

LEE JIA YEE

B.Sc. (Honours) Statistical Computing and Operations Research

**EXPLORING DISTANCE
MEASURES FOR TIME SERIES
DATA: A COMPARATIVE
ANALYSIS**

LEE JIA YEE

**BACHELOR OF SCIENCE
(HONOURS) STATISTICAL
COMPUTING AND
OPERATIONS RESEARCH**

**FACULTY OF SCIENCE
UNIVERSITY TUNKU ABDUL
RAHMAN
MAY 2024**

**EXPLORING DISTANCE MEASURES FOR TIME SERIES DATA: A
COMPARATIVE ANALYSIS**

By

LEE JIA YEE

A project report submitted to the
Department of Physical and Mathematical Science
Faculty of Science
Universiti Tunku Abdul Rahman
in partial fulfilment of the requirements for the degree of
Bachelor of Science (Honours)
Statistical Computing and Operations Research

MAY 2024

ABSTRACT

EXPLORING DISTANCE MEASURES FOR TIME SERIES DATA: A COMPARATIVE ANALYSIS

LEE JIA YEE

Time series similarity search is a method used to identify the identical pattern within two sets of time series data, finds widespread utility in clustering, anomaly detection, and forecasting. In real-world scenarios, vibration data are often vast, intricate, and noisy, with adjustments in time, amplitude, and phase shifting direct influence on search outcomes. Through a systematic evaluation, various distance measurement methods including Euclidean distance, Dynamic Time Warping, Fast Fourier Transform, Symbolic Aggregate Approximation, and Matrix Profile are performed under diverse conditions such as frequency shifting, amplitude scaling, state change, and noise. The comparative study encompasses not only quantitative assessments of accuracy but also considerations of computational efficiency and robustness. The findings reveal Matrix Profile generally outperforms classic measures like Euclidean distance, Dynamic Time Warping, and Fast Fourier Transform in accuracy, but performs poorly compared to Symbolic Aggregate Approximation. While Matrix Profile exhibits shorter computational time than Symbolic Aggregate Approximation, it slightly extends beyond other classic measures. Thus, Matrix Profile presents competitive advantages among

distance measurement methodologies. By providing a comprehensive examination of similarity measurement techniques, this study equips the idea for the strength and weaknesses of distance measures, providing valuable insight for decision-making in time series data mining activities.

ACKNOWLEDGEMENTS

I would like to convey my heartfelt gratitude towards my supervisor Dr. Beh Woan Lin for her guidance and helpful comments during my days of research. I am extremely thankful for her help in many technical aspects, valuable discussions and endless patience. It has been a great privilege and joy to study under her guidance and supervision.

In addition, I would like to thank my family for their love and support. Also, I truly appreciate my coursemates and friends for their encouragement and companionship on my journey of completing this research.

Finally, I am really grateful to all those who helped me in completing this project. This project would not have been possible without the collective support from all of you.

DECLARATION

I hereby declare that the project report is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTAR or other institutions.

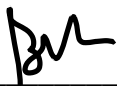


LEE JIA YEE

APPROVAL SHEET

This project report entitled “**EXPLORING DISTANCE MEASURES FOR TIME SERIES DATA: A COMPARATIVE ANALYSIS**” was prepared by LEE JIA YEE and submitted as partial fulfilment of the requirements for the degree of Bachelor of Science (Hons) Statistical Computing and Operations Research at Universiti Tunku Abdul Rahman.

Approved by:



(Dr. Beh Woan Lin)

Date: 14/4/2024

Supervisor

Department of Physical and Mathematical Science

Faculty of Science

Universiti Tunku Abdul Rahman

FACULTY OF SCIENCE
UNIVERSITI TUNKU ABDUL RAHMAN

Date: 03/04/2024

PERMISSION SHEET

It is hereby certified that **LEE JIA YEE** (ID No: **19ADB02132**) has completed this final year project entitled “**EXPLORING DISTANCE MEASURES FOR TIME SERIES DATA: A COMPARATIVE ANALYSIS**” under the supervision of Dr. Beh Woan Lin (Supervisor) from the Department of Physical and Mathematical Science, Faculty of Science.

I hereby give permission to the University to upload the softcopy of my final year project in pdf format into the UTAR Institutional Repository, which may be made accessible to the UTAR community and public.

Yours truly,



(LEE JIA YEE)

TABLE OF CONTENTS

ABSTRACT	i
ACKNOWLEDGEMENTS.....	iii
DECLARATION	iv
APPROVAL SHEET.....	v
PERMISSION SHEET	vi
TABLE OF CONTENTS	vii
LIST OF TABLES.....	ix
LIST OF FIGURES	x
LIST OF ABBREVIATIONS	xi
CHAPTER	
INTRODUCTION	1
1.1 Introduction.....	1
1.2 Problem Statement	2
1.3 Research Objective	3
1.4 Significance of Study.....	4
1.5 Chapter Layout.....	4
LITERATURE REVIEW	5
2.1 Introduction.....	5
2.2 Distance Measures in Time Series	8
2.3 Summary	14
METHODOLOGY	16
3.1 Introduction.....	16
3.2 Datasets	17
3.3 Scenario.....	20
3.3.1 Frequency Shift.....	20
3.3.2 Amplitude Scale.....	23
3.3.3 Stage Change	25
3.3.4 Noise	27
3.4 Distance Measures	30
3.4.1 Euclidean Distance (ED)	30
3.4.2 Dynamic Time Warping (DTW)	33
3.4.3 Fast Fourier Transform (FFT)	36

3.4.4	Symbolic Aggregate Approximation (SAX)	38
3.4.5	Matrix Profile (MP)	43
3.5	Experimental Flow	45
RESULTS AND DISCUSSION		47
4.1	Introduction	47
4.2	Robustness Test (Accuracy)	47
4.2.1	Frequency Shift	47
4.2.2	Amplitude Scale	50
4.2.3	Stage Change	52
4.2.4	Noise.....	54
4.2.5	Time Complexity	56
CONCLUSION		58
5.1	Summary of Research	58
5.2	Significance and Implication of Study.....	59
5.3	Limitations and Recommendations.....	60
REFERENCES		61
APPENDICES		65

LIST OF TABLES

Table		Page
2.2	Summary of comparison studies	12
3.2	Summary of datasets in Chunk	19
3.4.1	Algorithm of Euclidean Distance	31
3.4.2	Algorithm of Dynamic Time Warping	34
3.4.3	Algorithm of Fast Fourier Transform	37
3.4.4	Algorithm of Symbolic Aggregate Approximation	40
3.4.4	SAX breakpoints search table	41
3.4.5	Algorithm of Matrix Profile (mpdist)	44
4.2.1	Average MAPE for Frequency Shift	48
4.2.2	Average MAPE for Amplitude Scale	50
4.2.3	Average MAPE for Stage Change	53
4.2.4	Average MAPE for Noise	54
4.3	Average Processing Time	57

LIST OF FIGURES

Figure		Page
3.2	Overview of Dataset	17
3.3.1	Sample for Frequency Shifting	22
3.3.2	Sample for Amplitude Scaling	24
3.3.3	Sample for Stage Change	26
3.3.4	Sample for White Noise	29
3.4.1	Sample of Euclidean Distance	32
3.4.2	Sample for Dynamic Time Warping	35
3.5	Progress Flow	45
4.2.1	Graph of Frequency Shift Accuracy	48
4.2.2	Graph of Amplitude Scale Accuracy	50
4.2.3	Graph of Stage Change Accuracy	52
4.2.4	Graph of Noise Accuracy	54

LIST OF ABBREVIATIONS

APCA	Adaptive Piecewise Constant Approximation
ARIMA	Autoregressive Integrated Moving Average
CDM	Compression-based Similarity Measure
CHEB	Chebyshev Polynomials
DCT	Discrete Cosine Transformation
DFT	Discrete Fourier Transform
DISSIM	Integral of Euclidean Distance
DTW	Dynamic Time Warping
DWT	Discrete Wavelet Transform
ED	Euclidean Distance
EDR	Edit Distance on Real Sequences
ERP	Edit Distance with Real Penalty
FDTW	Fast Fourier Transform with Dynamic Time Warping
FED	Fast Fourier Transform with Euclidean Distance
FFT	Fast Fourier Transform
HMM	Hidden Markov Model
IPLA	Indexable Piecewise Linear Approximation
LCSS	The Longest Common SubSequence
MAPE	Mean Absolute Percentage Error
MP	Matrix Profile
NND	Nearest Neighbor Distance
PAA	Piecewise Aggregate Approximation
PCA	Principal Component Analysis

SAX	Symbolic Aggregate Approximation
SpADe	Spatial Assembling Distance
SVD	Singular Value Decomposition
Swale	Sequence Weighted Alignment model
TQuEST	Threshold Queries
TWED	Time Warp Edit Distance

CHAPTER 1

INTRODUCTION

1.1 Introduction

A time series is a form of data that records the observation in the sequence of time and they can reflect various phenomena and events that change over time (Hu *et al.*, 2023). Nowadays, with the rapid development of Internet of Things (IoT) technology, time series data are widely deployed in various fields, ranging from industrial manufacturing to medical care or even biological studies, economics and geology. Examples of time series data can be the stock market price, sensor readings for temperature, heart rate, etc. The data is generated constantly in a large amount with high speed. Over time, massive amounts of time series data are being generated and increase the complexity of the data. The massive volume and complexity bring greater challenges for data analysis. Therefore, time series data mining is needed to assist in extracting useful information from the data.

Time series data mining is a field of study that focuses on extracting meaningful patterns, trends, and insights from time series data (Mörchen and Fabian Mörchen aus Dillenburg, 2006). In essence, it deals with analyzing data points collected at successive time intervals to uncover hidden relationships, forecast future values, and understand underlying structures within the data. In actual life, there is a massive dataset that could make the computation difficult and costly. Thus, time series data mining is essential for us to perform the study in a large dataset. Time Series similarity search is a technique of data mining that can assist in extracting information from enormous datasets by determining the degree of similarity between two datasets. Theoretically, if the time series has the same

pattern or same sources, they have a high similarity. Otherwise, if the two-time series look very different, they have high dissimilarity or distance. Time series similarity search has claimed wide usage in a variety of fields including economics, medicine, industry and even music. Numerous functions can be performed with similarity search and have been discovered by researchers such as anomaly detection, clustering, rules discovery, classification and forecasting.

1.2 Problem Statement

In time series data analysis, selecting an appropriate similarity measure from a range of choices is crucial for accurately assessing the resemblance between data readings recorded at different times or under varying conditions. However, the effectiveness of similarity measures can vary significantly depending on external factors. Any changes to the data in frequency, amplitude, or presence of noise will result in different outcomes and this will lead to a differ in performance for each similarity measures. Thus, how to select a suitable similarity measure will become the main concern in this study.

1.3 Research Objective

The objective of this study is to conduct an analysis of similarity measures in the context of

1. To evaluate the accuracy and efficiency of a range of similarity measures, including Euclidean Distance, Dynamic Time Warping, Fast Fourier Transform, Symbolic Aggregate Approximation and Matrix profile to detect the changes in time series in terms of frequency, amplitude and stages.
2. To evaluate the precision of Euclidean Distance, Dynamic Time Warping, Fast Fourier Transform, Symbolic Aggregate Approximation and Matrix profile in computing the distance for time series under the disturbance of noise.

1.4 Significance of Study

This study is significant as it can serve as a reference for other researchers. It helps identify the most suitable distance measures for similarity search experiments and tasks in various scenarios based on their needs. It can lead to cost reduction and time savings for the research procedure by ensuring a desirable result is obtained. By choosing the best similarity measures in sensor data, organizations from various fields can optimize their resources, enhance data quality, and improve the overall performance of the time series data network.

1.5 Chapter Layout

This report will review journals relevant to the similarity search including their application and list of studies involving the comparison of similarity measures in Chapter 2. Then, Chapter 3 will discuss the methodology of this study as well as the data used in this study while the result obtained in this study will be discussed in Chapter 4. Lastly, Chapter 5 will conclude the findings of this study and the future scope is discussed.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, we will discuss the previous study done by researchers on time series similarity search, including their views and findings. Also, to access the functionality of distance measure, some commonly used applications of the similarity search are explored.

Similarity search is widely used for clustering purposes in which clustering is a technique of grouping sets of data points into clusters or groups based on the similarity between the data points. The purpose of clustering is to find a homogeneous group by partitioning the data in such a way that data points within the same cluster are more similar to each other than to those in other clusters while maximizing the dissimilarity between clusters (Kleist, 2015; Komitova *et al.*, 2022). For instance, Yohansa, Notodiputro and Erfiani have described the time series clustering of COVID-19 cases in Daerah Khusus Ibukota (DKI) Jakarta by using Dynamic Time Warping (DTW). By measuring the distance between daily case data of COVID-19 cases, the author enables clustering covid-19 cases into different districts to identify the spread and distribution patterns of the virus across DKI Jakarta. They have successfully identified 6 clusters from the cases. The clustering proved to be efficient and showed the Mean Absolute Percentage Error (MAPE) values ranging from 10% to 20% while comparing to the clustering result from Autoregressive Integrated Moving Average (ARIMA) models (Yohansa, Notodiputro and Erfiani, 2022).

