"Anti–Bayesian" Flat and Hierarchical Clustering Using Symmetric Quantiloids[1]

Hugo Lewi Hammer[2], Anis Yazidi[3], B. John Oommen[4]

Abstract

A Pattern Recognition (PR) system that does not involve labelled samples requires the clustering of the samples into their respective classes before the training and testing can be achieved. All of the reported clustering algorithms (except the one reported in Hammer et al. (2015)) operate on Bayesian principles, which is understandable because these principles constitute the basis of *optimal* PR. Recently, Oommen and his co-authors have proposed a novel, counter-intuitive and pioneering PR scheme that is radically opposed to the Bayesian principle. The rational for this paradigm, referred to as the "Anti-Bayesian" (AB) paradigm, involves classification based on the non-central *quantiles* of the distributions. This paper, extends the results of Hammer et al. (2015) in many directions. Firstly, we generalize our previous AB clustering Hammer et al. (2015) to handle arbitrary d-dimensional spaces using so-called *"quantiloids"*. Secondly, we extend the AB paradigm to consider how the clustering can be achieved in hierarchical ways, where we analyze both the Top-Down and the Bottom-Up clustering options. Extensive experimentation, on artificial and on challenging real-life data, demonstrates that our clustering achieves results competitive to the state-of-the-art flat, Top-Down and Bottom-Up clustering approaches, demonstrating the power of the AB paradigm.

Keywords: anti-bayes, bayesian principle, clustering, quantiloids

[1] A preliminary version of this paper can be found in the *Proceedings of IEA/AIE'16, the 2016 International Conference on Industrial, Engineering & Other Applications of Applied Intelligent Systems, Japan, August 2016.*

[2] Author's status: *Associate Professor.* This author can be contacted at: Oslo and Akershus University College, Department of Computer Science, Pilestredet 35, Oslo, Norway. E-mail: hugo.hammer@hioa.no.

[3] Author's status: *Associate Professor.* This author can be contacted at: Oslo and Akershus University College, Department of Computer Science, Pilestredet 35, Oslo, Norway. E-mail: anis.yazidi@hioa.no.

[4] *Chancellor's Professor* ; *Fellow: IEEE* and *Fellow: IAPR.* This author can be contacted at: School of Computer Science, Carleton University, Ottawa, Canada : K1S 5B6. This author is also an *Adjunct Professor* with the University of Agder in Grimstad, Norway. E-mail address: oommen@scs.carleton.ca.

1. Introduction

Clustering is the task of grouping data points in a way that elements that exhibit some similarity, or that inherently belong to the same class, end up in the same group. It is a fundamental task in data analysis and inference, and it is, arguably, among the most popular machine learning and data mining techniques [7] [21].

A range of different clustering methods have been proposed and each of them vary with the understanding of what a cluster, actually, is. For instance, density models, such as OPTICS [1] and DBSCAN [4], coalesce most dense regions in the space into a single cluster. As opposed to this, in hierarchical clustering [12] [15], the aim is to arrange the data points into an underlying hierarchy which then determines the various clusters. A third group of clustering algorithms constitute the so-called "centroid" methods where all the points within a computed cluster are represented by a single point, for example the cluster's centroid. The most prominent example of a scheme within this family is the acclaimed k-means clustering algorithm where a centroid is represented by the mean value of the points in the cluster. The central strategy motivating *these* clustering schemes involves classifying *unassigned* data points to the different clusters based on the distances to the means (or centroids) of the clusters.

From the above, one can informally see that any specific pattern classification algorithm can be conceptually expanded to yield a clustering scheme. Thus, if we have k previously-determined clusters, an unknown unlabelled sample can be assigned to any one of the k classes by the corresponding classification algorithm, whence the specific cluster can be grown to include this specific sample.

Almost all the well-known classifiers involved in pattern classification are based on a Bayesian principle which aims to maximize the *a posteriori* probability. Quite recently, Oommen and his co-authors proposed a completely counter-intuitive paradigm, known as CMQS, the Classification by Moments of Quantile Statistics. CMQS works with a counter-intuitive philosophy, and essentially compares the testing sample with points from each class which are distant from the mean – as opposed to the Bayesian principle which essentially compares it to the clusters' means or the centroids[5].

The question that begged investigation and that was considered open was that of invoking these "Anti–Bayesian" (AB) PR algorithms to design the corresponding clustering algorithms.

[5] A very brief overview of CMQS-based PR is presented in Section 2.

This is the avenue of research undertaken here.

The pioneering steps taken in this direction were reported in [6], where we introduced a novel alternative to the k-means clustering algorithm. The algorithm presented in [6] follows the same steps dictated by a typical k-means clustering algorithm. The main difference, however, is the manner by which it assigns the data points to the already-formed clusters. Indeed, rather than follow a Bayesian classification methodology, it traverses one of the AB-based PR CMQS-based schemes reported earlier. In fact, unlike the k-means clustering strategies that rely on centroid-based criteria, we resort to *quantiles positions distant from the cluster means* [16] [18] [14], which is a strategy just as counter-intuitive and non-obvious as the CMQS schemes themselves.

Central to the development of such CMQS clustering algorithms is the concept of a "Quantiloid"[6]. We will elaborate on the phenomenon of Quantiloids in the next section.

It is pertinent to mention that by working with Quantiloids, we will have effectively extended our previous work [6]. However, apart from doing it in the "vanilla" manner, we shall accomplish it by also invoking hierarchical clustering approaches. In fact, we shall introduce AB clustering algorithms that represent the two well-known families of hierarchical clustering methods, namely the Top-Down and Bottom-Up methods. The paper will attain its goal when we have experimentally verified that the AB principles are also valid in the case of such hierarchical clustering methods, and for d-dimensional spaces.

Before we embark on the contents of the paper, we should emphasize that we are not attempting to design an/or implement a superior clustering scheme. Rather, we are presenting a completely new paradigm for clustering. Thus, in one sense, while we are content with results comparable to those obtainable through traditional clustering mechanisms, it is more important to observe that the results we have reported are due to strategies (or one could even say, "philosophies") that were previously unknown within the field of unsupervised classification. The consequence of this assertion is that since these novel methods are orthogonal to the ones reported in the field, the possibilities to merge (fuse) these methods with the ones currently used, are huge. The research avenues that are open due to the introduction of the concept of "quantiloids" is, in our opinion, vast.

[6] To the best of our knowledge, the concept of "Quantiloids" has not been utilized in the literature except in the scientific package [11] as explained in Section 3.1. However, the use of "Quantiloids" in clustering is pioneered here.

1.1. Some Key Remarks

Since some of the initial results on “Anti–Bayesian” clustering were presented earlier [6], it is scientifically and ethically necessary for us to highlight the differences between our previous work [6] and the results presented in this current paper.

- Firstly, the prior work [6] does not differentiate between the hierarchical and flat clustering modes of operation.

- Secondly, in our earlier work [6], we had only tackled the case of assigning a new point to a cluster with two-dimensional data. This assignment was achieved by evaluating the distance between the point’s closest corners, which is a concept that is relatively easy to describe and implement. In this work:

  - We have extended our work to data that has any number of dimension. The extension from two-dimensional representations to higher-dimensional representations, is unarguably, non-trivial;
  - In the two-dimensional case, we utilized the distance between a point and the corners of a cluster as the metric for comparing the distance between the point and the cluster. Obviously, this is a computational expensive exercise, as it requires, for the given data point, the closest corner to each cluster. In this present paper, we have utilized a completely different metric, i.e., the distance between vectors of quantiloids;
  - We note that in d dimensions, there are 2^d possible corners for each cluster. Thus the simple generalization of our earlier work is prohibitively expensive as we need to the find the closest distance between a point and 2^d possible other points. By invoking the concept of quantiloids the complexity is rendered to be linear as a function of the dimensions instead of the exponential complexity.

- The third, and more important contribution of this paper is to define the distance between two clusters and not merely the distance between a point and a cluster - as done in our previous work. We are surprised that the Referee did not observe or appreciate this fundamental contribution.

- We should also mention that the latter distance is useful for hierarchical clustering while the distance between a cluster and a point is useful for what we have called flat clustering. Note that the latter distance between the quantiloids is a natural extension of the concept of the "distance between centroids".

- One should observe that even the methodology of how to compute the distance between quantiloids is non-obvious and far from trivial. This is because, given two clusters, for each dimension, we have to determine the right pairs of points to be used to compare the quantiloids. This has led us to proposing the innovative operation of “borrowing quantiles”.

1.2. Structure of the paper

In Section 2, we briefly review some related work and focus on the state-of-the-art of the AB classification framework which, indeed, forms the basis for the “Anti–Bayesian” clustering algorithms. In Section 3, we present the fundamental principles of AB clustering. Section 4 explains how AB clustering is done in one and two-dimensional spaces, and this is followed in Sections 5 and 1.1 where we demonstrate the development of AB flat clustering in d-dimensional spaces. The principles of hierarchical AB clustering are given in Section 7. In Sections 8 and 9, we report the experimental results that we have obtained which compare our AB flat and hierarchical clustering schemes to their Bayesian counterparts on both synthetic and real-life data sets. Section 10 concludes the paper.

2. Related Work on “Anti–Bayesian” Pattern Recognition

In this section, we very briefly review the related work on AB classification. Initially, in [16], the authors worked with the quantiles for the data distributions, and showed how we could achieve near-optimal classification for various uni-dimensional distributions. For uni-dimensional quantile-based PR, their methodology is based on comparing the testing sample with the $\frac{n-k+1}{n+1}^{th}$ percentile of the first distribution and the $\frac{k}{n+1}^{th}$ percentile of the second distribution. These results were shown to be applicable for the distributions that are members of the symmetric and asymmetric exponential family. By considering the entire spectrum of the possible values of k, the results in [16] [18] [14], showed that the specific value of k is usually not so crucial, and that the same results were also true for multi-dimensional features.

