PCA-based Similarity Search: Pre-processing & Distance Measures

L. Karamitopoulos, G.Evangelidis

Dpt. of Applied Informatics, University of Macedonia, Thessaloniki, Greece
Tel: +30 2310891844, Fax: +30 2310891800, E-mail:{lkaramit,gevan}@uom.gr

Time series appear frequently in several domains such as in multimedia, business, industry or medicine. A multivariate time series dataset is a set of co-evolving time series that relates to a specific object (e.g. the motion of a person). The increasing need for analyzing efficiently the huge amount of this information leads to the application of data mining techniques. At the core of these techniques lies the concept of similarity since most of them require searching for similar patterns, such as in query by content, clustering or classification. Nevertheless, when dealing with multivariate time series datasets, similarity should be sought between the whole datasets and not only between the individual time series, since there are usually important correlations among them that shouldn't be lost. In this paper, we discuss the application of Principal Component Analysis (PCA) on multivariate time series datasets for the purpose of similarity search. PCA is applied in order to reduce the high dimensionality of such data while retaining as much as possible of the variation present in the data. We provide a thorough description of the pre-processing phase with respect to PCA assumptions and limitations, as well as, to the most frequently appeared distortions in data. Furthermore, we experimentally explore the potential usefulness of incorporating Piecewise Aggregate Approximation into this phase. Finally, we discuss the various aspects of the proposed PCA-based similarity (dissimilarity) measures.

1. Introduction

Several procedures generate huge amounts of data in the form of time series in almost every domain such as in business, industry, medicine and multimedia. A time series is a collection of observations made sequentially through time. At each time point one or more measurements may be monitored corresponding to one or more attributes under consideration. The resulting time series is called univariate or multivariate respectively. The increasing need for analyzing efficiently the huge amount of this information leads to the application of data mining techniques [5]. At the core of these techniques lies the concept of similarity since most of them require searching for similar patterns, such as in query by content, clustering or classification.

In the case of univariate time series, there has been a lot of research covering all aspects of similarity search, namely, pre-processing [6], indexing [1][4], distance measures [7][12] and representation [11]. The similarity is measured between two 1-dimensional time series, that is, there is one attribute of consideration measured for two objects sequentially through time (e.g. the daily closing price of two stocks during the last 90 days). However, the case of multivariate time series has not been extensively explored with respect to all aspects mentioned previously. In this case, the similarity is measured between two p-dimensional time series, that is, there are *p* attributes of consideration measured for two objects sequentially through time

(e.g. the daily closing price, the volume and the daily change in closing price of two stocks during the last 90 days). One approach in dealing with multivariate time series is to consider (concatenate) them as long univariate time series and apply respective techniques [10]. This approach, although simple, ignores the correlations that usually exist among attributes. Another approach is to consider a multivariate time series as a whole, retaining the information that might be hidden in the correlations among the attributes. This approach is more complicated and usually requires expensive computations, however, it may improve similarity search providing at the same time useful information for post hoc analysis [18]. For this purpose, Principal Component Analysis (PCA) seems to be an appropriate tool, since it is often applied for the purpose of reducing the high dimensionality of multivariate datasets (i.e. improve the computation cost), while retaining as much as possible of the variation present in the data.

In this paper, we discuss the application of PCA on multivariate time series for the purpose of similarity search. A thorough description of the pre-processing phase is provided with respect to PCA assumptions and limitations, as well as, to the most frequently appeared distortions in data, namely, offset translation, amplitude scaling, time warping and noise. We explore the potential usefulness of applying a common representation scheme in the time series data mining literature, Piecewise Aggregate Approximation (PAA) [13][19], during the pre-processing phase on the purpose of reducing the expensive cost that PCA incurs. Experiments were performed on a dataset, which relates to the Australian Sign Language (AUSLAN) and has been extensively utilized in the literature.