Next, Similarity Search can also be used for classification. Classification is a fundamental task in supervised machine learning to predict the categorical class label of new instances based on past observations or labelled data (Kleist, 2015; Dove *et al.*, 2023). In similarity search, classification is achieved by mapping the new instance with the item in predefined classes. Then the object will be classified as the classes that have the lowest distance from it. Silva *et al.* (2015) have proof of the usage of similarity search for music recognition in 2015. In this study, the authors design a new model, Similarity Matrix Profile (SiMPle) to study the similarity in music. By comparing the audio recordings based on their structural and melodic similarities, SiMPle is proven to be effective in various tasks within the field of Music Information Retrieval including cover song recognition and audio thumbnailing even from multiple audio streams.

Time series similarity searches are also very powerful in detecting the regularities or patterns within data. In the context of time series data, there will be a wide range of unpredictable pattern occurs. The pattern might include trends, seasonal variation, cyclic behaviour and others (Kleist, 2015). From the undulated time series, the repeated pattern or the optimal value may give some meaningful information. The search for repeated patterns is called motif detection, where a motif is a replicated subsequence that occurs within a time series dataset (Komitova *et al.*, 2022). Motif detection can be used as a subroutine in other data mining tasks such as clustering and classification (Mörchen and Fabian Mörchen aus Dillenburg, 2006). This is because we can segmentize the long series of signals into multiple frequently repeating patterns that each pattern representing the same activities.

Searching for the optimal pattern is known as anomaly detection can also be achieved by similarity search, where anomalies are the data points that differ significantly from the typical behaviour of the system such as sudden spikes, drops or shifts in the time series (Komitova *et al.*, 2022). Recently, there has been an increased interest of researchers in the field of anomaly detection in a diverse range of domain applications. Anomaly detection is found to be useful in indicating unusual events, errors, faults or any fraudulent activities. For example, Sivaraks and Ratanamahatana (2015) proposed a robust and accurate anomaly detection algorithm (RAAD) in the medical area to compare the heartbeats to identify the anomaly candidates.

Lastly, another famous application for similarity search is to deal with forecasting. Prediction or forecasting, in the context of time series data mining, refers to the process of estimating future values or trends based on historical observation (Kleist, 2015). To deal with forecasting, the trends, patterns or events from the previous data are studied. The frequently used method was the autoregressive integrated moving average (ARIMA) model or Hidden Markov Model (HMM). By forecasting future trends, patterns or events, organizations can make informed decisions, plan resources, manage risks and optimize strategies earlier. As an example, Mishra *et al.* use DTW to forecast hydrological data (Mishra *et al.*, 2015). Using similarity search, Liang, Wang and Wu. and Zhao *et al.* are also analyzing to forecast the stock market with the help of similarity search (Liang, Wang and Wu, 2021; Zhao *et al.*, 2021).

In the various fields of study, distance measurement has proven its importance and its implications. For different situations, researchers are choosing different

distance measures for help. So this raises the problem of whether the distance measures chosen can lead to an accurate analysis.

2.2 Distance Measures in Time Series

The distance measure is the main component in the similarity search. It serves the purpose of quantifying the similarity or dissimilarity between two or more objects. Some researchers also named them similarity measures or time series representation methods. The only difference is that time series representation methods include the transformation of the raw time series data before calculating the dissimilarities (Komitova *et al.*, 2022). The distance obtained from the distance measure must follow several criteria, such as 1) Uniqueness: $d(x, x) = 0$, 2) Symmetry: $d(x, y) = d(y, x)$, 3) Non-negativity: $d(x, y) \geq 0$ (Dove *et al.*, 2023).

In general, the distance measure can be categorised into four categories which are shape-based, edit-based, feature-based and structural-based (Dove *et al.*, 2023). A shape-based similarity measure is a technique used to quantify the similarity between two-time series based on the shapes of their temporal patterns, such as Lp-norms distance, Dynamic Time Warping (DTW), Spatial Assembling Distance (SpADe) and Threshold Queries (TQuEST). Edit-based is focused on quantifying the distance by considering the minimal steps of transformation. The measures used edit-based theorem are Edit Distance with Real Penalty (ERP), Edit Distance on Real Sequences (EDR), the Longest Common Subsequence (LCSS) and Time Warp Edit Distance (TWED) (Kleist, 2015; Dove *et al.*, 2023). Feature-based similarity measures concentrate on comparing the similarity between time series data by extracting and comparing

specific features or attributes of the data. The measures included Discrete Fourier Transform (DFT) or Discrete Wavelet Transform (DWT) (Senthil and Suseendran, 2019; Dove *et al.*, 2023). Lastly, model-based similarity measures involve comparing time series data based on the similarity of the underlying models that represent them such as Autoregressive Integrated Moving Average (ARIMA), Hidden Markov Model (HMM) and Compression-based Similarity Measure (CDM). Instead of directly comparing the raw data points or extracted features, these methods assess how well different models capture the behaviour of the time series (Senthil and Suseendran, 2019; Dove *et al.*, 2023).

There are plenty of distance measures that have been developed and refined by researchers to date. While every strategy can accomplish the desired goal, each performs differently depending on the circumstances. There is no a perfect option that works in every situation. Each similarity measure has benefits and drawbacks of its own. Also as the number of methods used grows, we are curious about how well each way works. Thus, many studies focusing on studying the efficiency of similarity measures are done. Ding *et al.* compared eight representation methods and nine similarity measures on 38-time series datasets in 2008 to determine their accuracy in classifying varying sizes with the help of a 1-NN classifier (Hui Ding *et al.*, 2008). Although the author suggests that no single similarity measure is superior for time series, an interesting discovery was that the accuracy of elastic measures decreases with larger data sizes.

Senthil and Suseendran compare six different similarity measures for trajectory clustering in outdoor surveillance scenes. The measures include Euclidean distance (ED), PCA+Euclidean distance (PCA+ED), Hausdorff distance,

Hidden Markov Model (HMM), Longest Common Subsequence (LCSS) distance, and Dynamic Time Warping (DTW) distance. The performance is evaluated using the Correct Clustering Rate (CCR) and Time Cost (TC). In his study, he concluded that the Hausdorff and HMM distances are found to be ineffective for trajectory clustering in outdoor surveillance scenes. While LCSS and DTW distances measure shape similarity well, their high computational cost weakens their competitive ability. The PCA+ED produces better results at a lower cost, but it has limitations in distinguishing speed variation. (Senthil and Suseendran, 2019)

Sivaraks and Ratanamahatana also investigated six different measures of trajectory similarity and their application in clustering GPS trajectories of foraging trips made by birds. The measures included Dynamic Time Warping (DTW), Fréchet distance, Nearest Neighbor Distance (NND), Longest Common Subsequence (LCSS), and Edit Distance on Real Sequences (EDR). The author suggested that DTW and Fréchet distance performed the best while NND had the worst performance in this study (Sivaraks and Ratanamahatana, 2015).

In clustering, another researcher focused on selecting suitable distance measures for accurate clustering against low and high-dimension datasets (Shirkhorshidi, Aghabozorgi and Ying Wah, 2015). The author has evaluated the performance of 14 distance measures in his study. The concept of the Rand index is used to measure the clustering accuracy. Overall, the authors recommend the Average Distance measure across different scenarios. The result of this research also proved that the performance of similarity search will be varied according to the dimension of the dataset.

Kianimajd et al. present an additional concept for evaluating the resilience of similarity search in cardiovascular disease (CVD) diagnosis data under various conditions, including baseline variation, time scaling and shift, amplitude scaling and shifting, and white Gaussian noise. In his findings, he states that the robustness of the similarity measure is based on insensitivity. Among them, Fourier transform and Euclidean distance showed higher sensitivity to amplitude shift, time scale and shifting. Also, Dynamic time warping is insensitive to White Gaussian Noise and Discrete Wavelet transform is insensitive to amplitude shift (Kianimajd *et al.*, 2017). Thus, this study shows the impact of different variations on the similarity search.

Kljun, Teršek and Erikštrumbelj conducted a study that is somewhat similar in that they examine and contrast the effectiveness of 12-time series similarity measures in clustering concerning warping, scaling, and datasets varying in length. The evaluation of clustering is done by using the Rand index (Kljun, Teršek and Erikštrumbelj, 2020). In contrast to the others, they discovered that Piccolo distance performed well overall. Due to the different similarity measures selected by researchers, this study gives a different suggestion regarding the effect of variation. However, a similar pattern is found in previous studies, where similarity measures are performed differently while considering warping, scaling, and datasets varying in length.

The summary of the analysis is tabulated and shown below:

Table 2.2: Summary of comparison studies

Author	Title	Distance Measures
Hui Ding et al. (2008)	Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measures	ED, DTW, LCSS, ERP, EDR, Swale, SpADe, TQuEST, DFT, SVD, DCT, PAA, APCA, CHEB, SAX, and IPLA
Senthil and Suseendran (2019)	Data mining techniques using time series research	ED, PCA+ED, Hausdorff distance, HMM-based distance, LCSS, and DTW
Sivaraks and Ratanamahatana (2015)	Robust and accurate anomaly detection in ECG artifacts using time series motif discovery	DTW, Fréchet distance, NND, LCSS, and EDR
Shirkhorshidi, Aghabozorgi and Ying Wah (2015)	A Comparison study on similarity and dissimilarity measures in clustering continuous data	ED, Average Distance, Weighted ED, Chord, Mahalanobis, Cosine Measure, Manhattan, Mean Character Difference, Index of Association, Canberra Metric, Czekanowski Coefficient, Coefficient of Divergence, Pearson Coefficient
Kianimajd et al. (2017)	Comparison of different methods of measuring similarity in physiologic time series	Minkowski, ED, DTW, Pearson Correlation Coefficient, Mahalanobis, DFT, DWT
Kljun, Teršek and Erikštrumbelj (2020)	A review and comparison of time series similarity measures	Lp norms, DISSIM, DTW, EDR, ERP, LCSS, TQuEST, Cross-correlation, CDM, Piccolo distance, Prediction-based distance, and embedding-based similarity

From the list of comparisons done, we found that some famous similarity measures are frequently used. Thus, based on previous analysis, five distance measures are selected for this research. The first similarity measure is Euclidean Distance (ED). Although ED has limitations in handling unequal-length datasets and is sensitive to time shifting and noise, it is widely used due to being parameter-free and having low computational complexity (Hui Ding *et al.*, 2008; Shirخورshidi, Aghabozorgi and Ying Wah, 2015). At the same time, ED can maintain a reliable result. Next, Dynamic Time Warping (DTW) is selected. As an advanced method from ED, DTW allows the comparison of unequal-length datasets through its warping ability. This feature provides a competitive advantage in analyzing time-shifted and noisy datasets (Kianimajd *et al.*, 2017).

Next, Discrete Fourier Transform (DFT) and Symbolic Aggregate Approximation (SAX) are also famous feature-based similarity measures that are frequently found in research. This is because both methods allow dimensionality reduction and thus have efficient computation complexity (Dove *et al.*, 2023). Also, due to its transformation characteristics, they are insensitive to amplitude shifting or scaling (Kianimajd *et al.*, 2017). In this research, a derivative method from Fourier analysis called Fast Fourier Transform (FFT) is chosen because it inherits characteristics from Discrete Fourier Transform (DFT) but with a faster algorithm (Brigham and Morrow, 1967).

Lastly, the Matrix Profile has been studied in this experiment. While the Matrix Profile is not widely included in performance comparisons, it is considered advantageous due to its ability to handle various situations effectively. This is attributed to its scalability, simplicity, and versatility (Michael Yeh *et al.*, 2016).

Here is also a successful application of Matrix Profile done by Pizon, Kulisz and Lipski conducted Matrix Profile in the maintenance systems with high levels of output satisfaction (Pizon, Kulisz and Lipski, 2021). Li et al. demonstrated the successful application of Matrix Profile in enhancing the performance of a planetary gearbox (Li *et al.*, 2023).

From the analysis, it is found that the Rand Index is commonly used for evaluating the performance of similarity measures (Kljun, Teršek and Erikštrumbelj, 2020; Shirخورshidi, Aghabozorgi and Ying Wah, 2015). However, the Rand Index is only accessible for clustering purposes. In the case of Yohansa, Notodiputro and Erfiani (2022), time series clustering of COVID-19 cases in DKI Jakarta, another evaluation method MAPE is used to compare the result from DTW and ARIMA model providing a possible way for evaluation(Yohansa, Notodiputro and Erfiani, 2022). Thus, based on the evidence, MAPE is chosen as the evaluation metric for assessing the performance of similarity measures

2.3 Summary

Previous studies have demonstrated the varied performance of similarity measures depending on the dataset and environment. While some measures perform well in specific tasks or conditions, there is no universal solution for all cases. From the literature review, we found some powerful distance measures with different strengths. Although a comparison is done, the choice of distance measure leads to varying conclusions. Most of the research only addresses one

or two scenarios, lacking comprehensive coverage. Matrix profiles are not commonly included in comparisons of distance measures. Thus, this study aims to compare the performance of commonly used and powerful distance measures which are Euclidean distance, Dynamic Time Warping, Fast Fourier Transform, Symbolic Aggregate Approximation and Matrix Profile along with different scenarios like frequency shifting, amplitude scaling, stage change, and noise. The evaluation will focus on using Mean Absolute Percentage Error (MAPE) to assess performance comprehensively. Cases of scenarios are made available in studies for a wide range of discovery.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The goal of this study is to identify the performance of distance measures in terms of robustness and time complexity. In this chapter, we define the six similarity measures on the concept of how the distance measures worked in identifying the dissimilarity in time series data. The distance measures included Euclidean Distance (ED), Dynamic Time Warping (DTW), Fast Fourier Transform with Euclidean Distance (FED), Fast Fourier Transform with Dynamic Time Warping (FDTW), Symbolic Aggregate Approximation (SAX) and Matrix Profile (MP).

