Mean Shift Clustering Algorithm for Data with Missing Values

Loai AbdAllah and Ilan Shimshoni

[1] Department of Mathematics, University of Haifa, Israel
Department of Mathematics and Computer Science, The College of Sakhnin for Teacher Education, Israel
[2] Department of Information Systems, University of Haifa, Israel
loai1984@gmail.com & ishimshoni@mis.haifa.ac.il

Abstract. Missing values in data are common in real world applications. There are several methods that deal with this problem. In this research we developed a new version of the *mean shift* clustering algorithm that deals with datasets with missing values. We use a weighted distance function that deals with datasets with missing values, that was defined in our previous work. To compute the distance between two points that may have attributes with missing values, only the *mean* and the *variance* of the distribution of the attribute are required. Thus, after they have been computed, the distance can be computed in $O(1)$. Furthermore, we use this distance to derive a formula for computing the *mean shift* vector for each data point, showing that the *mean shift* runtime complexity is the same as the Euclidian *mean shift* runtime. We experimented on six standard numerical datasets from different fields. On these datasets we simulated missing values and compared the performance of the *mean shift* clustering algorithm using our distance and the suggested mean shift vector to other three basic methods. Our experiments show that *mean shift* using our distance function outperforms *mean shift* using other methods for dealing with missing values.

Keywords: Missing values; Distance metric; Weighted Euclidian distance; Clustering; Mean Shift.

1 Introduction

Mean shift is a non-parametric iterative clustering algorithm. The fact that mean shift does not require prior knowledge of the number of clusters, and does not constrain the shape of the clusters, makes it ideal for handling clusters of arbitrary shape and number. It is also an iterative technique, but instead of the means, it estimates the modes of the multivariate distribution underlying the feature space. The number of clusters is obtained automatically by finding the centers of the densest regions in the space (the modes). The density is evaluated using kernel density estimation which is a non-parametric way to estimate the density function of a random variable. It is also called the Parzen window technique.

Mean shift was first proposed by Fukunaga and Hostetler [7]. It was then adapted by Cheng [3] for the purpose of image analysis. Later Comaniciu and Meer [5,4] successfully applied it to image segmentation and tracking. Tao, Jin and Zhang [13] use it in color image segmentation and DeMenthon and Megret [6] employed it for spatio-temporal segmentation of video sequences in a $7D$ feature space.

The problem with this algorithm is that it can not deal with datasets that contain missing values which are common in many real world datasets. Missing values can be caused by human error, equipment failure, system generated errors, and so on. In this paper we have developed a *mean shift* clustering algorithm over datasets with missing values based on the distance function we developed in [1].

Several methods have been proposed to deal with missing data. These methods can be classified into two basic categories: (a) **Case deletion** method, which ignores all the instances with missing values and performs the analysis on the rest. This method has two obvious disadvantages: (1) A substantial decrease in the size of the dataset available for the analysis. (2) The data are not always missing at random that may affect the distribution of the other features [15]. (b) Missing data imputation, which replaces each missing value with a known value according to the dataset distribution. It is important to note that by using this method the mean shift procedure can run like on the complete dataset (each missing replaced with known value). But, as we show in this paper, they perform poorly and our proposed method yields better results.

A common method that imputes missing data is the **Most Common Attribute Value (MCA)** method. The value of the attribute that occurs most often is selected to be the value for all the unknown values of the attribute [9]. One main drawback of this method is that it ignores the other possible values of the attribute and their distribution.

The main idea of the **Mean Imputation (MI)** method is to replace a data point with missing values with the mean of all the instances in the data. However, using a fixed instance to replace all the instances with missing values will change the characteristics of the original dataset. Ignoring the relationship among attributes will bias the performance of subsequent data mining algorithms. A variant of this method is to replace the missing data for a given attribute with the **M**ean of all known values of that **A**ttribute-**MA** (i.e., the mean of each attribute) in the coordinate where the instance with missing data belongs [10]. This method has the same drawback as the MCA method: both methods represent the missing value with one value and ignore all the other possible values.

