DUST: A Generalized Notion of Similarity between Uncertain Time Series

Smruti R. Sarangi
IBM Research - India
Bangalore, India
srsarangi@in.ibm.com

Karin Murthy
IBM Research - India
Bangalore, India
karin.murthy@in.ibm.com

ABSTRACT

Large-scale sensor deployments and an increased use of privacy-preserving transformations have led to an increasing interest in mining uncertain time series data. Traditional distance measures such as Euclidean distance or dynamic time warping are not always effective for analyzing uncertain time series data. Recently, some measures have been proposed to account for uncertainty in time series data. However, we show in this paper that their applicability is limited. In specific, these approaches do not provide an intuitive way to compare two uncertain time series and do not easily accommodate multiple error functions.

In this paper, we provide a theoretical framework that generalizes the notion of similarity between uncertain time series. Secondly, we propose $DUST$, a novel distance measure that accommodates uncertainty and degenerates to the Euclidean distance when the distance is large compared to the error. We provide an extensive experimental validation of our approach for the following applications: classification, top-k motif search, and top-k nearest-neighbor queries.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data Mining*

General Terms

Algorithms, Experimentation

Keywords

Time Series, Uncertain Data, Similarity, Distance Measure, Data Mining

1. INTRODUCTION

Distance measures used for similarity search and data mining are often focused towards data without uncertainty. However, recently there has been a move to acknowledge that in many application domains, data is uncertain and the uncertainty has to be captured and accounted for. There is a large body of research on managing, modeling, querying, and mining uncertain data (see [18, 2] for recent surveys on the topic). However, not many approaches deal with time series or streaming data.

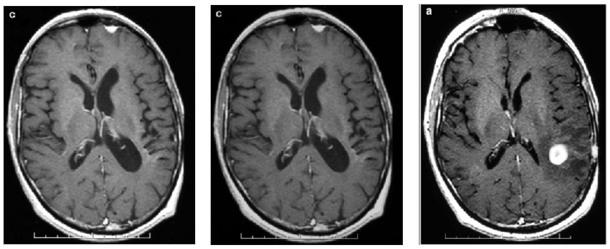
(a) Brain image (b) Brain image from (a) slightly blurred (c) Brain image with tumor

Figure 1: Three brain tumor images (MRI Scan)

There are two main reasons why time series data may be uncertain. First, physical data collection methods are imperfect. For example, the accuracy of a wireless sensor is associated with a certain error distribution. Second, to preserve privacy a certain degree of uncertainty is sometimes intentionally introduced into a time series. For example, privacy-preserving methods may aggregate or perturb time series data.

Traditional distance measures such as the Euclidean distance or dynamic time warping do not always work well for uncertain time series data. Section 4 will validate this for a large variety of data sets and different kinds of uncertainty. Here we only show an illustrative example. We extracted time series data from brain scan images taken from [16]. Figure 1 shows three brain scan images: the first image shows a normal brain, the second image shows a slightly blurred version of the first image, and the third one shows an image of the brain when it had a tumor. We extracted a time series of length 350 for each image by cutting through the image at the height of the tumor shown in Figure 1(c) and extracting the gray scale value for each pixel.

According to Euclidean distance, the time series generated from the image in Figure 1(a) is 5% closer to the time series generated for Figure 1(c) than to the time series generated for Figure 1(b). However, intuitively we expect the image in Figure 1(a) to be more similar to its slightly blurred version in Figure 1(b) than to the version in Figure 1(c) which contains the tumor. In this example, Euclidean distance and

KDD'10, July 25–28, 2010, Washington, DC, USA.

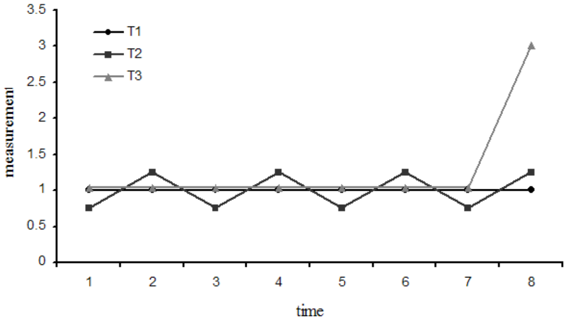

Figure 2: Three 8-dimensional time series

most other traditional distance measures produce an unintuitive result, a problem which the new distance measure proposed in this paper addresses.

For time series data the overall uncertainty arises from the uncertainty at each time stamp. Thus, even though the uncertainty for each individual value of a time series may be very small, the uncertainty compounds with the number of elements in a time series. Figure 2 shows three 8-dimensional time series T_1, T_2, and T_3. Assume for now that the values in T_1 and T_3 are values without uncertainty whereas the values in T_2 are uncertain and affected by a normally distributed error function which is zero beyond three standard deviations (3σ). The distances between values in T_1 and T_2 are all small and bounded by 3σ whereas there is one large distance beyond 3σ between values in T_1 and T_3. The Euclidean distance between T_1 and T_2 is the same as the Euclidean distance between T_1 and T_3. However, many application domains would like to consider T_1 and T_2 as more similar than T_1 and T_3. The probability that T_1 and T_3 are the same is effectively zero whereas there is some likelihood that T_1 and T_2 would have been the same if the sensor producing T_2 would have been faultless.

In this paper, we present a new distance measure, $DUST$, that allows us to compute distances between uncertain time series in an intuitive fashion. We extend the notion of similarity between time series proposed by prior work [21, 3] such that if two sets of sensor readings have a chance of being equal, the distance between them is lower as compared to the case in which the two sets can never be the same. We prove that $DUST$ obeys most of the properties of an ideal distance measure. Furthermore, we observe that when the error is very small compared to the separation between points belonging to the two time series, $DUST$ converges with traditional Euclidean distance.

Prior work has mostly considered error functions that follow a Normal distribution. However, [5, 20] have observed non-Gaussian error pdfs in actual sensor deployments. The $DUST$ distance measure can seamlessly handle such error distributions. In Section 3.5 we propose a method to compute the $DUST$ distance when the individual error distributions for the different time series elements are different and possibly non-Gaussian. Subsequently, we provide a method to efficiently compute the $DUST$ distance between two time series.

In Section 4 we extensively evaluate the $DUST$ distance on the UCR datasets [12]. We perform three experiments: classification, top-k motif search, and top-k nearest neighbor queries. The $DUST$ distance outperforms Euclidean distance and dynamic time warping. It increases the classification accuracy by about 10%, and is able to substantially mitigate the effect of sensor error. It is also far more resilient to error for motif and nearest-neighbor detection as compared to Euclidean distance.