This chapter also introduced the methodology procedure, starting from the introduction of the datasets used, data processing steps taken and the whole experimental steps to come out with the results. In order to obtain a fair result for the performance of distance measures, some scenarios like frequency shifting, amplitude scaling, phase change and noise have been introduced to our datasets. Therefore, the theory of how these scenarios could be applied to the datasets has been discussed here.

3.2 Datasets

The dataset considered in this research is secondary data retrieved from Ooi et al. as stated in their journal (Ooi *et al.*, 2022). The dataset consists of vibration data collected continuously from real-world observations. The data is obtained from an 18-inch industrial fan operating at a low speed and using the ESP8266 wireless vibration sensor. The raw data consists of the fan's acceleration records in three dimensions, with an average sampling rate of 350 Hz. This data was collected using a direct read-and-send approach detailed in the journal (Ooi *et al.*, 2022). However, in this experiment, we are specifically 1-dimensional movement which is the x -axis movement dataset. The dataset therefore contained only one variable, that is x -axis acceleration order in time manner. This dataset contained 75000 observations in total. Since the dataset does not have a clearly defined data collection period, we can only plot the time series data using the index. The overview of the dataset is provided below:

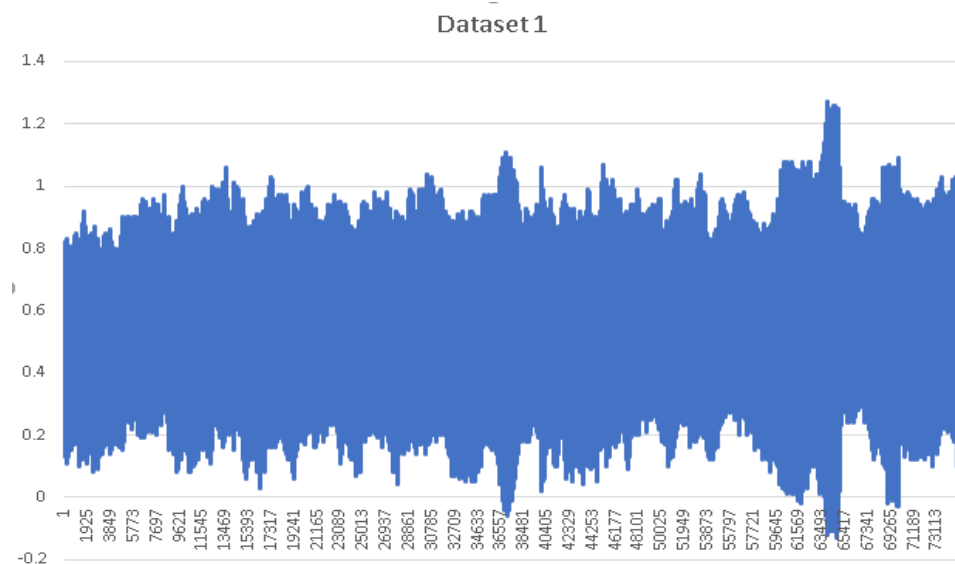


Figure 3.2: Overview of Dataset

To enable all similarity measures in this experiment, a similarity search of equal length is conducted. The dataset is divided into seven equal Chunks, each consisting of 10,000 observations for comparing similarities. In other words, the last 5000 observations are not being considered in this case. To obtain a semantic meaning of the result, the datasets are smoothed by normalising each Chunk according to their mean and standard deviation (Attig and Perner, 2011). The summary of the dataset is attached below:

Table 3.2: Summary of datasets in Chunk

	Chunk1	Chunk2	Chunk3	Chunk4	Chunk5	Chunk6	Chunk7
count	10000	10000	10000	10000	10000	10000	10000
mean	-1.16E-14	6.18E-15	-8.68E-15	-5.25E-15	-4.00E-15	-1.02E-14	3.44E-15
std	1	1	1	1	1	1	1
min	-2.87071	-3.19823	-3.23409	-3.12145	-3.05413	-3.0466	-2.84328
25%	-0.714	-0.65888	-0.68331	-0.75825	-0.6856	-0.64793	-0.71687
50%	0.087063	0.08435	0.081918	0.063735	0.0654	0.065185	-0.00807
75%	0.764888	0.641769	0.719611	0.731596	0.700861	0.713473	0.784124
max	2.79836	3.181124	2.887769	2.889303	3.011626	3.176968	2.993923

3.3 Scenario

In real life, there is a lot of variation that could change the time series data. Here we state the variation as the scenario occurs to the time series data. The changes in time series data will lead to varied outcomes in similarity search. Thus, distance measures should be selected based on their sensitivity to various scenarios. By learning the theory in this section, we could understand how the scenario affects datasets and the dissimilarity results.

3.3.1 Frequency Shift

Frequency Shift refers to the shifting of data in time resulting in a time variation in the data. The time variation may cause a delay or advance the data in time by a constant time interval without changing its original shape (Chaparro *et al.*, 2015). Meanwhile adding to the time variable will shift the data to the left (advance) and subtracting a constant number from the time variable will shift the data to the right (delay) (Chaparro *et al.*, 2015). Frequency shifting often occurs in real life such as the audio signal, sonar signal, electrocardiogram data, and others. Any variation in speed, latency, or temporal misalignment might be the cause of frequency shifting.

For instance, Kianimajd *et al.* (2017) have introduced a lot of variation concepts in their analysis of the performance of similarity measures in clustering and disease classification so that an automated system can efficiently perform Cardiac Vascular Diseases personal management. One of their variations was a study on the sensitivity of the similarity measures on frequency shifting (Kianimajd *et al.*, 2017). Their study successfully drew an image of how the

frequency shifting could be applied in the analysis. Another study has also taken into account frequency shifting to introduce a novel method for measuring the similarity of time interval datasets for process optimization purposes (Hafil, Jeschke and Meisen, 2017). The authors claim that deviation in time cannot always be avoided, and so by considering the frequency shifting, it can make the comparison more realistic and in a humanoid fashion (Hafil, Jeschke and Meisen, 2017).

However, it is clearly understood that advancing is impossible to implement in the real-time data but only delaying. Advanced data are only allowable in saved data or recorded signals. As both of our datasets are sensor data, where mostly implemented in the real-time collection, a delay in the datasets was added to demonstrate time shifting in this experiment with the given formulae attached:

$$X(t) = Y(t + a) \tag{1}$$

where $Y(t)$ is the original dataset, a is the positive value of the time interval added to the original dataset and $X(t)$ represents the new shifted dataset. For instance, let $Y(t)$ become a sinusoidal function with frequency =0.5 and amplitude = 1, with $a = 0.5$, the sample of frequency shifting will be like the figure below.

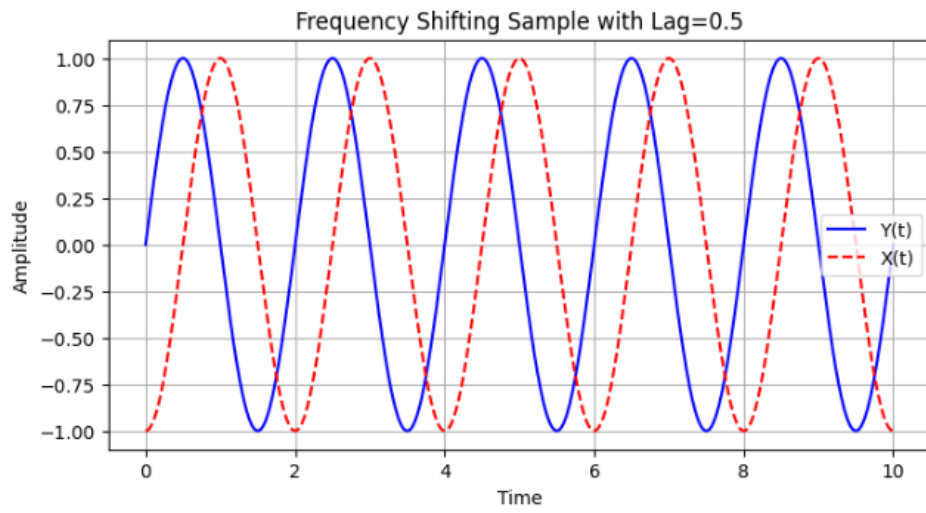


Figure 3.3.1: Sample for Frequency Shifting

3.3.2 Amplitude Scale

Amplitude Scale is defined as rescaling the amplitude of the data in either amplified or attenuated. While implementing amplitude scaling, the shape of the data such as the frequency will remain as original but the amplitude is altered. To create the amplitude scale, a constant value of a is multiplied by the original data. If the constant value a is greater than 1 ($a > 1$) then the data will be amplified, else if constant value a is smaller than 1, then the data will be attenuated. If a is equal to 1, the data will remain the same, where the amplitude scale is not formally working. Amplitude sometimes acts as the intensity of activity (Mörchen and Fabian Mörchen aus Dillenburg, 2006). The variations in amplitude levels could be brought on by various measurement scales or sensor sensitivity. Amplitude scale is a powerful technique in time series such that it can enlarge the analog signal which is frequently used in medical devices (Semmlow, 2018). For instance, we can easily detect the activities produced by the heart with the help of amplitude scaling by enlarging the electrical signal (Semmlow, 2018). Not only analogue signal, but amplitude scale is also suitable to adjust the intensity of sound signals or adopt in the image data to adjust the brightness or contrast of the image. Thus, we cannot ignore the effect of amplitude scaling while conducting our experiment.

In the previous study conducted by Kianimajd et al.(2017) , amplitude scale was also a factor conducted in his analysis which provided us with insight into the adoption of amplitude scaling in our experiment. The formulae to consider the amplitude scale are shown below:

$$X(t) = a Y(t) \tag{2}$$

where $Y(t)$ is the original dataset, a is the positive constant value multiplied by the original dataset and $X(t)$ represents the new amplitude scaled dataset. For instance, let $Y(t)$ become a sinusoidal function with frequency = 0.5 and amplitude = 1, with $a = 1.5$, the sample of amplitude scaled data will be like the figure below.

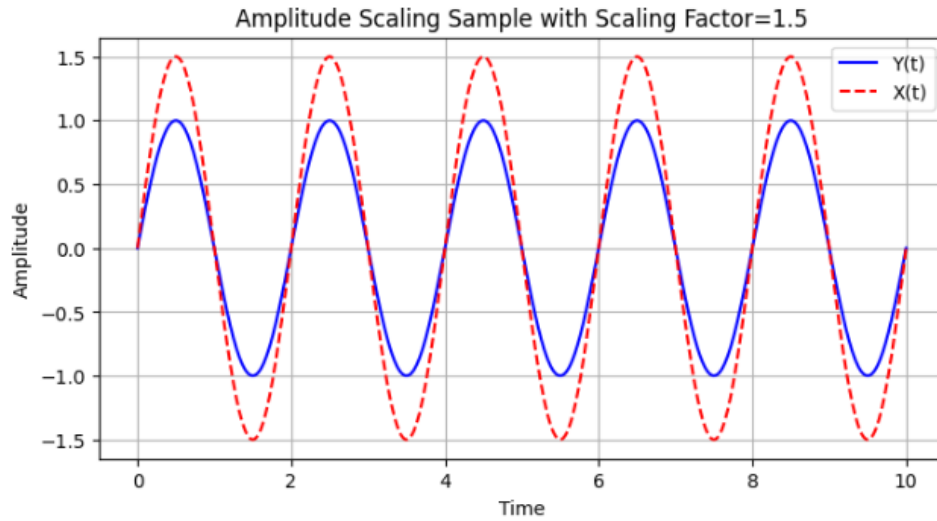


Figure 1.3.2: Sample for Amplitude Scale

3.3.3 Stage Change

Stage change is an immediate change in the pattern of the data and the moment of pattern change is called the change point. Any pattern changes such as changes in mean, variance or trend are examples of stage change (Lavielle, 2005). Stage change in time series data can also represent how a process changes over time. In the massive time series dataset, there might be a lot of change points present as the behaviour of data could change over time due to external events or internal systematic changes in dynamics (Lavielle, 2005). Behavioural change sometimes might bring us an important message. For instance, stage change can be used in speech recognition to detect audio segmentation and recognise boundaries between silence, sentences or words (Aminikhanghahi and Cook, 2017). Stage change is also frequently used in climate change detection such as identifying the level of greenhouse gases in the atmosphere (Aminikhanghahi and Cook, 2017).

Due to these reasons, change point detection was one of the major applications of similarity search. For instance, Liu et al. (2023) have compared various methods including Dynamic Time Warping to deal with his study of change-point detection and anomaly detection (Liu *et al.*, 2023).

