Similarity Matching for Uncertain Time Series: Analytical and Experimental Comparison

Michele Dallachiesa
University of Trento, Italy
dallachiesa@disi.unitn.it

Besmira Nushi
University of Trento, Italy
besmira.nushi@studenti.unitn.it

Katsiaryna Mirylenka
University of Trento, Italy
kmirylenka@disi.unitn.it

Themis Palpanas
University of Trento, Italy
themis@disi.unitn.eu

ABSTRACT

In the last years there has been a considerable increase in the availability of continuous sensor measurements in a wide range of application domains, such as Location-Based Services (LBS), medical monitoring systems, manufacturing plants and engineering facilities to ensure efficiency, product quality and safety, hydrologic and geologic observing systems, pollution management, and others.

Due to the inherent imprecision of sensor observations, many investigations have recently turned into querying, mining and storing uncertain data. Uncertainty can also be due to data aggregation, privacy-preserving transforms, and error-prone mining algorithms.

In this study, we survey the techniques that have been proposed specifically for modeling and processing uncertain time series, an important model for temporal data. We provide both an analytical evaluation of the alternatives that have been proposed in the literature, highlighting the advantages and disadvantages of each approach. We additionally conduct an extensive experimental evaluation with 17 real datasets, and discuss some surprising results. Based on our evaluations, we also provide guidelines useful for practitioners in the field.

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

1. INTRODUCTION

In the last decade there has been a dramatic explosion in the availability of measurements in a wide range of application domains, including traffic flow management, meteorology, astronomy, remote sensing, and object tracking. Applications in the above domains usually organize these sequential measurements into time series, i.e., sequences of data points ordered along the temporal dimension, making time series a data type of particular importance.

Several studies have recently focused on the problems of processing and mining time series with incomplete, imprecise and even misleading measurements [6, 12, 20, 21, 22]. Uncertainty in time series may occur for different reasons, such as the inherent imprecision of sensor observations, or privacy-preserving transformations. The following two examples illustrate these two cases:

- Personal information contributed by individuals and corporations is steadily increasing, and there is a parallel growing interest in applications that can be developed by mining these datasets, such as location-based services and social network applications. In these applications, privacy is a major concern, and it can be ensured by different privacy-preserving transforms [2, 10, 16], namely, noisy perturbations, noisy aggregates, and reduced granularity. The data can still be queried and mined but it requires a re-design of the existing methods in order to address the uncertainty introduced by these transforms.
- In manufacturing plants and engineering facilities, sensor networks are being deployed to ensure efficiency, product quality and safety [12]: unexpected vibration patterns in production machines, or changes in the composition of chemicals in industrial processes, are used to identify in advance possible failures, suggesting repairs or replacements. However, sensor readings are inherently imprecise because of the noise introduced by the equipment itself [6]. This translates to time series with uncertain values, and addressing this uncertainty can provide better results in terms of quality and efficiency.

While the problem of managing and processing uncertain data has been studied in the traditional database literature since the 80's [3], the attention of researchers was only recently focused on the specific case of uncertain time series.

ACM QUeST'11 November 1, 2011, Chicago, IL, USA
Copyright 2011 ACM ISBN 978-1-4503-1037-6 ...$10.00.

Two main approaches have emerged for modeling uncertain time series. In the first, a probability density function (pdf) over the uncertain values is estimated by using some a priori knowledge [24, 23, 18]. In the second, the uncertain data distribution is summarized by repeated measurements (i.e., samples) [5].

In this study, we revisit the techniques that have been proposed under these two approaches, with the aim of determining their pros and cons. This is the first study to undertake a rigorous comparative evaluation of the techniques proposed in the literature for similarity matching of uncertain time series. The importance of such a study is underlined by two facts: first, the widespread existence of uncertain time series; and second, the observation that similarity matching serves as the basis for developing various more complex analysis and mining algorithms. Therefore, acquiring a deep understanding of the techniques proposed in this area is essential for the further development of the field of uncertain time series processing.

In summary, we make the following contributions.

- We review the state of the art techniques for similarity matching in uncertain time series, and analytically evaluate them. Our analysis serves as a single-stop comparison of the proposed techniques in terms of requirements, input data assumptions, and applicability to different situations.
- We propose a methodology for comparing these techniques, based on the similarity matching task. This methodology provides a common ground for the fair comparison of all the techniques.
- We perform an extensive experimental evaluation, using 17 real datasets from diverse domains. In our experiments, we evaluate the techniques using a multitude of different conditions, and input data characteristics. Moreover, we stress-test the techniques by evaluating their performance on datasets for which they have not been designed to operate.
- Finally, we provide a discussion of the results (some of which are surprising), and complement this discussion with thoughts on interesting research directions, and useful guidelines for the practitioners in the field.

The rest of this paper is structured as follows. In Section 2 we survey the principal representations and distance measures proposed for similarity matching of uncertain time series. In Section 3, we analytically compare the methods proposed for uncertain time series modeling, and in Section 4, we present the experimental comparison. Finally, Section 5 concludes this study.

