國立臺灣大學管理學院資訊管理學系

碩士論文

Department of Information Management
College of Management
National Taiwan University

Master Thesis

基於自編碼器的動態管制界線偵測製程與量測異常

Autoencoder-based Dynamic Control Limits for Process and Sensor Anomaly Detection

張鎧

Kai Chang

指導教授：李家岩 博士

Advisor: Chia-Yen Lee, Ph.D.

中華民國 111 年 8 月

August 2022

doi:10.6342/NTU202203045

摘要

隨著數據量的增長以及技術的成熟，現今各個行業都開始仰賴各種人工智慧及資料科學相關的技術來幫助解決各種問題，而在擁有大量資料並且器材昂貴的製造業中，更為準確的量測以及進一步的預測更是需要資料科學的技術輔助，其中異常偵測又為製造業最常探討的議題。異常偵測旨在透過機台數值的分析來得知機台的健康狀況並且進一步進行機台維護策略，藉此取代過往只能透過人工檢查機台的方式來節省時間並減少不必要的成本，但過往異常偵測的議題往往圍繞在機台本身，每當機台數值發生異常便認為機台可能需要進行維護，實際上造成數值異常的原因有另一大主因，便是量測異常，也就是負責偵測數值的感測器本身發生了異常，此時需要維護的不是機台本身而是感測器。因此，本研究以自編碼器為基礎提出了一個能夠線上動態偵測製程與量測異常的 PSAD 架構，並且透過一系列的實驗設計來嘗試證實架構的有效性及穩固性。本研究貢獻在於能夠偵測到過往較少探討的量測異常情形並且同時對兩種異常進行線上動態偵測，以幫助各種使用到感測器進行數值量測的實際場域做出更好的維護決策。

關鍵詞：*故障預測與健康管理*、*製程異常*、*量測異常*、*動態管制界線*、*異常偵測*、*深度學習*

doi:10.6342/NTU202203045

Abstract

With the growth of data volume and the maturity of technology, various industries have begun to rely on various artificial intelligence and data science-related technologies to help solve various problems. In the manufacturing industry with a large amount of data and expensive equipment, more accurate measurement and further prediction require technical assistance from data science. Among this, anomaly detection is the most frequently discussed topic in the manufacturing industry. The purpose of anomaly detection is to know the health status of the machine through the analysis of the machine value and further implement the machine maintenance strategy to replace the manual inspection of the machine in the past to save time and reduce unnecessary costs. However, in the past, the issue of anomaly detection often revolved around the machine process itself. Whenever the machine value was abnormal, it was considered that the machine might need to be maintained. In fact, there is another major reason for the anomaly. That is, the sensor responsible for detecting the value itself has an anomaly. At this time, it is not the machine itself but the sensor that needs to be maintained. Therefore, based on the autoencoder, this research proposes a PSAD architecture that can dynamically detect process and sensor anomalies online, and try to verify the effectiveness and robustness of the architecture through a series of experimental

doi:10.6342/NTU202203045

designs. The contribution of this research lies in the ability to detect sensor anomalies that have been less discussed in the past and perform online dynamic detection of two anomalies at the same time, so as to help various practical fields that use sensors for numerical measurement to make better maintenance decisions.

Keywords: *Prognostic and Health Management, Process Anomaly, Sensor Anomaly, Dynamic Control Limits, Anomaly Detection, Deep Learning*

doi:10.6342/NTU202203045

Acknowledgements

求學的一路上我很幸運，小時候的教育是我好奇心的開始，到了國高中逐漸理解了讀書這件事情的樂趣，大學在政治大學學會了如何批判思考，在資管系學會了何為資訊、何為管理，也開始奠定了未來想發展的方向；最後，在台大資管所的訓練使我真正的領會到各個領域的關聯及背後運作機制，同時產學的經驗使我知道資訊技術在實際場域的使用情形。

感謝指導教授李家岩老師在這兩年的教導，著實是充實的兩年，不只是在於學術上，在處事上、待人上以及各種大小事都在老師身上學到了許多，如果沒遇到老師一定不會有這麼精彩的兩年，也不會遇到實驗室的同學及學長姊學弟妹，另外也要特別感謝在口試給予我指導的三位口試委員許嘉裕教授、莊皓鈞教授及魏志平教授，在口試過程給了我許多建議讓我看到在論文上的不足，使得我能進一步完善此篇論文。

也十分感謝在實驗室遇到的大家，學長姊在我什麼都不懂的情況下給予了許多幫助，學弟妹的加入讓實驗室更熱鬧，也更多活動，同屆的各位也在彼此遇到困難時互相扶持，幫助我度過了無數個難關，最重要的是實驗室給了我許多歡樂時光，感謝大家。

最後謝謝我爸媽從沒阻止我做任何決定，讓我一路上成為了我想成為的人，也希望這篇論文能帶給後人一些什麼那就好了！

doi:10.6342/NTU202203045

List of Contents

doi:10.6342/NTU202203045

List of Tables

doi:10.6342/NTU202203045

List of Figures

doi:10.6342/NTU202203045

doi:10.6342/NTU202203045

Chapter 1. Introduction

1.1 Background and Motivation

In the manufacturing industry, one of the most important issues is often the repair and maintenance of machines. And one of the most frequently used approaches is *Prognostics and health management (PHM) (*Mobley, 2002*)*. PHM is an approach to system life-cycle support that seeks to reduce inspection and time-based maintenance through accurate monitoring, incipient anomaly detection, and prediction of impending anomalies. When it comes to PHM, control charts are most often used to monitor machine values. Like (Blue et al., 2012) monitors the converted machine health indicators through the EWMA control chart.

During the operation of the machine, there are maybe two situations that cause the value to be abnormal. The most common situation is process anomaly. That is to say, in the process of machine production, the machine responsible for production may gradually decline or malfunction. At this time, *preventive maintenance (PM)* is required. In the field of process anomaly detection, such as (Kwak & Kim, 2021) uses the deep learning model to perform process anomaly detection of multiple channels; Another situation is sensor anomaly, which means that it is not that there is a problem in the manufacturing process of the machine, but the sensor that detects the value causes a

doi:10.6342/NTU202203045

value deviation. At this time, we usually need to replace the sensor instead of PM. However, it is difficult for us to judge whether this anomaly is a sensor anomaly. So, in practice, all anomalies are usually regarded as process anomalies. As long as the value is abnormal, we will perform PM in the past cases. This often makes us unable to completely solve the anomalies caused by the sensor and waste a lot of maintenance costs.

It can be seen from the above that it is currently difficult to deal with sensor anomalies, Therefore, this research proposes a *Process and Sensor Anomaly Detection* (*PSAD*) method to try to online separately detect process anomalies and sensor anomalies, and through different simulation data and various experiments to test the robustness of this method.

1.2 Research objective

The research objective of the proposed method is to monitor process anomalies and sensor anomalies online in practice.

Process anomaly means that the machine process has an anomaly different from the outlier, which usually leads to the instability of the machine value. Therefore, the process anomaly in this paper refers to the high variation situation that still exists after the outlier is removed. On the other hand, Sensor anomaly means that there is not a

doi:10.6342/NTU202203045

problem with the machine process, but the sensor itself that is responsible for measuring the value has an anomaly, which usually causes the machine value to continue to shift in a certain direction.

According to practical experience, sensor anomalies are happening all the time in the manufacturing industry. However, the health status of the sensor is often tested manually. This research hopes to decompose the simulated signal data into the process-affected part and the sensor-affected part. Finally, control them separately online and retrain their control limits as new data is collected.

Because it is difficult to obtain the data of sensor anomalies in the past, this research is based on the process anomaly data simulated in the past literature, and on top of this, we attempt to simulate the data affected by both process anomalies and sensor anomalies. Finally, through different experiments to test whether this method can have a good control effect in different situations.

1.3 Research Architecture

The rest of the research is organized as follows. Chapter 2. goes through the literature related to this research on different methodologies, including health indicators, dynamic control limits, and sensor anomalies from three perspectives. In Chapter 3. , the PSAD framework is proposed. Then the assumption and methods used in the PSAD

doi:10.6342/NTU202203045

framework will be demonstrated. Chapter 4. is divided into two scenarios. Scenario 1 tests the feasibility of the method through simulation data and demonstrates its actual monitoring situation. In scenario 2, based on the past literature, more complex anomaly data is simulated, and the effectiveness and robustness of the method are confirmed by various experiments representing different situations. In Chapter 5. , we discuss the conclusions drawn from the experiments and possible future research directions.

Chapter 2. Literature Review

In this chapter, we present the relevant past literature on three topics relevant to this study, respectively. Section 2.1 discusses the establishment of machine health indicators in the field of PHM. In section 2.2 , we refer to previous papers discussing the dynamic adjustment of control limits. Additionally, the sensor anomaly problem, which was less discussed in the past, is mentioned in section 2.3 .

2.1 Health Indicators

2.1.1 Statistics-based Approaches

The health status of the machine is not all intuitively obtained from the sensor value. Therefore, in practice, it is often necessary to convert the value into a health

doi:10.6342/NTU202203045

indicator in various ways. Control the health of the machine through health indicators. Next, this part will introduce various methodologies used in the past on PHM issues.

Chen and Blue (2009) use *Generalized moving variance (GMV)* as a health indicator. GMV is generated by a series of conversions. First, calculate the moving variance and covariance matrix for the *Status Variable Identification (SVID)* of each data, and then calculate the determinant of each matrix. To consider the moving covariance only, the influence of moving variance should be removed. Thus, another matrix, which depends solely on the relationships among SVIDs, is then used as the health indicator. Finally, they use the *Exponentially Weighted Moving Average (EWMA)* control chart to control the GMV health indicator according to (Yeh et al., 2006).

Blue et al. (2012, 2014) introduced a hierarchical monitoring scheme based on the original GMV research (Chen & Blue, 2009). They summarize the data in terms of step-based statistics by calculating the *Coefficients of Variation (CV)*. After the summarization, the temporal data matrix of one wafer becomes a CV matrix. Next, they employ the hierarchical clustering method with correlation-based similarity to bring together a highly correlated matrix of SVIDs. After they get Dendrogram which is used to display the agglomerative clustering results, sensors are divided into different groups. To get each group's information, they use *Principle Component Analysis (PCA)* (Johnson & Wichern, 2014) to obtain the *first principal component (*1_{st} PC*)* of each

doi:10.6342/NTU202203045

group. Finally, the concept of GMV can be again applied to generate an overall tool condition through calculating the GMV of each group's $1_{st}\ PC$, and get the final health indicator.

Chen et al. (2019) use the *partial least squares (PLS)* for feature selection. Based on the PLS model, the *Variable Importance for Projection (VIP)* statistic from (Wold et al., 1993) is used for feature selection. After using PLS-VIP as a dimensional reduction approach, they choose $1_{st}\ PC$ to be the tool health indicator. Finally, the EWMA control chart was employed for abnormality detection of the PLS score.

2.1.2 Machine Learning-based Approaches

Due to the advancement of technology, more and more approaches based on machine learning are now applied in the field of PHM like (Meng & Li, 2019) mentions.

Among them, the machine learning-related model that has been particularly strong in the PHM field in recent years is the autoencoder model (An & Cho, 2015) that will also be used in this research. Kwak and Kim (2021) regards the reconstruction error generated by the deep convolution autoencoder as the machine health indicator, and uses this to control multiple channels at the same time. Then use three different anomaly detection algorithms like *isolation forest (iForest)(*Liu et al., 2008*), local outlier factors*

doi:10.6342/NTU202203045

*(LOF)(*Breunig et al., 2000*),* or *one-class SVM (OCSVM)(*Schölkopf et al., 1999*)* to judge the degree of anomaly.

2.2 Dynamic Control Limits

When it comes to PHM, control charts are usually one of the main methods of anomaly detection. Begin with the *Shewhart* chart (Montgomery, 2020) that controls some basic statistics like mean, standard deviation, etc. Then the *cumulative sum (CUSUM)* chart (Lucas, 1985) appeared because it may be difficult to detect small deviations simply by using the Shewhart chart, so it is convenient to use CUSUM to accumulate small changes for monitoring. Next, the emergence of the EWMA control chart allows us to take past data into account and further control. After so many evolutions, there is still an important topic to discuss about the control chart, that is, the update of the control limits. At this time, the concept of dynamic control limits was proposed. According to (Zhang & Woodall, 2015) and (Tighkhorshid et al., 2019), using fixed control limits to monitor health indicators may lead to change in pre-specified in-control *Average Run Length* (ARL_0) overtime and increasing probability of false alarms. Therefore, the Dynamic Probability Control Limit (DPCL) can be used to improve the performance of the control chart.

doi:10.6342/NTU202203045

There are several pieces of research about how to determine DPCL. One of the most commonly seen methods is to adjust the control limits by time-varying sample sizes like (Shen et al., 2013) and (Morales et al., 2021). Another method is to calculate the degree of risk of different objects like (Zhang & Woodall, 2015) and (Tighkhorshid et al., 2019).

The above studies all address the importance of dynamically adjusting control limits for the control chart. To achieve the same effect, this research will obtain new control limits by retraining the model after collecting new data, so that the model can perform online dynamic control of the machine and the sensor.

