

A Dissimilarity Measure for Comparing Subsets of Data: Application to Multivariate Time Series*

Matthew Eric Otey Srinivasan Parthasarathy
Department of Computer Science and Engineering
The Ohio State University
Contact: srini@cse.ohio-state.edu

Abstract

Similarity is a central concept in data mining. Many techniques, such as clustering and classification, use similarity or distance measures to compare various subsets of multivariate data. However, most of these measures are only designed to find the distances between a pair of records or attributes in a data set, and not for comparing whole data sets against one another. In this paper we present a novel dissimilarity measure based on principal component analysis for doing such comparisons between such data sets, and in particular time series data sets. Our measure accounts for the correlation structure of the data, and can be tuned by the user to account for domain knowledge. Our measure is useful in such applications as change point detection, anomaly detection, and clustering in fields such as intrusion detection, clinical trial data analysis, and stock analysis.

1 Introduction

Similarity is a central concept in data mining. Research in this area has primarily progressed along two fronts: object similarity [3, 17, 12] and attribute similarity [9, 24]. The former quantifies the distance between two objects (rows) in the database, while the latter refers to the distance between attributes (columns). A related problem is that of determining the similarity or dissimilarity of two subsets of data. Basic approaches have involved using classification [15], clustering [18], and mining contrast sets [6]. However, these approaches build models of the data sets, instead of quantifying their dif-

ferences. In this paper we examine the notion of quantifying the dissimilarity between different subsets of data, and in particular, different multivariate time series. We propose a novel dissimilarity measure that can be used to quantify the differences between two data sets.

One motivating application for such a metric could be for analyzing clinical drug trials to detect the efficacy and hepatotoxicity of drugs. Here one can view each patient in the trial as a different time series data set, for which multiple observations at varying time points of various analytes are measured and stored. A dissimilarity measure in this context can help cluster patients into groups of similarity or alternatively detect anomalous patients. Another application could be in financial stock market analysis where different subsets of the data (for example, different sectors or time periods) can be examined for change point detection, anomaly detection and clustering. This requires the development of a suitable dissimilarity measure.

A suitable dissimilarity measure has several requirements. First, it must take into account as much of the information contained in the data sets as possible. For example, simply calculating the Euclidean distance between the centroids of two data sets is ineffective, as this approach ignores the correlations present in the data sets. Second, it must be user-tunable in order to account for domain knowledge. For example, in some domains it may be that differences in the means of two data sets may not be as important as differences in their correlation structures. In this case, differences in the mean should be weighted less than differences in the correlations. Third, the dissimilarity measure should be tolerant of missing and noisy data, since in many domains data collection is imperfect, leading to many missing attribute values.

In this paper we propose a novel dissimilar-

*This work is supported in part by NSF grants (CAREER-IIS-0347662) and (NGS-CNS-0406386), and a grant from Pfizer, Incorporated.

ity metric based on principal component analysis (PCA). Our measure consists of three components that separately take into account differences in the means, correlations, and variances of the data sets (time series) being compared. As such, our measure takes into account much of the information in the data set. It is also possible to weight the components differently, so one can incorporate domain knowledge into the measure. Finally, our measure is robust towards noise and missing data. We demonstrate the efficacy of the proposed metric in a variety of application domains, including anomaly detection, change detection and data set clustering, on both synthetic and real data sets.

The rest of the paper is organized as follows. We first briefly review related work in Section 2. We then present our dissimilarity measure in Section 3, and discuss several applications of the measure. In Section 4, we present experimental results showing the performance of our measure when used for several applications on stock market data sets. Finally in Section 5 we conclude with directions for future work.

2 Related Work

As mentioned above, there have been many metrics proposed that find the distance or similarity between the records of a data set [3, 17, 12], or the between the attributes of a data set [9, 24]. However, these metrics are defined only between a pair of records or attributes. Similarity metrics for comparing two data sets have been used in image recognition [16], and hierarchical clustering [18]. The Hausdorff distance [16] between two sets A and B is the minimum distance r such that all points in A are within distance r of some point in B , and vice-versa. Agglomerative hierarchical clustering frequently makes use of the single-link and complete-link distances between two clusters [18] to decide which pair of clusters can be merged. The single-link distance between two clusters is the minimum pairwise distance between points in cluster A , and points in cluster B , while the complete-link distance is the maximum pairwise distance between points in cluster A , and points in cluster B . There is also an average-link distance [14], which is the average of all pairwise distances between points in cluster A , and points in cluster B . However, these metrics do not explicitly take into account the correlations between attributes in the data sets (or clusters). Parthasarathy and Ogihara [21] propose a similarity metric for clustering data sets based on frequent itemsets. By this metric, two data sets

are considered similar if they share many frequent itemsets, and these itemsets have similar supports. This metric takes into account correlations between the attributes, but it is only applicable for data sets with categorical or discrete attributes.

There has also been work for defining distance metrics that take into account the correlations present in continuous data. The most popular metric is the Mahalanobis distance [22], which accounts for the covariances of the attributes of the data. However this can only be used to calculate the distance between two points in the same data set. Yang *et al* [25] propose an algorithm for subspace clustering (i.e. subsets of both points and attributes in a data set) that finds clusters whose attributes are positively correlated with each other. Böhm *et al* [7] modify the DBSCAN algorithm [11] by using PCA to find clusters of points that are not only density-connected, but correlation-connected as well. That is to say, they find subsets of a data set that have similar correlations. To determine if two points of the data set should be merged into a single cluster, they must be in each other’s “correlation” neighborhood which is determined by a PCA-based approximation to the Mahalanobis distance. This approach is more flexible than Yang *et al*’s in that it can find clusters with negative correlations between the attributes. However, their measure is unable to find subsets of data with similar correlations that are not density-connected. Furthermore, both Yang *et al*’s and Böhm *et al* approaches are interested only in finding clusters of points within a single data set, instead of clustering multiple data sets. Finally, Yang and Shahabi [26] use an extension to the Frobenius norm called Eros to calculate the similarity of two time series. A component of our similarity measure is very similar to Eros (see Section 3.1.2). Unlike Eros, however, our measure contains other components they do not consider. For example, they do not consider the differences in the means of the two time series. Furthermore, in our approach, the weights of the different components can be adjusted based on domain knowledge.