2. SIMILARITY MATCHING FOR UNCERTAIN TIME SERIES

Time series are sequences of points, typically real valued numbers, ordered along the temporal dimension. We assume constant sampling rates and discrete timestamps. Formally, a time series S is defined as $S =< s_1, s_2, ..., s_n >$ where n is the length of S, and s_i is the real valued number of S at timestamp i. Where not specified otherwise, we assume normalized time series with zero mean and unit variance. Notice that normalization is a preprocessing step that requires particular care to address specific situations [13].

In this study, we focus on uncertain time series where uncertainty is localized and limited to the points. Formally, an uncertain time series T is defined as a sequence or independent random variables $< t_1, t_2, ..., t_n >$ where t_i is the random variable modeling the real valued number at timestamp i. All the three models we review and compare fit under this general definition.

The problem of similarity matching has been extensively studied in the past [4, 9, 17, 11, 7, 15, 14, 13] : given a user-supplied query sequence, a similarity search returns the most similar time series according to some distance function. More formally, given a collection of time series $C = \{S_1, ..., S_N\}$, where N is the number of time series, we are interested in evaluating the range query function $RQ(Q, C, \epsilon)$:

$$RQ(Q, C, \epsilon) = \{S | S \in C \wedge distance(Q, S) \leq \epsilon\} \quad (1)$$

In the above equation, ϵ is a user-supplied distance threshold. A survey of representation and distance measures for time series can be found in [8].

A similar problem arises also in the case of uncertain time series, and the problem of probabilistic similarity matching has been introduced in the last years. Formally, given a collection of uncertain time series $C = \{T_1, ..., T_N\}$, we are interested in evaluation the probabilistic range query function $PRQ(Q, C, \epsilon, \tau)$:

$$PRQ(Q, C, \epsilon, \tau) = \{T | T \in C | Pr(distance(Q, S) \leq \epsilon) \geq \tau\} \quad (2)$$

In the above equation, ϵ and τ are the user-supplied distance threshold and the probabilistic threshold, respectively.

In the recent years three techniques have been proposed to evaluate PRQ queries, namely MUNICH[1] [5], PROUD [23], and DUST [18]. In the following sections, we discuss each one of these three techniques.

2.1 MUNICH

In [5], uncertainty is modeled by means of repeated observations at each timestamp, as depicted in Figure 1(a).

Assuming two uncertain time series, X and Y, MUNICH proceeds as follows. First, the two uncertain sequences X, Y are materialized to all possible certain sequences: $TS_X = \{< v_{11}, ..., v_{n1} >, ..., < v_{1s}, ..., v_{ns} >\}$ (where v_{ij} is the j-th observation in timestamp i), and similarly for Y with TS_Y. Thus, we have now defined TS_X, TS_Y. The set of all possible distances between X and Y is then defined as follows:

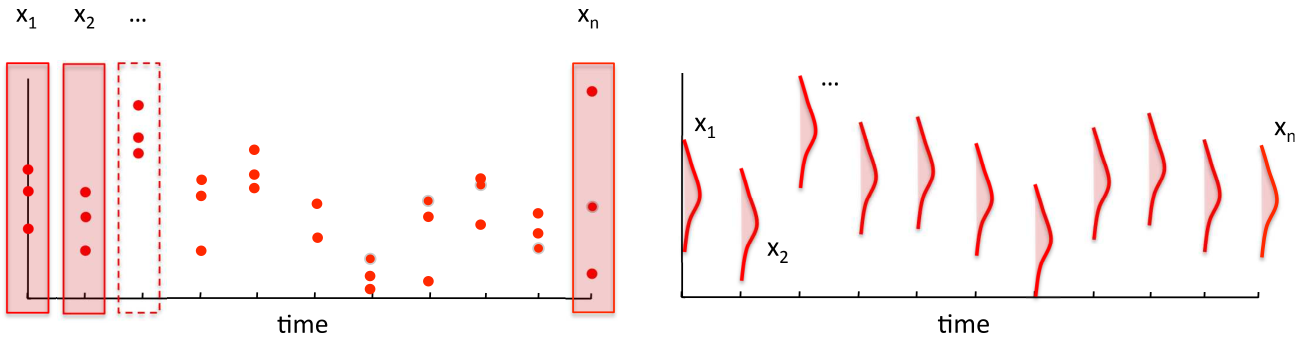
$$dists(X, Y) = \{L^p(x, y) | x \in TS_X, y \in TS_Y\} \quad (3)$$

The uncertain L^p distance is formulated by means of counting the feasible distances:

$$Pr(distance(X, Y) \leq \epsilon) = \frac{|\{d \in dists(X, Y) | d \leq \epsilon\}|}{|dists(X, Y)|} \quad (4)$$

Once we compute this probability, we can determine the result set of PRQs similarity queries by filtering all uncertain sequences using Equation 4.

[1] We will refer to this method as *MUNICH* (it was not explicitly named in the original paper), since all the authors were affiliated with the University of Munich.