We present related work in Section 2, the theory and implementation of the $DUST$ distance in Section 3, an experimental evaluation of $DUST$ in Section 4, and we conclude the paper in Section 5.

2. RELATED WORK

2.1 Similarity of Uncertain Time Series

There has been a considerable amount of work on representing and querying uncertain data. However, to the best of our knowledge there are few papers that address querying and mining of uncertain time series data.

In a 2008 paper Charu Aggarwal and Philip Yu presented a framework for clustering uncertain data streams [1]. They assume that some statistics are known about the uncertainty. Based on this they create micro-clusters, and dynamically update them as new data points arrive based on an expected value of similarity. This approach does not use a distance measure, and is thus not applicable to general data mining tasks.

In 2009, two independent papers [21, 3] introduced the notion of a a probabilistic bounded range query (PBRQ) for time series data. Given a distance bound ϵ and a probability threshold τ, two time series are considered to be similar if the probability that the distance between them is equal or less than ϵ, is equal or greater than τ.

$$PBRQ_{\epsilon,\tau}(T, DB) = \{T' \in DB | Pr(DIST(T, T') \leq \epsilon) \geq \tau\}$$

However, the two approaches differ in their definition of the distance function $DIST$ used to compare two uncertain time series.

Johannes Aßfalg and others [3] assume that the uncertainty of a time series is represented by a set of sample observations at each time slot. Thus, an uncertain time series T represents a set of regular time series S(T) where each regular time series is constructed by picking one sample point for each time slot. The distance between two uncertain time series T_1 and T_2 is defined as the set of distances between all combinations from $S(T_1)$ and $S(T_2)$. First, not all application domains provide multiple sample points for each time slot, and second, this approach is not computationally efficient. In our approach, $DUST$, we only deal with closed form formulae and lookup tables.

Mi-Yen Yeh and others [21] present their scheme PROUD to handle uncertainty for data streams. The uncertainty at each time point is modeled as a continuous random variable for which only the mean and standard deviation are known. The distance between two time series is a random variable. This is sufficient for computing the result of a probabilistic bounded range query but again it does not allow us to directly compute the distance between two time series. Another limitation of PROUD is that in order to make the computation of a PBRQ more efficient and to allow early pruning of candidates, PROUD assumes that the uncertain deviation is the same for all time points of a series. We consider this a limitation, which our scheme does not have.

2.2 Sensor Error Characterization

We typically see different kinds of faults in sensor datasets. [19] distinguishes between single-sample spikes, longer duration noisy readings, and anomalous constant offset readings. Those faults may be detected by cleaning approaches that take into account dependencies between readings at different time points (see for example, [10]). The distance measure we propose does not cover such cases and is limited to errors that occur independent of events at other time points. Also, we assume that the error is due to the inherent imprecision of a sensor. To detect random effects of external sources more sophisticated cleaning approaches are necessary.

3. DISTANCE BETWEEN UNCERTAIN TIME SERIES

Let $T_1[1...n]$ and $T_2[1...n]$ be two time series. Throughout the paper we denote the distance between two time series T_1 and T_2 by upper-case letters (for example, $DIST(T_1, T_2)$). We denote the distance between two time series values with lower-case letters (for example, $dist(T_1[i], T_2[i])$. Also, the lower case letter p denotes the probability distribution function (pdf), and the upper case letter P refers to the probability. We first review approaches to measure the distance for time series without uncertainty and then extend the results to uncertain times series.

3.1 Time Series Without Uncertainty

Several approaches have been proposed for the case where there is no uncertainty. Ding and others provide a survey of most of the existing approaches in [8].

Two of the most common approaches are Euclidean Distance (EUCL) [9] and Dynamic Time Warping (DTW) [4]. The Euclidean distance between two time series is defined as:

$$EUCL(T_1, T_2) = \sqrt{\Sigma_{i=1}^{n}(T_1[i] - T_2[i])^2} \tag{1}$$

The Dynamic Time Warping (DTW) distance is defined as:

$$\begin{aligned} DTW(i, j) =& \mathcal{D}(T_1[i], T_2[j]) + min(DTW(i-1, j-1), \\ & DTW(i, j-1), DTW(i-1, j)) \end{aligned} \tag{2}$$

where $DTW(i, j)$ is short for $DTW(T_1[1 \dots i], T_2[1 \dots j])$.

EUCL is called a *lockstep* measure because it computes the distance between corresponding elements in both the time series, whereas DTW is called an *elastic* measure. Along with these two common distances, the survey in [8] also describes other distance measures such as Longest Common Subsequence, Edit Distance with Real Penalty, Edit Distance on Real Sequence, DISSIM, and Sequence Weighted Model. Discussing these distance measures is beyond the scope of this paper.

We emphasize two key findings from [8].

1. For small dimensional data sets DTW is superior to EUCL. Most of the sophisticated elastic measures have an accuracy similar to DTW.

2. For large dimensional data sets the performance of EUCL is similar to DTW.

Hence, we only consider DTW and EUCL in our paper.

3.2 Generalized Distance Between Uncertain Time Series

3.2.1 Desired Properties of a Distance Measure

We first describe some desirable properties for a distance measure. Ideally, a distance measure should be metric and fulfill the following conditions:

1. Non-negativity : $d(A, B) \geq 0$.

2. Identity of indiscernibles : $d(A, B) = 0$ iff $A = B$

3. Symmetry : $d(A, B) = d(B, A)$

4. Triangle Inequality : $d(A, B) + d(A, C) \geq d(B, C)$

In the case of a distance measure for uncertain time series data, the distance measure should also obey the following additional property.

5. The distance should be similar to EUCL or DTW if the magnitude of the error is very small.

Time series with a very small error as compared to distances between data values, become very similar to time series without uncertainty. Thus, with decreasing error, the distance measure for uncertain time series should asymptotically converge with the distance measures for time series without uncertainty.

3.2.2 Generalized Distance Measure

In this subsection we generalize the notion of distance between uncertain time series from definitions that have been used in other papers [21, 3, 2]. Consider two time series T_1 and T_2. Let $T[i]$ refer to the i^{th} element in a time series T. Each element x in a time series is an uncertain value and can be represented as $x = r(x) + \mathcal{E}(x)$. Here $r(x)$ is the real value and $\mathcal{E}(x)$ represents the error. Like [21, 3], we assume that all the error distributions for elements in a time series are independent. According to [21, 3, 2] two time series are considered similar if

$$P(DIST(T_1, T_2) \leq \epsilon) \geq \tau$$

where ϵ is a very small number and τ is relatively close to 1. As discussed in Section 2 the distance function $DIST$ varies for different approaches. Again, note that the above notion does not provide an absolute number for the similarity of two time series. Johannes and others [3] further state that T_1 and T_2 are closer than T_1 and T_3 if $P(DIST(T_1, T_2) \leq \epsilon) > P(DIST(T_1, T_3) \leq \epsilon)$.