Thus, stage change has been adopted in our experiment to study the effect on the performance of distance measures. For an easy demonstration purpose, we have applied stage change by only changing the mean of time series for a certain segment. The formulae to consider stage change are shown below:

$$X(t) = \begin{cases} Y(t) & , b \leq t < c \\ Y(t) + a & , c \leq t \leq d \end{cases} \quad (3)$$

where $Y(t)$ is the original dataset while a is the positive constant value added to the original dataset to move the segment of time series data upward. $X(t)$ represents the new stage changed dataset which is a combination of two data, where one is the same as original data in the time interval of b until c , and another is the amplitude shifted data from time interval c until d . For instance, let $Y(t)$ become a sinusoidal function with frequency = 0.5 and amplitude = 1, with $a = 0.5$, $b = 0$, $c = 6$ and $d = 10$ the sample of amplitude scaled data will be like the figure below.

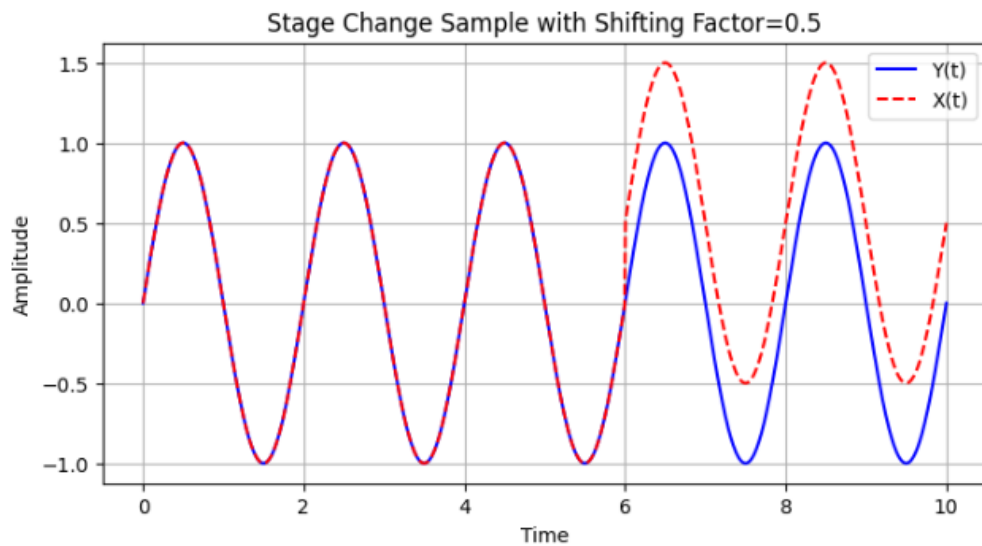


Figure 3.3.3: Sample for Stage Change

3.3.4 Noise

A noise refers to a corruption of data that possibly leads us the meaningless information. In fact, noisy data or random processes are happening everywhere in real life. For instance, the heartbeat, breathing frequency, weather, or catastrophic events are all random and are examples of noise. The noise can be caused by environmental issues, hardware failures, programming errors or even human spelling errors could be the factor of creating noise (Bunde, 2023).

Noise can be distributed into a few categories. The most frequently seen noises are white noise, gaussian noise, pink noise, brown noise and red noise. Briefly explain, a white noise is the random signal with equal intensity added to the original data that is uncorrelation in time. Red noise has 0 mean, constant variance and a finite range of correlations in time, in which the correlation coefficient ranges from $0 < r < 1$ (Bunde, 2023). Else if the correlation range is infinite, they are sometimes referred to as pink noise (Bunde, 2023).

Among the synthetic noise distributions, Gaussian noise may be the best simulation of real noise. This is because the noise in real life is particularly complex and could be a combination of many different sources. The real noise can be regarded as the sum of many independent random variables with different probability distributions and their normalized sum tends to increase with the number of noise sources close to a Gaussian distribution according to the Central Limit Theorem (Huo, 2022). Therefore, Gaussian noise is a simple and good approximate simulation when dealing with this complex situation and the unknown real noise distribution.

Gaussian noise is a statistically independent random process that holds the characteristic of stationary which is the noise has zero mean and constant variance (Marmarelis, 2004). As the noise is inescapable in real life, many researchers have utilised this scenario in their study, especially in the similarity search area. One example is the study done by Kianimajd et al.(2017) also inspects the performance of distance measures under noisy data. Apart from that, Lin et al. also reviewed the effect of Gaussian noise while the authors proposed a novel method of partitioning clustering algorithm for time series (Lin *et al.*, 2004). As an inheritance of ideas from the study, white Gaussian noise is added to our dataset while we access the performance of distance measures. The formulae to consider white Gaussian noise are shown below:

$$X(t) = Y(t) + W(t), \tag{4}$$

$$W(t) = WGN \sim N(0, \sigma^2) \tag{5}$$

where $Y(t)$ is the original dataset, $W(t)$ is the white Gaussian noise distribution added in this experiment with 0 mean and constant variance σ^2 . $X(t)$ represents the new noisy dataset after the white Gaussian noise is added. For instance, let $Y(t)$ become a sinusoidal function with frequency = 0.5 and amplitude = 1, then set $\sigma = 0.2$, the sample of the noise-affected data will be like the figure below.

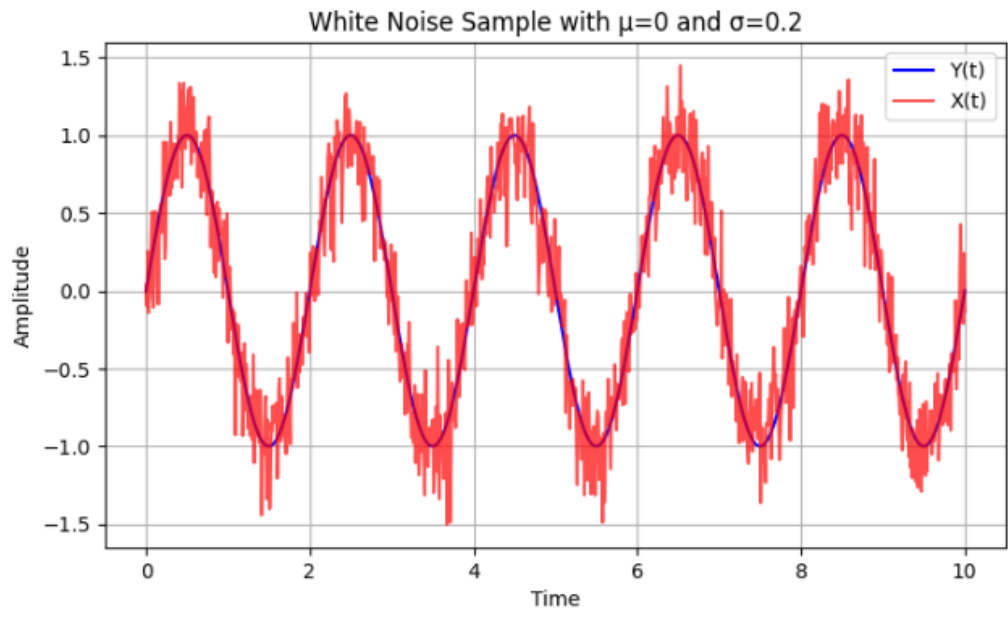


Figure 3.3.4: Sample for White Noise

3.4 Distance Measures

Distance measures, also known as distance measurement or dissimilarity measurement are one the components forming similarity search (Li *et al.*, 2022). As per its abbreviation, distance measures are used to evaluate the pairwise distance between data points. Similarity between datasets can be formulated by computing the distance between them. By evaluating the similarities, applications like classification, clustering, pattern recognition or prediction could be achieved. Here, we are only focused on several popular distance measures, such as Euclidean Distance (ED), Dynamic Time Warping (DTW), Fast Fourier Transform (FFT), Symbolic Aggregate Approximation (SAX) and Matrix Profile (MP). In this section, we will discuss the concept of how distance measures are employed in our experiment.

3.4.1 Euclidean Distance (ED)

Euclidean Distance (ED) is a matrix that calculates the distance between points frequently used since decades years ago. It is also known as L2-norm distance and is part of the member of the L_p-norms distance family when $p = 2$ (Górecki and Piasecki, 2018). Meanwhile, ED inherits the characteristics of lock step measures which can only quantify the straight line distance between two points. Thus, ED is very easy to implement with only the time complexity of $O(nd)$, where n stands for the number of data points, and d stands for the dimensionality of the dataset (Kljun, Teršek and Erikštrumbelj, 2020; Shifaz et al., 2021). Since our data is only in one dimension in this case, the time complexity will be $O(n)$. However, a small limitation of ED is that it only allows the comparison of the two datasets with equal length. The demonstration of ED is schematically

present in Figure 3.4.1 Due to its characteristics of free-parameter and simplicity, It is a very classic distance measure that is commonly included in the field of study in similarity search. The formula for constructing Euclidean distance is given as:

$$ED(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (6)$$

Let $X(t) = \{x_0, x_1, \dots, x_n\}$ and $Y(t) = \{y_0, y_1, \dots, y_n\}$ be the time series data with n data points each. Then x_i and y_i represent the position of vector form data point from time series $X(t)$ and time series $Y(t)$ respectively and $ED(X, Y)$ represents the distance among time series X and Y . So in this formulae, we will first calculate the square of the difference between data points one-to-one. Then, we will sum up all the differences before we square root it and obtain the final answer. The process will be done with the help of Scikit-learn.euclidean packages in Python and the algorithm for calculating ED are given below:

Table 3.4.1: Algorithm of Euclidean Distance

Algorithm 1: Euclidean Distance	
<hr/>	
	Input: Time series x, time series y
	Output: Euclidean distance between x and y, processing time
1	Start ← Get current time
2	n ← number of data points in each time series
3	Set Power_diff equal 0
	For i ← 0 to n do:
	Power_diff = power_diff + pow($x[i]-y[i],2$)
	End
	Euc_dist(x,y) = sqrt(power_diff)
4	End ← Get current time
5	Processing time ← Difference between start and end

In our experiment, we have pre-created the data frame from both of our datasets that consist of data distributed in chunks. The chunks act as an attribute for data storing, with the chunks' names i given. For instance, the chunk name for the dataset in this paper is Chunk 1 until Chunk 7. The chunks are dragged out and become the time series x or y continuously to calculate their ED among them by the given equation. Since ED is a lock-step measure, time series x and time series y will have the same number of data points and the computation is only allowed for the same index of data points. At the same time, the processing time is taken by considering the start time and end time of calculation.

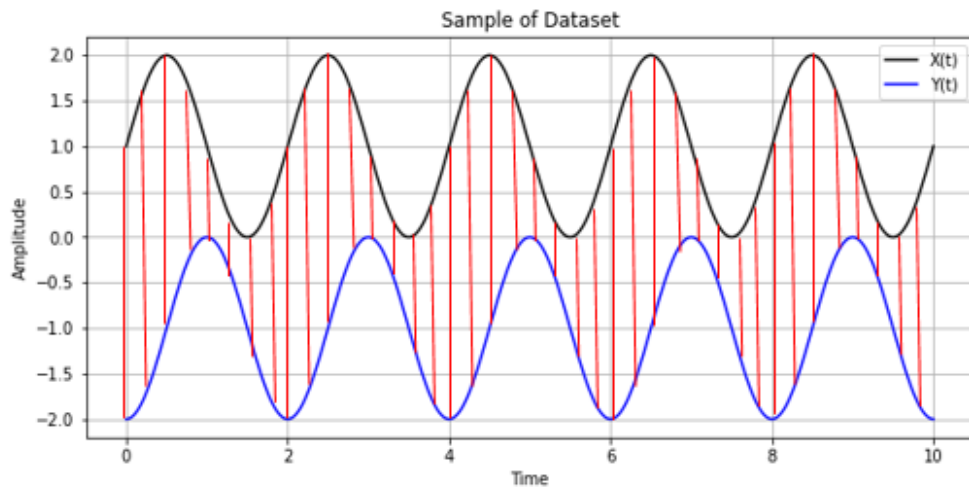


Figure 3.4.1: Sample of Euclidean Distance

3.4.2 Dynamic Time Warping (DTW)

Dynamic Time Warping (DTW) is another commonly used distance measure generically from ED. It has the same working theory as ED which calculates the distance between two points by calculating the square root of the sum square difference. In contrast to ED, DTW is an elastic distance measure that makes it possible to determine the distance between two points at various phases. More precisely, this approach allows for the distance computation of two data points in one-to-many alignment. The sample of alignment is drawn in Figure 3.4.2. However, the computations are only allowed within the warping window. So, the warping window is an important parameter to control the elasticity of distance measures (Senin, 2008). By minimizing the cumulative distance between aligned points within a window, the algorithm determines the best warping path. This makes it possible to identify similar patterns even when they are in different temporal sequences.

