A Distance Function for Data with Missing Values and its Application

Loai AbdAllah and Ilan Shimshoni

Abstract—Missing values in data are common in real world applications. Since the performance of many data mining algorithms depend critically on it being given a good metric over the input space, we decided in this paper to define a distance function for unlabeled datasets with missing values. We use the Bhattacharyya distance, which measures the similarity of two probability distributions, to define our new distance function. According to this distance, the distance between two points without missing attributes values is simply the Mahalanobis distance. When on the other hand there is a missing value of one of the coordinates, the distance is computed according to the distribution of the missing coordinate. Our distance is general and can be used as part of any algorithm that computes the distance between data points. Because its performance depends strongly on the chosen distance measure, we opted for the k nearest neighbor classifier to evaluate its ability to accurately reflect object similarity. We experimented on standard numerical datasets from the UCI repository from different fields. On these datasets we simulated missing values and compared the performance of the kNN classifier using our distance to other three basic methods. Our experiments show that kNN using our distance function outperforms the kNN using other methods. Moreover, the runtime performance of our method is only slightly higher than the other methods.

Keywords—Missing values, Distance metric, Bhattacharyya distance.

I. Introduction

Missing values in data are common in many real world datasets. There are many serious data quality problems in real datasets such as: incomplete, redundant, inconsistent and noisy data. Missing values can be caused by human error, equipment failure, system generated errors, and so on. Missing values in a dataset are common in real world applications. According to the study of Cabena [3], about 20% of the effort is spent on the problem and data understanding, about 60% on data preparation and about 20% on data mining and analysis of knowledge.

Since many algorithms in data mining require a distance function as a basic component, our goal in this paper is to define a new unsupervised distance function, that can deal with unlabeled datasets with missing values. In this paper we use the Bhattacharyya distance in order to define the new distance function. Bhattacharyya defined a distance to measure the similarity of two probability distributions. In the simple case when we measure the distance between two points with no missing values, our distance is simply the Mahalanobis distance. When on the other hand there is a missing value of one of the coordinates, we developed a method to compute the distance according to the distribution of the missing coordinate using the Bhattacharyya distance. We estimate the result of randomly choosing a value from the coordinate's distribution and computing its Bhattacharyya distance.

Our metric is general and can be used with any approach which uses a distance metric. But because its performance depends strongly on the chosen distance measure, we opted for the k nearest neighbor classifier [5] to evaluate its ability to accurately reflect object similarity.

The paper is organized as follows. Previous methods which deal with missing values are reviewed in Section II. The distance function using Bhattacharyya distance is described in Section III. Experimental results on numerical datasets is presented in Section IV. Finally, our conclusions are presented in Section V.

II. Related work

Several methods have been proposed to deal with missing data. These methods can be classified into three basic categories: (a) Case deletion; (b) Learning without handling missing data; (c) Missing data imputation. Some of these methods deal with labeled data while others deal with unlabeled data. Our approach which deals with unlabeled data, does not fall into any of those categories. Instead, we define a distance metric for data with missing attributes using the Bhattacharyya distance, which measures the similarity of two probability distributions. When a value is missing in one point, we compute the similarity of the distribution of coordinate with the measured value of the second point. In this section we will review some popular methods that deal with missing values.

The most common method is the **Case Deletion** method, that 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 completely at random.

Another simple and common method is the **Most Common Attribute Value** method. The value of the attribute that occurs most often is selected to be the value for all the unknown values of the attribute. The CN2 algorithm [4] uses this idea. One main drawback in this method is, that it does not pay any attention to the relationship between attributes. A variation of this method which deals with labeled data, is a restriction of the first method to the concept, i.e., to all examples within the same value of the class as an example with a missing attribute value [2]. This time the value of the attribute, which occurs the most common within the same class is selected to be the

L. AbdAllah is with the Department of Mathematics, University of Haifa, Israel and with the Department of Mathematics and Computer Science, The College of Saknin, e-mail: Loai1984@gmail.com

I. Shimshoni is with the Department of Information Systems, University of Haifa, Israel, e-mail: ishimshoni@mis.haifa.ac.il

value for all the unknown values of the attribute. This method is also called the maximum relative frequency method, or the maximum conditional probability method (given concept) [7].

