanimity rule partition.
- $|\mathbb{U}_X|$: Size of the search space after applying the unanimity rule based prune (fragment clusters based prune).
- $|m(\mathbb{P})|$: Number of clusters in the majority rule partition.
- $|\mathbb{M}_X|$: Size of the search space after applying the majority rule based prune (the proposed prune).

In all tables, the sizes of the search spaces are given in powers of 10. This way, the order-of-magnitude differences among the sizes of the different search spaces can be easily appreciated. In order to provide an uniform notation, even the small values are given in powers of ten, e.g. if $|\mathbb{P}_X| = 203$ we will write 10^2.

Results reported in all tables correspond to the median values of individual results after 5 repetitions.

4.1 Analysis by increasing the number of objects and varying the independence degree

In this section we compare the sizes of $|\mathbb{P}_X|$, $|\mathbb{U}_X|$ and $|\mathbb{M}_X|$ for different dataset sizes. We generated 10 partitions, where each partition has a number of clusters equal to a random number in the interval $[2, n/2]$. In Table 1, only the first feature dimension of the objects was used for the computation of partitions. The idea is to generate partitions with high degree of dependency, i.e. partitions with similar distribution of objects in the clusters. In the case of Table 2, a medium degree of dependency is explore by using features $d = 0, 1$. Finally, in Table 3 all features $d = 0, 1, 2$ are used to analyze the behavior of the prunes in the case of ensembles with highly independent partitions.

Table 1. Comparison of $|\mathbb{P}_X|$, $|\mathbb{U}_X|$ and $|\mathbb{M}_X|$ for different dataset sizes $|X|$. Partitions were generated with a high degree of dependency ($d = 0$). We use $m = 10$, and for each partition $k = random(2, n/2)$. Results are the average of 5 trials.

$\|X\|$	1000	3375	8000	15625	27000
$\|\mathbb{P}_X\|$	10^{1928}	10^{7981}	10^{21465}	10^{45847}	10^{84822}
$\|u(\mathbb{P})\|$	10	15	20	25	30
$\|\mathbb{U}_X\|$	10^{6}	10^{10}	10^{14}	10^{19}	10^{24}
$\|m(\mathbb{P})\|$	6	9	10	10	14
$\|\mathbb{M}_X\|$	10^{2}	10^{4}	10^{5}	10^{5}	10^{8}

Table 2. Comparison of $|\mathbb{P}_X|$, $|\mathbb{U}_X|$ and $|\mathbb{M}_X|$ for different dataset sizes $|X|$. Partitions were generated with a medium degree of dependency ($d = 0, 1$). We use $m = 10$, and for each partition $k = random(2, n/2)$. Results are the average of 5 trials.

$\|X\|$	1000	3375	8000	15625	27000
$\|\mathbb{P}_X\|$	10^{1928}	10^{7981}	10^{21465}	10^{45847}	10^{84822}
$\|u(\mathbb{P})\|$	98	195	345	558	689
$\|\mathbb{U}_X\|$	10^{113}	10^{268}	10^{539}	10^{963}	10^{1239}
$\|m(\mathbb{P})\|$	6	7	14	15	25
$\|\mathbb{M}_X\|$	10^{2}	10^{3}	10^{9}	10^{10}	10^{19}

Table 3. Comparison of $|\mathbb{P}_X|$, $|\mathbb{U}_X|$ and $|\mathbb{M}_X|$ for different dataset sizes $|X|$. Partitions were generated with a low degree of dependency ($d = 0, 1, 2$). We use $m = 10$, and for each partition $k = random(2, n/2)$. Results are the average of 5 trials.

$\|X\|$	1000	3375	8000	15625	27000
$\|\mathbb{P}_X\|$	10^{1928}	10^{7981}	10^{21465}	10^{45847}	10^{84822}
$\|u(\mathbb{P})\|$	777	2094	5877	7122	9014
$\|\mathbb{U}_X\|$	10^{1429}	10^{4589}	10^{15096}	10^{18801}	10^{24587}
$\|m(\mathbb{P})\|$	12	35	43	50	71
$\|\mathbb{M}_X\|$	10^{7}	10^{30}	10^{39}	10^{48}	10^{75}

From this experiment we can appreciate the following:

- The cardinality of $\mathbb{M}_X$ is always considerable lower than the cardinality of $\mathbb{U}_X$ and the cardinality of $\mathbb{U}_X$ is also always much lower than the cardinality of $\mathbb{P}_X$.
- As the number of elements in the dataset increases the size of all search spaces also increases. However, $|\mathbb{P}_X|$ grows faster than $|\mathbb{U}_X|$ and at the same time $|\mathbb{U}_X|$ grows faster than $|\mathbb{M}_X|$.
- Increasing the independence in the partitions in the ensemble, the cardinality of the resulting search spaces after both prunes also increases. The higher the dependency between partitions, the higher the probability of finding groups of objects that were placed in the same cluster in all partitions or in more than half of the partitions.
- Despite the original search space $\mathbb{P}_X$ is huge in all cases, the reduced search space after the majority rule based prune $\mathbb{M}_X$ is sometimes very small. In this case, the exact solution of the median partition problem could be even found by following an exhaustive search. On the other hand, the reduced search space after the unanimity rule prune $\mathbb{U}_X$ is, many times, too big to be useful in practice.