2.3 Sensor Anomaly

As mentioned in the first part, because of the difficulty, little research has been done to detect sensor anomalies in the past. However, in practice, the reasons for the abnormal values of the machine are often not only due to the manufacturing process. Therefore, there are still research discussing related issues.

Lee et al. (2020) propose a framework for In-Line Predictive Monitoring (ILPM), which describes the process of detecting potential anomalies and prealarm using off-line and in-line data. The framework is divided into two phases. The first phase is

doi:10.6342/NTU202203045

process parameter monitoring (PPM), which decides whether the prealarm is triggered or not. The second phase is equipment parameter monitoring (EPM), which decides whether the anomaly is caused by equipment parameters or not. The concept of confirming whether the equipment parameters are abnormal is similar to detecting sensor anomalies. Both are to confirm whether the anomaly of the machine is caused by reasons other than the process.

Liu et al. (2019) believes that the machine and the sensor itself will continue to be degraded over time. The paper describes the degradation process using the Wiener process, which has been shown to successfully capture degradation characteristics in real-life machines. (Ye & Xie, 2015), (Zhai & Ye, 2017). This paper introduces a CBM policy with Kalman filter as the main method in the case where the machine is accompanied by a degraded sensor. A maintenance cost model is obtained by adding new measurement data and updating the degradation level of the machine and sensor. After constructing a preliminary model, the optimal maintenance policy is obtained by minimizing the long-run cost rate. Although this paper doesn't focus on anomaly detection, this paper explores the problem of sensor degradation. Because the proposed method of simulating degradation problem is more reliable and complex, we will use its simulation method to conduct experiments in Chapter 4.

doi:10.6342/NTU202203045

Table 2.1 Summary of PHM-related literature

	Method	Dynamic Control Limits	Online	Process Anomaly	Sensor Anomaly
(Blue et al., 2012)	GMV	-	✓	✓	-
(Shen et al., 2013)	EWMA	✓	✓	✓	-
(Liu et al., 2019)	Kalman Filter	-	-	✓	✓ (Maintenance cost model)
(Lee et al., 2020)	Bayesian Monitoring	-	✓	✓	✓ (Machine parameter)
(Kwak & Kim, 2021)	CAE	-	-	✓	-
This study (2022)	AE	✓	✓	✓	✓

The above three subsections describe the three core objectives of this research and the related content of past literature research. However, as seen in Table 2.1, the literature rarely combines and discusses these three points in the past. Most of the PHM-related literature does not address issues related to sensor anomalies. A few mentioned measurement anomalies such as (Liu et al., 2019) and (Lee et al., 2020). The former aims to build a maintenance cost model rather than a method that can dynamically detect anomalies online. On the other hand, although the latter mentioned that the anomaly is not only the situation that the machine itself needs to be repaired, but it mainly discusses the adjustment of the machine parameters rather than the problem of the sensor itself. Therefore, it is far from the real sensor anomaly detection. In

doi:10.6342/NTU202203045

conclusion, the main motivation of this study comes from the few previous studies related to sensor anomalies in the field of PHM, and we will combine the above three topics at the same time as the core of this research.

doi:10.6342/NTU202203045

Chapter 3. PSAD Framework

This chapter proposes the PSAD framework. To separate process anomaly and sensor anomaly, this research decomposes the simulation data in time series. After that, use the processed data to train the AE model separately. A trained AE model can be used for producing a vector of the reconstruction error. Finally, we monitor the error with the dynamic control limits. In the case of online control, when enough data is collected, the AE model will be retrained to update the control limits.

We organize out Chapter 3. as follows: In section 3.1 , We present the PSAD framework of this study and explain each part. In section 3.2 , the assumption of the proposed method is initiated. Section 3.3 describes the methodologies used in this study.

3.1 Model Framework

This section proposes the PSAD model framework. The steps of the proposed overall model framework shown in Figure 3.1 are explained as follows.

doi:10.6342/NTU202203045

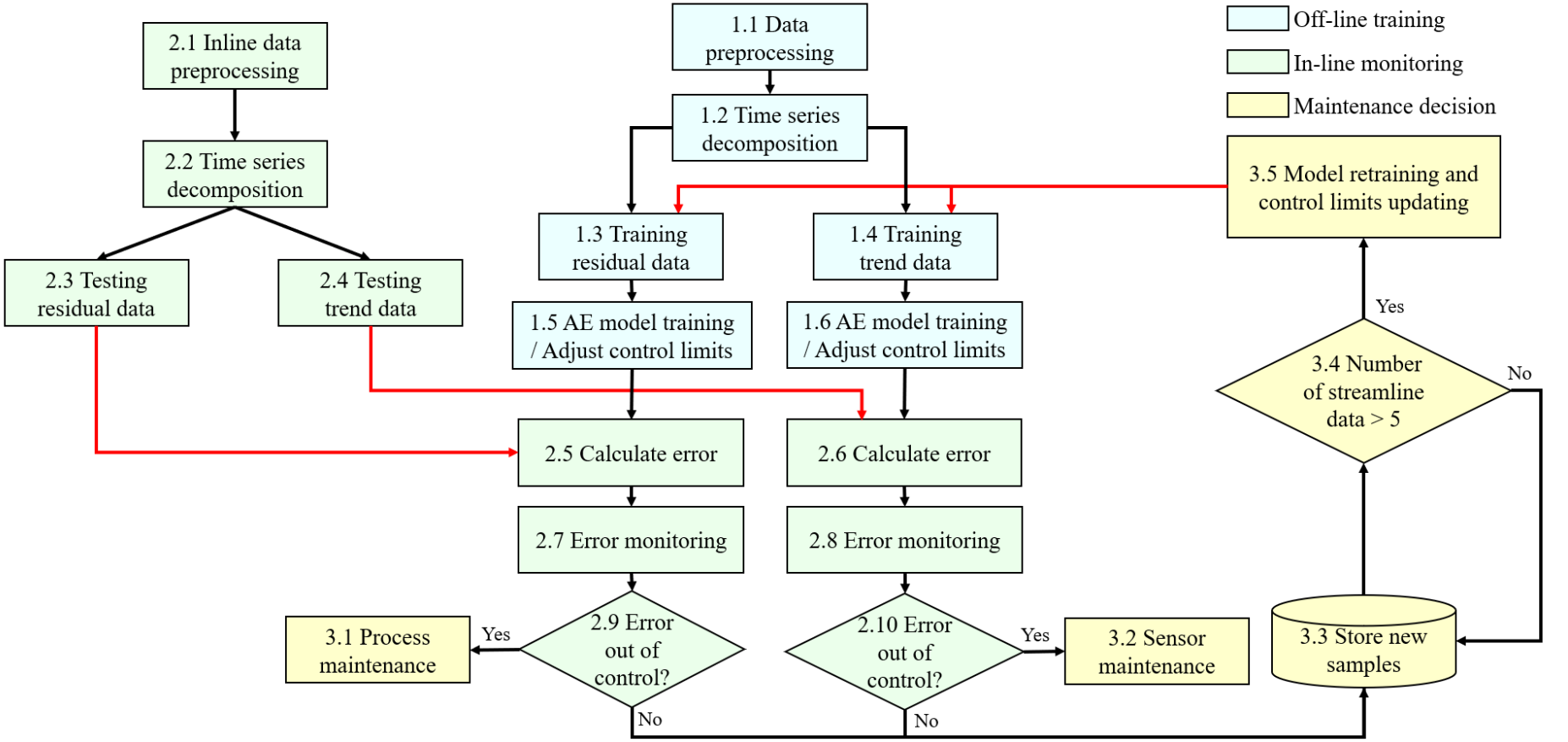


Figure 3.1 PSAD framework

doi:10.6342/NTU202203045

3.1.1 Off-Line Training Module

This module tells how the model uses offline data to train and build the initial AE model and dynamic control limits.

Steps 1.1: The first step collects data from the manufacturing execution system (MES) that was considered normal in the past as initial training data. Collected training data will be preprocessed by removing the outlier, imputing the missing values, identifying the noise, and then dividing the cleaned data into a training data set. The training data set is then cut into several points using a moving time-window strategy (Keogh et al., 2004). It allows us to consider the relationship of the channel and change the dimension of the input into the autoencoder at the same time.

Steps 1.2-1.4: Before training the AE model, to monitor process and sensor anomaly separately, the training data will be decomposed to the residual data representing the process conditions and the trend data representing the sensor conditions.

Steps 1.5-1.6: Residual data and trend data will be used to train an AE model and get a set of reconstruction errors each. The reconstruction error of training data will be used to establish the initial control limits.

doi:10.6342/NTU202203045

3.1.2 In-Line Monitoring Module

This module describes how the inline data is converted into health indicators and monitored by the control limits.

Steps 2.1: When the model completes offline preparation, the inline data will go through the same preprocessing process as the training data and then be thrown into the model.

Steps 2.2-2.4: Like training data, the inline data need to be decomposed to residual data and trend data for testing process anomalies and sensor anomalies.

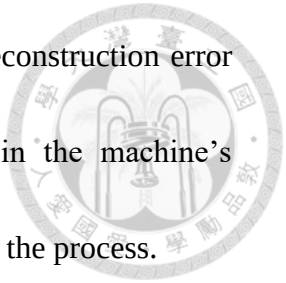
Steps 2.5-2.6: To obtain the health indicators that can be used to monitor, the residual and trend data used in the test will be thrown into the AE model trained in the first module to obtain the reconstruction errors.

Steps 2.7-2.10: The reconstruction error will be monitored by the dynamic control limits and confirmed whether it exceeds the control limits.

3.1.3 Maintenance Decision Module

The last module describes how the model makes decisions about the results of the monitoring and retrains the model with new data.

doi:10.6342/NTU202203045



Steps 3.1: If the monitoring of the residual part detects that reconstruction error exceeds the control limits, it is considered to be an anomaly in the machine's manufacturing process. The machine needs to be used maintained in the process.

Steps 3.2: If the monitoring of the trend part detects that it exceeds the control limits, it is considered to be an anomaly in the machine's sensor. The machine itself does not need to be maintained, instead the sensor needs to be checked or replaced.

Steps 3.3-3.5: When both the residual part and the trend part of the monitoring are not out of control, the data is considered to be the normal data of the machine. Therefore, it will be stored in a database that stores healthy machines' data. When the quantity of points stored in the database reaches 5, these data will be treated as new sampling data and thrown into the original model for retraining. In fact, the amount of data represented by each point depends on the size and the overlapping rate of the sliding window during preprocessing. The number of points that need to be retrained can also be adjusted according to different situations. Finally, after retraining, the AE model and the control limits will be updated to complete the online dynamic control limits process.

3.2 Assumption

Based on domain knowledge, it is believed that the sensor anomaly will cause the value to continue to deviate in one direction and the process anomaly will cause the

doi:10.6342/NTU202203045

machine value to drift randomly, which also means increased variance. At the same time, because the effect of sensor anomaly on the machine value may be very complicated, so we make the first assumption: sensor anomaly will cause the machine value to continue to drift in one direction linearly or non-linearly and process anomaly will cause the variance of machine value to increase.

3.3 Methodology

3.3.1 Time Series Decomposition

The objective of this research is to detect the process anomaly and the sensor anomaly separately. Therefore, in the first step, we need to decompose the data into the part that the manufacturing process affects and the part that the machine's sensor affects. According to the assumption in section 3.2 , we believe that the influence of the manufacturing process on the data value is mainly reflected in the variance, and on the other hand the influence of the sensor on the data value is continuously shifting in the same direction.

Therefore, we can use the time series decomposition approach (West, 1997) to decompose the data into the residual part which represents the random drift caused by the manufacturing process, and the trend part which represents the one-direction drift caused by sensor anomaly. In this research, we use Seasonal-Trend Decomposition

doi:10.6342/NTU202203045

using Loess (STL) (Cleveland et al., 1990). The procedure is in charge of decomposing a sensor signal into three components: a trend component, a seasonal component, and a residual component. We will take the trend component and the residual component for use. When the trend component is abnormal, we judge that the machine is abnormal in the sensor. On the other hand, when the residual component is abnormal, we judge that the machine is abnormal in the process. In addition, outliers usually appear in the residual term. In order to avoid mistaking outliers as anomalies, outliers should be removed before the data is thrown into the model for monitoring.