As warping windows could directly affect the efficiency of distance measures, window selection is very critical. This is because the large window leads to a high cost for the experiment while the small window may create poor accuracy results. When the window is equal to zero, DTW produces a one-to-one alignment, which is equivalent to the Euclidean distance (Shifaz et al., 2021). In our experiment, the size of datasets is affordable. Thus, full window dynamic time warping is taken to ensure the accuracy of distance measures.

$$DTW(X, Y) = \min_{\pi} \sqrt{\sum_{(i,j) \in \pi} d(x_i - y_j)^2} \quad (7)$$

Let $X(t) = \{x_0, x_1, \dots, x_n\}$ and $Y(t) = \{y_0, y_1, \dots, y_n\}$ is the time series data with n datapoint each. Then x_i and y_j represent the position of vector form data point

from time series $X(t)$ and time series $Y(t)$ respectively, π represents all the possible combinations of (i,j) , and $DTW(X, Y)$ representing the distance among time series X and Y . So in this formulae, the ED is calculated for all matching x_i and y_j . The minimum ED from all points is summed up to obtain the DTW distance. DTW have the time complexity of $O(nm)$, where n represents the length of time series x and m represents the length of time series y (Senin, 2008; Shifaz et al., 2021). Since all the chunks have equal lengths in our experiment, the time complexity of DTW can be also stated as $O(n^2)$. The process will be done with the help of `dtadistance.dtw` function in Python, and the algorithm for calculating DTW is given below:

Table 3.4.2: Algorithm of Dynamic Time Warping

Algorithm 2: Dynamic Time Warping

Input: Time series x , time series y
Output: Distance between x and y , processing time

- 1 Start \leftarrow Get current time
- 2 $n \leftarrow$ number of datapoints in each time series
- 3 For $i \leftarrow 0$ to n do:
 - DTW[$i,0$] $\leftarrow \infty$
 - DTW[$0,i$] $\leftarrow \infty$
- End
- For $i \leftarrow 0$ to $n-1$ do:
 - For $j \leftarrow 0$ to $n-1$ do:
 - DTW[$i+1,j+1$] $\leftarrow |x[i]-y[j]|^2 + \min(\text{DTW}[i,j], \text{DTW}[i, j+1], \text{DTW}[i+1,j])$
 - End
- end
- 4 End \leftarrow Get current time
- 5 Processing time \leftarrow Difference between start and end

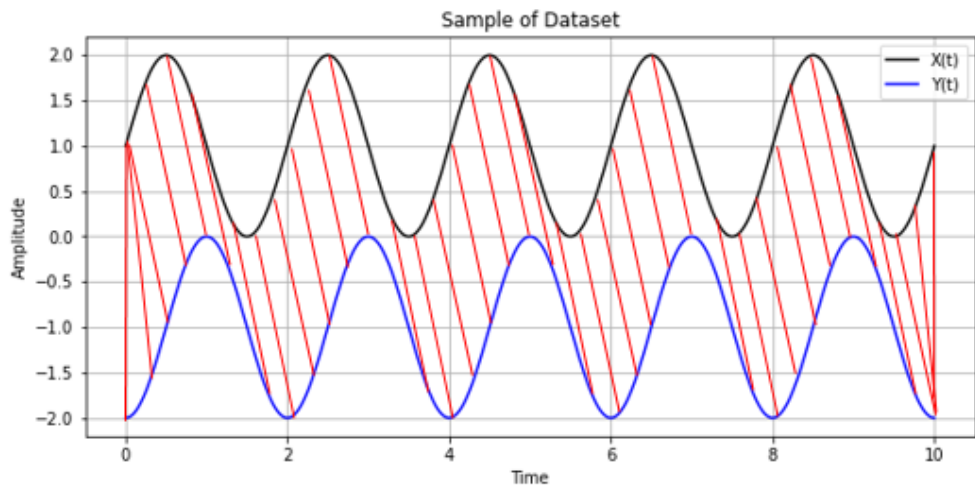


Figure 3.4.2: Sample for Dynamic Time Warping

3.4.3 Fast Fourier Transform (FFT)

Fourier analysis also known as harmonic analysis in a time series is a way of function that decomposes the series into a sum of periodic components (Peter Bloomfield, 2004). The analysis always be done by using the Fourier transform to transform the time series data from the time domain to the frequency domain by comparing the fluctuation to the sinusoids. The result is created from the transformation also called the spectrum. The sample of transformation is shown in Figure 3.4.3. The theorem creates the basic Discrete Fourier Transform (DFT) where the formula of transformation is given by (Gupta *et al.*, 2022):

$$F(k) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} f(x) \cdot e^{-\frac{j2\pi xk}{N}} \quad , 0 \leq k \leq N \quad (8)$$

In which,

$\{F(k)\}$: F_0, F_1, \dots, F_{N-1} is the sequence of complex number from frequency data

$\{f(k)\}$: f_0, f_1, \dots, f_{N-1} represent the original time series data.

N : number of data

K : sample times of function

However, the transformations are very tedious and require a lot of calculation. To tackle the issue, Cooley and Tukey (1965) discovered a faster algorithm for the transformation. By reducing the number of complex multiplication and addition, it speeds up the computational complexity of DFT from $O(n^2)$ to $O(n \log n)$ (Cooley and Tukey, 1965; Brigham and Morrow, 1967). For example, when $n=100$, the number of multiplication for DFT is 10000 while FFT only left 200 multiplications needed. As FFTs have a lower computational complexity, this makes them an essential tool for handling big datasets.

While FFT is only a tool for data transformation, it shall work with other distance measures to compute the distance. Therefore, in our experiment, the FFT is performed to transfer data using `np.fft.fft` built-in function. After we obtain the frequency data, two approaches of Euclidean Distance (FED) and Dynamic Time Warping (FDTW) are performed to complete a full similarity search to ensure the robustness of FFT works in different situations. The algorithm to apply FED or FDTW are shown below:

Table 3.4.3: Algorithm of Fast Fourier Transform

Algorithm 3: Fast Fourier Transform	
<hr/>	
	Input: Dataset in DataFrame (<code>chunks_df</code>)
	Output: Distance between x and y, processing time
1	Start \leftarrow Get current time
2	Transform <code>chunks_df</code> to frequency data by FFT
3	<code>magnitude_spectra</code> \leftarrow get the magnitude of frequency data
4	Calculate Euclidean distance between magnitude spectra
5	Calculate Dynamic Time Warping between magnitude spectra
6	End \leftarrow Get current time
7	Processing time \leftarrow Difference between start and end

3.4.4 Symbolic Aggregate Approximation (SAX)

Symbolic Aggregate Approximation (SAX) is another well-known distance measure in data mining. This framework was proposed by Lin et al. in 2003 to compute the distance from the symbolic representation of data. SAX drills the concept from Piecewise Aggregate Approximation (PAA) to lower the dimensionality of the datasets. The general concept of SAX implementation shall begin by dividing the normalized datasets into a few equal-length segments and grouping them according to their segment mean. Subsequently, while ensuring our datasets satisfy a Gaussian distribution via a normal probability plot, we can make SAX discretize the segment into a collection of discrete alphabetic symbols depending on the SAX breakpoints search table shown in Table 3.4.4. (Lin et al., 2003). The sample of SAX working theory is demonstrated in Figure 3.4.4

Following that, let X and Y become the original time series data with the n signal. x and y are the discretised data in alphabetic symbols, and w is the window of segment. The distance for SAX is calculated by the given formula Lin et al. (2003):

$$MINDIST(X, Y) = \sqrt{\frac{n}{w}} \sqrt{\sum_{i=1}^w (dist(x, y))^2} \quad (9)$$

where $dist(x, y)$ is the sum of the breakpoint difference between time series X and time series Y follows:

$$dist(x, y) = \begin{cases} 0, & \text{if } |r - c| \leq 1 \\ \beta_{\max(r, c)-1} - \beta_{\max(r, c)}, & \text{otherwise} \end{cases} \quad (10)$$

Given r is the numeric value for the alphabetical symbols from x and c is the numeric value for the alphabetical symbols from y . For instance, $a=1, b=2, c=3$

etc. While β is obtained from the Table 4. For example, $x=\{aabc\}$ and $y=\{bcba\}$, the breakpoint distance among them is $0+0+0+0.43=0.43$.

As we can see, SAX is highly dependent on the alphabetical size to decide how it discretizes the datasets, the results are easily affected by this factor. Theoretically, the alphabetical size, a shall not be too small nor too large, as it will produce a meaningless comparison, nor it shall not be too large. The range of alphabetical size must be an integer that lies within $2 < a < 20$ (Lin et al., 2003). The best alphabetical size will give the highest Tightness of the Lower Bound which is the ratio of SAX distance, $MINDIST(X, Y)$ to the Euclidean distance, $D(X, Y)$ as shown:

$$Tightness\ of\ Lower\ Bound = \frac{MINDIST(X,Y)}{D(X,Y)} \quad (11)$$

Therefore, SAX allows n arbitrary time series data reduced to a string of arbitrary length w ($w \ll n$) and it serves a lower bound to the Euclidean distance while maintaining the quality of the result (He *et al.*, 2020).

Attached is the algorithm to implement the SAX in this experiment with the help of the saxpy packages in Python:

Table 3.4.4: Algorithm of Symbolic Aggregate Approximation

Algorithm 4: Symbolic Aggregate Approximation

Input: Dataset in DataFrame (chunks_df)
Output: Distance between x and y, processing time

- 1 Start \leftarrow Get current time
- 2 Initialize min_segments to 2 and max_segments to 20
- 3 Initialize best_segment to None
- 4 For n_segments in range (min_segments, max_segments + 1):
 - a. Initialize total_tlb to 0
 - b. Initialize value to 0
 - c. Initialize distances matrix of size (len(selected)) filled with zeros
 - d. Initialize sdist matrix of size (len(selected)) filled with zeros
 - e. Set chunk1 to the data from chunks_df corresponding to the first chunk
 - f. For i, chunk2_column in enumerate(selected):
 - i. Extract the data for chunk2 from chunks_df
 - ii. Convert chunk1 and chunk2 to SAX representation with n_segments
 - iii. Calculate Euclidean distance between chunk1 and chunk2
 - iv. Calculate the minimum distance between SAX representations of chunk1 and chunk2
 - v. Calculate the Tight Lower Bound (TLB) using the minimum distance and Euclidean distance
 - vi. Add TLB to total_tlb
 - g. If value < total_tlb:
 - i. Update value with total_tlb
 - ii. Update best_segment with n_segments
- 5 Get sdist when n_segments equals to best_segment
- 6 End \leftarrow Get current time
- 7 Processing time \leftarrow Difference between start and end

The best segment for both datasets found as 20 via the algorithm given.

Table 3.4.4: SAX breakpoints search table

β_i	2	3	4	5	6	7	8	9	10	11
β_1	0	-0.43	-0.67	-0.84	-0.97	-1.07	-1.15	-1.22	-1.28	-1.34
β_2		0.43	0	-0.25	-0.43	-0.57	-0.67	-0.76	-0.84	-0.91
β_3			0.67	0.25	0	-0.18	-0.32	-0.43	-0.52	-0.6
β_4				0.84	0.43	0.18	0	-0.14	-0.25	-0.35
β_5					0.97	0.57	0.32	0.14	0	-0.11
β_6						1.07	0.67	0.43	0.25	0.11
β_7							1.15	0.76	0.52	0.35
β_8								1.22	0.84	0.6
β_9									1.28	0.91
β_{10}										1.34

Table 3.4.4 Continue: SAX breakpoints search table

β_i	12	13	14	15	16	17	18	19	20
β_1	-1.38	-1.43	-1.47	-1.5	-1.53	1.56	1.59	1.62	1.64
β_2	-0.97	-1.02	-1.07	-1.11	-1.15	1.19	1.22	1.25	1.28
β_3	-0.67	-0.74	-0.79	-0.84	-0.89	0.93	0.97	1	1.04
β_4	-0.43	-0.5	-0.57	-0.62	-0.67	0.72	0.76	0.8	0.84
β_5	-0.21	-0.29	-0.37	-0.43	-0.49	0.54	0.59	0.63	0.67
β_6	0	-0.1	-0.18	-0.25	-0.32	0.38	0.43	0.48	0.52
β_7	0.21	0.1	0	-0.08	-0.16	0.22	0.28	0.34	0.39
β_8	0.43	0.29	0.18	0.08	0	0.07	0.14	0.2	0.25
β_9	0.67	0.5	0.37	0.25	0.16	0.07	0	0.07	0.13
β_{10}	0.97	0.74	0.57	0.43	0.32	0.22	0.14	0.07	0
β_{11}	1.38	1.02	0.79	0.64	0.49	0.38	0.28	0.2	0.13
β_{12}		1.43	1.07	0.84	0.67	0.54	0.43	0.34	0.25
β_{13}			1.47	1.11	0.89	0.72	0.59	0.48	0.39
β_{14}				1.5	1.15	0.93	0.76	0.63	0.52
β_{15}					1.53	1.19	0.97	0.8	0.67
β_{16}						1.56	1.22	1	0.84
β_{17}							1.59	1.25	1.04
β_{18}								1.62	1.28
β_{19}									1.64

3.4.5 Matrix Profile (MP)

In comparison to the other methods, the Matrix Profile is a relatively fresh distance measure recently proposed by Yeh *et al.* in 2016. Ideally, Matrix Profile offers a novel method for determining the separation between two data sets via a sliding window technique. First, a sliding window of size m will be created to provide insight into any inherent structure of the data. This is because any abnormal pattern or anomalies will result in a high distance value that can warn us of this situation. In this instance, we assume M is 1000 out of 10000 data points in each chunk. For each subsequence created by the sliding window, the Matrix Profile computes the nearest neighbours within the time series by 1NN and stores in a matrix called similarity join set J_{AB} (Yeh *et al.*, 2016). Given that A and $B = \{T_{1,m}, T_{2,m}, \dots, T_{n-m+1,m}\}$, is all the possible sub-sequences of T obtained from the sliding window from time series X and Y respectively, in which n represent the length of time series X and m is the length of time series Y .