Let $DIST(T_1, T_2)$ be denoted by the random variable X. For sufficiently small values of ϵ, $P(X \leq \epsilon) = p(X = 0)\ \epsilon$. To eliminate ϵ, we assume that even for large ϵ the distance between two uncertain values is only a function of $p(X = 0)$. We use this assumption to build a new distance *dust* for computing the distance between two uncertain values. We show in Section 4 that making this assumption produces good results for a wide variety of data sets and data mining tasks.

We observe:

$$\begin{aligned} & P(DIST(T_1,T_2) \le \epsilon) > P(DIST(T_1,T_3) \le \epsilon) \\ \approx\ & p(DIST(T_1,T_2) = 0) > p(DIST(T_1,T_3) = 0) \\ \Longleftrightarrow & \Pi_i p(dist(T_1[i],T_2[i]) = 0) > \\ & \Pi_i p(dist(T_1[i],T_3[i]) = 0) \\ \Longleftrightarrow & \Sigma_i - log(p(dist(T_1[i],T_2[i]) = 0)) \le \\ & \Sigma_i - log(p(dist(T_1[i],T_3[i]) = 0)) \end{aligned} \tag{3}$$

Intuitively, we want *dist* to measure the distance between two uncertain values x and y as the distance between the respective true values $r(x)$ and $r(y)$. Hence, we define *dist* as $dist(x,y) = Eucl(r(x), r(y))$.

Also, we experimentally observe that the distance between two uncertain values x and y is mostly dependent on $\Delta x = |x - y|$. When the underlying distribution of the time series values is uniform or Gaussian and the error function is uniform or Gaussian, this is exactly true. We prove this in the Appendix. Based on the observation that the distance mostly depends on Δx, we define a function $\phi(\Delta x) = p(dist(0, \Delta x) = 0)$, which is independent of x and y and only depends on their difference.

Based on the above observations, the *dust* distance is defined as follows:

$$\begin{aligned} dust(x,y) &= \sqrt{-log(\phi(|x-y|)) - \kappa} \\ \kappa &= -log(\phi(0)) \end{aligned} \tag{4}$$

The constant κ ensures that $dust(x,x) = 0$ for all x. We define the distance measure $DUST$ as

$$DUST(T_1,T_2) = \sqrt{\Sigma_1^n\ dust(T_1[i],T_2[i])^2} \tag{5}$$

Note that both the use of κ for *dust* and the definition of $DUST$ assume that the two compared sequences have the same number of elements and that two corresponding elements have the same error distribution.[1]

Using Equations 3, 4, and 5, we establish the following relationship between previously defined similarity measures and the $DUST$ distance measure:

$$\begin{aligned} & P(DIST(T_1,T_2) \le \epsilon) > P(DIST(T_1,T_3) \le \epsilon) \\ \Longleftrightarrow & \Sigma_1^n\ dust(T_1[i],T_2[i])^2 \le \Sigma_1^n\ dust(T_1[i],T_3[i])^2 \\ \Longleftrightarrow & DUST(T_1,T_2) \le DUST(T_1,T_3) \end{aligned}$$

Let us now see to what extent the *dust* distance and as such the $DUST$ distance obeys properties (1)-(5) introduced in Section 3.2.1. Since the probability of equality for two dissimilar elements is less than that of two similar elements, we have $dust(x,y) > dust(x,x)$. $dust(x,x)$ by definition is zero. This proves Property 1. We added the constant κ in Eqn 4 to ensure that $d(A,B) = 0$ if $A = B$ (first part of Property 2). We experimentally verified that the second part of Property 2 ($d(A,B) = 0 \Rightarrow A = B$) holds for most standard error distributions. Probabilities obey commutativity, thus Property 3 holds. We examine Property 4 (Triangle Inequality) in Section 3.3. In Section 3.4.2 we evaluate the *dust* distance for several common error functions. We observe that the *dust* distances converge to the Euclidean distance for small errors. This experimentally verifies Property 5.

3.3 Triangle Inequality

Let us assume a normally distributed error with mean 0 and standard deviation σ. Error functions are often modeled as normal distributions. However, in most practical situations the error lies between -3σ and 3σ and is unlikely to go beyond the 3σ range. Let us consider an example with three sensor readings x, y, and z, where $|x - y| = 2\sigma$ and $|y - z| = 2\sigma$. Now consider three time series $T_1 = xxxxx$, $T_2 = yyyyy$, $T_3 = xxxxz$. We have $EUCL(T_1,T_2) = 4.46\sigma$ and $EUCL(T_2,T_3) = 4.46\sigma$. By the triangle inequality we have $|x - z| \le 4\sigma$ and thus $EUCL(T_1,T_3) \le 4\sigma$. However, if the distance between x and z is 4σ, then the fact that T_1 and T_3 are closer than other pairs of distances, breaks our intuition. If we consider raw probability, then T_1 and T_3 can never be equal because x and z are 4σ apart. T_1 and T_2 can still be equal in a statistical sense.

To overcome this shortcoming of the Euclidean distance, the *dust* distance has to break the triangle inequality for small distances. Only then it can produce the intuitively correct result. Note however, that for larger *dust* distances the triangle inequality holds. Figure 3 illustrates an example where the triangle inequality is violated for small *dust* distances. For the example, the triangle inequality only holds if the separation between values is always greater than 4 σ. As the $DUST$ distance combines individual *dust* distances, the triangle inequality for $DUST$ may also be violated.

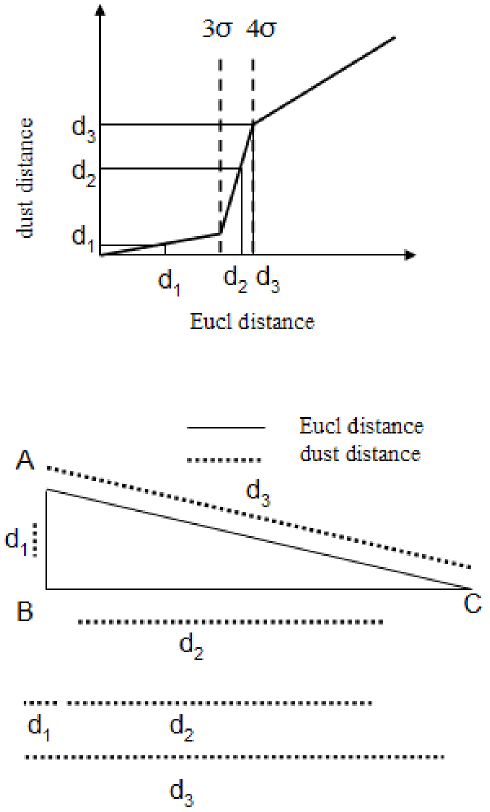

Figure 3: Violation of triangle inequality for small *dust* distances

[1] For other cases, we can use DTW in conjunction with a non-normalized *dust* distance without κ.