The main idea of the **Mean/Mode Imputation** method is to replace a data point with missing values with the mean/mode of all the instances in the data. But, using a fixed instance to replace all the instances with missing values will change the characteristic of the original dataset; ignoring the relationship among attributes will bias the performance of subsequent data mining algorithms. A variation of this method is to replace the missing data for a given attribute by the mean or mode of all known values of that attribute in the class where the instance with missing data belongs [8].

The **k-Nearest Neighbor Imputation** method uses the kNN algorithm (using only the known values) to estimate and replace the missing data [10], [1]. Efficiency is the biggest trouble for this method. While the k-NN algorithm look for the most similar instances, the whole dataset, which is usually quite huge, has to be searched. On the other hand, the selected value of k and the measure of similarity will impact the results greatly.

The **k-means Imputation** method for predicting missing attribute values using simple k-means clustering. The missing attributes are assigned with one possible value each time and the dataset is clustered using k-means to check whether the instance is clustered in the correct class. If so then the assigned value is made as permanent. Otherwise the clustering is performed with the next possible value [9].

III. Distance Function using Bhattacharyya Distance

A. Background

A. Bhattacharyya was a statistician who worked in the 1930s at the Indian Statistical Institute. He defined a distance to measure the similarity of two probability distributions. We use this metric in order to define a metric between two samples with missing attribute values. So first we will review the Bhattacharyya distance, and then we will describe how we use it within our distance function.

Consider two univariate probability density functions, f_1, f_2 in the same domain. The Bhattacharyya distance is defined as

$$D_B(f_1, f_2) = -\ln\left(BC(f_1, f_2)\right)$$

where BC is the Bhattacharyya coefficient, which is a measure of the amount of overlap between two statistical samples or populations. For discrete probability distributions the Bhattacharyya coefficient will be:

$$BC(f_1, f_2) = \sum_{x \in X} \sqrt{f_1(x) \cdot f_2(x)},$$

and

$$BC(f_1, f_2) = \int \sqrt{f_1(x) f_2(x)} \mathrm{d}x,$$

for continuous distributions.

We will now consider the special case of Gaussian distributions. Let $f_1(x), f_2(x)$ be two univariate Gaussian probability density functions where $\mu_1 \neq \mu_2$ and $\sigma_1 \neq \sigma_2$ and:

$$f_1(x) = \mathcal{N}(\mu_1, \sigma_1^2)$$

$$f_2(x) = \mathcal{N}(\mu_2, \sigma_2^2)$$

The Bhattacharyya coefficient is defined as:

$$BC(f_1, f_2) = \int \sqrt{f_1(x) f_2(x)} \mathrm{d}x, = \sqrt{\frac{2\sigma_1\sigma_2}{(\sigma_1^2 + \sigma_2^2)}} \exp\left\{\frac{-(\mu_1 - \mu_2)^2}{4(\sigma_1^2 + \sigma_2^2)}\right\}.$$

Therefore the Bhattacharyya distance D_B is:

$$D_B(f_1(x), f_2(x)) = -\ln\left(BC(f_1(x), f_2(x))\right) = -\ln\left(\sqrt{\frac{2\sigma_1\sigma_2}{(\sigma_1^2 + \sigma_2^2)}} \exp\left\{\frac{-(\mu_1 - \mu_2)^2}{4(\sigma_1^2 + \sigma_2^2)}\right\}\right) = \frac{1}{2}\ln\left(\frac{\sigma_1^2 + \sigma_2^2}{2\sigma_1\sigma_2}\right) + \frac{1}{4}\frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2}.$$

