CHAPTER III

SIMILARITY ANALYSIS IN TIME SERIES DATA MINING

3.1 INTRODUCTION

In this chapter, similarity search, for a given query sequence, all the sequences in the database that are similar to the given query are found. For the past two decades, Laxman and Sastry (2006) introduced similarity search and retrieval of data in the time series database has received a considerable attention by the researchers, because of its widespread applications in several fields. The domain where similarity search is useful includes the following:

Meteorologist frequently come across a situation where needs to find those periods in a given time series, that have similar weather conditions as of the given query period. Medical practitioners may be interested to know the details about the patients who have similar electrocardiogram (ECG) of another patient. Durga Toshniwal and Joshi (2005) searching for similar patterns in a set of time series helps us to perform data mining activities such as prediction, classification, hypothesis testing, retrieval, indexing, change detection, frequent pattern mining, and segmentation of time series. Similarity can be
computed by measuring the distance between the two sequences. To quantify the similarity, various similarity measures viz. Euclidean distance, Minkowski distance, Manhattan distance, edit distance, etc have been used. Some of the challenges in similarity search are: selection of suitable similarity measure, high dimensionality, false dismissal owing to data approximation, and selection of index structure.

Some of the vital challenges in similarity search are:

- Identifying suitable similarity measure.
- High dimensionality.
- Identifying Index structure.

In real world applications, the time series will have highly voluminous data. In this work, stock market data, collected from www.nseindia.co.in. The study variable here is the closing price related to the two reputed companies such as TCS and WIPRO. The data specifies the daily stocks from Jan 2009- Dec 2010. In order to process the entire range of values in finding the similarity will lead to increase in computational time, the dimensionality reduction techniques are applied. There are various techniques viz. Piecewise Constant Approximation (PCA), Piecewise Linear Approximation (PLA), Piecewise Vector Quantization (PVQ), derivative time series segment approximation (DSA), signature extraction and data transformation used in reducing data dimensions.
3.2 NOTATIONS AND DEFINITIONS

This section provides various concepts, notations and definitions related to the proposed work.

Let $T$ be the original time series with $n$ time stamped feature values observed over the time. Then the time series of feature $T$ is represented as $T = t_1, t_2, ..., t_n$, in real time, the time series data set contains huge volume of data that is not suitable for further processing. The given series needs to be approximated and the approximated form of the original time series is $T' = t'_1, t'_2, ..., t'_n$. The other notations are $N$ number of time series in the dataset, $D$ the Euclidean distance between two series. The important definitions related to the work are discussed below.

3.3 SIMILARITY SEARCH IN TIME SERIES ANALYSIS

Similarity search finds data sequences that differ only slightly from the given query sequence, given a set of time series sequences, $S$, there are two types of similarity searches:

(i) Subsequence matching.

(ii) Whole sequence matching.

Subsequence matching

Subsequence matching finds the sequences in $S$ that contain subsequences that are similar to a given query sequence $x$.

Whole sequence matching
Whole sequence matching finds a set of sequences in S that are similar to each other. In this work, whole sequence matching is applied.

**VECTOR QUANTIZATION**

Vector quantization is a classical quantization technique from signal processing which allows the modeling of probability density functions by the distribution of prototype vectors. It was originally used for data compression. It works by dividing a large set of points (vectors) into groups having approximately the same number of points closest to them. Each group is represented by its centroid point, as in k-means and some other clustering algorithms.

### 3.4 INDEXING METHODS FOR SIMILARITY SEARCH

When a similarity query is submitted to the system, the index can be used to retrieve the sequences that are at most a certain small distance away from the query sequence. Post-processing is then performed by computing the actual distance between sequences in the time domain and discarding any false matches.

Various kinds of indexing methods have been explored to speed up the similarity search. For example, R-trees and R’-trees have been used to store minimum bounding rectangles.
DIMENSIONALITY REDUCTION

Derivative time series segment approximation (DSA) representation model, which originally features derivative estimation, segmentation and segment approximation to provide both high sensitivity in capturing the main trends of time series and data compression. We extensively compare DSA with state-of-the-art similarity methods and dimensionality reduction technique in clustering and classification frameworks. Experimental evidence from effectiveness and efficiency tests on various data sets shows that DSA is well-suited to support both accurate and fast similarity detection. However, DSA effected by it extremity. Therefore, DSA value along with coefficient of variation technique is used to approximate the time series.