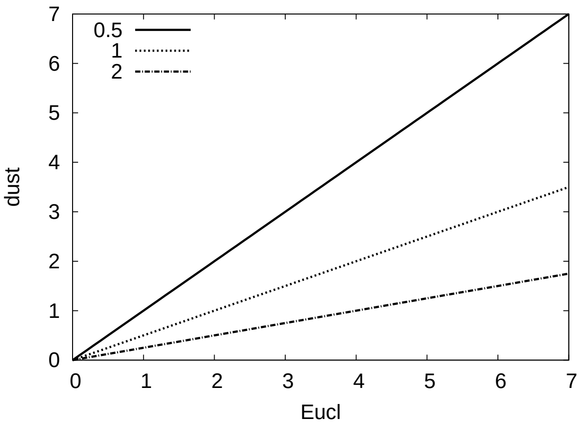

Figure 4: *dust* **distance for Gaussian error**

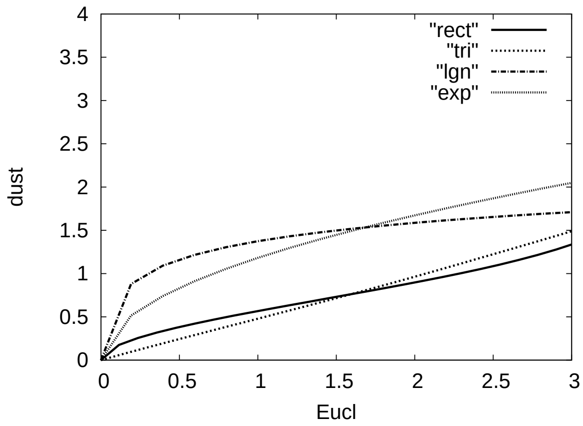

Figure 5: *dust* **distances for different error functions**

3.4 Computing the dust Distance

We now describe how the *dust* distance between two values x and y can be computed. As stated in Section 3.2.2 we need to compute $\phi(\Delta x) = p(dist(0, \Delta x) = 0)$. This is equivalent to computing $p(r(x) = r(y)|x, y)$.

$$p(r(x) = r(y)|x, y) = \int_z p(r(x) = z|x)p(r(y) = z|y)dz \quad (6)$$

We thus need to compute: $p(r(x) = z|x)$. By Bayes' Theorem this is equal to:

$$\begin{aligned} p(r(x) = z|x) &= \frac{p(x|r(x) = z)p(r(x) = z)}{p(x)} \\ &= \frac{p(x|r(x) = z)p(r(x) = z)}{\int_v p(x|r(x) = v)p(r(x) = v)dv} \end{aligned} \quad (7)$$

We need to compute the two probability densities $p(r(x) = v)$ and $p(x|r(x) = v)$. The former probability requires the distribution of the data we work with. The distribution is dependent on the process that is generating the time series. Keogh and others [13] observe that most time series in the UCR dataset follow a Gaussian distribution. Cho and others [6] observe that the underlying process follows a uniform distribution. $p(x|r(x) = v)$ is the pdf of the error function evaluated at $x - v$.

We can simplify the distance function $dust(x, x + \Delta x)$ to a function $f_{dust}(\Delta x)$, which maps the Euclidean distance to a distance computed according to Equations 4 and 6. In the next sections we evaluate f_{dust} for some of the most common error functions. For simplicity, we assume that the underlying distribution of the time series values is uniform.

3.4.1 Normal Distribution

In most cases, we assume that the error is distributed normally. The normal distribution is given as:

$$\mathcal{N}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

Here the mean is 0, and the standard deviation is σ. In the Appendix we prove that $f_{dust}(x) \propto x/\sigma$. We show the results for different standard deviations in Figure 4. We observe that for the case of normally distributed error there is **NO difference between Euclidean distance and** ***dust***. The distance is just scaled by a constant factor. If the error function is the same for all the elements in a time series, then using $DUST$ makes no difference. In this case, the use of a sophisticated distance measure that accommodates uncertainty is not necessary.

3.4.2 Other Distributions

Let us now take a look at some other distributions. We consider a rectangular distribution *rect*, where the error is uniform between $-\eta$ and η and a triangular distribution *tri*, where the error is 0 at $\pm\eta$ and peaks at 0. In addition to such standard distributions, there is a set of error distributions called heavy-tailed distributions [5], which are asymmetric and have a lot of large errors with a relatively higher probability than in the Normal distribution. We pick the log normal distribution (*lgn*) as an example for a heavy-tailed distribution:

$$lgn(x) = \frac{1}{x\sigma\sqrt{2\pi}} exp(-\frac{(ln(x) - \mu)^2}{2\sigma^2})$$

Finally, Sudano and others observe an asymmetric exponentially distributed error (*edf*) in their sensor system [20]. This error function is given by:

$$edf(x) = \lambda exp(-\lambda x)$$

We plot the four functions *rect*, *tri*, *lgn*, and *edf* in Figure 5. The variance for all the error functions is 1. We observe that the *tri* distribution is an approximate straight line. This is because its shape is roughly similar to that of a normal distribution between $\pm\eta$. The rectangular distribution has a slightly flatter error curve. However, we observe that the *dust* distances for *lgn* and *edf* are far higher than for *rect* and *tri* . The reason for this is their heavy tailed nature. They have higher probabilities for larger deviations. This reduces the probability of equality; consequently, the *dust* distance increases. Please note that after 0.2 all curves are roughly straight lines. That is after a distance of 0.2 the *dust* distance exhibits a similar behavior as the Euclidean distance.

3.4.3 *Why Use* DUST*?*

Given the above observation, a question arises about the applicability of $DUST$ and other uncertain time series mining techniques. We observe that if all the values have the same error distribution, then we are better off using Euclidean distance as it is computationally more efficient. However, $DUST$ is required in the case of multiple error distributions. $DUST$ gives a theoretically sound way of computing distances between two time series where individual time stamps may be associated with different error distributions.