In [17], the authors further proposed a new border identification algorithm, namely the AB Border Identification scheme. For each class, this method selects, as the corresponding border points, a small number of data points that lie close to the discriminant function’s boundary, but where these points are not within the central part of the class conditional distributions.

The results of [16] [18] and [14] were used to design numerous Prototype Reduction Schemes in [19], and an AB text classification scheme in [13].

3. The "Anti–Bayesian" Clustering Solution

3.1. Quantiloids

As alluded to earlier, the solution we propose is based on the concept of "Quantiloids". What then is a Quantiloid? The quantiloid associated with the real number, θ, is, quite simply, for a uni-dimensional distribution, the unique point where the Cumulative Distribution Function (CDF) has the value θ. This is the unique point where the probability mass (i.e., the integral of the Probability Density Function (PDF)) attains the value of θ.

While this is an elementary concept for uni-dimensional variables, the concept can be extended for multi-dimensional vectors to be the hyper-surface under which the CDF has the value θ. The goal of this paper is to develop quantiloid-based clustering algorithms that work in an AB paradigm just as the centroid-based clustering algorithms worked within the "Bayesian" paradigm. Indeed, rather than characterizing a cluster by its centroid, we shall attempt to characterize it by its quantiloids, which will then lead to the various AB clustering algorithms.

It should be mentioned that in the package named *Weighted Correlation Network Analysis* (WGCNA) the authors programmed modules in 'R' to perform Weighted Correlation Network Analysis [11]. In that package, the authors employed the word "Quantiloids" to describe the concept of using quantiles as follows: "If samples within each class are heterogeneous, a single centroid may not represent each class well. This function can deal with within-class heterogeneity by clustering samples (separately in each class), then using a one representative (mean, eigensample) or quantile for each cluster in each class to assign test samples" [11]. We applaud the authors of the package for suggesting the use of quantiles as a tool for classification. However, we could not find any documentation that explains their methodology.

Although the concept of quantiloids is valid for multi-dimensional vectors, the question of *how* they can be computed and represented is still open. We shall thus restrict ourselves to uni-dimensional quantiloids by processing the multi-dimensional distribution in terms of its uni-dimensional marginals.

3.2. "Anti–Bayesian" Classification Rules

We first summarize the AB classification rules designed and proven in [16], [18] and [14] for uni-dimensional features.

To do this, we use the notation that for the j^{th} dimension of the feature vector of class ω_i, $q_p^{i,j}$ is the quantiloid for the value p, i.e., the position where the feature's CDF has a value of p. In the case when both the classes are characterized by only a *single* feature X, q_p^i is ω_i's quantiloid for the value p, i.e. more formally $q_p^i = Pr(X < p|X \in \omega_i)$. Observe that we encounter the cases when the quantiloids overlap (i.e., $q_{1-p}^1 < q_p^2$) or when they do not overlap (i.e., $q_{1-p}^1 > q_p^2$). Using this notation, the uni-dimensional AB classification rules for the testing sample x^* are:

Case 1: When the quantiloids are non-overlapping (see Figure 1 on the left):

$$\begin{array}{lll} \text{If } x^* < q_{1-p}^1 & & \Rightarrow x^* \in \omega_1; \\ \text{If } x^* > q_p^2 & & \Rightarrow x^* \in \omega_2; \\ \text{If } (q_{1-p}^1 < x^* < q_p^2) & \wedge\ (\|x^* - q_{1-p}^1\| < \|x^* - q_p^2\|) & \Rightarrow x^* \in \omega_1; \\ \text{If } (q_{1-p}^1 < x^* < q_p^2) & \wedge\ (\|x^* - q_{1-p}^1\| > \|x^* - q_p^2\|) & \Rightarrow x^* \in \omega_2. \end{array} \tag{1}$$

The reader will observe that the cases are mutually exclusive and that the classification border is:$\frac{q_{1-p}^1+q_p^2}{2}$.

Case 2: When the quantiloids are overlapping (see Figure 1 on the right):

$$\begin{array}{lll} \text{If } x^* < q_p^2 & & \Rightarrow x^* \in \omega_1; \\ \text{If } x^* > q_{1-p}^1 & & \Rightarrow x^* \in \omega_2; \\ \text{If } (q_p^2 < x^* < q_{1-p}^1) & \wedge\ (\|x^* - q_p^1\| < \|x^* - q_{1-p}^2\|) & \Rightarrow x^* \in \omega_1; \\ \text{If } (q_p^2 < x^* < q_{1-p}^1) & \wedge\ (\|x^* - q_p^1\| > \|x^* - q_{1-p}^2\|) & \Rightarrow x^* \in \omega_2. \end{array} \tag{2}$$

In this case, the comparison is based on the distant quantiloids and so the classification border is:$\frac{q_p^1+q_{1-p}^2}{2}$.

The reader will observe that the latter case (Case 2) is the one that uses the so-called "Dual" scenario (please see [16], [18] and [14]), and where the extreme quantiloids are used for the classification as opposed to the quantiloids that are close to the discriminant. In the symmetric cases analyzed in [16], [18] and [14], it is easy to see that the assignments in the so-called "Dual" scenario reduce to those involving comparisons to the quantiloids that are *close to the discriminant*, but where the assignment is to the class *that is the more distant one*. The decision rule for this is given below.

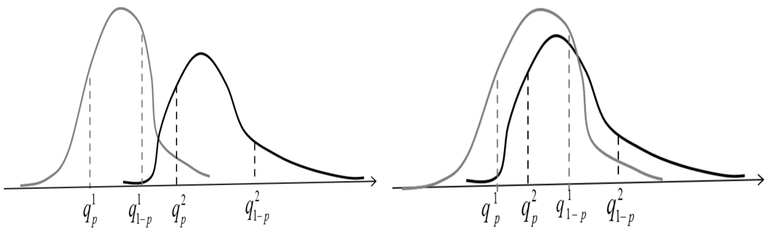

Figure 1: The AB scheme: (a) When the quantiloids are non-overlapping on the left, and (b) When the quantiloids are overlapping on the right.

Case 2 (Revised): When the quantiloids are overlapping (again see Figure 1 on the right):

$$\begin{array}{lll} \text{If } x^* < q_p^2 & & \Rightarrow x^* \in \omega_1; \\ \text{If } x^* > q_{1-p}^1 & & \Rightarrow x^* \in \omega_2; \\ \text{If } (q_p^2 < x^* < q_{1-p}^1) & \wedge\ (\|x^* - q_p^2\| < \|x^* - q_{1-p}^1\|) & \Rightarrow x^* \in \omega_2; \\ \text{If } (q_p^2 < x^* < q_{1-p}^1) & \wedge\ (\|x^* - q_p^2\| > \|x^* - q_{1-p}^1\|) & \Rightarrow x^* \in \omega_1. \end{array} \tag{3}$$

The difference between the two versions of Case 2 (Eq. (2) and (3)) lies in the assignments made in the last two statements, where they, however, are *done to the non-adjacent classes.* In this case, the comparison is based on the closer quantiloids and so the classification border is:$\frac{q_p^2+q_{1-p}^1}{2}$. To distinguish between these two scenarios, we shall refer to this version of the "Dual" scenario as the "Swapped Border" scenario.

The cases when the second distribution (for ω_2) is to the left of the first (for ω_1), is shown in Figure 2. Observe that this is identical to the case of the figure on the left of Figure 1, except that the identities of the classes is interchanged.

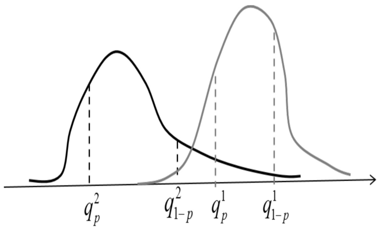

Figure 2: This figure depicts the case of when the quantiloids do not overlap but when second distribution (for ω_2) is to the left of the first (for ω_1).

There is one additional scenario, and that occurs when there is a huge overlap between the distributions (See Figure 3). The classification decision rule to be used is not that obvious because the classes are highly overlapping. Apart from this, the classification of an unknown sample itself is not just non-obvious, it is actually "meaningless". This case never occurred in our experiments.

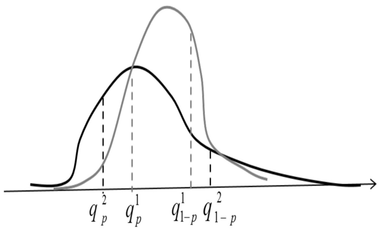

Figure 3: This figure depicts the scenario when there is a huge overlap between the distributions.

4. The AB One and Two-dimensional Clustering

In the interest of completeness, before we proceed, we briefly present the principles of the "Anti–Bayesian" flat[7] clustering scheme presented in [6], based on the one and two dimensional PR cases explained in Section 3.2. Then, in Section 5, we extend the principles of the flat clustering to the multidimensional case where the dimension of the data can exceed two. In Section 7, we present our solution to the hierarchical clustering solutions.

The algorithm presented in [6] is based on the above-described AB classification framework [16] [18] [14], to cluster unclassified points into k clusters. The algorithm follows the same steps as a k-means clustering algorithm. The main difference relies on two things: How to characterize the clusters in terms of multiple quantiles and not the centroids, and how to the compute the distance between a point to its nearest neighbor where the distance is computed between point-based quantiles.

We explain below the AB clustering strategy, proposed in [6], for one and two-dimensional data.