3.5 COEFFICIENT OF VARIATION

The Coefficient of Variation (CV) is a unit – free measure. It is always expressed in percentage. The CV will be small if the variation is small. Among the two groups, the one with less CV is said to be more consistent. CV is calculated using equation (3.1).

\[
CV = \frac{\sigma}{\mu} \times 100 \quad \text{... (3.1)}
\]

The CV is unreliable if the mean is near zero. The CV is informative if it is given along with the mean and standard deviation. Otherwise it may be misleading.
3.6 DATA MATRIX

This represents $n$ objects, such as persons, with $p$ variables (also called measurements or attributes). Such as age, height, weight, gender, race, and so on. The structure is in the form of relatial table, or $n$-by-$p$ matrix ($n$ objects $X_p$ variables).

$$
\begin{array}{cccc}
  x_{11} & \cdots & x_{1p} \\
  \vdots & \ddots & \vdots \\
  x_{n1} & \cdots & x_{np} \\
\end{array}
$$

... (3.2)

3.7 DISSIMILARITY MATRIX

This stores a collection of proximities that are available for all pairs of $n$ objects. It is often represented by a $n \times n$ table as shown below.

$$
\begin{bmatrix}
  0 & 0 \\
  d(2,1) & 0 \\
  d(3,1) & d(3,2) & 0 \\
  \vdots & \vdots & \vdots \\
  d(n,1) & d(n,2) & \cdots & \cdots & 0 \\
\end{bmatrix}
$$

... (3.3)

where $d(i, j)$ measured difference or dissimilarity between is objects $i$ and $j$. Since $d(i, j) = d(j, i)$ and $d(i, i) = 0$, we have the matrix in (3.3). Measures of dissimilarity are discussed throughout this section.

The data matrix is often called a two-mode matrix, where as the dissimilarity matrix is called as one-mode matrix, since the rows and columns of the former represent different entities, while that of the
latter represent the same entity. Many clustering algorithms operate on a dissimilarity matrix. If the data are presented in the form of a data matrix, it can be first transformed into a dissimilarity matrix before applying such clustering algorithms.

3.8 SIMILARITY MEASURES

The distances are normally used to measure the similarity or dissimilarity between two data objects. Some of the distance measures are, Minkowski, Euclidean, Squared Euclidean, Manhattan and Chebyshev.

MINKOWSKI DISTANCE

Minkowski distance is a generalization of both Euclidean distance and Manhattan distance. It is defined as:

$$d(i, j) = \left( \sum_{k=1}^{n} |x_{ik} - x_{jk}|^q \right)^{\frac{1}{q}} \quad \ldots (3.4)$$

where q is a positive integer. The Minkowski distance is a metric as a result of the Minkowski inequality.

EUCLIDEAN DISTANCE

Euclidean distance or Euclidean metric is the "ordinary" distance between two points that one would measure with a ruler, and is given by the Pythagorean formula. By using this formula as distance, Euclidean space (or even any inner product space) becomes a metric space. The associated norm is called the Euclidean norm. The
Euclidean distance between two data points involves computing the square root of the sum of the squares of the differences between corresponding values. It represents the Euclidean distance when \( q = 2 \).

\[
d(i, j) = \sqrt{\sum_{k=1}^{n} (x_{ik} - x_{jk})^2} \quad \ldots \ (3.5)
\]

where \( i = (x_{i1}, x_{i2}, \ldots, x_{ip}) \), and \( j = (x_{j1}, x_{j2}, \ldots, x_{jp}) \), are two \( p \)-dimensional data objects.