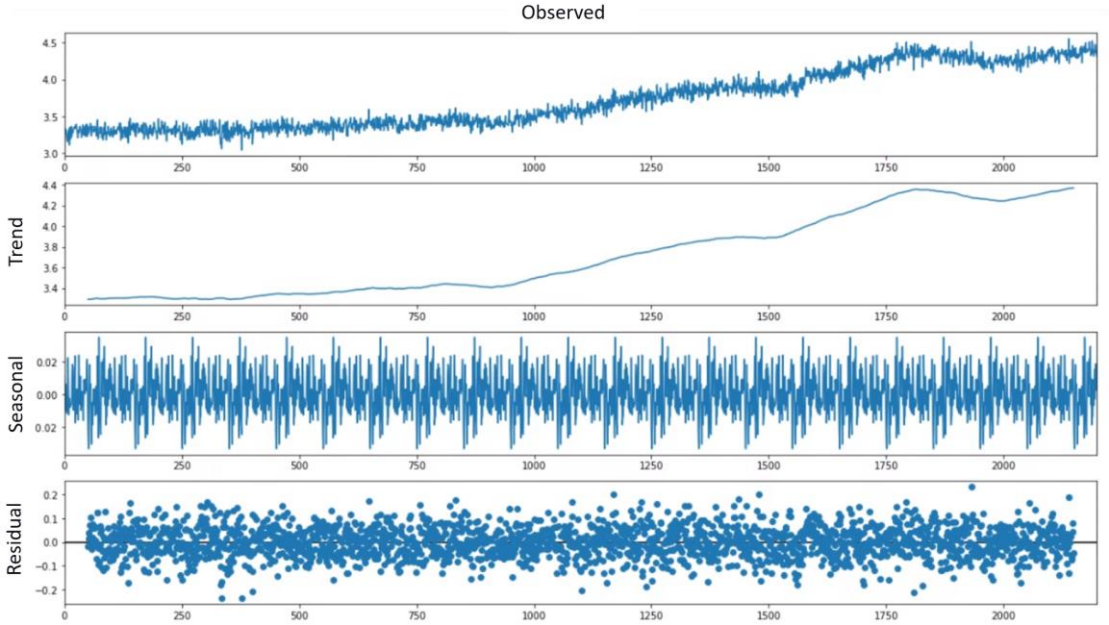


Figure 3.2 Time series decomposition

3.3.2 Autoencoder

Autoencoder has been proven by a lot of literature to perform well in the field of anomaly detection like (Sakurada & Yairi, 2014) and (An & Cho, 2015). After we get

doi:10.6342/NTU202203045

the residual data and the trend data, we need to convert each of them into health indicators. This is where we need the autoencoder. It can help us to obtain the reconstruction error as our health indicators without manual feature engineering. The following introduction and formulation of autoencoder are mainly based on (An & Cho, 2015).

An autoencoder is a neural network that is trained by unsupervised learning, which is trained to learn reconstructions that are close to its original input. When the reconstruction error is smaller, the closer it is to the original training data. An autoencoder consists of two parts, an encoder and a decoder. A neural network with a single hidden layer has an encoder and a decoder as in equation (1) and equation (2), respectively. W and b is are the weight and bias of the neural network and σ is the nonlinear transformation function.

$$h = \sigma(W_{xh} * x + b_{xh}). \quad (1)$$

$$z = \sigma(W_{hx} * h + b_{hx}) \quad (2)$$

$$\|x - z\| \quad (3)$$

The encoder in equation (1) is responsible for converting input *x* to a hidden representation *h* by an affine mapping following a nonlinearity. The decoder in equation (2) is responsible for mapping the hidden representation *h* to the original input space as reconstruction by the same transformation as the encoder. Finally, the difference

doi:10.6342/NTU202203045

between the original input *x* and the reconstruction *z* becomes the reconstruction error in equation (3). The objective function of the training autoencoder is to minimize the reconstruction error. When the training is completed and the testing data is thrown into the autoencoder, the smaller the reconstruction error is, the closer to the training data, which means that it is relatively healthy. Conversely, when the reconstruction error is larger, it means that the degree of data anomaly is larger, thereby achieving the effect of detecting anomalies. We have also considered using CAE or LSTM Autoencoder to replace the most basic autoencoder as an alternative. However, in the experiments of this study, the monitor effect of them did not significantly improve, and improving the deep learning model is not the focus of this paper, so the most basic AE was still used.

doi:10.6342/NTU202203045

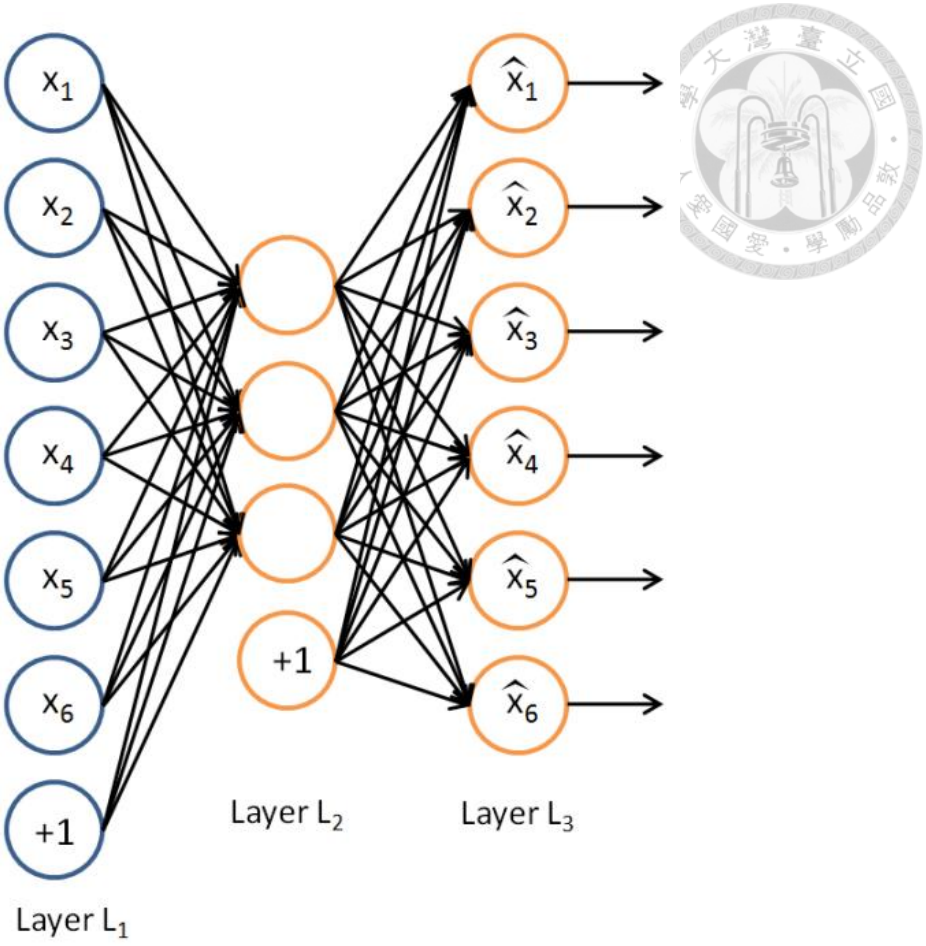


Figure 3.3 Autoencoder (Ng, 2011)

3.3.3 Dynamic Control Limits

After obtaining the reconstruction error, we need a mechanism to confirm whether the reconstruction error belongs to an abnormal level. This study hopes to monitor the anomalies of the data through a more statistical method. Therefore, the method of dynamic control limits is proposed.

We use the reconstruction error from the training data representing the health data as control limits to monitor the reconstruction error from the test data. As a more appropriate control limits range, we choose three times the standard deviation of the

doi:10.6342/NTU202203045

reconstruction error of the training data as the upper and lower control limits. When the reconstruction error of the test data is higher than the upper control limit or lower than the lower control limit, we consider the data to be abnormal. For these control limits to achieve the goal of online dynamic adjustment, the limits will be updated with the retraining of autoencoder.

Chapter 4. Simulation Study and Experiment Result

4.1 Simulation Study and Experiment Design

In order to check the effectiveness and robustness, there are two scenarios that simulate different data separately. The first scenario is mainly used to show the effect of the method and the actual usage. The second scenario mainly uses more complex simulation data and various experiments to confirm the robustness of the method and prove that the method is better than the more traditional baseline method.

doi:10.6342/NTU202203045

4.1.1 Scenario 1

Simulated Data Generation

To demonstrate the feasibility of the method, we conducted a simulation to characterize it and show its actual monitoring situation. The simulation is mainly based on (Kwak & Kim, 2021) and (Lee et al., 2020). Because the exact locations of the normal and abnormal patterns in the simulation data are known in advance, we can assign the normal and abnormal classes to the different channel, in this study each channel represents a value detected by a sensor. So, we can evaluate the ability of the model to distinguish abnormal channels through these channels whose classification is known in advance.

We generated 10000 normal signal data with 20 channels from $N(\mu, \sigma)$ as training data, where μ refers to the mean randomly generated from $N(10, 1)$, and σ refers to the usual variation level of data points, which is also a parameter of the experiment. We then adopted a *sliding window strategy* (Keogh et al., 2004) to consider the channel relationship and multiple time steps simultaneously and to change the input dimensions to fit the AE model. According to (Kwak & Kim, 2021), we set the time window size to 100 with 50% overlapping among the windows. During the experiment, 20% of the training data will be randomly used as validation data

doi:10.6342/NTU202203045

After generating the training data, we generated 2000 signal data with 20 channels from $N(\mu, \sigma)$ as testing data. Most of the parameters and the preprocessing are the same as the training data, except that 3 out of 20 channels are abnormal. The first and third channels have process anomaly, and the second and third channels have sensor anomaly. That is to say, a process anomaly and a sensor anomaly all occurred in the third channel. Our goal is to find the channel in anomaly and find out what kind of anomaly happened to them.

In order to distinguish abnormal and normal channels, process anomaly and sensor anomaly have special patterns in the data respectively.

Process anomaly: As mentioned in the previous assumption, we believe that the anomaly of the process will cause the variation of the machine to become larger in a certain period of time. Therefore, when the channel has a process anomaly, we will change the variation level σ to the process drift parameter p for a certain period of time, which is formulated like:

$$\sigma_t = \begin{cases} \sigma, & for\ t < t_{p_0}\ or\ t \geq t_{p_1} \\ p, & for\ t_{p_0} \leq t < t_{p_1} \end{cases} \quad (4)$$

where t_{p_0}, t_{p_1}, and p are the hyperparameters of the model. The parameter t_{p_0}, t_{p_1} are start and end time of the process anomaly, which are set to 750 and 1000. The process drift parameter p is the variation range for the process anomaly.

doi:10.6342/NTU202203045

Sensor anomaly: As mentioned in the previous assumption, we believe that the anomaly of the sensor will cause the machine value to continue to drift in one direction linearly or non-linearly. Therefore, when the channel has a sensor anomaly, we will add a sensor drift parameter *s* during the data generation process, which is formulated like:

$$\mu_t = \begin{cases} \mu_{t-1}, & if\ \ t < t_s \\ s\mu_{t-1}, & if\ \ t \geq t_s \end{cases} \quad (5)$$

where the parameter t_s is the start time of the sensor anomaly, which is set to half of the total testing time, 1000. The sensor drift parameter *s* is the drift level for the sensor anomaly.

Experiment Design

In order to test the performance of the model in this simulation, there are three independent variables in the experiment, variation level σ, process drift level *p*, and sensor drift level *s*. The parameter setting of σ are 1 and 1.5, which represent the usual variation level of the data. The parameter setting of process drift level *p* are 2 and 3, which represent the degree of variation when process anomaly occurs. The parameter setting of sensor drift level *s* are 1.0002, 1.0003, 1.0005, and 1.001, which represent the degree of drift when sensor anomaly occurs. Figure 4.1 shows the actual data situation for four different sensor drift parameters. From sensor drift level = 1.0002, where the shift is least obvious, to sensor drift level = 1.001, where it is most obvious, if we can call detect it well, it means that this method has a good effect on this type of data.

doi:10.6342/NTU202203045

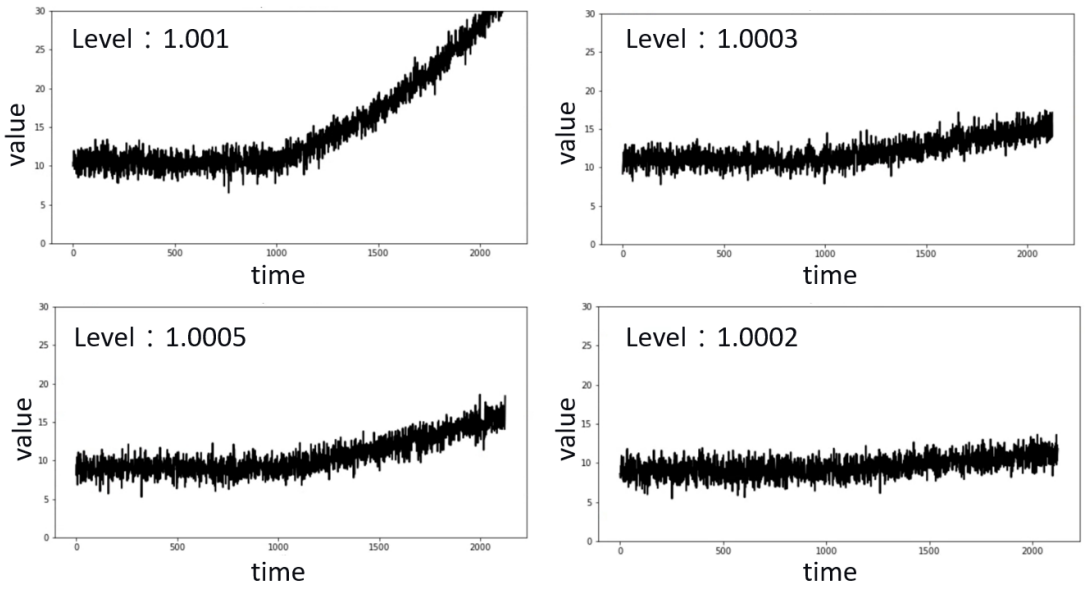


Figure 4.1 Simulation data with sensor anomaly
(sensor drift level: 1.0002, 1.0003, 1.0005, and 1.001).