"Anti–Bayesian" Uni-dimensional Clustering: Consider the case when we are dealing with uni-dimensional data. Let $\{x_1^1, x_2^1, \ldots, x_{n_1}^1\}$ and $\{x_1^2, x_2^2, \ldots, x_{n_2}^2\}$ be $n_1 + n_2$ random samples from two unknown probability distributions $f_1(x)$ and $f_2(x)$ characterizing the classes ω_1 and ω_2 respectively. Our task is to classify a new point z to ω_1 or ω_2, i.e., to see whether it is a sample that comes from $f_1(x)$ or $f_2(x)$. If we assume that the variances of $f_1(x)$ and $f_2(x)$ are equal, the optimal classification strategy is to assign z to ω_1 or ω_2 if z is closer to the means of $f_1(x)$ or $f_2(x)$ respectively. This, in turn, assigns z to ω_1 if the average of the

[7] The previous methods are referred to as being "flat" because they do not invoke any hierarchical paradigm. This paper first shows how the "flat" method can be used for multi-dimensional clustering and then proceeds to consider hierarchical methods.

samples $\{x_1^1, x_2^1, \ldots, x_{n_1}^1\}$ is closer to z than the average of $\{x_1^2, x_2^2, \ldots, x_{n_2}^2\}$. It is otherwise assigned to ω_2. The reader should observe that this is precisely how points are assigned to clusters in the k-means paradigm.

In the AB classification approach, classification is achieved based on quantile-based comparisons rather than comparisons with regard to the mean. To render this formal, we denote the quantiles as follows: $q_p^1 = Pr(X < p|X \in \omega_1)$, and $q_p^2 = Pr(X < p|X \in \omega_2)$. Although, in practice, the quantiles have to be, estimated (or learned), for ease of clarification, in the descriptions below, we assume that the quantiles are known. We also know that for $p < 0.5$, $q_p^1 < \text{Median}(X)$ so that for $p < 0.5$, q_{1-p}^1 is always greater than q_p^1. The analogous conditions are satisfied by q_p^2 and q_{1-p}^2. With this in place, the AB classification method operates as follows:

1. Determine which of the distributions $f_1(x)$ or $f_2(x)$ is to the left by using the quantiles of the distributions. We have three possible cases:

   Case 1: If $q_p^1 < q_p^2$ and $q_{1-p}^1 < q_{1-p}^2 \implies f_1(x)$ is to the left of $f_2(x)$.

   Case 2: If $q_p^1 > q_p^2$ and $q_{1-p}^1 > q_{1-p}^2 \implies f_2(x)$ is to the left of $f_1(x)$.

   Case 3: Else, we determine their relative positions by comparing the averages of the quantiles as follows:

   If $\frac{q_p^1 + q_{1-p}^1}{2} < \frac{q_p^2 + q_{1-p}^2}{2} \implies f_1(x)$ is to the left of $f_2(x)$.

   Else[8] $f_2(x)$ is to the left of $f_1(x)$.

2. Once the relative positions of the distributions are determined, the classification rule must now be specified. For simplicity, we merely describe this for Case 1 since the rules for the "mirrored" cases are analogous. The Anti-Bayesian rule classifies using the *right* quantile of the left distribution and the *left* quantile of the right distribution. If $B = \frac{q_{1-p}^1 + q_p^2}{2}$, we classify as follows:

   If $z < B$, classify z to ω_1.

   Else, classify z to ω_2.

   This approach works even when the distributions overlap such that q_{1-p}^2 is to the left of q_p^1 as shown by the figure on the right of Figure 1.

[8] This case occurs rarely in practice except when the classes are highly overlapping, in which case the PR problem is often meaningless.

As mentioned earlier, Figure 1 depicts the above two cases. We see that for Cases 1 and 2, $f_1(x)$ and $f_2(x)$ are the distributions to the left and right, respectively.

"Anti–Bayesian" Two-dimensional Clustering: The AB clustering for two-dimensional data was achieved in [6] as follows. We assume that $\{x^1_{11}, \ldots, x^1_{n_1 1}\}$ and $\{x^1_{12}, \ldots, x^1_{n_1 2}\}$ are $2n_1$ independent samples from $f_1(x)$. The two-dimensional vector points for ω_1 are obtained as the pairs: $\{X^1_i = (x^1_{i1}, x^1_{i2}),\ i = 1, 2, \ldots n_1\}$. Similarly, we assume that $\{x^2_{11}, \ldots, x^2_{n_2 1}\}$ and $\{x^2_{12}, \ldots, x^2_{n_2 2}\}$ are $2n_2$ independent samples from $f_2(x)$. The two-dimensional vector points for ω_2 are obtained as the pairs: $\{X^2_i = (x^2_{i1}, x^2_{i2}),\ i = 1, 2, \ldots n_2\}$. Again our task is to classify a *vector* point Z to ω_1 or ω_2 as per $f_1(X)$ or $f_2(X)$. The classification is done as per the ideas in [16] [18] and [14]. It is a natural generalization of the uni-dimensional case above and follows two steps:

1. Define the rectangle with corners (q^1_{1-p}, q^1_{1-p}), (q^1_{1-p}, q^1_p), (q^1_p, q^1_{1-p}) and (q^1_p, q^1_p) for $f_1(X)$, and the analogous rectangle corners for $f_2(X)$. Locate the corners in the two rectangles that are closest to each other.

2. If Z is closer to the corner of the quantile rectangle of $f_1(X)$, classify Z to ω_1. Else classify Z to ω_2.

Figure 4 shows the classification procedure for the two typical cases.

We are now set to explain how AB Multi-dimensional Clustering can be achieved when the number of dimensions is greater than two.

5. The AB Multi-dimensional Clustering

We now consider the extensions of the results in Section 4 to the multi-dimensional scenario. To explain this, we state that in [6], as explained above, we used the concept of the closest quantile corners in two dimensions. For the multi-dimensional scenario, instead of measuring the distances between the centroids as as done in the Bayesian paradigm, we measure the distances between the quantiloids[9].

5.1. The Quantiloids Used

In the d-dimensional feature space, let $Q^1 = [Q^1_1, Q^1_2, ..., Q^1_d]$ and $Q^2 = [Q^2_1, Q^2_2, ..., Q^2_d]$ denote the quantiloids of the distributions (clusters) of $f_1(X)$ and $f_2(X)$ respectively. Q^1

[9] This generalizes the concept used in our paper [6] (explained above), where the corners of the rectangles encountered in two dimensions are, in one sense, the quantiloids as explained in Section 3.1.

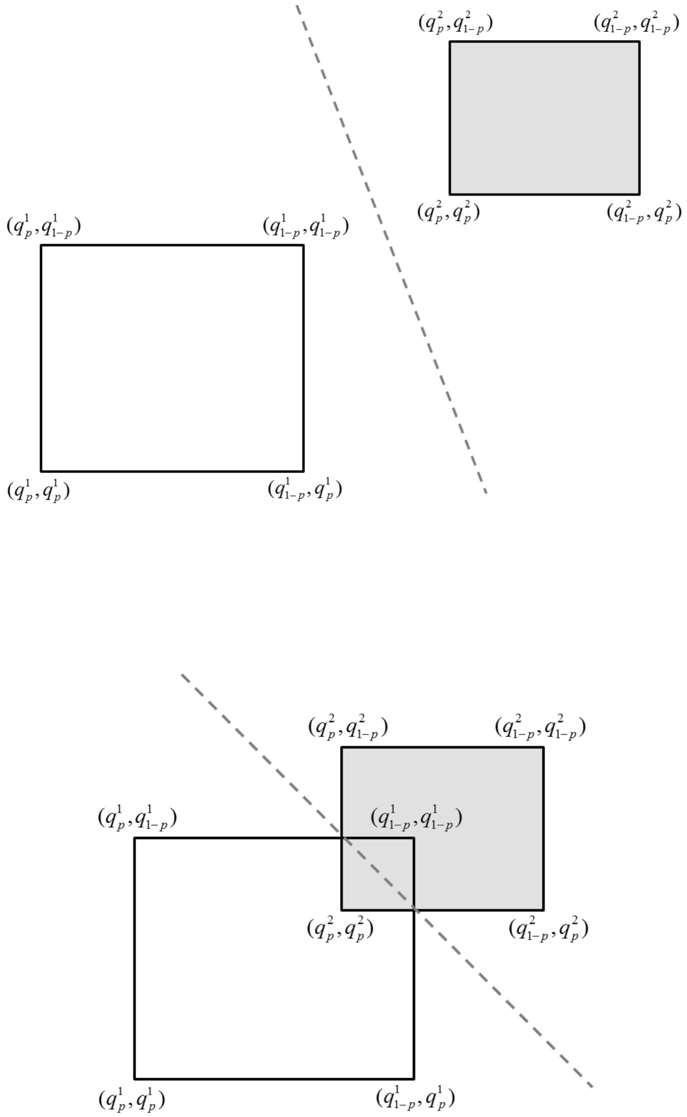

Figure 4: Classification in the two dimensional scenario for two typical cases. In each case, the gray dashed line shows the border of the discriminant regions when Z is classified to ω_1 or ω_2 as per $f_1(X)$ or $f_2(X)$.

and Q^2 are computed as follows:

- In each dimension, we decide which distribution (cluster) is to the left and which is to the right. To decide this, we exactly follow the principles explained in Section 4 above.
- For each of the three cases defined in Section 4 the elements in the i^{th} dimension of the quantiloid vectors is computed as follows:
  - Case 1: Here $f_1()$ is to the left of $f_2()$. Here we set $Q^1_i = q^1_{i,p}$ and $Q^2_i = q^2_{i,1-p}$. In this case, we also have to consider the case when an exception occurs, i.e., when there is a degree of overlap between them. Indeed, if the $f_1()$ and $f_2()$ are close in the i^{th} dimension such that the quantiles overlap, i.e. that $q^1_{i,p}$ is to the right of

$q^2_{i,1-p}$, then the point should be classified to ω_1 if it is closer to $q^2_{i,1-p}$ and to ω_2 if it is closer to $q^1_{i,p}$. This corresponds to "Swapped Border" scenario (Case 2 (Revised)) in Section 3.2. With such overlapping quantiles we therefore set $Q^1_i = q^2_{i,1-p}$ and $Q^2_i = q^1_{i,p}$.