In the case of sensor data not all sensors may be of the same type and sensors may be manufactured by different vendors. Hence, it is natural to have different error distributions. Several works in the broader engineering community [17, 15, 7] have mentioned this problem. For example, Ciarlini and others [7] observe that it is not possible to place sensors to monitor the materials in a cultural heritage site. It is too invasive. However, we can place a multitude of sensors in close proximity. Each one of them will have a different error distribution.

3.5 Combining Multiple Distributions

In this section, we show a way to combine different error distributions. In Figure 4 we observed that the *dust* distance function has different slopes for different error distributions. The difference in slope is acceptable if the Euclidean distance is of the same order as the error margin. However, as the separation of two points increases, the standard deviation of the error becomes increasingly irrelevant. Hence, at this point the *dust* distance should become the same for different error distributions.

Let us assume that we have a set of error distributions with the respective standard deviations $\sigma_1 \dots \sigma_n$. Let σ_e be a value significantly smaller than $\sigma_1 \dots \sigma_n$. We assume that for larger separations all the sensors approach this error value. We observe experimentally that the results are not very sensitive to the choice of σ_e (as long as σ_e is small enough). Let the original error distribution be $f(x)$, and let the adjusted error function be $f'(x)$. We have

$$f'(x) = \begin{cases} \eta_1 \leq x \leq \eta_2 & f(x) \\ x < \eta_1 & \mathcal{N}(0, \sigma_e) \\ x > \eta_2 & \mathcal{N}(0, \sigma_e) \end{cases} \qquad (8)$$

$\mathcal{N}(\mu, \sigma)$ is the Gaussian distribution. The constants η_1 and η_2 capture the fact that we are not interested in errors beyond the $[\eta_1, \eta_2]$ interval. For the case of a normal distribution with zero mean, $\eta_1 = -3\sigma$ and $\eta_2 = 3\sigma$. For the case of a triangular distribution (see Section 3.4.2) $\eta_1 = -\eta$ and $\eta_2 = \eta$. Figure 6 shows different *dust* distance curves for Normal distributions with standard deviations equal to 1, 1.5, 2 and 3. Here $\sigma_e = 1$. We see that all distributions have different slopes up till 6σ, then they merge with the line corresponding to $\sigma_e = 1$.

3.6 Calculating the DUST Distance Efficiently

As described in Section 3.4, we need to compute a function that maps the Euclidean distance between two values x and $x + \Delta x$ to the *dust* distance. We referred to this function as $f_{dust}(\Delta x)$. We compute a large number of sample points representing this function, and compress them to form a piecewise linear representation. If the difference of the slope between adjacent segments is more than 25%, we start a new segment. Each segment is a 3-tuple (x, y, m). The segment starts from point (x, y), and it has a slope m. The last segment is open ended and corresponds to the case in which the curve converges with a straight line (see Figure 6).

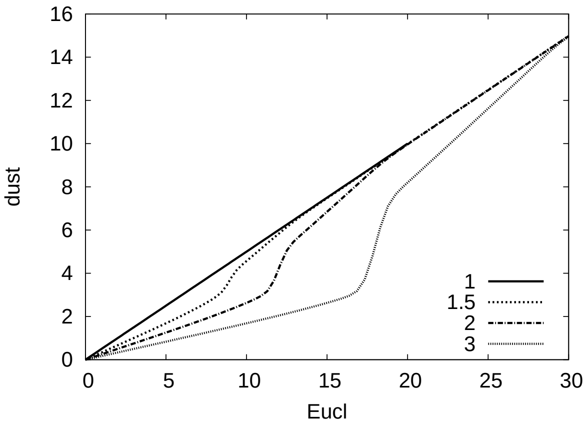

Figure 6: $DUST$ distances for different Gaussian distributions

We construct such a look-up table for all the applicable error functions. In the worst-case, every time stamp $T[i]$ in the time series data is associated with a different error function and we need to construct n different look-up tables. Note that these look-up tables are computed off-line and stored. Thus, the complexity of an on-line *dust* distance calculation is low. To calculate a *dust* distance for Δx, we first identify the appropriate look-up table and then the appropriate segment within the look-up table. We then subtract the starting point of the segment from Δx and multiply the obtained value by the slope for the segment. As we do a binary search to identify the appropriate segment, the worst-case complexity of a *dust* distance is $O(log(n))$ where n is the number of segments. We typically have somewhere between 5 to 15 segments.

When using $DUST$ to perform 1-NN classification, most of the time series are relatively far away and only a few are close by. Hence, for most time series the distance at each time stamp falls in the range of the last segment. To optimize the computation, we first detect if the distance is within the range of the last segment. We only search for the appropriate segment if this is not the case.

4. EXPERIMENTAL VALIDATION

4.1 Overview

We evaluate the effectiveness of the $DUST$ distance on three different data mining tasks: 1-NN classification, motif detection, and top-k nearest-neighbor search. For classification, we look at the UCR classification datasets [12]. We randomly perturb a fraction of the values and plot the accuracy of the classification for $DUST$ and the Euclidean distance.

For the case of motif detection and nearest-neighbor search, we need to define a metric for comparing the results between different distance measures. We propose the following axiom. **Effective distance measures on uncertain data should allow us to reason about the original data**