Recently, Aggarwal has argued for user interaction when designing distance functions [2] between points. He presents a parametrized Minkowski distance metric and a parametrized cosine similarity metric that can be tuned for different domains. He also proposes a framework for automatically tuning the metric to work appropriately in a given domain. Based on these ideas in the next section we present a tunable metric for computing a measure of dissimilarity across data (sub)sets.

3 Algorithms

In this section we first present our dissimilarity measure and demonstrate its effectiveness with a small example data set. We then discuss various applications of our dissimilarity measure in detail that demonstrate its utility and flexibility.

3.1 Dissimilarity Measure

Our goal is to quantify the dissimilarity of two homogeneous k -dimensional data sets $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$. This measure of dissimilarity should take into account not only the distances between the data points in $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$, but the correlations between the attributes of the data sets as well.

In general, the dissimilarity of two data sets $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$ is denoted as $D(\overline{\mathbf{X}}, \overline{\mathbf{Y}})$. We define the function D in terms of three dissimilarity functions that take into account the differences in location, rotation, and variance between the data sets. Each of these components are discussed separately below. These three components are combined by means of a product, or by a weighted sum, which allows one to weight the components differently, so as to incorporate domain knowledge. For example, in the domain of network intrusion detection, one may be concerned with time series data sets where column i represents the i th computer on a given subnetwork, and row j represents the number of bytes received between times t_{j-1} and t_j . When comparing subsets of this data set taken from different time points, large differences in the mean may be indicative of a denial-of-service attack. Alternatively, differences in the correlation of the number of bytes received by two different machines may be indicative of one of the machines being used by an unauthorized user. Depending on what the user wishes to detect, the measure can be tuned in different ways.

3.1.1 Distance Component

To determine the distance between two data sets, there are a wide variety of distance metrics we can use. We have implemented several different distance metrics, including the single-link and complete-link distances, among others (see Section 2). In this work we consider two distance measures for the centroids of the data sets. The Euclidean distance between the centroids of each data set is given by:

$$D_d(\overline{\mathbf{X}}, \overline{\mathbf{Y}}) = |\mu_{\overline{\mathbf{X}}} - \mu_{\overline{\mathbf{Y}}}|_2. \quad (1)$$

The other distance measure we use is the Mahalanobis distance, given by:

$$D_d(\overline{\mathbf{X}}, \overline{\mathbf{Y}}) = (\mu_{\overline{\mathbf{X}}} - \mu_{\overline{\mathbf{Y}}}) \Sigma_{\overline{\mathbf{X}\mathbf{Y}}}^{-1} (\mu_{\overline{\mathbf{X}}} - \mu_{\overline{\mathbf{Y}}})^T \quad (2)$$

where $\Sigma_{\overline{\mathbf{X}\mathbf{Y}}}$ is the covariance matrix of the combination of data sets $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$.

3.1.2 Rotation Component

The next component measures the degree to which the data set $\overline{\mathbf{X}}$ must be rotated so that its principle components point in the same direction as those of $\overline{\mathbf{Y}}$. The principal components of a data set are the set of orthogonal vectors such that the first vector points in the direction of greatest variance in the data, the second points in the orthogonal direction of the second greatest variance in the data, and so on [20, 23]. We consider $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$ to be most similar to each other when their principal components, paired according to their ranks, are aligned, and most dissimilar when all of the components of $\overline{\mathbf{X}}$ are orthogonal to those of $\overline{\mathbf{Y}}$.

More formally, given a data set $\overline{\mathbf{X}}$, consider the singular value decomposition (SVD) of its covariance matrix:

$$cov(\overline{\mathbf{X}}) = U \Lambda_X X^T \quad (3)$$

where the columns of X are the principal components of the data set $\overline{\mathbf{X}}$, arranged from left to right in order of decreasing variance in their respective directions, and Λ_X is the diagonal matrix of singular values (eigenvalues). Note that one can also find the singular value decomposition of the correlation matrix of $\overline{\mathbf{X}}$ as an alternative to the covariance matrix. To determine the rotation dissimilarity between the two data sets $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$, we measure the angles between their principal components.

Since the columns of X and Y are unit vectors, it follows that the diagonal of the matrix $X^T Y$ is the cosine of the angles between the corresponding principal components, and so our rotation dissimilarity measure D_r is defined as the sum of the angles between the components:

$$D_r(\overline{\mathbf{X}}, \overline{\mathbf{Y}}) = trace(cos^{-1}(abs(X^T Y))). \quad (4)$$

Since the signs of the principal components can be ignored, taking the absolute value ensures that we will only be concerned with acute angles. It can be easily shown that if $\overline{\mathbf{X}}$ and $\overline{\mathbf{Y}}$ are n -dimensional data sets, then $D_r(\overline{\mathbf{X}}, \overline{\mathbf{Y}})$ only takes on values in the set $[0, \frac{n\pi}{2}]$, where a value of 0 infers that the principal components are exactly aligned according to

the size of their corresponding eigenvalues, a value of $\frac{n\pi}{2}$ infers that the principal components are completely orthogonal. We note that D_r is very similar to the Eros similarity measure presented in [26]. The central difference is that we take the arc cosine of $X^T Y$ so that D_r measures dissimilarity instead of similarity as Eros does.

Note that the rotation dissimilarity measure D_r also accounts for some aspects of the differences in the covariance structures of $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$, since it measures the amount of rotation needed so that their respective principal components are aligned in order of decreasing variance. However, we still must account for the amount of variance in each direction, or the “shape” of the data sets.