In Section 2 we briefly provide the background of Principal Component Analysis (PCA). Section 3 discusses the various aspects of pre-processing and distance measures. Furthermore, Piecewise Aggregate Approximation is presented and proposed as a further step in the pre-processing phase. In Section 4, we describe the dataset, the methods and the results of the conducted experiments. Finally, a conclusion and future work is presented in Section 5.

2. Background on Principal Component Analysis

Suppose there is an object for which *p* variables (attributes) X_i are being measured sequentially through time for *n* time instances. The corresponding dataset $\mathbf{X} = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_p}]$ consists of *p* vectors $\mathbf{x_i}$, where $\mathbf{x_i} = [x_{1i}, x_{2i}, \ldots, x_{ni}]'$ is a column vector that contains the *n* measurements of the variable X_i (i.e. $\mathbf{x_i}$ comprises a univariate time series). Each row of **X** corresponds to the measurements of all variables at a specific time instance. Therefore, each row of **X** can be considered as a point in *p*-dimensional space.

Principal Component Analysis derives a new set of orthogonal and uncorrelated composite variates $Y_{(j)}$, which are called principal components:

$$Y_{(j)} = a_{1j} X_1 + a_{2j} X_2 + \ldots + a_{pj} X_p \text{ , where } j = 1, 2, \ldots, p \quad (1)$$

As it is shown from Equation 1, each principal component is a linear combination of the original variables and it is derived in such a manner that each successive component will account for a smaller portion of variation in **X**. The derivation of principal components is based on **Σ,** where **Σ** denotes the covariance matrix of **X**. Another approach is to obtain the principal components from the correlation matrix and that has the advantage of dealing with the problem of variables measured in

different units. A thorough comparison between the two methods can be found in [8].The results from the application of PCA can be presented in matrix forms. In particular, the component weights a_{ij} define component's position in the space and they form a matrix of size *p* x *p*. Let C denote the matrix of component weights, with each column corresponding to a principal component. $Y_{(j)}$'s can be used to compute component scores, i.e., the representation of **X** in the principal component space. Let Y denote the matrix of size *n* x *p*, which comprises of these scores. In addition to these matrices, variances of the principal components in increasing order form a matrix **S** of size *p* x 1.

Intuitively, Principal Component Analysis transforms a dataset **X** by rotating the original axes of a *p*-dimensional space and deriving a new set of axes (components) in such a manner that the first axis accounts for the maximum variance, the second axis accounts for the maximum of the remaining variance and so on. Hopefully, the first few (*k*) components will retain most of the variation present in all of the original variables (*p*) and thus, an essential dimensionality reduction may be achieved by projecting the original data on this new *k*-dimensional space, as long as, $k << p$.

3. PCA-based Similarity Search

3.1 Pre-processing

Considering univariate time series, similarity search is based on shapes, meaning that two time series are considered similar when their shapes are considered to be "close enough". Apparently, the notion of "close enough" depends heavily on the application itself, a fact that affects the decision of the pre-processing steps to be followed. The pre-processing phase in similarity search aims at dealing with several commonly appeared distortions in raw data, namely, offset translation, amplitude scaling, noise, and time warping. Distance measures may be affected seriously by the presence of any of these distortions, resulting most of the times in missing similar shapes. For example, two time series may have identical shapes where the first one is vertically shifted (offset translation) and/or is of different scale (amplitude scaling) with respect to the second one. Offset translation can be handled by subtracting the corresponding mean from their values, whereas amplitude scaling can be handled by dividing the values by the corresponding standard deviations. Another case is when two time series exhibit similar patterns at different rates. This compressions or decompressions in time should be taken into account. Dynamic Time Warping (DTW) [2] has been successfully applied in order to accommodate this distortion, providing at the same time a distance measure. Finally, noise can be handled in various ways with the most common one to be applying a moving average method.