**SQUARED EUCLIDEAN DISTANCE**

The Euclidean Squared distance metric uses the same equation as the Euclidean distance metric, but does not take the square root. As a result, clustering with the Euclidean Squared distance metric is faster than clustering with the regular Euclidean distance.

\[
d(i, j) = \sum_{k=1}^{n} (x_{ik} - x_{jk})^2 \quad \ldots \ (3.6)
\]

**MANHATTAN DISTANCE**

It is also known as Manhattan distance, and absolute value distance or \( L_1 \) distance. It represents distance between points in a city road grid. It examines the absolute difference between coordinates of a pair of objects. City block distance is computed using equation. when \( q = 1 \),

\[
d(i, j) = \sum_{k=1}^{n} |x_{ik} - x_{jk}| \quad \ldots \ (3.7)
\]
CHEBYSHEV DISTANCE

In mathematics, Chebyshev distance (or Tchebyshev distance), maximum metric, or $L_\infty$ metric is a metric defined on a vector space where the distance between two vectors is the greatest of their differences along any coordinate dimension. The Chebyshev distance between two vectors is

$$d_{ij} = \max_k |x_{ik} - x_{jk}|$$

... (3.8)

The above distance satisfies the following mathematic requirements of a distance function:

1. $d(i, j) \geq 0$ This states that distance is a nonnegative number;
2. This states that the distance of an object to itself is 0;
3. $d(i, j) = d(j, i)$ This states that distance is a symmetric function; and
4. $d(i, j) \leq d(i, h) + d(h, j)$: This is a triangular inequality which states that going directly from point i to point j is no more than making a detour over any other point h.

3.9 CLUSTER ANALYSIS

The cluster analysis is the process of grouping a set objects into classes or clusters so that objects within a cluster have similar in comparison to one another, but are dissimilar to objects in other clusters. Several clustering techniques have been proposed over the
years. They are (i) Hierarchical Vs Partitional, (ii) Divisive Vs agglomerative and (iii) Incremental Vs Non-incremental.

3.9.1 CLUSTER ANALYSIS IN TIME SERIES

Time series clustering has become an important topic, particularly for similarity search amongst long time series such as stock market, financial time series, image processing, bioinformatics etc. There are two main categories in time series clustering as summarized by Keogh et al. (2003). “Whole clustering” is the clustering performed on many individual time series to group similar series into clusters. “Subsequence clustering” is based on sliding window extractions of a single time series and aims to find similarity and differences among different time windows of a single time series.

Clustering is an unsupervised learning process for grouping a dataset into subgroups. If two time series are similar, then the resultant clusters of the both the time series will be same. Cluster analysis is a technique used for combining observations into groups such that:

i. Each group is homogeneous with respect to certain characteristics, i.e. Observations in each group are similar to each other.

ii. Each group should be different from other groups with respect to the characteristics, i.e. observations of one group should be different from the observations of other groups.
A data stream is an ordered sequence of points $x_1, x_2, \ldots, x_n$. These data can be read or accessed only once or a small number of times. A time series is a sequence of real numbers, each number indicating a value at a time point. Xiang Lian et al. in (2008) proposed that all types of time series data applications needs an efficient and effective similarity search over stream data is essential. Hirano et al. (2007) proposed an algorithm for clustering the time series medical data. Their method represents the time series of test results as trajectories in multidimensional space, and compares their structural similarity by using the multi scale (stage) comparison technique. It enables us to find the part-to-part correspondences between two trajectories, taking into account the relationships between different tests. The resultant distinction can be further used with clustering algorithms for finding the groups of similar cases. The method was applied to the cluster analysis of Albumin-Platelet data in the chronic hepatitis dataset.

A good clustering method will produce high quality clusters with intra-cluster distance are minimized and inter cluster distance are maximized. The objective of cluster analysis is to group the observations into clusters such that each cluster is as homogenous as possible with respect to the clustering variables.

The various steps in cluster analysis are:

- Select a measure of similarity.
- Decide the clustering technique.
- Select the type of clustering method.
There are several kinds of methods to obtain clusters. In general, major clustering methods can be classified into the following categories.