In addition to the experimental design, we need a baseline for comparison to prove that this method is superior to other methods in the past and not because of insufficient data complexity. The baseline used in scenario 1 is *isolation forest (iForest)(*Liu et al., 2008*).* iForest is an isolation-based algorithm that performs recursive random splits on variables. The algorithm works by dividing all observation points into subspaces, and the number of times required to separate a point is taken as anomaly score. When a point can be identified with only a few steps, it represents an anomaly, because it is easy to be separated from normal data. Table 4.1 shows the algorithm of iForest.

Table 4.1 iForest algorithm (Liu et al., 2008)

Algorithm: $iForest(X, t, \varphi)$
Inputs: X – input data, t – number of trees, φ – sub-sampling size
Outputs: a set of *t iTrees*

doi:10.6342/NTU202203045

1. **Initialize** *Forest*
2. set height limit $l = ceiling(log_2 \varphi)$
3. **for** i = 1 to t **do**
4. $X' \leftarrow sample(X, \varphi)$
5. $Forest \leftarrow Forest \cup iTree(X', 0, l)$
6. **end for**
7. **return** *Forest*

4.1.2 Scenario 2

Simulation Data Generation

Because of the lack of empirical data on sensor anomalies to support the feasibility of the method, we simulated a second, more complex scenario to demonstrate the robustness of the method. The second simulation is mainly based on (Liu et al., 2019), which simulated the degradation of system and sensor by the Wiener process. The concept of sensor degradation is similar to the sensor anomaly situation proposed in this study. The system mentioned in this paper is similar to the definition of our machine process, so the following are all referred to as process

According to (Liu et al., 2019), believes that the machine will continue to degrade over time. The study uses the Wiener process to describe the underlying degradation process. Let stochastic process $X(t)$, $t \geq 0$ denote the associated degradation process over the operating time t, which is expressed as

doi:10.6342/NTU202203045

$$X(t) = X(0) + \lambda t + \sigma B(t) \quad (6)$$

where λ is the drift coefficient, σ is the diffusion coefficient and $B(t)$ is the standard *Brownian motion*. $X(0)$ is the initial degradation level, and $\sigma B(t) \sim N(0, \sigma^2 t)$ stands for the randomness of the degradation process. Without loss of generality, it is assumed $X(0) = 0$. $X(t)$ will represent the degradation process of the machine manufacturing process in this study.

Likewise, the sensor actually suffers from a degradation process. We believe that a degraded sensor will start to drift and measure inaccurately, which is modeled as the Wiener process, that is,

$$S(t) = S(0) + \eta t + \delta B(t) \quad (7)$$

where $S(t)$ is the sensor degradation level at time *t*, η and δ are the drift and diffusion coefficients, respectively. Note that η can be positive or negative, denoting the positive or negative drift. It is also assumed that the system and sensor degrade independently and the degradation parameters are known in advance.

Let $\{Y(t), t \geq 0\}$ denote the process of measurement, which represents the observation obtained by combining the process and sensor degradation state with uncertain noise. Y(t) is given as

$$Y(t) = X(t) + S(t) + \varepsilon \quad (8)$$

doi:10.6342/NTU202203045

where $X(t)$ and $S(t)$ are process and sensor degradation state mentioned above, and ε is the statistically independent and identically distributed measurement error, following normal distribution $\varepsilon \sim N(0, \gamma^2)$ at any time point.

Every point $y_k = Y(t_k)$ represent the observation at time t_k.So we generate 10000 data points with 20 channels as training data according to the following formula:

$$\begin{cases} x_k = x_{k-1} + \lambda(t_k - t_{k-1}) + u_k \\ s_k = s_{k-1} + \eta(t_k - t_{k-1}) + v_k \\ y_k = x_k + s_k + \varepsilon_k \end{cases} \quad (9)$$

where $u_k = \sigma[B(k) - B(k-1)]$ and $v_k = \delta[B(k) - B(k-1)]$. $\{u_k, k \geq 0\}$, $\{u_k, k \geq 0\}$, and $\{\varepsilon_k, k \geq 0\}$ follow statistically independent and identically normal distribution, i.e., $u_k \sim N\left(0, \sigma^2(t_k - t_{k-1})\right)$, $v_k \sim N\left(0, \delta^2(t_k - t_{k-1})\right)$, and $\varepsilon_k \sim N(0, \gamma^2)$. In this study, $t_k - t_{k-1} = 0.01$ and $\gamma = 0.1$.

After generating the training data, we generated 2000 signal data with 20 channels as the testing data. Most of the parameters and the preprocessing are the same as the training data, except that 3 out of 20 channels are abnormal. The first and third channels have process anomaly, and the second and third channels have sensor anomaly, just like scenario 1. In order to distinguish abnormal and normal channels, process anomaly and sensor anomaly have special patterns in the data respectively. During the experiment, the last 20% of the training data will be used as validation data

doi:10.6342/NTU202203045

Process anomaly: As mention above, $X(t)$ denote the state of the machine process. According to our assumption, process anomaly will cause the variance of machine value to increase. Therefore, in the scenario 2, if a process anomaly occurs, we will raise the variation parameter σ in (6).

Sensor anomaly: As mention above, $S(t)$ denote the state of the sensor. According to our assumption, sensor anomaly will cause the machine value to continue to drift in one direction linearly or non-linearly. Therefore, in the scenario 2, if a sensor anomaly occurs, we will raise the drift parameter η in (7).

Experiment Design

doi:10.6342/NTU202203045

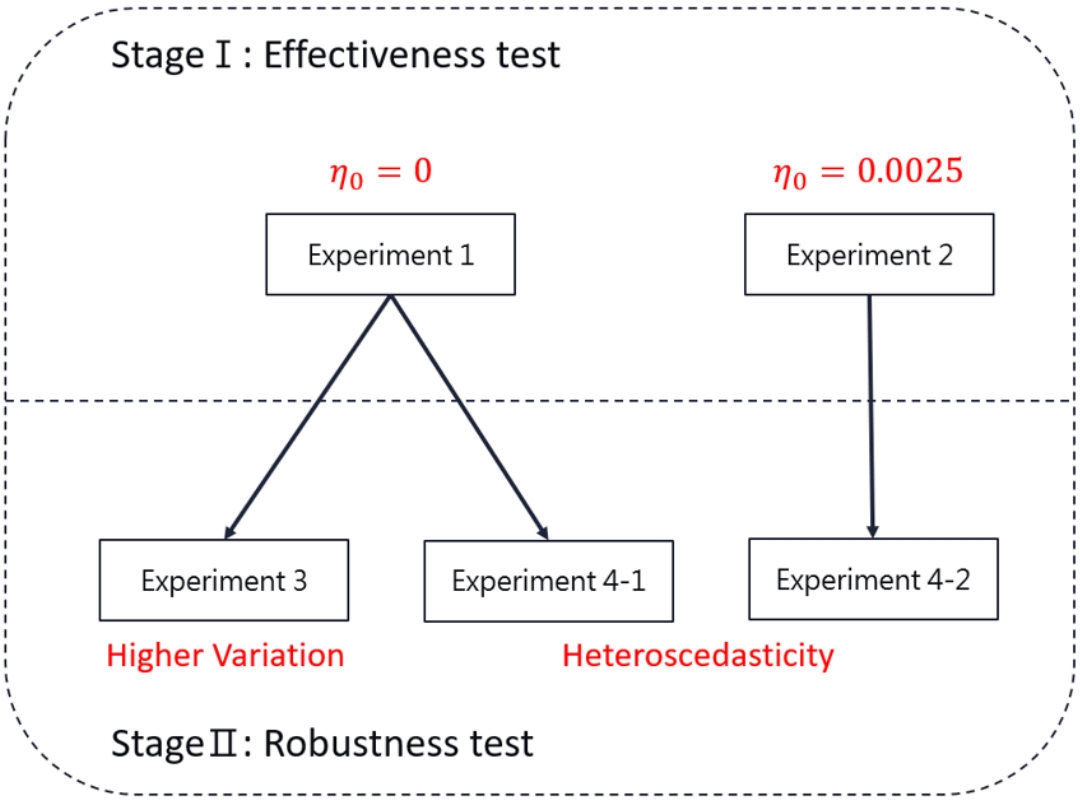


Figure 4.2 Experiment plan of scenario 2

As shown in the Figure 4.2, in order to prove the effectiveness and robustness of the method, four experiments were conducted through two stages respectively.

In the first stage we wanted to demonstrate that the model can indeed detect process and sensor anomalies in more complex data. At this stage, there are two experiments, which respectively represent whether the sensor has degradation situation under normal conditions. The usual drift coefficient of the machine process λ is set to 0.005, the usual variation coefficient of the machine process σ_0 is set to 0.1, and the usual variation coefficient of the sensor δ is set to 0.05.

doi:10.6342/NTU202203045

Experiment 1: The first experiment assumes that the sensor will not degrade under normal circumstances, that is to say, the usual sensor drift level $\eta_0 = 0$. So, the goal of this experiment is to find out the sensor that starts to slowly degrade under abnormal conditions. There are two independent variables in the experiment, abnormal sensor drift parameter η, and abnormal process drift level σ. The parameter setting of abnormal sensor drift parameter η are 0.01, 0.015 and 0.02, which represent the degree of sensor degradation under sensor anomaly. The parameter setting of abnormal process drift parameter σ are 0.5, 0.75, and 1, which represent the degree of variation of the machine under process anomaly. Figure 4.3 shows the actual data situation for three different sensor drift parameters, from sensor drift level = 0.01, where the shift is least obvious, to sensor drift level = 0.02, where it is most obvious.

doi:10.6342/NTU202203045

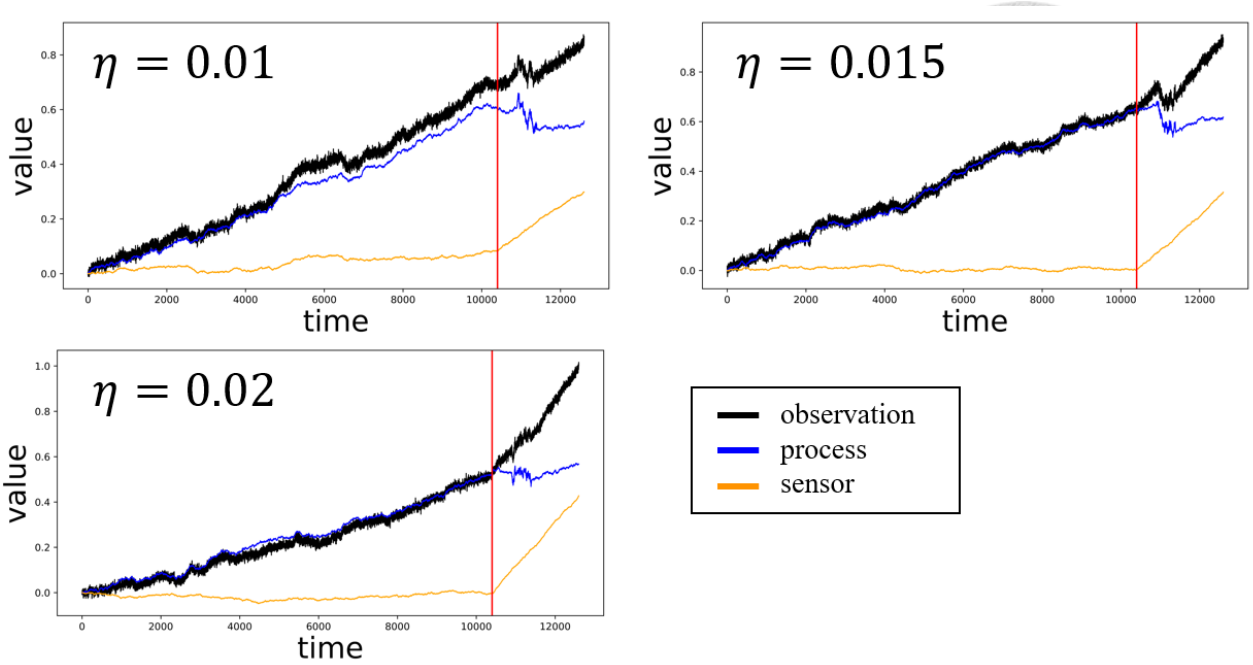


Figure 4.3 Experiment 1 data with different sensor drift levels
(sensor drift level: 0.01, 0.015, and 0.02).
The top line is the observation point, which consists of the process and sensor states below, and the red line represents the time point when the anomaly occurred

Experiment 2: The second experiment assumes that the sensor will slowly degrade under normal circumstances, that is to say, the usual sensor drift level $\eta_0 = 0.0025$. So, the goal of this experiment is to find out the sensor that begins to degrade violently under abnormal conditions. Same as experiment 1, there are two independent variables in the experiment, abnormal sensor drift parameter η, and abnormal process drift level σ. The parameter setting of abnormal sensor drift parameter η are 0.025, 0.05 and 0.075, which represent the degree of sensor degradation under sensor anomaly. The parameter setting of abnormal process drift parameter σ are 0.5, 0.75, and 1, which represent the degree of variation of the machine under process anomaly. Figure 4.4 shows the actual data situation for three different sensor drift parameters, from sensor

doi:10.6342/NTU202203045

drift level = 0.025, where the shift is least obvious, to sensor drift level = 0.075, where it is most obvious. On the other hand, Figure 4.5 shows the actual data situation for three different process drift parameters, from process drift level = 0.5, where the shift is least obvious, to process drift level = 1, where it is most obvious.