Considering multivariate time series and Principal Component Analysis, the offset translation is inherently handled, since principal components are derived from the covariance matrix **Σ** of the original dataset **X** for which Equation 2 holds:

$$\mathrm{cov}(X_i, Y_j) = \mathrm{cov}((X_i - \mathrm{mean}(X_i))\,(Y_j - \mathrm{mean}(Y_j))) \qquad (2)$$

Amplitude scaling, as mentioned before, can be handled by dividing each value by the corresponding standard deviation. In PCA, this transformation in conjunction with the previous one is equivalent to deriving the principal components from the correlation matrix. Therefore, there are two options: utilize the covariance matrix (handle offset translation) or utilize the correlation matrix (handle both offset

translation and amplitude scaling). Moreover, both matrices by definition capture the existing dispersion in variables, a fact that intuitively seems to deal with time warping [18]. Finally, noise can be handled as in the univariate case.

Another issue in the pre-processing phase is analyzing time series of different lengths. In the univariate case there are two approaches: either to transform the time series to be of equal length (e.g. by linear interpolation) or to handle it algorithmically during the phase of similarity search [3]. PCA requires variables (time series) of equal length for the same object, therefore, this is a limitation of this technique. However, similarity search is performed on objects, and thus, it is based on covariance or correlation matrices, which are all of the same size *p* x *p* (as long as the same *p* variables are being measured from each object).

3.2 Distance Measures

Similarity search requires a measure that quantifies the similarity or dissimilarity between two objects. In the multivariate case, an object is represented by a matrix of size *n* x *p* and thus, the corresponding measure should be based on the differences between matrices. In this paper we discuss distance measures based on Principal Component Analysis. We assume that PCA is applied on two objects and based on these results we wish to quantify their similarity. As it was stated in section 2, these results that may be presented in a matrix form are the component weights (C_{pxp}), variances (S_{px1}) and scores (Y_{nxp}). Utilizing C and/or S for determining a distance measure, an essential dimensionality reduction can be achieved, especially if *k* components ($k << p$) are to be retained [14][15][16][18]. On the other hand, it is more difficult to determine a distance measure based only on the score matrix Y, since scores from different objects map to different spaces. In the remainder of this section, we will present two similarity measures, as examples of utilizing component weights (C) and/or variances (S).

Krzanowski [14] proposed a similarity measure between two objects represented by the matrices A and B of the same number (*p*) of columns (variables) but not necessarily the same number of rows (time instances). This approach, first, applies PCA separately on both matrices and retains *k* principal components from each one. The selection of *k* can be based on any criterion, for example, the latent root or variance criterion [8]. Thereafter, the similarity measure proposed, that in this paper will be denoted by *Sim*, is defined as in Equation 3:

$$Sim(A,B) = trace(C_A C_B' C_B C_A') = \sum_{i=1}^{k}\sum_{j=1}^{k} \cos^2 \theta_{ij} \ , \ 0 \le Sim(A,B) \le k \qquad (3)$$

where, C_A and C_B are the component weights matrices of A and B respectively, and θ_{ij} is the angle between the i^{th} principal component of A and the j^{th} principal component of B. Intuitively, *Sim* measure is based on the angles between all the combinations of the first k components from the two matrices A and B.

Another similarity measure based on both *Sim* and *Frobenius norm* is proposed by Yang & Shahabi [18]. Contrary to *Sim*, all components are retained from each matrix and their variances (eigenvalues) form a weight vector *w*. This measure is called Eros and it is defined as in Equation 4:

$$Eros(A,B,w) = \sum_{i=1}^{p} w_i \left| \langle a_i, b_i \rangle \right| = \sum_{i=1}^{p} w_i \left| \cos \theta_i \right| \qquad (4)$$

where, θ_i is the angle between the i^{th} principal components of A and B. Intuitively, the *Eros* measure is based on the acute angles between the corresponding components from the two matrices A and B, taking into account their variances.