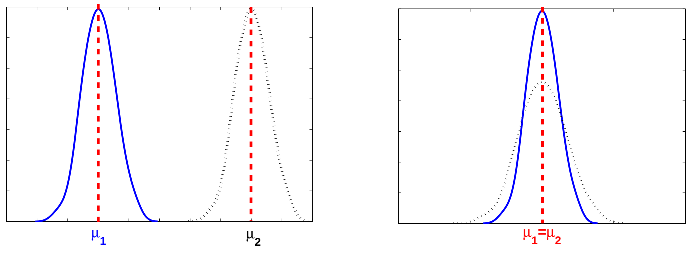
For multivariate normal distributions $f_i = N(\mu_i, \Sigma_i)$ the Bhattacharyya distance will be:

$$D_B = \frac{1}{2}\ln\left(\frac{\det\Sigma}{\sqrt{\det\Sigma_1 \det\Sigma_2}}\right) + \frac{1}{8}(\mu_1 - \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2),$$

where μ_i and Σ_i are the means and covariance of the distributions, and

$$\Sigma = \frac{\Sigma_1 + \Sigma_2}{2}.$$

From these equations can be concluded that the Bhattacharyya distance is a generalization of the Mahalanobis distance. When the variances of the two distributions are the same the first term of the distance is zero as this term depends solely on the variances of the distributions, and the distance will be the Mahalanobis distance between two means μ_1, μ_2. But, on the other hand, if the means are equal and the variances are different the Mahalanobis distance will be zero, in contrast to the Bhattacharyya distance which takes into account the differences between the variances (as shown in Figure 1).



(a) (b)

Fig. 1. Bhattacharyya distance for two special cases: (a) The variances are the same ($\sigma_1 = \sigma_2$), means are different ($\mu_1 \neq \mu_2$). (b) The means are equal ($\mu_1 = \mu_2$), variances are different ($\sigma_1 \neq \sigma_2$)

B. Our distance measure

We now turn to define our distance metric. Let A be a set of points, where each coordinate is measured by a different sensor. Given a measured value x_i for the ith coordinate i c_i, the conditional probability for c_i will be $P(c_i|x_i) \sim \mathcal{N}(x_i, \sigma_i^2)$, where x_i is the mean and σ_i^2 is the variance of the sensor which measured the coordinate c_i. When on the other hand the value of x_i is missing then the probability distribution for c_i might be given in advance or can be computed according to the known values for this coordinate from the data (i.e., $P(c_i) \sim \chi_i$), where χ is the distribution. In our derivation when the distribution is unknown we estimate it using the kernel density estimation method (KDE) from the measured values.

Note that since each specific coordinate is measured by the same sensor and under the same conditions, each coordinate has a specific variance σ_i^2. Our method can be generalized to deal with coordinates whose measurements are dependant, but for simplicity we assume that the coordinates measurements are independent. Under these assumptions we will treat each coordinate separately.

Given two sample points X and Y, 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.

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

$$D_B(x_i, y_i) = DB(N(x_i, \sigma_{i1}^2), N(y_i, \sigma_{i2}^2)) = \frac{1}{2}\ln\left(\frac{\sigma_{i_1}^2 + \sigma_{i_2}^2}{2\sigma_{i_1}\sigma_{i_2}}\right) + \frac{1}{4}\frac{(x_i - y_i)^2}{\sigma_{i_1}^2 + \sigma_{i_2}^2}.$$

Since x_i and y_i were measured by the same sensor $\sigma_{i_1} = \sigma_{i_2} = \sigma_i$ and thus

$$D_B(x_i, y_i) = \frac{1}{8}\frac{(x_i - y_i)^2}{\sigma_i^2}. \tag{1}$$

As mentioned above, this is the Mahalanobis distance which is the standard distance measurement between two points. I this case, the runtime complexity is $O(1)$.

2) One value is missing: Suppose that x_i is missing an the value y_i is given. Since the value of x_i is unknown, we ca not compute its Bhattacharyya distance. Instead we model th distance as a random selection of a point from the distributio of its coordinate χ_i and compute its distance. The mean o this computation is our distance. We will estimate this valu as follows: We divide the range of c_i $[\min(c_i), \max(c_i)]$ int $l-1$ equal intervals $(m_1, \ldots, m_l)$ as illustrated in Figure 2.

