In Proc. 24th Int. Conf. on Data Engineering (ICDE'08), Cancun, Mexico, 2008.

T-Time: Threshold-Based Data Mining on Time Series

Johannes Aßfalg, Hans-Peter Kriegel, Peer Kröger, Peter Kunath, Alexey Pryakhin, Matthias Renz

Institute for Informatics, Ludwig-Maximilians-Universität München, Germany
Email: {assfalg,kriegel,kroegerp,kunath,pryakhin,renz}@dbs.ifi.lmu.de

***Abstract*— Mining time series data is an important approach for the analysis in many application areas as diverse as biology, environmental research, medicine, or stock chart analysis. As nearly all data mining tasks on this kind of data depend on a distance function between two time series, a huge number of such functions has been developed during the last decades. The introduction of threshold-based distance functions presented a new concept of time series similarity and these functions were applied to data mining techniques on a wide spectrum of time series data. In this demonstration, we present the Java toolkit *T-Time* which is able to perform several data mining tasks for a complete range of threshold values in an interactive way. The results are visually presented in a very concise way so that the user can easily identify important threshold values. Combined with domain-specific knowledge, these pivotal values can yield novel insights beyond the means of the underlying data mining techniques the analysis is based on.**

I. Introduction

From environmental sensor station data to the results of scientific experiments and from stock charts to the dynamics of human behavior recorded in sociological studies: time series data can be derived from nearly every corner of the real world. For the analysis of these collections, efficient and effective data mining methods are required. Usually, these data mining techniques rely on a distance function on time series. In fact, the development of suitable distance functions for time series has attracted a lot of research effort recently.

Well-known distance functions include for example the Euclidean distance, or Dynamic Time Warping (DTW) [1]. Recently, we proposed the new concept of defining similarity between time series based on thresholds [2], [3]. Threshold similarity considers intervals during which the time series exceeds a certain threshold for comparing time series rather than using the exact time series values. Traditional distance functions for time series as described above consider all amplitude ranges to be equally important. In contrast to such approaches, the threshold-based similarity is able to base its distance notion on distinguished amplitude values. This approach has proven to be more suitable than traditional distance functions in a lot of real-world applications. Obviously, the choice of a suitable threshold value is very crucial.

The *T-Time* application presented in this work implements a *visual data mining* approach that presents the data in a clear and user-friendly way in order to enable interactive data exploration, e.g. cluster analysis. In particular, *T-Time*

1) assists the user in identifying potentially interesting threshold values;
2) enables the visual and interactive exploration of other data analysis parameters;
3) allows the user to interactively and visually extract novel knowledge from a large amount of data derived from data mining algorithms.

The main focus of *T-Time* therefore is the interactive and visual analysis of the impact of different threshold values on the results of data mining tasks. The concept of our application supports the extraction of novel insights in supervised as well as in unsupervised settings. If class labels are available, the user can easily scan for threshold values that yield high classification accuracies in cross-validation experiments. This subsequently allows for the identification of ranges of important amplitudes of the time series, i.e. ranges where small differences in the absolute values account for large differences in the meaning (different classes) of the time series. But even in an unsupervised situation where no pre-classified time series are available, *T-Time* can be very helpful. By a quick visual inspection of several clustering results derived for example by OPTICS [4] it is possible to discover important and interesting thresholds based on their ability to form distinct cluster structures.

Though *T-Time* is an application for the evaluation of threshold-based similarity, we also included DTW and the Euclidean distance in *T-Time* for the purpose of comparing different similarity notions. The collection of implemented distance functions can easily be extended when need arises. All visual data mining functionalities of *T-Time* can of course also be used with these distance functions.

In summary, *T-Time* is designed as a user-friendly tool that in the hands of domain experts can lead to novel conclusions beyond the means of standard data mining approaches.