3.1.3 Variance Component

We note that data sets can have different “shapes.” For example, in two dimensions, a data set with little or no correlation between its attributes has a scatter plot that is circular in shape, while the points of a data set with maximum correlation all lie along the same line. It may be the case that the principal components of $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$, are completely aligned, but they still have very different shapes. For example, consider data sets C and E in Figure 1. It will be shown in Section 3.2 that the principal components of C and E are nearly aligned, but it is obvious to see that they have different variance structures by looking at the shapes of their plots: data set C has a short ovular shape, while E is much more elongated.

To account for these differences in the shapes of the data sets, we examine the difference in the distributions of the variance over the principal components of $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$. More formally, consider the random variable $V_{\underline{\mathbf{X}}}$ having the probability mass function:

$$P(V_{\underline{\mathbf{X}}} = i) = \frac{\lambda_i^X}{\text{trace}(\Lambda_X)} \quad (5)$$

where Λ_X is the diagonal matrix of singular values from Equation 3, and λ_i^X is the i th singular value. $P(V_{\underline{\mathbf{X}}} = i)$ is then the proportion of the variance in the direction of the i th principal component. We can then compare the distributions of $V_{\underline{\mathbf{X}}}$ and $V_{\underline{\mathbf{Y}}}$ by finding the symmetric relative entropy:

$$SRE(V_{\underline{\mathbf{X}}}, V_{\underline{\mathbf{Y}}}) = \frac{1}{2}(H(V_{\underline{\mathbf{X}}}\|V_{\underline{\mathbf{Y}}}) + H(V_{\underline{\mathbf{Y}}}\|V_{\underline{\mathbf{X}}})) \quad (6)$$

where $H(X\|Y)$ is the relative entropy of two random variables X and Y . The relative entropy is a common measure of the distance between two prob-

ability distributions [8]. We can then define the variance dissimilarity as the symmetric relative entropy:

$$D_v(\underline{\mathbf{X}}, \underline{\mathbf{Y}}) = SRE(V_{\underline{\mathbf{X}}}, V_{\underline{\mathbf{Y}}}). \quad (7)$$

3.1.4 Final Dissimilarity Metric

The dissimilarity between $\underline{\mathbf{X}}$ and $\underline{\mathbf{Y}}$ can now be defined in two different manners. Our basic formulation is given by:

$$D_{\Pi}(\underline{\mathbf{X}}, \underline{\mathbf{Y}}) = D_d \times D_r \times D_v. \quad (8)$$

A more flexible formulation is as a linear combination of the components, given by:

$$D_{\Sigma}(\underline{\mathbf{X}}, \underline{\mathbf{Y}}) = \beta_0 + \beta_d \times D_d + \beta_r \times D_r + \beta_v \times D_v. \quad (9)$$

This formulation allows the components to be weighted differently (or completely ignored) by means of varying the values of their coefficients (i.e. β). To avoid an unwanted bias towards one or more of the components, the coefficients must chosen to normalize their respective components. This is straightforward for some components (for example D_r only takes on values in the range $[0, \frac{n\pi}{2}]$), but not for others (for example, when using the Euclidean distance for D_d on non-normalized data).

Since the coefficients allow the components to be weighted differently, a user can bias the measure to reflect domain knowledge. For example, D_{Σ} reduces to the basic Euclidean distance between the centroids of the data sets when β_d is set to 1 and the others are set to 0. However, on the other extreme, one may be more concerned with finding data sets with similar covariance structures, but may not be concerned with with relative locations of the data sets, and so β_r and β_v can be set to some positive value, while β_d is set to 0.

3.1.5 Missing Data

Our measure is also robust to missing data. If a data set $\underline{\mathbf{X}}$ has records with missing attribute values, and assuming that the data has a normal distribution, one can use the Expectation-Maximization [10] algorithm to find the maximum-likelihood values of the centroid $\mu_{\underline{\mathbf{X}}}$ and the covariance matrix $cov(\underline{\mathbf{X}})$. The principal components one finds are the sample principal components [19], and one can develop confidence intervals to test the closeness to the true (population) principal components. If the missing data is not excessive, then the maximum likelihood/sample estimates of the components will be accurate, and the computation of the dissimilarity metric can continue as before. Other approaches for

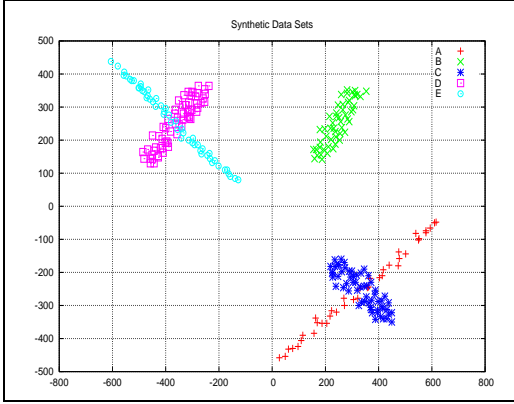


Figure 1. A plot of five synthetic data sets.

	A	B	C	D	E
A	–	511.43	5.3	854.06	867.04
B	511.43	–	512.87	604.31	617.37
C	5.3	512.87	–	858.64	871.64
D	854.06	604.31	858.64	–	13.69
E	867.04	617.37	871.64	13.69	–

Table 1. Dissimilarity: distance component.

handling missing data involve just ignoring records with missing data completely. In Section 4.5 we present results that show simply ignoring missing data does not drastically affect the performance of our measure.

3.2 Example

In this example, we will look at each component in turn to show it influences the final value of the dissimilarity measure. Consider Figure 1, where we have plotted five different data sets labeled A through E. Each data set is similar to the others in different ways. For example, sets A and E have similar shapes, D and E have similar centroids, and B and D have similar slopes.

In Table 1 we present the pairwise distance dissimilarities of the data sets. The bold face values represent the minimal dissimilarities between data sets. As we expect, data sets A and C are considerably similar to each other according to this measure, as are data sets D and E, while data set B is considerably dissimilar from all the other data sets. Note that while data sets A and C have similar means, they have extremely different covariance structures that are not taken into account by this measure.