Then, Pairwise Euclidean Distance is used to calculate the windowed sub-sequence's distance within the join set (Yeh *et al.*, 2016). The distance obtained is denoted by P_{AB} as the matrix profile distance. To stop the pointless matches, an exclusion zone is also created at the same time. Finally, the Matrix Profile is constructed and the minimum value is changed in the distance profile. (Yeh *et al.*, 2016). By computing the distance layer by layer, we can somehow reduce computational complexity and make it insensitive to the noise.

In fact, Matrix profiles contain various algorithms such as STAMP, STOMP, SCRIMP and others. Although all of them utilize the same concept as we discussed, their steps of implementation might differ. In our experiment, `mpdist` was selected to become the algorithm we employed in this study due to its

strength of being able to support invariances to amplitude, offset, phase, order, linear trend, and stutter also robust to various types of data irregularities and noise (Gharghabi *et al.*, 2018).

The highlight of mpdist is that it does not stop once we obtain the matrix profile value (P_{AB}), but a novelty step is added on by combining the distance profiles P_{AB} and P_{BA} to become a join matrix profile, P_{ABBA} . Lastly, the distance measure considers the k -th smallest value in P_{ABBA} , where $k=0$. As a result, it will only yield a time complexity of $O(\text{SubseqNum} \times m)$ (Gharghabi *et al.*, 2018). Attached is the algorithm of mpdist implementation in this experiment. The packages of matrixprofile. algorithms can be found in Python for the implementation.

Table 3.4.5: Algorithm of Matrix Profile (mpdist)

Algorithm 5: mpdist

Input: Dataset in DataFrame (chunks_df)
Output: Distance between x and y, processing time

- 1 Start \leftarrow Get current time
- 2 Initialize an empty dictionary mpdist_distances
- 3 For each pair of chunk columns chunk1_column and chunk2_column in selected:
 - a. If chunk1_column is not equal to chunk2_column:
 - i. Extract the data for chunk1 and chunk2 from chunks_df
 - ii. Calculate the MPdist distance between chunk1 and chunk2 with a window size of 1000
 - iii. Store the distance in the mpdist_distances dictionary with key as 'chunk1_column - chunk2_column'

- 4 Convert the mpdist_distances dictionary into a DataFrame mpdist_distances_df with column name 'Distance'
 - 5 End ← Get current time
 - 6 Processing time ← Difference between start and end
-

3.5 Experimental Flow

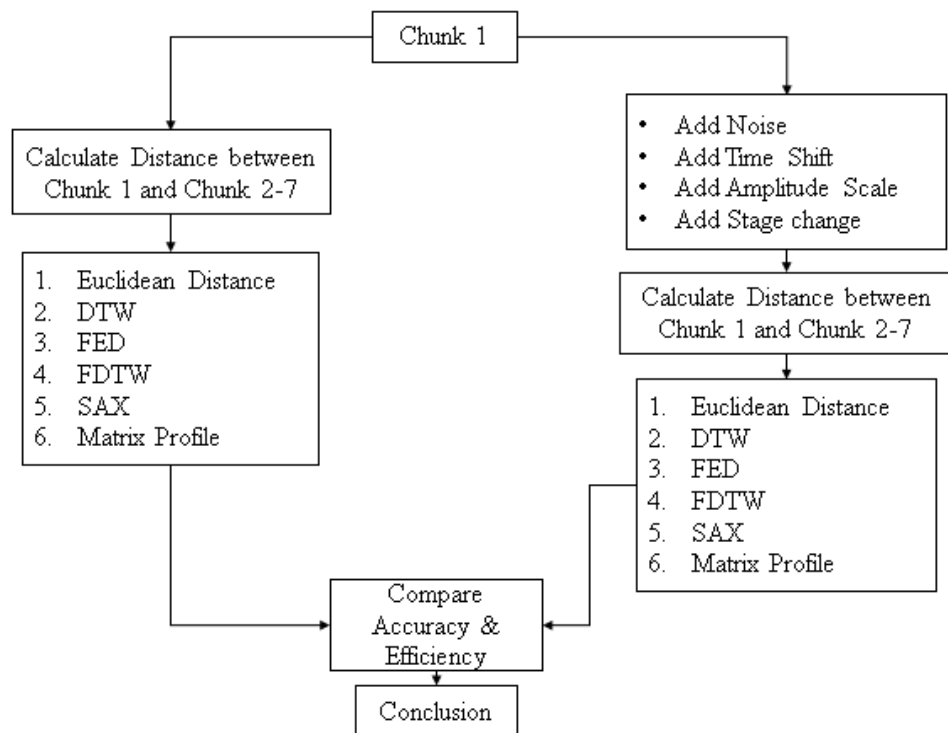


Figure 3.5: Progress Flow

In this experiment, we prepared our dataset as introduced in section 3.1. The distance between Chunk 1 and the other Chunk is calculated by using the distance measure we described in section 3.3. The distance obtained acts as a benchmark, which is when there is no disruption, the actual value that distance measure should obtain. At the same time, Chunk 1 has been added with four cases of scenario independently, noise, frequency shifting, amplitude scaling,

and stage change as stated in section 3.2. The computation of the distance between Chunk 1 data with the other Chunks is repeated once after adding the variation using the same distance measure. The performance of distance measure under the effect of the scenario is present by two measurements:

1. **Efficiency:** Efficiency in this case refers to how fast the distance measures can return an output. It is often related to the time complexity of distance measures. The processing time is determined in this experiment to represent its efficiency. The shorter the processing time, the more efficient the distance measure is.
2. **Accuracy:** In this experiment, we prefer distance measures that are insensitive to the scenario, in which the results are not easily affected by surrounding issues. Thus, the accuracy indicates how the later distance is different from the actual distance. Here, we use Mean Absolute Percentage Error (MAPE) for the accuracy measurement. MAPE is a measurement to identify the magnitude of error produced for a model in percentage with the given formula:

$$MAPE = \sum \left| \frac{\text{actual value} - \text{Predicted Value}}{\text{actual value}} \right| \quad (12)$$

where the actual value represents the distance calculated from the original standardized data where the predicted value represents the predicted distance if variation occurs. As for our prediction, if MAPE is high, represent the distance measures are sensitive to the variation and create high leverage with the original data. Else, the distance measure performed stably in the variation.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

In this experiment, the accuracy and efficiency of a total of six similarity measures are defined in the four scenarios, which are stage change, noise, amplitude scale, and frequency shift. The six similarity measures are Euclidean Distance (ED), Dynamic Time Warping (DTW), Fast Fourier Transform with Euclidean Distance (FED), Fast Fourier Transform with Dynamic Time Warping (FDTW), Symbolic Aggregate Approximation (SAX), and Matrix Profile (MP). The accuracy is defined by using MAPE in section 4.2 while efficiency is determined by computing the processing time in section 4.3.

4.2 Robustness Test (Accuracy)

4.2.1 Frequency Shift

To determine the effect of frequency shifting, the Chunk 1 standardized data is delayed by n pieces of data. Originally, Chunk 1 held data with index 0 to 9999, now Chunk 1 data are holding data with index n to $9999+n$. For an easily understood purpose, the n is named as frequency shifted level, which is the number of data that have shifted. To study the robustness of distance measure thoroughly, the MAPE is calculated in five variations of frequency shifted level, which are 200, 400, 600, 800 and 1000 respectively. The line chart below illustrates the performance of six distinct similarity measures across varying frequency shifted levels (X-axis) based on Mean Absolute Percentage Error (MAPE) values (Y-axis). Each line corresponds to a specific distance measure, as identified by the color-coded legends.

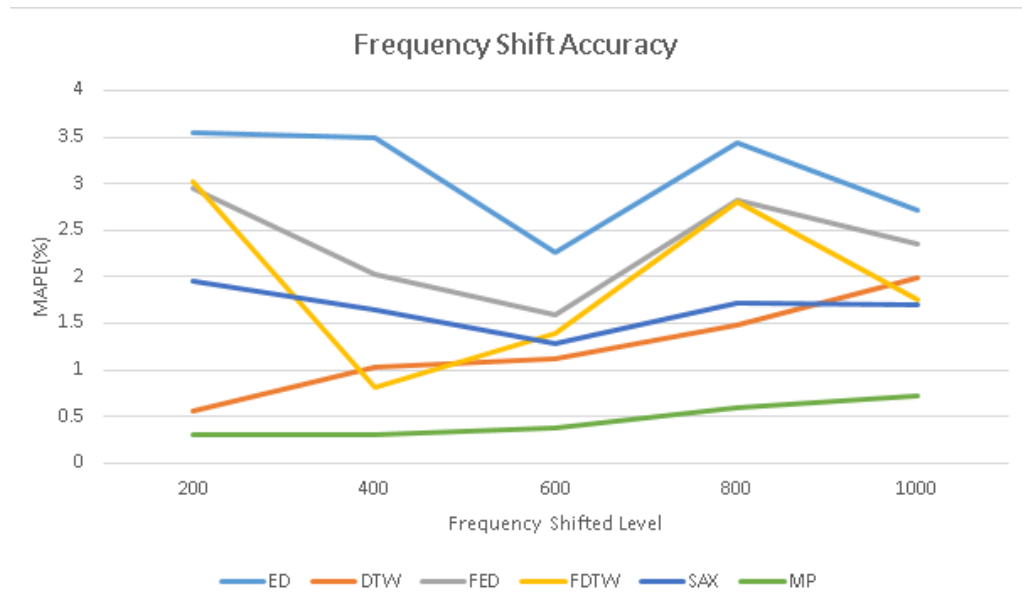


Figure 4.2.1: Graph of Frequency Shift Accuracy

Table 4.2.1: Average MAPE for Frequency Shift

Distance Measure	ED	DTW	FED	FDTW	SAX	MP
Average MAPE	3.090980	1.234967	2.348933	1.959600	1.656800	0.458637

In frequency shift, all distance measures perform well when the data points are shifted by up to 1000. The overall MAPE performance of frequency shifting for all measures is below 5%. Yet, there is no obvious pattern in the accuracy for different levels of frequency shifted across distance measures. However, Matrix Profile is found to perform well in frequency-shifted data when the frequency shift level is below 600. After that, the MAPE value for the Matrix profile has a slight increase but still has the lowest error rate. Euclidean distance performs the worst in frequency shifting. The performance of FED and FDTW is close but FDTW performs better than FED in this case. A special case is that FDTW experiences a significant drop in MAPE value when $n=400$. SAX maintains a

relatively consistent MAPE value across all frequency shift levels, ranging between 1% to 2%. Also, while dynamic time warping performs better than Euclidean distance, the MAPE of DTW increases obviously while the frequency shift level increases. On average, the accuracy ranking of MAPE measures should be $MP > DTW > SAX > FDTW > FED > ED$.

4.2.2 Amplitude Scale

In amplitude scaling, the original chunk 1 standardised data has been enlarged by an amplitude scaling factor to demonstrate the scenario. The scaling factors of 1.5, 2, 2.5, and 3 are applied respectively to assess the performance of distance measures at various levels of enlargement. The line chart below illustrates the performance of six distinct similarity measures across different levels of enlargement (X-axis) based on Mean Absolute Percentage Error (MAPE) values (Y-axis). Each line corresponds to a specific distance measure, as identified by the color-coded legends

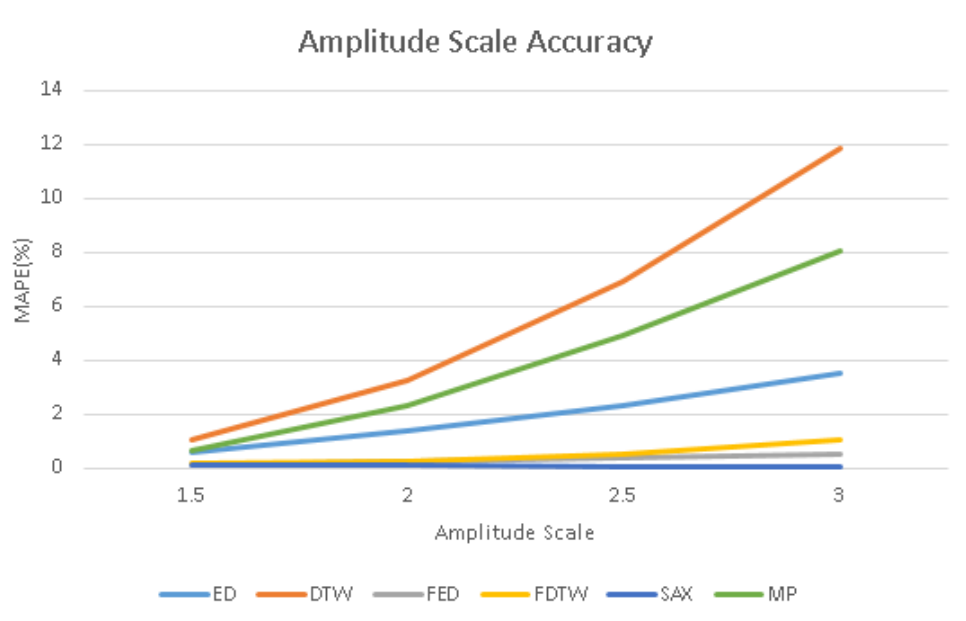


Figure 4.2.2: Graph of Amplitude Scale Accuracy

Table 4.2.2: Average MAPE for Amplitude Scale

Distance Measure	ED	DTW	FED	FDTW	SAX	MP
Average MAPE	1.958653	5.779699	0.331957	0.498645	0.079410	3.993286

In the amplitude scale, the MAPE of all distance measures increases gradually while the amplitude scale factor increases. Although there is a slight increase in MAPE, the performance of SAX, FED and FDTW is considered stable with low MAPE values all of the time. Meanwhile, the insensitivity of distance measures is followed by ED, MP and DTW. The impact on ED is most significant with higher amplitude scaling factors. The sensitivity of distance measures for all amplitude scaling factors constantly follows the order of SAX>FED>FDTW>ED>MP>DTW.