The k**-Nearest Neighbor Imputation** method uses the kNN algorithm (using only the known values) to estimate and replace the missing data [16,2] by looking for the most similar instances. Efficiency is the biggest obstacle for this method. Moreover, the selected value of k and the measure of similarity will affect the results greatly.

Finally, the **k-means Imputation** method predicts missing attribute values using simple k-means clustering. This approach deals with labeled datasets. [12].

Again, the main drawbacks of each suggested method can be summarized as inefficiency and inability to approximate the missing value. In our previous work [1] we developed a method to compute the distance function (MD_E) that is not only efficient but also takes into account the distribution of each attribute. In the computation procedure we take into account all the possible values with their probabilities, which are computed according to the attribute's distribution. This is in contrast to the MCA and the MA methods, which replace each missing value only with the mode or the mean of each attribute.

We can summarize our distance for the three possible cases of the two values: (1) Both values are known. In that case, the distance function is identical to the Euclidian distance. (2) One value is missing. In that case, the distance will be computed only according to the two statistics (*mean* and *variance*), where the distance equals the Euclidian distance between the known value and the *mean* plus the *variance* of that attribute (mean squared error). (3) Both values are missing. In that case, the distance will be computed only according to the *variance* of that attribute, and it equals twice the *variance*. Therefore, the runtime of this distance function is the same as for the Euclidian distance.

In order to develop a mean shift algorithm for data with missing values we derived a formula for computing the gradient function of the local estimated density. For this case the runtime complexity of the resulting gradient function using the MD_E distance is the same as the standard one using the Euclidean distance. To measure the ability of the suggested mean shift vector using the MD_E distance to represent the actual mean shift vector when the dataset is complete, we integrated it within the mean shift clustering algorithm on datasets with missing values. The developed algorithm not only yields better results than the other methods, as can be seen in the experiments, but also preserves the runtime of the mean shift clustering algorithms which deals with complete data. We experimented on six standard numerical datasets from different fields from the Speech and Image Processing Unit [14]. Our experiments show that the performance of the mean shift algorithm using our distance function and the proposed mean shift vector were superior to mean shift using other methods.

The paper is organized as follows. A review of our distance function (MD_E) is described in Section 2. An overview of the mean shift clustering algorithm is presented in Section 3. Section 4 describes how to compute and to integrate the (MD_E) distance and the computed mean shift vector within the mean shift clustering algorithm. Experimental results of running the mean shift clustering algorithm on the Speech and Image Processing Unit [14] datasets are presented in Section 5. Finally, our conclusions are presented in Section 6.

2 Our Distance Measure

In this section we give a short overview of our distance function developed in [1]. Let A be a set of points. For the ith coordinate C_i, the conditional probability

for C_i will be computed according to the known values for this coordinate from A (i.e., $P(c_i) \sim \chi_i$), where χ_i is the distribution of the ith coordinate.

Our method can be generalized to deal with coordinates whose measurements are dependant, but for simplicity we assume that these measurements are independent. Under these assumptions we will treat each coordinate separately.

Given two sample points X and Y from A, the goal is to compute the distance between them. Let x_i and y_i be the ith coordinate values from points X, Y respectively. There are three possible cases for the values of x_i and y_i: (1) Both values are given. (2) One value is missing. (3) Both values are missing.