- Case 2: Here $f_1()$ is to the right of $f_2()$ and, if the quantiles do not overlap, we set $Q^1_i = q^1_{i,1-p}$ and $Q^2_i = q^2_{i,p}$. If the quantiles overlap, we switch the quantiles, as described above, to account for the "Swapped Border" scenario as in Case 1.
- Case 3: Here we set $Q^1_i = \frac{q^1_{i,p}+q^1_{i,1-p}}{2}$ and $Q^2_i = \frac{q^2_{i,p}+q^2_{i,1-p}}{2}$. This is the case when the overlap is significant and the classification can be considered to be "meaningless". As mentioned earlier, this case occurs very rarely in the domain of clustering.

The first two scenarios encountered above can be explained in the following figures drawn in two dimensions. In each case, we have plotted the hyper-ellipsoids characterizing the hyper-plane of a specified height. These hyper-ellipsoids characterize the corresponding hyper-rectangles. If the overlap is small, the distances are measured from the nearest quantiloids, as seen in Figure 5.

If the overlap is significant, the distances are measured from the fartherest quantiloids (Case 2 of Section 3.2), or equivalently from the "Swapped Border" (Case 2 (Revised)) quantiloids explained in Section 3.2 and shown in Figure 6.

5.2. The Distance Measures Used

Based on the definition of the quantiloids, we are now ready to define two types of distances used in the framework of our AB clustering paradigm. The two types of distance metrics we use are listed below:

- *Data Point to Cluster (DPC) Distance:* Once the quantiloids have been computed following the procedure above, the points Z is classified to ω_1 if Z is closer, in terms of its Euclidean distance to Q^1 than to Q^2. Otherwise, Z is classified to ω_2. The DPC Distance has been used for flat clustering as well as for Top-Down clustering.

- *Cluster to Cluster (CC) Distance* The same notion can be used to characterize the distance between two clusters. The CC distance between two clusters C_1 and C_2 is the Euclidean distance between their corresponding quantiloids Q_1 and Q_2. The notion of the CC Distance is usually used for Bottom-Up clustering techniques.

Both of these will be clarified later.

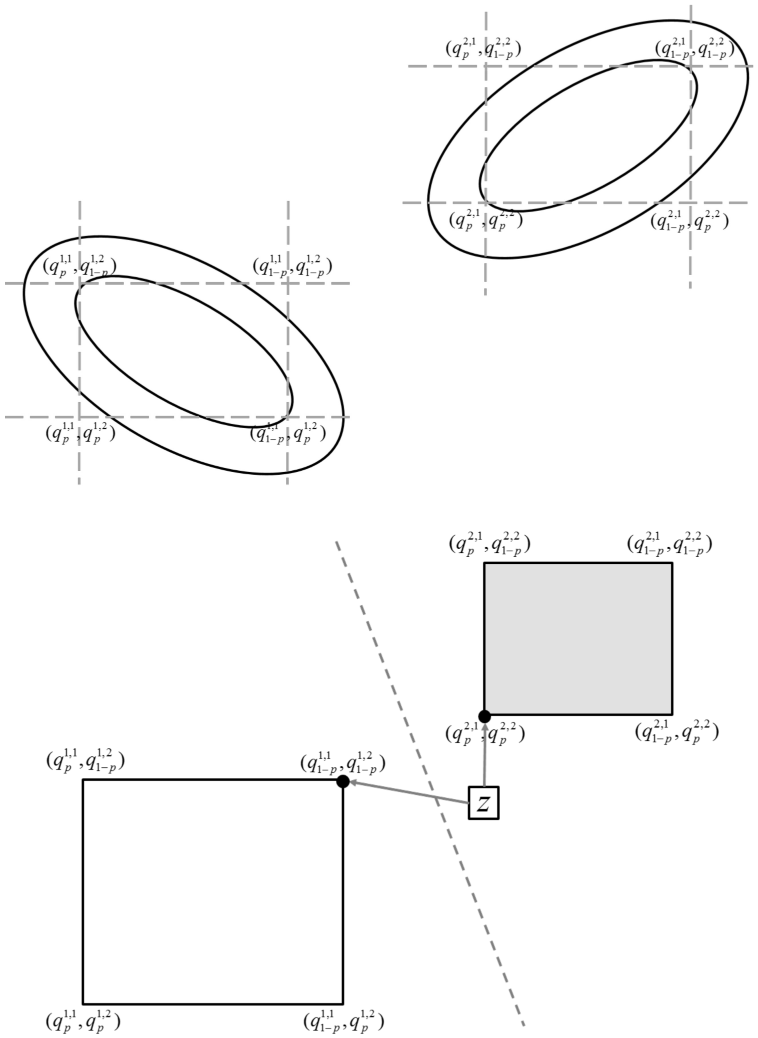
$(q_p^{2,1}, q_{1-p}^{2,2})$
$(q_{1-p}^{2,1}, q_{1-p}^{2,2})$
$(q_p^{2,1}, q_p^{2,2})$
$(q_{1-p}^{2,1}, q_p^{2,2})$
$(q_p^{1,1}, q_{1-p}^{1,2})$
$(q_{1-p}^{1,1}, q_{1-p}^{1,2})$
$(q_p^{1,1}, q_p^{1,2})$
$(q_{1-p}^{1,1}, q_p^{1,2})$
$(q_p^{2,1}, q_{1-p}^{2,2})$
$(q_{1-p}^{2,1}, q_{1-p}^{2,2})$
$(q_p^{2,1}, q_p^{2,2})$
$(q_{1-p}^{2,1}, q_p^{2,2})$
$(q_p^{1,1}, q_{1-p}^{1,2})$
$(q_{1-p}^{1,1}, q_{1-p}^{1,2})$
Z
$(q_p^{1,1}, q_p^{1,2})$
$(q_{1-p}^{1,1}, q_p^{1,2})$

Figure 5: The case when the multi-dimensional distributions have little overlap: (a) The two sets of ellipsoids representing the Gaussian distributions, and (b) The corresponding rectangles representing the quantiloids.

6. Remarks regarding Multi-Dimensional Clustering

In the light of what we have described above, it is appropriate to submit a few comparative remarks.

6.1. Comparison to Border Identification

The phenomenon of qunatiloids bears a marked similarity to the concepts used in Border Identification methods. In fact, the quantiloids of a cluster are the points in the d-dimensional space located at its non-central "border" as opposed to its centroid that is located at or near the cluster's center. It is interesting to note that these points can be used to adequately characterize the cluster – just as the centroid does.

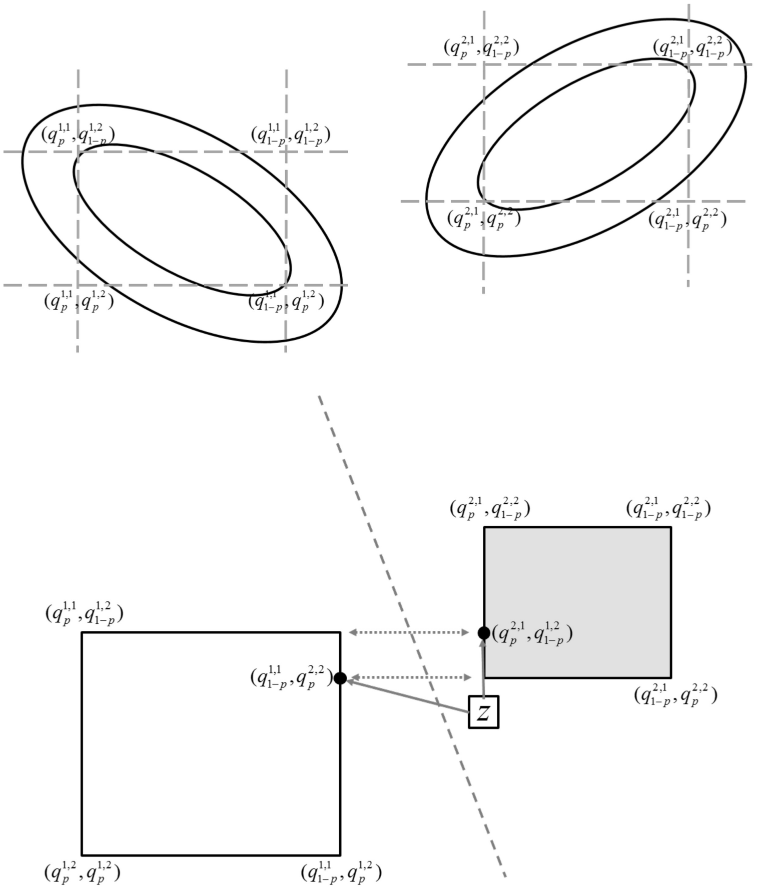
$(q_p^{2,1}, q_{1-p}^{2,2})$
$(q_{1-p}^{2,1}, q_{1-p}^{2,2})$
$(q_p^{1,1}, q_{1-p}^{1,2})$
$(q_{1-p}^{1,1}, q_{1-p}^{1,2})$
$(q_p^{2,1}, q_p^{2,2})$
$(q_{1-p}^{2,1}, q_p^{2,2})$
$(q_p^{1,1}, q_p^{1,2})$
$(q_{1-p}^{1,1}, q_p^{1,2})$
$(q_p^{2,1}, q_{1-p}^{2,2})$
$(q_{1-p}^{2,1}, q_{1-p}^{2,2})$
$(q_p^{1,1}, q_{1-p}^{1,2})$
$(q_p^{2,1}, q_{1-p}^{1,2})$
$(q_{1-p}^{1,1}, q_p^{2,2})$
$(q_{1-p}^{2,1}, q_p^{2,2})$
Z
$(q_p^{1,2}, q_p^{1,2})$
$(q_{1-p}^{1,1}, q_p^{1,2})$

Figure 6: The case when the multi-dimensional distributions have a large overlap: (a) The two sets of ellipsoids representing the Gaussian distributions, and (b) The corresponding rectangles representing the quantiloids. Observe that in this case, we have utilized the "Swapped Border" scenario to compute the quantiloids.