II. Theoretical Background

In this section, we will describe the basic notion of threshold-based distance functions for a pair of time series.

a) Interval Generation: A given threshold τ induces a sequence of so called *Threshold-Crossing Time Intervals* as follows:

Let $X = \langle (x_i, t_i) \in \mathbb{R} \times T : i = 1..N \rangle$ be a time series and $\tau \in \mathbb{R}$ be a threshold value. Then the *threshold-crossing time intervals* of X with respect to τ are a sequence $S_{\tau,X} = \langle (l_j, u_j) \in T^2 : j \in \{1, .., M\}, M \leq N \rangle$ of time intervals such that

$$\forall t \in T : (\exists j \in \{1, .., M\} : l_j < t < u_j) \Leftrightarrow x(t) > \tau.$$

b) Distance Functions on Intervals: There exist a number of possibilities to compare two intervals. An overview of standard approaches taking into account different combinations of interval start points, interval end points, or overlapping regions of intervals can be found in [5]. Among the most effective distance functions encountered during our research were the Overlap Measure and several distance functions based on the Minkowski metric. In the following, let A and B be two intervals where l_A denotes the start point of A, u_A denotes the end point of A, and l_B and u_B denote the corresponding points of interval B. Then the distance between the intervals A and B based on the Overlap Measure $d_{overlap}(A, B)$ is defined as follows:

$$d_{overlap}(A, B) = \min\{u_A, u_B\} - \max\{l_A, l_B\}$$

The distance functions $d_{minkowski_p}(A, B)$ based on the Minkowski metric are defined as follows:

$$d_{minkowski_p}(A, B) = \sqrt[p]{(l_A - l_B)^p + (u_A - u_B)^p}$$

In *T-Time* we included the functions defined by the three most common Minkowski parameter values $p = 1$, 2, and ∞, and the Overlap Measure.

c) Distance Functions on Interval Sequences: Having defined distance functions on pairs of intervals allows us to define distance functions on two sets of intervals corresponding to a pair of time series. Several distance measures for set-based objects have been introduced in the literature [6]. A very well performing measure is the Sum of Minimum Distances (SMD) that was implemented for *T-Time*. Furthermore, we included a set-kernel based approach [7]. As the set kernel is based on kernel functions defined on the elements of the sets (i.e., the intervals), we kernelized the distance functions described above with a Gaussian kernel as described in [8].

III. Practical Benefits of T-Time

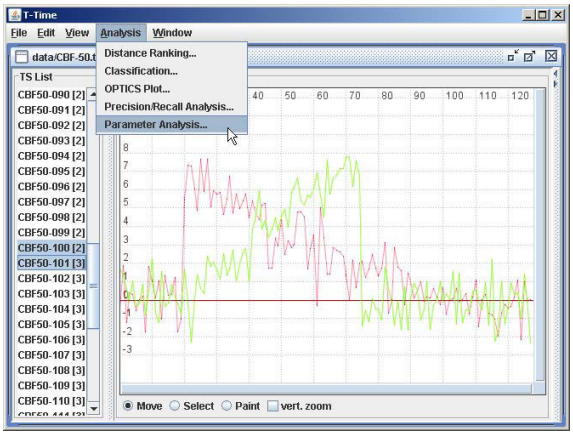


Fig. 1. Main Window of *T-Time*.

In both, supervised and unsupervised settings, *T-Time* guides the user through the process of identifying pivotal threshold values. In this section, we describe how *T-Time* can be used to identify interesting ranges of amplitude values.

A. T-Time User Interface

We implemented *T-Time* using Java 1.5. The main control window of *T-Time* allows the user to import collections of time series. Figure 1 depicts the corresponding view for an imported dataset. The left hand side of a dataset window features a textual entry for each time series. If available, class labels appear in brackets.

On the right hand side of the dataset window, the time series are displayed as diagrams for a brief visual inspection. Time series of different classes are displayed in different colors. By selecting several time series simultaneously, it is possible to directly compare them. In Figure 1, two time series belonging to different classes have been selected.

After selecting all or a subset of the time series, the user can start one of the numerous data mining algorithms included in the tool. The following sections show how different threshold values influence unsupervised as well as supervised data mining tasks. Furthermore we demonstrate how *T-Time* guides the user through the non-trivial process of identifying meaningful threshold values.

B. Supervised Analysis

If pre-classified time series are available, it is possible to perform a number of different analysis tasks using several distance measures each induced by a different threshold value. In case of a supervised analysis, i.e. class labels are available, *T-Time* detects ranges of threshold values that lead to a high class separability. Traditional distance functions as described above consider all amplitude ranges to be equally important. In contrast to such approaches, the threshold-based similarity is based on distinguished amplitude values which are specified by the threshold parameter τ. This concept has been proven to be superior for explaining differences in many real-world data sets.

In order to determine meaningful thresholds, *T-Time* employs classifiers like the *kNN* classifier. Cross-validation experiments can be performed to determine average classification accuracy values or the corresponding confusion matrix. Another possibility is to create precision-recall plots for different distance functions and varying thresholds.