Two values are known: When the values of x_i and y_i are given, the distance between them will be defined as the Euclidian distance:

$$D_E(x_i, y_i) = (x_i - y_i)^2. \tag{1}$$

One value is missing: Suppose that x_i is missing and the value y_i is given. Since the value of x_i is unknown, we cannot compute its Euclidian distance. Instead we model the distance as a random selection of a point from the distribution of its coordinate χ_i and compute its distance. The mean of this computation is our distance. We will estimate this value as follows: We divide the range of c_i (i.e., $[\min(c_i), \max(c_i)]$) into l equal intervals $(\Delta_1, \ldots, \Delta_l)$.Let m_j be the center point of interval Δ_j. For it we can estimate its probability density $p(m_j)$ using the KDE. The probability for the jth interval Δj is therefore:

$$P(\Delta j) = p(m_j) \cdot \frac{\max(c_i) - \min(c_i)}{l} \approx \int\limits_{x \in \Delta_j} p(x)dx.$$

As a result, we approximate the mean Euclidian distance (MD_E) between y_i and the distribution as:

$$MD_E(\chi_i, y_i) = \sum_{j=1}^{l} P(\Delta j) D_E(m_j, y_i).$$

This metric measures the distance between y_i and each suggested value of x_i and takes into account the probability for this value according to the evaluated probability distribution. Computing the distance in this way is an expensive procedure. We therefore turn to the continuous probability density function $p(x)$ in hope of getting an efficient formula. In this case,

$$\begin{aligned} MD_E(\chi_i, y_i) &= \int p(x)(x - y_i)^2 dx = \int p(x)x^2 + p(x)y_i^2 - p(x)2xy_i)dx \\ &= \Big(E[x^2] + y_i^2 - 2y_i E[x]\Big) = \Big((y_i^2 - E[x])^2 - E[x]^2 + E[x^2]\Big) \\ &= \Big((y_i - \mu_i)^2 + \sigma_i^2\Big), \end{aligned} \tag{2}$$

where μ_i, σ_i^2 are the *mean* and the *variance* for all the known values of the attribute. The distance computed according to the last equation has several important properties:

- It is identical to the Euclidian distance, when the dataset is complete. In this case $\mu_i = x_i, \sigma_i = 0$ (*Dirac delta function*).
- It can be applied to any distribution and can be used in any algorithm that uses a distance function.
- It is simple to implement.
- It is very efficient because, to compute the MD_E between two values when one of them is missing, we need only to compute in advance the two statistics (i.e., μ and σ) for each coordinate. After that the runtime is $O(1)$.
- When the variance is small, the real value of the missing value is close to the mean of the attribute and our distance will converge to the Euclidian distance.
- When the variance is large, the uncertainty is high, and as a result the distance should be large.
- It is basically the sum of the bias term $(y_i - \mu_i)^2$ and the *variance* σ_i^2, yielding the mean squared error (MSE).

In contrast to the other imputation methods we do not replace the missing value with any constant value. Moreover, it differs from the **Most Common Attribute Value** method, where the value of the most frequent attribute is selected to be the value for all the unknown values of the attribute, implying that the probability of the most common attribute value is 1 and 0 for all other possible values. Our distance also differs from the **Mean Attribute Value method**, where the mean of a specific attribute is selected to replace the unknown value of the attribute because the dispersion of the values in the distribution is not taken into account.

The two values are missing: In this case, in order to estimate the mean Euclidian distance, we have to randomly select values for both x_i and y_i. Both these values are selected from distribution χ_i. To compute the distance, the following double sum has to be computed.

$$MD_E(x_i, y_i) = \sum_{q=1}^{l-1} \sum_{j=1}^{l-1} P(\Delta_{1q}) P(\Delta_{2j}) D_E(m_{1q}, m_{2j}).$$

As we did for one missing value, we turn to the continuous case:

$$\begin{aligned} MD_E(x_i, y_i) &= \int \int p(x)p(y)(x-y)^2 dxdy \\ &= \left((E[x] - E[y])^2 + \sigma_x^2 + \sigma_y^2 \right) = 2\sigma_i^2. \end{aligned} \tag{3}$$

From (3) we can easily see that our metric reflects the similarity between points better than the other methods. In the previous two methods, all the missing values of an attribute are replaced by the mode or the mean of that attribute, and the distance is equal to 0 without paying any attention to the variance of the coordinates. In our metric, however, the distance depends on the variance for each coordinate, which is more logical because if the variance is larger then the distance between the possible values on average is larger.