In Table 2 we present the pairwise rotation dis-

	A	B	C	D	E
A	0	0.67	2.53	0.4	2.55
B	0.67	0	3.09	0.27	3.06
C	2.53	3.09	0	2.93	0.02
D	0.4	0.27	2.93	0	2.95
E	2.55	3.06	0.02	2.95	0

Table 2. Rotation dissimilarity.

	A	B	C	D	E
A	0	0.18	0.20	0.10	0.000009
B	0.18	0	0.0007	0.01	0.18
C	0.20	0.0007	0	0.01	0.21
D	0.096	0.009	0.014	0	0.099
E	0.000009	0.18	0.21	0.10	0

Table 3. Variance dissimilarity.

similarities of the data sets. As we expect, data sets A, B, and D are very similar to each other, since their principal components are pointed in nearly the same directions. We note that the most similar pair of data sets according to this measure is E and C, while according to the distance dissimilarity measure they are the most dissimilar pair of data sets.

In Table 3, we present the pairwise variance dissimilarities. In this case, data sets A and E are very similar to each other, which is expected, since the plots of each are both long and thin. We also note that while data sets B and C are very similar to each other according to the variance dissimilarity measure, they are also the most dissimilar pair according to the rotation dissimilarity measure.

In Table 4, we present the total pairwise dissimilarity of the data sets. In this case we use the product form (D_{Π}) of our measure. We find that data sets A and E are the most similar, due to the high similarity of the distribution of their variances across their principal components. Data set E is next most similar to data set D due to the proximity of their means, and E is also quite similar to data set C, since their principal components are rotated similarly. E is most dissimilar to data set

	A	B	C	D	E
A	0	60.07	2.73	32.94	0.02
B	60.07	0	1.14	1.38	341.14
C	2.73	1.14	0	36.07	4.52
D	32.94	1.38	36.07	0	3.99
E	0.02	341.14	4.52	3.99	0

Table 4. Total dissimilarity (D_{Π}).

```

procedure FindChangePoints(series  $T$ , int  $W_1$ , int  $W_2$ )
begin
  for each point  $t \in T$ 
     $Before = \{\text{the } W_1 \text{ points occurring before } t\}$ 
     $After = \{t\} \cup \{\text{the } W_2 - 1 \text{ points occurring after } t\}$ 
     $Score[t] = D(Before, After)$ 
  end
  Filter  $Score$  to find maxima
  Return the  $t$  corresponding to maxima of  $Score$ 
end.

```

Figure 2. The change point detection algorithm.

B due to large differences in their respective means, rotations, and variances. However, a basic distance-based dissimilarity measure (for example, using just D_d) would rank B as the second-most similar data set to E (after D), as can be seen from Table 1.

3.3 Applications

In this section we present an overview of how our dissimilarity measure can be used in several common data analysis techniques. The techniques we consider are change point detection, anomaly detection, and data set clustering.

3.3.1 Change Point Detection

One application of our dissimilarity measure is change point detection. In change point detection, one wants to find the point(s) in a time series where there has been an abrupt change in the process generating the series [5]. Our algorithm for off-line change point detection for multivariate time series is presented in Figure 2. It works by scanning over a time series T , comparing two successive windows of data points, the first of size W_1 and the second of size W_2 data points, using our dissimilarity measure D . It returns the maxima of D applied over T . It follows that the maximum value of D is achieved when the two successive windows are most different with respect to their means, rotations, or variances, signaling that the underlying distribution generating the time series has changed between the two windows. We present the experimental results of running change point detection on stock market data in Section 4.2.

3.3.2 Anomaly Detection

A closely related problem to change point detection is anomaly detection. Whereas change point detec-

tion seeks to discover points that mark a shift from one generating process to another, anomaly detection seeks to discover points that are outliers with regard to the current generating process. Outlier detection algorithms work by assigning an anomaly score to each point in a data set based on its dissimilarity to the other points in the set. The most dissimilar ones are marked as outliers. Since our measure is designed to measure the dissimilarity between a pair of data sets, we cannot directly measure the dissimilarity between a point and a data set. However, we can use our measure to assign an anomaly score to a point:

$$S_{\bar{\mathbf{X}}}(x) = D(\bar{\mathbf{X}}, \bar{\mathbf{X}} - x). \quad (10)$$

The anomaly score function $S_{\bar{\mathbf{X}}}(x)$ measures how much the mean and covariance structure of $\bar{\mathbf{X}}$ would change if the data point x was removed. If the value of $S_{\bar{\mathbf{X}}}(x)$ is large, then x must be introducing considerable distortion into the model of $\bar{\mathbf{X}}$.

We demonstrate the utility of our dissimilarity measures for outlier detection using the above approach with a toy data set. We compare our measures to the basic Mahalanobis distance metric, since it also incorporates information concerning the covariance structure of the data set (similar to the formulation in Equation 10, we calculate the distance from a point x to the centroid of $\bar{\mathbf{X}} - x$ using the covariance matrix of $\bar{\mathbf{X}} - x$):

$$S_{\bar{\mathbf{X}}}(x) = (\mu_{\bar{\mathbf{X}}-x} - x) \Sigma_{\bar{\mathbf{X}}-x}^{-1} (\mu_{\bar{\mathbf{X}}-x} - x)^T. \quad (11)$$

Our data set contains 150 points, and we find the top 15 outliers according to each measure. The results can be seen in Figure 3. In these plots, normal points are denoted by pluses and outliers are denoted by stars. In Figure 3(A) we show the outliers discovered using the Mahalanobis distance metric. In Figures 3(B) and (C) we show the outliers discovered using our D_{Π} and D_{Σ} measures respectively (in the case of D_{Σ} , we have chosen the β 's so that the components are normalized). As we expect, the results are quite similar, since they all take into account both the means and the covariances of the data. However, unlike the Mahalanobis distance metric and D_{Π} measure, the D_{Σ} measure is much more flexible, as the user is able to choose the values of the β 's. This flexibility is demonstrated in Figures 3(D)-(F), where we detect outliers using only the distance, rotation, and variance components respectively by setting the coefficient of the relative component to 1 and the others to 0. In each case, different outliers are found. For example, using the