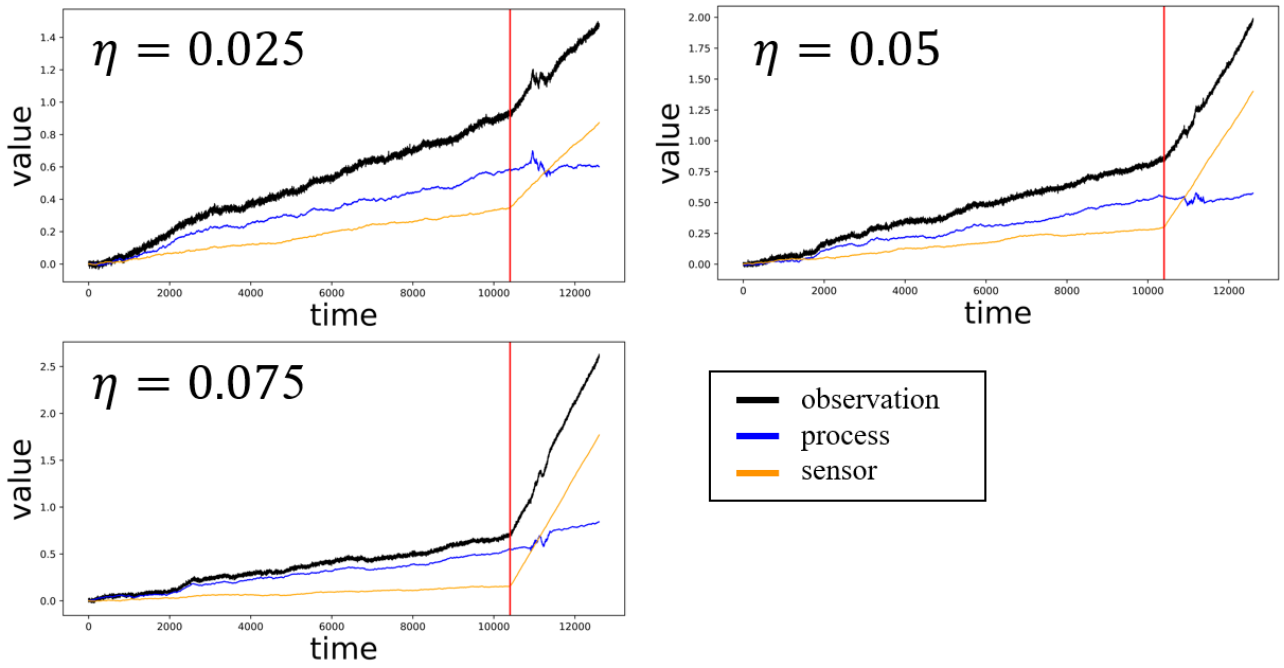


Figure 4.4 Experiment 2 data with different sensor drift levels
(sensor drift level: 0.025, 0.05, and 0.075).
The top line is the observation point, which consists of the process and sensor states below, and the red line represents the time point when the anomaly occurred

doi:10.6342/NTU202203045

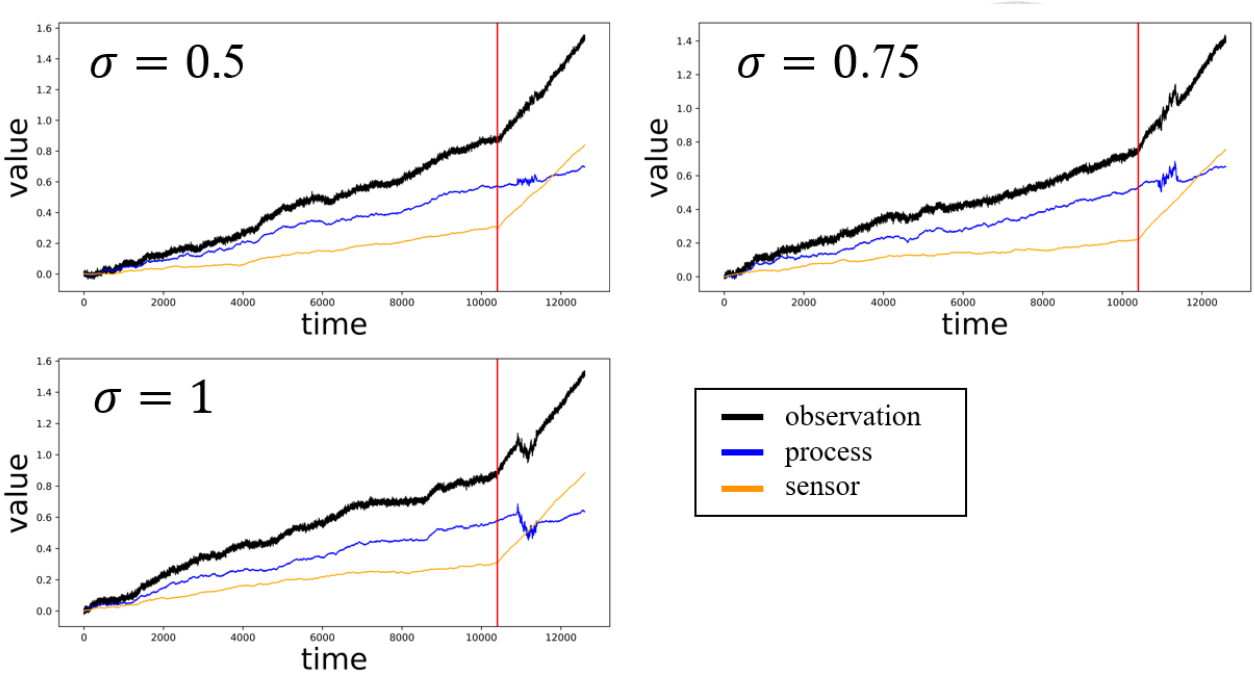


Figure 4.5 Experiment 2 data with different process drift levels

(process drift level: 0.5, 0.75, and 1).

The top line is the observation point, which consists of the process and sensor states below, and the red line represents the time point when the anomaly occurred

In the first stage we wanted to verify that the model can detect process anomalies and sensor anomalies well. In the second stage we wanted to demonstrate that the model can be robust enough in different situations, so we design different situations in the next two experiments to test whether the sensor anomaly can be successfully detected.

Experiment 3: In the third experiment, we wanted to confirm whether the proposed method can equally perform better than the other methods while the overall variation increases. Therefore, in this experiment, the usual process variation σ_0 and the usual sensor variation δ_0 will be doubled respectively, and the performance results under the three degrees of sensor drift level η will be viewed in the same way. In addition,

doi:10.6342/NTU202203045

this experiment is based on experiment 1, that is, the usual sensor drift level $\eta_0 = 0$. Figure 4.6 shows the actual data situation in experiment 3 under three different conditions.

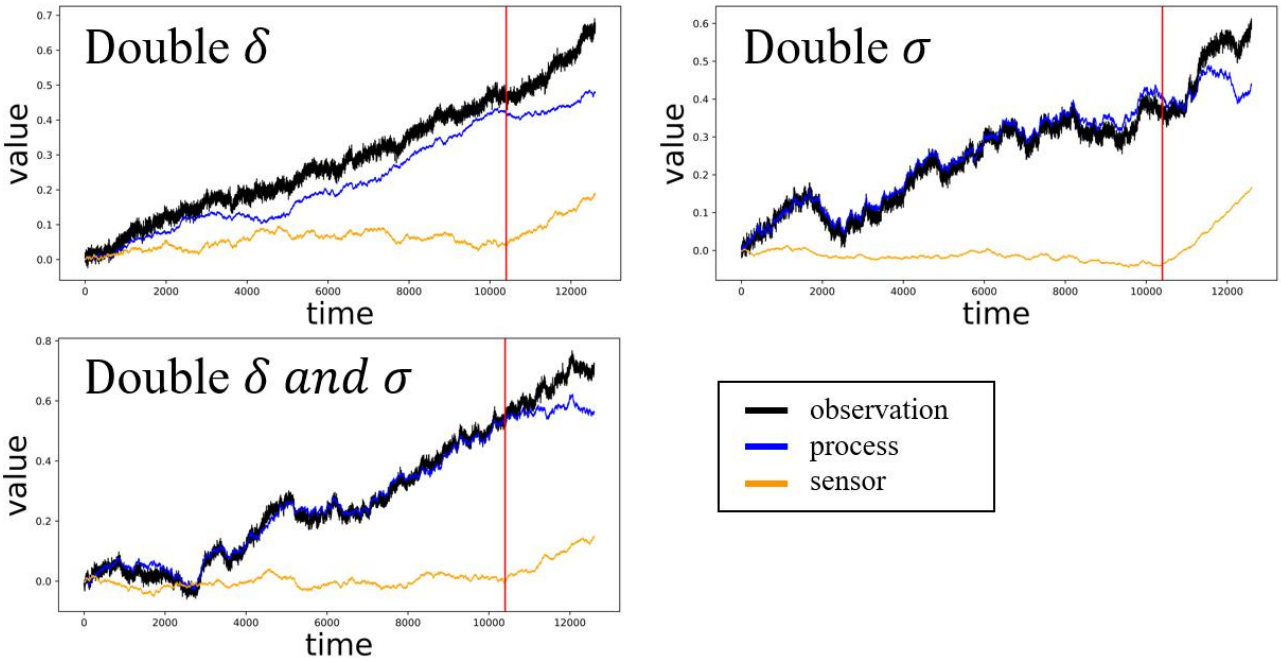


Figure 4.6 Experiment 3 data with different variation levels

The top line is the observation point, which consists of the process and sensor states below, and the red line represents the time point when the anomaly occurred

doi:10.6342/NTU202203045

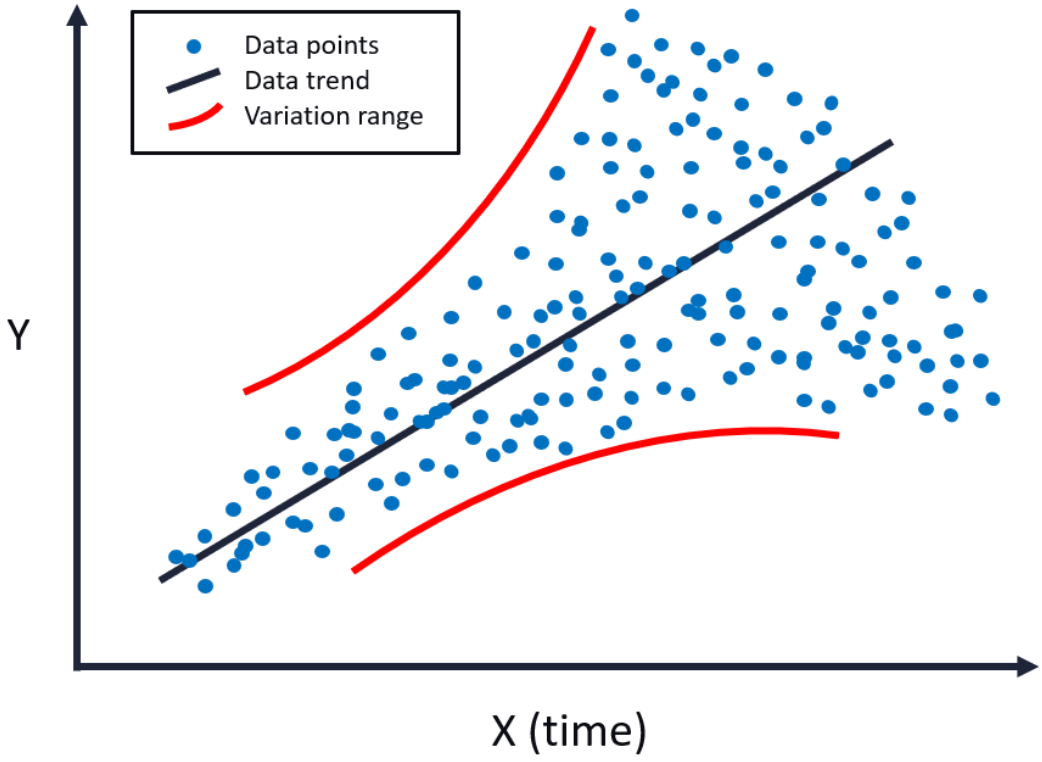


Figure 4.7 Data with heteroscedasticity

Experiment 4: In the fourth experiment, we wanted to explore the situation in which the model encountered *heteroscedasticity (*Cook & Weisberg, 1983*)* in the data. Heteroscedasticity is usually thought to occur when the sensor begins to drift. As shown in the Figure 4.7, heteroscedasticity refers to the situation in which when the mean of the data begins to shift, the variance of the data also begins to increase. Therefore, in this experiment, we will multiply the variation coefficient δ of the sensor by an additional time-dependent coefficient t to make its heteroscedasticity phenomenon as formulated below:

$$S(t) = S(0) + \eta t + \delta t B(t) \tag{10}$$

doi:10.6342/NTU202203045

At the same time, this experiment will be carried out under the assumptions of experiment 1 and experiment 2 respectively, called experiment 4-1 and 4-2. In the case of experiment 4-1, since there will be degradation only under abnormal conditions, there will be heteroscedasticity only when a sensor anomaly occurs. Figure 4.8 shows the actual data situation in experiment 4-1 for three different sensor drift parameters, from sensor drift level = 0.01, where the shift is least obvious, to sensor drift level = 0.02, where it is most obvious. On the other hand, In the case of experiment 4-2, the situation of heteroscedasticity occurs under normal conditions. Figure 4.9 shows the actual data situation in experiment 4-2 for three different sensor drift parameters, from sensor drift level = 0.025, where the shift is least obvious, to sensor drift level = 0.075, where it is most obvious.