3 Mean Shift Algorithm

For completeness we will now give a short overview of the mean shift algorithm. Here we only review some of the results described in [4,8] which should be consulted for the details. Assume that each data point $x_i \in \mathbb{R}^d, i = 1, ..., n$ is associated with a bandwidth value $h > 0$. The *sample point* density estimator at point x is

$$\hat{f}(x) = \frac{1}{nh^d} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right). \tag{4}$$

Based on a spherically symmetric kernel K with bounded support satisfying

$$K(x) = c_{k,d} k\left(\|x\|^2\right) \qquad \|x\| \leq 1 \tag{5}$$

is an adaptive nonparametric estimator of the density at location x in the feature space. The function $k(x), 0 \leq x \leq 1$, is called the *profile* of the kernel, and the normalization constant $c_{k,d}$ assures that $K(x)$ integrates to one. Employing the profile notation the density estimator (4) can be rewritten as

$$\hat{f}_{h,k}(x) = \frac{c_{k,d}}{nh^d} \sum_{i=1}^{n} k\left(\left\|\frac{x - x_i}{h}\right\|^2\right). \tag{6}$$

The first step in the analysis of a feature space with the underlying density $f(x)$ is to find the modes of the density. The modes are located among the zeros of the gradient $\nabla f(x) = 0$, and the mean shift procedure is an elegant way to locate these zeros without estimating the density.

The density gradient estimator is obtained as the gradient of the density estimator by exploiting the linearity of (6)

$$\nabla \hat{f}_{h,K}(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^{n} (x - x_i) k'\left(\left\|\frac{x - x_i}{h}\right\|^2\right). \tag{7}$$

We define the function $g(x) = -k'(x)$ that can always be defined when the derivative of the kernel profile $k(x)$ exists. Using $g(x)$ as the profile, the kernel $G(x)$ is defined as

$$G(x) = c_{g,d} g\left(\|x\|^2\right).$$

Introducing $g(x)$ into (7) yields

$$\nabla \hat{f}_{h,K}(x) = \frac{2c_{k,d}}{nh^{d+2}} \sum_{i=1}^{n} (x_i - x) g\left(\left\| \frac{x - x_i}{h} \right\|^2 \right)$$
$$= \frac{2c_{k,d}}{nh^{d+2}} \left[\sum_{i=1}^{n} g\left(\left\| \frac{x - x_i}{h} \right\|^2 \right) \right] \left[\frac{\sum_{i=1}^{n} x_i g\left(\left\| \frac{x-x_i}{h} \right\|^2 \right)}{\sum_{i=1}^{n} g\left(\left\| \frac{x-x_i}{h} \right\|^2 \right)} - x \right], \quad (8)$$

where $\sum_{i=1}^{n} g\left(\left\| \frac{x-x_i}{h} \right\|^2 \right)$ is assumed to be a positive number. Both terms of the product in (8) have special significance. The first term is proportional to the density estimate at x computed with the kernel G. The second term

$$m_G(x) = \frac{\sum_{i=1}^{n} x_i g\left(\left\| \frac{x-x_i}{h} \right\|^2 \right)}{\sum_{i=1}^{n} g\left(\left\| \frac{x-x_i}{h} \right\|^2 \right)} - x \quad (9)$$

is called the *mean shift vector*. The expression (9) shows that at location x the weighted mean of the data points selected with kernel G is proportional to the normalized density gradient estimate obtained with kernel K. The mean shift vector thus points toward the direction of maximum increase in the density. The implication of the mean shift property is that the iterative procedure

$$y_{j+1} = \frac{\sum_{i=1}^{n} x_i g\left(\left\| \frac{y_j - x_i}{h} \right\| \right)}{\sum_{i=1}^{n} g\left(\left\| \frac{y_j - x_i}{h} \right\| \right)} \qquad j = 1, 2, ... \quad (10)$$

is a hill climbing technique to the nearest stationary point of the density, i.e., a point in which the density gradient vanishes. The initial position of the kernel, the starting point of the procedure y_1 can be chosen as one of the data points x_i. Most often the points of convergence of the iterative procedure are the modes (local maxima) of the density. All points which converge to the same mode are considered members of a cluster. The number of clusters is therefore the number of modes.