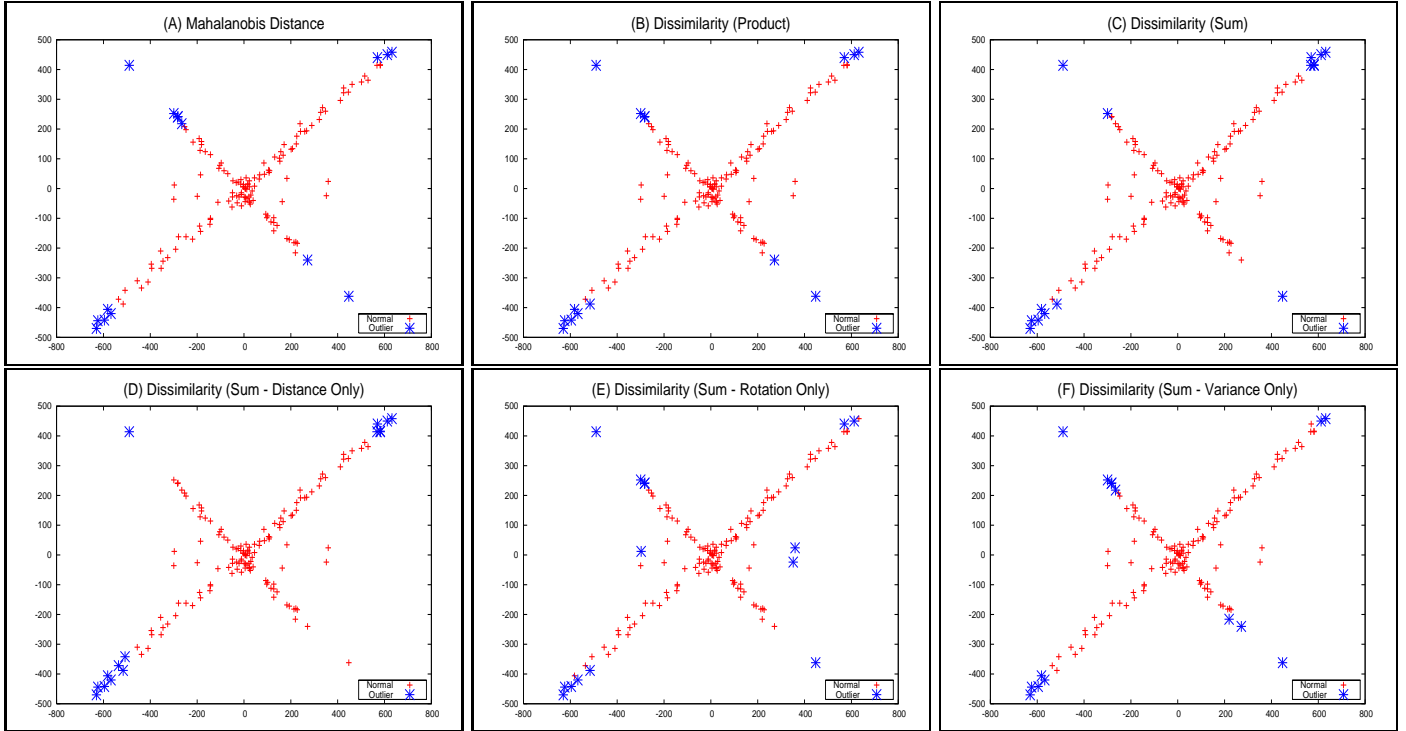


Figure 3. Outliers in a data set discovered using different measures: (A) Mahalanobis; (B) D_{Π} ; (C) D_{Σ} ; (D) D_{Σ} (Euclidean) distance component only; (E) D_{Σ} rotation component only; (F) D_{Σ} variance component only.

distance component only (Figure 3(D)), the outliers are those points on the extreme ends of the “arms” of the data set, whereas when we use the rotation component only (Figure 3(E)), the points not belonging to any of the “arms” are marked as outliers.

One can also use an alternative incremental form of anomaly detection that is applicable in domains where data sets are streaming or in the form of time series. In this form, one calculates $D(\overline{\mathbf{X}}, \overline{\mathbf{X}} \cup \{x\})$, where $\overline{\mathbf{X}}$ is a sliding window of k data points, and x is the first data point following the window. This is similar to change point detection, except that it only concerns information that arrives prior to x . We present experimental results of using this approach with our dissimilarity measure on stock market data in Section 4.3.

3.3.3 Data Set Clustering

One of the advantages of a dissimilarity measure for data sets is that it allows one to cluster the data sets into groups with similar means or variances, depending on how one weights the components. As a motivating example, consider a large business organization such as Wal-Mart, with national or interna-

tional interests. Such organizations usually rely on a homogeneous distributed databases to store their transaction data. This leads to time varying, distributed data sources. In order to analyze such a collection of databases, it seems important to cluster them into small number of groups to contrast global trends with local trends so as to develop advertising campaigns targeted at specific clusters.

It is straightforward to perform agglomerative hierarchical clustering of data sets using our dissimilarity measure. If one has n data sets, one can construct an n by n table containing the pairwise dissimilarities of the data sets. Once this table has been constructed, one can use any distance metric (e.g. single-link or complete-link) to perform the hierarchical clustering. We present experimental results on using hierarchical clustering for stock market data in Section 4.4. This table also facilitates non-hierarchical clustering approaches, such as the k-medoid approach [14]. This works by selecting several data sets at random to be medoids of the clusters, and then assigning the remaining data sets to a cluster with the most similar medoid. After this phase, the medoids are checked to see if replac-

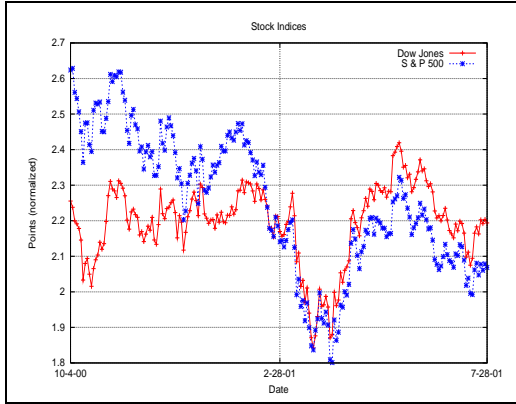


Figure 4. Plot of stock indices centered on change point.

ing any of them with other data sets would reduce the dissimilarity in their respective clusters. If so, the process repeats until no medoids are replaced, or some other stopping criterion is met.