(a) (b)

Figure 1: Example of an uncertain time series $X = \{x_1, ..., x_n\}$ modeled by means of repeated observations (a), and *pdf* estimation (b).

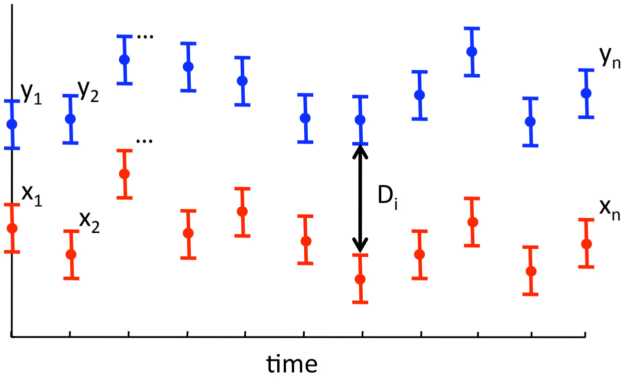


Figure 2: The probabilistic distance model.

Note that the naive computation of the result set is infeasible, because of the exponential computational cost: $|dists(X,Y)| = s_X^n s_X^n$ where s_X, s_Y are the number of samples at each timestamp of X, Y, respectively, and n is the length of the sequences. Efficiency can be ensured by upper and lower bounding the distances, and summarizing the repeated samples using minimal bounding intervals [5]. This framework has been applied to Euclidean and DTW distances and guarantees no false dismissals in the original space.

2.2 PROUD

In [23], an approach for processing queries over PRObabilistic Uncertain Data streams (PROUD) is presented. Inspired by the Euclidean distance, the PROUD distance is modeled as the sum of the differences of the streaming time series random variables, where each random variable represents the uncertainty of the value in the corresponding timestamp. This model is illustrated in Figure 1(b).

Given two uncertain time series X, Y, their distance is defined as:

$$distance(X,Y) = \sum_i {D_i}^2 \tag{5}$$

where $D_i = (x_i - y_i)$ are random variables, as shown in Figure 2.

According to the central limit theorem, we have that the cumulative distribution of the distances approaches a normal distribution:

$$distance(X,Y)_{norm} = \frac{distance(X,Y) - \sum_i E[d_i^2]}{\sqrt{\sum_i Var[D_i^2]}} \tag{6}$$

The normalized distance follows a standard normal distribution, thus we can obtain the normal distribution of the original distance as follows:

$$distance(X,Y) \propto N(\sum_i E[D_i^2], \sum_i Var[D_i^2]) \tag{7}$$

The interesting result here is that, regardless of the data distribution of the random variables composing the uncertain time series, the cumulative distribution of their distances (1) is defined similarly to their euclidean distance and (2) approaches a normal distribution. Recall that we want to answer PRQs similarity queries. First, given a probability threshold τ and the cumulative distribution function (*cdf*) of the normal distribution, we compute ϵ_{limit} such that:

$$Pr(distance(X,Y)_{norm} \leq \epsilon_{limit}) \geq \tau \tag{8}$$

The *cdf* of the normal distribution can be formulated in terms of the well known *error-function*, and ϵ_{limit} can be determined by looking up the statistics tables. Once we have ϵ_{limit}, we proceed by computing also the normalized ϵ:

$$\epsilon_{norm}(X,Y) = \frac{\epsilon^2 - E[distance(X,Y)]}{\sqrt{Var[distance(X,Y)]}} \tag{9}$$

Then, we have that if a candidate uncertain series Y satisfies the inequality:

$$\epsilon_{norm}(X,Y) \geq \epsilon_{limit} \tag{10}$$

then the following equation holds:

$$Pr(distance(X,Y)_{norm} \leq \epsilon_{norm}(X,Y)) \geq \tau \tag{11}$$

Therefore, Y can be added to the result set. Otherwise, it is pruned away. This distance formulation is statistically sound and only requires knowledge of the general characteristics of the data distribution, namely, its mean and variance.

2.3 DUST

In [18], the authors propose a new distance measure, DUST, that compared to MUNICH, does not depend on the existence of multiple observations and is computationally more efficient. Similarly to [23], DUST is inspired by the Euclidean distance, but works under the assumption that all the time series values follow some specific distribution. Given two uncertain time series X, Y, the distance between two uncertain values x_i, y_i is defined as the distance between their true (unknown) values $r(x_i), r(y_i)$: $dist(x_i, y_i) = L^1(r(x_i), r(y_i))$. This distance can then be used to define a function ϕ that measures the similarity of two uncertain values:

$$\phi(|x_i - y_i|) = Pr(dist(0, |r(x_i) - r(y_i)|) = 0) \qquad (12)$$

This basic similarity function is then used inside the *dust* dissimilarity function:

$$\begin{aligned} dust(x, y) &= \sqrt{-\log(\phi(|x-y|)) - k} \\ &\text{with} \\ k &= -\log(\phi(0)) \end{aligned}$$

The constant k has been introduced to support reflexivity. Once we have defined the *dust* distance between uncertain values, we are ready to extend it to the entire sequences:

$$DUST(X, Y) = \sqrt{\sum_i dust(x_i, y_i)^2} \qquad (13)$$

The handling of uncertainty has been isolated inside the ϕ function, and its evaluation requires to know exactly the data distribution. In contrast to the techniques we reviewed earlier, the DUST distance is a real number that measures the dissimilarity between uncertain time series. Thus, it can be used in all mining techniques for certain time series, by simply substituting the existing distance function.