6.2. Majority Voting in the d dimensions

Instead of using the Euclidean distance to compare the testing sample to the quantiloids, we could just as well have used the alternative concept discussed in [19] based on invoking majority voting. The idea motivating this is simple: In order to assign an d dimensional data point to one of two clusters C_1 or C_2, we could have applied our proposed uni-dimensional classification strategy *separately*, in each dimension. As a result of this, we will obtain d decisions after which we can invoke a majority voting to decide on the cluster that the unassigned sample Z should fall into. Of course, whenever d is even number, one should apply a random tie-breaking mechanism if the number of votes for C_1 is equal to those for C_2.

The reader should also observe that by invoking such a majority voting mechanism, we are ignoring the dependence between the dimensions as in the case of a *Naïve*-Bayesian process. In this context, we mention that the accuracy of such a majority voting phase improves as the number of dimensions increases.

7. Principles of our Hierarchical Clustering

It is well-known that clustering can also be achieved hierarchically, where the scheme is either of a Bottom-Up paradigm or of a Top-Down paradigm. These traditional paradigms can be extended to our AB paradigm by *merely modifying the concept of the distances invoked*, where in the AB scheme, the distance is based on the concept of quantiloids. Thus, in essence, our algorithms follow the classical hierarchical clustering philosophy [8] in all the relevant steps, except that we consider the distances to the qunatiloids rather than the distances to the centroids of the clusters.

To explain these, we present the hierarchical AB clustering methods. These are, precisely, the counter-parts of the classical hierarchical clustering methods [8]. The only difference is the way by which we specify the distances, i.e., whether we invoke the DPC or CC distance measures based on the principles of the quantiloids rather than centroids.

7.1. Bottom-Up AB Clustering

A Bottom-Up clustering works with the principle that all the points are individually specified in the d-dimensional space. The points and then gathered together to form clusters, to which the unclassified points are then subsequently added. Thus, the steps of a Bottom-Up AB clustering are described below:

- Compute all pair-wise similarity distances between the different clusters and populate the proximity matrix. The distance between the clusters is merely the Euclidean distance between their corresponding quantiloids.
- Identify the closest clusters in terms of their similarity and merge them into a single cluster. This results in updating the proximity matrix and decreasing its order by unity.
- Repeat the above steps until we obtain the desired (pre-specified) number of clusters.

7.2. Top-Down AB Clustering

A Top-Down clustering works with the principle that all the points are collectively grouped into a single cluster in the d-dimensional space. The most distant points are then separated to be the nuclei of two distinct clusters, and the points closest to these are then included into their respective clusters. Again, in an AB paradigm, the distances are measured in terms of the quantiloids rather than the centroids. Thus, the steps involved in Top-Down AB clustering are described below:

- Start at the top level with all the data points coalesced in a single cluster.
- Use a flat clustering scheme in order to split the cluster.
- Apply the procedure recursively until a termination condition on the depth of the tree is reached or until each data point (singleton) ends up as its own cluster (maximum depth). Usually, one invokes a termination condition which involves the desired (pre-specified) number of clusters.

8. Experimental Results: Synthetic data example

In order to test the validity of the concepts proposed in this paper, we conducted numerous experiments on synthetic data. In the interest of space and brevity, we report the salient ones here.

In all our experiments, we used $K = 3$ clusters. All the synthetic data were from multivariate Normal distributions, where we fixed d, the dimension of the space, to be 4.

8.1. Data Generation

We shall first explain how the data points were generated for Normally-distributed distributions. Let $N(\mu_k, \Sigma_k)$ denote a multivariate Normal distribution with an expectation vector μ_k and a covariance matrix Σ_k, where $k = 1, \ldots, K$ (where we are dealing with K clusters). To generate the K distributions, it is crucial that we determine how μ_k and Σ_k $(k = 1, \ldots, K)$ are set.

In our experiments, the expectations, $\{\mu_k\}$ were generated in two ways, each of which led to different sets of experiments:

1.1. The expectations of the classes were spread along a line in $\mathbb{R}^d$ as per:

$$\mu_k = t_k \cdot [1, 1, \ldots, 1], k = 1, 2, \ldots, K,$$

where $\{t_k\}$ were chosen such that the clusters were reasonably spread along the line, rendering their inter-class overlaps to be minimal. In the experiments we used $t_k = 1.8k, k = 1, \ldots, K$;

1.2. The expectations were uniformly spread on the $d-$dimensional cube $[0, D]^d$ where, again, D was chosen such that the clusters were reasonably spread in space. This too made their inter-class overlaps to be minimal. In the experiments for which we report the results, we used $D = 6$.

The covariance matrices Σ_k for each cluster was generated by the following procedure:

2.1. Set the diagonal element to be equal to 1, i.e. the marginal variance was equal to unity in all the dimensions;

2.2. The correlation between each of the variables in the i^{th} and j^{th} dimensions for $i = 1, \ldots, j < i$ was drawn uniformly from the interval $[-\rho_{\max}, \rho_{\max}]$, where $\rho_{\max} < 1$. In the experiments we used two different values of $\rho_{\max}$. Either $\rho_{\max} = 0$ (independence between feature dimensions) and $\rho_{\max} = 0.8$ (strong correlation);

2.3. We checked if the generated covariance matrix was positive definite. If it was not, we repeated Step 2, above[10].

In what follows, we let n denote the number of samples generated from each underlying cluster in the synthetic data. Thus, the total number of data points generated were $n \cdot K$. In the experiments that we report, we used two values of n, i.e., $n = 20$ and $n = 100$.

8.2. Quantile and "Distance" Estimations

Since we constantly need to estimate the quantiloids, we opted to achieve this using the corresponding quantiles in each of the projected dimensions. This was done non-parametrically and parametrically as below:

- Non-parametrically (referred to in the columns titled "AB Non-parametric" in the tables below): This was achieved in a manner similar to the work presented in [6].

- Parametrically: This was achieved by assuming normality Here we estimated the corresponding μ and σ and computed the respective quantiles from the Normal distributions (referred to in the columns titled "AB Parametric" in the tables).

The corresponding "distance" estimations for the experiments done were achieved as below:

[10] With $\rho_{\max} = 0.8$ and for $d = 3$ the matrix were almost always positive definite on the very first attempt. For $d = 5$, on the average, about every third matrix that was generated was positive definite.

- Row titled “Bottom up”: These represent the classical version where all the inter-point distances are computed. The distances were computed between the centroids in case of the Bayesian clustering, and between the corresponding quantiloids in the case of the AB clustering.

- Row titled “Bottom-Up Distance UD (uni-dimensional)”: In this case, we sorted the data by the first dimension and repeatedly merged the closest points in this dimension. This approach was a simplification of the general approach where we should have considered all the dimensions. The present approach required less computations. We expected that such a simplification would result in a reduced accuracy as we only relied on the first dimension of the data for executing the clustering, and this was, indeed, our experience.

- Row titled “Top-Down”: In this case, the points were repeatedly split in two using k-means and the AB analog ([6]).

- Row titled “Top-Down Distance UD (uni-dimensional)”: In this case, the data was sorted by the first dimension and repeatedly split in such a way that the distance (Bayes or AB) between the clusters was as large as possible. Again, this approach required less computations than the “Top-Down” one. As before, it is reasonable to expect such a simplification to result in a reduced accuracy. This was, indeed, the case.

8.3. Evaluation of Clustering Performance

Before we consider the performance of the various clustering strategies, it is prudent for us to understand how the performance of a clustering algorithm can be quantified. Indeed, the question of how is to be measured, is far from being obvious or trivial. Consider the following simple example. Suppose that six points $\{A_1, A_2, A_3, B_1, B_2, B_3\}$ are to be clustered into two clusters, with the goal that the elements $\{A_i\}$ and $\{B_i\}$ are located in the same respective clusters. Consider now the case when the results of a specific clustering algorithm are:

Cluster C_1: $\{A_1, B_2, B_3\}$,

Cluster C_2: $\{A_2, A_3, B_1\}$.

If the requirements of the clustering problem required that all the elements $\{A_i\}$ are to be in cluster C_1, and that all the elements $\{B_i\}$ are to be in cluster C_2, we could immediately see that the number of errors incurred by the above clustering is 4. On the other hand, if the clustering problem merely stipulated that the $\{A_i\}$'s were to be in one cluster, and that the $\{B_i\}$'s in another (irrespective of whether it is C_1 or C_2), the number of errors is 2. In our experiments, we used a simple paired comparison approach to get rid of the labeling issue described above. If two points were in the same clusters for the true clusters described by the "state of nature", they should also have ideally been in the same cluster in the results obtained from the clustering algorithm. Conversely, if two points were *not* in the same clusters for the true underlying clustering, they should not have been in the same cluster in the results of the clustering algorithm either. More formally, suppose that we are given the data points $x_1, x_2, \ldots, x_{nK}$ that we intend to cluster into K clusters. Let $C_{\text{truth}}(x)$ and $C_{\text{alg}}(x)$ denote the cluster of data point x in the true clusters described by the "state of nature" and as a result of using the clustering algorithm, respectively. Our measure of the performance of the clustering algorithm will be the portion of paired comparison agreements and can be formalized as

$$\frac{1}{nK(nK-1)}\sum_{i=1}^{nK}\sum_{j<i}[\mathbb{1}(C_{\text{truth}}(x_i) = C_{\text{truth}}(x_j))\mathbb{1}(C_{\text{alg}}(x_i) = C_{\text{alg}}(x_j))+ \\ +\mathbb{1}(C_{\text{truth}}(x_i) \neq C_{\text{truth}}(x_j))\mathbb{1}(C_{\text{alg}}(x_i) \neq C_{\text{alg}}(x_j))]$$

where $nK(nK-1)$ refers to the number of paired comparisons, and $\mathbb{1}(\cdot)$ is the indicator function.

8.4. Results: $n = 20$ Data Points

In this section, we report the comparative results obtained for four experimental settings for the case when the date involved $n = 20$ data points per cluster in the ground truth. The corresponding results for the case when $n = 20$ is given in 8.5. The analyses of the results for the cases when $n = 20$ and $n = 100$ are given in Section 8.6.