3.3 Our Approach

We propose a further step in the pre-processing phase that reduces the dimensionality of the raw data by applying Piecewise Aggregate Approximation (PAA) [13][19]. PAA divides a time series into segments of equal length and records the mean of the corresponding values of each one. Then PCA can be applied on these mean values. In the multivariate case, PAA can be applied extremely fast on each variable (time series) of the matrix $\mathbf{X}_{nxp}$ separately, resulting into a new matrix PX_{mxp}, where *m* equals the number of segments. The main advantage of this approach is that the number of measurements is greatly reduced and the subsequent analysis can have a much lower cost. In addition to that, the noise distortion may lessen. The disadvantage is that a new parameter *m* is added in the process and needs to be tuned. Moreover, this parameter requires *n*/*m* to be an integer and an adjustment is needed to accommodate this case. Tanaka et al. [17] utilize PCA in conjunction with PAA within motif discovery. PCA is applied first on the purpose of converting a multivariate time series to a univariate one (by retaining only the first component) and afterwards PAA is applied on this univariate series.

4. Experiments

4.1 Dataset - Methods

The experiments have been conducted on a real-world dataset relating to the Australian Sign Language (AUSLAN), which has been used extensively in the literature. The AUSLAN dataset contains sensor data gathered from 22 sensors placed on the hands of a native AUSLAN speaker. There are 95 distinct signs each one performed 27 times. More technical information can be found in [9].

We perform nearest neighbor classification and evaluate it by means of predictive accuracy. We use 9-fold cross validation, by dividing the dataset into 9 subsets. Each subset contains 3 examples of each one of the 95 signs. In total, there are 285 objects in each subset. The classification is tested 9 times, each time leaving out one of the 9 subsets and using it as a testing dataset while the other 8 subsets comprise the training dataset. Classification error rates are recorded for each test and corresponding statistics are computed. The observed differences in the error rates among the various methods were statistically tested. Due to the small number of subsets and to the violation of normality assumption in some cases, Wilcoxon Signed-Rank tests were performed at 5% significance level. All the necessary codes and experiments were developed in MATLAB, whereas the statistical analysis was performed in SPSS.

As distance measures, we selected the two PCA-based similarity measures presented in section 3, namely, *Sim* and *Eros*. For comparison reasons, we also included in the experiments the Euclidean Distance (*ED*) between two objects. However, Euclidean distance requires objects of equal size. In order to produce comparable results, we decided to transform the original dataset in order for the time series to be of equal length (60). For this purpose, linear interpolation was applied and all experiments were performed on this transformed dataset. For the

similarity measure *Sim*, the evaluation procedure was followed 4 times by retaining 1, 2, 3 or 4 principal components. A prior exploration of the dataset showed that when 4 principal components were retained, at least 95% of the variance was accounted by them. Regarding *Eros*, the weight vector *w* was computed by averaging the variances of each component across the objects of the training dataset and normalizing them so that $\Sigma w_i = 1$, for i = 1, 2, …22.

Principal Component Analysis is performed on the covariance matrices. Prior exploration showed that when the correlation matrix was used, meaning that a rescaling was performed on data, there were poor classification results. It seems that, in this specific dataset, different scales contribute to the identification of similar objects and thus, the distortion of amplitude scaling should not be handled.

In order to investigate the effects of incorporating Piecewise Aggregate Approximation (PAA) into the pre-processing phase, the evaluation procedure was followed twice. First, the nearest neighbor algorithm is applied on raw data and then it is applied on PAA compressed data. Two values (6 and 10) of the parameter *m* have been tested. This setting was decided in order to achieve more than 80% compression.

4.2 Results

Statistics for the classification error rates are presented in Tables 1 and 2. The results of the statistical tests are not presented due to limited space; however, the corresponding p-values are reported whenever it is necessary.

In Table 1, the first observation is that the *Sim* measure appears to provide better results when only one component was retained. It would be expected that as the number of retained components was increasing, the error rate would be decreasing. However, this does not hold in our case, probably because the first component has better discriminator power alone and not in conjunction with other components. Also, exploration of the data prior to these experiments showed that the first component accounts for more than 80% of the variance almost in all objects. Another observation is that *Eros* performance is better than its competitors with an average error rate of 9.43% and the corresponding differences were statistically significant with both *Sim* (p-value = 0.008) and *ED* (p-value = 0.008). Moreover, there was one test at which the error rate was only 4.21%, the smallest percentage that appeared in all the experiments. The *Sim* measure provided the second best results with an average error rate of 11.89%. Although *Euclidean distance (ED)* comes third (13.76%), it seems to perform relatively well, considering the fact that it has lower error rates than the *Sim* measures, which retain more than one component. Moreover, the difference with *Sim* cannot be considered statistically significant (p-value = 0.063).