This experiment wants to test whether the sensor anomaly can still be well detected and better than other methods in the case of heteroscedasticity.

doi:10.6342/NTU202203045

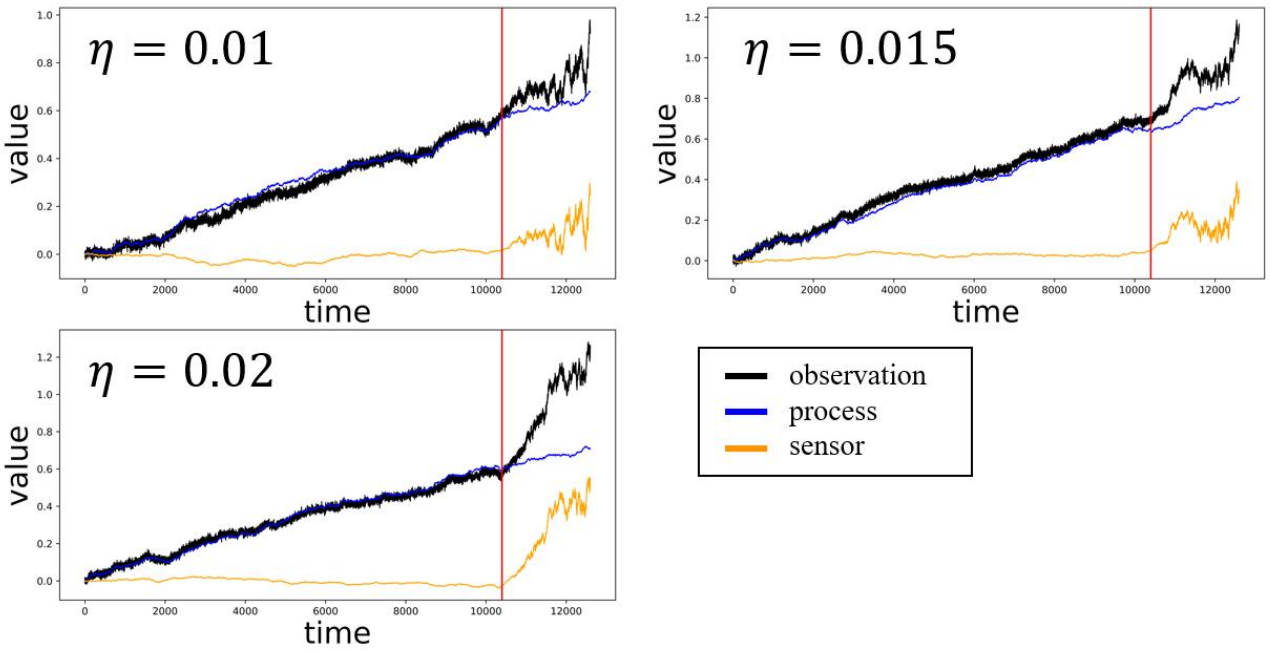


Figure 4.8 Experiment 4-1 data with different sensor drift levels

(sensor drift level: 0.01, 0.015, and 0.02).

The top line is the observation point, which consists of the process and sensor states below, and the red line represents the time point when the anomaly occurred

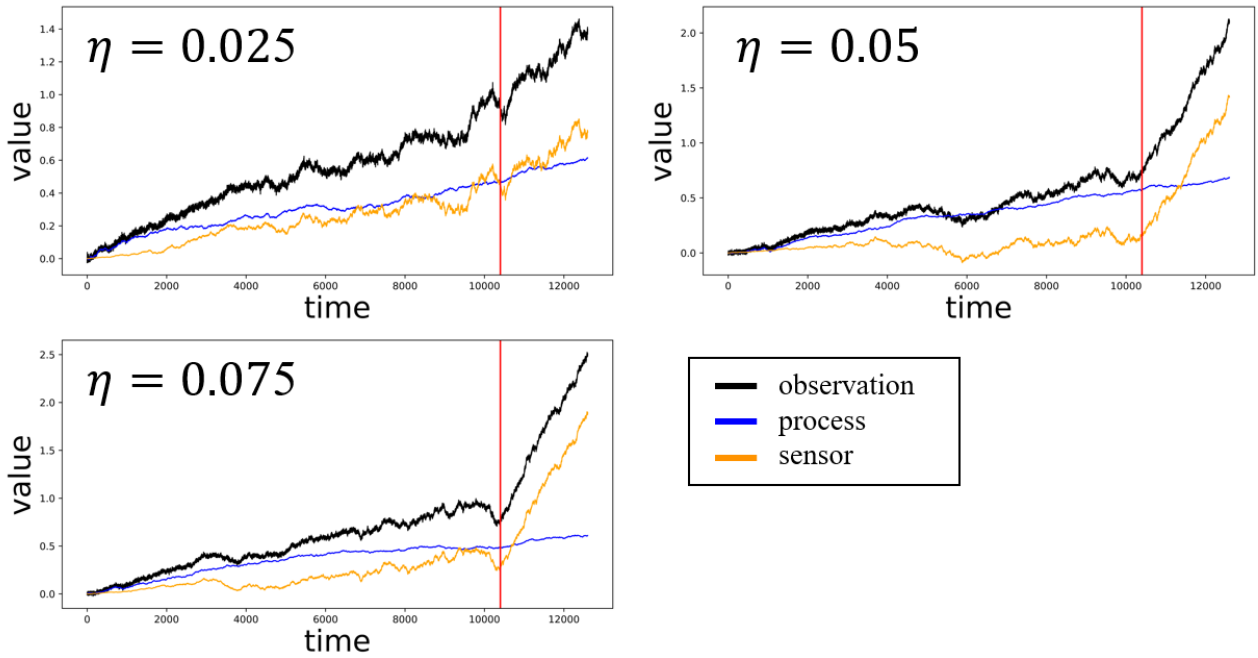


Figure 4.9 Experiment 4-2 data with different sensor drift levels

(sensor drift level: 0.025, 0.05, and 0.075).

The top line is the observation point, which consists of the process and sensor states below, and the red line represents the time point when the anomaly occurred

doi:10.6342/NTU202203045

In addition to the experimental design, we need a baseline for comparison to prove that this method is superior to other methods in the past and not because of insufficient data complexity. The baseline used in scenario 2 are the EWMA control chart and *Adaptive Windowing (ADWIN) (*Bifet & Gavalda, 2007*).*

Control charts are often used in the PHM field to monitor changes in machine values, and the EWMA chart is one of the most frequently used control charts. In the manufacturing industry, machine values at different time points are correlated. The advantage of the EWMA chart compared to other control charts is that it gives weight to past points, which makes each point consider the past trend. Therefore, EWMA control chart is a control chart that is very often used to monitor the health status of machines in the manufacturing industry, which is formulated in the following equation:

$$z_i = \lambda x_i + (1-\lambda) z_{i-1}. \quad (11)$$

x_i denotes the input of EWMA chart at the time point i where $x_0 = 0$. z_i represents the value of EWMA at time point i where $z_0 = \mu_0$. μ_0 is the average value of the EWMA of the training data. λ is a hyperparameter between 0 and 1, usually between 0.05 and 0.25. The upper control limits (UCL) and the lower control limits (LCL) are formulated in the following equation:

$$upper\ control\ limit\ (UCL) = \mu_0 + L\sigma\sqrt{(\frac{\lambda}{2-\lambda})[1-(1-\lambda)^{2i}]}\,. \quad (12)$$

doi:10.6342/NTU202203045

$$lower\ control\ limit\ (LCL) = \mu_0 - L\sigma\sqrt{(\frac{\lambda}{2-\lambda})[1-(1-\lambda)^{2i}]}\ . \quad (13)$$

The parameter σ denotes the standard deviation of the phase 1 data. The parameter L, which represents how many standard deviations the range of the control boundary is, is usually 3 or 4. When the EWMA value z_i is larger than the value of UCL or smaller than the value of LCL, the out-of-control data is considered to be an anomaly.

Because the data for Scenario 2 will continue to drift even under normal conditions, we make the EWMA control chart update the control limits every five points like our model, so that it does not happen that all normal channels are regarded as anomalies.

Another method to use as a baseline is ADWIN. ADWIN is another method that can be used to detect if the data is drifted. It uses a sliding window of variable size that can be recalculated online from the rate of change observed in the data in these windows. The algorithm automatically enlarges the window (*W*) when no significant changes are detected, and shrinks it when detected. The algorithm tries to find two sub-windows with different averages (*W*) to confirm that the current window is already based on a different distribution than the previous data. Figure 4.10 shows the output of the ADWIN algorithm under different degrees of change.

doi:10.6342/NTU202203045

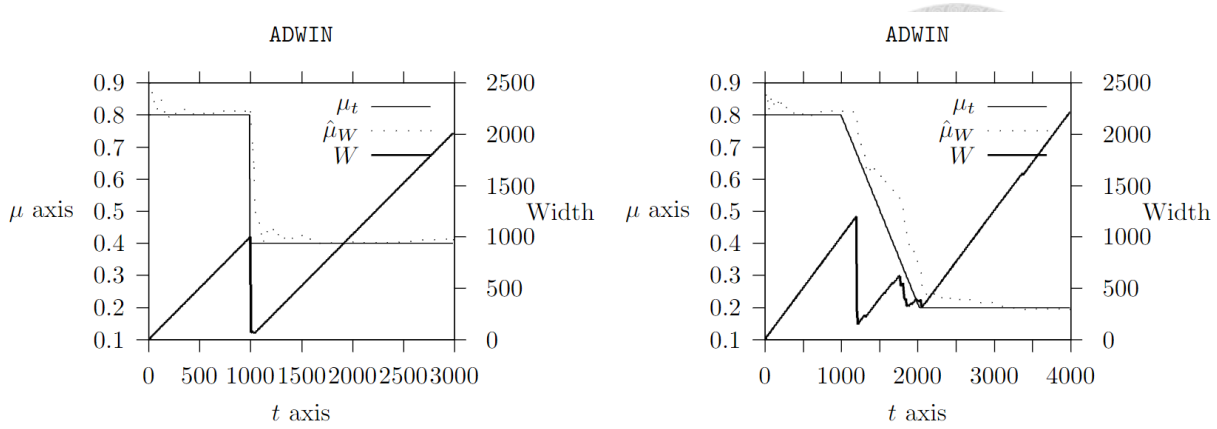


Figure 4.10 Output of algorithm ADWIN with abrupt and slow gradual change

(Bifet & Gavalda, 2007).