In each case, we report the error rates (recorded as per Section 8.3, where the errors in each cluster were the number of the misclassified points inferrred from the ground truth made available from the data generation process) with their corresponding 95% confidence intervals. We summarize the results tabulated in the following four tables as follows:

- Table 1 illustrates the case when the features were independent (the values between the dimensions), and the expectations were along the line. This is the scenario given in Item 1.1 in Section 8.1.

- Table 2 illustrates the case when the values between the dimensions were *dependent*, and the expectations were along the line as given in Item 1.1 in Section 8.1.

- Table 3 illustrates the case when the features were independent, and the expectations were along the cube. This is the scenario given in Item 1.2 in Section 8.1.
- Finally, Table 4 illustrates the case when the values between the dimensions were *dependent* and the expectations were along the cube as given in Item 1.2 in Section 8.1.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.073 (0.071, 0.075)	0.099 (0.096, 0.102)	0.1 (0.097, 0.103)
Bottom-Up	0.188 (0.183, 0.193)	0.283 (0.278, 0.289)	0.276 (0.272, 0.281)
Bottom-up Distance UD	0.328 (0.325, 0.332)	0.385 (0.38, 0.391)	0.385 (0.379, 0.39)
Top-Down	0.165 (0.162, 0.168)	0.18 (0.176, 0.184)	0.191 (0.188, 0.194)
Top-Down Distance UD	0.278 (0.274, 0.281)	0.336 (0.331, 0.341)	0.328 (0.323, 0.333)

Table 1: The clustering errors of the various methods for the case when the features were independent, and the expectations were along the line as specified in Item 1.1 in Section 8.1. Here $n = 20$, and the dimensionality $d = 4$.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.082 (0.079, 0.085)	0.113 (0.11, 0.117)	0.119 (0.115, 0.123)
Bottom-Up	0.202 (0.197, 0.206)	0.294 (0.288, 0.301)	0.292 (0.286, 0.299)
Bottom-up Distance UD	0.341 (0.337, 0.346)	0.391 (0.385, 0.397)	0.394 (0.388, 0.4)
Top-Down	0.169 (0.165, 0.172)	0.179 (0.175, 0.183)	0.195 (0.191, 0.199)
Top-Down Distance UD	0.280 (0.276, 0.284)	0.328 (0.324, 0.333)	0.322 (0.318, 0.327)

Table 2: The clustering errors of the various methods for the case when the features were dependent with $\rho_{\max} = 0.8$, and the expectations were along the line as specified in Item 1.1 in Section 8.1. Here $n = 20$, and the dimensionality was $d = 4$.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.084 (0.08, 0.088)	0.096 (0.091, 0.1)	0.094 (0.09, 0.098)
Bottom-Up	0.185 (0.177, 0.193)	0.263 (0.252, 0.274)	0.265 (0.254, 0.276)
Bottom-up Distance UD	0.415 (0.408, 0.423)	0.44 (0.432, 0.447)	0.437 (0.429, 0.444)
Top-Down	0.113 (0.108, 0.118)	0.128 (0.122, 0.133)	0.128 (0.123, 0.133)
Top-Down Distance UD	0.337 (0.331, 0.343)	0.362 (0.356, 0.367)	0.352 (0.346, 0.359)

Table 3: The clustering errors of the various methods for the case when the features were independent, and the expectations were along the cube as specified in Item 1.2 in Section 8.1. Here $n = 20$, and the dimensionality $d = 4$.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.098 (0.094, 0.103)	0.098 (0.094, 0.102)	0.099 (0.094, 0.103)
Bottom-Up	0.21 (0.201, 0.219)	0.277 (0.266, 0.288)	0.275 (0.265, 0.286)
Bottom-up Distance UD	0.423 (0.415, 0.43)	0.434 (0.427, 0.442)	0.443 (0.435, 0.45)
Top down	0.13 (0.125, 0.135)	0.132 (0.127, 0.137)	0.13 (0.125, 0.135)
Top-Down Distance UD	0.335 (0.329, 0.341)	0.369 (0.363, 0.375)	0.358 (0.352, 0.365)

Table 4: The clustering errors of the various methods for the case when the features were dependent with $\rho_{\max} = 0.8$, and the expectations were along the cube as specified in Item 1.2 in Section 8.1. Here $n = 20$, and the dimensionality was $d = 4$. $N = 4$.

8.5. Results: $n = 100$ Data Points

We now report the comparative results obtained for four experimental settings for the case when the date involved $n = 100$ data points. Again, in each case, we report the error rates (recorded as per Section 8.3) with their corresponding 95% confidence intervals. The results are tabulated in the following four tables:

- Table 5 illustrates the case when the features were independent (the values between the dimensions), and the expectations were along the line. This is the scenario given in Item 1.1 in Section 8.1.

- Table 6 illustrates the case when the values between the dimensions were *dependent*, and the expectations were along the line as given in Item 1.1 in Section 8.1.

- Table 7 illustrates the case when the features were independent, and the expectations were along the cube. This is the scenario given in Item 1.2 in Section 8.1.

- Finally, Table 8 illustrates the case when the values between the dimensions were *dependent* and the expectations were along the cube as given in Item 1.2 in Section 8.1.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.071 (0.063, 0.08)	0.071 (0.066, 0.076)	0.069 (0.066, 0.072)
Bottom-Up	0.232 (0.217, 0.246)	0.388 (0.344, 0.432)	0.325 (0.291, 0.36)
Bottom-up Distance UD	0.383 (0.361, 0.404)	0.414 (0.388, 0.44)	0.385 (0.364, 0.405)
Top-Down	0.18 (0.174, 0.185)	0.172 (0.161, 0.184)	0.189 (0.178, 0.199)
Top-Down Distance UD	0.271 (0.262, 0.279)	0.333 (0.317, 0.349)	0.317 (0.297, 0.337)

Table 5: The clustering errors of the various methods for the case when the features were independent, and the expectations were along the line as specified in Item 1.1 in Section 8.1. Here $n = 100$, and the dimensionality $d = 4$.

The analysis of the above tabulated results follows.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.067 (0.059, 0.076)	0.092 (0.081, 0.102)	0.115 (0.098, 0.132)
Bottom-Up	0.241 (0.23, 0.252)	0.41 (0.364, 0.456)	0.404 (0.358, 0.449)
Bottom-up Distance UD	0.396 (0.371, 0.421)	0.444 (0.415, 0.473)	0.409 (0.386, 0.432)
Top-Down	0.174 (0.162, 0.187)	0.174 (0.159, 0.189)	0.191 (0.176, 0.205)
Top-Down Distance UD	0.273 (0.261, 0.284)	0.336 (0.317, 0.356)	0.326 (0.31, 0.342)

Table 6: The clustering errors of the various methods for the case when the features were dependent with $\rho_{\max} = 0.8$, and the expectations were along the line as specified in Item 1.1 in Section 8.1. Here $n = 100$, and the dimensionality was $d = 4$.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.084 (0.068, 0.101)	0.108 (0.091, 0.126)	0.118 (0.1, 0.137)
Bottom-Up	0.248 (0.205, 0.291)	0.414 (0.359, 0.469)	0.417 (0.36, 0.474)
Bottom-up Distance UD	0.447 (0.415, 0.478)	0.503 (0.469, 0.537)	0.48 (0.449, 0.51)
Top-Down	0.099 (0.08, 0.118)	0.126 (0.107, 0.146)	0.147 (0.125, 0.168)
Top-Down Distance UD	0.335 (0.31, 0.359)	0.389 (0.361, 0.417)	0.355 (0.332, 0.378)

Table 7: The clustering errors of the various methods for the case when the features were independent, and the expectations were along the cube as specified in Item 1.2 in Section 8.1. Here $n = 100$, and the dimensionality $d = 4$.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.079 (0.064, 0.094)	0.089 (0.072, 0.107)	0.082 (0.065, 0.099)
Bottom-Up	0.247 (0.203, 0.291)	0.322 (0.268, 0.375)	0.369 (0.311, 0.428)
Bottom-up Distance UD	0.479 (0.451, 0.507)	0.481 (0.451, 0.512)	0.504 (0.476, 0.532)
Top-Down	0.108 (0.087, 0.129)	0.097 (0.08, 0.114)	0.126 (0.104, 0.148)
Top-Down Distance UD	0.346 (0.32, 0.372)	0.364 (0.342, 0.385)	0.334 (0.311, 0.358)

Table 8: The clustering errors of the various methods for the case when the features were dependent with $\rho_{\max} = 0.8$, and the expectations were along the cube as specified in Item 1.2 in Section 8.1. Here $n = 100$, and the dimensionality was $d = 4$. $N = 4$.

8.6. Analysis of the Results

We shall now draw some qualitative conclusions based on the empirical and quantitative results reported in Sections 8.4 and 8.5. This will help us illustrate the behavior of our devised flat and hierarchical clustering methods.

Flat Approaches: The first very interesting conclusion that we can draw from these results is that both the Bayes and the AB Flat approaches are comparable. This is, actually, quite remarkable because, unlike the Bayesian schemes, the AB assigns the unassigned samples based on distant quantiles and not on their centroids. Thus, for example, the clustering errors are 0.071, 0.071 and 0.069 (See Table 5) when the $n = 100$, and for the Bayes, the AB with a non-parametric estimation of the quantiloids, and the AB with a parametric estimation of the quantiloids. The difference is more noticeable when $n = 20$, but understandably the estimates of the quantiles in the $d = 4$-dimensional space is poor when we are utilizing only

$n = 20$ samples per cluster. The errors between the methods is small for both the cases when the means fall along the line, as specified in Item 1.1 in Section 8.1, or in the cube as specified in Item 1.2 in Section 8.1.

Top-Down Approaches: As in the case of the flat clustering, the Top-Down approach performs very well. Indeed, the AB paradigm performs almost as well as the Bayes. The same comment about this being non-intuitive is pertinent here too. Thus, for example, the clustering errors are 0.18, 0.172 and 0.189 (See Table 5) when the $n = 100$, and for the Bayes, the AB with a non-parametric estimation of the quantiloids, and the AB with a parametric estimation of the quantiloids. Remarkably, the AB method is marginally better than the Bayes.