Table 2 presents the statistics for the classification error rates, after applying Piecewise Aggregate Approximation (PAA) during the pre-processing phase. In these experiments, we tested the same three measures, but for *Sim* we kept only the “best” one from the previous experiments. By observing the results in Table 2, it is clear that PAA had very similar effects in classification error rates when different compression rates were used (p-values > 0.05). By comparing the three measures after PAA was applied, the results are similar with the ones drawn from Table 1. When *m* = 10, *Eros* provides statistically better results than *Sim* (p-value = 0.007) and *ED* (p-value = 0.007), whereas for *Sim* and *ED* it may be considered that they

provide statistically similar results (p-value = 0.858). Almost the same statistical results were obtained in the case of m = 6.

Sim(k)	Mean	St. Dev.	Min.	Max.
1	11.89	4.05	8.42	18.95
2	21.83	2.49	18.25	25.26
3	18.87	4.30	11.58	23.86
4	16.61	2.31	14.74	21.4
Eros	9.43	4.16	4.21	15.79
ED	13.76	3.05	9.12	17.54

Table 1. Classification Error Rates

	PAA(10)				PAA(6)			
	Mean	S.D.	Min.	Max.	Mean	S.D.	Min.	Max.
Sim (1)	12.05	3.89	8.77	19.30	12.08	3.60	8.42	17.54
Eros	9.01	3.62	5.61	14.74	9.24	3.92	4.21	14.74
ED	12.01	3.50	7.37	17.54	12.79	4.02	8.42	21.05

Table 2. Classification Error Rates After Pre-processing with PAA

The most important observation emerges from the comparison of the results between Table 1 and Table 2. The classification rates are very similar, meaning that the pre-processing with PAA did not affect the corresponding accuracies. More specifically in the case of m = 10, the differences in error rates cannot be considered statistically significant either for *Eros* (p-value = 0.149) or for *Sim* (p-value = 0.673), whereas for *ED* there was a statistically significant improvement (p-value = 0.013). This improvement may due to the smoothing effect that PAA imposes on data. Similar statistical results are obtained in the case of m = 6 except from the *ED*, where the improvement cannot be considered statistically significant (p-value = 0.213). The implication of this observation is that it is probable to apply Principal Component Analysis on compressed data (saving time and space) without sacrificing classification accuracy. Apparently, more experiments should be performed in order to obtain more evidence to support that claim.

5. Conclusion

In this paper, we discussed the application of Principal Component Analysis (PCA) on multivariate time series datasets for the purpose of similarity search. Emphasis was given in the pre-processing phase in two directions. First, we investigated the relation between PCA and the most frequently appeared distortions in data. Second, we experimentally explored the effect that a dimensionality reduction technique, such as the Piecewise Aggregate Approximation (PAA), may have on a data mining task, specifically, on classification. These experiments were conducted on a dataset and three similarity (dissimilarity) measures were utilized in the nearest neighbor algorithm, namely, *Sim*, *Eros* and the Euclidean distance. Finally, we discussed the various aspects of PCA-based similarity (dissimilarity) measures. There were two key observations with respect to the results of the experiments. First, *Eros* performed statistically better than *Sim* and *ED*. However, all measures provided classification error rates within a narrow range, meaning that the practical significance of this result is application depended. A second and probably most

important observation was that classification error rates were not affected by applying PCA on data that was first compressed by PAA. The implication of this observation could be of great importance within data mining context, since there could be an essential reduction in computation and space costs.

Future work will focus on developing a framework for the pre-processing phase in similarity search with PCA and on conducting experiments on more datasets in order to further validate the key observations of this paper. Principal Component Analysis has not been extensively explored in the context of similarity search in multivariate time series and hence, it has the potential to offer more in the Data Mining field.