For each value m_j we can estimate its probability density $p(m_j)$ using the KDE. The probability for the jth interval Δj is:

$$P(\Delta j) = p(m_j) \cdot \frac{\max(c_i) - \min(c_i)}{l-1}.$$

As a result, we approximate the Mean Bhattacharyya distance (MD_B) between y_i and the distribution as:

$$MD_B(\chi_i, y_i) = \sum_{j=1}^{l-1} P(\Delta_j) D_B(\mathcal{N}(m_j, \sigma_1), \mathcal{N}(y_i, \sigma_1)).$$

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.

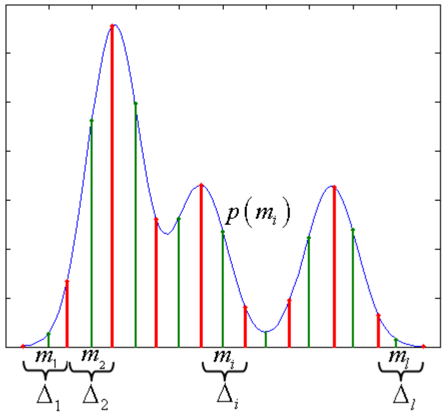
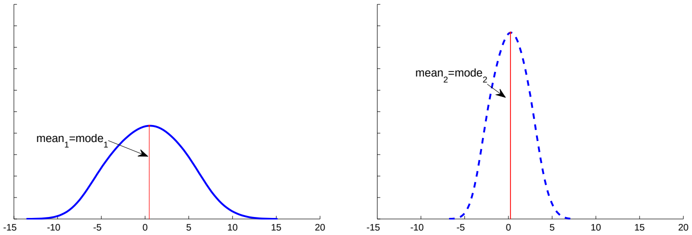


Fig. 2. An example for the normal kernel density estimation results for coordinate c_i. m_j denotes the selected points and $p(m_j)$ denotes the probability density for m_j.

This is in contrast to the **Most Common Attribute Value** method. There the value of the attribute that occurs most often is selected to be the value for all the unknown values of the attribute and imply that the probability of the most common attribute value is 1 and 0 for all other possible values. Furthermore our distance is different from the **Mean Attribute Value method**, where the mean of a specific attribute is selected to replace the unknown values of the attribute because it does not take into account the dispersion of the values in the distribution. Thus for example two distributions with the same mean and different variances (as can be seen in Figure 3) will get the same distance whereas in our method the distance increases as a function of the variance.



(a) (b)

Fig. 3. (a) and (b) show two distributions with the same mean and different variances. The distance computed for these two distributions is different.

Figure 4 illustrates the dependance of our distance on the variance of distribution χ_i. When the variance is close to the measurement variance σ_i^2 the distance will converge to the value achieved for a measured value. As the variance increases the distance increases until it converges to the distance achieved for the uniform distribution.

In this case (i.e., one value is missing), the runtime of our method is $O(l)$, because according to this metric the algorithm has to compute $l-1$ Bhattacharyya distances. On the other hand as l increases so does the accuracy of the distance

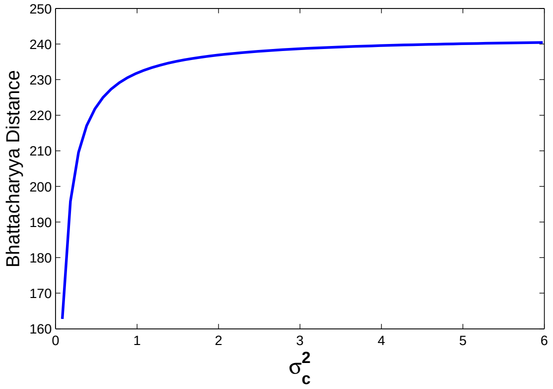


Fig. 4. The distance between a measured point and an unknown value with different values of the variance σ_c^2 of the distribution χ_i.

estimate. There is therefore a trade off between the accuracy of the estimate and the the complexity of the algorithm. From our experiments we did not find a significant change in the performance of the classification algorithms as a function of l.