However, one of the most useful *T-Time* applications is the automatic identification of distinguishing threshold values for threshold-based distance functions. In Figure 2, an example output is depicted. For a number of threshold values along the x-axis, classification accuracy values are plotted in y-axis direction. Usually one or a few distinct ranges of suitable threshold values can be identified in this way. In the depicted example, the most distinguishing threshold values can be found in the range between 3 and 6. We observed such a distinct range of meaningful threshold values for most real-world datasets. This underlines the practical importance of a threshold-based definition of time series similarity. Based on such kind of information and depending on the application

domain, conclusions about critical time series values can be drawn.

We applied *T-Time* to a set of classified time series representing human gene expression data. We used a dataset of the Gene Expression Omnibus (GEO)[1] [9] containing gene expression profiles of proliferating normal peripheral blood mononuclear cells (PBMC) infected with HIV type 1 RF assessed at five postinfection time points compared with those of matched uninfected PBMC. We then tried to detect pathological genes. The idea is to derive quality curves as depicted in Figure 2 for each subset of the dataset corresponding to a certain gene. As expected we found that most genes yielded no distinct peak when computing the quality curves with respect to the classification system (healthy vs. infected cells). However, a few genes did yield such a distinguished region. That means these genes act significantly different in healthy and in infected cells and are thus candidates to be highly pathological. For example, one of these genes is *NFYC* which plays a role in the transcription of the *MHCII* genes that are blocked by an HIV protein. Another gene featuring a noticeable quality curve is *PLAUR* whose expression is known to be affected by an HIV infection [10].

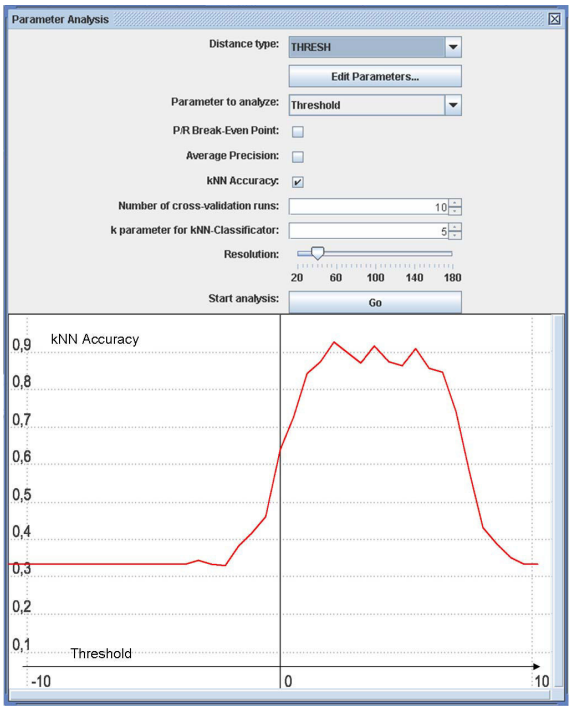
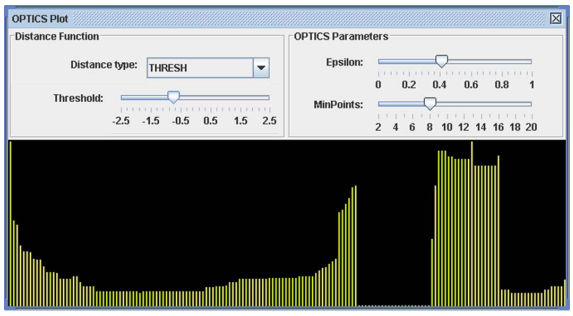


Fig. 2. Identification of Distinguishing Thresholds.

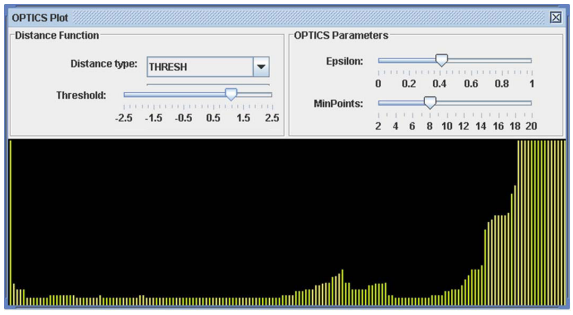
C. Unsupervised Analysis

Even if only unlabeled time series objects are available, *T-Time* can be of great help to analyze the impact of different distance functions and especially to identify ranges