4 Mean Shift Computing using The MD_E Distance

In our previous work in [1] we derived the MD_E distance and integrated it within the framework of the kNN and kMeans algorithms. In order to integrate the MD_E distance function within the framework of the mean shift algorithm, we will first compute the mean shift vector using the MD_E distance.

Using the MD_E distance the density estimator in (6) will written as

$$\hat{f}_{h,k}(x) = \frac{c_{k,d}}{nh^d} \sum_{i=1}^{n} k\left(\left\| \frac{x - x_i}{h} \right\|^2 \right) = \frac{c_{k,d}}{nh^d} \sum_{i=1}^{n} k\left(\frac{\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2}{h^2} \right). \quad (11)$$

Since each point x_i may contain missing attributes, $\hat{f}_{h,k}(x)$ will be:

$$\hat{f}_{h,k}(x) = \frac{c_{k,d}}{nh^d}\sum_{i=1}^{n} k\Bigg(\underbrace{\frac{\sum_{j=1}^{kn_i} MD_E(x^j, x_i^j)^2}{h^2}}_{\text{each } x_i \text{ has } kn_i \text{ known attributes}} + \underbrace{\frac{\sum_{j=1}^{unkn_i} MD_E(x^j, x_i^j)^2}{h^2}}_{\text{each } x_i \text{ has } unkn_i \text{ missing attributes}} \Bigg).$$

According to the definition of the MD_E distance, we obtain

$$\hat{f}_{h,k}(x) = \frac{c_{k,d}}{nh^d}\sum_{i=1}^{n} k\left(\frac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \frac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right). \quad (12)$$

Now we will compute the gradient of the density estimator in (12)

$$\begin{aligned}
\nabla\hat{f}_{h,k}(x) &= \tfrac{c_{k,d}}{nh^{d+2}}\textstyle\sum_{i=1}^{n}\left[\sum_{j=1}^{kn_i}(x^j - x_i^j)^2 + \sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2\right]' \\
&\quad \cdot k'\left(\tfrac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \tfrac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right) \\
&= \tfrac{c_{k,d}}{nh^{d+2}}\textstyle\sum_{i=1}^{n}\left[\sum_{j=1}^{kn_i}(x^j - x_i^j)^2\right]' \cdot k'\left(\tfrac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \tfrac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right) \\
&\quad + \left[\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2\right]' \cdot k'\left(\tfrac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \tfrac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right).
\end{aligned}$$

In our computation we will first deal with one coordinate l and then we will generate the computation for all the other coordinates.

$$\begin{aligned}
\Rightarrow f'_{x^l} &= \frac{2c_{k,d}}{nh^{d+2}}\sum_{i=1}^{n_l}(x^l - x_i^l)\cdot k'\left(\frac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \frac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right) \\
&\quad + \frac{2c_{k,d}}{nh^{d+2}}\sum_{i=1}^{m_l}(x^l - \mu^l)\cdot k'\left(\frac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \frac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right) \\
&= \frac{2c_{k,d}}{nh^{d+2}}\Bigg[x^l \cdot \sum_{i=1}^{n} k'\left(\frac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \frac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right) \\
&\quad - \sum_{i=1}^{n_l} x_i^l \cdot k'\left(\frac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \frac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right) \\
&\quad - \sum_{i=1}^{m_l} \mu^l \cdot k'\left(\frac{\sum_{j=1}^{kn_i}(x^j - x_i^j)^2}{h^2} + \frac{\sum_{j=1}^{unkn_i}(x^j - \mu^j)^2 + (\sigma^j)^2}{h^2}\right)\Bigg],
\end{aligned}$$

where there are n_l points for which the x^l coordinate is known, and there are m_l points where it is missing.