4.2 Analysis increasing number of clusters in the partitions

In this section we used a dataset of size $10 \times 10 \times 10 = 1000$. The three dimensions of each object are taken into account for the generation of the partitions in the ensemble. We generated different ensembles of $m = 10$ partitions with different number of clusters $k = 5, 20, 50, 100, 200$. The results of this experiment are reported in Table 4.

Table 4. Comparison of $|\mathbb{P}_X|$, $|\mathbb{U}_X|$ and $|\mathbb{M}_X|$ when the number of clusters k in the partitions is increased. Partitions were generated by using the full representation of objects $d = 1, 2, 3$. The dataset size is 1000 and we use $m = 10$ with k clusters.

k	5	20	50	100	200
$\|X\|$	1000	1000	1000	1000	1000
$\|\mathbb{P}_X\|$	10^{21465}	10^{21465}	10^{21465}	10^{21465}	10^{21465}
$\|u(\mathbb{P})\|$	41	161	523	647	769
$\|\mathbb{U}_X\|$	10^{37}	10^{211}	10^{891}	10^{1149}	10^{1412}
$\|m(\mathbb{P})\|$	1	3	10	31	115
$\|\mathbb{M}_X\|$	10^{0}	10^{1}	10^{6}	10^{26}	10^{139}

From Table 4 we can see that the size of the reduced search spaces increases together with the number of clusters in the partitions to be combined. This is expected, because the higher the number of clusters, the less the probability of finding groups of objects that are placed in the same cluster in all partitions or more than half of the partitions. Furthermore, when a small number of clusters (w.r.t the number of objects) is used, the reduction of the search space after the majority rule based prune is too big. The median partition could have very few clusters or even one cluster. This is a consequence of the chaining effect illustrated in Example 1. This kind of medians could be useless in practical applications.

4.3 Analysis increasing the number of partitions

In this section we used a dataset of size $10\times10\times10 = 1000$. The three dimensions of each object are taken into account for the generation of the partitions in the ensemble. We generated ensembles with different number of partitions $m = 5, 10, 20, 50, 100$, where each partition has a number of clusters equal to a random number in the interval $[2, n/2]$. The results of this experiment are reported in Table 5.

Table 5. Comparison of $|\mathbb{P}_X|$, $|\mathbb{U}_X|$ and $|\mathbb{M}_X|$ when the number of partitions k in the ensemble is increased. Partitions were generated by using the full representation of objects $d = 1, 2, 3$. The dataset size is 1000 and we generate m partitions, each one with $k = random(2, n/2)$ clusters.

m	5	10	20	50	100
$\lvert X\rvert$	1000	1000	1000	1000	1000
$\lvert\mathbb{P}_X\rvert$	10^{21465}	10^{21465}	10^{21465}	10^{21465}	10^{21465}
$\lvert u(\mathbb{P})\rvert$	601	694	789	772	746
$\lvert\mathbb{U}_X\rvert$	10^{1052}	10^{1249}	10^{1456}	10^{1418}	10^{1362}
$\lvert m(\mathbb{P})\rvert$	27	14	13	6	1
$\lvert\mathbb{M}_X\rvert$	10^{21}	10^{9}	10^{8}	10^{2}	10^{0}

While the size of the search space after unanimity rule based prune $|\mathbb{U}_X|$ remains stable, the cardinality of $\mathbb{M}_X$ decreases as the number of partitions in the ensemble increases. The higher the number of partitions in the ensemble, the higher the probability of finding groups of objects that are placed in the same cluster in more than half of the partitions. However, this reduction could be sometimes too big such that the resulting median partition has few clusters or just one. This could be inappropriate in practical applications.

5 Conclusions

We studied two possible reductions of the search space for the median partition problem. In the first case, we introduced a family of functions that allow the application of the fragment clusters based prune. This prune have been used in an intuitive manner or with a few measures for which the suitability of this prune has been proven. A characterization of the measures that allow this prune is presented. Furthermore, we introduced a stronger prune of the search space for the median partition problem. In this case, we also presented a family of dissimilarity measures that allow the application of this prune and we proved that the lattice metric fits in this family.