[1]http://www.ncbi.nlm.nih.gov/geo/



(a) Threshold $\tau_1 = -0.75$



(b) Threshold $\tau_2 = 1.1$

Fig. 3. Unsupervised Threshold Analysis.

of distinguishing threshold values. While in principle every clustering approach could be used, we decided to integrate OPTICS [4] into *T-Time* as its results can easily be interpreted visually. OPTICS is a variant of single-link clustering that avoids the single-link effect by using a density estimator for data grouping. OPTICS provides a linear ordering of the data objects that can be visualized by means of a so-called *reachability diagram*. This visualization of the hierarchical clustering structure is much clearer compared to dendrograms. Valleys in this reachability diagram indicate clusters. Of course, any other clustering or visualization technique can be modularly integrated in the analysis process. Thus, the visual approach of OPTICS integrates seamlessly into the concept of *T-Time*.

While our tool offers Euclidean Distance and DTW for the unsupervised analysis as well, our example for the unsupervised setting depicted in Figure 3 once again focuses on the threshold-based distance functions.

Note the different positions of the slide control for the threshold parameter in Figure 3(a) and in Figure 3(b). Our application enables the user to interactively vary the threshold τ. When the user changes the threshold value, a new OPTICS plot is generated and so the user can easily explore the impact of the threshold parameter on the cluster structure. Thus, the impact of the threshold on the cluster structure of the objects can be evaluated. In the depicted example, the threshold $\tau_1 = -0.75$ results in 3 clearly separated OPTICS clusters while the threshold $\tau_2 = 1.1$ yields only one large cluster. So, τ_1 could be more interesting for the user than threshold $\tau_2 = 1.1$, especially if for example the number of

clusters corresponds to the number of subtypes of a certain disease.

We successfully applied our toolkit to a dataset that consists of gene expression data corresponding to patient responses to the drug 'Tamoxifen'. The dataset was taken from the Pharmacogenetics and Pharmacogenomics Knowledge Base (PharmGKB)[2] [11]. We observed a dramatically changing cluster structure when varying the threshold. In case of $\tau = 0$ we can observe 3 clusters (indicated as valleys in the plot, whereas when dropping τ to -0.3, we can only observe 2 clusters with a completely different cluster membership of patients. Thus, with different thresholds, we can cluster the patients according to varying phenotypes. Subsequently a biologist might use this information to identify important genes and crucial expression levels.

REFERENCES

[1] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series." in *KDD Workshop*, 1994.

[2] J. Aßfalg, H.-P. Kriegel, P. Kröger, P. Kunath, A. Pryakhin, and M. Renz, "Similarity search on time series based on threshold queries." in *EDBT*, 2006.

[3] ——, "Threshold similarity queries in large time series databases." in *ICDE*, 2006.

[4] M. Ankerst, M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: Ordering points to identify the clustering structure." in *SIGMOD*, 1999.

[5] T. K. Johnson, *A reformulation of Coombs' Theory of Unidimensional Unfolding by representing attitudes as intervals.* Doctoral thesis, University of Sydney, Psychology, 2006.

[6] T. Eiter and H. Mannila, "Distance measure for point sets and their computation." in *Acta Informatica, 34*, 1997.

[7] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola, "Multi-instance kernels." in *ICML*, 2002.

[8] J. Aßfalg, H.-P. Kriegel, P. Kröger, P. Kunath, A. Pryakhin, and M. Renz, "Time series analysis using the concept of adaptable threshold similarity." in *SSDBM*, 2006.

[9] T. Barrett, D. T. DB, S. Wilhite, P. Ledoux, D. Rudnev, C. Evangelista, I. Kim, A. Soboleva, M. Tomashevsky, and R. Edgar, "Incbi geo: mining tens of millions of expression profiles–database and tools update." in *Nucleic Acids Research*, 2006.

[10] M. Storgaard, N. Obel, F. T. Black, and B. Moller, "Decreased urokinase receptor expression on granulocytes in hiv-infected patients," in *Scandinavian Journal of Immunology*, 2002.

[11] T. Klein, J. Chang, M. Cho, K. Easton, R. Fergerson, M. Hewett, Z. Lin, Y. Liu, S. Liu, D. Oliver, D. Rubin, F. Shafa, J. Stuart, and R. Altman, "Integrating genotype and phenotype information: An overview of the pharmgkb project." in *The Pharmacogenomics Journal*, 2001.

[2]http://www.pharmgkb.org/