Botton-Up Approaches: Bottom-Up approaches (as opposed to the others studied), in general, work rather poorly when it concerns both the Bayesian and AB schemes. A Bottom-Up approach performs a lot of merging of small clusters[11] and this requires the utilization of good "distance measures" between small-sized clusters. Understanding this is far from trivial especially in AB schemes. They face a major challenge here. This is more of a "conceptual" challenge, because the quantiles are not readily computed when (a) the sizes of the sets to be merged is small, and (b) when the cardinality of the data sets being processed are themselves, small. Strangely, the bottom up approaches performed poorer when the number of data points in each cluster increased ($n = 100$) for both the Bayes and the AB schemes. Since we are building larger clusters from smaller ones, and since the small clusters are better represented by their means (rather than their quantiles), we can conclusively infer that if one opts to use a Bottom-Up approach, it is always better to use a Bayesian methodology than an AB.

Data Size Considerations: One can unequivocally observe that other than for the scenario mentioned above, the accuracy of all the approaches improved as the number of points n increased from 20 to 100. This is, of course, intuitive and confirms that the accuracy of the estimation of the quantiloids increases with the number of data points. This, in turn, increases the accuracy of the AB flat and Top-Down approaches. Thus, the errors obtained were 0.13 and 0.099 respectively when the number of samples increased from 20 to 100 when

[11] Indeed, whenever we have a single data point x, in a cluster, it is not possible to compute the quantiles. In order to avoid this, we compute the quantiles for these clusters as $[x - \epsilon, x + \epsilon]$, with a small value of ϵ, i.e., (10^{-4}).

the features were dependent with $\rho_{\max} = 0.8$, and the expectations were along the cube as specified in Item 1.2 in Section 8.1 (please see the third rows of Tables 4 and 8). These results are typical.

Topology Considerations: One can also observe the remarkable conclusion that the topology of the points does not degrade the AB strategy. Indeed, when the expectations fell along a line, the results of the AB scheme and the Bayes were again comparable.

Non-parametric *vs.* Parametric Considerations: Another very interesting conclusion that we obtained is that computing the quantiles non-parametrically (column two in the tables) or parametrically (column three) affect the results minimally. However, the non-parametric method yielded superior accuracy for the AB schemes. This again, is intuitive, because the quantiles estimated using low-quality estimates of the mean and variances will, understandably, be poor. Thus, the errors obtained were 0.172 and 0.189 respectively when the estimates were non-parametric and parametric respectively, when the features were independent and the expectations were along the line as specified in Item 1.1 in Section 8.1 (please see the third rows of Tables 5).

Independence *vs.* Dependence Considerations: It is extremely interesting to note that the AB paradigm reported no problems when there was a dependence between the different dimensions of the data i.e., when $\rho_{\max} > 0$. Rather, by comparing Tables 3 and 4, and Tables 7 and 8, it appears as if the difference to the Bayes paradigm was *less* than when the data was dependent.

Uni-dimensional Approximations: The most daring step we took was when we attempted to reduce all our clustering decisions based on the behavior of a single dimension. This was, of course, a "shot in the dark", and it was not too surprising that both the Bayes and the AB schemes behaved poorly. Indeed, when we dealt with clustering based on this uni-dimensional paradigm, all schemes (bottom-up and top-down) without exception yielded a low accuracy. That being said, invoking an uni-dimensional clustering paradigm works better for expectations along the line than for the case when the expectation were within a cube. The reason for this very interesting observation may be due to the fact that when we generated the data along a line, we effectively "reduced" the dependence between the dimensions.

Real-life data sets: The problem associated with comparing different clustering algorithms on real-life data sets is that we are not aware of the ground truth – i.e., the true clustering. The other option is to compare the clustering achieved by the different methods. A study about how this can be accomplished is currently being undertaken. Although we do have some preliminary results, they are still half-baked are are not suitable for publication. We hope to publish them in the near future.

9. Experimental Results: Real-life data

In this section, we report the results of evaluating the performance of our Anti-Bayesian clustering algorithms for real-life data. To render the task challenging, we have done this testing on five recently-proposed data sets. They are:

- Seeds data set: This data set contains the measurements of the geometrical properties of kernels belonging to three different varieties of wheat. A soft X-ray technique and the so-called GRAINS package were used to construct seven real-valued attributes, namely the area A, perimeter P, compactness $C = 4\pi A/P^2$, length and width of the kernels, their asymmetry coefficients, and the lengths of their grooves. The data has a total of 210 observations. Additional details about this data set are found in [2].

- Perfume data set: This data consists of the odors of 20 different perfumes. The data was obtained by using a hand-held odor meter (OMX-GR sensor) per second for a period of 28 seconds, resulting in a total of 28 attributes per perfume. For more details about this set, please see [10].

- Stone flakes data set: Stone flakes are waste products that emerge from the process of production of stone tools in the prehistoric era. The variables are the means of eight different geometric and stylistic features of the flakes contained in different inventories. The data set has a total of 79 observations. Additional details on this data set are found in [20].

- Turkiye student evaluation data set: This data set contains a total of 5,820 evaluation scores provided by the students from Gazi University in Ankara (Turkey). There is a total of 28 course-specific questions with additional 5 attributes. For more details on the data set, please see [5].

- User knowledge modeling data set: This is a dataset that contains details about the status of students' knowledge concerning the subject of Electrical DC Machines. The data set has a total of 403 observations and five attributes. More detailed information about this dataset is found in [9].

There is a fundamental problem that has to be resolved when we are dealing with real-life data sets. This involves the challenge of evaluating the performance of the various clustering algorithms on these sets because the true identity of the clusters is unknown. Of course, this is because, unlike the synthetic data sets from the previous section, the real-life data examined does not provide the "true" clusters nor the points within these clusters, from which the data originates. We have to, therefore, resort to a strategy by which we can procure a good approximation of the ground truth. To achieve this, we have invoked the following procedure on the "raw" unprocessed real-life sets.

For a given data set with data point $\{x_1, x_2, \dots, x_n\}$, we do the following:

1. Let us suppose that we have J algorithms, $\{A_1, A_2, \dots, A_J\}$, all of which are accepted in the literature as being standard and accurate clustering algorithms[12].
2. Cluster the data using each of the algorithms $A_1, A_2, \dots, A_J$. Denote the results of these clustering algorithms as $C_1, C_2, \dots, C_J$.
3. Go through each pair of data points, $x_i, x_j, i \neq j$, in the same manner as in Section 8.3. If each of the clustering results $C_1, C_2, \dots, C_J$ agree, i.e., each algorithm has placed x_i and x_j in the same cluster or each algorithm has assigned x_i and x_j, in different clusters, we save the pair as a test pair. If some of the algorithms disagree, we do not use that pair when evaluating the other algorithms. Let Ω denote the set of all the resulting test pairs.
4. To measure the performance of another algorithm A^*, we cluster the data using this algorithm. Let C^* denote the result of clustering using A^*. We now go through all the test pairs from the previous step, i.e., from Item 3 above, and compute the proportion of the test pairs in which the clustering result C^* agrees with the clustering from $C_1, C_2, \dots, C_J$.

To actually execute the above test procedure, we use the following $J = 3$ alternative clustering algorithms:

- Affinity Propagation: The Affinity Propagation algorithm creates clusters by sending messages between pairs of samples until a convergence is attained. A dataset is then described using a small number of exemplars, which are identified as those most representative of other samples. The messages sent between the pairs represent the

[12] Obviously, it is impossible to get any understanding of the true nature of the data if we don't even have a strategy for obtaining an *approximation* of the ground truth. Our aim is to obtain this by using the J algorithms, $\{A_1, A_2, \dots, A_J\}$.

suitability for one sample to be the exemplar of the other, which is updated in response to the values from other pairs. This updating occurs iteratively until a convergent behavior is observed, at which point the final exemplars are chosen, and hence the final clustering is obtained.

- Gaussian Mixture Distribution: In this approach, we assume that each data point is an outcome from a Gaussian mixture distribution. We estimate the parameter of the Gaussian mixture using the EM algorithm [3]. To save computation time, we assume independence between the dimensions, i.e., we assume that the covariance matrix of each Gaussian distribution is a diagonal matrix. After the parameters are estimated, the points are placed into clusters based on the Mahalanobis distance from the mean vectors of the Gaussian distributions.
- Mini Batch k-means: The Mini Batch k-means is a variant of the k-means algorithm which uses mini-batches to reduce the computation time, while still attempting to optimize the same objective function. Mini-batches are subsets of the input data, randomly sampled in each training iteration. These mini-batches drastically reduce the amount of computation required to converge to a local solution.

For both the Gaussian mixture and Mini Batch k-means, the number of clusters can be set, while this is not an option for the affinity propagation algorithm. Therefore when computing the test pairs, Ω, using the approach above, we first run the Affinity Propagation algorithm. The number of clusters computed by this algorithm is, thus, subsequently used for the two other algorithms. The results of the clustering for the three algorithms for the five data sets is given in Table 9.

Data set	No. clusters	No. evaluated	No. agree	Portion agree
Seeds	11	43890	36850	0.839
Perfume	5	380	300	0.789
Stone flakes	9	6162	4934	0.800
Turkiye Eval.	301	33866580	31249324	0.923
User knowledge	29	162006	148282	0.915

Table 9: Results obtained by invoking the clustering using the three algorithms, namely Affinity Propagation, Guassian mixture, and the Mini Batch k-means clustering for each of the five data sets. The columns from left to right are the number of clusters from the Affinity Propagation algorithm, the total number of pairs of points evaluated, the number of pairs of points in which the three algorithms agree, i.e., the number of test pairs in the set Ω, and the portion of the pair of points in which the algorithms agree.