The proposed prune is able to do a dramatic reduction of the search space. Even for relatively big number of objects, for which the original search space is really huge, the reduced search space is many times small enough such that the median partition can be found by an exhaustive search. Even in the cases when the reduced search space is still big, any heuristic procedure could take advantage of the strong reduction with respect to the original size of the space.

Despite this prune can be beneficial in several problems, sometimes the median partition defined with a function that allows this prune, has a small number of clusters. In some extreme cases, it could even be just one cluster, making this kind of consensus useless. In practice, this limitation could be smoothed by generating an ensemble of partitions with a high number of clusters, which will be reduced in the consensus partition computation. This idea has been previously used in the clustering ensemble context [19].

The advantages of the proposed prune from the computational point of view are clear in our experiments with synthetic data. A further step would be to analyze the quality of the median partition obtained by this method on real datasets.

The two studied prunes correspond to two particular cases of the q-quota rules presented in Section 2: unanimity ($q = 1$) and majority ($q = 0.5$). The first one leads to a commonly weak reduction of the search space, while the second prune could be too strong sometimes. A possible good trade-off could be found for prunes associated to other quota rules, e.g. $q = 2/3$ or $3/4$. A characterization of the dissimilarity measures between partitions that would allow this kind of prunes is worth to be study.

Acknowledgments. This research has been supported by the RESTATE Programme, co-funded by the European Union under the FP7 COFUND Marie Curie Action - Grant agreement no. 267224.

References

1. Vega-Pons, S., Ruiz-Shulcloper, J.: A survey of clustering ensemble algorithms. International Journal of Pattern Recognition and Artifcial Intelligence. **25 (3)** (2011) 337 – 372
2. Fred, A.L., Jain, A.K.: Combining multiple clustering using evidence accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence **27** (2005) 835–850
3. Ayad, H.G., Kamel, M.S.: On voting-based consensus of cluster ensembles. Pattern Recognition **43** (2010) 1943 – 1953
4. Yoon, H.S., Ahn, S.Y., Lee, S.H., Cho, S.B., Kim, J.H.: Heterogeneous clustering ensemble method for combining different cluster results. In: BioDM 2006. Volume 3916 of LNBI. (2006) 8292

5. Strehl, A., Ghosh, J.: Cluster ensembles: a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. **3** (2002) 583–617
6. Vega-Pons, S., Correa-Morris, J., Ruiz-Shulcloper, J.: Weighted partition consensus via kernels. Pattern Recognition **43(8)** (2010) 2712–2724
7. Topchy, A.P., Jain, A.K., Punch, W.F.: Clustering ensembles: Models of consensus and weak partitions. IEEE Trans. Pattern Anal. Mach. Intell. **27**(12) (2005) 1866–1881
8. Punera, K., Ghosh, J.: Consensus-based ensembles of soft clusterings. Applied Artificial Intelligence **22**(7&8) (2008) 780–810
9. Mirkin, B.G.: Mathematical Classification and Clustering. Kluwer Academic Press, Dordrecht (1996)
10. Wakabayashi, Y.: Aggregation of Binary Relations: Algorithmic and Polyhedral Investigations. PhD thesis, Universitat Augsburg (1986)
11. Filkov, V., Skiena, S.: Integrating microarray data by consensus clustering. International Journal on Artificial Intelligence Tools **13(4)** (2004) 863 – 880
12. Luo, H., Jing, F., Xie, X.: Combining multiple clusterings using information theory based genetic algorithm,. In: IEEE International Conference on Computational Intelligence and Security. Volume 1. (2006) 84 – 89
13. Wu, O., Hu, W., Maybank, S.J., Zhu, M., Li, B.: Efficient Clustering Aggregation Based on Data Fragments. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **42**(3) (2012) 913–926
14. Chung, C.H., Dai, B.R.: A fragment-based iterative consensus clustering algorithm with a robust similarity. Knowledge and Information Systems (2013) 1–19
15. Singh, V., Mukherjee, L., Peng, J., Xu, J.: Ensemble clustering using semidefinite programming with applications. Machine Learning **79** (2010) 177 200
16. Vega-Pons, S., Jiang, X., Ruiz-Shulcloper, J.: Segmentation ensemble via kernels. In: First Asian Conference on Pattern Recognition (ACPR), 2011. 686–690
17. Spivey, M.Z.: A generalized recurrence for bell numbers. Journal of Integer Sequences **11** (2008) 1 – 3
18. Leclerc, B.: The median procedure in the semilattice of orders. Discrete Applied Mathematics **127** (2003) 285 – 302
19. Fred, A.: Finding consistent clusters in data partitions. In: 3rd. Int. Workshop on Multiple Classifier Systems. (2001) 309–318