Finally, we note that DUST is equivalent to the Euclidean distance, in the case where the error of the time series values follows the normal distribution.

3. ANALYTICAL COMPARISON

In this section, we compare the three models of similarity matching for uncertain time series, namely, MUNICH, PROUD and DUST, along the following dimensions: uncertainty models used and assumptions made by the algorithms; type of distance measures; and type of similarity queries.

3.1 Uncertainty Models and Assumptions

All three techniques we have reviewed are based on the assumption that the values of the time series are independent from one another. That is, the value at each timestamp is assumed to be independently drawn from a given distribution. Evidently, this is a simplifying assumption, since neighboring values in time series usually have a strong temporal correlation.

The main difference between MUNICH and the other two techniques is that MUNICH represents the uncertainty of the time series values by recording multiple observations for each timestamp. This can be thought of as sampling from the distribution of the value errors. In contrast, PROUD and DUST consider each value of time series to be a continuous random variable following a certain probability distribution.

The amount of preliminary information, i.e. a priori knowledge of the characteristics of the time series values and their errors, varies greatly among the techniques. MUNICH does not need to know the distribution of the time series values, or the distribution of the value errors. It simply operates on the observations available at each timestamp.

On the other hand, PROUD and DUST, need to know the distribution of the error at each value of the data stream. In particular, PROUD requires to know the standard deviation of the uncertainty error, and a single observed value for each timestamp. PROUD assumes that the standard deviation of the uncertainty error remains constant across all timestamps.

DUST uses the largest amount of information among the three techniques. It takes as input a single observed value of the time series for each timestamp, just like PROUD. In addition, DUST needs to know the distribution of the uncertainty error at each time stamp, as well as the distribution of the values of the time series. This means that, in contrast to PROUD, DUST can take into account mixed distributions for the uncertainty errors (albeit, they have to be explicitly provided in the input).

Overall, we observe that the three techniques make different initial assumptions about the amount of information available for the uncertain time series, and have different input requirements. Consequently, when deciding which technique to use, users should take into account the information available on the uncertainty of the time series to be processed.

3.2 Type of Distance Measures

All the considered techniques use some variation of the Euclidean distance. MUNICH and PROUD use this distance in a pretty straightforward manner. Moreover, MUNICH and DUST can be employed to compute the Dynamic Time Warping distance [19], which is a more flexible distance measure.

DUST is a new type of distance, specifically designed for uncertain time series. In other words, DUST is not a similarity matching technique per se, but rather a new distance measure. It has been shown that DUST is proportional to the Euclidean distance in the cases where the value errors are normally distributed [18]. Moreover, the authors of [18] note that if all the value errors follow the same distribution, then it is better to use the Euclidean distance. DUST becomes useful when the value errors are modeled by multiple error distributions.

3.3 Type of Similarity Queries

MUNICH and PROUD are designed for answering probabilistic range queries (defined in Section 2). DUST being a distance measure, it can be used to answer top-k nearest neighbor queries, or perform top-k motif search.

MUNICH and PROUD solve the similarity matching problem that is described by Equation 8, resulting to a set of time series that belong to the answer with a certain probability, τ. On the other hand, DUST produces a single value that is an exact (i.e., not probabilistic) distance between two uncertain time series.

In Section 4, we describe the methodology we used in order to compare all three techniques using the same task, that of similarity matching.

4. EXPERIMENTAL COMPARISON

In this section, we present the experimental evaluation of the three techniques. We first describe the methodology and datasets used, and then discuss the results of the experiments.

All techniques were implemented in C++, and the experiments were run on a PC with a 2.13GHz CPU and 4GB of RAM.

4.1 Experimental Setup

4.1.1 Datasets

Similarly to [5, 23, 18], we used existing time series datasets with exact values as the ground truth, and subsequently introduced uncertainty through perturbation. Perturbation models errors in measurements, and in our experiments we consider *uniform*, *normal* and *exponential* error distributions with zero mean and varying standard deviation within interval $[0.2, 2.0]$.

We considered 17 real datasets from the UCR classification datasets collection [1], representing a wide range of application domains: 50words, Adiac, Beef, CBF, Coffee, ECG200, FISH, FaceAll, FaceFour, Gun_Point, Lighting2, Lighting7, OSULeaf, OliveOil, SwedishLeaf, Trace, and synthetic_control. The training and testing sets were joined together, and we obtained on average 502 time series of length 290 per dataset. We stress the fact that each dataset contains several time series instances.