4 Experimental Results

4.1 Setup

In our experiments we utilize historical stock market data available from Yahoo! Finance [1]. We constructed several multivariate time series data sets, where each dimension is the adjusted closing price of a stock or stock index. The stock indices that we use are the Dow Jones (DJ), Standard and Poor’s 500 (S&P 500), and the 10-year Treasury Note (TN) indices from January 1962 until May 2005. We also used the stock prices of a set of six pharmaceutical companies (Abbott Laboratories, GlaxoSmithKline, Johnson and Johnson (J & J), Merck, Pfizer, and Wyeth) from August 1986 until June 2005. All of our implementations are done using Octave, an open-source version of Matlab.

4.2 Change Point Detection

In our first set of experiments, we examined our measure’s effectiveness when used for change point detection. One of our more impressive results comes from a bivariate data set containing the values of the Dow Jones and S&P 500 indices. In this experiment we normalized the data and set the window sizes W_1 and W_2 both equal to 100. We derived the principal components from the covariance matrix of the data. We eliminated the scores of the first and last

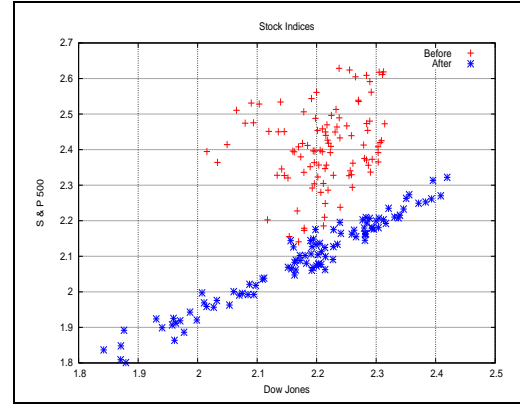


Figure 5. Scatter plot of stock indices.

months in order to avoid edge effects. The highest-scoring change point according to our dissimilarity metric (D_{II}) occurred on February 28, 2001. In Figure 4 we plot these two indices versus time, showing 100 points on either side of the change point. From this figure we can see that the indices become more highly correlated after the change point. The difference is more obvious in Figure 5. Here the values of the indices are plotted against each other, and the markers indicate whether the point comes from before or after the change point. As can be seen, the points fall into two distinct clusters depending on whether they come before or after the change point. We note that when we perform SVD using the correlation matrices instead of covariance matrices of the data, the results are very similar, though the change points may be shifted by a few instances. For the example above, when we use the correlation matrix, we calculate the change point as February 23, 2001.

4.3 Anomaly Detection

We test our incremental outlier detection algorithm on several data sets: *Indices*, which contains all three stock indices (DJ, S&P 500, and TN); *DJ/S&P 500* and *DJ/TN*, which contain only the two relevant indices; *Pharm.*, which contains all six pharmaceutical stocks (see Section 4.1); and *Pfizer/Merck*, which contains only the two relevant indices. In our experiment we use D_{II} , performing SVD on the covariance matrices, and vary the value of k (the size of the sliding window) over 12 different values (4, 5, 6, 8, 10, 15, 20, 30, 40, 60, 80, 100), and mark the dates of the top 30 outliers for each value of k , creating 12 lists of 30 dates each.

In Figure 6 we plot Pfizer’s and Merck’s stock prices during the year 2004. The vertical lines mark

Date	Description	Indices	DJ/S&P 500	DJ/TN	Pharm.	Pfizer/Merck
10-19-87	Market crash	92% (100%)	25% (58%)	92% (92%)	17% (75%)	25% (83%)
3-16-00	Largest DJ increase	0% (0%)	50% (0%)	8% (0%)	17% (0%)	0% (0%)
4-14-00	Largest DJ decrease	33% (8%)	58% (25%)	50% (17%)	0% (0%)	0% (0%)
9-17-01	WTC attack	8% (58%)	25% (67%)	42% (33%)	0% (0%)	0% (0%)
9-30-04	Vioxx™ warning	0% (0%)	0% (0%)	0% (0%)	58% (100%)	92% (100%)
12-17-04	Celebrex™ warning	0% (0%)	0% (0%)	0% (0%)	8% (0%)	75% (33%)

Table 5. Detection rates of notable outliers using the D_{Π} measure and a Mahalanobis metric-based D_{Σ} measure (in parenthesis).

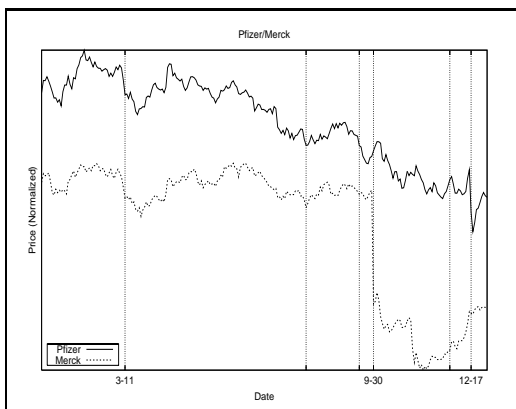


Figure 6. Outliers in 2004 Merck and Pfizer stock prices.