References

[1] Agrawal R., Faloutsos C., Swami A., "Efficient Similarity Search In Sequence Databases", *Proc. FODO 1993*, pp.69-84, Chicago, Illinois (USA), October 1993.

[2] Berndt D. J., Clifford J., "Using Dynamic Time Warping to Find Patterns in Time Series", *KDD Workshop 1994*, pp.359-370, Seattle, WA (USA), July 2004.

[3] Bozkaya T., Yazdani N., Ozsoyoglu M., "Matching and Indexing Sequences of Different Lengths", *Proc. CIKM 1997*, pp. 128-135, Las Vegas, Nevada (USA), November 1997.

[4] Faloutsos C., Ranganathan M., Manolopoulos Y., "Fast Subsequence Matching in Time Series Databases", *Proc. ACM SIGMOD 1994*, pp. 419-429, Minneapolis, Minnesota (USA), May 1994.

[5] Fayyad U.M., Piatetsky-Shapiro G., Smyth P., "From Data Mining to Knowledge Discovery: An Overview", *In Advances in knowledge Discovery and Data Mining*, pp. 1-30, AAAI / MIT Press 1996.

[6] Goldin D., Kanellakis P., "On Similarity Queries for Time-Series Data: Constraint Specification and Implementation", *Proc. CP 1995*, pp. 137-153, Cassis (France), September 1995.

[7] Gunopoulos D., Das G., "Time Series Similarity Measures", *Tutorial notes ACM SIGKDD 2000*, pp. 243-307, Boston, Massachusetts (USA), August 2000.

[8] Jolliffe I.T., Principal Component Analysis, 2nd Edition, Springer, 2004.

[9] Kadous, M. W., "Temporal Classification: Extending the Classification Paradigm to Multivariate Time Series", PhD Thesis (draft), School of Computer Science and Engineering, University of New South Wales, 2002.

[10] Kahveci T., Singh A., Gurel A., "Similarity Searching for Multi-Attribute Sequences", *Proc. SSDBM 2002*, pp.175-184, Edinburg (Scotland), July 2002.

[11] Keogh E., "Data Mining and Machine Learning in Time Series Databases", *Tutorial ECML/PKDD 2003*, Cavtat-Dubrovnik (Croatia), September 2003.

[12] Keogh E., Kasetty S., "On the Need for Time Series Data Mining Benchmarks: A Survey and Empirical Demonstration", *Proc. ACM SIGKDD 2002*, pp 102-111, Edmonton, Alberta (Canada), July 2002.

[13] Keogh E., Chakrabarti K., Pazzani M., Mehrotra S., "Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases", *Knowledge and Information Systems*, vol. 3, no. 3, pp. 263-286, February 2001.

[14] Krzanowski W., Between-groups Comparison of Principal Components, JASA, 74 (367), 1979.

[15] Li C., Zhai P., Zheng S. Q., Prabhakaran B., "Segmentation and Recognition of Multi-Attribute Motion Sequences", *Proc. ACM Multimedia 2004*, pp. 836-843, New York, New York (USA), October 2004.

[16] Singhal A. And Seborg D. E., Clustering Multivariate Time Series Data, Journal of Chemometrics, 19 (427-38), 2006.

[17] Tanaka Y., Iwamoto K., Uehara K., "Discovery of Time Series Motif from Multi-Dimensional Data Based on MDL Principle", *Machine Learning*, vol. 58, no. 2-3, pp. 269-300, February 2005.

[18] Yang K., Shahabi C., "A PCA-based Similarity Measure for Multivariate Time Series", *Proc. MMDB 2004*, pp. 65-74, Arlington, Virginia (USA), November 2004.

[19] Yi B. K., Faloutsos C., "Fast Time Sequence Indexing for Arbitrary L_p Norms", *Proc. VLDB 2000*, pp. 385-394, Cairo (Egypt), September 2000.