Since DUST requires to know the distribution of values of the time series, and additionally makes the assumption that this distribution is uniform [18], we tested the datasets to check if this assumption holds. According to the Chi-square test, the hypothesis that the datasets follow the uniform distribution was rejected (for all datasets) with confidence level $\alpha = 0.01$. Similarly, the Kolmogorov-Smirnov test for normality showed that the hypothesis that the time series values follow the normal distribution is rejected with the same confidence level.

4.1.2 Comparison Methodology

In our evaluation, we consider all three techniques, namely, MUNICH, PROUD, and DUST, and we additionally compare to Euclidean distance. When using Euclidean distance, we do not take into account the distributions of the values and their errors: we just use a single value for every timestamp, and compute the traditional Euclidean distance based on these values.

The goal of our evaluation is to compare the performance of the different techniques on the same task. Observe that we can not use the top-k search task for this comparison. The reason is that the MUNICH and PROUD techniques have a notion of probability (Equation 2). This means that these techniques can produce different rankings when the threshold ε changes. For example, assume that we increase ε (maintaining τ fixed). Then the ordering of the time series in a top-k ranking may change, since the probability that the time series are similar within distance $\varepsilon_1 \geq \varepsilon$ may increase. Thus, in the case of uncertain time series, MUNICH and PROUD might produce very different top-k answers even if ε varies a little. This, in turn, means that the top-k task is not suitable for comparing the three techniques.

We instead perform the comparison using the task of time series similarity matching. Even though DUST is not a similarity matching technique (like PROUD and MUNICH), it can still be used to find similar time series, when we specify a maximum threshold on the distance between time series. In [18], the evaluation of DUST was based on top-k similar time series. However, we note that this problem includes the problem of similarity matching [8], where the most similar time series form the answer to the top-k query.

Following the above discussion, in order to perform a fair comparison we need to specify distance thresholds for all three techniques. This translates to finding equivalent thresholds ε for each one of the techniques. We proceed as follows.

Since the distances in MUNICH and PROUD are based on the Euclidean distance, we will use the same threshold for both methods, ε_{eucl}. Then, we calculate an equivalent threshold for DUST, ε_{dust}. Given a query q and a dataset C, we identify the 10th nearest neighbor of q in C. Let that be time series c. We define ε_{eucl} as the Euclidean distance on the observations between q and c and ε_{dust} as the DUST distance between q and c. This procedure is repeated for every query q.

The quality of results of the different techniques is evaluated by comparing the query results to the ground truth. We performed experiments for each dataset separately, using each one of the time series as a query and performing a similarity search. In the graphs, we report the averages of all these results, as well as the 95% confidence intervals.

4.2 Results on Quality Performance

In order to evaluate the quality of the results, we used the two standard measures of *recall* and *precision*. Recall is defined as the percentage of the truly similar uncertain time series that are found by the algorithm. Precision is the percentage of similar uncertain time series identified by the algorithm, which are truly similar. Accuracy is measured in terms of F_1 score to facilitate the comparison. The F_1 score is defined by combining precision and recall:

$$F_1 = 2 * \frac{precision * recall}{precision + recall} \qquad (14)$$

We verify the results with the exact answer using the ground truth, and compare the results with the algorithm output (as described in Section 4.1.2).

4.2.1 Accuracy

The first experiment, represents a special case with restricted settings. This was necessary to do, because the computational cost of MUNICH was prohibitive for a full scale experiment. We compare MUNICH, PROUD, DUST and Euclidean on the Gun_Point dataset, truncating it to 60 time series of length 6. For each timestamp, we have 5 samples as input for MUNICH. Results are averaged on 5 random queries. For both MUNICH and PROUD we are using the optimal probabilistic threshold, τ, determined after repeated experiments. Distance thresholds are chosen (according to Section 4.1.2) such that in the ground truth set they return exactly 10 time series.

The results (refer to Figure 3) show that all techniques perform well (F_1 >80%) when the standard deviation of the errors is low ($\sigma = 0.2$), with MUNICH being the best performer (F_1=88%). However, as the standard deviation increases to 2, the accuracy of all techniques decreases. This is expected, since a larger standard deviation means that the time series have more uncertainty. The behavior of MUNICH though, is interesting: its accuracy falls sharply for $\sigma > 0.6$. (This trend was verified with different error distributions and datasets, but we omit these results for brevity.)

Figure 4 shows the results of the same experiment, but just for PROUD, DUST, and Euclidean. In this case (and for all the following experiments), we report the average results over the full time series for all datasets. Once again, the error distribution is normal (results for uniform and exponential distributions are very similar, and omitted for brevity), and PROUD is using the optimal threshold, τ, for every value of the standard deviation.

The results show that there is virtually no difference among the different techniques. This observation holds across the entire range of standard deviations that we tried.