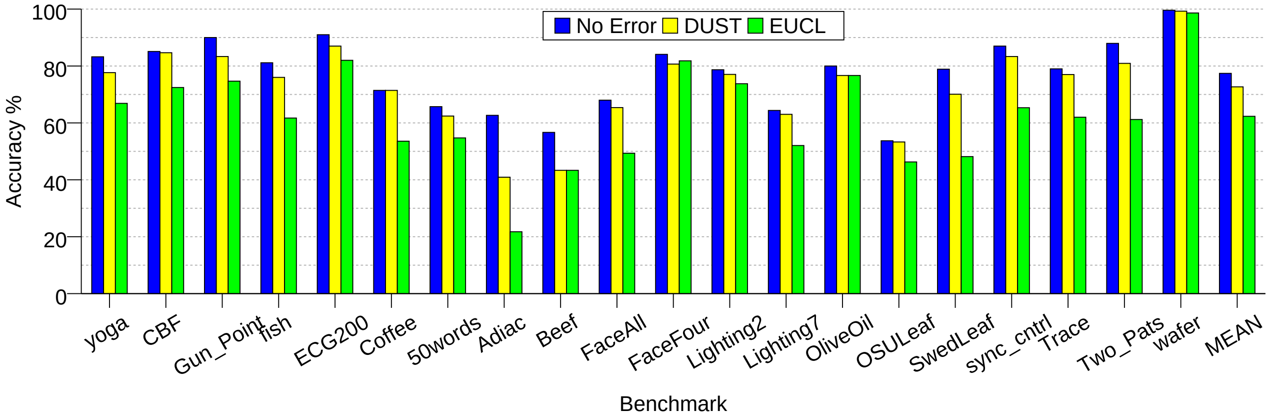

Figure 7: Classification accuracy for $DUST$ vs Euclidean Distance

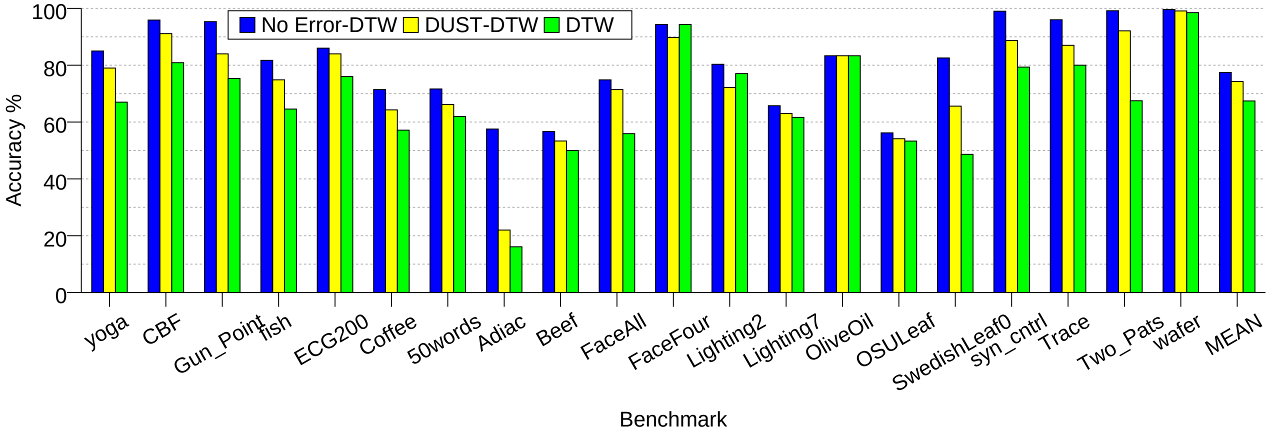

Figure 8: Classification accuracy for $DUST$ vs DTW

without uncertainty. To evaluate the effectiveness of different measures we propose an approach similar to Johannes et. al. [3]. We take original data, perturb it with different error functions, and then evaluate the results with different distance measures.

As the run times for computing the $DUST$ distance are fast, we focus our evaluation on the effectiveness of $DUST$. For all experiments, we computed the average over 10 different random runs. To show the observed trends we plot all graphs using Bazier curves.

4.2 Classification

In this section, we consider all the UCR datasets [12]. These datasets represent time series data, where the time series have been classified into a few classes. For each dataset there is a training set and a test set. The objective is to perform a 1-NN classification of the test set by finding the nearest match in the training set. We assume that the data has been generated by noisy sensors. Like prior work [3, 21] we artificially perturb the data. For the first 10% of the values we use a normal error function with standard deviation σ, and for the next 10% we use a standard deviation of $\sigma/2$. This captures the fact that out of a lot of sensors most of them are likely to be fairly accurate. Few of the sensors will have some error, and a few more will have a slightly larger value of error. Roughly similar trends were reported by others [17, 15, 7].

We evaluate the accuracy for six configurations: on original data using Euclidean distance (*No Error*) and DTW (*No Error-DTW*), on perturbed data using *EUCL*, *DUST*, *DTW*, and DTW with *dust* (*DUST-DTW*).

For each element in the time series, we vary the standard deviation of the error from 0.1 to 2 times the standard deviation of the element. We compute the classification results for the maximum standard deviation, which is 2. We show the results in Figures 7 and 8.

We observe that in both figures $DUST$ performs 5-15% better than conventional approaches. Only 3 benchmarks out of 20 are error resilient in the sense that the classification accuracy does not decrease significantly. These are FaceFour, OliveOil, and Wafer. For all the other benchmarks there is close to a 10-20% loss in accuracy between *No Error* and *EUCL* or *DTW*. The last group of bars in Figures 7 and 8 show the mean values. In Figure 7, the average classification accuracy is 77% for the case with no error, 72% with $DUST$ and 62% with *EUCL*. Similarly for the case with *DTW*, the accuracy is 78% for no error, 74% with *DUST*, and 67% with *DTW*. We observe that *DTW*

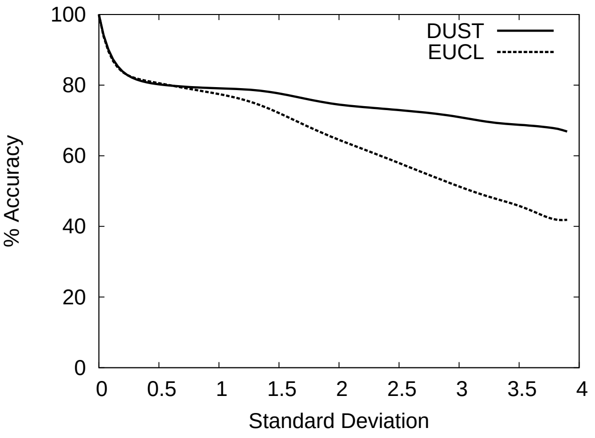

Figure 9: Accuracy vs error for fish

is marginally better than $EUCL$. Overall, we conclude that $DUST$ makes up for more than 50% of the accuracy lost due to uncertainty. In Figure 7 we observe that for Coffee and CBF, $DUST$ performs so well that it almost completely makes up for the introduced error.

Figure 9 shows the accuracy versus standard deviation for the benchmark *fish*. Both $DUST$ and $EUCL$ start at the same point for small error. However, as the error increases the curves start to diverge. We observe that the $DUST$ distance is far more resilient to uncertainty in the data.

4.3 Motif Detection

Motifs are defined as follows. A subsequence of a time series $T[1 \dots n]$ is a contiguous set of values $T_{sub}[j \dots k]$, where $k > j$. A motif is a set of two time series T_1' and T_2' such that $T_1' \subset T$ and $T_2' \subset T$, and out of all such subsets the distance between T_1' and T_2' is minimum. In our experiments we do not consider this general case. Like [14], we consider motifs with a fixed size n which are non-overlapping. We find top-k motifs, which are the top-k closest pairs. Motifs are used to find frequently occurring patterns in time series and can be used to construct time series dictionaries, and are also the basis for sophisticated clustering algorithms [11].

For detecting top-k motifs we use both the time series datasets that were used in [14]. The first dataset captures the behavior of an insect over time. The second is an EEG (Electroencephalogram) dataset. We find motifs of size 128.