- Partitioning methods
- Hierarchical methods
- Density based methods
- Grid based methods
- Model based methods

The quality of a clustering result depends on both the similarity measure used by the method and its implementation. The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns

### 3.9.2 PARTITIONING METHODS

Given a database of n objects or data tuples, a partitioning method constructs k partitions of the data, where each partition represents a cluster and k \( \leq \) n.

It classifies the data into k groups, which together satisfy the following requirements.

i. Each group must contain at least one object.

ii. Each object must belong to exactly one group.

The general criterion of a good partitioning is that objects in the same cluster are “close” or related to each other, whereas objects of
different clusters are “far apart” or very different. Nonhierarchical, each instance is placed in exactly one of k non-overlapping clusters. Since only one set of clusters is output, the user normally has to input the desired number of clusters k.

The partitioned techniques usually produce clusters by optimizing a criterion function defined either locally or globally. A global criterion, such as the Euclidean square-error measure, represents each cluster by a prototype or centroid and assigns the samples to clusters according to the most similar prototypes. A local criterion, such as the minimal mutual neighbour distance, forms clusters by utilizing the local structure or context in the data. The most commonly used partitional clustering strategy is based on the square error criterion. The general objective is to obtain the partition that, for a fixed number of clusters, minimizes the total square error.

3.9.3 K-MEANS CLUSTER

It was developed in 1967 by MacQueen and then modified in 1975 by Hartigan and Wang. The k-means assigns each point to the cluster whose center also called centroid is nearest. The center is the average of all the points in the cluster, that is, its coordinates are the arithmetic mean for each dimension separately over all the points in the cluster.

K-Means algorithm is a partitioning, prototype based and supervised clustering algorithm. The input of k-means algorithm is a dataset consisting of n vectors and it outputs k-clusters which
together form a mutually exclusive and exhaustive partitioning of the data set. The k-means algorithm expects the user to specify k, namely the number of clusters to be formed; this is the main reason for classifying, k-means algorithm as a supervised clustering algorithm.

The k-means clustering technique is very simple and we immediately begin with a description of the k-means.

The basic steps are

i. Choose the number of clusters, \( k \).

ii. Randomly generate \( k \) clusters and determine the cluster centers, or directly generate \( k \) random points as cluster centers.

iii. Assign each point to the nearest cluster center, where "nearest" is defined with respect to one of the distance measures discussed above.

iv. Re-compute the new cluster centers.

v. Repeat the two previous steps until some convergence criterion is met (usually that the assignment hasn't changed).

The step-wise details of k-means algorithm are as follows.

**Step1:** Randomly take a set \( k \) vectors each consisting of \( p \)-components as initial centroids.

**Step2:** Compute the distance between each object and the \( k \)-vectors determined in step 1 and assign them to the cluster corresponding to minimum distance.

**Step3:** Re-compute the cluster centroids using the formula

\[ \text{centroid} = \frac{1}{n} \sum_{i=1}^{n} x_i \]
\[ m_i = \sum_{j \in C_i} X_j \left| C_i \right|, i = 1, 2, \ldots, k \]

Repeat steps 2 and 3 until the cluster centroids stabilize up to a desired level of approximation.

### 3.9.4 K-MEDOIDS CLUSTER

Partitioning Around Medoids (PAM) was developed by Kaufman and Rousseeuw in 1990. To find \( k \) clusters, PAM’s approach is to determine a representative object for each cluster. This representative object, called a medoid, is meant to be the most centrally located object within the cluster. Once the medoids have been selected, each non-selected object is grouped with the medoid to which it is the most similar. This is also a partitioning, prototype based and supervised clustering algorithm.

The input of k-medoids algorithm is a dataset consisting of \( n \) vectors and it outputs \( k \)-clusters which together form a mutually exclusive and exhaustive partitioning of the data set.

The k-medoids algorithm expects the user to specify \( k \), namely the number of clusters to be formed; this is the main reason for classifying, k-medoids algorithm as a supervised clustering algorithm. This algorithm falls under the category of “prototype-based” algorithms, typically identified objects identifying ultimate clusters.
Partitioning Around Medoids (PAM) algorithm is as follows:

Step 1: Randomly select $k$ of the $n$ data points as the mediods.