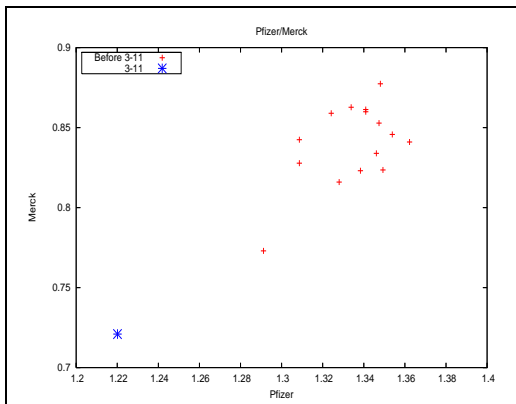


Figure 7. The Pfizer/Merck March 11 outlier and preceding days.

the days of the outliers. 2004 was tumultuous year for these stocks, as it contains six of the top 30 outliers according to our measure when we set k equal to 15. Note that our measure is able to detect large changes in the means, as is the case for the September 30 and December 17 outliers, as well as more subtle outliers, such as the one occurring on March 11. In Figure 7 we show a scatter plot of Pfizer and Merck stock prices for March 11 and the 15 trading days preceding it. This clarifies why March 11 is marked as an outlier: In the 15 trading days prior to March 11, the Pfizer's and Merck's stock prices were relatively uncorrelated, but on March 11, the prices of both sank sharply.

We also verify that our dissimilarity measures can detect known anomalies. To do this we search the 12 lists of outliers for several well-known dates. For example, we pick October 19, 1987, since the stock market suffered a major crash on that day, and March 16, 2000, as that day currently holds the record for the largest increase of the Dow Jones index. We also pick September 30, 2004 and December 17, 2004, as those are the days when information was announced concerning serious side-effects of Merck's Vioxx™ and Pfizer's Celebrex™ drugs, respectively¹. We compare the basic D_{Π} measure against the D_{Σ} measure that uses the Mahalanobis metric for the distance component (see equation 2).

In Table 5 we present our results as the percentage of the lists in which each date appeared for all of the data sets for both the D_{Π} measure and the Mahalanobis-based D_{Σ} measure (which is given in parenthesis). The measures discern these anomalous days fairly well. The first four rows indicate anomalous days for the overall stock market, and the anomalies are reflected in the market index data sets. The last two rows indicate anomalous days for the Pharmaceutical sector, as the announcement

¹Vioxx is a trademark of Merck and Company, Incorporated. Celebrex is a trademark of Pfizer, Incorporated.

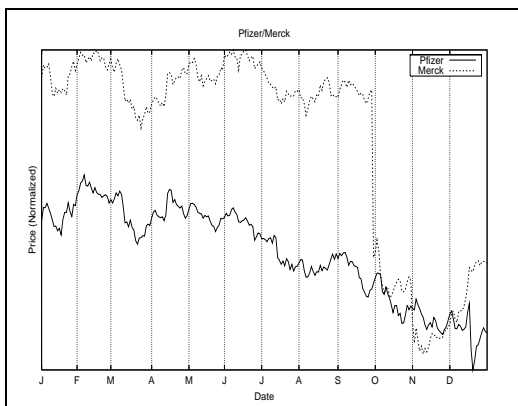


Figure 8. Pfizer/Merck 2004 stock prices.

concerning VioxxTM and CelebrexTM had adverse effects on the price of Merck’s and Pfizer’s stocks, respectively. However, the effect of the announcements was not so great on the values of the market indices.

We note that in some cases, the Mahalanobis-based D_{Σ} approach out-performs the basic D_{Π} measure, but in other cases, the D_{Π} measure out-performs the Mahalanobis-based approach. For example, the Mahalanobis-based approach more consistently detects the October 19, 1987 market crash and the September 30, 2004 VioxxTM announcement, while the D_{Π} measure more consistently detects the largest Dow Jones increase (March 16, 2000) and decrease (April 14, 2000). The reason is manner in which the two approaches detect anomalies. The Mahalanobis-based approach is biased towards the Mahalanobis distance from stock prices on the current day to the mean of the prices on the previous k days. Therefore, it is good at detected larges changes in mean. The D_{Π} measure, however, also detects changes in the correlations. The Dow Jones anomalies involve a large change in the mean value of the Dow Jones index, but this is not as drastic as the change in correlation that results when it is paired with other indices that do not have such large changes in mean value.

4.4 Data Set Clustering

In our next experiment, we examine the effects of using our dissimilarity measure to perform agglomerative hierarchical clustering. Note, as mentioned earlier, one can also use k-medoids clustering here. Currently, we use the *Pfizer/Merck* data set, and extract the records for each month during the year 2004 to form 12 separate data sets (see Figure 8).

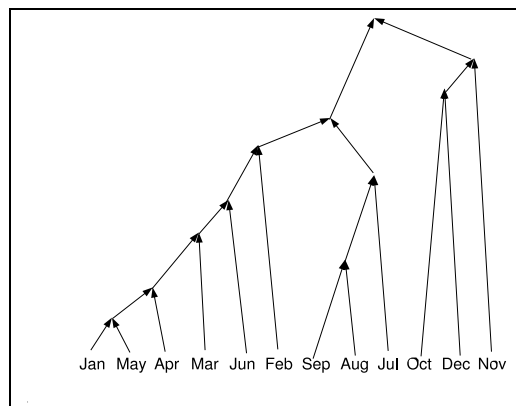


Figure 9. Dendrogram resulting from clustering of monthly data sets.

Our original intent was to demonstrate the effectiveness of this approach on clustering clinical trial patient data from Pfizer, Incorporated, and use it as a mechanism for detecting hepatotoxicity signals. However, due to delays in getting permission to publish results from this data, we are unable to include these results at this point in time.

We build a table of pair-wise dissimilarities between the monthly *Pfizer/Merck* data sets using the D_{Σ} measure, with a slight bias towards the distance component to account for the drop in stock prices in the latter part of the year. Using this table, we perform hierarchical clustering using the single-link distance metric. The dendrogram resulting from the clustering can be seen in Figure 9. The results are expected: the data sets for January through June are clustered early, as they have similar means and positive correlations, and the data sets for October through December are not clustered together until very near the end, due to their large differences in means compared to the other months. We see that October and December cluster with each other first, which is notable since they are months most influenced by the VioxxTM and CelebrexTM announcements, respectively.