As proposed in Section 4.1, we compute the accuracy of $DUST$ as follows. We first find the top-10 motifs without any error. Then we perturb the data as described in Section 4.2. We then compute the top-10 motifs using $DUST$ and $EUCL$. Subsequently, we compute the intersections of these sets with the set of motifs computed earlier when there was no error. We assume that the data is normalized with a standard deviation of 1. The results are shown in Figures 10 and 11.

As expected, we observe that for extremely small errors, there is no difference between $DUST$ and $EUCL$. The accuracy of $DUST$ is slightly lower for smaller standard deviations. We will investigate the reasons for this phenomenon as part of our future work. However, for larger standard deviations, $DUST$ maintains the same level of accuracy for a large range, whereas $EUCL$ continues to degrade.

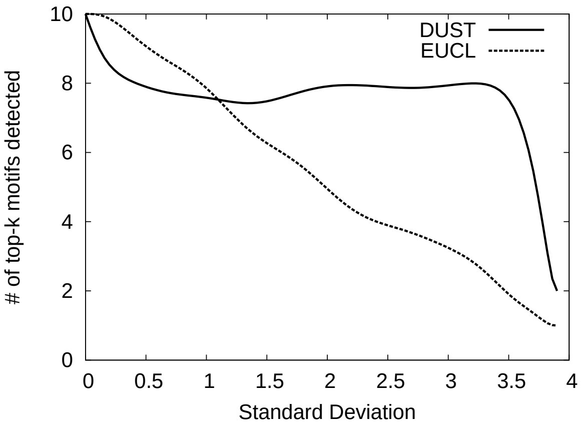

Figure 10: Top-k motifs (EEG dataset)

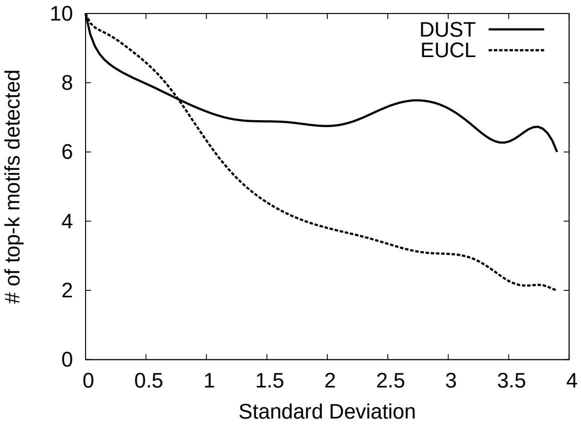

Figure 11: Top-k motifs (Insect dataset)

4.4 Top-k Nearest-Neighbor Search

We now consider the problem of finding the k nearest neighbors. For this purpose we need a dataset with a large number of entries. The larger the number of entries, the more difficult it is to ensure that the set of k nearest neighbors remains the same. We scanned the UCR datasets and picked one of the datasets with the largest number of training examples. Both wafer and Two_Patterns had 999 entries each. We randomly chose Wafer.

We chose the first entry of the test set and found its k nearest neighbors. Then, we perturbed the training data, and computed the k nearest neighbors again. We report the intersection of these two sets.

Figure 12 and 13 show the error rates for different percentages of erroneous sensors in the time series, starting at 10% up to 40%. We observe that there is a sharp dip in the accuracy between 20% and 30%. We will investigate this phenomenon as a part of future work.

Figure 14 and 15 show the accuracy as a function of the standard deviation for different error functions. *Rect* is the

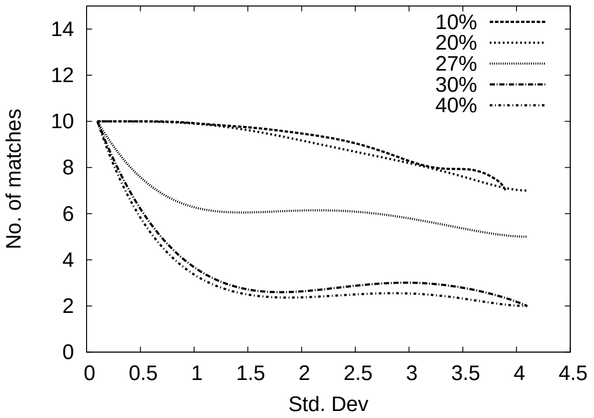

Figure 12: Accuracy vs Std Dev. for different % of erroneous sensors using $DUST$

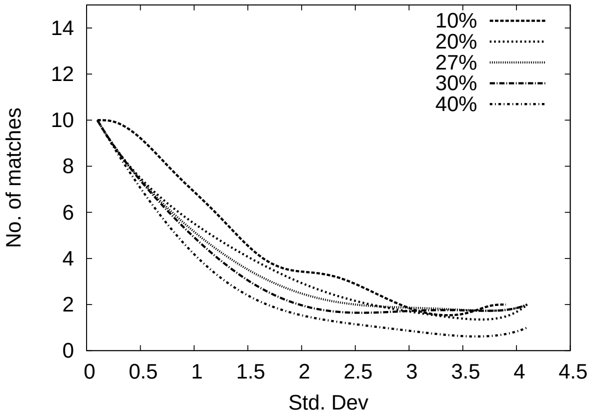

Figure 13: Accuracy vs Std Dev. for different % of erroneous sensors using $EUCL$

rectangular distribution, *Lgn* is the log-normal distribution, and *Edf* is the exponential distribution (see Section 3.4.2). We observe that the accuracy of $DUST$ is fairly consistent across all the error functions. However, the accuracy of $EUCL$ dips sharply for log-normal and rectangular error functions. In all cases $DUST$ is considerably more resilient to errors.

5. CONCLUSION

In this paper, we presented $DUST$, a novel approach for measuring the similarity between uncertain time series. We arrived at $DUST$ after studying the kind of error distributions involved in sensor deployments. We observed that often different error distributions are involved in producing a single time series. However, none of the previously published similarity measures for uncertain time series can accommodate this phenomenon. $DUST$ provides the unique ability to combine any number of arbitrary error distributions. We note here that the applicability of $DUST$ is not confined to only sensor-based systems, it is a generic distance measure that can be used for a large variety of data and applications.