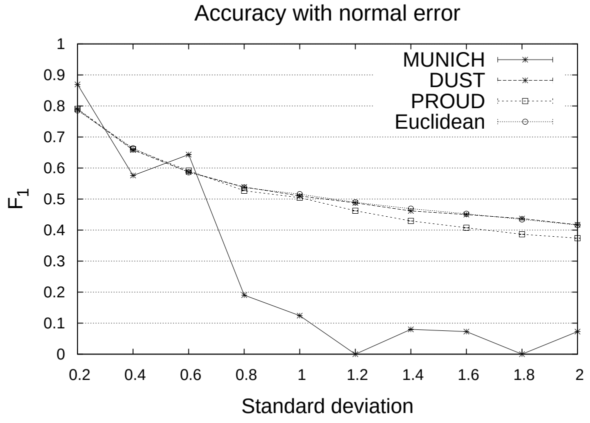


Figure 3: F_1 score for MUNICH, PROUD, DUST and Euclidean on Gun_Point truncated dataset, when varying the error standard deviation (normal error distribution).

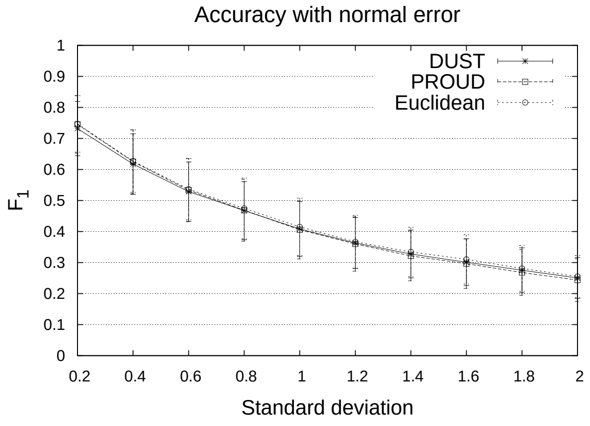


Figure 4: F_1 score for PROUD, DUST and Euclidean, averaged over all datasets, when varying the error standard deviation (normal error distribution).

4.2.2 *Precision and Recall*

In order to better understand the behavior of the different techniques, we take a closer look at precision and recall. Figures 5 and 6 show respectively precision and recall for PROUD, as a function of the error standard deviation, when the distribution of the error follows a uniform, a normal, and an exponential distribution (results for DUST and Euclidean exhibit the same trends). PROUD is using the optimal threshold, τ, for every value of the standard deviation.

The graphs show that recall always remains relatively high (between 63%-83%). On the contrary, precision is heavily affected, falling from 70% to a mere 16% as standard deviation increases from 0.2 to 2. Therefore, an increased standard deviation does not have a significant impact on the false positives, but introduces many false negatives, which may be undesirable.

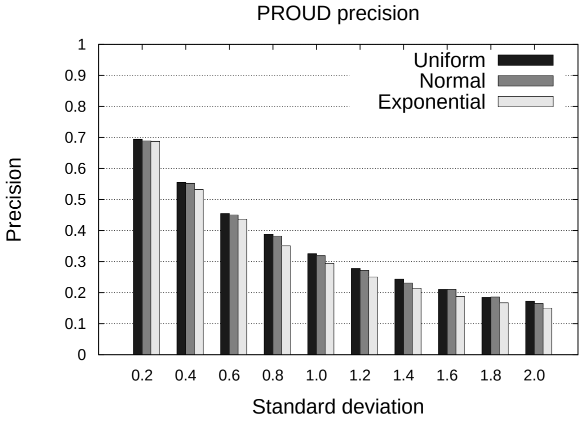


Figure 5: Precision and recall for PROUD, averaged over all datasets, when varying error standard deviation and error distribution.

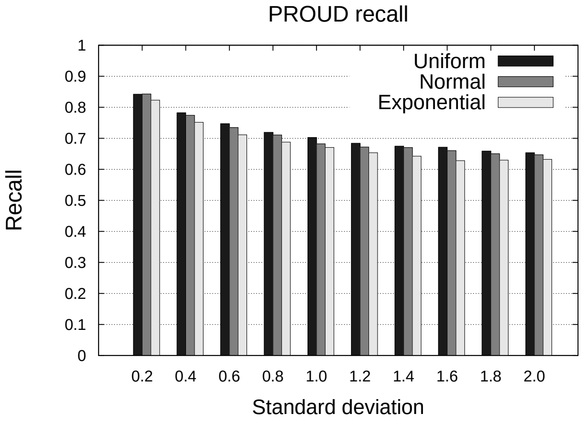


Figure 6: Precision and recall for PROUD, averaged over all datasets, when varying error standard deviation and error distribution.

4.2.3 *Mixed Error Distributions*

While in all previous experiments the error distribution is constant across all the values of a time series, in this experiment we evaluate the accuracy of PROUD, DUST, and

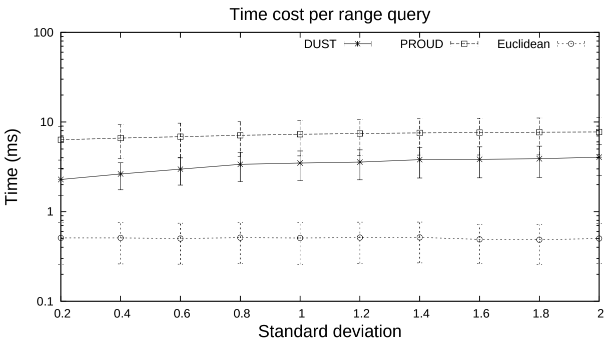


Figure 8: Average time per query for PROUD, DUST, and Euclidean, averaged over all datasets, when varying the error standard deviation with normal error distribution.