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

$$MD_B = \sum_{q=1}^{l-1}\sum_{j=1}^{l-1} P(\Delta_{1q})P(\Delta_{2j})DB(\mathcal{N}(m_{1q},\sigma_i),\mathcal{N}(m_{2j},\sigma_i)).$$

Consider again the examples in Figure 3. The MD_B of the first distribution with the larger variance will naturally be larger than the MD_B of the second distribution with the smaller variance. Figure 5 shows the dependance of MD_B on the variance σ_c^2 of the distribution χ_i. As the distribution is more dispersed, the value of the MD_B increases. In this example the distributions χ_i were Gaussian but the relationship is general.

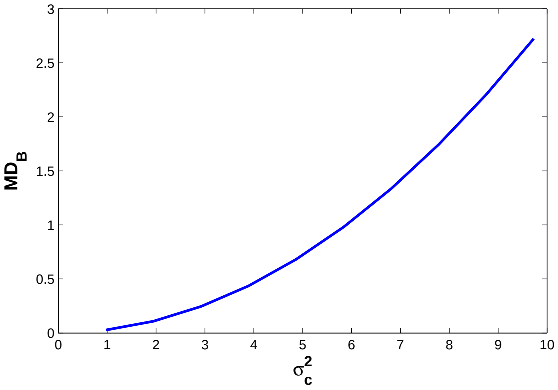


Fig. 5. The value of MD_B as a function of the variance σ_c^2 of the distribution χ_i.

As in this case no value has to be known in order to compute the MD_B the distance between two missing values from a specific coordinate will be fixed, and has to be computed only once. It therefore does not have any effect on the runtime of the algorithm.

IV. Experiments on Numerical Datasets

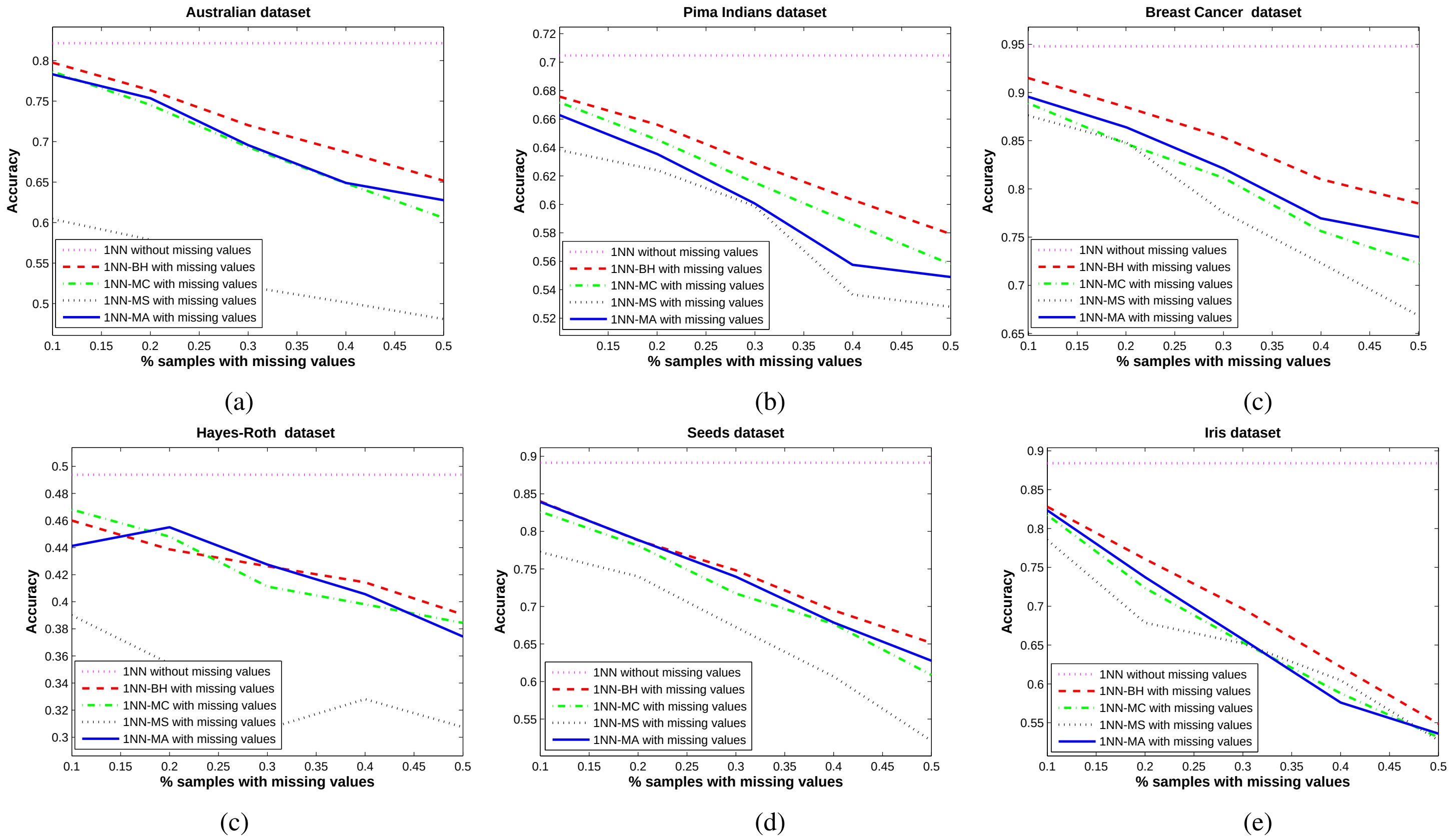
In order to measure the ability of the new distance function to reflect the actual similarity or dissimilarity between instances with missing values we compare the performance of the kNN ($k = 1$) classifier on complete data (i.e., without missing values) to the performance of the kNN classifier using our distance (KNN-BH), the kNN-MC (i.e., Most Common attribute value), the kNN-MA (i.e., the Mean value of each Attribute), and the kNN-MI(Mean Imputation) that replaces a data point with missing values with the mean of all the instances in the data, on the same datasets with missing values. All the algorithms were implemented in Matlab.