4.2.3 Stage Change

To create a stage change data, the first chunk of standardized data has been added by a constant value. The constant value acts as the stage change factor to identify the distance with other chunks. The distance is compared to the original distance obtained by using MAPE to investigate the sensitivity of the distance measures when stage change occurs. The experiments are conducted with five different stage change factors (0.2, 0.4, 0.6, 0.8, and 1) to assess how distance measures perform under various conditions. The line chart below shows how six different similarity measures perform under different stage change factors (X-axis) using Mean Absolute Percentage Error (MAPE) values (Y-axis). Each line corresponds to a specific distance measure, as identified by the color-coded legends.

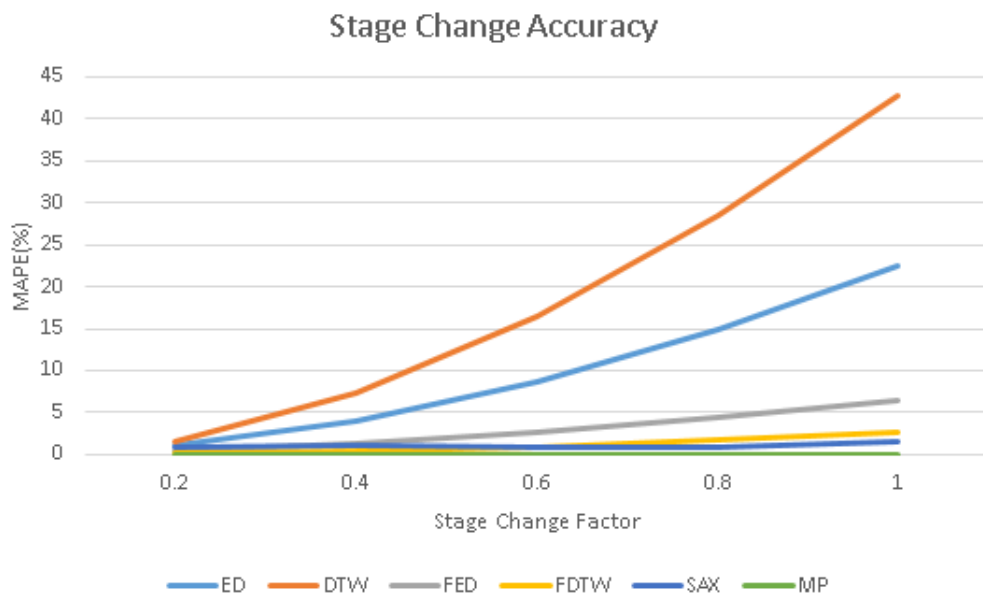


Figure 4.2.3: Graph of Stage Change Accuracy

Table 4.2.3: Average MAPE for Stage Change

Distance Measure	ED	DTW	FED	FDTW	SAX	MP
Average MAPE	10.198166	19.31905	3.10839	1.171329	1.034995	3.52E-14

The stage change accuracy graph revealed that as the stage change factor in the data increases, the error percentage (MAPE) also increases for all distance measures. However, MP performed steadily and was the best in all cases. At the same time, SAX has a result very close to the MP. FFT is also insensitive to stage change. FFT, FED, and FDTW have slightly higher MAPE values compared to SAX and MP. Besides, ED performed poorly in this case while DTW performed the worst. Also, DTW and ED are claimed that very sensitive to stage changes, where the MAPE gradually increases as the stage change factor increases. The average MAPE for distance measures consistently ranks in the order of MP > SAX > FDTW > FED > ED > DTW.

4.2.4 Noise

White noise is introduced into the original Chunk 1 standardized data to illustrate how sensor data can be affected by noise. The white noise created is followed by 0 mean and a constant standard deviation. This section examines how distance measures perform at various standard deviation levels. The standard deviations considered are 0.1, 0.15, 0.2, and 0.25. The following line chart shows how six different similarity measures perform as the standard deviation varies (X-axis) using Mean Absolute Percentage Error (MAPE) values (Y-axis). Each line corresponds to a specific distance measure, as identified by the color-coded legends.

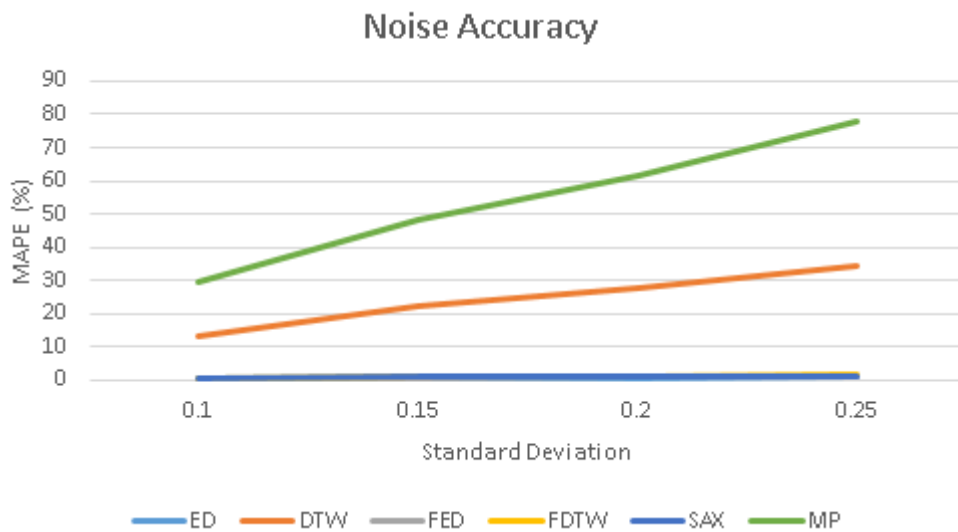


Figure 4.2.4: Graph of Noise Accuracy

Table 4.2.4: Average MAPE for Noise

Distance Measure	ED	DTW	FED	FDTW	SAX	MP
Average MAPE	0.772869	24.26806	0.65666	0.981739	0.839368	54.28727

In noise-affected data, the Fast Fourier transform (FED and FDTW), Symbolic Aggregate Approximation and Euclidean Distance show consistent performance as the standard deviation increases. These four methods exhibit high insensitivity to white noise when the standard deviation is below 0.25. In contrast, Matrix Profile and dynamic time warping are responsive to any existing noise. With a higher standard deviation, the error percentage rises. Among them, Matrix Profile performs the worst in the presence of Noise. The accuracy is followed by DTW, FDTW, SAX, ED and FED.

4.2.5 Time Complexity

The processing time for Python to run out the distance for each distance measure in different situations has been summarized. Generally, the performance of the distance measures is performed steadily in all situations with very little fluctuation. However, when the data has been scaled by amplitude, the processing time for DTW, FDTW, SAX, and MP has increased significantly.

Besides, the processing time for ED and FED are very short indicating the highest efficiency. The efficiency is then followed by DTW and FDTW. In this case, we found that the performance of ED and FED also DTW and FDTW are very close. This might claim that FFT may not significantly affect the processing time in this case and its performance depends on the distance measures accompanied. While MP requires a longer time compared to ED, DTW, FED and FDTW, it however takes much shorter time compared to SAX. Also, the processing time for MP falls within the range of 26s to 40s to compute 70000 data points. This is still reasonable, and the method still holds high efficiency. However, SAX has a processing time that differs hugely compared to the other distance measures and proves that the method lacks efficiency. In summary, the efficiency of distance measures is followed by the order of ED > FED > DTW > FDTW > MP > SAX.

Table 4.2.5: Average Processing Time

Average Processing Time (s)	Distance Measures					
	ED	DTW	FED	FDTW	SAX	MP
Stage Change	0.003523	9.673528	0.006259	9.801108	1291.142947	26.93634
Noise	0.004151	9.992925	0.003905	10.84566	1406.09986	27.85925
Frequency Shifting	0.000761	9.815669	0.003126	10.01978	1291.14295	27.30177
Amplitude Scaling	0.000756	14.14587	0.003251	14.45454	1321.69096	39.29264

CHAPTER 5

CONCLUSION

5.1 Summary of Research

In this paper, we tackle the issue of calculating the efficiency and accuracy of several well-known distance measures in various scenarios, such as the SAX, DTW, FFT, and ED. The scenario includes time shift, amplitude scale stage change and noise. Based on the result, the MP performs best in frequency shift and amplitude scale scenarios but shows poor performance in the presence of white noise. DTW, FED, FDTW, and ED yield similar results when white noise is present. SAX is recommended to select when there is a stage change present in the data. Nevertheless, ED is easily influenced by frequency shifting, whereas DTW struggles with stage-changed data and data with amplitude scale. For maximizing efficiency in distance measures without compromising accuracy, ED may be the optimal selection. When considering both efficiency and accuracy, MP provides substantial competitive advantages over most other distance measures. These results are applicable to the majority of large datasets and can serve as a valuable reference for determining the distance measures to be employed.

5.2 Significance and Implication of Study

This study consolidates findings from prior research to analyze distance measures across various categories and scenarios. The empirical finding in this study provides a new understanding of the strengths and weaknesses of distance measures in different scenarios. Overall, this study reinforces the importance of choosing the most appropriate distance measure based on specific situations, aiding researchers in their work. The choice of distance measures is relevant not only in manufacturing but also in other fields that deal with processing extensive time series data. This study offers insights that could optimize outcomes in practical applications.

5.3 Limitations and Recommendations

There are a lot of distance measures discoveries by researchers. This study is only focusing on the performance of limited distance measures. One limitation is that other distance measures may outperform the selected ones. The paper suggests a methodology selection based on the chosen measures. Additionally, this study focuses solely on comparing datasets of the same length and faces challenges related to data length requirements. Furthermore, the study's findings are limited to single-dimensional datasets and do not extend to multi-dimensional datasets. As an immediate expansion, additional circumstances in which subsequent research will examine the performance of distance measures for datasets with uneven lengths and multi-dimension could be considered. Additionally, expanding the comparison by including more distance measures could offer valuable insights for future studies.

REFERENCES

- Aminikhanghahi, S. and Cook, D.J. (2017) 'A survey of methods for time series change point detection', *Knowledge and Information Systems*, 51(2), pp. 339–367. Available at: <https://doi.org/10.1007/s10115-016-0987-z>.
- Arvind Gupta *et al.* (2022) 'AN OVERVIEW OF FOURIER TRANSFORM ON IMAGE', *AGPE THE ROYAL GONDWANA RESEARCH JOURNAL*, 3(10), pp. 13–18.
- Attig, A. and Perner, P. (2011) 'The Problem of Normalization and a Normalized Similarity Measure by Online Data', *Transactions on Case-Based Reasoning*, 4(1), pp. 3–17. Available at: www.ibai-institut.de.
- Brigham, E. O and Morrow, R.E. (1967) 'The fast Fourier transform', *IEEE Spectrum*, 4(12), pp. 63–70.
- Bunde, A. (2023) 'The Different Types of Noise and How They Effect Data Analysis', *Chemie-Ingenieur-Technik*, 95(11), pp. 1758–1767. Available at: <https://doi.org/10.1002/cite.202300031>.
- Chaparro, L.F. *et al.* (2015) 'Signals and Systems Using MATLAB © Second Edition', in *Elsevier eBooks*. 2nd edn. Available at: www.mathworks.com/.
- Cooley, J.W. and Tukey, J.W. (1965) 'An Algorithm for the Machine Calculation of Complex Fourier Series', *Mathematics of Computation*, 19(90), pp. 297–301.
- Dove, S. *et al.* (2023) 'A user-friendly guide to using distance measures to compare time series in ecology', *Ecology and Evolution*, 13(10). Available at: <https://doi.org/10.1002/ece3.10520>.
- Gharghabi, S. *et al.* (2018) 'Matrix Profile XII: MPdist: A Novel Time Series Distance Measure to Allow Data Mining in More Challenging Scenarios', in *Proceedings - IEEE International Conference on Data Mining, ICDM*. Institute of Electrical and Electronics Engineers Inc., pp. 965–970. Available at: <https://doi.org/10.1109/ICDM.2018.00119>.
- Górecki, T. and Piasecki, P. (2018) 'An Experimental Evaluation of Time Series Classification Using Various Distance Measures', *Archives of Data Science, Series A*, 5(1). Available at: <https://doi.org/10.5445/KSP/1000087327/07>.
- Hafil, M., Jeschke, S. and Meisen, T. (2017) 'Similarity Analysis of Time Interval Data Sets Regarding Time Shifts and Rescaling', in *Proceedings ITISE*. Granada, pp. 995–1006.
- He, Z. *et al.* (2020) 'A boundary distance-based symbolic aggregate approximation method for time series data', *Algorithms*, 13(11), pp. 1–20. Available at: <https://doi.org/10.3390/a13110284>.