$$f'_{x^l} = \frac{2c_{k,d}}{nh^{d+2}} \cdot \left[\sum_{i=1}^{n} g\left(\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2\right)\right]$$
$$\cdot \left[\frac{\sum_{i=1}^{n_l} x_i^l \cdot g\left(\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2\right) + \sum_{i=1}^{m_l} \mu^l \cdot g\left(\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2\right)}{\sum_{i=1}^{n} g\left(\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2\right)} - x^l\right].$$

As a result the mean shift vector using the MD_E distance is defined as:

$$m_{MD_E,G}(x) = \frac{\sum_{i=1}^{n_l} x_i^l \cdot g\left(\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2\right) + \sum_{i=1}^{m_l} \mu^l \cdot g\left(\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2\right)}{\sum_{i=1}^{n} g\left(\sum_{j=1}^{d} MD_E(x^j, x_i^j)^2\right)} - x^l. \quad (13)$$

Now we can use this equation to run the mean shift procedure over datasets with missing values. Computing the mean shift vector using (13) seems like the computed mean shift vector using MA-method, where the mean of a specific attribute is selected to replace the unknown value of the attribute, except that in (13) the weights for the incomplete data points are lower than the computed mean shift using the MA-method, because the MD_E distance equals the Euclidian distance between the known value and the *mean* plus the *variance* of that attribute which is bigger than the Euclidian distance between the known value and the *mean* which is the distance when the MA-method is used.

The mean shift procedure starts the iterative process from each point in the dataset. It therefore also starts from incomplete points where in some cases the distance from a given incomplete point to all the other data points is larger than the bandwidth h. In that case we consider two cases, finite kernels or infinite kernels. If the kernel is infinite the computation will work as described above in (13). When the kernel is finite, there may be cases that there are no points within the radius h. In this case the mean shift will be the first nearest neighbor to the incomplete point, and here we again have two cases for the nearest neighbor. Where the nearest point is the incomplete point itself the mean shift iteration will stop and this point will also be the mode of the point. In the other case the nearest point will be another point from the data and then the next iteration of the mean shift procedure will start with this point, and the algorithm will iteratively continue until convergence. Formally, this is done by replacing the finite kernel g with an infinite kernel g_{inf} where $|g(x) - g_{inf}(x)| < \varepsilon$ for $0 \leq x \leq 1$ for an infinitesimal value ε.

5 Mean Shift Experiments on Numerical Datasets

In order to measure the ability to implement the mean shift algorithm over datasets with missing values we compare the performance of the mean shift

clustering algorithm on complete data (i.e., without missing values) to its performance on data with missing values, using our distance measure ($MS-MD_E$) and then again using MS-(MCA, MA, MI), where each missing value in each attribute is replaced using the MCA, MA or MI method respectively and then a standard mean shift is run. We use the Rand index [11], which is a measure of similarity between two data clusterings, to compare how similar the results of the standard mean shift clustering algorithm were to the results of the other algorithms for datasets with missing values.

We ran our experiments on six standard numerical datasets from the Speech and Image Processing Unit [14] from different fields: the Flame dataset, the Jain dataset, the Path based dataset, the Spiral dataset, the Compound dataset, and the Aggregation dataset. The dataset characteristics are shown in Table 1.

Table 1. Speech and Image Processing Unit Dataset properties

Dataset	Dataset size	Clusters
Flame	240×2	2
Jain	373×2	2
Path based	300×2	3
Spiral	312×2	3
Compound	399×2	6
Aggregation	788×2	7

A set consisting of 10%-40% of the data was randomly drawn from each dataset. These randomly drawn sets serve as samples of missing data, where one coordinate from each instance was randomly selected to be missing.