4.5 Robustness to Missing Data

Finally, we examine the effects of ignoring records containing missing data. In this experiment we used the *DJ/S&P 500* data set, and progressively removed 1%, 5%, 10%, 15%, and 20% of the records (i.e. that data set with 20% of the records removed is a subset of the data set with 15% of the records removed, and so on). For each of these data sets, we calculated the top 20 change points using the algo-



Figure 10. Percentage of change points found for differing degrees of missing data.

algorithm in Section 3.3.1, with W equal to 15, 40, and 100. We then compared each set of 20 change points to the top 20 change points found when there is no missing data. We counted the number of change points that matched to within some time window (on the order of one week for W equal to 15, and one month for W equal to 100). The percentage of correct matches for each data set and each value of W is presented in Figure 10. Our detection rates from a high of 100% (all change points found) for 1% missing data to a low of 70% when 15% to 20% of the data is missing. Note that in this experiment we assume we do not know which records are missing—we calculate the change point based on the W non-missing records coming before the point, and the W non-missing records coming after it. Therefore, the change point is calculated using only a subset of the records used if there was no missing data, plus some “extra” records would not be considered if there were no missing data. This fact, coupled with our detection rates, indicates that our approach is fairly robust missing data.

5 Conclusion

In this paper we presented a dissimilarity measure for data sets that takes into account the means and correlation structures of the data sets. This dissimilarity measure is tunable, allowing the user to adjust its parameters based on domain knowledge. The measure has many different applications for time series analysis, including change point detection, anomaly detection, and clustering, and our experimental results on time series data sets show its effectiveness in these areas. In future we want use

our dissimilarity measure to detect anomalous data sets. This is applicable to clinical trial data, where patients are represented by a multivariate time series of blood analyte values, and detection of anomalous patients can lead to early discovery of possibly serious side effects of the drug being tested. We have conducted experiments on this, and our results are extremely promising. However, at this point we do not have permission to share these results. We also plan to explore the incremental aspects of our measure in order to apply to dynamic and streaming data sets. For example, the computational costs of calculating our dissimilarity measure on dynamic or streaming can be reduced by using incremental PCA techniques [4, 13]. Such incremental techniques can also enhance execution speeds performing anomaly detection and change point detection off-line, where sliding windows are used to scan the data.

References

- [1] Yahoo! finance. In <http://finance.yahoo.com>.
- [2] C. C. Aggarwal. Towards systematic design of distance functions for data mining applications. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 9–19, August 2003.
- [3] R. Agrawal, C. Faloutsos, and A. N. Swami. Efficient similarity search in sequence databases. In *FODO '93: Proceedings of the 4th International Conference on Foundations of Data Organization and Algorithms*, pages 69–84. Springer-Verlag, 1993.
- [4] M. Artac, M. Jogan, and A. Leonardis. Incremental pca or on-line visual learning and recognition. In *ICPR*, 2002.
- [5] M. Basseville and I. V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice Hall, 1993.
- [6] S. D. Bay and M. J. Pazzani. Detecting change in categorical data: Mining contrast sets. In *Knowledge Discovery and Data Mining*, pages 302–306, 1999.
- [7] C. Bohm, K. Kailing, P. Kroger, and A. Zimek. Computing clusters of correlation connected objects. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, pages 455–466. ACM Press, 2004.
- [8] T. M. Cover and J. A. Thomas. *Elements of Information Theory*. John Wiley and Sons, 1991.
- [9] G. Das, H. Mannila, and P. Ronkainen. Similarity of attributes by external probes. In *Knowledge Discovery and Data Mining*, pages 23–29, 1998.
- [10] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, B39:1–38, 1977.

- [11] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In E. Simoudis, J. Han, and U. Fayyad, editors, *Second International Conference on Knowledge Discovery and Data Mining*, pages 226–231, Portland, Oregon, 1996. AAAI Press.
- [12] R. Goldman, N. Shivakumar, S. Venkatasubramanian, and H. Garcia-Molina. Proximity search in databases. In *Proc. 24th Int. Conf. Very Large Data Bases, VLDB*, pages 26–37, 24–27 1998.
- [13] P. M. Hall, D. Marshall, and R. R. Martin. Incremental eigenanalysis for classification. In *BMVC*, May 1998.
- [14] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, 2001.
- [15] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2001.
- [16] D. P. Huttenlocher, G. A. Klanderman, and W. A. Rucklidge. Comparing images using the hausdorff distance. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(9):850–863, 1993.
- [17] H. V. Jagadish, A. O. Mendelzon, and T. Milo. Similarity-based queries. In *PODS '95: Proceedings of the fourteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems*, pages 36–45, New York, NY, USA, 1995. ACM Press.
- [18] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.
- [19] R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis*. Prentice Hall, fifth edition, 2002.
- [20] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.
- [21] S. Parthasarathy and M. Ogihara. Clustering distributed homogeneous datasets. In *PKDD '00: Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 566–574, London, UK, 2000. Springer-Verlag.
- [22] J. C. Principe, N. R. Euliano, and W. C. Lefebvre. *Neural and Adaptive Systems: Fundamentals through Simulations*. John Wiley and Sons, 2000.
- [23] R. Reymont and K. G. Joreskog. *Applied Factor Analysis in the Natural Sciences*. Cambridge University Press, 1996.
- [24] R. Subramonian. Defining diff as a data mining primitive. In *KDD 1998*, 1998.
- [25] J. Yang, W. Wang, H. Wang, and P. Yu. δ -clusters: Capturing subspace correlation in a large data set. In *Proceedings of the 18th International Conference on Data Engineering (ICDE'02)*, page 517. IEEE Computer Society, 2002.
- [26] K. Yang and C. Shahabi. A pca-based similarity measure for multivariate time series. In *MMDB '04: Proceedings of the 2nd ACM international workshop on Multimedia databases*, pages 65–74, New York, NY, USA, 2004. ACM Press.