- Hu, C. *et al.* (2023) ‘Survey of Time Series Data Generation in IoT’, *Sensors*, 23(15). Available at: <https://doi.org/10.3390/s23156976>.
- Hui Ding *et al.* (2008) ‘Querying and Mining of Time Series Data: Experimental Comparison of Representations and Distance Measures’, in *Proceedings of the VLDB Endowment*, pp. 1542–1552.
- Huo, F. (2022) *Analysis of noise effects with Deep Learning and Structural Health Monitoring applications*. Master’s Degree Thesis. Politecnico di Torino.
- Kianimajd, A. *et al.* (2017) ‘Comparison of different methods of measuring similarity in physiologic time series’, in *IFAC-PapersOnLine*. Elsevier B.V., pp. 11005–11010. Available at: <https://doi.org/10.1016/j.ifacol.2017.08.2479>.
- Kleist, C. (2015) *Time Series Data Mining Methods: A Review*. Master’s Thesis. Humboldt-Universität zu Berlin.
- Kljun, M., Teršek, M. and Erikštrumbelj, E.E. (2020) ‘A review and comparison of time series similarity measures’, *Zbornik mednarodne Elektrotehniške in računalniške konference*, pp. 367–370.
- Komitova, R. *et al.* (2022) ‘Time Series Data Mining for Sport Data: a Review’, *International Journal of Computer Science in Sport*, 21(2), pp. 17–31. Available at: <https://doi.org/10.2478/ijcss-2022-0008>.
- Lavielle, M. (2005) ‘Using penalized contrasts for the change-point problem’, *Signal Processing*, 85(8), pp. 1501–1510. Available at: <https://doi.org/10.1016/j.sigpro.2005.01.012>.
- Li, A. *et al.* (2022) ‘Distance measures in building informatics: An in-depth assessment through typical tasks in building energy management’, *Energy and Buildings*, 258. Available at: <https://doi.org/10.1016/j.enbuild.2021.111817>.
- Li, S. *et al.* (2023) ‘A periodic anomaly detection framework based on matrix profile for condition monitoring of planetary gearboxes’, *Measurement: Journal of the International Measurement Confederation*, 218. Available at: <https://doi.org/10.1016/j.measurement.2023.113243>.
- Liang, M., Wang, X. and Wu, S. (2021) ‘A novel time-sensitive composite similarity model for multivariate time-series correlation analysis’, *Entropy*, 23(6). Available at: <https://doi.org/10.3390/e23060731>.
- Lin, J. *et al.* (2003) ‘A Symbolic Representation of Time Series, with Implications for Streaming Algorithms’, *DMKD03: 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, 13, pp. 2–11.
- Lin, J. *et al.* (2004) ‘Iterative Incremental Clustering of Time Series’, *In EDBT*, pp. 106–122.

- Liu, J. *et al.* (2023) ‘Anomaly and change point detection for time series with concept drift’, *World Wide Web*, 26(5), pp. 3229–3252. Available at: <https://doi.org/10.1007/s11280-023-01181-z>.
- Marmarelis, V.Z. (2004) ‘Appendix II: Gaussian White Noise’, in *Nonlinear Dynamic Modeling of Physiological Systems*. Wiley, pp. 499–501. Available at: <https://doi.org/10.1002/9780471679370.app2>.
- Michael Yeh, C.-C. *et al.* (2016) ‘Matrix Profile I: All Pairs Similarity Joins for Time Series: A Unifying View that Includes Motifs, Discords and Shapelets’, *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pp. 1317–1322.
- Mishra, S. *et al.* (2015) ‘Discovering flood rising pattern in hydrological time series data mining during the pre monsoon period’, *Indian Journal of Geo-Marine Sciences*, 44(3), pp. 303–317.
- Mörchen, F. and Fabian Mörchen aus Dillenburg, von (2006) *Time Series Knowledge Mining*. PhD Dissertation. Philipps-Universität at Marburg.
- Ooi, B.Y. *et al.* (2022) ‘Using Compressive Sampling to Fill Interbatch Data Gap From Low-Cost IoT Vibration Sensor’, *IEEE Internet of Things Journal*, 9(12), pp. 9820–9830. Available at: <https://doi.org/10.1109/JIOT.2022.3151051>.
- Peter Bloomfield (2004) *Fourier analysis of time series: An Introduction*. United State of America: John Wiley & Sons.
- Pizon, J., Kulisz, M. and Lipski, J. (2021) ‘Matrix profile implementation perspective in Industrial Internet of Things production maintenance application’, in *Journal of Physics: Conference Series*. IOP Publishing Ltd. Available at: <https://doi.org/10.1088/1742-6596/1736/1/012036>.
- Semmlow, J. (2018) ‘The Big Picture: Bioengineering Signals and Systems’, *Circuits, Signals and Systems for Bioengineers*, pp. 3–50. Available at: <https://doi.org/10.1016/B978-0-12-809395-5.00001-1>.
- Senin, P. (2008) *Dynamic Time Warping Algorithm Review*. University of Hawaii.
- Senthil, D. and Suseendran, G. (2019) ‘Data mining techniques using time series research’, *International Journal of Recent Technology and Engineering*, 8(2 Special Issue 11), pp. 121–129. Available at: <https://doi.org/10.35940/ijrte.B1020.0982S1119>.
- Shifaz, A. *et al.* (2021) ‘Elastic Similarity and Distance Measures for Multivariate Time Series’, *Knowl Inf Syst* 65, pp. 2665–2698. Available at: <http://arxiv.org/abs/2102.10231>.
- Shirkhorshidi, A.S., Aghabozorgi, S. and Ying Wah, T. (2015) ‘A Comparison study on similarity and dissimilarity measures in clustering continuous data’, *PLoS ONE*, 10(12). Available at: <https://doi.org/10.1371/journal.pone.0144059>.

Silva, D.F. *et al.* (2015) ‘Fast Similarity Matrix Profile for Music Analysis and Exploration’, *IEEE TRANSACTIONS ON MULTIMEDIA*, 14(8).

Sivaraks, H. and Ratanamahatana, C.A. (2015) ‘Robust and accurate anomaly detection in ECG artifacts using time series motif discovery’, *Computational and Mathematical Methods in Medicine*, 2015. Available at: <https://doi.org/10.1155/2015/453214>.

Yohansa, M., Notodiputro, K.A. and Erfiani, E. (2022) ‘Dynamic Time Warping Techniques for Time Series Clustering of Covid-19 Cases in DKI Jakarta’, *ComTech: Computer, Mathematics and Engineering Applications*, 13(2), pp. 63–73. Available at: <https://doi.org/10.21512/comtech.v13i2.7413>.

Zhao, F. *et al.* (2021) ‘A similarity measurement for time series and its application to the stock market’, *Expert Systems with Applications*, 182. Available at: <https://doi.org/10.1016/j.eswa.2021.115217>.

APPENDIX A

Acceptance Letter of International Conference on Advanced Materials and Applied Sciences (IConMAS 2024) Conference Proceedings



International Conference on Advanced Materials and Applied Sciences 2024
Universiti Pertahanan Nasional Malaysia, Kem Sg. Besi, 57000, Kuala Lumpur, Malaysia

Name : Dr. Woan Lin Beh
Institution : Universiti Tunku Abdul Rahman
Address : Universiti Tunku Abdul Rahman,
Jalan Universiti, Bandar Barat
Other 31900 Malaysia
Paper ID : IConMAS 2024: 021-014
Author : Beh Woan Lin
Co-Author : Lee Jia-Yee (daisybeh@gmail.com); Ooi Boon-Yaik (behwl@utar.edu.my); Khng Xin-Yi (behwl@utar.edu.my)
Paper Title : Comparative analysis of similarity measurement techniques for vibration sensor data: A comprehensive study
Date : February 21st, 2024

NOTIFICATION OF ACCEPTANCE

Dear Dr Woan Lin Beh,

On behalf of the International Conference on Advanced Materials and Applied Sciences (IConMAS 2024) Committee, we are pleased to inform you that your abstract:

ID: (IConMAS 2024: 021-014)

Title: "Comparative analysis of similarity measurement techniques for vibration sensor data: A comprehensive study"

has been **ACCEPTED** for the conference. Congratulations!

An invoice for your conference fee will be sent to you shortly. Please make payment according to information provided in the invoice in order to confirm your presentation slot.

Please be informed that **FULL PAPER SUBMISSION** must be made by email at iconmas@upnm.edu.my and submit before Friday, April 26th, 2024.

Again, thank you very much for your submission.

Secretariat IConMAS 2024
International Conference on Advanced Materials and Applied Sciences (IConMAS 2024)
Email: iconmas@upnm.edu.my
Website: <https://www.iconmas.com>

THIS IS A COMPUTER-GENERATED DOCUMENT. NO SIGNATURE IS REQUIRED.

APPENDIX B

Abstract for IConMAS 2024

Comparative analysis of similarity measurement techniques for vibration sensor data: A comprehensive study

W L Beh¹, J Y Lee¹, B Y Ooi² and X Y Kh'ng²

¹Faculty of Science, Universiti Tunku Abdul Rahman, 31900 Kampar, Perak, Malaysia.

²Faculty of Information and Communication Technology, Universiti Tunku Abdul Rahman, 31900 Kampar, Perak, Malaysia.

ABSTRACT

Abstract. In the domain of vibration analysis, precise evaluation of similarity among sensor data is crucial for various applications ranging from fault detection to structural health monitoring. Time series similarity search is a method used to identify the identical pattern within two sets of time series data, finds widespread utility in clustering, anomaly detection, and forecasting. In real-world scenarios, vibration data are often vast, intricate, and noisy, with adjustments in time, amplitude, and phase shifting direct influence on search outcomes. Through a systematic evaluation, various distance measurement methods including Euclidean distance, Dynamic Time Warping, Fast Fourier Transform, Symbolic Aggregate Approximation, and Matrix Profile are performed under diverse conditions such as frequency shifting, amplitude scaling, state change, and noise. The comparative study encompasses not only quantitative assessments of accuracy but also considerations of computational efficiency and robustness. The findings reveal Matrix Profile generally outperforms classic measures like Euclidean distance, Dynamic Time Warping, and Fast Fourier Transform in accuracy, but performs poorly compared to Symbolic Aggregate Approximation. While Matrix Profile exhibits shorter computational time than Symbolic Aggregate Approximation, it slightly extends beyond other classic measures. Thus, Matrix Profile presents competitive advantages among distance measurement methodologies. By providing a comprehensive examination of similarity measurement techniques, this study equips practitioners with valuable insights for informed decision-making in vibration sensor data analysis, ultimately contributing to advancements in fault diagnosis, condition monitoring, and predictive maintenance in various engineering domains.

Keywords: Vibration sensor data, Similarity measurement techniques, Comparative analysis

Instruction: Presentation Scope (Please bold/underline ONE only) :

Pure Mathematics

Applied Mathematics

Computational Mathematics

Statistics & Applied Statistics

Operational Research

Mathematics Education

Engineering & Industrial Applications

APPENDIX C

Supervisor Comments on Originality Report Generated by Turnitin

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1 of 1




FACULTY OF SCIENCE

Full Name(s) of Candidate(s)	Lee Jia Yee
ID Number(s)	19ADB02132
Programme / Course	Statistical Computing and Operations Research
Title of Final Year Project	Exploring Distance Measures for Time Series Data: A Comparative Analysis

Similarity	Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)
Overall similarity index: 8 % Similarity by source Internet Sources: 6 % Publications: 6 % Student Papers: 2 %	Overall similarity low
Number of individual sources listed of more than 3% similarity: 0	No comment
Parameters of originality required and limits approved by UTAR are as follows: (i) Overall similarity index is 20% and below , and (ii) Matching of individual sources listed must be less than 3% each , and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.



 Signature of Supervisor
 Name: Dr. Beh Woan Lin
 Date: 14/4/2024

 Signature of Co-Supervisor
 Name: _____
 Date: _____

APPENDIX D

Summary Page of Originally Report

EXPLORING DISTANCE MEASURES FOR TIME SERIES DATA: A COMPARATIVE ANALYSIS

ORIGINALITY REPORT

8%	6%	6%	2%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	In-seok Lee, Seung Hwan Park, Jun-Geol Baek. "Random-forest-based real-time contrasts control chart using adaptive breakpoints with symbolic aggregate approximation", Expert Systems with Applications, 2020 Publication	1%
2	webthesis.biblio.polito.it Internet Source	1%
3	Chaochen Hu, Zihan Sun, Chao Li, Yong Zhang, Chunxiao Xing. "Survey of Time Series Data Generation in IoT", Sensors, 2023 Publication	1%
4	link.springer.com Internet Source	<1%
5	www.researchgate.net Internet Source	<1%
6	core.ac.uk Internet Source	<1%
7	acikbilim.yok.gov.tr	