Step 2: Associate each data point to the closest medoid. ("closest" here is defined using any valid distance metric, most commonly Euclidean distance, Manhattan distance or Minkowski distance)

Step 3: For each mediod $m$
   i. For each non-mediod data point $o$
   ii. Swap $m$ and $o$ and compute the total cost of the configuration

Step 4: Select the configuration with the lowest cost.

Repeat steps 2 to 4 until there is no change in the medoid.

### 3.10 Computational Results

When comparing two stocks of data over a long period of months, it is to discover how many months the stocks have similar movements, though the same trend may appear in different months for different stocks.

Here, the analysis work is carried out using Derivative Segment Approximation (DSA) and Coefficient of Variation technique in order to reduce the computational time and dimensions of time series data.

To cluster the time series, k-means clustering $k=4$ is used. Setting the value for $k$ is still a main issue in k-means and k-medoids clustering. In the present case, $k$ is set to the value which is slightly greater than square root of the number of elements in the time series.
which satisfy the condition \( k = 2^n \), where \( n \) is an integer. The results of 
\( k \)-means and \( k \)-medoids cluster are shown as below

Table 3.1: Formation of Cluster Means

<table>
<thead>
<tr>
<th></th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th>Cluster 3</th>
<th>Cluster 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.545548</td>
<td>38.89513</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.095514</td>
<td>1.316328</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.842779</td>
<td>5.017402</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24.568042</td>
<td>14.046541</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The sequences related to TCS and Wipro stocks are shown as below,

\[ [3 2 4 1 2 3 1 3 2 4 3 2 1 4 3 3 3 1 2 1 2 3] \quad [4 2 3 1 4 4 2 4 4 2 4 4 2 4 3 3 2 2 2 3] \]

To demonstrate the similarity, the six distances such as Euclidean, Manhattan, Squared Euclidean, Euclidean, Minkowski, and Chebyshev distances are applied. The comparison of these distances is determined by within clusters sum of squares by clusters using histograms. The results are as follows:

![Figure 3.1: Sum of Squares of within Clusters using k-means](image)

Figure 3.1: Sum of Squares of within Clusters using k-means
Figure 3.2: Sum of Squares of within Clusters using k-means

Table 3.2: Formation of Cluster Medoids

<table>
<thead>
<tr>
<th>Manhattan</th>
<th>squared</th>
<th>Euclidean</th>
<th>Minkowski</th>
<th>Chebyshev</th>
</tr>
</thead>
<tbody>
<tr>
<td>16.474119</td>
<td>14.198092</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.555929</td>
<td>1.624601</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22.626055</td>
<td>38.895153</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.096592</td>
<td>6.029034</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The k-medoid sequences related to TCS and Wipro stocks are shown as below,

\[ [1 2 3 4 2 1 4 1 2 3 1 2 4 2 1 1] \quad [1 2 2 3 1 1 2 1 1 2 1 1 2 1 2 4 2 2 4] \]

Figure 3.3: Sum of Squares of within Clusters using k-medoids
Figure 3.4: Sum of Squares of within Clusters using k-medoids

Figure 3.5: Accuracy of Similarity

3.11 CONCLUSION

Many proposed works carried out using time series similarity analysis in finding similarity between daily stocks of reputed companies. The suitable analysis is carried out in reducing the dimensions of time series data and the k-means and k-medoids cluster is adopted in finding the similarity matches of two time series data. Then the whole sequence matching process is carried out to seek
all the possible sequences of the given time series. When compared to two sequences such as TCS and Wipro, it is observed that only k-means 17% and k-medoids 43% exhibit similar pattern. And the other months of two respective stocks exhibits a dissimilar movement.

K-Medoids is better than k-means when comparing the similarity using histograms. The various distances such as Manhattan, Euclidean and Squared Euclidean distances are applied for measuring the similarity. It is found that the implementation of Chebyshev distance by within clusters sum of squares by clusters appears to be minimum, using histograms.