Euclidean when we have different error distributions present in the same time series (Figure 7). Each time series has been perturbated with normal error, but of varying standard deviation. Namely, the error for 20% of the values has standard deviation 1, and the rest 80% has standard deviation 0.4.

We note that this is a case that PROUD cannot handle, since it does not have the ability to model different error distributions within the same time series (in this experiment, PROUD was using a standard deviation setting of 0.7). Therefore, PROUD does not produce better results than Euclidean. On the other hand, DUST is taking into account these variations of the error, and achieves a slightly improved accuracy (3% more than PROUD and Euclidean).

4.3 Time

In Figure 8, we report the CPU time per query for the normal error distribution when varying the error standard deviation in the range $[0.2, 2.0]$. Results for uniform and exponential distributions are very similar, and omitted for brevity.

The graph shows that the standard deviation of the normal distribution only slightly affects performance for DUST. As expected, Euclidean is not affected at all, and exhibits the best time performance of all techniques.

We note that for PROUD we did not use the wavelet synopsis, since we did not use any summarization technique for the other techniques either. However, it is possible to apply PROUD on top of a Haar wavelet synopsis. This results in CPU time for PROUD that is equal or less to the CPU time of Euclidean, while maintaining high accuracy [23].

We did not include the time performance for MUNICH in this graph, because it is orders of magnitude more expensive (i.e., in the order of min).

5. DISCUSSION AND CONCLUSIONS

In this work, we reviewed the existing techniques for similarity matching in uncertain time series, and performed analytical and experimental comparisons of the techniques. Based on our evaluation, we can provide some guidelines for the use of these techniques.

MUNICH and PROUD are based on the Euclidean distance, while DUST proposes a new distance measure. Nevertheless, DUST outperforms Euclidean only if the distribution of the observation errors is mixed, and the parameters of this distribution are known.

An important factor for choosing among the available techniques is the information that is available about the distribution of the time series and its errors. When we do not have enough, or accurate information on the distribution of the error, PROUD and DUST do not offer an advantage in terms of accuracy when compared to Euclidean. Nevertheless, Euclidean does not provide quality guarantees while MUNICH, PROUD and DUST do.

The probabilistic threshold τ has a considerable impact on the accuracy of the MUNICH and PROUD techniques. However, it not obvious how to set τ, and no theoretical analysis has been provided on that. The only way to pick the correct value is by experimental evaluation, which can sometimes become cumbersome.

Our experiments showed that MUNICH is applicable only in the cases where the standard deviation of the error is relatively small, and the length of the time series is also small (otherwise the computational cost is prohibitive). However, we note that this may not be a restriction for some real applications. Indeed, MUNICH's high accuracy may be a strong point when deciding the technique to use.

In conclusion, we note that the area of uncertain time series processing and analysis is new, with many interesting problems. In this study, we evaluated the state of the art techniques for similarity matching in uncertain time series, because it can be the basis for more complex algorithms. We believe that the results we report and the experience we gained will be useful for the further research investigations in this area.

Acknowledgments: Part of this work was supported by the FP7 EU IP project KAP (grant agreement no. 260111).

6. REFERENCES

[1] Keogh, E., Xi, X., Wei, L. & Ratanamahatana, C. A. (2006). The UCR Time Series Classification/Clustering Homepage: www.cs.ucr.edu/ eamonn/time_series_data/. Accessed on 17 May 2011.

[2] C. Aggarwal. On Unifying Privacy and Uncertain Data Models. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 386–395. IEEE, 2008.

[3] C. Aggarwal. *Managing and Mining Uncertain Data.* Springer-Verlag New York Inc, 2009.

[4] R. Agrawal, C. Faloutsos, and A. Swami. Efficient similarity search in sequence databases. *Foundations of Data Organization and Algorithms*, pages 69–84, 1993.

[5] J. Aßfalg, H.-P. Kriegel, P. Kröger, and M. Renz. Probabilistic similarity search for uncertain time series. In *SSDBM*, pages 435–443, 2009.

[6] M. Ceriotti, M. Corra, L. D'Orazio, R. Doriguzzi, D. Facchin, S. Guna, G. P. Jesi, R. L. Cigno, L. Mottola, A. L. Murphy, M. Pescalli, G. P. Picco, D. Pregnolato, and C. Torghele. Is There Light at the Ends of the Tunnel? Wireless Sensor Networks for Adaptive Lighting in Road Tunnels. In *International Conference on Information Processing in Sensor Networks (IPSN)*, pages 187–198, 2011.