We ran our experiments on six standard numerical datasets from the Machine Learning Repository (UCI) [6]from different fields: the Australian credit approval dataset, the Pima Indians diabetes dataset, the Breast Cancer dataset, the Hayes Roth dataset, the Seeds dataset and the Iris dataset. The first three dataset were a two-class classification problem, while the last three datasets were a three-class classification problem. The Australian dataset contains 690 instances. The Pima Indians diabetes dataset contains 762 instances. The Breast Cancer dataset contains 683 instances divided into two classes. The Hayes Roth dataset contains 160 instances. The Seeds dataset dataset contains 210 instances, and the Iris dataset contains 150. The characteristics of all the datasets can be seen in Table I. All these datasets were labeled, but this knowledge was used only to evaluate the accuracy of the resulting classifier. In all experiments these datasets are assumed to be unlabeled.

TABLE I
Dataset properties

Dataset	Dataset size	Classes
Australian	690 × 14	2
Pima Indians	762 × 8	2
Breast Cancer	683 × 8	2
Hayes Roth	160 × 5	3
Seeds	210 × 7	3
Iris	150 × 4	3

In the first stage of the experiments, from each dataset a set of size 10%-50% of the dataset is randomly drawn to be samples with missing values, where at least one coordinate from each instance was selected randomly to be the missing value. After that, from each dataset a set of 10% of the dataset was drawn randomly to be the training dataset (i.e., labeled) and the rest is the testing dataset. (Note that the training dataset may contains instances with missing values.) Then the accuracy was evaluated for each set of missing values by the ability of the kNN classifier to label the data. The results are averaged over 10 different runs on each dataset. A resulting curve was constructed for each dataset to evaluate how well the algorithm performed.



(a) (b) (c)

(c) (d) (e)

Fig. 6. Results of 1NN without missing values, 1NN-BH, 1NN-MC, 1NN-MS and 1NN-MA algorithms over six numerical datasets with missing values.