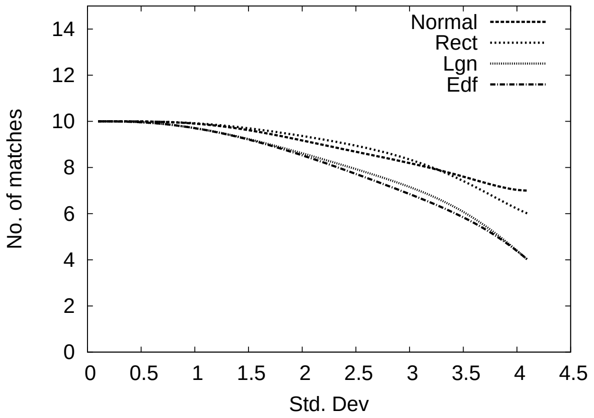

Figure 14: Accuracy vs Std Dev. for different error functions using $DUST$

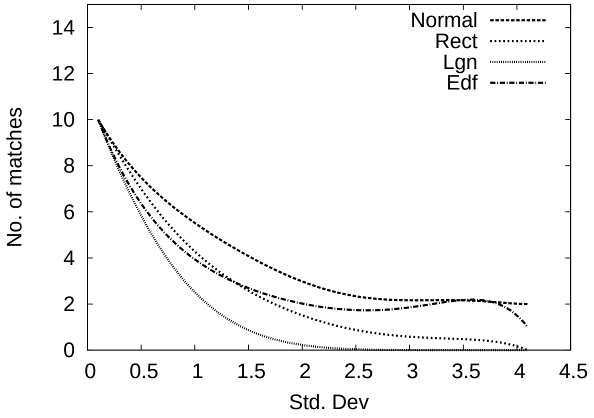

Figure 15: Accuracy vs Std Dev. for different error functions using $EUCL$

We also identified scenarios in which the use of sophisticated measures that accommodate uncertainty fails to provide a significant benefit over traditional measures. For example, if the same Normal distribution is producing the uncertainty for all sensor readings in a time series, then Euclidean distance produces similar results as measures that accommodate uncertainty.

We validated our approach for a wide variety of publicly available data sets and a broad range of parameters. In almost all cases $DUST$ significantly outperformed traditional distance measures such as Euclidean distance and dynamic time warping.

6. REFERENCES

[1] C. C. Aggarwal and P. S. Yu. A framework for clustering uncertain data streams. In *Proceedings of the 2008 IEEE 24th International Conference on Data Engineering*, 2008.

[2] C. C. Aggarwal and P. S. Yu. A survey of uncertain data algorithms and applications. *IEEE Transactions*

on Knowledge and Data Engineering, 21(5):609–623, 2009.

[3] J. Aßfalg, H. Kriegel, P. Kröger, and M. Renz. Probabilistic similarity search for uncertain time series. In *Proceedings of the 21st International Conference on Scientific and Statistical Database Management*, 2009.

[4] D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *Proceedings of the 1994 AAAI Workshop*, 1994.

[5] R. Braff and C. Shively. A method of over bounding ground based augmentation system (gbas) heavy tail error distributions. *Journal of Navigation*, 58(1):83–103, 2005.

[6] S. Cho. Bidirectional data aggregation scheme for wireless sensor networks. In *Proceedings of the 3rd International Conference on Ubiquitous Intelligence and Computing*, 2006.

[7] P. Ciarlini and U. Maniscalco. Mixture of soft sensors for monitoring air ambient parameters. In *Proceedings of the XVIII IMEKO World Congress*, 2006.

[8] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2):1542–1552, 2008.

[9] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. Fast subsequence matching in time-series databases. *SIGMOD Record*, 23(2):419–429, 1994.

[10] S. R. Jeffery, M. Garofalakis, and M. J. Franklin. Adaptive cleaning for rfid data streams. In *Proceedings of the 32nd International Conference on Very Large Databases*, 2006.

[11] E. Keogh, J. Lin, and W. Truppel. Clustering of time series subsequences is meaningless: Implications for previous and future research. *Knowledge and Information Systems*, 8(2), 2005.

[12] E. Keogh, X. Xi, L. Wei, and C. A. Ratanamahatana. The ucr time series classification/clustering homepage. `www.cs.ucr.edu/~eamonn/time_series_data`, Accessed on Feb 5th 2010.

[13] J. Lin, E. J. Keogh, L. Wei, and S. Lonardi. Experiencing sax: a novel symbolic representation of time series. *Data Mining and Knowledge Discovery*, 15(2):107–144, 2007.

[14] A. Mueen, E. J. Keogh, Q. Zhu, S. Cash, and B. Westover. Exact discovery of time series motifs. In *Proceedings of the SIAM International Conference on Data Mining*, 2009.

[15] S. V. R. Nageswara. Algorithms for fusion of multiple sensors having unknown error distributions. In *Proceedings of the 15th Symposium on Energy Engineering Sciences*, 1997.

[16] Nature Publishing Group. Bone marrow transplantation. `www.nature.com/bmt/journal/v31/n8/fig_tab/1703917f2.html`, Accessed on Feb 5th 2010.

[17] S. Palit. Signal extraction from multiple noisy sensors. *Signal Processing*, 61(3):199–212, 1999.

[18] A. D. Sarma, O. Benjelloun, A. Halevy, S. Nabar, and J. Widom. Representing uncertain data: Models, properties, and algorithms. *The International Journal on Very Large Data Bases*, 18(5), 2009.

[19] A. Sharma, L. Golubchick, and R. Govindam. On the prevalence of sensor faults in real-world deployments. In *Proceedings of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, 2007.

[20] J. Sudano. Dynamic real-time sensor performance evaluation. In *Proceedings of the 5th International Conference on Information Fusion*, 2002.

[21] M. Yeh, K. Wu, P. S. Yu, and M. Chen. Proud: A probabilistic approach to processing similarity queries over uncertain data streams. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, 2009.

APPENDIX

Let us consider two time series values x and y. $\Delta x = |x - y|$. Let the error be normally distributed with mean 0, and standard deviation σ.

Let us first assume that the data in the time series is distributed uniformly in a range that is much larger than the error. Using Equations 6 and 7, we calculate $\phi(\Delta x)$(see Equation 4) to be:

$$\frac{1}{2\sigma\pi} e^{-\frac{\Delta x^2}{4\sigma^2}}$$

Hence, $dust(x, x + \Delta x) = \Delta x / 2\sigma$. This distance is dependent only on Δx and is inversely proportional to σ.

Let us now assume that the original time series values follow a Normal distribution ($\mu = 0$, $\sigma = 1$). In this case, $\phi(\Delta x)$ is:

$$\frac{1+\sigma^2}{\sqrt{2\pi}} e^{-\frac{1+\sigma^2}{\sigma^2}} e^{-\frac{\Delta x^2}{4(1+\sigma^2)^2\sigma^2}}$$

Hence, $dust(x, x + \Delta x) = \Delta x / (2\sigma(1 + \sigma^2))$. Again, the distance is only dependent on Δx.