The results are averaged over 10 different runs on each dataset. For all the cases the bandwidth $h = 4$ was used (The standard mean shift worked well for this value on all the data sets). A resulting curve was constructed for each dataset to evaluate how well the algorithm performed, by plotting the Rand Index.

As can be seen in Figure 1, for the Flame, Spiral, Path based,Compound, and Aggregation datasets, the curves show that our mean shift clustering algorithm outperformed the other methods for all missing value percentages, while for the Jain dataset its benefit became apparent when the percent of the missing values was large, as can be seen in Figure 1(b). Moreover, we can see from these curves that the $MS - MC$ method outperforms the $MS - MA$ method for the Flame and Path Based datasets and the $MS - MC$ outperforms $MS - MA$ for the other datasets. It means that we cannot decide unequivocally which algorithm is better. On the other hand we surely can state that the $MS-MD_E$ outperforms the other methods. If the percentage of the missing values further increases the performance of the algorithm degrades gracefully.

6 Conclusions

Missing attribute values are very common in real-world datasets. Several methods have been proposed to measure the similarity between objects with missing values. In this work, we have proposed a new mean shift clustering algorithm over dataset with missing values using the MD_E distance that was presented

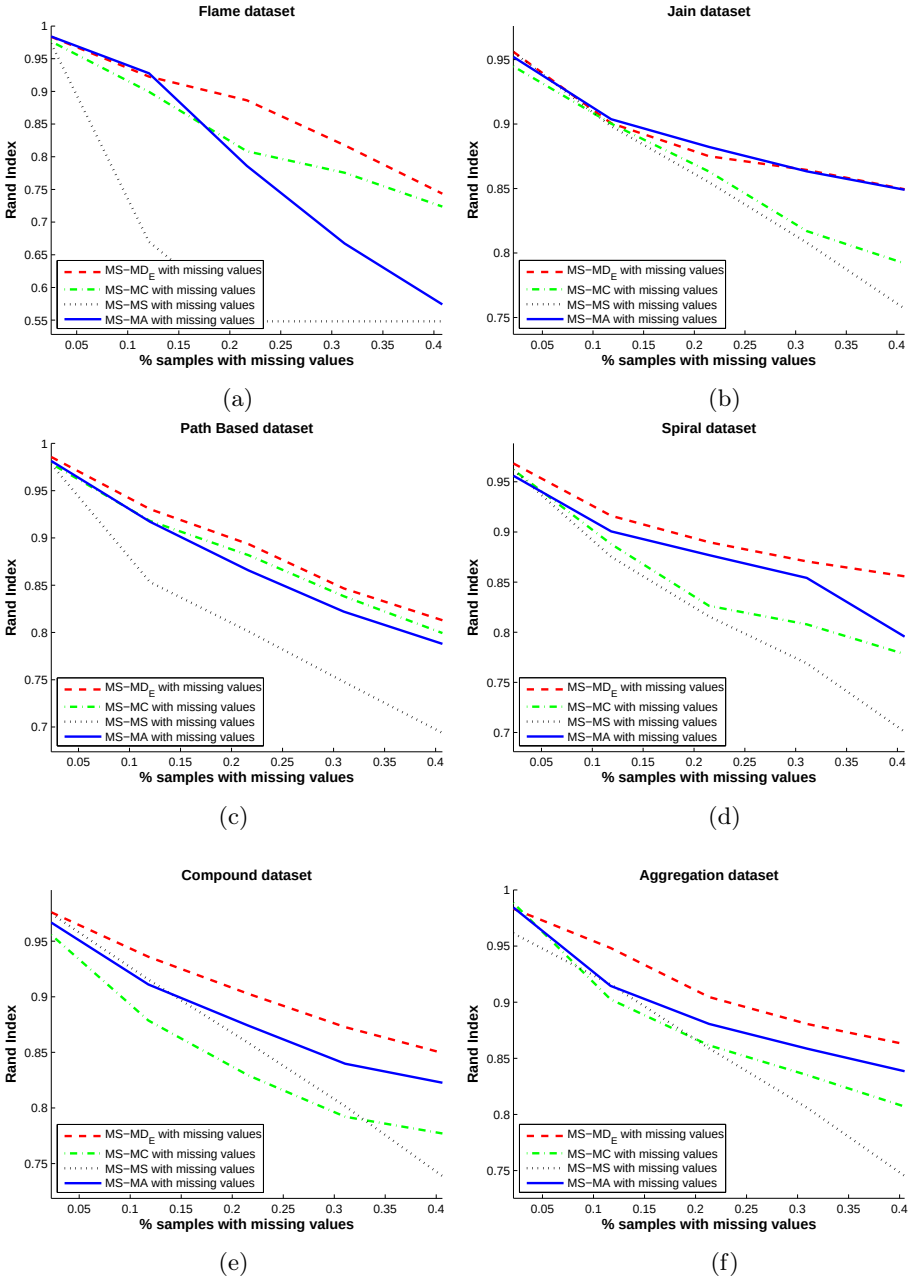

Fig. 1. Results of mean shift clustering algorithm using the different distance functions on the six datasets from the Speech and Image Processing Unit.

in [1]. In order to do that we derived a formula for the mean shift vector for a given dataset when it contains points with missing values. The computational complexity for computing the mean shift vector using the MD_E distance is the same as that of the standard mean shift vector using the Euclidian distance.

From the experiments we conclude that our method is more appropriate for measuring the mean shift vectors for objects with missing values, especially when the percent of missing values is large.

References

