Project 2:
Time Series

Report Created by: Avi Tombak

avitomba@buffalo.edu

University at Buffalo

CSE454 Fall 2021

11/19/2021

Table of Contents

Summary

This project processes a time series dataset, in particular, a deterministic stationary time series of synthetic control data provided by the University of California Irvine. This project is implemented using a Matlab program. The time series data set is processed by using two representation techniques, piece-wise aggregate approximation (PAA) and symbolic aggregate approximation (SAX). Using these representation techniques, classification on the dataset samples is performed. These classifications are done using Euclidean and Manhattan distance methods. These classification results are then analyzed via confusion matrices.

Keywords: Time Series, Dataset, Sample, Piece-wise Aggregate Approximation, Symbolic Aggregate Approximation, Representation, Classification, Euclidean Distance, Manhattan Distance, Confusion Matrix, Matlab

Dataset

The dataset used in this project is provided by the University of California Irvine.

https://archive.ics.uci.edu/ml/datasets/Synthetic+Control+Chart+Time+Series

It contains six-hundred samples of time series representations for synthetically generated control charts. These samples are classified into six distinct classifications, normal, cyclic, increasing trend, decreasing trend, upward shift, and downward shift. The dataset is organized with each class grouped together, starting with one-hundred normal samples, one-hundred cyclic samples, so on and so forth. This dataset is stored in an ASCII file with six-hundred rows and sixty columns, where each row signifies a sample in the dataset, and each column signifies an observation within a sample. Therefore, each time series sample consists of sixty observations.

Process for Working with Time Series

The process for working with time series datasets draws many parallels to the engineering design process. In particular, this process can be broken down into five distinct steps. Figuring out the problem, reading in the data, data preprocessing, data sampling and representation, and data analysis. Each step can be summarized as follows:

1) Figure out the problem

- Derived from project handout
    - Explore time series representation and classification behaviors
    - Utilize the synthetic control dataset

2) Read in the data

- This can be done utilizing a numerical matrix in a Matlab program, in particular one of size 600x60 for this dataset

3) Data preprocessing

- The dataset can be standardized to have a mean of zero and standard deviation of one
- The individual classes can be pulled from the entire dataset
- Each sample can be assigned a class label to associate with its intended classification

4) Data sampling and representation

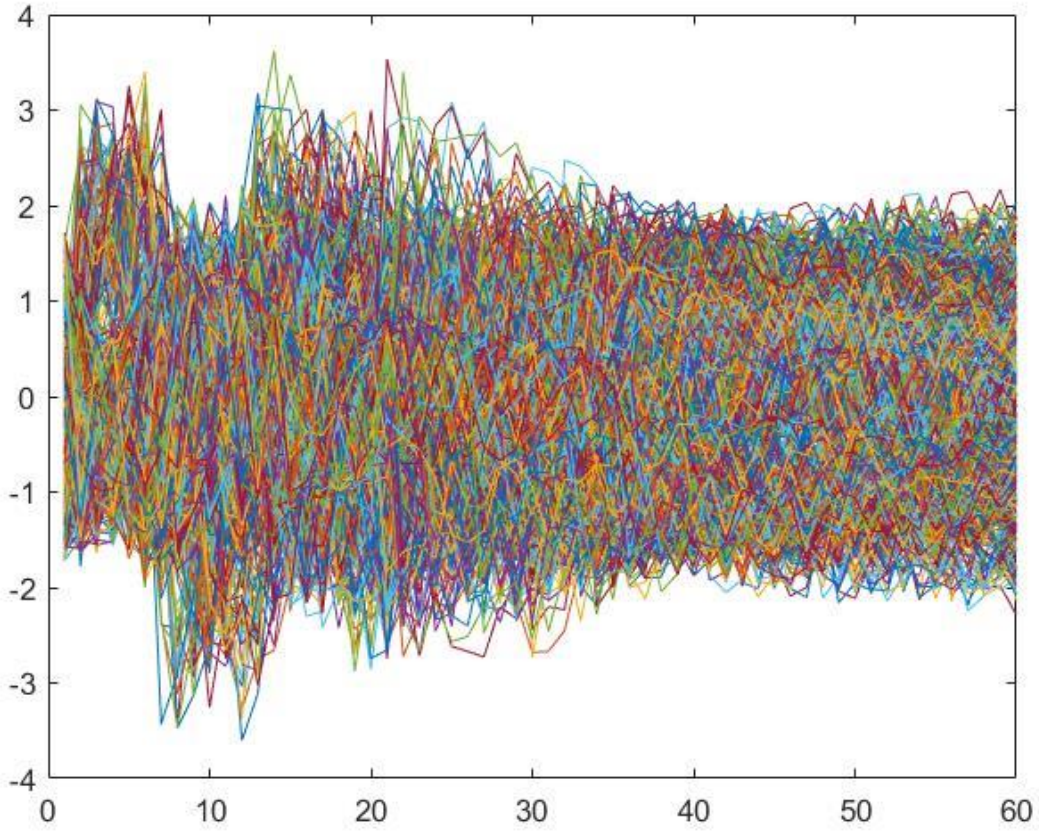
- Two representation techniques are to be used, PAA and SAX
- Data can be sampled in order to establish a training and testing set
    - Where the testing set corresponds to the original dataset
    - Where the training set corresponds to a selected subset of the original dataset

5) Data analysis

- Classification can be done on the samples utilizing distance calculation methods
    - in particular both Euclidean and Manhattan distances
- Classification accuracy can be measured with a structure known as a confusion matrix

Original Time Series Plots

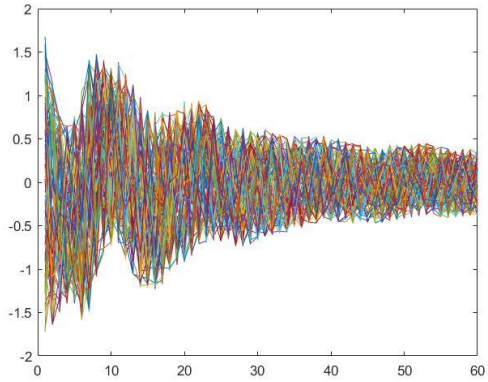
Plot of all six-hundred samples in the dataset:



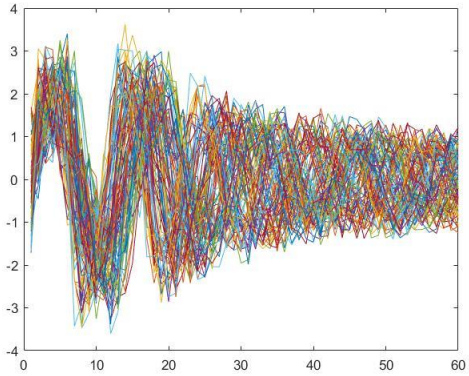
Original Time Series Plots

Plots of each distinct class and its one-hundred samples:

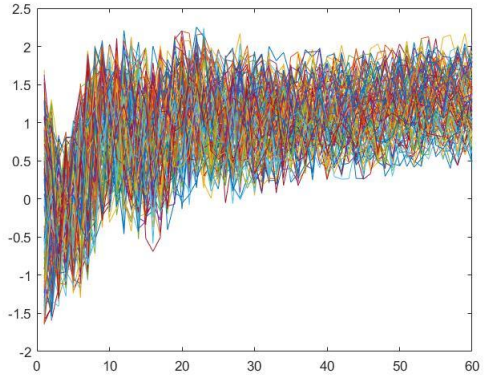
Normal:



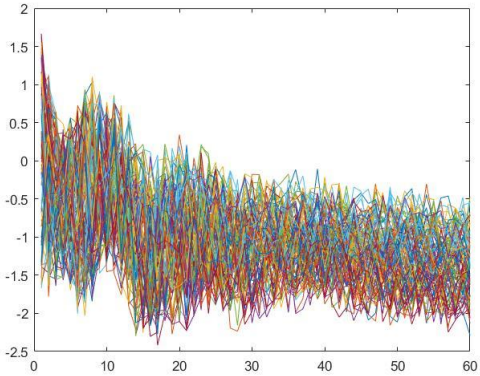
Cyclic:



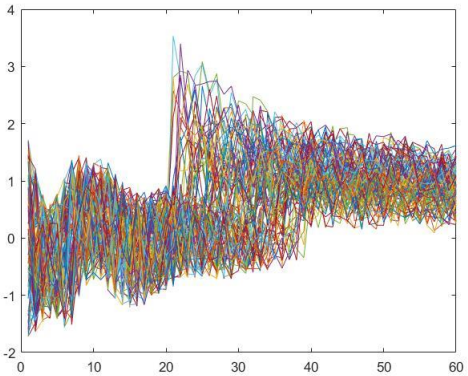
Increasing Trend:



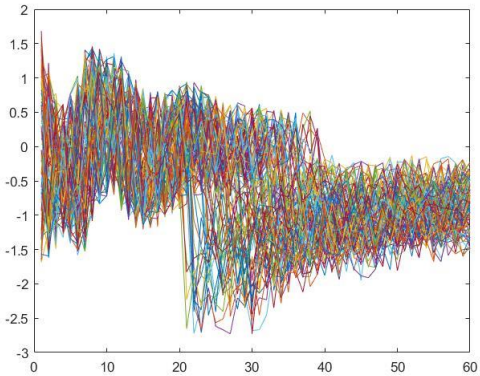
Decreasing Trend:



Upward Shift:



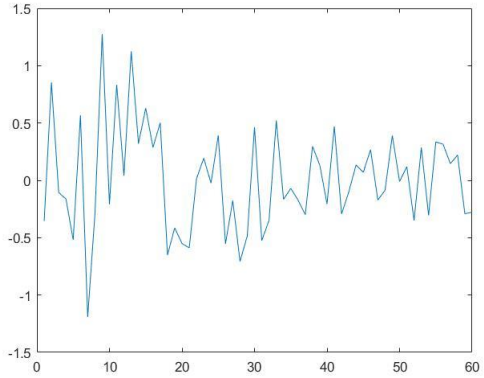
Downward Shift:



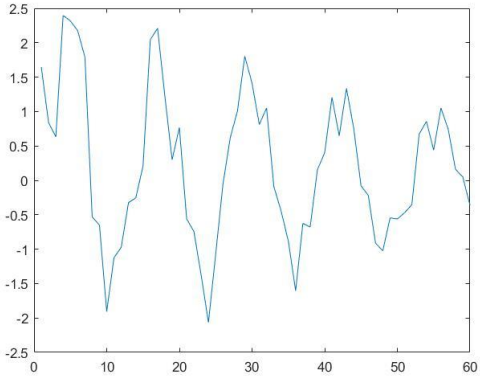
Original Time Series Plots

Plots of an individual sample from each class:

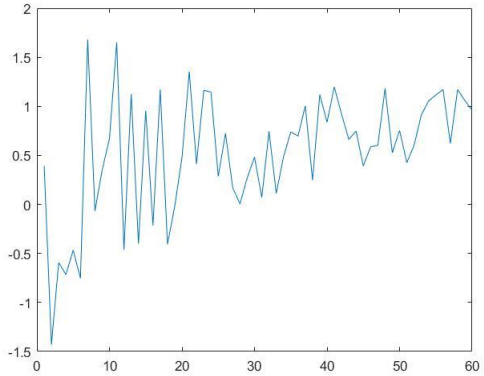
Normal:



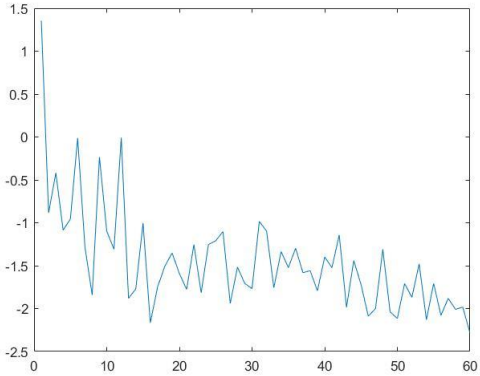
Cyclic:



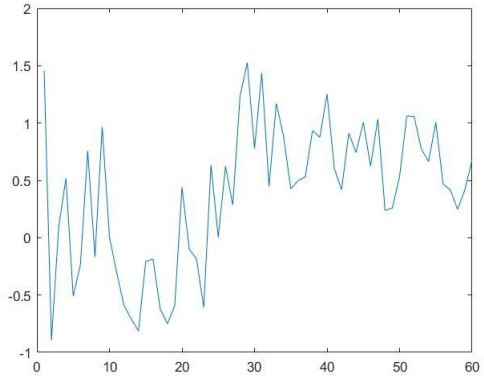
Increasing Trend:



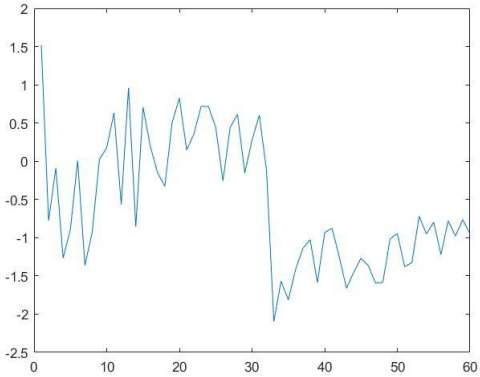
Decreasing Trend:



Upward Shift:



Downward Shift:



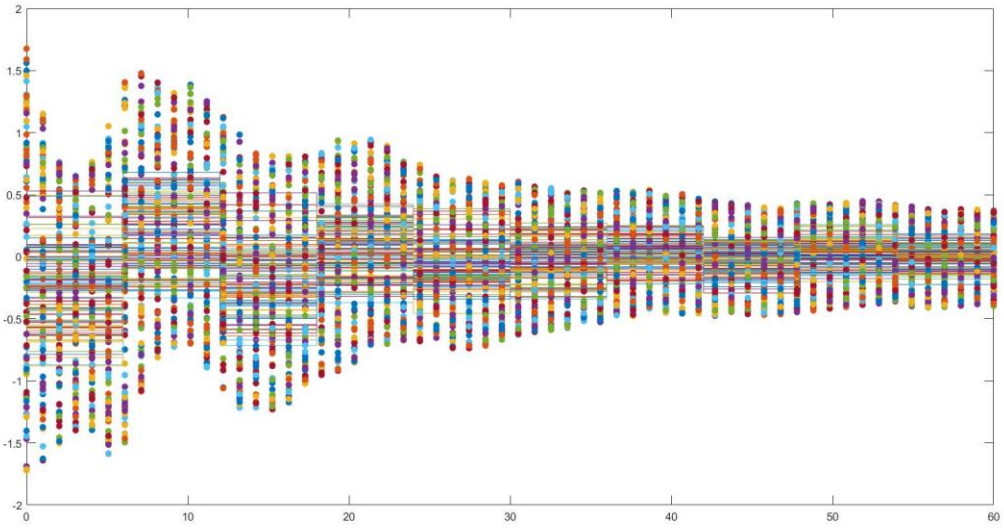
PAA Approach and Implementation

Piece-wise aggregate approximation (PAA) is a representation technique that reduces data based on aggregate values within piece-wise regions corresponding to a time series. It requires the use of expert knowledge of the dataset, including the number of samples and number of observations for the dataset. With this expert knowledge and segmentation of the time series, the samples can be simplified with an approximation of each segment.

In this implementation, the segment count chosen is ten, meaning that for the sixty observations within a sample, every five is aggregated into a distinct piece-wise region. The PAA representation technique was applied to both the testing (original) and training datasets.

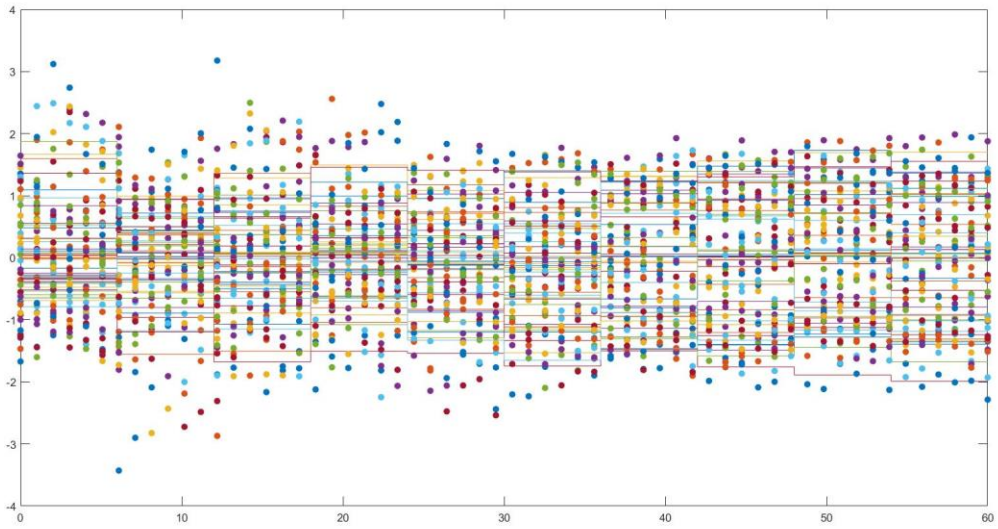
PAA Plots

PAA of Testing Dataset:



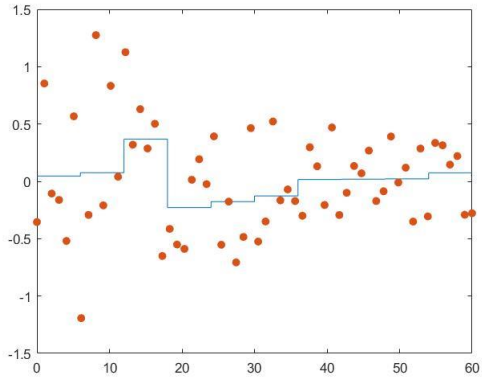
PAA Plots

PAA of Training Dataset:

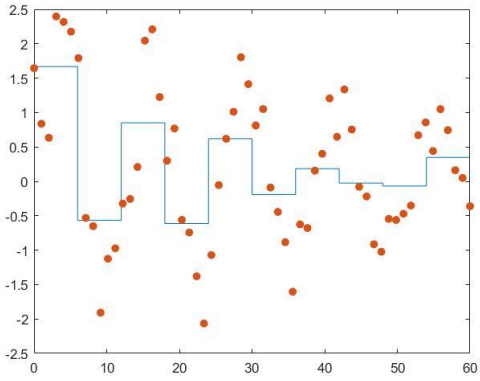


PAAs of individual samples of each class:

Normal PAA:



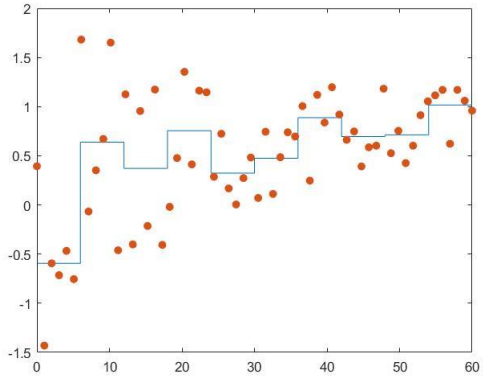
Cyclic PAA:



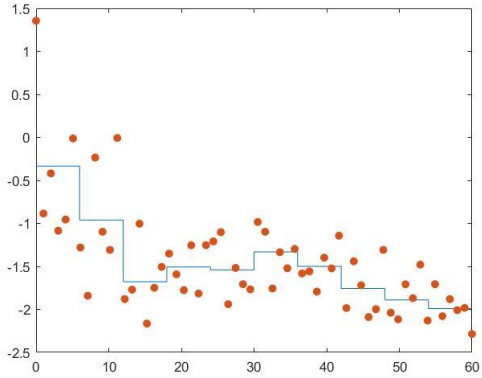
PAA Plots

PAAs of individual samples of each class continued:

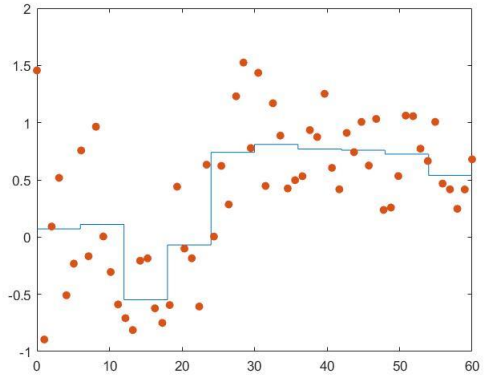
Increasing Trend PAA:



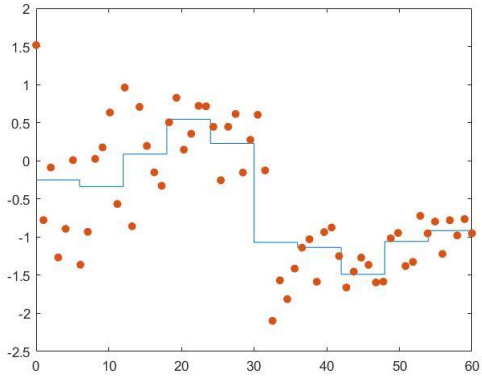
Decreasing Trend PAA:



Upward Shift PAA:



Downward Shift PAA:



SAX Approach and Implementation

Not Available.

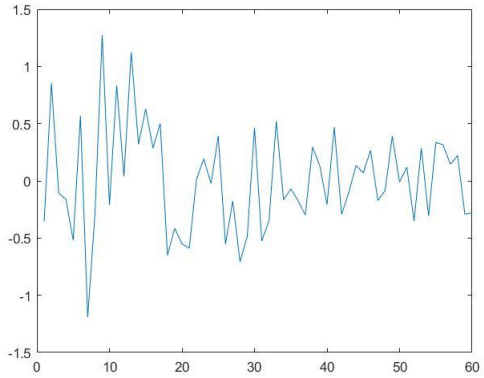
SAX Plots

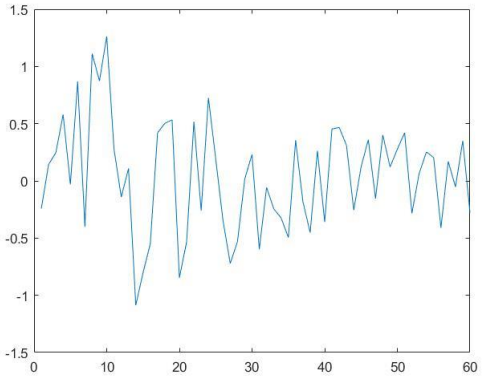
Not Available.

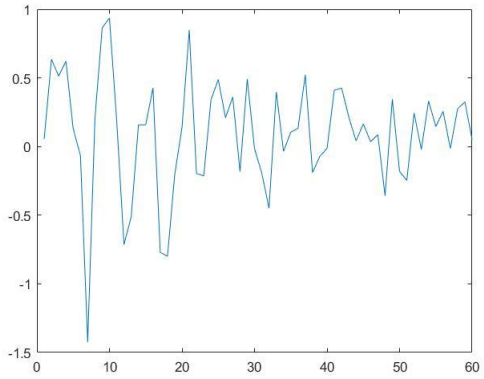
Training and Testing Data Generation

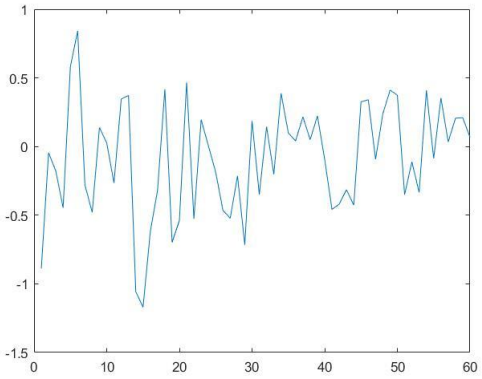
Since classification processing was to be done on the entire dataset, the original dataset was used as the testing dataset. The training dataset was derived from taking selected samples of the testing dataset matched with their correct classification label. The method of selecting these training samples was to take one from each class at a corresponding index. Overall seven indices were selected, therefore forty-two total samples were used in the training set. The selected indices for each class were chosen randomly. The specific training time series samples for each class are as follows:

Normal Training Samples:



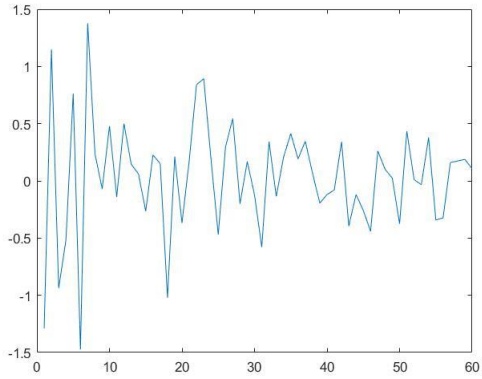


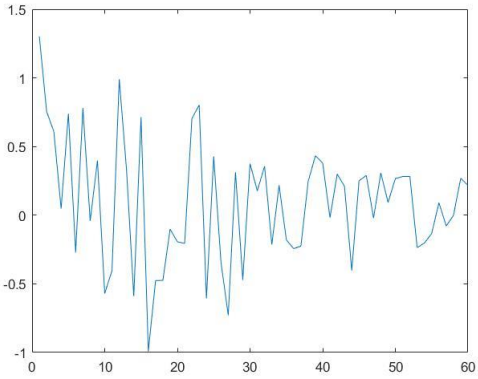


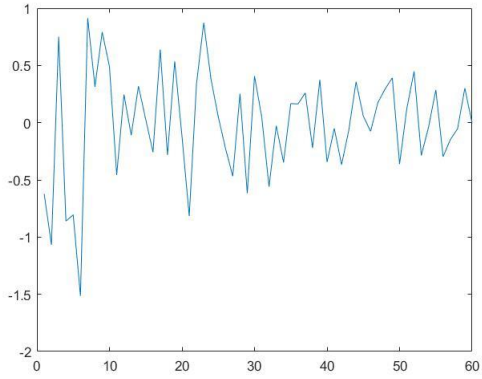


Training and Testing Data Generation

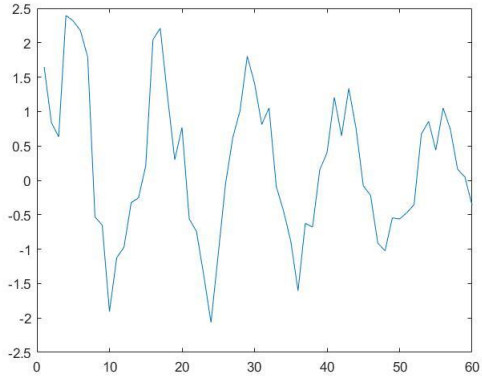
Normal Training Samples Continued:

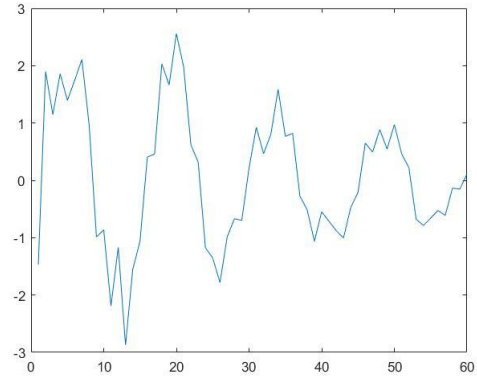






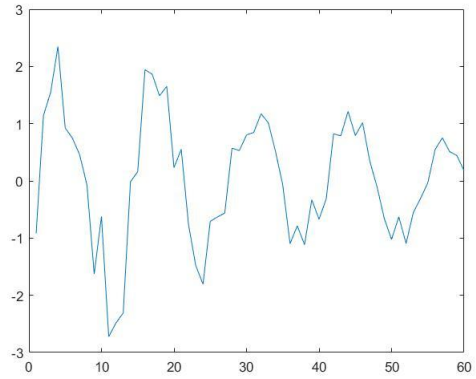
Cyclic Training Samples:

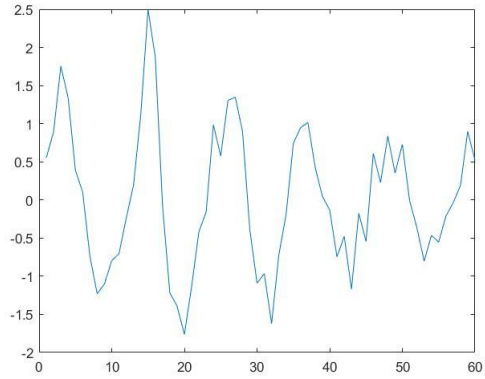


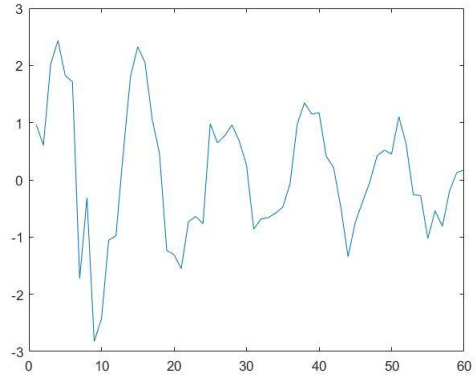


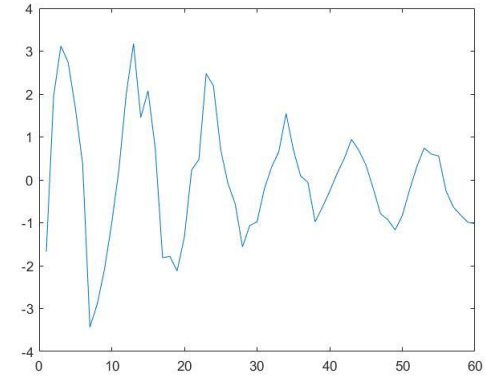
Training and Testing Data Generation

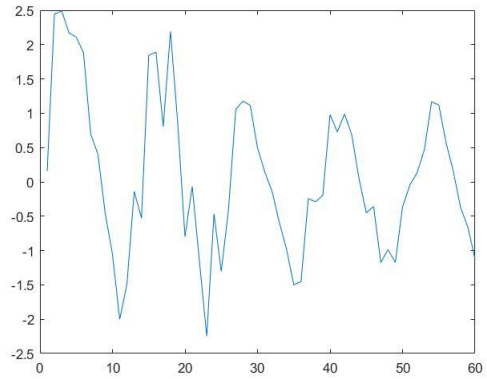
Cyclic Training Samples Continued:





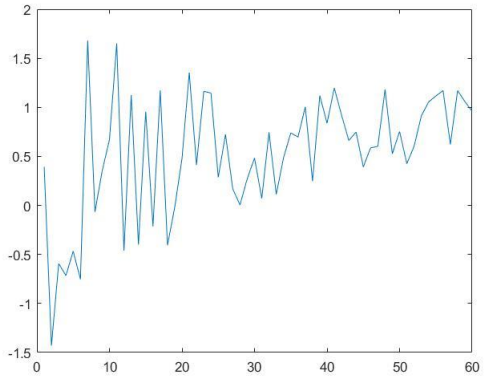


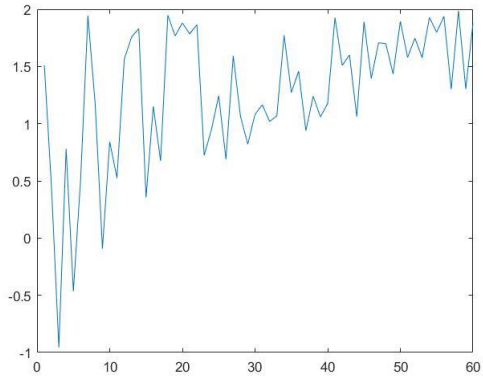


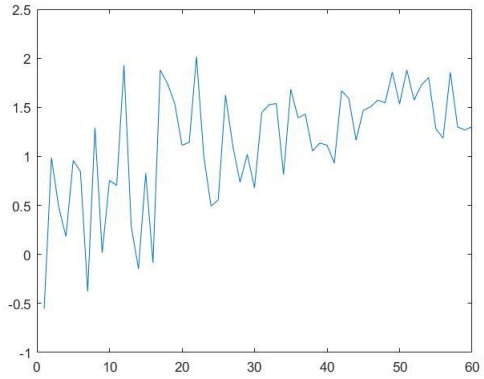


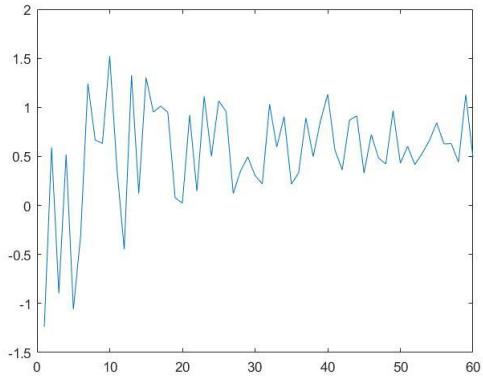
Training and Testing Data Generation

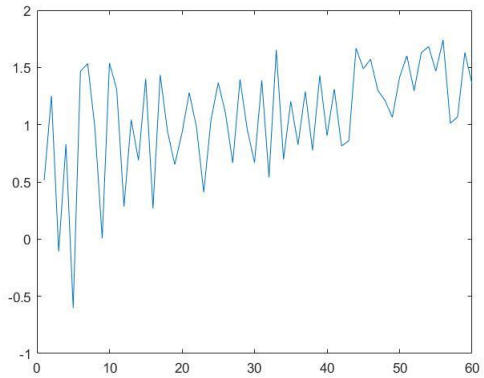
Increasing Trend Training Samples:

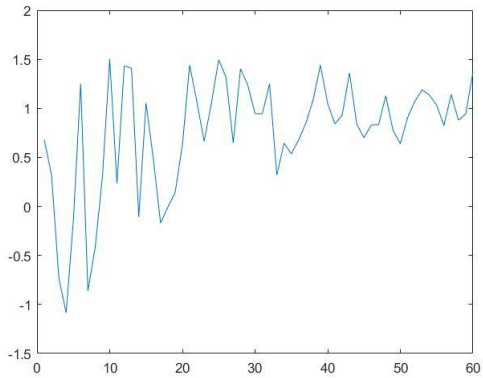






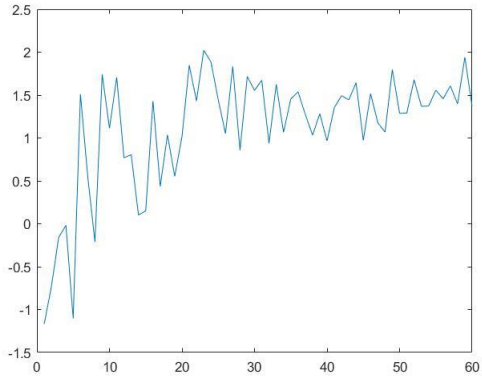




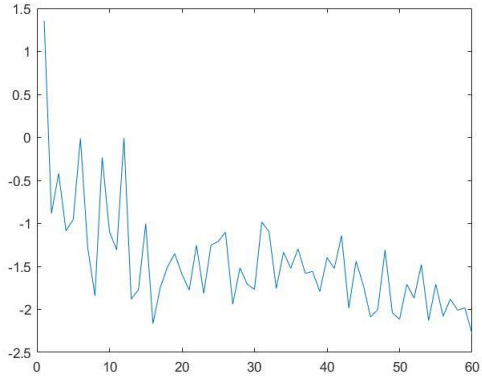


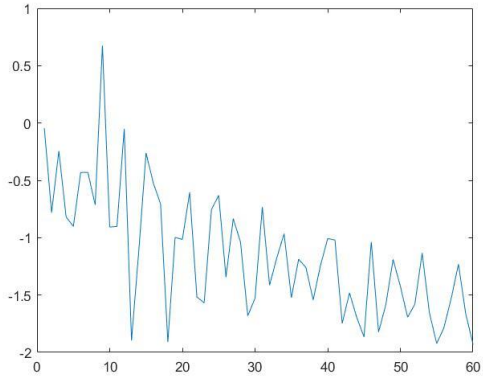
Training and Testing Data Generation

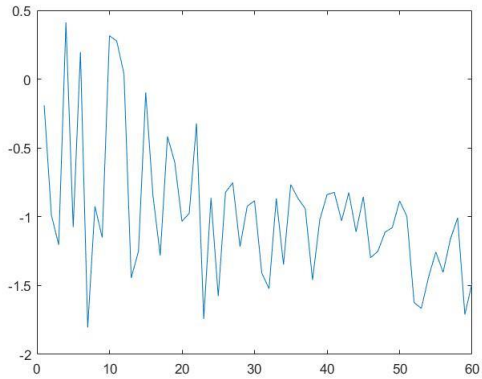
Increasing Trend Training Samples Continued:

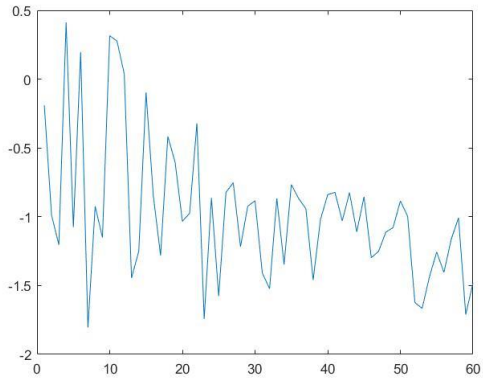


Decreasing Trend Training Samples:



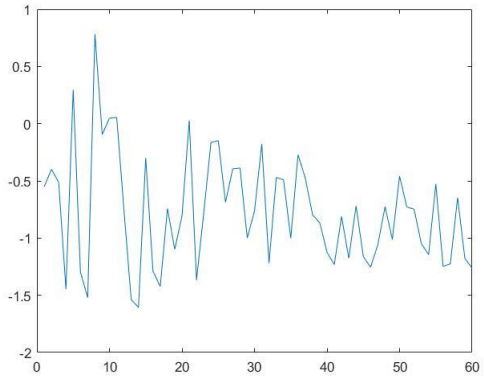


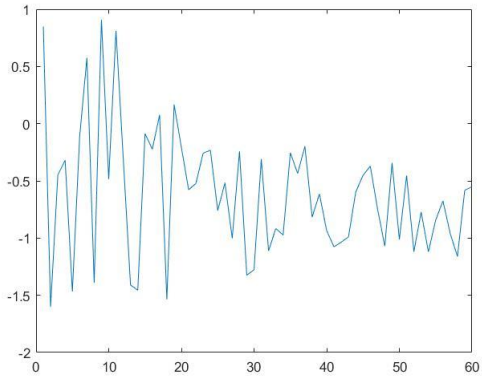


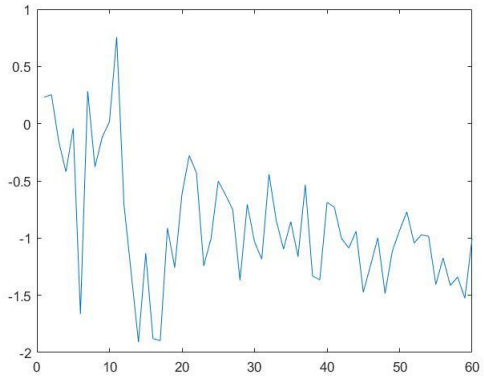


Training and Testing Data Generation

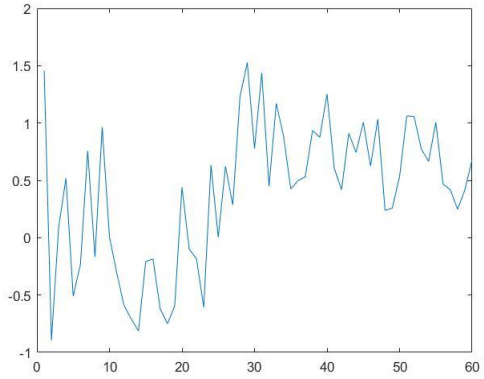
Decreasing Trend Training Samples Continued:

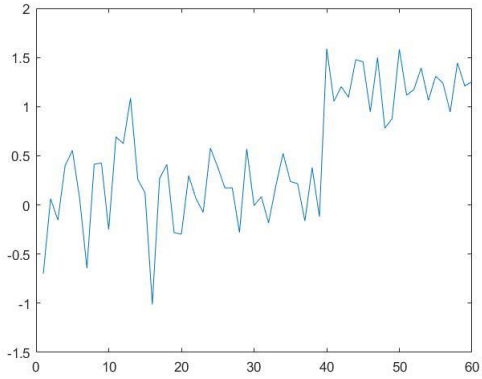






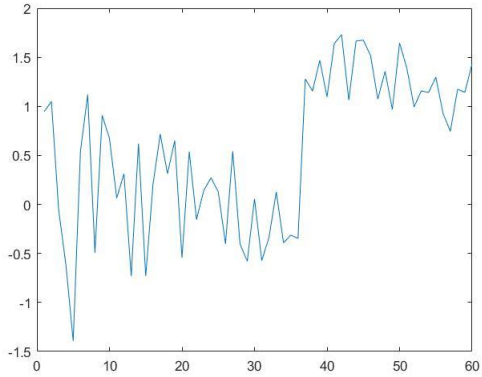
Upward Shift Training Samples:

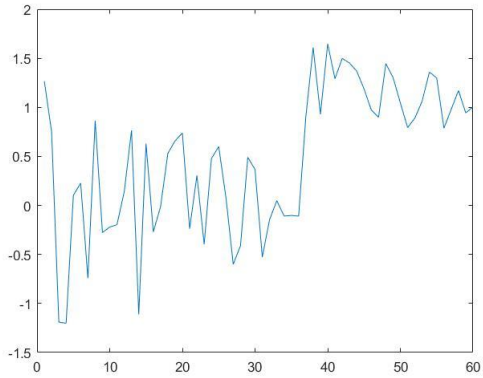


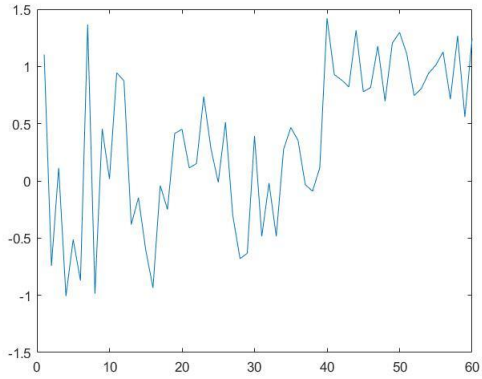


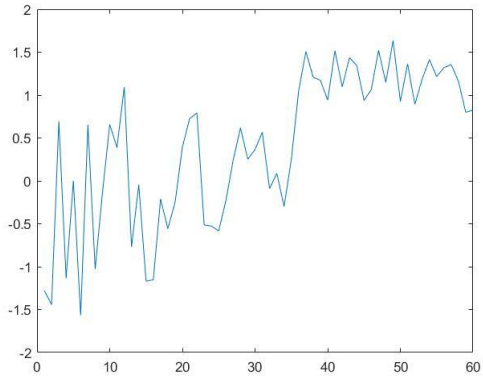
Training and Testing Data Generation

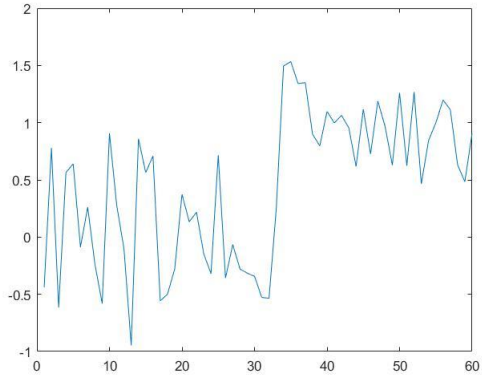
Upward Shift Training Samples Continued:





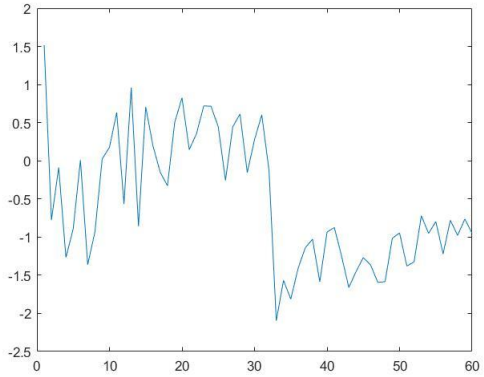


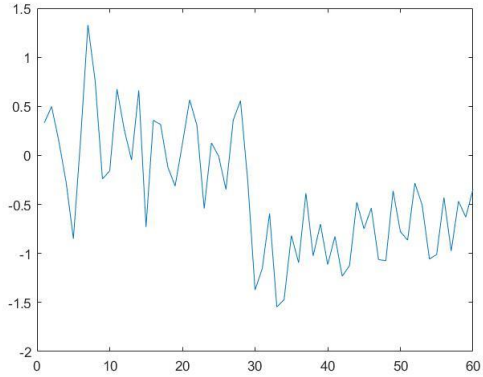


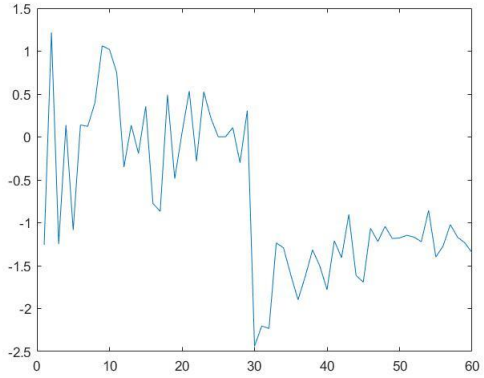


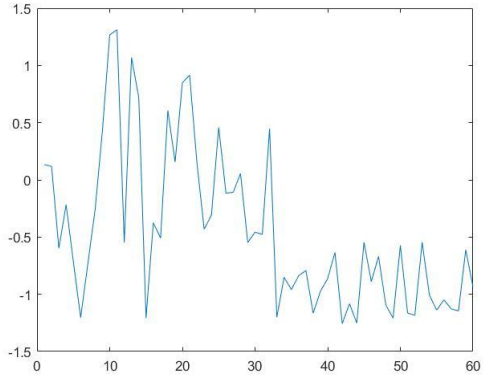
Training and Testing Data Generation

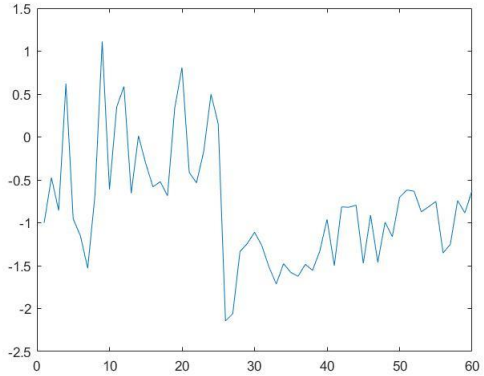
Downward Shift Training Samples:

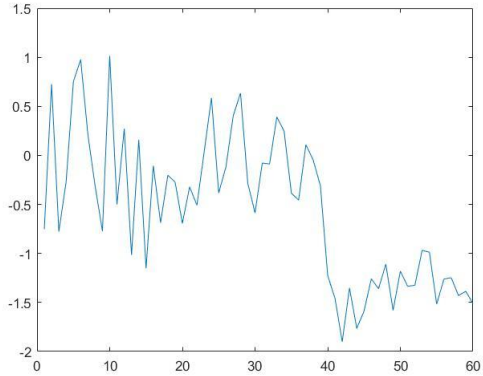






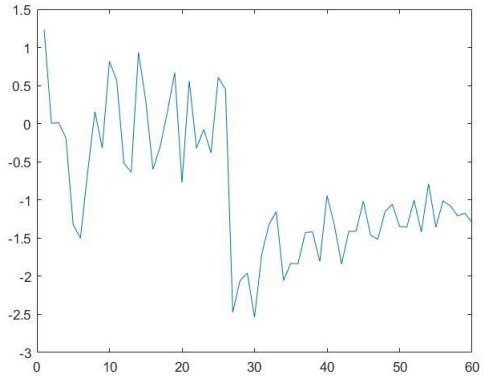






Training and Testing Data Generation

Downward Shift Training Samples:



By utilizing seven samples of each class for the training dataset, more variation is captured within the training set, leading to more accurate classification results. Overall the ratio of training samples to testing samples is forty-two to six-hundred, so the training dataset is seven percent of the testing dataset.

Classification Process

The original dataset had six distinct classes, each with one-hundred samples. With this time series processing, classification inferences can be made based on comparing the training data and testing data. This can be done by comparing the distance between training samples and testing samples. Distance can be calculated in a variety of methods, in this project Euclidean distance and Manhattan distance were used.

Euclidean: $d= \sqrt{(\Sigma((s[i]-r[i])^2))}$

Euclidean: $d= \Sigma(|(s[i]-r[i])|)$

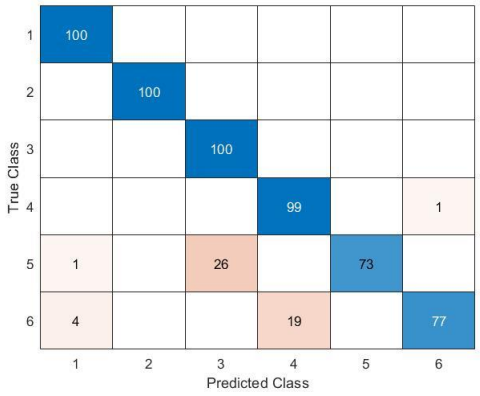
where s and r are two time series or samples, in this case one from the training set, one from the testing set.

Every sample from the training set is classified with the same label as the closest distant sample from testing set. With this method, the time series processing can predict the classes of the training set samples.

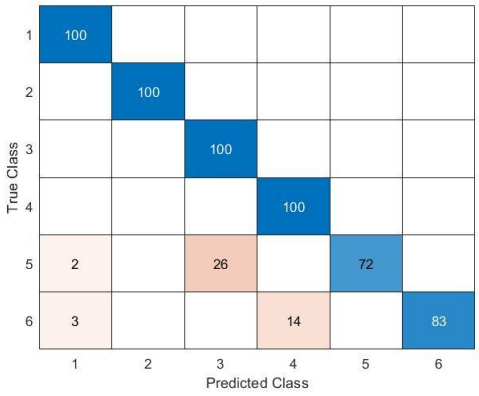
Results

Classification on the original dataset and the PAA dataset using the testing dataset and distance methods was performed. The accuracy of this class prediction process on the original dataset was found to be 91.5% for Euclidean distance and 92.5% for Manhattan distance. The accuracy of this process on the PAA dataset was found to be 93% for both Euclidean and Manhattan distances, with slightly different classification results. This can be visualized with the following confusion matrices:

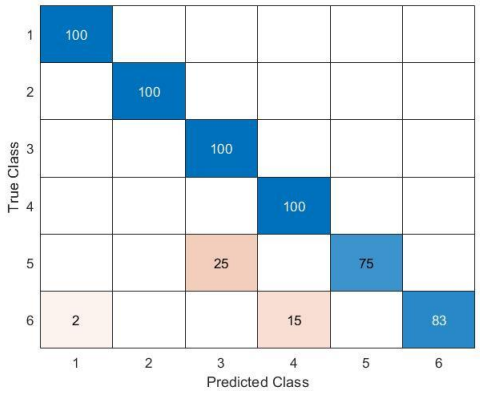
Original Dataset Euclidean Distance:



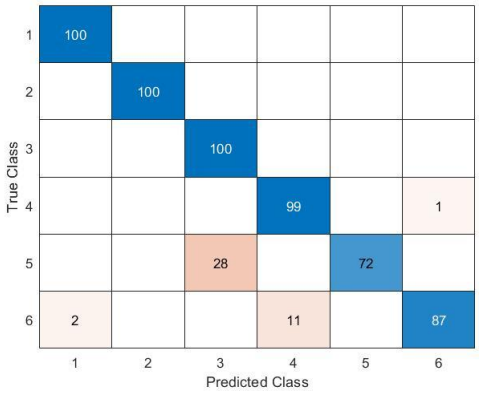
Original Dataset Manhattan Distance:



PAA Euclidean Distance:



PAA Manhattan Distance:



The variety and volume of samples in the training dataset allowed for a higher accuracy than a training set with fewer samples, however finding more diverse samples for each respectively class could further improve accuracy. Two common misclassifications were between upward shift and increasing trend, and downward shift and decreasing trend. Since the time series do have similar trends, it is understandable how this misclassification could occur, a potential solution would be to introduce more samples of these four classes into the training set so that more distinctions between the classes could be interpreted in the classification process.