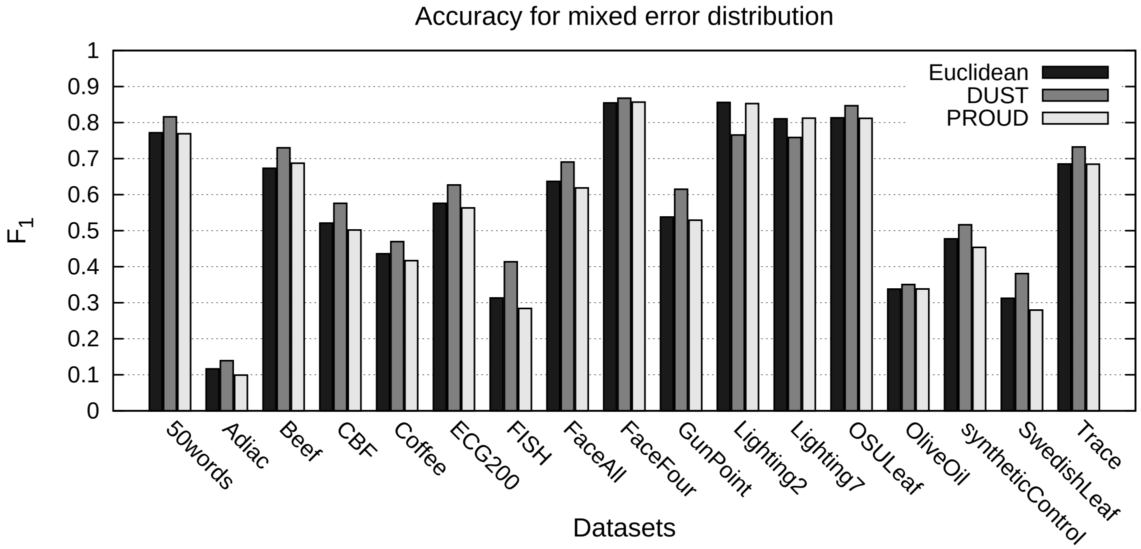


Figure 7: F_1 score for PROUD, DUST, and Euclidean on all the datasets with mixed error distribution (normal, 20% with standard deviation 1.0, and 80% with standard deviation 0.4).

[7] K. Chan and A. Fu. Efficient time series matching by wavelets. In *Data Engineering, 1999. Proceedings., 15th International Conference on*, pages 126–133. IEEE, 2002.

[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. *ACM SIGMOD Record*, 23(2):419–429, 1994.

[10] B. C. M. Fung, K. Wang, R. Chen, and P. S. Yu. Privacy-preserving data publishing: A survey of recent developments. *ACM Comput. Surv.*, 42(4), 2010.

[11] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra. Dimensionality reduction for fast similarity search in large time series databases. *Knowledge and Information Systems*, 3(3):263–286, 2001.

[12] L. Krishnamurthy, R. Adler, P. Buonadonna, J. Chhabra, M. Flanigan, N. Kushalnagar, L. Nachman, and M. Yarvis. Design and deployment of industrial sensor networks: experiences from a semiconductor plant and the north sea. In *Proceedings of the 3rd international conference on Embedded networked sensor systems*, pages 64–75. ACM, 2005.

[13] J. Lin, E. 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] Y. Moon, K. Whang, and W. Han. General match: a subsequence matching method in time-series databases based on generalized windows. In *Proceedings of the 2002 ACM SIGMOD international conference on Management of data*, pages 382–393. ACM, 2002.

[15] Y. Moon, K. Whang, and W. Loh. Duality-based subsequence matching in time-series databases. In *Data Engineering, 2001. Proceedings. 17th International Conference on*, pages 263–272. IEEE, 2002.

[16] S. Papadimitriou, F. Li, G. Kollios, and P. S. Yu. Time series compressibility and privacy. In *VLDB*, pages 459–470, 2007.

[17] N. Roussopoulos, S. Kelley, and F. Vincent. Nearest neighbor queries. In *Proceedings of the 1995 ACM SIGMOD international conference on Management of data*, pages 71–79. ACM, 1995.

[18] S. Sarangi and K. Murthy. DUST: a generalized notion of similarity between uncertain time series. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 383–392. ACM, 2010.

[19] J. Shieh and E. J. Keogh. sax: indexing and mining terabyte sized time series. In *KDD*, pages 623–631, 2008.

[20] M. Stonebraker, J. Becla, D. J. DeWitt, K.-T. Lim, D. Maier, O. Ratzesberger, and S. B. Zdonik. Requirements for science data bases and scidb. In *CIDR*, 2009.

[21] D. Suciu, A. Connolly, and B. Howe. Embracing uncertainty in large-scale computational astrophysics. In *MUD*, pages 63–77, 2009.

[22] T. T. L. Tran, L. Peng, B. Li, Y. Diao, and A. Liu. Pods: a new model and processing algorithms for uncertain data streams. In *SIGMOD Conference*, pages 159–170, 2010.

[23] M. Yeh, K. Wu, P. 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*, pages 684–695. ACM, 2009.

[24] Y. Zhao, C. C. Aggarwal, and P. S. Yu. On wavelet decomposition of uncertain time series data sets. In *CIKM*, pages 129–138, 2010.