A. Results

As can be seen from Figure 6, the kNN-BH was superior and outperforms the other algorithms. The learning curves are constructed by computing the ratio of correctly classified instances to the whole unlabeled data.

For the Australian, Pima Indians, Breast Cancer and Iris datasets, the curves show that the kNN-BH obviously outperforms the other methods, while for the two rest datasets the benefit of the kNN-BH algorithm appears when the percent of the missing values becomes large as can be seen in Figures 6(e&(f)). This improvement in kNN-BH accuracy is due to the ability of the proposed metric which uses the Bhattacharyya distance to better measure the actual similarity between the objects with missing values. Moreover, according to the results curves the performances of the kNN-MC and kNN-MA were comparable, while the performance of the kNN-MS was poorly.

V. 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 unsupervised distance learning metric based on data attributes distribution using the Bhattacharyya distance and used it within the kNN classifier framework. According to this distance, the distance between two points without missing attributes values is simply the Mahalanobis distance. When on the other hand there is a missing value of one of the coordinates, the distance is computed according to the distribution of the missing coordinate.

In contrast to many approaches in this field, our method does not require any knowledge about the classes and the algorithm is applied to unsupervised datasets.

In our experiments we used the one nearest neighbor classifier to measure the ability of our metric to reflect the actual similarity between objects with missing values. We compared the performance of the kNN method using our metric with three basic methods. From the experiment we conclude that our distance is a more appropriate function to measure the similarity between objects with missing value especially when the percent of the missing values is becomes large. This is because when the missing data is small, the missing value does not influence the similarity value significantly.

This proposed distance is general and can be used as part of any algorithm that computes the distance between data points. Moreover, this distance can be used for different datasets in different application areas.

Acknowledgment

This research was supported by the 450mm metro consortium of the ministry of industry and commerce. Moreover it was supported by a graduate student fellowship from the ministry of science and technology.

REFERENCES

[1] Gustavo Batista and Maria Carolina Monard. An analysis of four missing data treatment methods for supervised learning. *Applied Artificial Intelligence*, 17(5-6):519–533, 2003.

[2] Knonenko. Bratko and E. I. Roskar. Experiments in automatic learning of medical diagnostic rules. *Technical Report, Jozef Stefan Institute, Lljubljana, Yugoslavia*, 1984.

[3] Krzysztof J Cios and Lukasz A Kurgan. Trends in data mining and knowledge discovery. *Advanced techniques in knowledge discovery and data mining*, pages 1–26, 2005.

[4] Peter Clark and Tim Niblett. The cn2 induction algorithm. *Machine learning*, 3(4):261–283, 1989.

[5] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27, 1967.

[6] A. Frank and A. Asuncion. UCI machine learning repository, 2010.

[7] 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.

[8] 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.

[9] Nambiraj Suguna and Keppana G Thanushkodi. Predicting missing attribute values using k-means clustering. *Journal of Computer Science*, 7(2):216–224, 2011.

[10] Shichao Zhang. Shell-neighbor method and its application in missing data imputation. *Applied Intelligence*, 35(1):123–133, 2011.

Loai AbdAllah received his B.Sc. in Mathematics and Management Information Systems from the University of Haifa, his M.Sc. in Mathematics from the University of Haifa, where he is currently working toward the Ph.D. degree in Mathematics in the University of Haifa. Loai was a member of the Departments of Mathematics and Computer Science at the College of Sakhnin from in October 2011. His current research interest is in data mining.

Ilan Shimshoni Ilan Shimshoni recieved his B.Sc. in mathematics from the Hebrew University in Jerusalem, his M.Sc. in computer science from the Weizmann Institute of Science, and his Ph.D. in computer science from the University of Illinois at Urbana Champaign (UIUC). Ilan was a postdoctorate fellow at the faculty of computer science at the Technion, from 1995-1998, and was a member of the faculty of industrial engineering and management from 1998-2005. He joined the department of Information Systems (IS) at Haifa University in October 2005.