1. Loai AbdAllah and Ilan Shimshoni. A distance function for data with missing values and its applications on $k$nn and $k$means algorithms. *Submitted to Int. J. Advances in Data Analysis and Classification.*
2. G. Batista and M.C. Monard. An analysis of four missing data treatment methods for supervised learning. *Applied Artificial Intelligence*, 17(5-6):519–533, 2003.
3. Yizong Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. PAMI*, 17(8):790–799, 1995.
4. D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. PAMI*, 24(5):603–619, 2002.
5. D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based Object Tracking. *IEEE Trans. PAMI*, 25(5):564–577, 2003.
6. Daniel DeMenthon and Remi Megret. *Spatio-temporal segmentation of video by hierarchical mean shift analysis.* Computer Vision Laboratory, Center for Automation Research, University of Maryland, 2002.
7. K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40, 1975.
8. B. Georgescu, I. Shimshoni, and P. Meer. Mean shift based clustering in high dimensions: A texture classification example. In *Proceedings of the 9th International Conference on Computer Vision*, pages 456–463, 2003.
9. Jerzy Grzymala-Busse and Ming Hu. A comparison of several approaches to missing attribute values in data mining. In *Rough Sets and Current Trends in Computing*, pages 378–385. Springer, 2001.
10. Matteo Magnani. Techniques for dealing with missing data in knowledge discovery tasks. *Obtido http://magnanim.web.cs.unibo.it/index.html*, 15(01):2007, 2004.
11. W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.
12. Nambiraj Suguna and Keppana G Thanushkodi. Predicting missing attribute values using k-means clustering. *Journal of Computer Science*, 7(2):216–224, 2011.
13. W. Tao, H. Jin, and Y. Zhang. Color image segmentation based on mean shift and normalized cuts. *IEEE Trans. on Systems, Man, and Cybernetics, Part B*, 37(5):1382–1389, 2007.
14. Speech University of Eastern Finland and Image Processing Unit. Clustering dataset, http://cs.joensuu.fi/sipu/datasets/.
15. S. Zhang, Z. Qin, C.X. Ling, and S. Sheng. Missing is useful": missing values in cost-sensitive decision trees. *IEEE Trans. on Knowledge and Data Engineering*, 17(12):1689–1693, 2005.
16. Shichao Zhang. Shell-neighbor method and its application in missing data imputation. *Applied Intelligence*, 35(1):123–133, 2011.