4.1.3 Model Configuration and Training Parameters

Figure 4.11 describe the architecture of the AE used as a reconstruction model for the proposed method. The autoencoder used in this study is relatively common in configuration. It is hoped that a good detection result can be achieved through a set of processes proposed in this study, rather than relying solely on powerful deep learning models. The model information is shown in Table 4.2. The parameters in the model information are determined by Grid Search in advance to determine the most suitable parameters. During the training process, it is also ensured that the training loss is low enough and no overfitting occurs to ensure that the model has good reconstruction results.

doi:10.6342/NTU202203045

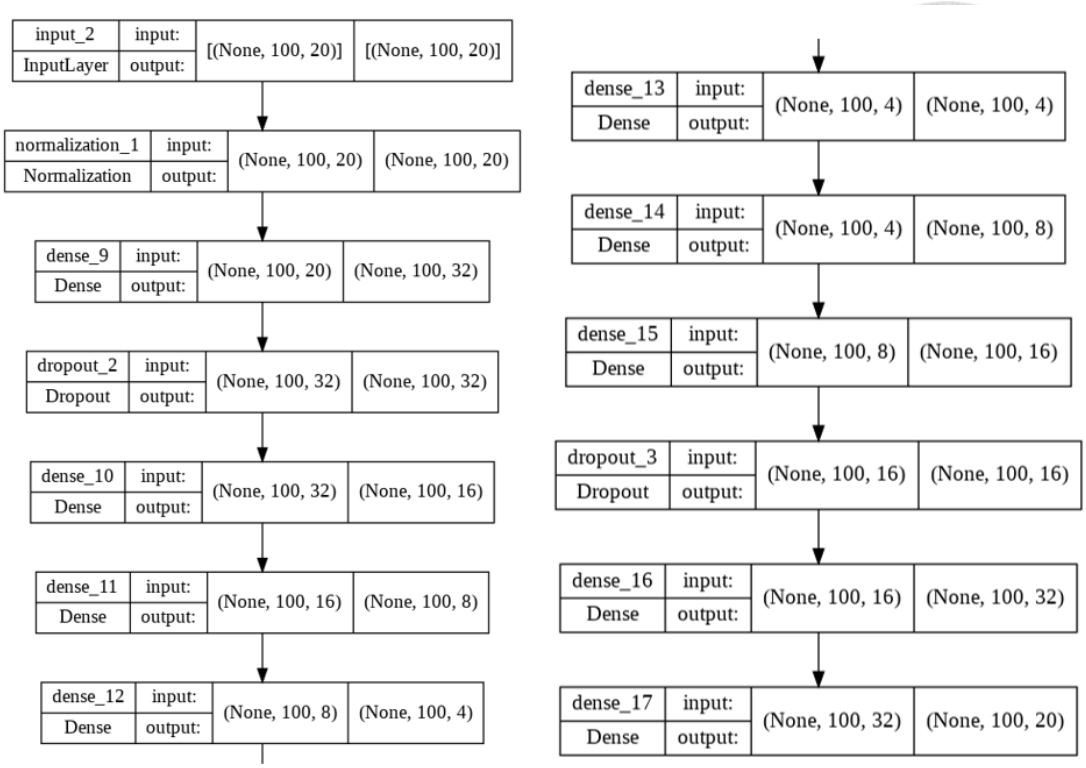


Figure 4.11 AE model architecture

Table 4.2 AE model information

Model Info.	Model	Autoencoder
	Loss function	Mean Absolute Error
	Parameter	# of Iterations = 25 (early stopping patience = 5)
		Learning Rate = 0.001
		Optimizer Adam

4.1.4 Evaluation Metrics

To appropriately evaluate the models, we used F1-score (F1) as our evaluation metrics. F1 is often used as a good measure of accuracy, which is calculated as:

$$\mathrm{F1} = \frac{2 \times precision \times recall}{precision + recall}. \quad (14)$$

doi:10.6342/NTU202203045

where recall is the proportion of positive observations that are correctly detected, and precision is the proportion of positive observations to all of the predicted observations (i.e., recall = TP/(TP + FN) and precision = TP/(TP + FP), where TP is the number of true positives, FN is the number of false negatives, and FP is the number of false positive).

4.2 Result & Discussion

To evaluate the anomaly detection performance for each scenario and experiment, we will repeat each experiment 30 times and calculate the mean and the standard deviation of F1 and compare with the respective baselines. In order to distinguish normal and abnormal points, in the experiment of this study, the AE model and EWMA control chart rule for judging anomaly: if the trend exceeds the control for 5 consecutive points and continues to rise, it will be judged as sensor anomaly, and the residual exceeds the control for 3 consecutive points, it will be judged as process anomaly. These rules can be adjusted according to different fields through different practical experience. As for the update control limits part, the control limits are updated every 10 points because the data of Scenario 1 changes slowly. For more complex and rapidly changing data in Scenario 2, the boundary is updated every 5 points.

doi:10.6342/NTU202203045

4.2.1 Scenario 1

Table 4.3 Experiment results of scenario 1.

Variation Level σ	Process Drift Level	Sensor Drift Level	AE Trend F1-score	AE Residual F1-score	iForest Trend F1-score	iForest Residual F1-score
1	2	1.0002	**0.97 (0.05)**	1.00 (0.02)	0.78 (0.09)	1.00 (0.00)
1	2	1.0003	**0.97 (0.05)**	1.00 (0.00)	0.74 (0.08)	1.00 (0.00)
1	2	1.0005	**0.97 (0.05)**	1.00 (0.00)	0.75 (0.08)	1.00 (0.00)
1	2	1.001	**0.96 (0.08)**	1.00 (0.00)	0.75 (0.07)	1.00 (0.00)
1	3	1.0002	**0.96 (0.06)**	1.00 (0.02)	0.73 (0.08)	1.00 (0.00)
1	3	1.0003	**0.95 (0.06)**	1.00 (0.00)	0.75 (0.08)	1.00 (0.00)
1	3	1.0005	**0.96 (0.06)**	1.00 (0.00)	0.75 (0.08)	1.00 (0.00)
1	3	1.001	**0.94 (0.06)**	1.00 (0.00)	0.75 (0.08)	1.00 (0.00)

Variation Level σ	Process Drift Level	Sensor Drift Level	AE Trend F1-score	AE Residual F1-score	iForest Trend F1-score	iForest Residual F1-score
1.5	2	1.0002	**1.00 (0.00)**	1.00 (0.00)	0.77 (0.09)	0.96 (0.08)
1.5	2	1.0003	**1.00 (0.00)**	1.00 (0.00)	0.79 (0.08)	0.93 (0.12)
1.5	2	1.0005	**1.00 (0.02)**	1.00 (0.02)	0.77 (0.09)	0.97 (0.10)
1.5	2	1.001	**0.98 (0.04)**	1.00 (0.00)	0.80 (0.09)	0.95 (0.08)
1.5	3	1.0002	**0.99 (0.04)**	1.00 (0.02)	0.75 (0.07)	1.00 (0.00)
1.5	3	1.0003	**0.99 (0.03)**	1.00 (0.00)	0.76 (0.07)	1.00 (0.00)
1.5	3	1.0005	**0.99 (0.03)**	1.00 (0.00)	0.76 (0.11)	1.00 (0.00)
1.5	3	1.001	**0.97 (0.06)**	1.00 (0.00)	0.74 (0.08)	1.00 (0.00)

Table 4.3 shows the experiment results of scenario 1. It can be clearly seen that the method proposed in this paper performs very well in both sensor anomalies and process anomalies. On the other hand, although iForest performs well in the part of

doi:10.6342/NTU202203045

process anomalies, the trend items representing sensor anomalies are still not well detected. This may mean that iForest is less sensitive to anomalies that gradually produce small changes. Figure 4.12 and Figure 4.13 show the actual monitoring situation for normal and abnormal channels in residual and trend terms. This example is to retrain the model every 10 points, and it can be seen that after the model is retrained over time, the red control limit also changes dynamically. In practice, people can easily see which channel has a problem with these control charts and make immediate maintenance decisions.

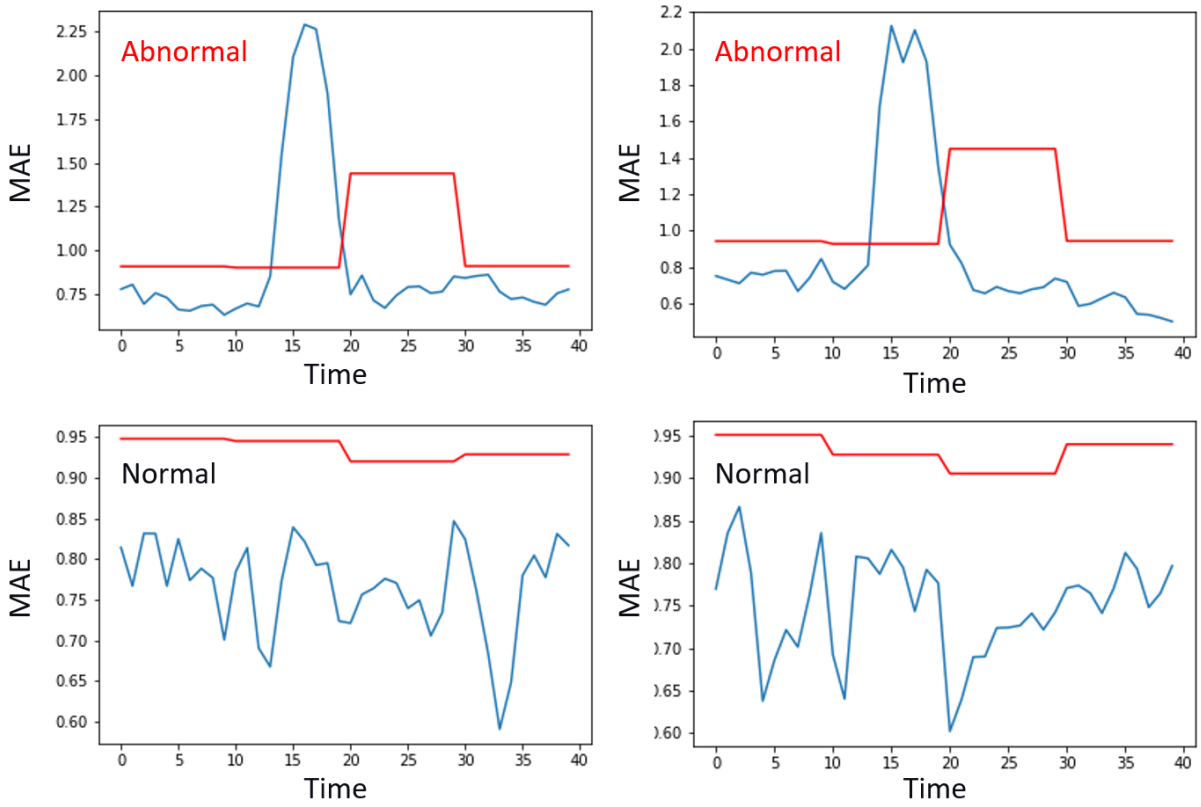


Figure 4.12 Monitoring situations for normal and abnormal channels in residual terms

doi:10.6342/NTU202203045

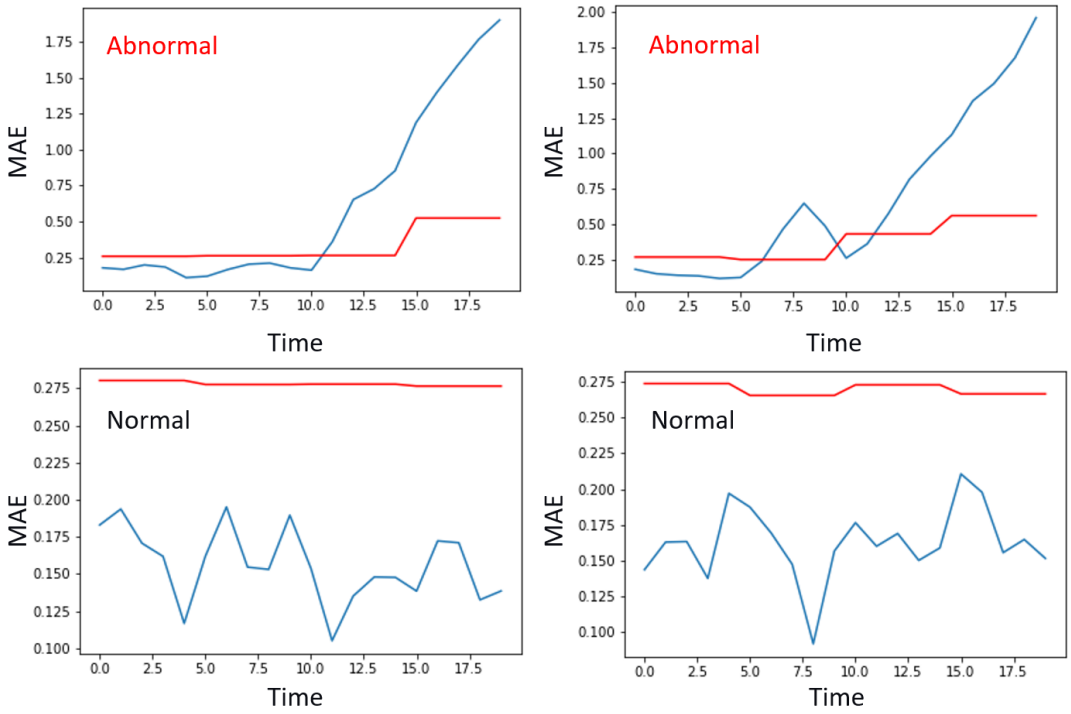


Figure 4.13 Monitoring situations for normal and abnormal channels in trend terms

4.2.2 Scenario 2

Table 4.4 and Table 4.5 show the result of experiment 1 and experiment 2. In the trend part, it can be seen that the detection result will decrease slightly when the sensor drift level becomes smaller, but it is still a good result compared to the baselines. In the residual part, although the detection results of ADWIN are not bad, the method proposed in this study is still relatively good. It can be seen from these two experiments that the proposed method can effectively detect whether the sensor will degrade under normal conditions. It also means that no matter when the detection target is slow degradation or severe degradation, it can be solved by this method.