Having established a set of test pairs, Ω, using the procedure and clustering algorithms presented above, we evaluate the clustering performance of the 15 algorithms presented in this paper, i.e., the same 15 algorithms evaluated in the previous section involving synthetic datasets. For each clustering result for each of the 15 algorithms, we use the same number of clusters used by the three alternative algorithms used to compute the pair of points in the test set Ω. For all the algorithms based on the k-means or Mini Batch k-means algorithm, the clustering depends on the initial clustering used. In order to reduce the Monte Carlo error with respect to this, we repeated the computation of the test set, Ω, with different initial conditions for the k-means Mini Batch scheme. Further, the Flat clustering and Top-down clustering algorithms described earlier, also depend on the initial clustering. Therefore, for each of the 20 times that the test set Ω were computed, the clusterings based on the Flat clustering and the Top-down clustering paradigms were repeated 50 times. By doing this, we are guaranteed that the experiments conducted and the results reported are rigorous.

Tables $10 - 14$ show the results for each of the five data sets. Not that the Buttup Up algorithm were not run for the Turkiye student evaluation data set. The algorithm is too computer demanding for such a large data set.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.026	0.029	0.029
Bottom-Up	0.191	0.512	0.394
Bottom-up Distance UD	0.217	0.428	0.428
Top-Down	0.049	0.055	0.051
Top-Down Distance UD	0.505	0.469	0.511

Table 10: Results of the various clustering algorithms for the Seed data set. The table reports the portion of the pair of points in the test set, Ω, in which the clustering algorithms disagree with the three alternative clustering algorithms.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.057	0.099	0.077
Bottom-Up	0.412	0.412	0.412
Bottom-up Distance UD	0.572	0.572	0.572
Top-Down	0.087	0.077	0.063
Top-Down Distance UD	0.456	0.456	0.456

Table 11: Results of the various clustering algorithms for the Perfume data set. The table reports the portion of the pair of points in the test set, Ω, in which the clustering algorithms disagree with the three alternative clustering algorithms.

For the different data sets, we see that the flat clustering and the top-down algorithms perform the best. This is also in agreement with what we observed in the synthetic example. For these algorithms we see that the Bayesian and the Anti-Bayesian alternatives perform about equally well. The performance of the Bottom-up Distance UD and Top-Down Distance

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.047	0.052	0.049
Bottom-Up	0.290	0.805	0.751
Bottom-up Distance UD	0.588	0.600	0.474
Top-Down	0.077	0.088	0.118
Top-Down Distance UD	0.647	0.223	0.371

Table 12: Results of the various clustering algorithms for the Stone Flakes data set. The table reports the portion of the pair of points in the test set, Ω, in which the clustering algorithms disagree with the three alternative clustering algorithms.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.001	0.002	0.052
Bottom-Up	- - -	- - -	- - -
Bottom-up Distance UD	0.737	0.627	0.394
Top-Down	0.041	0.038	0.091
Top-Down Distance UD	0.271	0.034	0.602

Table 13: Results of the various clustering algorithms for the Turkiye student evaluation data set. The table reports the portion of the pair of points in the test set, Ω, in which the clustering algorithms disagree with the three alternative clustering algorithms.

	Bayes	AB Non-parametric	AB Parametric
Flat clustering	0.013	0.015	0.017
Bottom-Up	0.930	0.872	0.910
Bottom-up Distance UD	0.601	0.873	0.595
Top-Down	0.049	0.083	0.090
Top-Down Distance UD	0.707	0.124	0.181

Table 14: Results of the various clustering algorithms for the Knowledge Modeling data set. The table reports the portion of the pair of points in the test set, Ω, in which the clustering algorithms disagree with the three alternative clustering algorithms.

UD algorithms are quite poor in terms of their respective performances, and this is again in agreement with what we observed in the synthetic example. Also for these algorithms, the Bayesian and the Anti-Bayesian alternatives perform about equally well. But it is worth emphasizing that the Anti-Bayesian alternatives *performed far better then the Bayesian alternative* when using the Top-Down Distance UD algorithm on the Stone Flakes data set. For the Bottom-Up algorithm the Bayesian and the Anti-Bayesian alternatives perform about equally well except that the Bayesian alternative performs the best in the case of Seed and the Stone flakes data sets.

Overall, we see that the Anti-Bayesian algorithms perform very well compared to the Bayesian alternatives on the different real-life data sets. This is quite remarkable when we consider that the points are non-intuitively assigned to clusters based on quantiloids that are distant from the means. It is also important to note that both the Mini Batch k-means and the Gaussian mixture approach are closely related to the Bayesian paradigm. One would, therefore, expect that the Bayesian algorithms presented in this paper agree to a higher degree with the pairs of points in the test set, Ω, which is also what one observes. The fact that an AB-based scheme could attain to a comparable accurcay is, really, quite noteworthy.

10. Conclusion

In this paper, we have considered an "Anti-Bayesian" (AB) paradigm for clustering. All of the reported clustering algorithms (except the one reported in [6]) operate on Bayesian principles, (where the Bayesian principle corresponds to assigning the unlabelled samples to the cluster whose mean (or centroid) is the closest). Our aim here has been to see if the "Anti-Bayesian" (AB) classification philosophy, introduced recently by Oommen and his co-authors, can be extended into the domain of clustering. The AB principle involves classification based on the non-central *quantiles* of the distributions, which involves utilizing the information resident in the outlier samples.

In this paper, we have extended the first-reported AB clustering methods proposed in [6]. This paper has extended the results of [6] in many directions. Firstly, we have generalized our previous AB clustering [6], initially proposed for handling uni-dimensional and two-dimensional spaces, to arbitrary d-dimensional spaces using their so-called *"quantiloids"*. Secondly, we have extended the AB paradigm to consider how the clustering can be achieved in hierarchical ways, where we have analyzed both the Top-Down and the Bottom-Up clustering options. The AB paradigm can also use an anti-Naïve-Bayesian *computational* mechanisms. The paper contains the results of extensive experimentation on artificial datasets and on five recently-introduced real-life datasets. These results clearly demonstrate that our AB clustering schemes achieve results competitive to the state-of-the-art Flat, Top-Down and Bottom-Up clustering approaches.

In the future, we envisage an ambitious goal of devising an AB clustering method based on applying majority voting on the decision made in each dimension of the quantile vector. Further, since the novel methods introduced are, in one sense, orthogonal to the ones reported in the field, the possibilities to merge (fuse) these methods with the ones currently used, are huge. The research avenues that are open due to the introduction of the concept of "quantiloids" is, in our opinion, vast.

[1] Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. (1999). Optics: Ordering points to identify the clustering structure. pages 49–60. ACM Press.

[2] Charytanowicz, M., Niewczas, J., Kulczycki, P., Kowalski, P. A., Łukasik, S., and Żak, S. (2010). Complete gradient clustering algorithm for features analysis of x-ray images. In *Information technologies in biomedicine*, pages 15–24. Springer.

[3] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38.

[4] Ester, M., Kriegel, H.-P., S, J., and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. pages 226–231. AAAI Press.

[5] Gunduz, N. and Fokoue, E. (2013). UCI machine learning repository.

[6] Hammer, H. L., Yazidi, A., and Oommen, B. J. (2015). A novel clustering algorithm based on a non-parametric "anti-bayesian" paradigm. In *Current Approaches in Applied Artificial Intelligence - 28th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, IEA/AIE 2015, Seoul, South Korea, June 10-12, 2015, Proceedings*, pages 536–545.

[7] Jain, A. K. and Dubes, R. C. (1988). *Algorithms for Clustering Dats.* Prentice Hall, Englewood Cliffs, New Jersey, USA.

[8] Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, 31(3):264–323.

[9] Kahraman, H. T., Sagiroglu, S., and Colak, I. (2013). The development of intuitive knowledge classifier and the modeling of domain dependent data. *Knowledge-Based Systems*, 37:283–295.

[10] Karlik, B. and Al-Bastaki, Y. (2004). Real time monitoring odor sensing system using omx-gr sensor and neural network. *WSEAS Transactions on Electronics*, 1(2):337–342.

[11] Langfelder, P. and Horvath, S. (2015). WGCNA. `http://www.inside-r.org/packages/cran/WGCNA/docs/nearestCentroidPredictor`. [Online; accessed 11-November-2015].

[12] Murtagh, F. and Contreras, P. (2011). Methods of hierarchical clustering. *CoRR*, abs/1105.0121.

[13] Oommen, B. J., Khoury, R., and Schmidt, A. (2015). Text classification using novel "anti-bayesian" techniques. In *Computational Collective Intelligence*, pages 1–15. Springer.

[14] Oommen, B. J. and Thomas, A. (2014). "Anti–Bayesian" parametric pattern classification using order statistics criteria for some members of the exponential family. *Pattern Recognition*, 47(1):40–55.

[15] Sibson, R. (1973). SLINK: An optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1):30–34.

[16] Thomas, A. and Oommen, B. J. (2013a). The fundamental theory of optimal "anti-bayesian" parametric pattern classification using order statistics criteria. *Pattern Recognition*, 46(1):376–388.

[17] Thomas, A. and Oommen, B. J. (2013b). A novel border identification algorithm based on an "anti-Bayesian" paradigm. In *Computer Analysis of Images and Patterns*, pages 196–203. Springer.

[18] Thomas, A. and Oommen, B. J. (2013c). Order statistics-based parametric classification for multi-dimensional distributions. *Pattern Recognition*, 46(12):3472–3482.

[19] Thomas, A. and Oommen, B. J. (2013d). Ultimate order statistics-based prototype reduction schemes. In *Proceedings of AI'13, the 2013 Australasian Joint Conference on Artificial Intelligence*, pages 421–433. Springer.

[20] WEBER, T. (2009). The lower/middle palaeolithic transition-is there a lower/middle palaeolithic transition? *Preistoria Alpina*, 44:17–24.

[21] Xu, R. and Wunsch, D. (2005). Survey of clustering algorithms. *Trans. Neur. Netw.*, 16(3):645–678.