doi:10.6342/NTU202203045

Table 4.4 Results of experiment 1

Process Drift Level	Sensor Drift Level	AE Trend F1-score	AE Residual F1-score	EWMA Trend F1-score	EWMA Residual F1-score	ADWIN Trend F1-score	ADWIN Residual F1-score
0.5	0.01	**0.84 (0.11)**	**0.96 (0.08)**	0.43 (0.07)	0.47 (0.00)	0.39 (0.04)	0.84 (0.07)
0.5	0.015	**0.88 (0.10)**	**0.92 (0.12)**	0.41 (0.03)	0.47 (0.00)	0.40 (0.07)	0.83 (0.08)
0.5	0.02	**0.91 (0.09)**	**0.92 (0.14)**	0.41 (0.02)	0.47 (0.00)	0.42 (0.05)	0.85 (0.06)
0.75	0.01	**0.85 (0.13)**	**0.99 (0.03)**	0.44 (0.07)	0.53 (0.13)	0.39 (0.03)	0.85 (0.07)
0.75	0.015	**0.85 (0.08)**	**0.99 (0.04)**	0.44 (0.07)	0.50 (0.09)	0.40 (0.06)	0.87 (0.06)
0.75	0.02	**0.92 (0.07)**	**0.99 (0.03)**	0.44 (0.09)	0.57 (0.15)	0.40 (0.05)	0.86 (0.06)
1	0.01	**0.90 (0.10)**	**0.99 (0.04)**	0.44 (0.07)	0.62 (0.19)	0.40 (0.03)	0.86 (0.06)
1	0.015	**0.90 (0.09)**	**1.00 (0.00)**	0.42 (0.07)	0.61 (0.17)	0.40 (0.04)	0.85 (0.08)
1	0.02	**0.91 (0.10)**	**0.99 (0.04)**	0.42 (0.06)	0.65 (0.18)	0.40 (0.06)	0.84 (0.07)

Table 4.5 Results of experiment 2

Process Drift Level	Sensor Drift Level	AE Trend F1-score	AE Residual F1-score	EWMA Trend F1-score	EWMA Residual F1-score	ADWIN Trend F1-score	ADWIN Residual F1-score
0.5	0.025	**0.96 (0.07)**	**0.92 (0.09)**	0.46 (0.01)	0.47 (0.00)	0.42 (0.03)	0.83 (0.07)
0.5	0.05	**0.95 (0.07)**	**0.90 (0.12)**	0.45 (0.02)	0.47 (0.00)	0.45 (0.07)	0.87 (0.06)
0.5	0.075	**0.93 (0.07)**	**0.90 (0.04)**	0.45 (0.02)	0.47 (0.00)	0.53 (0.09)	0.86 (0.05)
0.75	0.025	**0.90 (0.07)**	**0.99 (0.04)**	0.47 (0.07)	0.51 (0.10)	0.43 (0.04)	0.84 (0.07)
0.75	0.05	**0.92 (0.08)**	**1.00 (0.00)**	0.45 (0.02)	0.50 (0.09)	0.44 (0.06)	0.87 (0.05)
0.75	0.075	**0.95 (0.07)**	**0.99 (0.04)**	0.46 (0.01)	0.50 (0.09)	0.49 (0.08)	0.83 (0.08)
1	0.025	**0.92 (0.09)**	**0.98 (0.04)**	0.49 (0.08)	0.56 (0.16)	0.41 (0.04)	0.85 (0.07)
1	0.05	**0.93 (0.09)**	**0.98 (0.04)**	0.46 (0.01)	0.67 (0.17)	0.45 (0.07)	0.85 (0.07)
1	0.075	**0.92 (0.07)**	**0.98 (0.04)**	0.46 (0.02)	0.64 (0.18)	0.50 (0.08)	0.87 (0.04)

doi:10.6342/NTU202203045

Table 4.6 shows the result of experiment 3. It can be seen that most of the detection results start to drop after the variation doubles, but the proposed method is still better than the baselines. In addition, after the sensor drift level is increased, it can still maintain a good result.

Table 4.6 Results of experiment 3

Process Variation σ_0	Sensor Variation δ_0	Sensor Drift Level	AE Trend F1-score	EWMA Trend F1-score	ADWIN Trend F1-score
0.1	**0.1**	0.01	**0.86 (0.13)**	0.38 (0.06)	0.36 (0.04)
0.1	**0.1**	0.015	**0.88 (0.12)**	0.37 (0.04)	0.38 (0.03)
0.1	**0.1**	0.02	**0.90 (0.10)**	0.38 (0.04)	0.40 (0.04)
0.2	0.05	0.01	**0.71 (0.19)**	0.32 (0.07)	0.36 (0.05)
0.2	0.05	0.015	**0.84 (0.10)**	0.28 (0.07)	0.37 (0.07)
0.2	0.05	0.02	**0.88 (0.09)**	0.29 (0.05)	0.38 (0.08)
0.2	**0.1**	0.01	**0.66 (0.17)**	0.31(0.07)	0.36 (0.06)
0.2	**0.1**	0.015	**0.75 (0.20)**	0.28 (0.06)	0.38 (0.08)
0.2	**0.1**	0.02	**0.83 (0.15)**	0.29 (0.06)	0.36 (0.05)

Parameters in bold red represent doubled variation.

Table 4.7 and Table 4.8 shows the result of experiment 4-1 and 4-2. In experiment 4-1, when the sensor has sensor anomaly, it will be accompanied by the phenomenon of heteroscedasticity. It can be seen from the experimental results that the proposed method will not be affected by this, and still maintains a good prediction result. On the other hand, EWMA was affected and the accuracy began to decline.

doi:10.6342/NTU202203045

In Experiment 4-2, under normal conditions, the sensor will drift and accompanied by the phenomenon of heteroscedasticity, which makes the overall prediction difficulty much higher, but it can be seen that the proposed method still performs better than the baselines. In addition, like the results of the previous experiments, when the degree of sensor drift level increases, the predicted results still have a certain level.

Table 4.7 Results of experiment 4-1

Sensor Drift Level	AE Trend F1-score	EWMA Trend F1-score	ADWIN Trend F1-score
0.01	**0.83 (0.15)**	0.35 (0.10)	0.36 (0.07)
0.015	**0.89 (0.09)**	0.32 (0.08)	0.38 (0.08)
0.02	**0.88 (0.09)**	0.28 (0.08)	0.40 (0.07)

Table 4.8 Results of experiment 4-2

Sensor Drift Level	AE Trend F1-score	EWMA Trend F1-score	ADWIN Trend F1-score
0.025	**0.64 (0.15)**	0.23 (0.09)	0.34 (0.08)
0.05	**0.70 (0.07)**	0.20 (0.06)	0.33 (0.08)
0.075	**0.74 (0.10)**	0.19 (0.06)	0.36 (0.08)

doi:10.6342/NTU202203045

Chapter 5. Conclusion and Future Research

5.1 Conclusion

This study proposes the PSAD framework that can divide data into two perspectives for detection. Compared with the past, as long as an anomaly is encountered, it is considered to be a process problem of the machine. This method can decide whether to maintain the sensor or the machine itself through a series of processing. On the other hand, this method achieves the effect of online control by performing model retraining with the update of data to obtain new control limits.

As far as the model itself is concerned, it can be seen from the experimental results that no matter whether the sensor is degraded at ordinary times, or in the case of high variation and heteroscedasticity, it can be seen that the method can achieve certain results. This means that the method can have good effectiveness and robustness while monitoring multiple channels at the same time. If the model is fine-tuned with appropriate domain knowledge according to different fields, it can be very helpful for PHM in various fields of manufacturing.

5.2 Future Study

In the method proposed in this study, there are still many directions that can be tried in the future. For example, this research does not focus on the model itself, there

doi:10.6342/NTU202203045

are many more complex deep learning models that can be used, and different models can be selected according to different data characteristics.

In terms of data, the actual data may be much more complicated. At this time, there may be hidden interactions between multiple channels. These are the directions that can be extended in the future. The important thing is that in the field of PHM, there are still many issues related to sensor anomalies that have not yet been clarified. With the progress of related research, it will definitely have a huge contribution to practice.

References

An, J., & Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, *2*(1), 1-18.

Bifet, A., & Gavalda, R. (2007). Learning from time-changing data with adaptive windowing. Proceedings of the 2007 SIAM international conference on data mining,

Blue, J., Gleispach, D., Roussy, A., & Scheibelhofer, P. (2012). Tool condition diagnosis with a recipe-independent hierarchical monitoring scheme. *IEEE Transactions on Semiconductor Manufacturing*, *26*(1), 82-91.

Blue, J., Roussy, A., & Pinaton, J. (2014). FDC R2R variation monitoring for sensor level diagnosis in tool condition hierarchy. 25th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC 2014),

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF: identifying density-based local outliers. Proceedings of the 2000 ACM SIGMOD international conference on Management of data,

doi:10.6342/NTU202203045

Chen, A., & Blue, J. (2009). Recipe-independent indicator for tool health diagnosis and predictive maintenance. *IEEE Transactions on Semiconductor Manufacturing*, *22*(4), 522-535.

Chen, C.-C., Ting, W.-J., & Chien, C.-F. (2019). Big Data Analytics for Tool Health Monitoring in Panel Industry. 2019 IEEE International Conference on Smart Manufacturing, Industrial & Logistics Engineering (SMILE),

Cleveland, R. B., Cleveland, W. S., McRae, J. E., & Terpenning, I. (1990). STL: A seasonal-trend decomposition. *J. Off. Stat*, *6*(1), 3-73.

Cook, R. D., & Weisberg, S. (1983). Diagnostics for heteroscedasticity in regression. *Biometrika*, *70*(1), 1-10.

Johnson, R. A., & Wichern, D. W. (2014). *Applied multivariate statistical analysis* (Vol. 6). Pearson London, UK:.

Keogh, E., Chu, S., Hart, D., & Pazzani, M. (2004). Segmenting time series: A survey and novel approach. In *Data mining in time series databases* (pp. 1-21). World Scientific.

Kwak, M., & Kim, S. B. (2021). Unsupervised Abnormal Sensor Signal Detection With Channelwise Reconstruction Errors. *IEEE Access*, *9*, 39995-40007. https://doi.org/10.1109/access.2021.3064563

Lee, C.-Y., Wu, C.-S., & Hung, Y.-H. (2020). In-line predictive monitoring framework. *IEEE Transactions on Automation Science and Engineering*.

Liu, B., Do, P., Iung, B., & Xie, M. (2019). Stochastic filtering approach for condition-based maintenance considering sensor degradation. *IEEE Transactions on Automation Science and Engineering*, *17*(1), 177-190.

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation forest. 2008 eighth ieee international conference on data mining,

doi:10.6342/NTU202203045

Lucas, J. M. (1985). Counted data CUSUM's. *Technometrics*, *27*(2), 129-144.

Meng, H., & Li, Y.-F. (2019). A review on prognostics and health management (PHM) methods of lithium-ion batteries. *Renewable and Sustainable Energy Reviews*, *116*. https://doi.org/10.1016/j.rser.2019.109405

Mobley, R. K. (2002). *An introduction to predictive maintenance*. Elsevier.

Montgomery, D. C. (2020). *Introduction to statistical quality control*. John Wiley & Sons.

Morales, V. H., Panza, C. A., & Blanco, J. (2021). Monitoring the Nonconforming Fraction with a Dynamic Scheme When Sample Sizes are Time-varying.

Ng, A. (2011). Sparse autoencoder. *CS294A Lecture notes*, *72*(2011), 1-19.

Sakurada, M., & Yairi, T. (2014). Anomaly detection using autoencoders with nonlinear dimensionality reduction. Proceedings of the MLSDA 2014 2nd workshop on machine learning for sensory data analysis,

Schölkopf, B., Williamson, R. C., Smola, A. J., Shawe-Taylor, J., & Platt, J. C. (1999). Support vector method for novelty detection. NIPS,

Shen, X., Zou, C., Jiang, W., & Tsung, F. (2013). Monitoring poisson count data with probability control limits when sample sizes are time varying. *Naval Research Logistics (NRL)*, *60*(8), 625-636. https://doi.org/10.1002/nav.21557

Tighkhorshid, E., Amiri, A., & Amirkhani, F. (2019). A risk-adjusted EWMA chart with dynamic probability control limits for monitoring survival time. *Communications in Statistics-Simulation and Computation*, 1-22.

West, M. (1997). Time series decomposition. *Biometrika*, *84*(2), 489-494.

doi:10.6342/NTU202203045

Wold, S., Johansson, E., & Cocchi, M. (1993). PLS: partial least squares projections to latent structures.

Ye, Z. S., & Xie, M. (2015). Stochastic modelling and analysis of degradation for highly reliable products. *Applied Stochastic Models in Business and Industry*, *31*(1), 16-32.

Yeh, A. B., Lin, D. K., & McGrath, R. N. (2006). Multivariate control charts for monitoring covariance matrix: a review. *Quality Technology & Quantitative Management*, *3*(4), 415-436.

Zhai, Q., & Ye, Z.-S. (2017). RUL prediction of deteriorating products using an adaptive Wiener process model. *IEEE Transactions on Industrial Informatics*, *13*(6), 2911-2921.

Zhang, X., & Woodall, W. H. (2015). Dynamic probability control limits for risk-adjusted Bernoulli CUSUM charts. *Stat Med*, *34*(25), 3336-3348. https://doi.org/10.1002/sim.6547

doi:10.6342/NTU202203045

About the Author

Kai Chang received the B.S. degree in Management Information Systems from National Chengchi University, Taipei, Taiwan, in 2020, and the M.S. degree in Information Management from National Taiwan University, Taipei, Taiwan, in 2022.

His research interests are manufacturing data science which includes data mining, machine learning and deep learning. He was a research assistant in Productivity Optimization Laboratory from July 2020 to August 2022.

doi:10.6342/NTU202203045